1980

MOS

DATABOOK

National Semiconductor

# MOS
# DATABOOK

---

# NATIONAL
# SEMICONDUCTOR

# MOS
# DATABOOK

## Introduction:

This is the MOS Data Book from National Semiconductor. It contains information on products fabricated from all the MOS processes in high volume production today; NMOS, CMOS, and PMOS. These products are unique functions in electronic systems. Their common denominator is providing the most cost-effective solution to a system need.

To minimize system costs, many of these products have customized features, such as the COPS™ family of single chip micro-controllers, MAXI-ROMS™ and Custom Circuits. Each of these has minimum order quantities to maintain production efficiencies.

A large part of this book is Standard Circuits which, due to large scale integration, have incorporated features which make many of them unique and cost effective in specific applications and/or markets. The organization of this book highlights those areas. The Standards have no minimum quantities.

Of concern to everyone who uses LSI products is the quality and reliability levels of the product. National is careful not to cut corners in this respect, designing in both quality and reliability from the ground up. From quality control of the raw materials through design engineering, wafer fab, test and assembly, the emphasis is uniformly there and we are proud of the results. Additional information may be obtained through your local National Semiconductor Sales Office.

Other large scale integration MOS product lines not included in this book are RAM and PROM memories and Microprocessors, which are covered in other National Semiconductor Data Books



**Introduction to COPS™**

**COPS Devices** 1

**COPS Application Information** 2

**MOS/ROMs** 3

**Speech Synthesis** 4

**Standard MOS/LSI** 5
Digital Clocks
Television/Radio
Games/Calculators
Telecommunications
Display Drivers
Oscillators
Electronic Data Processing

**Application Notes** 6

**Custom MOS/LSI** 7

**Ordering Information/ Physical Dimensions** 8

# Table of Contents


**Section 1 COPS Devices**

COPS™ Selection Guide ..... 1-3
COP402/COP402M ROMless N-Channel Microcontrollers ..... 1-5
COP404L ROMless N-Channel Microcontroller ..... 1-24
COP410L/COP411L Single-Chip N-Channel Microcontrollers ..... 1-40
COP420/COP421 and COP320/COP321 Single-Chip N-Channel Microcontrollers ..... 1-56
COP420C/COP421C and COP320C/COP321C Single-Chip CMOS Microcontrollers ..... 1-79
COP420L/COP421L Single-Chip N-Channel Microcontrollers ..... 1-96
COP444L/COP445L Single-Chip N-Channel Microcontrollers ..... 1-115
COP430/COP431/COP432 A to D Converters ..... 1-134
COP450/COP451 PROM-RAM Interface Chip ..... 1-136
COP452 Frequency/Counter Peripheral ..... 1-138
COP470 V.F. Display Driver ..... 1-140
COP472 Liquid Crystal Display Controller ..... 1-147
COP498 Low Power CMOS RAM and Timer ..... 1-153
COP499 Low Power CMOS Memory ..... 1-155

**Section 2 COPS Application Information**

COP400 COPS™ Family User's Guide ..... 2-3
COP Note 1 Analog to Digital Conversion Techniques with COPs Family Microcontrollers ..... 2-84
COP Note 2 COP Television Controller ..... 2-119
COP Brief 1 SIO Input/Output Register Description ..... 2-124
COP Brief 2 Easy Logarithms for COP400 ..... 2-126
COP Brief 3 Use of Macro-Assembled Code ..... 2-137
COP Brief 4 L-Bus Considerations ..... 2-139
COP Brief 5 Software and Opcode Differences in the COP444L Instruction Set ..... 2-140
COP Brief 6 RAM Keep-Alive ..... 2-141
COP Brief 7 Microbus Programming Considerations ..... 2-142
COP Brief 8 COPS Peripheral Chips ..... 2-143
COP Brief 9 Serial Interface Between COPS Microcontrollers and Peripheral Chips ..... 2-145
COP Brief 10 COP410L/COP411L Hardware Subroutine Stack Emulator ..... 2-147
COP Brief 11 Power Seat with Memory ..... 2-151
COP Brief 12 An Automotive Diagnostics Display ..... 2-155
COP Brief 13 An Electronic Speedometer and Odometer with Permanent Mileage Accumulation ..... 2-157

**Section 3 MOS/ROMs**

MM52116(2316E) 16,384-Bit (2048 × 8) Read Only Memory ..... 3-3
MM52116FDW, MM52116FDX Character Generators ..... 3-6
MM52132 32,768-Bit (4096 x 8) MAXI-ROM™ ..... 3-13
MM52164 65,536-Bit (8192 x 8) MAXI-ROM™ ..... 3-16
MM52264 65,536-Bit (8192 × 8) Clocked MAXI-ROM™ ..... 3-19

**Section 4 Speech Synthesis**

DIGITALKER™ Speech Synthesis System ..... 4-3

# Table of Contents (continued)

**Section 5 STANDARD MOS/LSI**

Digital Clock Product Selection Guide ..... 5-3
Television/Radio Product Selection Guide ..... 5-4
Games/Calculators Product Selection Guide ..... 5-5
Telecommunication Device Product Selection Guide ..... 5-6
Display Driver Product Selection Guide ..... 5-7
Oscillator/Divider Product Selection Guide ..... 5-8
Electronic Data Processing Product Selection Guide ..... 5-9

**Digital Clocks**

MM5309, MM5311, MM5312, MM5313, MM5314, MM5315 Digital Clocks ..... 5-13
MM5316 Digital Alarm Clocks ..... 5-20
MM5387AA, MM53108 Digital Alarm Clocks ..... 5-25
MM53110 Series Auto Clock and Elapsed Timer ..... 5-31
MM53113 Digital Alarm Clock ..... 5-36
MM53124 Automobile Clock and Elapsed Timer ..... 5-41
MM53224 Automobile Clock and Elapsed Timer ..... 5-45
MM5402, MM5405 Digital Alarm Clocks ..... 5-49
MM5406 Delux Display and Clock Radio ..... 5-54
MM5407 Digital Thermometer ..... 5-61
MM5455 Digital Alarm Clock ..... 5-66
MM5456, MM5457 Digital Alarm Clocks ..... 5-70
MM58143, MM58144, MM58183, MM58184 LCD Alarm Clock Circuits ..... 5-74
MM7317B, MM7318B Alarm Clock Calendar ..... 5-81

**Television/Radio**

MM5321 TV Camera Sync Generator ..... 5-91
MM5322 Color Bar Generator Chip ..... 5-97
MM53100, MM53105 Programmable TV Timers ..... 5-102
MM53118AA TV Digital Tuning ..... 5-107
MM5430, MM5431 AM/FM Radio Frequency Display ..... 5-115
MM5439 Microprocessor Compatible Phase Lock Loop (PLL) ..... 5-120
MM55108, MM55110 PLL Frequency Synthesizer with Receive/Transmit Mode ..... 5-127
MM55121 Serial Data/PLL Frequency Synthesizer ..... 5-132
MM55122 Serial Data/PLL Frequency Synthesizer ..... 5-137
MM55123 Serial Data/PLL Frequency Synthesizer ..... 5-142
MM55124, MM55126 PLL Frequency Synthesizer ..... 5-147
MM5837 Digital Noise Source ..... 5-151
MM5840 TV Channel Number (16-Channel) and Time Display Circuit ..... 5-153
MM58106 Digital Clock and TV Display Circuit ..... 5-158
MM58142 TV Synthesizer ..... 5-163
MM58146 TV Clock and Channel Display ..... 5-168
MM58313 Varactor Tuner Display Circuit ..... 5-173

# Table of Contents (continued)

**Section 5 STANDARD MOS/LSI** (continued)

**Games/Calculators**

MM5780 Educational Arithmetic Game ..... 5-181
MM57455 Advanced Educational Arithmetic Game ..... 5-187
MM57459 8-Digit LED Direct-Drive Memory Calculator ..... 5-190

**Telecommunications**

MM5393, MM5394, MM53143, MM53144 Push Button Pulse Dialer Circuits ..... 5-197
MM5395, MM53125 DTMF (Touch Tone® ) Generators ..... 5-202
MM53130 DTMF (Touch Tone® ) Generator ..... 5-207
MM53190 Push-Button Pulse Dialer ..... 5-212

**Display Drivers**

MM5445, MM5446, MM5447, MM5448 VF Display Drivers ..... 5-219
MM5450, MM5451 LED Display Drivers ..... 5-222
MM5452, MM5453 Liquid Crystal Display Drivers ..... 5-227
MM5480 LED Display Driver ..... 5-232
MM5481 LED Display Driver ..... 5-235
MM58201 Multiplexed LCD Driver ..... 5-238

**Oscillators**

MM5368 CMOS Oscillator Divider Circuit ..... 5-245
MM5369 Series 17 Stage Mask Programmable Oscillator/Divider ..... 5-248
MM53107 Series 17-Stage Oscillator/Divider ..... 5-251

**Electronic Data Processing**

MM5034, MM5035 Octal 80-Bit Static Shift Register ..... 5-257
MM5303 Universal Fully Asynchronous Receiver/Transmitter ..... 5-260
MM5307 Baud Rate Generator/Programmable Divider ..... 5-266
MM5330 4½-Digit Panel Meter Logic Block ..... 5-271
MM53200 Encode/Decoder ..... 5-278
MM54240 Asynchronous Receiver/Transmitter Remote Controller ..... 5-280
MM57109 MOS/LSI Number-Oriented Processor ..... 5-284
MM57436 Decimal/Binary Up/Down Counter ..... 5-309
MM57499 96- or 144-Key Serial Keyboard Interface (SKI) ..... 5-316
MM5863 12-Bit Binary A/D Building Block ..... 5-331
MM5865 Universal Timer ..... 5-337
MM58167 Microprocessor Compatible Real Time Clock ..... 5-347
MM58174 Microprocessor-Compatible Real Time Clock ..... 5-353

Touch Tone® is a Registered Trademark of Bell Telephone

# Table of Contents (continued)

**Section 6 Application Notes**

AN143 Using National Clock Integrated Circuit in Timer Applications ..... 6-3
AN155 Digital Voltmeters and the MM5330 ..... 6-7
AN156 Specifying A/D and D/A Converters ..... 6-15
AN168 MM5865 Universal Timer Applications ..... 6-21
AN169 A 4-Digit, 7-Function Stop Watch/Timer ..... 6-31
AN196 Programmable TV Timer/Time-Channel Display ..... 6-45
AN249 MM54240 Asynchronous Receiver/Transmitter Remote Controller Applications ..... 6-55
AN250 Applications and Uses of the MM5321 TV Camera Sync Generator ..... 6-59
AN251 A Broadcast Quality TV Sync Generator Made Economical through LSI ..... 6-63
MOS Brief 19 Integrated Circuit Combination Provides Digital Frequency Readout with Digital Clock for Radios ..... 6-69

**Section 7 Custom MOS/LSI**

Your Choice for Custom MOS/LSI Circuits ..... 7-3
Working Together to Get the Job Done with Custom MOS/LSI ..... 7-4
The Custom MOS/LSI Development Cycle ..... 7-5
A Working Partnership ..... 7-6

**Section 8**

Ordering Information ..... 8-3
Physical Dimensions ..... 8-4

# Alpha-Numerical Index


COP320 Single-Chip N-Channel Microcontroller ..... 1-56
COP320C Single-Chip CMOS Microcontroller ..... 1-79
COP321 Single-Chip N-Channel Microcontroller ..... 1-56
COP321C Single-Chip CMOS Microcontroller ..... 1-79
COP400 COPS™Family User's Guide ..... 2-3
COP402 ROMless N-Channel Microcontroller ..... 1-5
COP402M ROMless N-Channel Microcontroller ..... 1-5
COP404L ROMless N-Channel Microcontroller ..... 1-24
COP410L Single-Chip N-Channel Microcontroller ..... 1-40
COP411L Single-Chip N-Channel Microcontroller ..... 1-40
COP420 Single-Chip N-Channel Microcontroller ..... 1-56
COP420C Single-Chip CMOS Microcontroller ..... 1-79
COP420L Single-Chip N-Channel Microcontroller ..... 1-96
COP421 Single-Chip N-Channel Microcontroller ..... 1-56
COP421C Single-Chip CMOS Microcontroller ..... 1-79
COP421L Single-Chip N-Channel Microcontroller ..... 1-96
COP430 A to D Converter ..... 1-134
COP431 A to D Converter ..... 1-134
COP432 A to D Converter ..... 1-134
COP444L Single-Chip N-Channel Microcontroller ..... 1-115
COP445L Single-Chip N-Channel Microcontroller ..... 1-115
COP450 PROM-RAM Interface Chip ..... 1-136
COP451 PROM-RAM Interface Chip ..... 1-136
COP452 Frequency/Counter Peripheral ..... 1-138
COP470 V.F. Display Driver ..... 1-140
COP472 Liquid Crystal Display Controller ..... 1-147
COP498 Low Power CMOS RAM and Timer ..... 1-153
COP499 Low Power CMOS Memory ..... 1-155
MM5034 Octal 80-Bit Static Shift Register ..... 5-257
MM5035 Octal 80-Bit Static Shift Register ..... 5-257
MM5303 Universal Fully Asynchronous Receiver/Transmitter ..... 5-260
MM5307 Baud Rate Generator/Programmable Divider ..... 5-266
MM5309 Digital Clock ..... 5-13
MM5311 Digital Clock ..... 5-13
MM5312 Digital Clock ..... 5-13
MM5313 Digital Clock ..... 5-13
MM5314 Digital Clock ..... 5-13
MM5315 Digital Clock ..... 5-13
MM5316 Digital Alarm ClockClock ..... 5-20
MM5321 TV Camera Sync Generator ..... 5-91
MM5322 Color Bar Generator Chip ..... 5-97
MM5330 4½-Digit Panel Meter Logic Block ..... 5-271
MM5368 CMOS Oscillator Divider Circuit ..... 5-245
MM5369 Series 17 Stage Mask Programmable Oscillator/Divider ..... 5-248
MM5387AA Digital Alarm Clock ..... 5-25

# Alpha-Numerical Index (Continued)


MM5393 Push Button Pulse Dialer Circuit ..... 5-197
MM5394 Push Button Pulse Dialer Circuit ..... 5-197
MM5395 DTMF (Touch Tone® ) Generator ..... 5-202
MM5402 Digital Alarm Clock ..... 5-49
MM5405 Digital Alarm Clock ..... 5-49
MM5406 Deluxe Display and Clock Radio ..... 5-54
MM5407 Digital Thermometer ..... 5-61
MM5430 AM/FM Radio Frequency Display ..... 5-115
MM5431 AM/FM Radio Frequency Display ..... 5-115
MM5439 Microprocessor Compatible Phase Lock Loop (PLL) ..... 5-120
MM5445 VF Display Driver ..... 5-219
MM5446 VF Display Driver ..... 5-219
MM5447 VF Display Driver ..... 5-219
MM5448 VF Display Driver ..... 5-219
MM5450 LED Display Driver ..... 5-222
MM5451 LED Display Driver ..... 5-222
MM5452 Liquid Crystal Display Driver ..... 5-227
MM5453 Liquid Crystal Display Driver ..... 5-227
MM5455 Digital Alarm Clock ..... 5-66
MM5456 Digital Alarm Clock ..... 5-70
MM5457 Digital Alarm Clock ..... 5-70
MM5480 LED Display Driver ..... 5-232
MM5481 LED Display Driver ..... 5-235
MM5780 Educational Arithmetic Game ..... 5-181
MM5837 Digital Noise Source ..... 5-151
MM5840 TV Channel Number (16-Channel) and Time Display Circuit ..... 5-153
MM5863 12-Bit Binary A/D Building Block ..... 5-331
MM5865 Universal Timer ..... 5-337
MM7317B Alarm Clock Calendar ..... 5-81
MM7318B Alarm Clock Calendar ..... 5-81
MM52116(2316E) 16,384-Bit (2048 x 8) Read Only Memory ..... 3-3
MM52116FDW Character Generator ..... 3-6
MM52116FDX Character Generator ..... 3-6
MM52132 32,768-Bit (4096 x 8) MAXI-ROM™ ..... 3-13
MM52164 65,536-Bit (8192 x 8) MAXI-ROM ..... 3-16
MM52264 65,536-Bit (8192 x 8) Clocked MAXI-ROM ..... 3-19
MM53100 Programmable TV Timer ..... 5-102
MM53105 Programmable TV Timer ..... 5-102
MM53107 Series 17-Stage Oscillator/Divider ..... 5-251
MM53108 Digital Alarm Clock ..... 5-25
MM53110 Series Auto Clock and Elapsed Timer ..... 5-31
MM53113 Digital Alarm Clock ..... 5-36
MM53118AA TV Digital Tuning ..... 5-107
MM53124 Automobile Clock and Elapsed Timer ..... 5-41

# Alpha-Numerical Index (Continued)


MM53125 DTMF (Touch Tone®) Generator ..... 5-202
MM53130 DTMF (Touch Tone®) Generator ..... 5-207
MM53143 Push Button Pulse Dialer Circuit ..... 5-197
MM53144 Push Button Pulse Dialer Circuit ..... 5-197
MM53190 Push Button Pulse Dialer ..... 5-212
MM53200 Encode/Decoder ..... 5-278
MM53224 Automobile Clock and Elapsed Timer ..... 5-45
MM54240 Asynchronous Receiver/Transmitter Remote Controller ..... 5-280
MM55108 PLL Frequency Synthesizer with Receive/Transmit Mode ..... 5-127
MM55110 PLL Frequency Synthesizer with Receive/Transmit Mode ..... 5-127
MM55121 Serial Data/PLL Frequency Synthesizer ..... 5-132
MM55122 Serial Data/PLL Frequency Synthesizer ..... 5-137
MM55123 Serial Data/PLL Frequency Synthesizer ..... 5-142
MM55124 PLL Frequency Synthesizer ..... 5-147
MM55126 PLL Frequency Synthesizer ..... 5-147
MM57109 MOS/LSI Number-Oriented Processor ..... 5-284
MM57436 Decimal/Binary Up/Down Counter ..... 5-309
MM57455 Advanced Educational Arithmetic Game ..... 5-187
MM57459 8-Digit LED Direct-Drive Memory Calculator ..... 5-190
MM57499 96- or 144-Key Serial Keyboard Interface (SKI) ..... 5-316
MM58106 Digital Clock and TV Display Circuit ..... 5-158
MM58142 TV Synthesizer ..... 5-163
MM58143 LCD Alarm Clock Circuit ..... 5-74
MM58144 LCD Alarm Clock Circuit ..... 5-74
MM58146 TV Clock and Channel Display ..... 5-168
MM58167 Microprocessor-Compatible Real Time Clock ..... 5-347
MM58174 Microprocessor-Compatible Real Time Clock ..... 5-353
MM58183 LCD Alarm Clock Circuit ..... 5-74
MM58184 LCD Alarm Clock Circuit ..... 5-74
MM58201 Multiplexed LCD Driver ..... 5-238
MM58313 Varactor Tuner Display Circuit ..... 5-173


Touch Tone® is a Registered Trademark of Bell Telephone

Introduction to COPS™

# COPS Single Chip Microcontroller and Peripherals

# COPS™ Family Introduction:

National Semiconductor manufactures a wide-ranging and sophisticated family of single-chip microcomputers to meet the total needs of the microcontroller marketplace.

**Computer on a Chip**

Each member of the COP400 series of single-chip microcontrollers is a complete "computer-on-a-chip," containing all system timing, internal logic, ROM, RAM, and I/O necessary to implement dedicated control functions in a wide variety of applications. The COP400 family of devices feature instruction sets, internal architectures and I/O schemes designed to ease keyboard input/display output and efficient binary and BCD data manipulation. On-chip ROM sizes range from 512 × 8 to 2,048 × 8 bits, RAM from 32 × 4 to 128 × 4 bits, instruction sets from 43 to 57 commands, I/O lines from 16 to 36, and instruction cycle execution times ranging from 4 to 16 microseconds. The family is fabricated using three processes: an advanced, high speed N-channel MOS; a low power NMOS; and an even lower power complementary MOS.

To ease program development on the new COP400 series a specially designed COPS Product Development System (PDS), has been introduced.

**Programmable Features**

The range of the microcontrollers available in the COP400 family allow the user to specify the optimum device for use in a particular application. Not only can the user pick a part with RAM, ROM, I/O and speed optimized for specific tasks, but the family also offers a choice of ports with differing electrical characteristics. Each part contains a number of clock, I/O and other options, mask programmed into the device at the same time the ROM is coded with the user's dedicated program. This allows great flexibility in matching particular COP400 microcontrollers to the user need.

All COP400 devices feature single supply operation and fast standardized test procedures that verify the internal logic and user program. The flexible I/O configuration of the COP400 microcontrollers allow them to interface and drive a wide range of devices using a minimal amount of external parts. Typical interfaced devices include: keyboards and displays (direct segment and direct digit drive), external data memories, printers, other COP devices, A/D and D/A converters, power control devices such as SCRs and TRIACs, mechanical actuators, general purpose microprocessors, shift registers and external ROM storage devices.

**Applications**

The COP400 devices are aimed at such high volume applications as clocks, timers, laboratory instruments, radio controllers, applicance controllers, programmable sequencers, scales, cash registers, calculators, microcontroller computational elements, toys, games, and automotive computers.

**Performance**

The COP420/420L/420C devices constitute the center-piece configurations of the family, with 1k × 8 ROM, 64 × 4 RAM, true vectored interrupt plus restart, three level subroutine stack, 23 I/O lines, 57 command instruction set, internal time base counter for real time processing, internal binary counter register with serial I/O capability, general purpose and TRI-STATE outputs, LED direct drive, and software/hardware compatibility with the rest of the COP400 family — all within a 28-pin dual-in-line package. The NMOS COP420 operates over a 4.5 to 6.3 volt single supply range and has a 4 microsecond instruction cycle execution time. Operating supply current is 20 milliamperes at 5 volts. The low power (40 mW, max.) NMOS COP420L differs from the

COP420 in that it has a 4.5 to 9.5 volt supply range, a 16 $\mu$s instruction cycle execution time, a divide by 32 crystal clock option and direct LED digit drive capability. The COP420C is the CMOS version with a 2.4 to 6.0 volt operating supply range and a dual clock mode option for operation at low speed (244 $\mu$s) with low power consumption or high speed (16 $\mu$s) when necessary to perform internal data computations at a faster rate. The COP420C also provides the user with a sleep (timed pause) mode entered under program control with very low power consumption (15 $\mu$A).

The 24-pin COP421/421L devices are identical to the COP420/420L versions except that they have 19 I/O lines instead of 23, and no interrupt capability. The COP410L/411L have the same electrical specifications as the COP420L/421L but half the program storage (512 x 8 ROM), half the data storage (32 x 4 RAM), only 43 instructions, two instead of three stack levels, no interrupt capability, 19 and 16 I/O lines respectively. They are in 24-pin and 20-pin packages, respectively.

The COP440/444L are expanded versions of the COP420/420L devices, with the same instruction set but double the memory (2,048 x 8 ROM and 128 x 4 RAM). The 28-pin COP444L has 24 I/O lines and the 40-pin COP440 (future product) has 36 lines of I/O. The 40-pin COP402 and COP404L are ROM-less version of the COP420 and COP444L, respectively, available for prototyping, or in quantity for small volume applications using up to 1024 x 8 and 2048 x 8 bits of external ROM.

**Bus Compatibility**

A key feature of the COP420 and COP 420C is that they are MICROBUS™ compatible, an option that allows it to be used as a peripheral microprocessor device, inputting and outputting data from and to any host microprocessor in National's MICROBUS-compatible family of 8-and 16-bit microprocessors. MICROBUS is a standard interconnect system for 8-bit parallel data transfer between MOS/LSI CPUs and interface devices. The COP402M is the ROM-less MICROBUS-compatible version of the COP402. It is intended for use in prototyping systems in low volume applications which use the COP420 as a host CPU peripheral component.

**Development Systems**

To aid in the efficient and speedy programming of the COP400 series microcontrollers National has developed the COP Product Development System (PDS), built around a 16-bit microcomputer, 32k bytes of R/W memory and 12k bytes of PROM firmware. The disk-based system features an editor and assembler for handling source code entry, conversion to object code and maintaining documentation. An in-circuit emulator card attachment allows object code to be executed under the careful control of a COP Monitor de-bug utility. The PDS also features a circuit fixture for incoming inspection of COP400 devices.

National is continually expanding the COP400 family. Future members will include expanded software and hardware capabilities, alternative electrical specification devices, and smaller devices suitable for use in less demanding applications.

**Schools**

A COPS™ training course is available for instruction in programming, interfacing, and applications. For further information on the course, contact your local National Semiconductor Sales Office.

Section 1

# COPS Devices

# NATIONAL SEMICONDUCTOR COP400 MICROCONTROLLER FAMILY GUIDE

| | Specifications / COP: | ROMless Devices | | | Single-Chip Microcontrollers | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 402 | 402M | 404L | 410L | 411L | 420 | 420L | 420C | 421 | 421L | 421C | 444L | 445L |
| MEMORY | ROM x 8 | up to 1024 ext. | | up to 2048 ext. | 512 | | 1024 | | | 1024 | | | 2048 | |
| | RAM x 4 | 64 | | 128 | 32* | | 64* | | | 64* | | | 128* | |
| INPUTS/OUTPUTS | Inputs | 4 | | | 0 | | 4 | | | 0 | | | 4 | 0 |
| | Bidirectional TRI-STATE™ I/O | 8 | | | 8 | | 8 | | | 8 | | | 8 | |
| | Bidirectional I/O | 4 | | | 4 | 3 | 4 | | | 4 | | | 4 | |
| | Outputs | 4 | | | 4 | 2 | 4 | | | 4 | | | 4 | |
| | Serial I/O and External Event Counter | Yes | | | Yes | | Yes | | SIO | Yes | | SIO | Yes | |
| GENERAL | Interrupt | Yes | No | Yes | No | | Yes | | | No | | | Yes | No |
| | Stack Levels | 3 | | | 2 | | 3 | | | 3 | | | 3 | |
| | MICROBUS™ Option | No | Yes | No | No | | Yes | No | Yes | No | | | No | |
| | Instruction Cycle (μs) | 4 | | 16 | 16 | | 4 | 16 | | 4 | 16 | | 16 | |
| POWER/PKG | Supply Voltage | 4.5-6.3 | | 4.5-9.5 | 4.5-6.3*** | | 4.5-6.3 | 4.5-6.3*** | 2.4-6.0 | 4.5-6.3 | 4.5-6.3*** | 2.4-6.0 | 4.5-6.3*** | |
| | Supply Current (mA) | 30 | | 15 | 5 | | 30 | 8 | ** | 30 | 8 | ** | 11 | |
| | Package Size (pins) | 40 | | | 24 | 20 | 28 | | | 24 | | | 28 | 24 |

*RAM keep-alive option (except 420C/421C). **Fast: 800μA, Slow: 35μA, Sleep: 15μA. ***4.5-9.5V optionally available

National Semiconductor

# COP402/COP402M ROMless N-Channel Microcontrollers

## General Description

The COP402 and COP402M ROMless Microcontrollers are members of the COPS™ family, fabricated using N-channel silicon gate MOS technology. Each part contains CPU, RAM and I/O, and is identical to a COP420 device, except the ROM has been removed; pins have been added to output the ROM address and to input ROM data. In a system, the COP402 or 402M will perform exactly as the COP420; this important benefit facilitates development and debug of a COP420 program prior to masking the final part. These devices are also appropriate in low volume applications, or when the program may require changing. The COP402M is identical to the COP402, except the MICROBUS™ interface option has been implemented.

The COP402 may also be used to emulate the COP410L, 411L, 420L or 420C by appropriately reducing the clock frequency.

## Features

- Low cost
- Exact circuit equivalent of COP420
- Standard 40-pin dual-in-line package
- Interfaces with standard PROM or ROM
- 64x4 RAM, addresses up to 1kx8 ROM
- MICROBUS™ compatible (COP402M)
- Powerful instruction set
- True vectored interrupt, plus restart
- Three-level subroutine stack
- 4.0μs instruction time
- Single supply operation (4.5V to 6.3V)
- Internal time-base counter for real-time processing
- Internal binary counter register with serial I/O capability
- Software/hardware compatible with other members of COP400 family

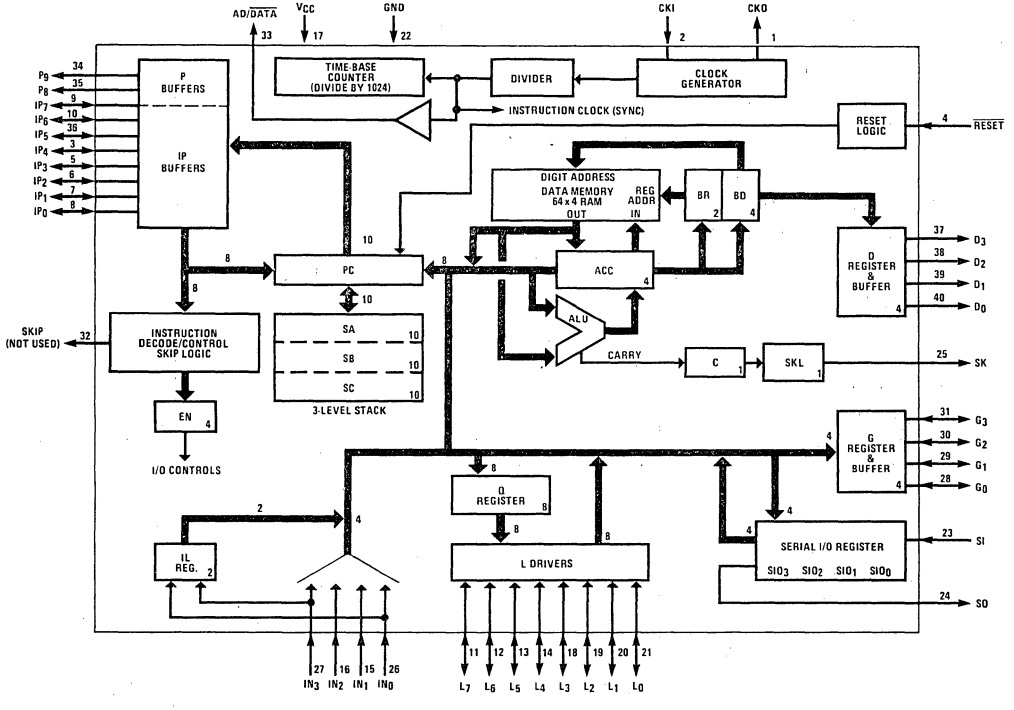



**Figure 1. COP402/402M Block Diagram**

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Relative to GND | −0.5V to +7V |
| Ambient Operating Temperature | 0°C to +70°C |
| Ambient Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |
| Power Dissipation | 0.75 Watt at 25°C<br>0.4 Watt at 70°C |

*Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.*

## DC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 6.3V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Operating Voltage ($V_{CC}$) | | 4.5 | 6.3 | V |
| Operating Supply Current | $V_{CC}=5V$, $T_A=25°C$<br>(all inputs and outputs open) | | 30 | mA |
| Input Voltage Levels | | | | |
| CKI Input Levels | | | | |
| Logic High ($V_{IH}$) | | 2.0 | | V |
| Logic Low ($V_{IL}$) | | | 0.4 | V |
| RESET Input Levels | | | | |
| Logic High | | 0.7 $V_{CC}$ | | V |
| Logic Low | | | 0.6 | V |
| RESET Hysteresis | | 1.0 | | V |
| SO Input Level (Test mode) | | 2.0 | 3.0 | V |
| Input Levels | | | | |
| Logic High | $V_{CC}=$ max | 3.0 | | V |
| Logic High | $V_{CC}=5V\pm5\%$ | 2.0 | | V |
| Logic Low | | −0.3 | 0.8 | V |
| Logic Low ($IN_{0:3}$ with Load) | | −0.3 | 0.5 | V |
| Output Voltage Levels (Note 2) | | | | |
| TTL Operation | $V_{CC}=5V\pm5\%$ | | | |
| Logic High ($V_{OH}$) | $I_{OH}=-100\mu A$ | 2.4 | | V |
| Logic Low ($V_{OL}$) | $I_{OL}=+1.6mA$ | −0.3 | 0.4 | V |
| CMOS Operation | | | | |
| Logic High ($V_{OH}$) | $I_{OH}=-10\mu A$ | $V_{CC}-1$ | | V |
| Logic Low ($V_{OL}$) | $I_{OL}=+10\mu A$ | −0.3 | 0.2 | V |
| Output Current Levels | | | | |
| LED Direct Drive Output | $V_{CC}=6V$ | | | |
| Logic High ($I_{OH}$) | $V_{OH}=2.0V$ | 2.5 | 14 | mA |
| TRI-STATE® Output Leakage Current | | −10 | +10 | μA |
| IP7 ~ IP0 Output Voltage Levels | | | | |
| $V_{OH}$ | $I_{OH}=-50\mu A$ | 2.7 | | V |
| $V_{OL}$ | $I_{OL}=+360\mu A$ | −0.3 | 0.4 | V |

## AC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 6.3V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Instruction Cycle Time — $t_C$ | figure 3 | 4 | 10 | μs |
| CKI Using Crystal (figure 8) | | | | |
| Input Frequency — $f_I$ | ÷16 mode | 1.6 | 4 | MHz |
| Duty Cycle (Note 2) | figure 3a | 30 | 55 | % |
| INPUTS: (figure 3a) | | | | |
| $IN_3$-$IN_0$, $G_3$-$G_0$, $L_7$-$L_0$ | | | | |
| $t_{SETUP}$ | | 1.7 | | μs |
| $t_{HOLD}$ | | 312 | | ns |
| SI, $IP_7$-$IP_0$ | | | | |
| $t_{SETUP}$ | | 0.3 | | μs |
| $t_{HOLD}$ | | 250 | | ns |
| OUTPUTS: | | | | |
| COP TO CMOS PROPAGATION DELAY | $4.5V \leq V_{CC} \leq 6.3V$, $C_L = 50pF$, $V_{OH} = 0.7V_{CC}$, $V_{OL} = 0.3V_{CC}$ | | | |
| SK as a Logic-Controlled Clock | | | | |
| $t_{PD1}$ | | | 1.1 | μs |
| $t_{PD0}$ | | | 0.3 | μs |
| SO, SK as a Data Output | | | | |
| $t_{PD1}$ | | | 1.4 | μs |
| $t_{PD0}$ | | | 0.3 | μs |
| $t_{PD1}$ | $V_{OH} = 2V$ | | 0.7 | μs |
| $D_3$-$D_0$, $G_3$-$G_0$ | | | | |
| $t_{PD1}$ | | | 1.6 | μs |
| $t_{PD0}$ | | | 0.6 | μs |
| $L_7$-$L_0$ (LED Direct Drive) | | | | |
| $t_{PD1}$ | $V_{OH} = 2V$ | | 2.4 | μs |
| $t_{PD0}$ | | | 0.4 | μs |
| COP TO TTL PROPAGATION DELAY | fanout = 1 Standard TTL Load, $V_{CC} = 5V \pm 5\%$, $C_L = 50pF$, $V_{OH} = 2.4V$, $V_{OL} = 0.4V$ | | | |
| AD/$\overline{DATA}$ | | | | |
| $t_{PD1}$ | | | 0.5 | μs |
| $t_{PD0}$ | | | 0.5 | μs |
| SKIP | | | | |
| $t_{PD1}$ | | | 0.6 | μs |
| $t_{PD0}$ | | | 0.6 | μs |

## AC Electrical Characteristics (continued) $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 6.3V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| OUTPUTS (cont.): | | | | |
| SK as a Logic-Controlled Clock | | | | |
| $t_{PD1}$ | | | 0.8 | μs |
| $t_{PD0}$ | | | 0.8 | μs |
| SK as a Data Output, SO | | | | |
| $t_{PD1}$ | | | 1.0 | μs |
| $t_{PD0}$ | | | 1.0 | μs |
| $D_3$-$D_0$, $G_3$-$G_0$ | | | | |
| $t_{PD1}$ | | | 1.3 | μs |
| $t_{PD0}$ | | | 1.3 | μs |
| $L_7$-$L_0$ | | | | |
| $t_{PD1}$ | | | 1.4 | μs |
| $t_{PD0}$ | | | 0.4 | μs |
| $IP_7$-$IP_0$, $P_9$, $P_8$ | | | | |
| $t_{PD1}$ | | | 1.5 | μs |
| $t_{PD0}$ | | | 1.5 | μs |
| CKO (figure 3a) | | | | |
| $t_{PD1}$ | | | 0.2 | μs |
| $t_{PD0}$ | | | 0.2 | μs |
| MICROBUS™ TIMING (COP402M) | $C_L$ = 50pF, $V_{CC}$ = 5V ± 5% | | | |
| A. Read Operation (figure 4) | | | | |
| Chip Select Stable before $\overline{RD}$ — $t_{CSR}$ | | 50 | | ns |
| Chip Select Hold Time for $\overline{RD}$ — $t_{RCS}$ | | 5 | | ns |
| $\overline{RD}$ Pulse Width — $t_{RR}$ | | 300 | | ns |
| Data Delay from $\overline{RD}$ — $t_{RD}$ | | | 250 | ns |
| $\overline{RD}$ to Data Floating — $t_{DF}$ | | | 200 | ns |
| B. Write Operation (figure 5) | | | | |
| Chip Select Stable before $\overline{WR}$ — $t_{CSW}$ | | 20 | | ns |
| Chip Select Hold Time for $\overline{WR}$ — $t_{WCS}$ | | 20 | | ns |
| $\overline{WR}$ Pulse Width — $t_{WW}$ | | 300 | | ns |
| Data Setup Time for $\overline{WR}$ — $t_{DW}$ | | 200 | | ns |
| Data Hold Time for $\overline{WR}$ — $t_{WD}$ | | 40 | | ns |
| INTR Transition Time from $\overline{WR}$ — $t_{WI}$ | | | 700 | ns |

**Note 1:** Duty Cycle = $t_{WI}/(t_{WI} + t_{WO})$.

**Note 2:** See figure 11 for additional I/O characteristics.

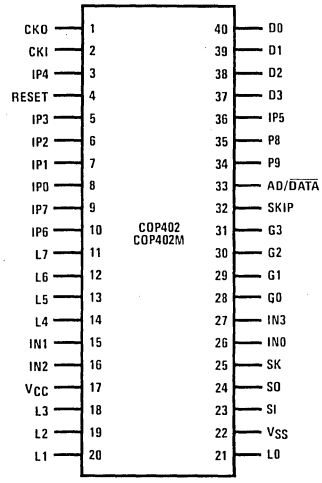



**Figure 2. Connection Diagram**

| Pin | Description |
|---|---|
| $L_7$-$L_0$ | 8 bidirectional I/O ports with TRI-STATE® |
| $G_3$-$G_0$ | 4 bidirectional I/O ports |
| $D_3$-$D_0$ | 4 general purpose outputs |
| $IN_3$-$IN_0$ | 4 general purpose inputs |
| SI | Serial input (or counter input) |
| SO | Serial output (or general purpose output) |
| SK | Logic-controlled clock (or general purpose output) |
| AD/$\overline{\text{DATA}}$ | Address out/data in flag |

| Pin | Description |
|---|---|
| SKIP | Instruction skip output |
| CKI | System oscillator input |
| CKO | System oscillator output |
| $\overline{\text{RESET}}$ | System reset input |
| $V_{CC}$ | Power supply |
| GND | Ground |
| IP7-IP0 | 8 bidirectional ROM address and data ports |
| P8, P9 | 2 ROM address outputs |

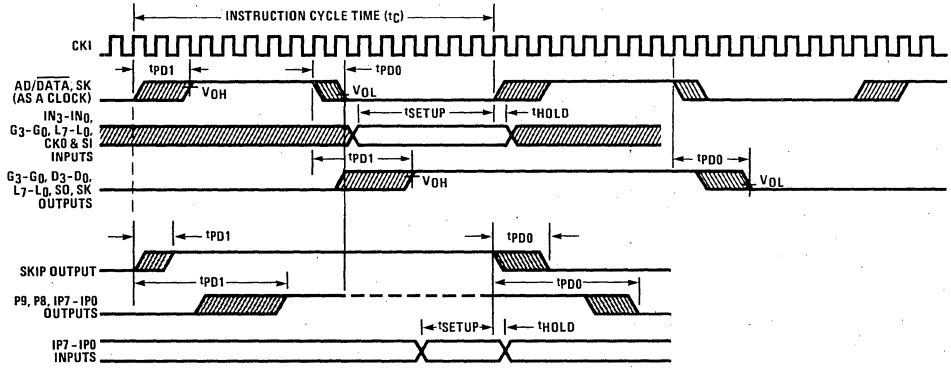



**Figure 3a. Input/Output Timing Diagrams (Crystal ÷16 Mode)**

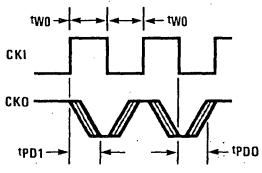



**Figure 3b. CKO Output Timing**

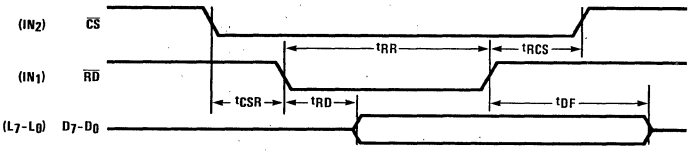



Figure 4. MICROBUS™ Read Operation Timing

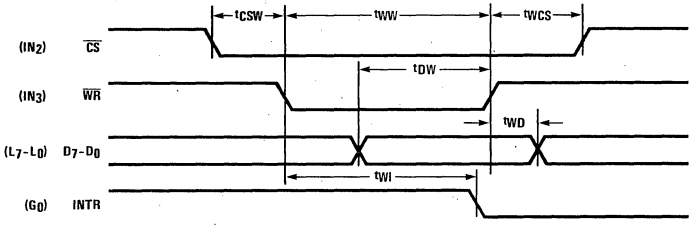



Figure 5. MICROBUS™ Write Operation Timing

## FUNCTIONAL DESCRIPTION

A block diagram of the COP402 is given in figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1" (greater than 2 volts). When a bit is reset, it is a logic "0" (less than 0.8 volts).

### Program Memory

Program Memory consists of a 1,024-byte external memory (typically PROM). Words of this memory may be program instructions, program data or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID and LQID instructions, ROM must often be thought of as being organized into 16 pages of 64 words each.

ROM addressing is accomplished by a 10-bit PC register. Its binary value selects one of the 1,024 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential **10-bit binary count** value. Three levels of subroutine nesting are implemented by the 10-bit subroutine save registers, SA, SB and SC, providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the Instruction Decode, Control and Skip Logic circuitry.

### Data Memory

Data memory consists of a 256-bit RAM, organized as 4 data registers of 16 4-bit digits. RAM addressing is implemented by a 6-bit B register whose upper 2 bits (Br) select 1 of 4 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) is usually loaded into or from, or exchanged with, the A register (accumulator), it may also be loaded into or from the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the 6-bit contents of the operand field of these instructions. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

### Internal Logic

The 4-bit **A register** (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd portions of the B register, to load and input 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

A **4-bit adder** performs the arithmetic and logic functions of the COP402/402M, storing its results in A. It also outputs a carry bit to the 1-bit **C register**, most often employed to indicate arithmetic overflow. The C register, in conjunction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SK or can enable SK to be a sync clock each instruction cycle time. (See XAS instruction and EN register description, below.)

Four **general-purpose inputs, $IN_3$-$IN_0$**, are provided; $IN_1$, $IN_2$ and $IN_3$ may be selected, by a mask-programmable option, as Read Strobe, Chip Select and Write Strobe inputs, respectively, for use in MICROBUS™ applications.

The **D register** provides 4 general-purpose outputs and is used as the destination register for the 4-bit contents of Bd.

The **G register** contents are outputs to 4 general-purpose bidirectional I/O ports. $G_0$ may be mask-programmed as a "ready" output for MICROBUS™ applications.

The **Q register** is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control. (See LEI instruction.) With the MICROBUS™ option selected, Q can also be loaded with the 8-bit contents of the L I/O ports upon the occurrence of a write strobe from the host CPU.

The **8 L drivers**, when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. As explained above, the MICROBUS™ option allows L I/O port data to be latched into the Q register. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the LED Direct Drive output configuration option) with Q data being outputted to the Sa-Sg and decimal point segments of the display.

The **SIO register** functions as a 4-bit serial-in/serial-out shift register or as a binary counter depending on the contents of the EN register. (See EN register description, below.) Its contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O by connecting SO to external serial-in/parallel-out shift registers.

The **XAS instruction** copies C into the SKL latch. In the counter mode, SK is the output of SKL. In the shift register mode, SK outputs SKL ANDed with internal instruction cycle clock.

The **EN register** is an internal 4-bit register loaded under program control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register ($EN_3$-$EN_0$).

1. The least significant bit of the enable register, $EN_0$, selects the SIO register as either a 4-bit shift register or a 4-bit binary counter. With $EN_0$ set, SIO is an asynchronous binary counter, *decrementing* its value by one upon each low-going pulse ("1" to "0") occurring on the SI input. Each pulse must be at least two instruction cycles wide. SK outputs the value of SKL. The SO output is equal to the value of $EN_3$. With $EN_0$ reset, SIO is a serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time. (See 4 below.) The SK output becomes a logic-controlled clock.

2. With $EN_1$ set the $IN_1$ input is enabled as an interrupt input. Immediately following an interrupt, $EN_1$ is reset to disable further interrupts.

3. With $EN_2$ set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting $EN_2$ disables the L drivers, placing the L I/O ports in a high-impedance input state. If the MICROBUS™ option is being used, $EN_2$ does not affect the L drivers.

4. $EN_3$, in conjunction with $EN_0$, affects the SO output. With $EN_0$ set (binary counter option selected) SO will output the value loaded into $EN_3$. With $EN_0$ reset (serial shift register option selected), setting $EN_3$ enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting $EN_3$ with the serial shift register option selected disables SO as the shift register output; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0." The table below provides a summary of the modes associated with $EN_3$ and $EN_0$.

**Interrupt**

The following features are associated with the $IN_1$ interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

a. The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1) onto the stack, pushing in turn the contents of the other subroutine-save registers to the next lower level (PC + 1 → SA → SB → SC). Any previous contents of SC are lost. The program counter is set to hex address 0FF (the last word of page 3) and $EN_1$ is reset.

**Enable Register Modes — Bits $EN_3$ and $EN_0$**

| $EN_3$ | $EN_0$ | SIO | SI | SO | SK |
|---|---|---|---|---|---|
| 0 | 0 | Shift Register | Input to Shift Register | 0 | If SKL = 1, SK = SYNC<br>If SKL = 0, SK = 0 |
| 1 | 0 | Shift Register | Input to Shift Register | Serial Out | If SKL = 1, SK = SYNC<br>If SKL = 0, SK = 0 |
| 0 | 1 | Binary Counter | Input to Binary Counter | 0 | If SKL = 1, Sk = 1<br>If SKL = 0, SK = 0 |
| 1 | 1 | Binary Counter | Input to Binary Counter | 1 | If SKL = 1, SK = 1<br>If SKL = 0, SK = 0 |

b. An interrupt will be acknowledged only after the following conditions are met:
   1. $EN_1$ has been set.
   2. A low-going pulse ("1" to "0") at least two instruction cycles wide occurs on the $IN_1$ input.
   3. A currently executing instruction has been completed.
   4. All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction, the interrupt will not be acknowledged until the second JP instruction has been executed.

c. Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon the popping of the stack. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skip logic status is saved and program control is transferred to the interrupt servicing routine at hex address OFF. At the *end* of the interrupt routine, a RET instruction is executed to "pop" the stack and return program control to the instruction following the original ASC. *At this time*, the skip logic is enabled and skips this instruction because of the previous ASC carry. Subroutines and the LQID instruction should not be nested within the interrupt servicing routine since their popping of the stack enables any previously saved main program skips, interfering with the orderly execution of the interrupt routine.

d. The first instruction of the interrupt routine at hex address 0FF must be a NOP.

e. A LEI instruction can be put immediately before the RET to re-enable interrupts.

### MICROBUS™ Interface

The COP402M can be used as a peripheral microprocessor device, inputting and outputting data from and to a host microprocessor ($\mu$P). $IN_1$, $IN_2$, and $IN_3$ general purpose inputs become MICROBUS™ compatible read-strobe, chip-select, and write-strobe lines, respectively. $IN_1$ becomes $\overline{RD}$ — a logic "0" on this input will cause Q latch data to be enabled to the L ports for input to the $\mu$P. $IN_2$ becomes $\overline{CS}$ — a logic "0" on this line selects the COP402M as the $\mu$P peripheral device by enabling the operation of the $\overline{RD}$ and $\overline{WR}$ lines and allows for the selection of one of several peripheral components. $IN_3$ becomes $\overline{WR}$ — a logic "0" on this line will write bus data from the L ports to the Q latches for input to the COP402M. $G_0$ becomes INTR, a "ready" output reset by a write pulse from the $\mu$P on the $\overline{WR}$ line, providing the "handshaking" capability necessary for asynchronous data transfer between the host CPU and the COP402M.

This option has been designed for compatibility with National's MICROBUS™ — a standard interconnect system for 8-bit parallel data transfer between MOS/LSI CPUs and interfacing devices. (See MICROBUS™, National Publication.) The functioning and timing relationships between the COP402M signal lines affected by this option are as specified for the MICROBUS™ interface, and are given in the AC electrical characteristics and shown in the timing diagrams (figures 4 and 5). Connection to the MICROBUS™ is shown in figure 6.

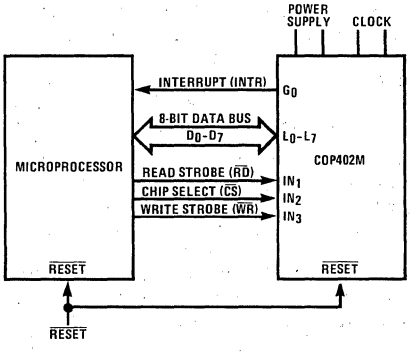



**Figure 6. MICROBUS™ Option Interconnect**

### Initialization

The Reset Logic will initialize (clear) the device upon power-up if the power supply rise time is less than 1ms and greater than 1$\mu$s. If the power supply rise time is greater than 1ms, the user must provide an external RC network and diode to the $\overline{RESET}$ pin as shown below. The $\overline{RESET}$ pin is configured as a Schmitt trigger input. If not used it should be connected to $V_{CC}$. Initialization will occur whenever a logic "0" is applied to the $\overline{RESET}$ input, provided it stays low for at least two instruction cycle times.

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, G, and SO are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. *Data Memory (RAM) must be cleared by the user's program.* The first instruction at address 0 must be a CLRA.

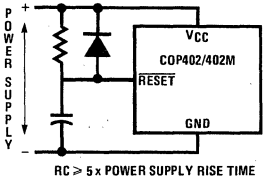


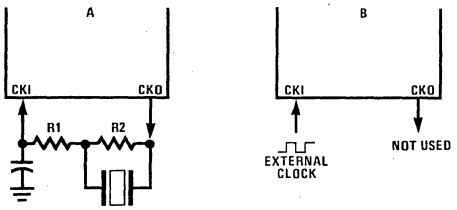

**Figure 7. Power-Up Clear Circuit**

### Oscillator

There are two basic clock oscillator configurations available as shown by figure 8.

**a. Crystal Controlled Oscillator.** CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency divided by 16.

**b. External Oscillator.** CKI is driven by an external clock signal. The instruction cycle time is the clock frequency divided by 16.




| Crystal Value | Component Values | | |
|---|---|---|---|
| | R1 | R2 | C |
| 4MHz | 1k | 1M | 27pF |
| 3.58MHz | 1k | 1M | 27pF |
| 2.09MHz | 1k | 1M | 56pF |

**Figure 8. COP402/402M Oscillator**

### External Memory Interface

The COP402 and COP402M are designed for use with an external Program Memory. This memory may be implemented using any devices having the following characteristics:

1. random addressing
2. TTL-compatible TRI-STATE® outputs
3. TTL-compatible inputs
4. access time = 1.7μs, max

Typically these requirements are met using bipolar or MOS PROMs.

During operation, the address of the next instruction is sent out on P9, P8, and IP7 through IP0 during the time that AD/$\overline{\text{DATA}}$ is high (logic "1" = address mode). Address data on the IP lines is stored into an external latch on the high-to-low transition of the AD/$\overline{\text{DATA}}$ line; P9 and P8 are dedicated address outputs, and do not need to be latched. When AD/$\overline{\text{DATA}}$ is low (logic "0" = data mode), the output of the memory is gated onto IP7 through IP0, forming the input bus. Note that the AD/$\overline{\text{DATA}}$ output has a period of one instruction time, a duty cycle of approximately 50%, and specifies whether the IP lines are used for address output or instruction input. A simplified block diagram of the external memory interface is shown in figure 9.

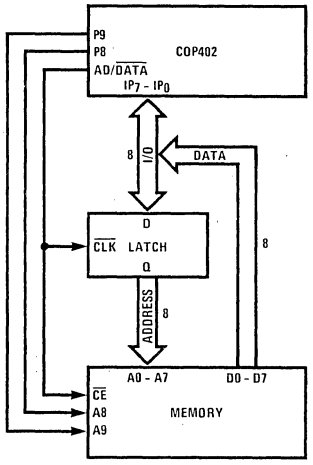



**Figure 9. External Memory Interface to COP402**

### Input/Output

COP402 outputs have the following configurations, illustrated in figure 10:

**a. Standard** — an enhancement-mode device to ground in conjunction with a depletion-mode device to $V_{CC}$, compatible with TTL and CMOS input requirements.

**b. High Drive** — same as a. except greater current sourcing capability.

**c. Push-Pull** — an enhancement-mode device to ground in conjunction with a depletion-mode device paralleled by an enhancement-mode device to $V_{CC}$. This configuration has been provided to allow for fast rise and fall times when driving capacitive loads.

**d. LED Direct Drive** — an enhancement-mode device to ground and to $V_{CC}$, meeting the typical current sourcing requirements of the segments of an LED display. The sourcing device is clamped to limit current flow. These devices may be turned off under program control (see Functional Description, EN Register), placing the outputs in a high-impedance state to provide required LED segment blanking for a multiplexed display.

**e. TRI-STATE® Push-Pull** — an enhancement-mode device to ground and $V_{CC}$ intended to meet the requirements associated with the MICROBUS™ option. These outputs are TRI-STATE® outputs, allowing for connection of these outputs to a data bus shared by other bus drivers.

f. Inputs have an on-chip depletion load device to $V_{CC}$, as shown in figure 10f.

The above input and output configurations share common enhancement-mode and depletion-mode

devices. Specifically, all configurations use one or more of six devices (numbered 1-6, respectively). Minimum and maximum current ($I_{OUT}$ and $V_{OUT}$) curves are given in figure 11 for each of these devices.

The SO,SK outputs are configured as shown in figure 10c. The D and G outputs are configured as shown in figure 10a. Note that when inputting data to the G ports, the G outputs should be set to "1." The L outputs are configured as in figure 10d on the COP402. On the COP402M the L outputs are as in figure 10e.

An important point to remember if using configuration d with the L drivers is that even when the L drivers are disabled, the depletion load device will source a small amount of current. (See figure 11.)

IP7 through IP0 outputs are configured as shown in figure 10c; P9, P8, SKIP and AD/$\overline{\text{DATA}}$ are configured as shown in figure 10b.

## COP402/402M INSTRUCTION SET

Table 1 is a symbol table providing internal architecture, instruction operand and operational symbols used in the instruction set table.

Table 2 provides the mnemonic, operand, machine code, data flow, skip conditions and description associated with each instruction in the COP402/402M instruction set.

The following information is provided to assist the user in understanding the operation of several unique instructions and to provide notes useful to programmers in writing programs.

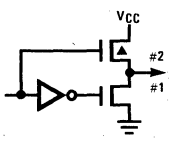
### XAS Instruction

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register. The contents of SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output. (See Functional Description, EN Register, above.) If SIO is selected as a shift register, an XAS instruction must be performed once every 4 instruction cycles to effect a continuous data stream.

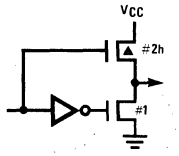
### JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM addressed by the 10-bit word, $PC_{9:8}$, A, M. $PC_9$ and $PC_8$ are not affected by this instruction.

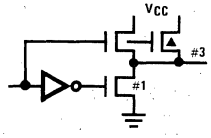
Note that JID requires 2 instruction cycles.



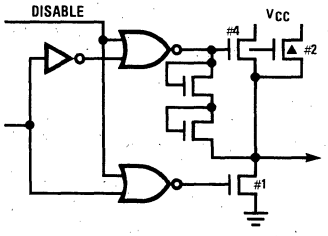

a. Standard



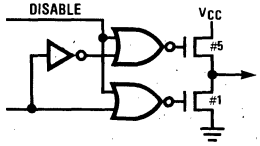

b. High Drive



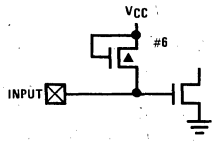

c. Push-Pull




d. LED



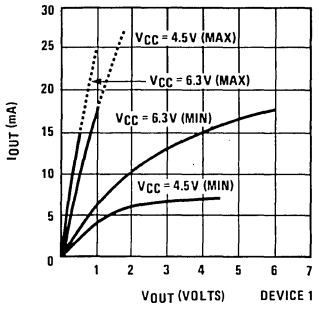

e. TRI-STATE® Push-Pull



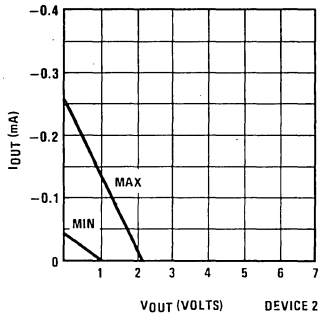

f. Input with Load

(▲IS DEPLETION DEVICE)

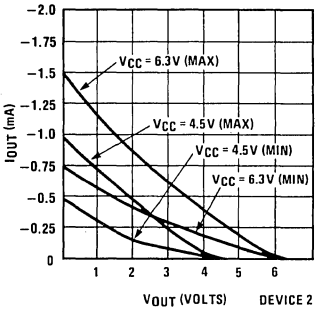
**Figure 10. Input/Output Configurations**



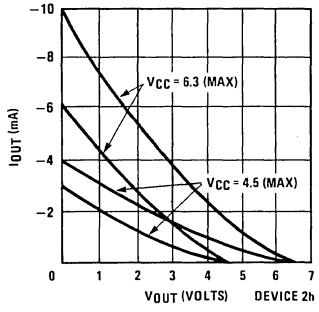

**Output Sink Current**



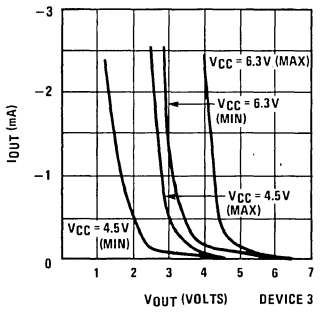

**Depletion Load OFF Source Current**



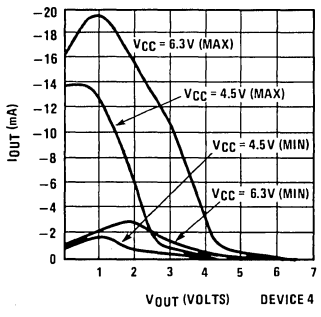

**Standard Output Source Current**



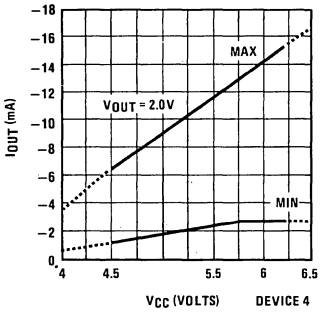

**High Drive Source Current**



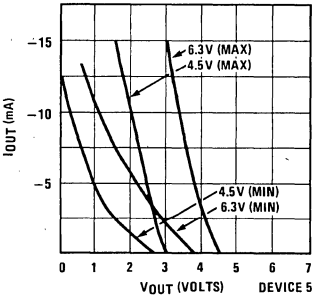

**Push-Pull Source Current**



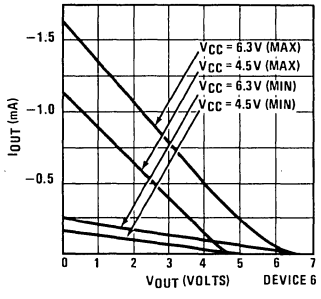

**LED Output Source Current**




**LED Output Direct LED Drive**




**TRI-STATE® Output Source Current**




**Input Load Source Current**

**Figure 11. Input/Output Characteristics**

**Table 1. COP402/402M Instruction Set Table Symbols**

| Symbol | Definition |
|---|---|
| **INTERNAL ARCHITECTURE SYMBOLS** | |
| A | 4-bit Accumulator |
| B | 6-bit RAM Address Register |
| Br | Upper 2 bits of B (register address) |
| Bd | Lower 4 bits of B (digit address) |
| C | 1-bit Carry Register |
| D | 4-bit Data Output Port |
| EN | 4-bit Enable Register |
| G | 4-bit Register to latch data for G I/O Port |
| IL | Two 1-bit Latches associated with the $IN_3$ or $IN_0$ Inputs |
| IN | 4-bit Input Port |
| IP | 8-bit Bidirectional ROM Address & Data Port |
| L | 8-bit TRI-STATE I/O Port |
| M | 4-bit contents of RAM Memory pointed to by B Register |
| P | 2-bit ROM Address Port |
| PC | 10-bit ROM Address Register (program counter) |
| Q | 8-bit Register to latch data for L I/O Port |
| SA | 10-bit Subroutine Save Register A |
| SB | 10-bit Subroutine Save Register B |
| SC | 10-bit Subroutine Save Register C |
| SIO | 4-bit Shift Register and Counter |
| SK | Logic-Controlled Clock Output |

| Symbol | Definition |
|---|---|
| **INSTRUCTION OPERAND SYMBOLS** | |
| d | 4-bit Operand Field, 0-15 binary (RAM Digit Select) |
| r | 2-bit Operand Field, 0-3 binary (RAM Register Select) |
| a | 10-bit Operand Field, 0-1023 binary (ROM Address) |
| y | 4-bit Operand Field, 0-15 binary (Immediate Data) |
| RAM(s) | Contents of RAM location addressed by s |
| ROM(t) | Contents of ROM location addressed by t |
| **OPERATIONAL SYMBOLS** | |
| + | Plus |
| − | Minus |
| → | Replaces |
| ↔ | Is exchanged with |
| = | Is equal to |
| $\overline{A}$ | The ones complement of A |
| ⊕ | Exclusive-OR |
| : | Range of values |

**Table 2. COP402/402M Instruction Set Table (Note 1)**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **ARITHMETIC INSTRUCTIONS** | | | | | | |
| ASC | | 30 | \|0 0 1 1\|0 0 0 0\| | A + C + RAM(B) → A<br>Carry → C | Carry | Add with Carry, Skip on Carry |
| ADD | | 31 | \|0 0 1 1\|0 0 0 1\| | A + RAM(B) → A | None | Add A to RAM |
| ADT | | 4A | \|0 1 0 0\|1 0 1 0\| | A + $10_{10}$ → A | None | Add Ten to A |
| AISC | y | 5- | \|0 1 0 1\| y \| | A + y → A | Carry | Add Immediate, Skip on Carry (y ≠ 0) |
| CASC | | 10 | \|0 0 0 1\|0 0 0 0\| | $\overline{A}$ + RAM(B) + C → A<br>Carry → C | Carry | Complement and Add with Carry, Skip on Carry |
| CLRA | | 00 | \|0 0 0 0\|0 0 0 0\| | 0 → A | None | Clear A |
| COMP | | 40 | \|0 1 0 0\|0 0 0 0\| | $\overline{A}$ → A | None | Ones complement of A to A |
| NOP | | 44 | \|0 1 0 0\|0 1 0 0\| | None | None | No Operation |
| RC | | 32 | \|0 0 1 1\|0 0 1 0\| | "0" → C | None | Reset C |
| SC | | 22 | \|0 0 1 0\|0 0 1 0\| | "1" → C | None | Set C |
| XOR | | 02 | \|0 0 0 0\|0 0 1 0\| | A ⊕ RAM(B) → A | None | Exclusive-OR A with RAM |

Table 2. COP402/402M Instruction Set Table (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **TRANSFER OF CONTROL INSTRUCTIONS** | | | | | | |
| JID | | FF | 1 1 1 1 \| 1 1 1 1 | ROM ($PC_{9:8}$,A,M) → $PC_{7:0}$ | None | Jump Indirect (Note 3) |
| JMP | a | 6- | 0 1 1 0 \| 0 0 \| $a_{9:8}$ | a → PC | None | Jump |
| | | -- | $a_{7:0}$ | | | |
| JP | a | -- | 1 \| $a_{6:0}$ (pages 2,3 only) | a → $PC_{6:0}$ | None | Jump within Page (Note 4) |
| | | | or | | | |
| | | -- | 1 1 \| $a_{5:0}$ (all other pages) | a → $PC_{5:0}$ | | |
| JSRP | a | -- | 1 0 \| $a_{5:0}$ | PC + 1 → SA → SB → SC<br>0010 → $PC_{9:6}$<br>a → $PC_{5:0}$ | None | Jump to Subroutine Page (Note 5) |
| JSR | a | 6- | 0 1 1 0 \| 1 0 \| $a_{9:8}$ | PC + 1 → SA → SB → a → PC | None | Jump to Subroutine |
| | | -- | $a_{7:0}$ | | | |
| RET | | 48 | 0 1 0 0 \| 1 0 0 0 | SC → SB → SA → PC | None | Return from Subroutine |
| RETSK | | 49 | 0 1 0 0 \| 1 0 0 1 | SC → SB → SA → PC | Always Skip on Return | Return from Subroutine then Skip |
| **MEMORY REFERENCE INSTRUCTIONS** | | | | | | |
| CAMQ | | 33 | 0 0 1 1 \| 0 0 1 1 | A → $Q_{7:4}$<br>RAM(B) → $Q_{3:0}$ | None | Copy A, RAM to Q |
| | | 3C | 0 0 1 1 \| 1 1 0 0 | | | |
| CQMA | | 33 | 0 0 1 1 \| 0 0 1 1 | $Q_{7:4}$ → RAM(B)<br>$Q_{3:0}$ → A | None | Copy Q to RAM, A |
| | | 2C | 0 0 1 0 \| 1 1 0 0 | | | |
| LD | r | -5 | 0 0 \| r \| 0 1 0 1 | RAM(B) → A<br>Br ⊕ r → Br | None | Load RAM into A, Exclusive-OR Br with r |
| LDD | r,d | 23 | 0 0 1 0 \| 0 0 1 1 | RAM(r,d) → A | None | Load A with RAM pointed to directly by r,d |
| | | -- | 0 0 \| r \| d | | | |
| LQID | | BF | 1 0 1 1 \| 1 1 1 1 | ROM($PC_{9:8}$,A,M) → Q<br>SB → SC | None | Load Q Indirect (Note 3) |
| RMB | 0 | 4C | 0 1 0 0 \| 1 1 0 0 | 0 → $RAM(B)_0$ | None | Reset RAM Bit |
| | 1 | 45 | 0 1 0 0 \| 0 1 0 1 | 0 → $RAM(B)_1$ | | |
| | 2 | 42 | 0 1 0 0 \| 0 0 1 0 | 0 → $RAM(B)_2$ | | |
| | 3 | 43 | 0 1 0 0 \| 0 0 1 1 | 0 → $RAM(B)_3$ | | |
| SMB | 0 | 4D | 0 1 0 0 \| 1 1 0 1 | 1 → $RAM(B)_0$ | None | Set RAM Bit |
| | 1 | 47 | 0 1 0 0 \| 0 1 1 1 | 1 → $RAM(B)_1$ | | |
| | 2 | 46 | 0 1 0 0 \| 0 1 1 0 | 1 → $RAM(B)_2$ | | |
| | 3 | 4B | 0 1 0 0 \| 1 0 1 1 | 1 → $RAM(B)_3$ | | |

**Table 2. COP402/402M Instruction Set Table (continued)**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| MEMORY REFERENCE INSTRUCTIONS (continued) | | | | | | |
| STII | y | 7- | \|0 1 1 1\| y \| | y → RAM(B)<br>Bd + 1 → Bd | None | Store Memory Immediate and Increment Bd |
| X | r | -6 | \|0 0\| r \|0 1 1 0\| | RAM(B) ⟷ A<br>Br ⊕ r → Br | None | Exchange RAM with A, Exclusive-OR Br with r |
| XAD | r,d | 23 | \|0 0 1 0\|0 0 1 1\| | RAM(r,d) ⟷ A | None | Exchange A with RAM pointed to directly by r,d |
| | | -- | \|1 0\| r \| d \| | | | |
| XDS | r | -7 | \|0 0\| r \|0 1 1 1\| | RAM(B) ⟷ A<br>Bd − 1 → Bd<br>Br ⊕ r → Br | Bd decrements past 0 | Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r |
| XIS | r | -4 | \|0 0\| r \|0 1 0 0\| | RAM(B) ⟷ A<br>Bd + 1 → Bd<br>Br ⊕ r → Br | Bd increments past 15 | Exchange RAM with A and Increment Bd, Exclusive-OR Br with r |
| REGISTER REFERENCE INSTRUCTIONS | | | | | | |
| CAB | | 50 | \|0 1 0 1\|0 0 0 0\| | A → Bd | None | Copy A to Bd |
| CBA | | 4E | \|0 1 0 0\|1 1 1 0\| | Bd → A | None | Copy Bd to A |
| LBI | r,d | -- | \|0 0\| r \| (d − 1)\|<br>(d = 0, 9:15)<br>or | r,d → B | Skip until not a LBI | Load B Immediate with r,d (Note 6) |
| | | 33 | \|0 0 1 1\|0 0 1 1\| | | | |
| | | -- | \|1 0\| r \| d \|<br>(any d) | | | |
| LEI | y | 33 | \|0 0 1 1\|0 0 1 1\| | y → EN | None | Load EN Immediate (Note 7) |
| | | 6- | \|0 1 1 0\| y \| | | | |
| XABR | | 12 | \|0 0 0 1\|0 0 1 0\| | A ⟷ Br (0,0 → $A_3$,$A_2$) | None | Exchange A with Br |
| TEST INSTRUCTIONS | | | | | | |
| SKC | | 20 | \|0 0 1 0\|0 0 0 0\| | | C = "1" | Skip if C is True |
| SKE | | 21 | \|0 0 1 0\|0 0 0 1\| | | A = RAM(B) | Skip if A Equals RAM |
| SKGZ | | 33 | \|0 0 1 1\|0 0 1 1\| | | $G_{3:0}$ = 0 | Skip if G is Zero (all 4 bits) |
| | | 21 | \|0 0 1 0\|0 0 0 1\| | | | |
| SKGBZ | | 33 | \|0 0 1 1\|0 0 1 1\| | 1st byte | | Skip if G Bit is Zero |
| | 0 | 01 | \|0 0 0 0\|0 0 0 1\| | 2nd byte | $G_0$ = 0 | |
| | 1 | 11 | \|0 0 0 1\|0 0 0 1\| | | $G_1$ = 0 | |
| | 2 | 03 | \|0 0 0 0\|0 0 1 1\| | | $G_2$ = 0 | |
| | 3 | 13 | \|0 0 0 1\|0 0 1 1\| | | $G_3$ = 0 | |
| SKMBZ | 0 | 01 | \|0 0 0 0\|0 0 0 1\| | | $RAM(B)_0$ = 0 | Skip if RAM Bit is Zero |
| | 1 | 11 | \|0 0 0 1\|0 0 0 1\| | | $RAM(B)_1$ = 0 | |
| | 2 | 03 | \|0 0 0 0\|0 0 1 1\| | | $RAM(B)_2$ = 0 | |
| | 3 | 13 | \|0 0 0 1\|0 0 1 1\| | | $RAM(B)_3$ = 0 | |
| SKT | | 41 | \|0 1 0 0\|0 0 0 1\| | | A time-base counter carry has occurred since last test | Skip on Timer (Note 3) |

**Table 2. COP402/402M Instruction Set Table (continued)**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| INPUT/OUTPUT INSTRUCTIONS | | | | | | |
| ING | | 33 | \|0 0 1 1\|0 0 1 1\| | G → A | None | Input G Ports to A |
| | | 2A | \|0 0 1 0\|1 0 1 0\| | | | |
| ININ | | 33 | \|0 0 1 1\|0 0 1 1\| | IN → A | None | Input IN Inputs to A (Notes 2 and 8) |
| | | 28 | \|0 0 1 0\|1 0 0 0\| | | | |
| INIL | | 33 | \|0 0 1 1\|0 0 1 1\| | $IL_3$,"1","0",$IL_0$ → A | None | Input IL Latches to A (Notes 2 and 3) |
| | | 29 | \|0 0 1 0\|1 0 0 1\| | | | |
| INL | | 33 | \|0 0 1 1\|0 0 1 1\| | $L_{7:4}$ → RAM(B)<br>$L_{3:0}$ → A | None | Input L Ports to RAM, A |
| | | 2E | \|0 0 1 0\|1 1 1 0\| | | | |
| OBD | | 33 | \|0 0 1 1\|0 0 1 1\| | Bd → D | None | Output Bd to D Outputs |
| | | 3E | \|0 0 1 1\|1 1 1 0\| | | | |
| OGI | y | 33 | \|0 0 1 1\|0 0 1 1\| | y → G | None | Output to G Ports Immediate |
| | | 5- | \|0 1 0 1\| y \| | | | |
| OMG | | 33 | \|0 0 1 1\|0 0 1 1\| | RAM(B) → G | None | Output RAM to G Ports |
| | | 3A | \|0 0 1 1\|1 0 1 0\| | | | |
| XAS | | 4F | \|0 1 0 0\|1 1 1 1\| | A ⟷ SIO, C → SKL | None | Exchange A with SIO (Note 3) |

**Note 1:** All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, $A_3$ indicates the most significant (left-most) bit of the 4-bit A register.

**Note 2:** The ININ instruction is not available on the 24-pin COP421 since this device does not contain the IN inputs.

**Note 3:** For additional information on the operation of the XAS, JID, LQID, INIL, and SKT instructions, see below.

**Note 4:** The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

**Note 5:** A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

**Note 6:** LBI is a single-byte instruction if d = 0, 9, 10, 11, 12, 13, 14, or 15. The machine code for the lower 4 bits equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 ($1001_2$), the lower 4 bits of the LBI instruction equal 8 ($1000_2$). To load 0, the lower 4 bits of the LBI instruction should equal 15 ($1111_2$).

**Note 7:** Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register.)

**Note 8:** COP402M will always read a "1" into A1 with the ININ instruction.

### INIL Instruction

INIL (Input IL Latches to A) inputs 2 latches, $IL_3$ and $IL_0$ (see figure 12) and CKO into A. The $IL_3$ and $IL_0$ latches are set if a low-going pulse ("1" to "0") has occurred on the $IN_3$ and $IN_0$ inputs since the last INIL instruction, provided the input pulse stays low for at least two instruction times. Execution of an INIL inputs $IL_3$ and $IL_0$ into A3 and A0 respectively, and resets these latches to allow them to respond to subsequent low-going pulses on the $IN_3$ and $IN_0$ lines. If CKO is mask programmed as a general purpose input, an INIL will input the state of CKO into A2. If CKO has not been so programmed, a "1" will be placed in A2. A "0" is always placed in A1 upon the execution of an INIL. The general purpose inputs $IN_3$-$IN_0$ are input to A upon the execution of an ININ instruction. (See table 2, ININ Instruction.) INIL is useful in recognizing pulses of short duration or pulses which occur too often to be read conveniently by an ININ instruction.

### LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 10-bit word $PC_9$, $PC_8$, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB → SC) and replaces the least significant 8 bits of PC as follows: A → $PC_{7:4}$, RAM(B) → $PC_{3:0}$, leaving $PC_9$ and $PC_8$ unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped"(SC → SB → SA → PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SB → SC, the previous contents of SC are lost. Also, when LQID pops the stack, the previously pushed contents of SB are left in SC. The net result is that the contents of SB are placed in SC (SB → SC). Note that LQID takes two instruction cycle times to execute.

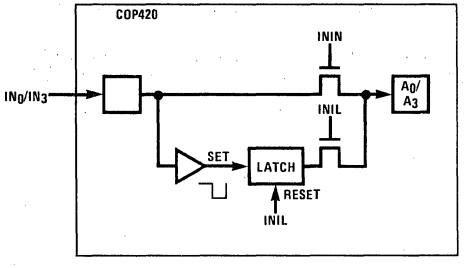



**Figure 12. $IN_0/IN_3$ Latches**

### SKT Instruction

The SKT (Skip on Timer) instruction tests the state of an internal 10-bit time-base counter. This counter divides the instruction cycle clock frequency by 1024 and provides a latched indication of counter overflow. The SKT instruction tests this latch, executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction, therefore, allow the controller to generate its own time-base for real-time processing rather than relying on an external input signal.

For example, using a 2.097MHz crystal as the time-base to the clock generator, the instruction cycle clock frequency will be 131kHz (crystal frequency ÷ 16) and the binary counter output pulse frequency will be 128Hz. For time-of-day or similar real-time processing, the SKT instruction can call a routine which increments a "seconds" counter every 128 ticks.

### Instruction Set Notes

a. The first word of a program (ROM address 0) must be a CLRA (Clear A) instruction.

b. Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths take the same number of cycle times whether instructions are skipped or executed.

c. The ROM is organized into 16 pages of 64 words each. The Program Counter is a 10-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example: a JP located in the last word of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3, 7, 11, or 15 will access data in the next group of 4 pages.

## TYPICAL APPLICATIONS

### PROM-Based System

The COP402 may be used to exactly emulate the COP420. Figure 13 shows the interconnect to implement a COP420 hardware emulation. This connection uses two MM5204 EPROMs as external memory. Other memory can be used such as bipolar PROM or RAM.

Pins IP7-IP0 are bidirectional inputs and outputs. When the AD/$\overline{\text{DATA}}$ clocking output turns on, the EPROM drivers are disabled and IP7-IP0 output addresses. The 8-bit latch (MM74C373) latches the addresses to drive the memory.

When AD/$\overline{\text{DATA}}$ turns off, the EPROMs are enabled and the IP7-IP0 pins will input the memory data. P8 and P9 output the most significant address bits to the memory. (SKIP output may be used for program debug if needed.)

The other 28 pins of the COP402 may be configured exactly the same as a COP420. The COP402M chip can be used if the MICROBUS™ feature of the COP420 is needed.

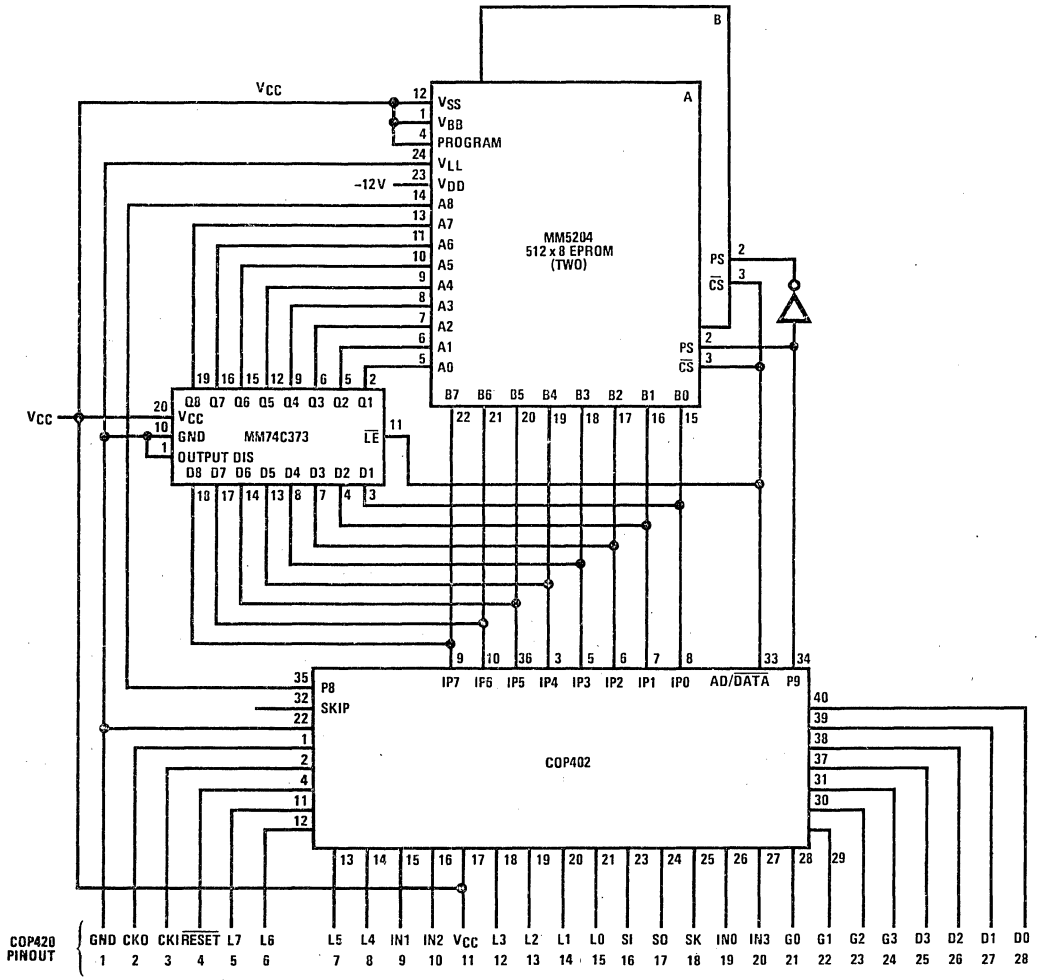



**Figure 13. COP402 Used to Emulate a COP420**

### RAM-Based System

If the user desires more program flexibility than is available with PROM, a RAM memory system may be constructed as outlined below; data and addresses are entered by switch, and data may be reviewed on LED indicators.

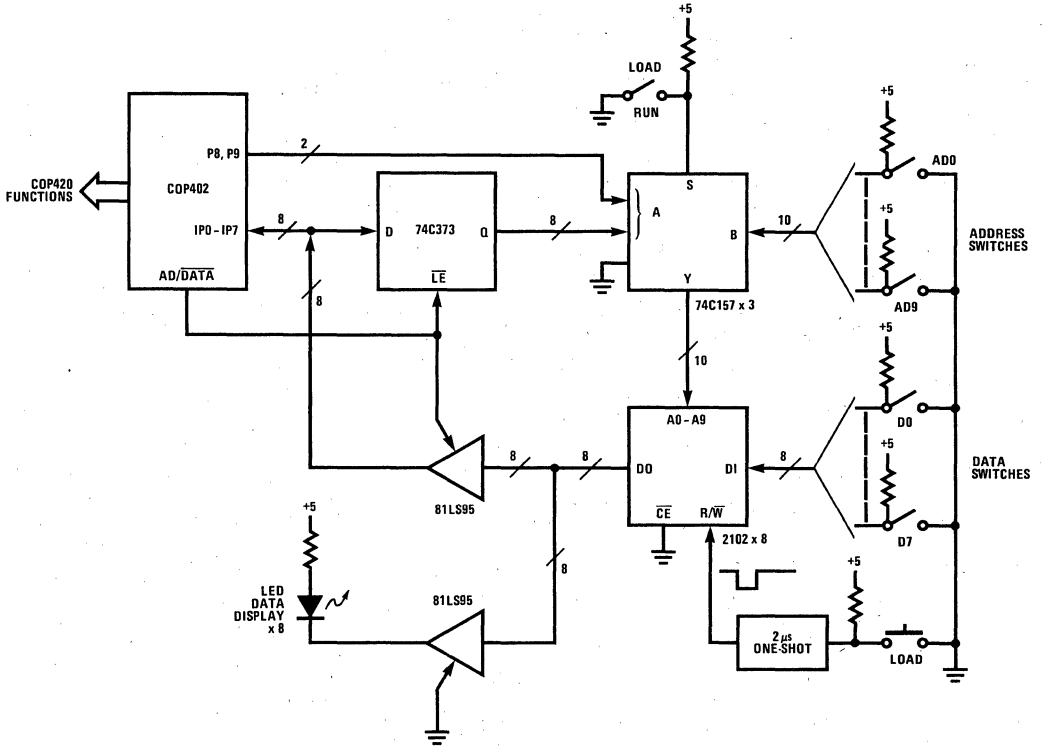



**Figure 14. RAM-Based External Memory for COP402**

## COP402 MASK OPTIONS

### COP402 Mask Options

The following COP420 options have been implemented in this basic version of the COP402. Subsequent versions of the COP402 will implement different combinations of available options; such versions will be identified as COP402-A, COP402-B, etc.

| Option Value | Comment |
|---|---|
| Option 1 = 0 | Ground Pin — no option available |
| Option 2 = 0 | CKO is clock generator output to crystal |
| Option 3 = 0 | CKI is crystal input ÷16 (may be overridden externally) |
| Option 4 = 0 | RESET pin has load device to $V_{CC}$ |
| Option 5 = 2 (402) | L7 has LED direct-drive output |
| = 3 (402M) | L7 has TRI-STATE® push-pull output |
| Option 6 = 2,3 | L6 same as L7 |
| Option 7 = 2,3 | L5 same as L7 |
| Option 8 = 2,3 | L4 same as L7 |
| Option 9 = 0 (402) | IN1 has load device to $V_{CC}$ |
| = 1 (402M) | Hi Z |
| Option 10 = 0 (402) | IN2 has load device to $V_{CC}$ |
| = 1 (402M) | Hi Z |
| Option 11 = 0 | $V_{CC}$ pin — no option available |
| Option 12 = 2,3 | L3 same as L7 |
| Option 13 = 2,3 | L2 same as L7 |
| Option 14 = 2,3 | L1 same as L7 |
| Option 15 = 2,3 | L0 same as L7 |
| Option 16 = 0 | SI has load device to $V_{CC}$ |
| Option 17 = 2 | SO has push-pull output |
| Option 18 = 2 | SK has push-pull output |
| Option 19 = 0 | IN0 has load device to $V_{CC}$ |
| Option 20 = 0 (402) | IN3 has load device to $V_{CC}$ |
| = 1 (402M) | Hi Z |
| Option 21 = 0 | G0 has standard output |
| Option 22 = 0 | G1 same as G0 |
| Option 23 = 0 | G2 same as G0 |
| Option 24 = 0 | G3 same as G0 |
| Option 25 = 0 | D3 has standard output |
| Option 26 = 0 | D2 same as D3 |
| Option 27 = 0 | D1 same as D3 |
| Option 28 = 0 | D0 same as D3 |
| Option 29 = 0 (402) | normal operation |
| = 1 (402M) | MICROBUS™ operation |
| Option 30 = N/A | 40-pin package |

National Semiconductor

PRELIMINARY

# COP404L ROMless N-Channel Microcontroller

## General Description

The COP404L ROMless Microcontroller is a member of the COPS™ family, fabricated using N-channel, silicon gate MOS technology. The COP404L contains CPU, RAM, I/O and is identical to a COP444L device except the ROM has been removed and pins have been added to output the ROM address and to input the ROM data. In a system the COP404L will perform exactly as the COP444L. This important benefit facilitates development and debug of a COP program prior to masking the final part. The COP404L is also appropriate in low volume applications, or when the program might be changing. The COP404L may be used to emulate the COP444L, COP445L, COP420L, and the COP421L

## Features

- Exact circuit equivalent of COP444L
- Low cost
- Powerful instruction set
- 128 × 4 RAM, addresses 2048 × 8 ROM
- True vectored interrupt, plus restart
- Three-level subroutine stack
- 15μs instruction time
- Single supply operation (4.5-9.5V)
- Low current drain (15mA max @ 5V)
- Internal time-base counter for real-time processing
- Internal binary counter register with MICROWIRE™ compatible serial I/O
- General purpose and TRI-STATE® outputs
- LSTTL/CMOS compatible in and out
- Direct drive of LED digit and segment lines
- Software/hardware compatible with other members of COP400 family

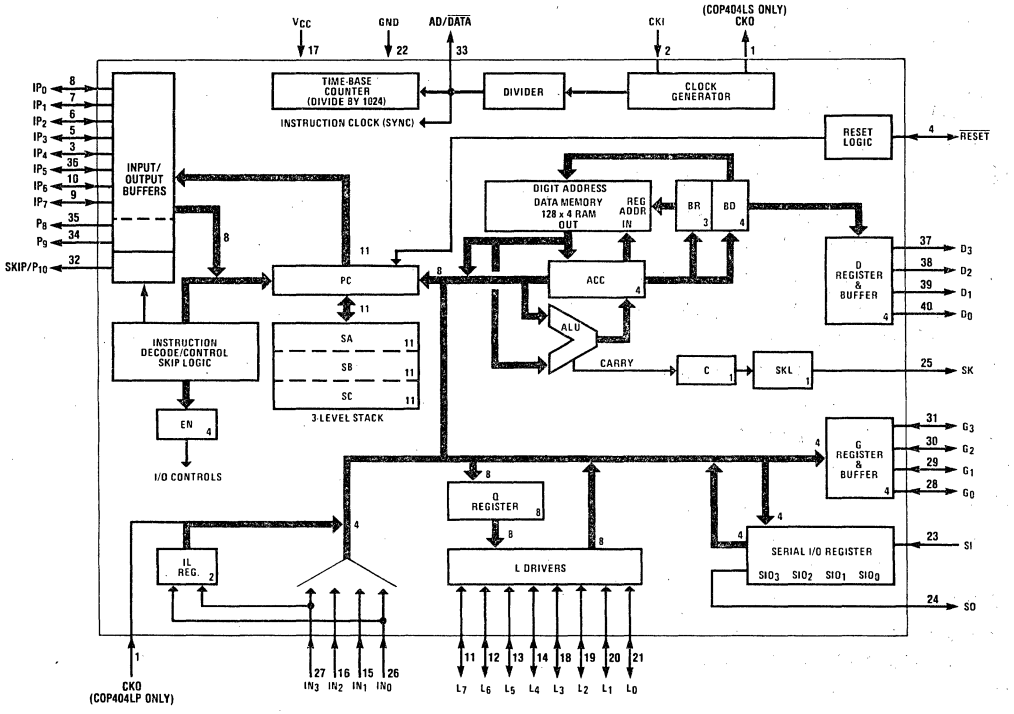



Figure 1. COP404L Block Diagram

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Relative to GND | −0.3V to +10V |
| Ambient Operating Temperature | 0°C to +70°C |
| Ambient Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |
| Power Dissipation | 0.75 Watt at 25°C<br>0.4 Watt at 70°C |

*Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.*

## DC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Operating Voltage ($V_{CC}$) | | | | |
| 404LS | (Note 2) | 4.5 | 9.5 | V |
| 404LP | $\overline{RESET}$ open (Note 2) | 7.5 | 9.5 | V |
| | $R_L = 510\Omega$ (Note 2) | 4.5 | 9.5 | V |
| Operating Supply Current | $V_{CC} = 5V$, $T_A = 25°C$ (all inputs and outputs open) | | 15 | mA |
| $V_{CC}$ Power-Low Failsafe Trip Level (404LS) | $\overline{RESET}$ open | 4.5 | 7.5 | V |
| Input Voltage Levels | | | | |
| CKI Input Levels | | | | |
| Crystal Input | | | | |
| Logic High ($V_{IH}$) | $V_{CC} = 9.5V$ | 3.0 | | V |
| Logic High ($V_{IH}$) | $V_{CC} = 5V \pm 10\%$ | 2.0 | | V |
| Logic Low ($V_{IL}$) | | | 0.4 | V |
| Schmitt Trigger Input (÷4) | | | | |
| Logic High ($V_{IH}$) | | 0.7 $V_{CC}$ | | V |
| Logic Low ($V_{IL}$) | | | 0.6 | V |
| $\overline{RESET}$ Input Levels | | | | |
| Logic High | | 0.7 $V_{CC}$ | | V |
| Logic Low | | | 0.6 | V |
| $\overline{RESET}$ Hysteresis | | 1.0 | | V |
| SO Input Level (Test mode) | | 2.0 | 3.0 | V |
| All Other Inputs | | | | |
| Logic High | $V_{CC} = 9.5V$ | 3.0 | | V |
| Logic High | with TTL trip level options | 2.0 | | V |
| Logic Low | selected, $V_{CC} = 5V \pm 10\%$ | | 0.8 | V |
| Logic High | with high trip level options | 3.6 | | V |
| Logic Low | selected | | 1.2 | V |
| Output Voltage Levels | | | | |
| LSTTL Operation | $V_{CC} = 5V \pm 5\%$ | | | |
| Logic High ($V_{OH}$) | $I_{OH} = -25\mu A$ | 2.7 | | V |
| Logic Low ($V_{OL}$) | $I_{OL} = 0.36mA$ | | 0.4 | V |
| IP0-IP7, P8, P9, SKIP/P10 | | | | |
| Output Voltage Levels | $R_L = 15k\Omega$ (Note 1) | | | |
| Logic High | $I_{OH} = -100\mu A$ | 2.7 | | V |
| Logic Low | $I_{OL} = 1.6mA$ | | 0.4 | V |
| Output Current Levels | | | | |
| Output Sink Current | | | | |
| SO and SK Outputs ($I_{OL}$) | $V_{CC} = 9.5V$, $V_{OL} = 1.0V$ | 4.5 | 22 | mA |
| | $V_{CC} = 4.5V$, $V_{OL} = 1.0V$ | 2.2 | 11 | mA |
| $L_0$-$L_7$ Outputs | $V_{CC} = 9.5V$, $V_{OL} = 1.0V$ | 2.0 | 9.0 | mA |
| | $V_{CC} = 4.5V$, $V_{OL} = 1.0V$ | 1.0 | 4.5 | mA |
| $G_0$-$G_3$ and $D_0$-$D_3$ Outputs | $V_{CC} = 9.5V$, $V_{OL} = 1.0V$ | 30 | 150 | mA |
| | $V_{CC} = 4.5V$, $V_{OL} = 1.0V$ | 15 | 75 | mA |
| $\overline{RESET}$ Output (COP404LP) | $V_{CC} = 4.5V$, $V_{OL} = 1.0V$ | 250 | | μA |

## DC Electrical Characteristics (continued) $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Output Source Current: | | | | |
| Standard Configuration, All Outputs ($I_{OH}$) | $V_{CC} = 9.5V$, $V_{OH} = 4.75V$ | −70 | −450 | μA |
| | $V_{CC} = 4.5V$, $V_{OH} = 2.25V$ | −26 | −190 | μA |
| Push-Pull Configuration, SO and SK Outputs ($I_{OH}$) | $V_{CC} = 9.5V$, $V_{OH} = 4.75V$ | −1.45 | −15.5 | mA |
| | $V_{CC} = 4.5V$, $V_{OH} = 2.25V$ | −0.07 | −2.8 | mA |
| $L_0$-$L_7$ Outputs | $V_{CC} = 9.5V$, $V_{OH} = 2.0V$ | −3.0 | −30 | mA |
| | $V_{CC} = 6.0V$, $V_{OH} = 2.0V$ | −3.0 | −20 | mA |
| TRI-STATE® Output Leakage Current | | −10 | +10 | μA |

## AC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise specified.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Instruction Cycle Time | | 15 | 40 | μs |
| CKI | | | | |
| Input Frequency $f_i$ | (÷32 mode) | 0.8 | 2.097 | MHz |
| Duty Cycle | | 30 | 60 | % |
| Rise Time | $f_i = 2.097$ MHz | | 120 | ns |
| Fall Time | | | 80 | ns |
| INPUTS: | | | | |
| SI, IP7-IP0 | | | | |
| $t_{SETUP}$ | | 2.0 | | μs |
| $t_{HOLD}$ | | 1.0 | | μs |
| $IN_3$-$IN_0$, $G_3$-$G_0$, $L_7$-$L_0$ | | | | |
| $t_{SETUP}$ | | 4.0 | | μs |
| $t_{HOLD}$ | | 1.0 | | μs |
| OUTPUTS: | | | | |
| COP TO CMOS PROPAGATION DELAY | $V_{OH} = 0.7\ V_{CC}$, $V_{OL} = 0.3\ V_{CC}$, $C_L$ = 50pF | | | |
| SO, SK Outputs | | | | |
| $t_{pd1}$ | | | 4.0 | μs |
| $t_{pd0}$ | | | 1.2 | μs |
| $D_3$-$D_0$, $G_3$-$G_0$, $L_7$-$L_0$ | | | | |
| $t_{pd1}$ | | | 6.5 | μs |
| $t_{pd0}$ | | | 3.0 | μs |
| IP7-IP0, P8, P9, SKIP/P10 | $R_L = 15k\Omega$ (Note 1) | | | |
| $t_{pd1}$ | | | 7.0 | μs |
| $t_{pd0}$ | | | 7.0 | μs |
| COP TO LSTTL PROPAGATION DELAY | $V_{CC} = 5V \pm 5\%$, $V_{OH} = 2.7V$, $V_{OL} = 0.4V$, $C_L = 50pF$ | | | |
| SO, SK Outputs | | | | |
| $t_{pd1}$ | | | 3.5 | μs |
| $t_{pd0}$ | | | 3.0 | μs |
| $D_3$-$D_0$, $G_3$-$G_0$, $L_7$-$L_0$ | | | | |
| $t_{pd1}$ | | | 5.0 | μs |
| $t_{pd0}$ | | | 5.0 | μs |
| IP7-IP0, P8, P9, SKIP/P10 | $R_L = 15k\Omega$ (Note 1) | | | |
| $t_{pd1}$ | | | 7.0 | μs |
| $t_{pd0}$ | | | 7.0 | μs |

**Note 1:** Pull-up resistors required on COP404LP *only*; COP404LS has Push-Pull drivers on these outputs.

**Note 2:** $V_{CC}$ voltage change must be less than 0.5V/ms to maintain proper operation.

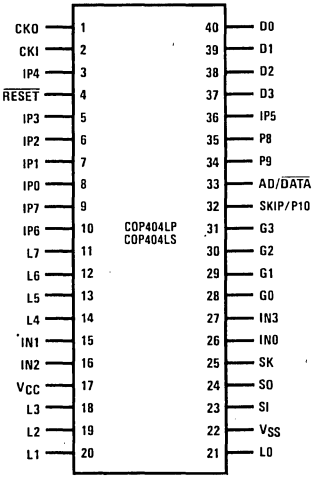



Figure 2. Connection Diagram

| Pin | Description | Pin | Description |
|---|---|---|---|
| $L_7$-$L_0$ | 8 bidirectional I/O ports with TRI-STATE® | CKI | System oscillator input |
| $G_3$-$G_0$ | 4 bidirectional I/O ports | CKO | General purpose input (COP404LP) System oscillator output (COP404LS) |
| $D_3$-$D_0$ | 4 general purpose outputs | $\overline{RESET}$ | System reset input |
| $IN_3$-$IN_0$ | 4 general purpose inputs | $V_{CC}$ | Power supply |
| SI | Serial input (or counter input) | GND | Ground |
| SO | Serial output (or general purpose output) | IP7-IP0 | 8 bidirectional ROM address and data ports |
| SK | Logic-controlled clock (or general purpose output) | P8, P9 | 2 ROM address outputs |
| AD/$\overline{DATA}$ | Address out/data in flag | SKIP/P10 | Instruction skip output and ROM address output |

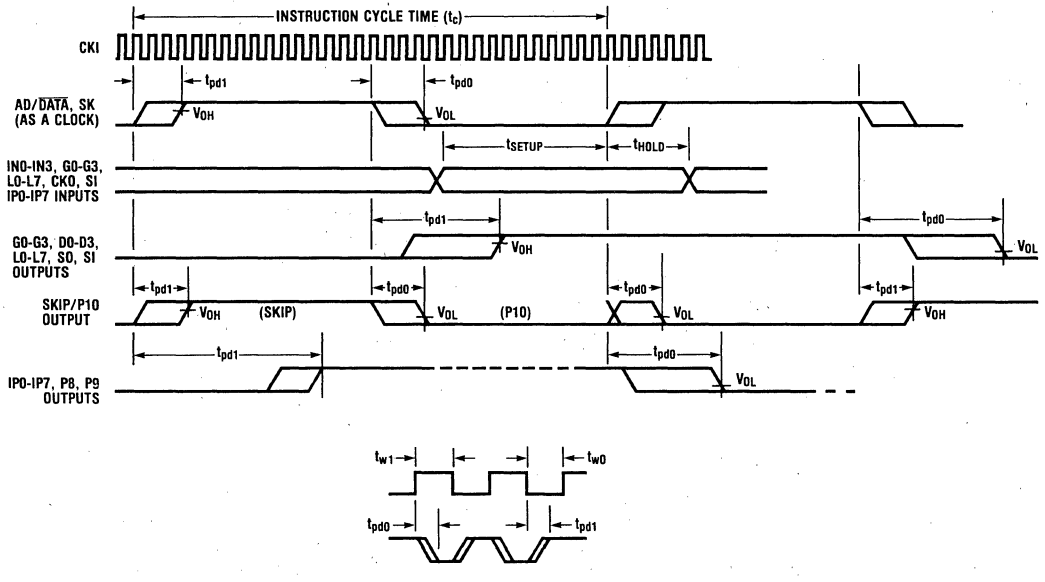



Figure 3. Input/Output Timing Diagram

## FUNCTIONAL DESCRIPTION

A block diagram of the COP404L is given in figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1" (greater than 2 volts). When a bit is reset, it is a logic "0" (less than 0.8 volts).

### Program Memory

Program Memory consists of a 2048 byte external memory. As can be seen by an examination of the COP404L instruction set, these words may be program instructions, program data or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID and LQID instructions, ROM must often be thought of as being organized into 32 pages of 64 words each.

ROM addressing is accomplished by a 11-bit PC register. Its binary value selects one of the 2048 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 11-bit binary count value. Three levels of subroutine nesting are implemented by the 11-bit subroutine save registers, SA, SB and SC, providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the Instruction Decode, Control and Skip Logic circuitry.

### Data Memory

Data memory consists of a 512-bit RAM, organized as 8 data registers of 16 4-bit digits. RAM addressing is implemented by a 7-bit B register whose upper 3 bits (Br) select 1 of 8 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) is usually loaded into or from, or exchanged with, the A register (accumulator), it may also be loaded into or from the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the 7-bit contents of the operand field of these instructions. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

### Internal Logic

The 4-bit A register (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd portions of the B register, to load and input 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

A 4-bit adder performs the arithmetic and logic functions, storing its results in A. It also outputs a carry bit to the 1-bit C register, most often employed to indicate arithmetic overflow. The C register, in conjunction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SK or can enable SK to be a sync clock each instruction cycle time. (See XAS instruction and EN register description, below.)

Four general-purpose inputs, $IN_3$-$IN_0$, are provided.

The D register provides 4 general-purpose outputs and is used as the destination register for the 4-bit contents of Bd. The D outputs can be directly connected to the digits of a multiplexed LED display.

The G register contents are outputs to 4 general-purpose bidirectional I/O ports. G I/O ports can be directly connected to the digits of a multiplexed LED display.

The Q register is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control. (See LEI instruction.)

The 8 L drivers,when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the LED Direct Drive output configuration option) with Q data being outputted to the Sa-Sg and decimal point segments of the display.

The SIO register functions as a 4-bit serial-in/serial-out shift register or as a binary counter depending on the contents of the EN register. (See EN register description, below.) Its contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O by connecting SO to external serial-in/parallel-out shift registers.

The XAS instruction copies C into the SKL latch. In the counter mode, SK is the output of SKL; in the shift register mode, SK outputs SKL ANDed with the clock.

The EN register is an internal 4-bit register loaded under program control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register ($EN_3$-$EN_0$).

1. The least significant bit of the enable register, $EN_0$, selects the SIO register as either a 4-bit shift register or a 4-bit binary counter. With $EN_0$ set, SIO is an asynchronous binary counter, *decrementing* its value by one upon each low-going pulse ("1" to "0") ocurring on the SI input. Each pulse must be at least two instruction cycles wide. SK outputs the value of SKL. The SO output is equal to the value of $EN_3$. With $EN_0$ reset, SIO is a serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time. (See 4 below.) The SK output becomes a logic-controlled clock.
2. With $EN_1$ set the $IN_1$ input is enabled as an interrupt input. Immediately following an interrupt, $EN_1$ is reset to disable further interrupts.
3. With $EN_2$ set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting $EN_2$ disables the L drivers, placing the L I/O ports in a high-impedance input state.

4. $EN_3$, in conjunction with $EN_0$, affects the SO output. With $EN_0$ set (binary counter option selected) SO will output the value loaded into $EN_3$. With $EN_0$ reset (serial shift register option selected), setting $EN_3$ enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting $EN_3$ with the serial shift register option selected disables SO as the shift register output; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0." The table below provides a summary of the modes associated with $EN_3$ and $EN_0$.

### Interrupt

The following features are associated with the $IN_1$ interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

a. The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1) onto the stack, pushing in turn the contents of the other subroutine-save registers to the next lower level (PC + 1 → SA → SB → SC). Any previous contents of SC are lost. The program counter is set to hex address OFF (the last word of page 3) and $EN_1$ is reset.

b. An interrupt will be acknowledged only after the following conditions are met:
   1. $EN_1$ has been set.
   2. A low-going pulse ("1" to "0") at least two instruction cycles wide occurs on the $IN_1$ input.
   3. A currently executing instruction has been completed.
   4. All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction, the interrupt will not be acknowledged until the second JP instruction has been executed.

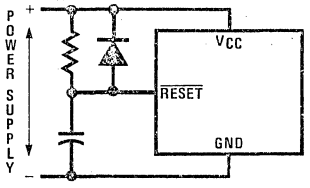
c. Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon popping of the stack. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skip logic status is saved and program control is transferred to the interrupt servicing routine at hex address OFF. At the *end* of the interrupt routine, a RET instruction is executed to "pop" the stack and return program control to the instruction following the original ASC. *At this time*, the skip logic is enabled and skips this instruction because of the previous ASC carry. Subroutines and LQID instructions should not be nested within the interrupt service routine, since their popping the stack will enable any previously saved main program skips, interfering with the orderly execution of the interrupt routine.

d. The first instruction of the interrupt routine at hex address OFF must be a NOP.

e. A LEI instruction can be put immediately before the RET to re-enable interrupts.

### Initialization

The Reset Logic will initialize (clear) the device upon power-up if the power supply rise time is less than 1ms and greater than 1$\mu$s. If the power supply rise time is greater than 1ms, the user must provide an external RC network and diode to the $\overline{RESET}$ pin as shown below. The $\overline{RESET}$ pin is configured as a Schmitt trigger input. If not used it should be connected to $V_{CC}$. Initialization will occur whenever a logic "0" is applied to the $\overline{RESET}$ input, provided it stays low for at least three instruction cycle times.




RC ≥ 5 x POWER SUPPLY RISE TIME

NOTE: IF POWER-LOW FAILSAFE OPTION IS SELECTED, THE RC AND DIODE CIRCUIT IS NOT USED

**Enable Register Modes — Bits $EN_3$ and $EN_0$**

| $EN_3$ | $EN_0$ | SIO | SI | SO | SK |
|---|---|---|---|---|---|
| 0 | 0 | Shift Register | Input to Shift Register | 0 | If SKL = 1, SK = CLOCK<br>If SKL = 0, SK = 0 |
| 1 | 0 | Shift Register | Input to Shift Register | Serial Out | If SKL = 1, SK = CLOCK<br>If SKL = 0, SK = 0 |
| 0 | 1 | Binary Counter | Input to Binary Counter | 0 | If SKL = 1, SK = 1<br>If SKL = 0, SK = 0 |
| 1 | 1 | Binary Counter | Input to Binary Counter | 1 | If SKL = 1, SK = 1<br>If SKL = 0, SK = 0 |

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, and G registers are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. *Data Memory (RAM) is not cleared upon initialization.* The first instruction at address 0 must be a CLRA.

### External Memory Interface

The COP404L is designed for use with an external Program Memory. This memory may be implemented using any devices having the following characteristics:

1. random addressing
2. TTL-compatible TRI-STATE® outputs
3. TTL-compatible inputs
4. access time = 5μs max

Typically these requirements are met using bipolar or MOS PROMs.

During operation, the address of the next instruction is sent out on P10, P9, P8, and IP7 through IP0 during the time that AD/$\overline{\text{DATA}}$ is high (logic "1" = address mode). Address data on the IP lines is stored into an external latch on the high-to-low transition of the AD/$\overline{\text{DATA}}$ line; P9 and P8 are dedicated address outputs, and do not need to be latched. SKIP/P10 outputs address data when AD/$\overline{\text{DATA}}$ is low. When AD/$\overline{\text{DATA}}$ is low (logic "0" = data mode), the output of the memory is gated onto IP7 through IP0, forming the input bus. Note that the AD/$\overline{\text{DATA}}$ output has a period of one instruction time, a duty cycle of approximately 50%, and specifies whether the IP lines are used for address output or instruction input.

### Oscillator

Two basic clock oscillator configurations have been implemented, as shown in figure 4.

**a. Crystal Controlled Oscillator** (COP404LS only). CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency divided by 32

**b. External Oscillator** (COP404LP only). CKI is an external clock input signal. The external frequency is divided by 32 to give the instruction cycle time. CKO is used as a general purpose input.

### CKO as an Input

On the COP404LP, CKO has been configured as a general-purpose input. The logic level applied to CKO will be read into bit 2 of A (accumulator) upon execution of an INIL instruction.

### Input/Output Configurations

COP404L outputs have the following configurations, illustrated in figure 5:

**a. Standard** — an enhancement mode device to ground in conjunction with a depletion-mode device to $V_{CC}$, compatible with LSTTL and CMOS input requirements. (Used on D and G outputs.)

**b. Open-Drain** — an enhancement-mode device to ground only, allowing external pull-up as required by the user's application. (Used on IP, P and SKIP/P10 outputs on COP404LP *only*).

**c. Push-Pull** — An enhancement-mode device to ground in conjunction with a depletion-mode device paralleled by an enhancement-mode device to $V_{CC}$. This configuration has been provided to allow for fast rise and fall times when driving capacitive loads. (Used on SO and SK outputs on COP404LP and 404LS; also used on IP, P and SKIP/P10 outputs on COP404LS *only*.)

**d. LED Direct Drive** — an enhancement-mode device to ground and to $V_{CC}$, meeting the typical current sourcing requirements of the segments of an LED display. The sourcing device is clamped to limit current flow. These devices may be turned off under program control (See Functional Description, EN Register), placing the outputs in a high-impedance state to provide required LED segment blanking for a multiplexed display. (Used on L outputs).

COP404L inputs have an on-chip depletion load device to $V_{CC}$.

The above input and output configurations share common enhancement-mode and depletion-mode devices. Specifically, all configurations use one or more of six devices (numbered 1-6, respectively). Minimum and maximum current ($I_{OUT}$ and $V_{OUT}$) curves are given in figure 6 for each of these devices to allow the designer to effectively use these I/O configurations in designing a system.

An important point to remember is that even when the L drivers are disabled, the depletion load device will source a small amount of current (see figure 6, device 2); however, when the L-lines are used as inputs, the disabled depletion device can *not* be relied on to source sufficient current to pull an input to a logic "1".

### Power-Low Failsafe Option (COP404LP only)

If the power supply voltage drops, an on-chip level detection circuit will force the $\overline{\text{RESET}}$ pin low and reset the chip while the power supply is still within the operating range. Reset will occur with $V_{CC}$ between 4.5 and 7.5 volts, allowing normal system operation between 7.5 and 9.5 volts. $\overline{\text{RESET}}$ is an output in this mode and can drive other circuits.

This feature, implemented on the COP404LP *only*, can be overridden by connecting $\overline{\text{RESET}}$ to $V_{CC}$ through a 510Ω resistor.

### COP404LP and COP404LS

Two versions of the basic COP404L have been implemented: the COP404LP, with open-drain memory interface drivers, is used only in the COP400-E04L Emulator Card; the COP404LS, with push-pull memory interface, is intended for use in small to medium volume production applications.

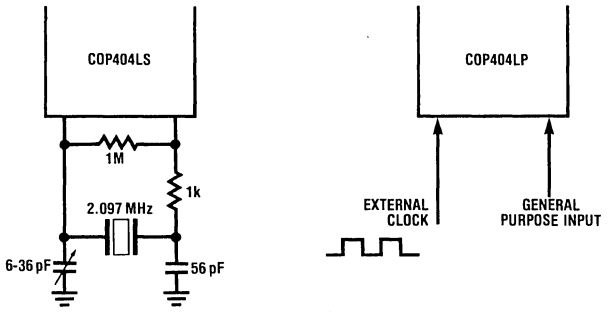
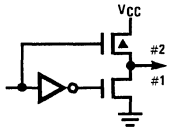
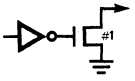



Figure 4. Oscillator

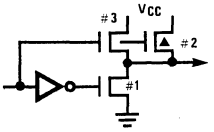




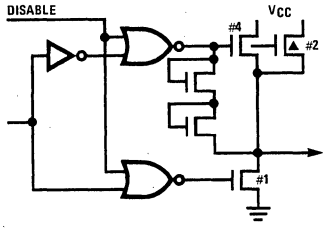

a. Standard Output



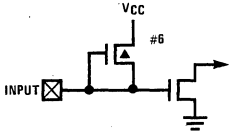

b. Open-Drain Output




c. Push-Pull Output




d. L Output (LED)

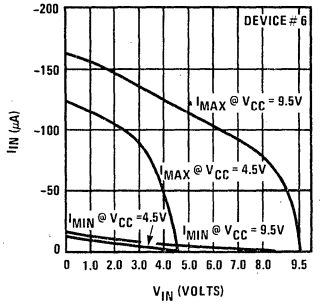



e. Input with Load

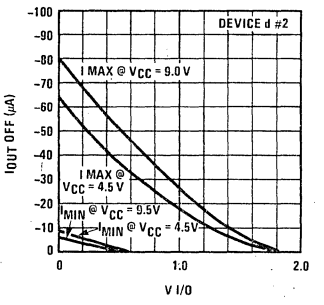
(▲IS DEPLETION DEVICE)

Figure 5. Output Configurations

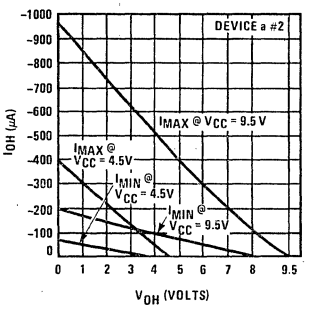
Current for Inputs with Load Device



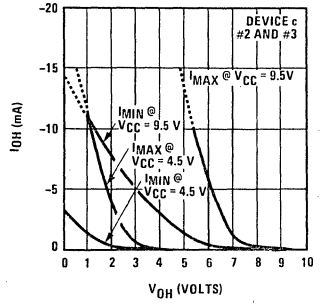

Input Current for $L_0$ through $L_7$ when Output Programmed Off by Software



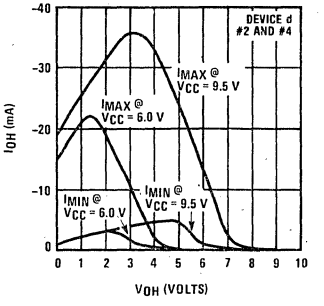

Source current for Standard Output Configuration



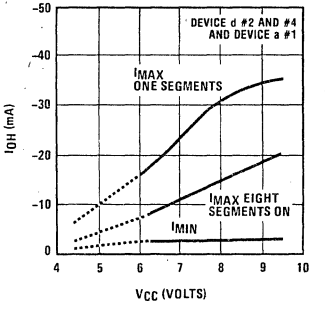

Source Current for SO and SK in Push-Pull Configuration



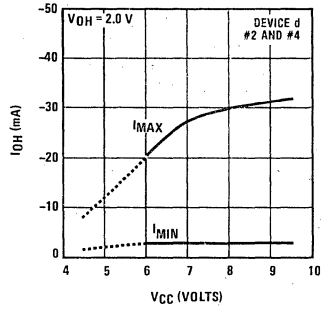

L Output Source Current



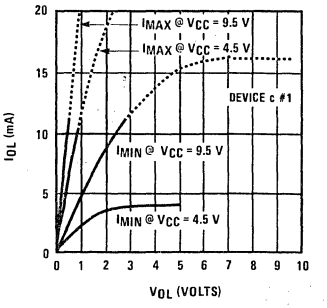

LED Output Direct Segment and Digit Drive



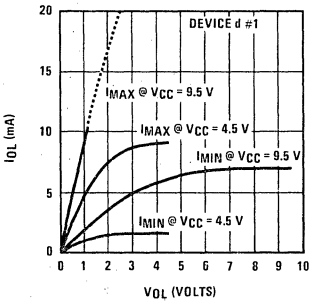

LED Output Direct Segment Drive



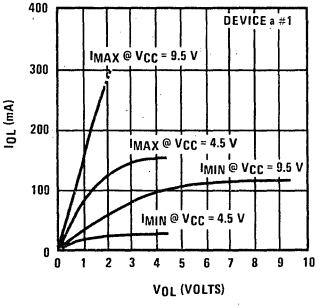

Output Sink Current for SO and SK




Output Sink Current for $L_0$ through $L_7$




Output Sink Current $G_0$-$G_3$ and $D_0$-$D_3$




Output Sink Current IP0-IP7, P8, P9, SKIP/P10

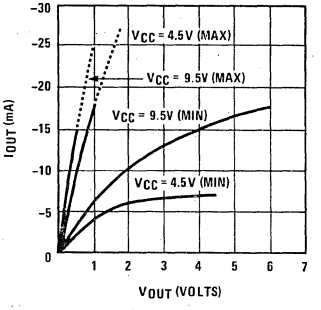



Figure 6. I/O Characteristics

## COP404L INSTRUCTION SET

Table 1 is a symbol table providing internal architecture, instruction operand and operation symbols used in the instruction set table.

Table 2 provides the mnemonic, operand, machine code, data flow, skip conditions and description associated with each instruction in the COP404L instruction set.

**Table 1. COP404L Instruction Set Table Symbols**

| Symbol | Definition |
|---|---|
| INTERNAL ARCHITECTURE SYMBOLS | |
| A | 4-bit Accumulator |
| B | 7-bit RAM Address Register |
| Br | Upper 3 bits of B (register address) |
| Bd | Lower 4 bits of B (digit address) |
| C | 1-bit Carry Register |
| D | 4-bit Data Output Port |
| EN | 4-bit Enable Register |
| G | 4-bit Register to latch data for G I/O Port |
| IL | Two 1-bit Latches associated with the $IN_3$ or $IN_0$ Inputs |
| IN | 4-bit Input Port |
| IP | 8-bit Bidirectional ROM address and data port |
| L | 8-bit TRI-STATE I/O Port |
| M | 4-bit contents of RAM Memory pointed to by B Register |
| P | 3-bit ROM address port |
| PC | 11-bit ROM Address Register (program counter) |
| Q | 8-bit Register to latch data for L I/O Port |
| SA | 11-bit Subroutine Save Register A |
| SB | 11-bit Subroutine Save Register B |
| SC | 11-bit Subroutine Save Register C |
| SIO | 4-bit Shift Register and Counter |
| SK | Logic-Controlled Clock Output |

| Symbol | Definition |
|---|---|
| INSTRUCTION OPERAND SYMBOLS | |
| d | 4-bit Operand Field, 0-15 binary (RAM Digit Select) |
| r | 3-bit Operand Field, 0-7 binary (RAM Register Select) |
| a | 11-bit Operand Field, 0-2047 binary (ROM Address) |
| y | 4-bit Operand Field, 0-15 binary (Immediate Data) |
| RAM(s) | Contents of RAM location addressed by s |
| ROM(t) | Contents of ROM location addressed by t |
| OPERATIONAL SYMBOLS | |
| + | Plus |
| − | Minus |
| → | Replaces |
| ↔ | Is exchanged with |
| = | Is equal to |
| $\overline{A}$ | The ones complement of A |
| ⊕ | Exclusive-OR |
| : | Range of values |

Table 2. COP404L Instruction Set

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| ARITHMETIC INSTRUCTIONS | | | | | | |
| ASC | | 30 | \|0011\|0000\| | A + C + RAM(B) → A<br>Carry → C | Carry | Add with Carry, Skip on Carry |
| ADD | | 31 | \|0011\|0001\| | A + RAM(B) → A | None | Add RAM to A |
| ADT | | 4A | \|0100\|1010\| | A + $10_{10}$ → A | None | Add Ten to A |
| AISC | y | 5- | \|0101\| y \| | A + y → A | Carry | Add Immediate, Skip on Carry (y ≠ 0) |
| CASC | | 10 | \|0001\|0000\| | $\overline{A}$ + RAM(B) + C → A<br>Carry → C | Carry | Complement and Add with Carry, Skip on Carry |
| CLRA | | 00 | \|0000\|0000\| | 0 → A | None | Clear A |
| COMP | | 40 | \|0100\|0000\| | $\overline{A}$ → A | None | Ones complement of A to A |
| NOP | | 44 | \|0100\|0100\| | None | None | No Operation |
| RC | | 32 | \|0011\|0010\| | "0" → C | None | Reset C |
| SC | | 22 | \|0010\|0010\| | "1" → C | None | Set C |
| XOR | | 02 | \|0000\|0010\| | A ⊕ RAM(B) → A | None | Exclusive-OR RAM with A |
| TRANSFER OF CONTROL INSTRUCTIONS | | | | | | |
| JID | | FF | \|1111\|1111\| | ROM ($PC_{10:8}$, A,M) → $PC_{7:0}$ | None | Jump Indirect (Note 2) |
| JMP | a | 6-<br>-- | \|0110\|0\|$a_{10:8}$\|<br>\|$a_{7:0}$\| | a → PC | None | Jump |
| JP | a | -- | \|1\| $a_{6:0}$ \|<br>(pages 2,3 only)<br>or | a → $PC_{6:0}$ | None | Jump within Page (Note 3) |
| | | -- | \|11\| $a_{5:0}$ \|<br>(all other pages) | a → $PC_{5:0}$ | | |
| JSRP | a | -- | \|10\| $a_{5:0}$ \| | PC + 1 → SA → SB → SC<br>00010 → $PC_{10:6}$<br>a → $PC_{5:0}$ | None | Jump to Subroutine Page (Note 4) |
| JSR | a | 6-<br>-- | \|0110\|1\|$a_{10:8}$\|<br>\|$a_{7:0}$\| | PC + 1 → SA → SB → SC<br>a → PC | None | Jump to Subroutine |
| RET | | 48 | \|0100\|1000\| | SC → SB → SA → PC | None | Return from Subroutine |
| RETSK | | 49 | \|0100\|1001\| | SC → SB → SA → PC | Always Skip on Return | Return from Subroutine then Skip |

Table 2. COP404L Instruction Set (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **MEMORY REFERENCE INSTRUCTIONS** | | | | | | |
| CAMQ | | 33 | 0011\|0011 | $A \rightarrow Q_{7:4}$<br>$RAM(B) \rightarrow Q_{3:0}$ | None | Copy A, RAM to Q |
| | | 3C | 0011\|1100 | | | |
| CQMA | | 33 | 0011\|0011 | $Q_{7:4} \rightarrow RAM(B)$<br>$Q_{3:0} \rightarrow A$ | None | Copy Q to RAM, A |
| | | 2C | 0010\|1100 | | | |
| LD | r | -5 | 00\| r \|0101<br>(r = 0:3) | $RAM(B) \rightarrow A$<br>$Br \oplus r \rightarrow Br$ | None | Load RAM into A,<br>Exclusive-OR Br with r |
| LDD | r,d | 23 | 0010\|0011 | $RAM(r,d) \rightarrow A$ | None | Load A with RAM pointed to directly by r,d |
| | | -- | 0\| r \| d | | | |
| LQID | | BF | 1011\|1111 | $ROM(PC_{10:8},A,M) \rightarrow Q$<br>$SB \rightarrow SC$ | None | Load Q Indirect (Note 2) |
| RMB | 0 | 4C | 0100\|1100 | $0 \rightarrow RAM(B)_0$ | None | Reset RAM Bit |
| | 1 | 45 | 0100\|0101 | $0 \rightarrow RAM(B)_1$ | | |
| | 2 | 42 | 0100\|0010 | $0 \rightarrow RAM(B)_2$ | | |
| | 3 | 43 | 0100\|0011 | $0 \rightarrow RAM(B)_3$ | | |
| SMB | 0 | 4D | 0100\|1101 | $1 \rightarrow RAM(B)_0$ | None | Set RAM Bit |
| | 1 | 47 | 0100\|1101 | $1 \rightarrow RAM(B)_1$ | | |
| | 2 | 46 | 0100\|0110 | $1 \rightarrow RAM(B)_2$ | | |
| | 3 | 4B | 0100\|1011 | $1 \rightarrow RAM(B)_3$ | | |
| STII | y | 7- | 0111\| y | $y \rightarrow RAM(B)$<br>$Bd + 1 \rightarrow Bd$ | None | Store Memory Immediate and Increment Bd |
| X | r | -6 | 00\| r \|0110<br>(r = 0:3) | $RAM(B) \leftrightarrow A$<br>$Br \oplus r \rightarrow Br$ | None | Exchange RAM with A,<br>Exclusive-OR Br with r |
| XAD | r,d | 23 | 0010\|0011 | $RAM(r,d) \leftrightarrow A$ | None | Exchange A with RAM pointed to directly by r,d |
| | | -- | 1\| r \| d | | | |
| XDS | r | -7 | 00\| r \|0111<br>(r = 0:3) | $RAM(B) \leftrightarrow A$<br>$Bd - 1 \rightarrow Bd$<br>$Br \oplus r \rightarrow Br$ | Bd decrements past 0 | Exchange RAM with A and Decrement Bd,<br>Exclusive-OR Br with r |
| XIS | r | -4 | 00\| r \|0100<br>(r = 0:3) | $RAM(B) \leftrightarrow A$<br>$Bd + 1 \rightarrow Bd$<br>$Br \oplus r \rightarrow Br$ | Bd increments past 15 | Exchange RAM with A and Increment Bd,<br>Exclusive-OR Br with r |
| **REGISTER REFERENCE INSTRUCTIONS** | | | | | | |
| CAB | | 50 | 0101\|0000 | $A \rightarrow Bd$ | None | Copy A to Bd |
| CBA | | 4E | 0100\|1110 | $Bd \rightarrow A$ | None | Copy Bd to A |
| LBI | r,d | -- | 00\| r \|(d - 1)<br>(r = 0:3;<br>d = 0, 9:15)<br>or | r,d → B | Skip until not a LBI | Load B Immediate with r,d (Note 5) |
| | | 33 | 0011\|0011 | | | |
| | | -- | 1\| r \| d<br>(any r, any d) | | | |
| LEI | y | 33 | 0011\|0001 | $y \rightarrow EN$ | None | Load EN Immediate (Note 6) |
| | | 6- | 0110\| y | | | |
| XABR | | 12 | 0001\|0010 | $A \leftrightarrow Br$ ($0 \rightarrow A_3$) | None | Exchange A with Br |

**Table 2. COP404L Instruction Set (continued)**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **TEST INSTRUCTIONS** | | | | | | |
| SKC | | 20 | 0010\|0000 | | C = "1" | Skip if C is True |
| SKE | | 21 | 0010\|0001 | | A = RAM(B) | Skip if A Equals RAM |
| SKGZ | | 33 | 0011\|0011 | | $G_{3:0}$ | Skip if G is Zero (all 4 bits) |
| | | 21 | 0010\|0001 | | | |
| SKGBZ | | 33 | 0011\|0011 | 1st byte | | Skip if G Bit is Zero |
| | 0 | 01 | 0000\|0001 | 2nd byte | $G_0 = 0$ | |
| | 1 | 11 | 0001\|0001 | | $G_1 = 0$ | |
| | 2 | 03 | 0000\|0011 | | $G_2 = 0$ | |
| | 3 | 13 | 0001\|0011 | | $G_3 = 0$ | |
| SKMBZ | 0 | 01 | 0000\|0001 | | $RAM(B)_0 = 0$ | Skip if RAM Bit is Zero |
| | 1 | 11 | 0001\|0001 | | $RAM(B)_1 = 0$ | |
| | 2 | 03 | 0000\|0011 | | $RAM(B)_2 = 0$ | |
| | 3 | 13 | 0001\|0011 | | $RAM(B)_3 = 0$ | |
| SKT | | 41 | 0100\|0001 | | A time-base counter carry has occurred since last test | Skip on Timer (Note 2) |
| **INPUT/OUTPUT INSTRUCTIONS** | | | | | | |
| ING | | 33 | 0011\|0011 | G → A | None | Input G Ports to A |
| | | 2A | 0010\|1010 | | | |
| ININ | | 33 | 0011\|0011 | IN → A | None | Input IN Inputs to A |
| | | 28 | 0010\|1000 | | | |
| INIL | | 33 | 0011\|0011 | $IL_3$, CKO, "0", $IL_0$ → A | None | Input IL Latches to A (Note 2) |
| | | 29 | 0010\|1001 | | | |
| INL | | 33 | 0011\|0011 | $L_{7:4}$ → RAM(B) | None | Input L Ports to RAM,A |
| | | 2E | 0010\|1110 | $L_{3:0}$ → A | | |
| OBD | | 33 | 0011\|0011 | Bd → D | None | Output Bd to D Outputs |
| | | 3E | 0011\|1110 | | | |
| OGI | y | 33 | 0011\|0011 | y → G | None | Output to G Ports Immediate |
| | | 5- | 0101\| y | | | |
| OMG | | 33 | 0011\|0011 | RAM(B) → G | None | Output RAM to G Ports |
| | | 3A | 0011\|1010 | | | |
| XAS | | 4F | 0100\|1111 | A ↔ SIO, C → SKL | None | Exchange A with SIO (Note 2) |

**Note 1:** All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, $A_3$ indicates the most significant (left-most) bit of the 4-bit A register.

**Note 2:** For additional information on the operation of the XAS, JID, LQID, INIL, and SKT instructions, see below.

**Note 3:** The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

**Note 4:** A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

**Note 5:** LBI is a single-byte instruction if d = 0, 9, 10, 11, 12, 13, 14, or 15. The machine code for the lower 4 bits equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 ($1001_2$), the lower 4 bits of the LBI instruction equal 8 ($1000_2$). To load 0, the lower 4 bits of the LBI instruction should equal 15 ($1111_2$).

**Note 6:** Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register.)

The following information is provided to assist the user in understanding the operation of several unique instructions and to provide notes useful to programmers in writing COP404L programs.

### XAS Instruction

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register. The contents of SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output. (See Functional Description, EN Register, above.) If SIO is selected as a shift register, an XAS instruction must be performed once every 4 instruction cycles to effect a continuous data stream.

### JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM addressed by the 11-bit word, $PC_{10:8}$, A, M. $PC_{10}$, $PC_9$ and $PC_8$ are not affected by this instruction.

Note that JID requires 2 instruction cycles.

### INIL Instruction

INIL (Input IL Latches to A) inputs 2 latches, $IL_3$ and $IL_0$ (see figure 7) and CKO into A. The $IL_3$ and $IL_0$ latches are set if a low-going pulse ("1" to "0") has occurred on the $IN_3$ and $IN_0$ inputs since the last INIL instruction, provided the input pulse stays low for at least two instruction times. Execution of an INIL inputs $IL_3$ and $IL_0$ into A3 and A0 respectively, and resets these latches to allow them to respond to subsequent low-going pulses on the $IN_3$ and $IN_0$ lines. INIL will input the state of CKO into A2 on the COP404LP ("1" into A2 for the COP404LS). A "0" is always placed in A1 upon the execution of an INIL. The general purpose inputs $IN_3$–$IN_0$ are input to A upon execution of an ININ instruction. (See table 2, ININ instruction.) INIL is useful in recognizing pulses of short duration or pulses which occur too often to be read conveniently by an ININ instruction.

Note: IL latches are not cleared on reset.

### LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 11-bit word $PC_{10}$, $PC_9$, $PC_8$, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB →SC) and replaces the least significant 8 bits of PC as follows: A → $PC_{7:4}$, RAM(B) → $PC_{3:0}$, leaving $PC_{10}$, $PC_9$ and $PC_8$ unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped" (SC → SB → SA →PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SB → SC, the previous contents of SC are lost. Also, when LQID pops the stack, the previously pushed contents of SB are left in SC. The net result is that the contents of SB are placed in SC (SB → SC). Note that LQID takes two instruction cycle times to execute.

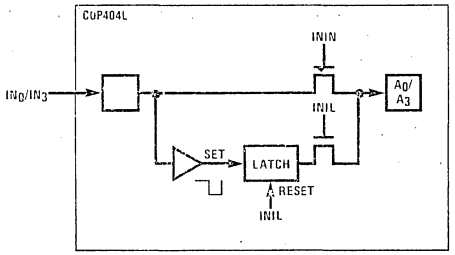



Figure 7. INIL Hardware Implementation

### SKT Instruction

The SKT (Skip On Timer) instruction tests the state of an internal 10-bit time-base counter. This counter divides the instruction cycle clock frequency by 1024 and provides a latched indication of counter overflow. The SKT instruction tests this latch, executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction, therefore, allow the COP404L to generate its own time-base for real-time processing rather than relying on an external input signal.

For example, using a 2.097 MHz oscillator as the time-base to the clock generator, the instruction cycle clock frequency will be 65kHz (crystal frequency ÷ 32) and the binary counter output pulse frequency will be 64Hz. For time-of-day or similar real-time processing, the SKT instruction can call a routine which increments a "seconds" counter every 64 ticks.

### Instruction Set Notes

a. The first word of a COP404L program (ROM address 0) must be a CLRA (Clear A) instruction.

b. Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths take the same number of cycle times whether instructions are skipped or executed.

c. The ROM is organized into 32 pages of 64 words each. The Program Counter is an 11-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example: a JP located in the last word of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3, 7, 11, 15, 19, 23 or 27 will access data in the next group of four pages.

## TYPICAL APPLICATIONS

### PROM-Based System

The COP404L may be used to exactly emulate the COP444L. Figure 8 shows the interconnect to implement a COP444L hardware emulation. This connection uses a MM2716 EPROM as external memory. Other memory can be used such as bipolar PROM or RAM.

Pins IP7-IP0 are bidirectional inputs and outputs. When the AD/$\overline{\text{DATA}}$ clocking output turns on, the EPROM drivers are disabled and IP7-IP0 output addresses. The 8-bit latch (MM74C373) latches the addresses to drive the memory.

When AD/$\overline{\text{DATA}}$ turns off, the EPROM is enabled and the IP7-IP0 pins will input the memory data. P8, P9 and SKIP/P10 output the most significant address bits to the memory. (SKIP output may be used for program debug if needed.)

The other 28 pins of the COP404L may be configured exactly the same as a COP444L. The COP404L $V_{CC}$ can vary from 4.5V to 9.5V. However, 5 volts is used for the memory.

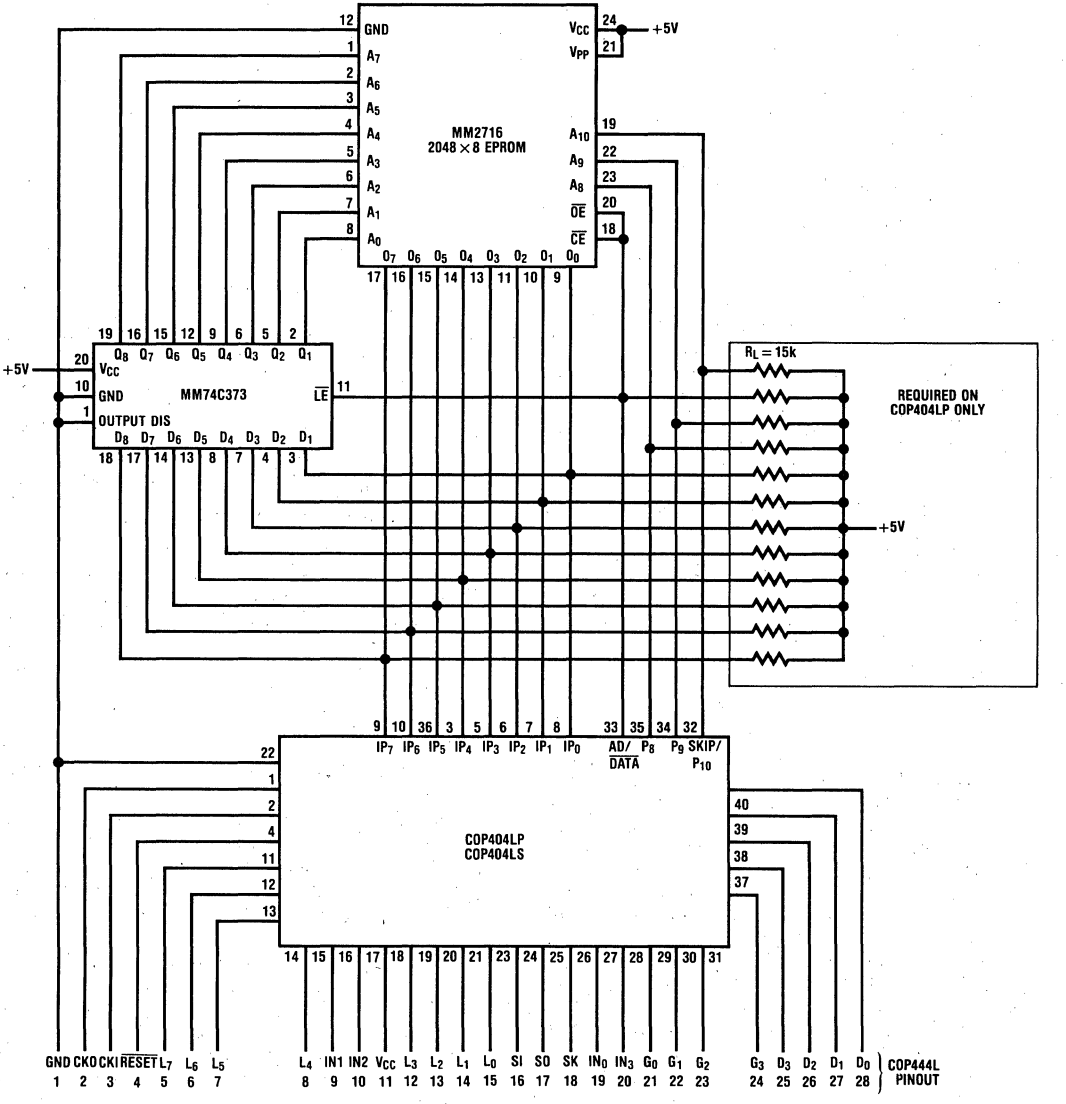



Figure 8. COP404L System Diagram

**COP404L Mask Options**

The following COP444L options have been implemented on the basic versions of the COP404L:

| Option Value | Comment | Option Value | Comment |
|---|---|---|---|
| Option 1 = 0 | Ground, no option available | Option 18 = 2 | SK has push-pull output |
| Option 2 = 0 (404LS) | CKO is clock generator output to crystal/resonator | Option 19 = 0 | IN0 has load device to $V_{CC}$ |
| | | Option 20 = 0 | IN3 has load device to $V_{CC}$ |
| = 2 (404LP) | CKO is general purpose input with load device to $V_{CC}$ | Option 21 = 0 | $G_0$ ⎫ |
| | | Option 22 = 0 | $G_1$ ⎬ have very high current |
| Option 3 = 0 | CKI is oscillator input (divide by 32) | Option 23 = 0 | $G_2$ ⎬ standard output |
| | | Option 24 = 0 | $G_3$ ⎭ |
| Option 4 = 0 | $\overline{\text{RESET}}$ pin has load device to $V_{CC}$ | Option 25 = 0 | $D_3$ ⎫ |
| | | Option 26 = 0 | $D_2$ ⎬ have very high current |
| Option 5 = 2 | $L_7$ ⎫ | Option 27 = 0 | $D_1$ ⎬ standard output |
| Option 6 = 2 | $L_6$ ⎬ have LED direct-drive | Option 28 = 0 | $D_2$ ⎭ |
| Option 7 = 2 | $L_5$ ⎬ output | Option 29 = 0 | L ⎫ |
| Option 8 = 2 | $L_4$ ⎭ | Option 30 = 0 | IN ⎬ have standard input |
| Option 9 = 0 | IN1 has load device to $V_{CC}$ | Option 31 = 0 | G ⎬ levels |
| Option 10 = 0 | IN2 has load device to $V_{CC}$ | Option 32 = 0 | SI ⎭ |
| Option 11 = 1 | $V_{CC}$ 4.5 to 9.5V operation | Option 33 = 0 | $\overline{\text{RESET}}$ has Schmitt trigger input |
| Option 12 = 2 | $L_3$ ⎫ | Option 34 = 0 | CKO has standard input levels |
| Option 13 = 2 | $L_2$ ⎬ have LED direct-drive | Option 35 = N/A | 40-pin package |
| Option 14 = 2 | $L_1$ ⎬ output | Option 36 = 0 (404LS) | $\overline{\text{RESET}}$ pin used normally |
| Option 15 = 2 | $L_0$ ⎭ | = 1 (404LP) | Power-Low failsafe enabled |
| Option 16 = 0 | SI has load to $V_{CC}$ | | |
| Option 17 = 2 | SO has push-pull output | | |

National Semiconductor

# COP410L/COP411L Single-Chip N-Channel Microcontrollers

## General Description

The COP410L and COP411L Single-Chip N-Channel Microcontrollers are members of the COPS™ family, fabricated using N-channel, silicon gate MOS technology. These Controller Oriented Processors are complete microcomputers containing all system timing, internal logic, ROM, RAM and I/O necessary to implement dedicated control functions in a variety of applications. Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture and I/O scheme designed to facilitate keyboard input, display output and BCD data manipulation. The COP411L is identical to the COP410L, but with 16 I/O lines instead of 19. They are an appropriate choice for use in numerous human interface control environments. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized Controller Oriented Processor at a low end-product cost.

## Features

- Low cost
- Powerful instruction set
- 512x8 ROM, 32x4 RAM
- 19 I/O lines (COP410L)
- Two-level subroutine stack
- 16$\mu$s instruction time
- Single supply operation (4.5 – 6.3V)
- Low current drain (5mA max @ 5V)
- Internal binary counter register with serial I/O capability
- General purpose and TRI-STATE® outputs
- LSTTL/CMOS compatible in and out
- Direct drive of LED digit and segment lines
- Software/hardware compatible with other members of COP400 family
- Extended temperature range device to be available (– 40°C to + 85°C)
- Wider supply range (4.5 – 9.5V) optionally available

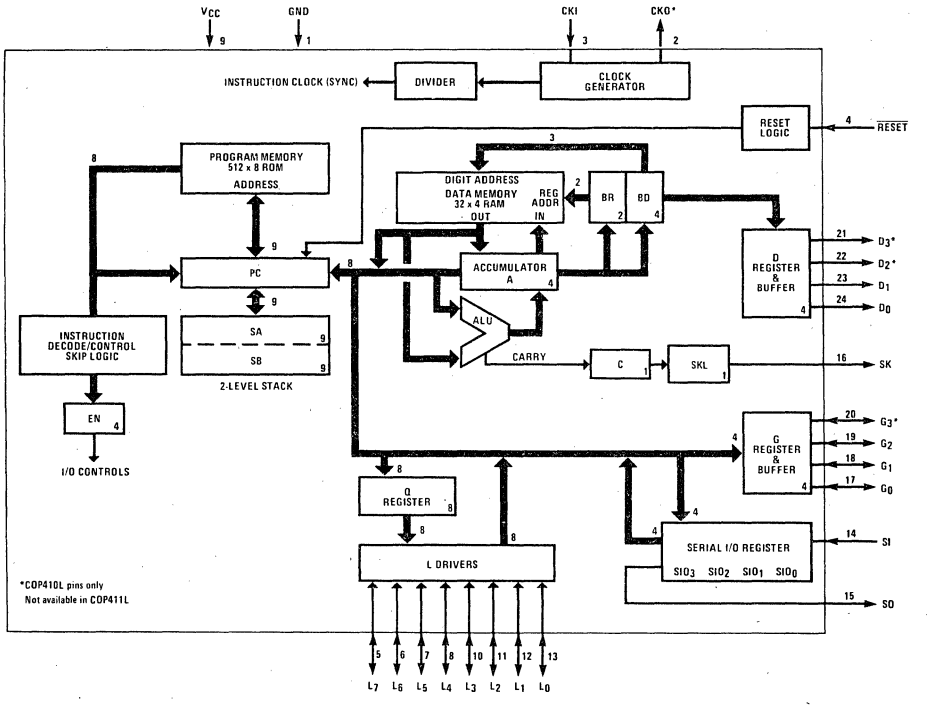



Figure 1. COP410L/411L Block Diagram

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Relative to GND | −0.3V to +10V |
| Ambient Operating Temperature | 0°C to +70°C |
| Ambient Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |
| Power Dissipation | 0.75 Watt at 25°C<br>0.4 Watt at 70°C |

*Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.*

## DC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Operating Voltage ($V_{CC}$) | | 4.5 | 9.5 | V |
| Operating Supply Current | $V_{CC}=5V$, $T_A=25°C$<br>(all inputs and outputs open) | | 5 | mA |
| Input Voltage Levels | | | | |
| CKI Input Levels | | | | |
| Ceramic Resonator Input | | | | |
| Logic High ($V_{IH}$) | | 2.0 | | V |
| Logic Low ($V_{IL}$) | | | 0.4 | V |
| Schmitt Trigger Input | | | | |
| Logic High ($V_{CC}$) | | 0.7 $V_{CC}$ | | V |
| Logic Low ($V_{IL}$) | | | 0.6 | V |
| $\overline{\text{RESET}}$ Input Levels | | | | |
| Logic High | | 0.7 $V_{CC}$ | | V |
| Logic Low | | | 0.6 | V |
| $\overline{\text{RESET}}$ Hysteresis | | 1.0 | | V |
| SO Input Level (Test mode) | | 2.0 | 3.0 | V |
| All Other Inputs | | | | |
| Logic High | $V_{CC} \leq 9.5V$ | 3.0 | | V |
| Logic High | with TTL trip level options | 2.0 | | V |
| Logic Low | selected, $V_{CC} = 5V \pm 10\%$ | | 0.8 | V |
| Logic High | with high trip level options | 3.6 | | V |
| Logic Low | selected | | 1.2 | V |
| Output Voltage Levels | | | | |
| LSTTL Operation | $V_{CC}=5V \pm 5\%$ | | | |
| Logic High ($V_{OH}$) | $I_{OH}=-25\mu A$ | 2.7 | | V |
| Logic Low ($V_{OL}$) | $I_{OL}=0.36mA$ | | 0.4 | V |

## DC Electrical Characteristics (continued)

$0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise specified.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Output Current Levels | | | | |
| Output Sink Current | | | | |
| SO and SK Outputs ($I_{OL}$) | $V_{CC}=9.5V$, $V_{OL}=1.0V$ | 4.5 | 22 | mA |
| | $V_{CC}=4.5V$, $V_{OL}=1.0V$ | 2.2 | 11 | mA |
| $L_0$-$L_7$ Outputs, $G_0$-$G_3$ and LSTTL $D_0$-$D_3$ Outputs ($I_{OL}$) | $V_{CC}=9.5V$, $V_{OL}=1.0V$ | 2.0 | 9.0 | mA |
| | $V_{CC}=4.5V$, $V_{OL}=1.0V$ | 1.0 | 4.5 | mA |
| $D_0$-$D_3$ Outputs with High Sink Current Options ($I_{OL}$) | $V_{CC}=9.5V$, $V_{OL}=1.0V$ | 15 | 75 | mA |
| | $V_{CC}=4.5V$, $V_{OL}=1.0V$ | 7.5 | 35 | mA |
| $D_0$-$D_3$ Outputs with Very High Sink Current Options ($I_{OL}$) | $V_{CC}=9.5V$, $V_{OL}=1.0V$ | 30 | 150 | mA |
| | $V_{CC}=4.5V$, $V_{OL}=1.0V$ | 15 | 70 | mA |
| Output Source Current: | | | | |
| Standard Configuration, All Outputs ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −70 | −450 | μA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −26 | −190 | μA |
| Push-Pull Configuration, SO and SK Outputs ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −1.4 | −15 | mA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −0.07 | −2.8 | mA |
| LED Configuration, $L_0$-$L_7$ Outputs, Low Current Driver Option ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=2.0V$ | −1.5 | −15 | mA |
| | $V_{CC}=6.0V$, $V_{OH}=2.0V$ | −1.5 | −10 | mA |
| LED Configuration, $L_0$-$L_7$ Outputs, High Current Driver Option ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=2.0V$ | −3.0 | −30 | mA |
| | $V_{CC}=6.0V$, $V_{OH}=2.0V$ | −3.0 | −20 | mA |
| TRI-STATE® Configuration, $L_0$-$L_7$ Outputs, Low Current Driver Option ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −2.4 | −24 | mA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −0.06 | 4.0 | mA |
| TRI-STATE® Configuration, $L_0$-$L_7$ Outputs, High Current Driver Option ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −4.9 | −48 | mA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −0.12 | −8.0 | mA |
| CKO Output | | | | |
| RAM Power Supply Option Power Requirement | $V_R=3.3V$ | | 1.5 | mA |
| TRI-STATE® Output Leakage Current | | −10 | +10 | μA |

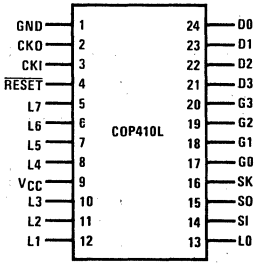
## AC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise specified.

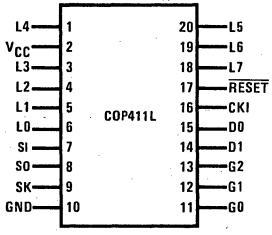
| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Instruction Cycle Time — $t_C$ | | 15 | 40 | μs |
| CKI Using Ceramic Resonator | | | | |
| Input Frequency — $f_I$ | ÷8 mode | 0.2 | 0.53 | MHz |
| Duty Cycle | | 30 | 55 | % |
| CKI Using External Clock | | | | |
| Input Frequency — $f_I$ | ÷8 mode | 0.2 | 0.53 | MHz |
| Duty Cycle | | 30 | 60 | % |
| Rise Time | $f_I = 0.5$ MHz | | 0.5 | μs |
| Fall Time | | | 0.2 | μs |
| CKI Using RC | $R = 56\,k\Omega \pm 5\%$ | | | |
| Frequency | $C = 100\,pF \pm 10\%$ | 140 | 270 | kHz |
| Instruction Cycle Time | | 15 | 28 | μs |
| CKO as SYNC Input | | | | |
| $t_{SYNC}$ | | 400 | | ns |
| INPUTS: | | | | |
| $IN_3$-$IN_0$, $G_3$-$G_0$, $L_7$-$L_0$ | | | | |
| $t_{SETUP}$ | | | 8 | μs |
| $t_{HOLD}$ | | | 1 | μs |
| SI | | | | |
| $t_{SETUP}$ | | | 2 | μs |
| $t_{HOLD}$ | | | 1 | μs |
| OUTPUTS: | | | | |
| COP TO CMOS PROPAGATION DELAY | $V_{OH} = 0.7\,V_{CC}$, $V_{OL} = 0.3\,V_{CC}$, $C_L = 50\,pF$ | | | |
| All Standard Output Configurations | | | | |
| $t_{PD1}$ | | | 6.5 | μs |
| SO, SK Outputs | | | | |
| $t_{PD1}$ (push-pull) | | | 4.0 | μs |
| $t_{PD0}$ | | | 1.2 | μs |
| $D_3$-$D_0$, $G_3$-$G_0$ | | | | |
| $t_{PD0}$ | | | 2.7 | μs |
| $L_7$-$L_0$ | | | | |
| $t_{PD0}$ | | | 2.7 | μs |
| $t_{PD1}$ (push pull) | | | 3.0 | μs |

## AC Electrical Characteristics (continued)

$0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise specified.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| COP TO LSTTL PROPAGATION DELAY | $V_{CC} = 5V \pm 5\%$, $V_{OH} = 2.7V$<br>$V_{OL} = 0.4V$, $C_L = 50pF$ | | | |
| SO, SK Outputs | | | | |
| $t_{PD1}$ (standard) | | | 5 | μs |
| $t_{PD1}$ (push-pull) | | | 3.5 | μs |
| $t_{PD0}$ | | | 3 | μs |
| $L_7-L_0$ Outputs | | | | |
| $t_{PD1}$ (push-pull) | | | 1.5 | μs |
| $L_7-L_0$, $G_3-G_0$, $D_3-D_0$ Outputs | | | | |
| $t_{PD1}$ (standard) | | | 5.0 | μs |
| $t_{PD0}$ | | | 5.0 | μs |
| CKO (figure 3b) | | | | |
| $t_{PD1}$ | | | 0.6 | μs |
| $t_{PD0}$ | | | 0.6 | μs |




| Pin | Description |
|---|---|
| $L_7-L_0$ | 8 bidirectional I/O ports with TRI-STATE® |
| $G_3-G_0$ | 4 bidirectional I/O ports ($G_2-G_0$ for COP411L) |
| $D_3-D_0$ | 4 general purpose outputs ($D_1-D_0$ for COP411L) |
| SI | Serial input (or counter input) |
| SO | Serial output (or general purpose output) |
| SK | Logic-controlled clock (or general purpose output) |

| Pin | Description |
|---|---|
| CKI | System oscillator input |
| CKO | System oscillator output (or RAM power supply or SYNC input) (COP410L only) |
| $\overline{\text{RESET}}$ | System reset input |
| $V_{CC}$ | Power supply |
| GND | Ground |

**Figure 2. Connection Diagrams**

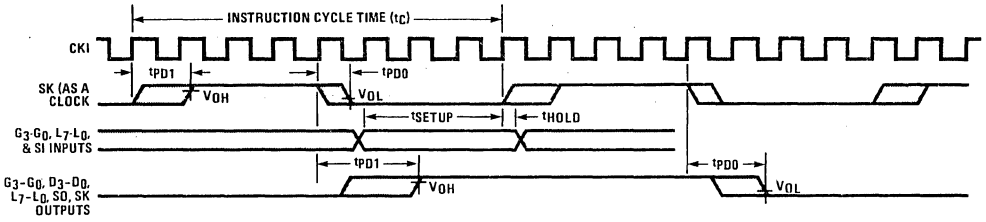



Figure 3. Input/Output Timing Diagrams (Ceramic Resonator Divide-by-8 Mode)

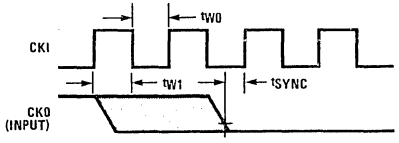



Figure 3a. Synchronization Timing

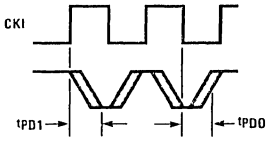



Figure 3b. CKO Output Timing

## FUNCTIONAL DESCRIPTION

A block diagram of the COP410L is given in Figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1" (greater than 2 volts). When a bit is reset, it is a logic "0" (less than 0.8 volts).

### Program Memory

Program Memory consists of a 512-byte ROM. As can be seen by an examination of the COP410L/411L instruction set, these words may be program instructions, program data or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID and LQID instructions, ROM must often be thought of as being organized into 8 pages of 64 words each.

ROM addressing is accomplished by a 9-bit PC register. Its binary value selects one of the 512 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 9-bit binary count value. Two levels of subroutine nesting are implemented by the 9-bit subroutine save registers, SA and SB, providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the Instruction Decode, Control and Skip Logic circuitry.

### Data Memory

Data memory consists of a 128-bit RAM, organized as 4 data registers of 8 4-bit digits. RAM addressing is implemented by a 6-bit B register whose upper 2 bits (Br) select 1 of 4 data registers and lower 3 bits of the 4-bit Bd select 1 of 8 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) is usually loaded into or from, or exchanged with, the A register (accumulator), it may also be loaded into the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the XAD 3,15 instruction. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

The most significant bit of Bd is not used to select a RAM digit. Hence each physical digit of RAM may be selected by two different values of Bd as shown in Figure 4 below. The skip condition for XIS and XDS instructions will be true if Bd changes between 0 and 15, but NOT between 7 and 8 (see Table 3).

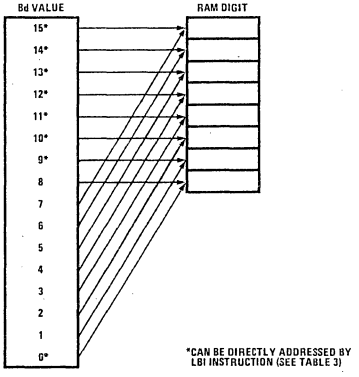



Figure 4. RAM Digit Address to Physical RAM Digit Mapping

### Internal Logic

The 4-bit A register (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Bd portion of the B register, to load 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

A 4-bit adder performs the arithmetic and logic functions of the COP410L/411L, storing its results in A. It also outputs a carry bit to the 1-bit C register, most often employed to indicate arithmetic overflow. The C register, in conjunction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SK or can enable SK to be a sync clock each instruction cycle time. (See XAS instruction and EN register description, below.)

The G register contents are outputs to 4 general-purpose bidirectional I/O ports.

The Q register is an internal, latched, 8-bit register, used to hold data loaded from M and A, as well as 8-bit data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control. (See LEI instruction.)

The 8 L drivers, when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the LED Direct Drive output configuration option) with Q data being outputted to the Sa-Sg and decimal point segments of the display.

The SIO register functions as a 4-bit serial-in/serial-out shift register or as a binary counter depending on the contents of the EN register. (See EN register description, below.) Its contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O by connecting SO to external serial-in/parallel-out shift registers.

The XAS instruction copies C into the SKL Latch. In the counter mode, SK is the output of SKL in the shift register mode, SK outputs SKL ANDed with internal instruction cycle clock.

The EN register is an internal 4-bit register loaded under program control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register ($EN_3$-$EN_0$).

1. The least significant bit of the enable register, $EN_0$, selects the SIO register as either a 4-bit shift register or a 4-bit binary counter. With $EN_0$ set, SIO is an asynchronous binary counter, *decrementing* its value by one upon each low-going pulse ("1" to "0") occurring on the SI input. Each pulse must be at least two instruction cycles wide. SK outputs the value of SKL. The SO output is equal to the value of $EN_3$. With $EN_0$ reset, SIO is a serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time. (See 4 below.) The SK output becomes a logic-controlled clock.
2. $EN_1$ is not used. It has no effect on COP410L/COP411L operation.
3. With $EN_2$ set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting $EN_2$ disables the L drivers, placing the L I/O ports in a high-impedance input state.
4. $EN_3$, in conjunction with $EN_0$, affects the SO output. With $EN_0$ set (binary counter option selected) SO will output the value loaded into $EN_3$. With $EN_0$ reset (serial shift register option selected), setting $EN_3$ enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting $EN_3$ with the serial shift register option selected disables SO as the shift register output; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0." Table I provides a summary of the modes associated with $EN_3$ and $EN_0$.

### Initialization

The Reset Logic will initialize (clear) the device upon power-up if the power supply rise time is less than 1 ms and greater than 1 $\mu$s. If the power supply rise time is greater than 1 ms, the user must provide an external RC

**Table 1. Enable Register Modes — Bits $EN_3$ and $EN_0$**

| $EN_3$ | $EN_0$ | SIO | SI | SO | SK |
|---|---|---|---|---|---|
| 0 | 0 | Shift Register | Input to Shift Register | 0 | If SKL = 1, SK = Clock<br>If SKL = 0, SK = 0 |
| 1 | 0 | Shift Register | Input to Shift Register | Serial Out | If SKL = 1, SK = Clock<br>If SKL = 0, SK = 0 |
| 0 | 1 | Binary Counter | Input to Binary Counter | 0 | If SKL = 1, Sk = 1<br>If SKL = 0, SK = 0 |
| 1 | 1 | Binary Counter | Input to Binary Counter | 1 | If SKL = 1, SK = 1<br>If SKL = 0, SK = 0 |

network and diode to the RESET pin as shown below (Figure 5). The RESET pin is configured as a Schmitt trigger input. If not used it should be connected to $V_{CC}$. Initialization will occur whenever a logic "0" is applied to the RESET input, provided it stays low for at least three instruction cycle times.

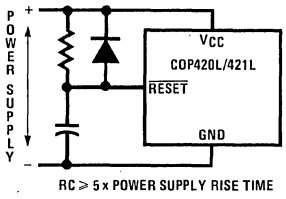



**Figure 5. Power-Up Clear Circuit**

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, and G registers are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. *Data Memory (RAM) must be cleared by the user's program.* The first instruction at address 0 must be a CLRA.

### Oscillator

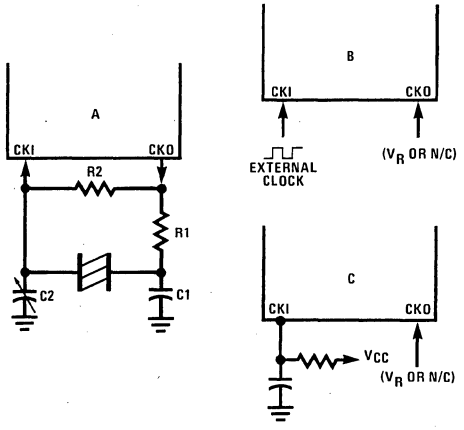
There are four basic clock oscillator configurations available as shown by Figure 6.

a. **Resonator Controlled Oscillator.** CKI and CKO are connected to an external ceramic resonator. The instruction cycle frequency equals the resonator frequency divided by 8. This is not available in the COP411L.

b. **External Oscillator.** CKI is an external clock input signal. The external frequency is divided by 8 to give the instruction frequency time. CKO is now available to be used as the RAM power supply ($V_R$), as a SYNC input, or no connection. (Note: No CKO on COP411L)

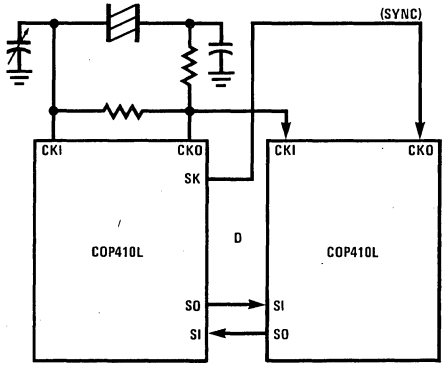
c. **RC Controlled Oscillator.** CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 4. CKO is available as the RAM power supply ($V_R$) or no connection.

d. **Externally Synchronized Oscillator.** Intended for use in multi-COP systems, CKO is programmed to function as an input connected to the SK output of another COP chip operating at the same frequency (COP chip with L or C suffix) with CKI connected as shown. In this configuration, the SK output connected to CKO must provide a SYNC (instruction cycle) signal to CKO, thereby allowing synchronous data transfer between the COPs using only the SI and SO serial I/O pins in conjunction with the XAS instruction. Note that on power-up SK is automatically enabled as a SYNC output. (See Functional Description, Initialization, above.) This is not available in the COP411L.




**Ceramic Resonator Oscillator**

| Resonator Value | Component Values | | | |
|---|---|---|---|---|
| | R1 (Ω) | R2 (Ω) | C1 (pF) | C2 (pF) |
| 455 kHz | 1k | 1M | 80 | 80 |




**RC Controlled Oscillator**

| R (kΩ) | C (pF) | Instruction Cycle Time in μs) |
|---|---|---|
| 51 | 100 | 19 ± 15% |
| 82 | 56 | 19 ± 13% |

**Figure 6. COP410L/411L Oscillator**

### CKO Pin Options

In a resonator controlled oscillator system, CKO is used as an output to the resonator network. As an option CKO can be a SYNC input as described above. As another option, CKO can be a RAM power supply pin ($V_R$), allowing its connection to a standby/backup power supply to maintain the integrity of RAM data with minimum power drain when the main supply is inoperative or shut down to conserve power. Using no connection option is appropriate in applications where the COP410L system timing configuration does not require use of the CKO pin.

**Note:** Considerable *caution* should be exercised in the control of the RESET and $V_{CC}$ pin when using the separate RAM power pin. Data could be altered by improper control of these pins during power-up and power-down. Contact COPS Applications (408/737-5582) for assistance.

### I/O Options

COP410L/411L inputs and outputs have the following optional configurations, illustrated in Figure 7:

a. **Standard** — an enhancement-mode device to ground in conjunction with a depletion-mode device to $V_{CC}$, compatible with LSTTL and CMOS input requirements. Available on SO, SK, and all D and G outputs.

b. **Open-Drain** — an enhancement-mode device to ground only, allowing external pull-up as required by the user's application. Available on SO, SK, and all D and G outputs.

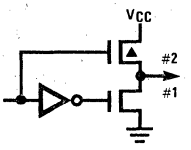
c. **Push-Pull** — an enhancement-mode device to ground in conjunction with a depletion-mode device paralleled by an enhancement-mode device to $V_{CC}$. This configuration has been provided to allow for fast rise and fall times when driving capacitive loads. Available on SO and SK outputs only.

d. **Standard L** — same as **a.**, but may be disabled. Available on L outputs only.

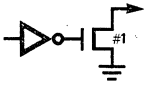
e. **Open Drain L** — same as **b.**, but may be disabled. Available on L outputs only.

f. **LED Direct Drive** — an enhancement mode device to ground and to $V_{CC}$, meeting the typical current sourcing requirements of the segments of an LED display. The sourcing device is clamped to limit current flow. These devices may be turned off under program control (see Functional Description, EN Register), placing the outputs in a high-impedance state to provide required LED segment blanking for a multiplexed display. Available on L outputs only.

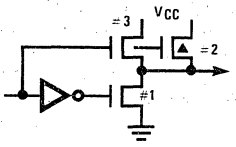
g. **TRI-STATE® Push-Pull** — an enhancement-mode device to ground and $V_{CC}$. These outputs are TRI-STATE® outputs, allowing for connection of these



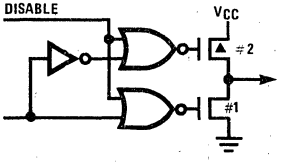

a. Standard Output



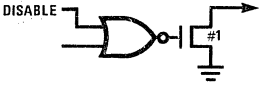

b. Open-Drain Output



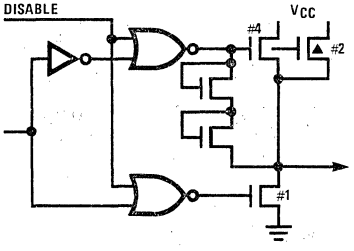

c. Push-Pull Output




d. Standard L Output

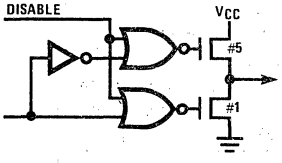



e. Open-Drain L Output

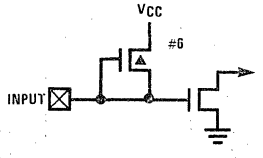



(▲IS DEPLETION DEVICE)

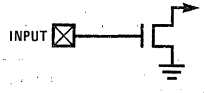
f. LED (L Output)




g. TRI-STATE® Push-Pull (L Output)




h. Input with Load




i. Hi-Z Input

Figure 7. Input and Output Configurations

outputs to a data bus shared by other bus drivers. Available on L outputs only.

**h.** An on-chip depletion load device to $V_{CC}$.

**i.** A Hi-Z input which must be driven to a "1" or "0" by external components.

The above input and output configurations share common enhancement-mode and depletion-mode devices. Specifically, all configurations use one or more of six devices (numbered 1-6, respectively). Minimum and maximum current ($I_{OUT}$ and $V_{OUT}$) curves are given in Figure 8 for each of these devices to allow the designer to effectively use these I/O configurations in designing a COP410L/411L system.

The SO, SK outputs can be configured as shown in **a.**, **b.**, or **c.** The D and G outputs can be configured as shown in **a.** or **b.** Note that when inputting data to the G ports, the G outputs should be set to "1." The L outputs can be configured as in **d.**, **e.**, **f.**, or **g.**

An important point to remember if using configuration **d.** or **f.** with the L drivers is that even when the L drivers are disabled, the depletion load device will source a small amount of current. (See Figure 8, device 2.)

### COP411L

If the COP410L is bonded as a 20-pin device, it becomes the COP411L, illustrated in Figure 2, COP410L/411L Connection Diagrams. Note that the COP411L does not contain D2, D3, G3, or CKO. Use of this option of course precludes use of D2, D3, G3, and CKO options. All other options are available for the COP411L.

## Typical Performance Curves

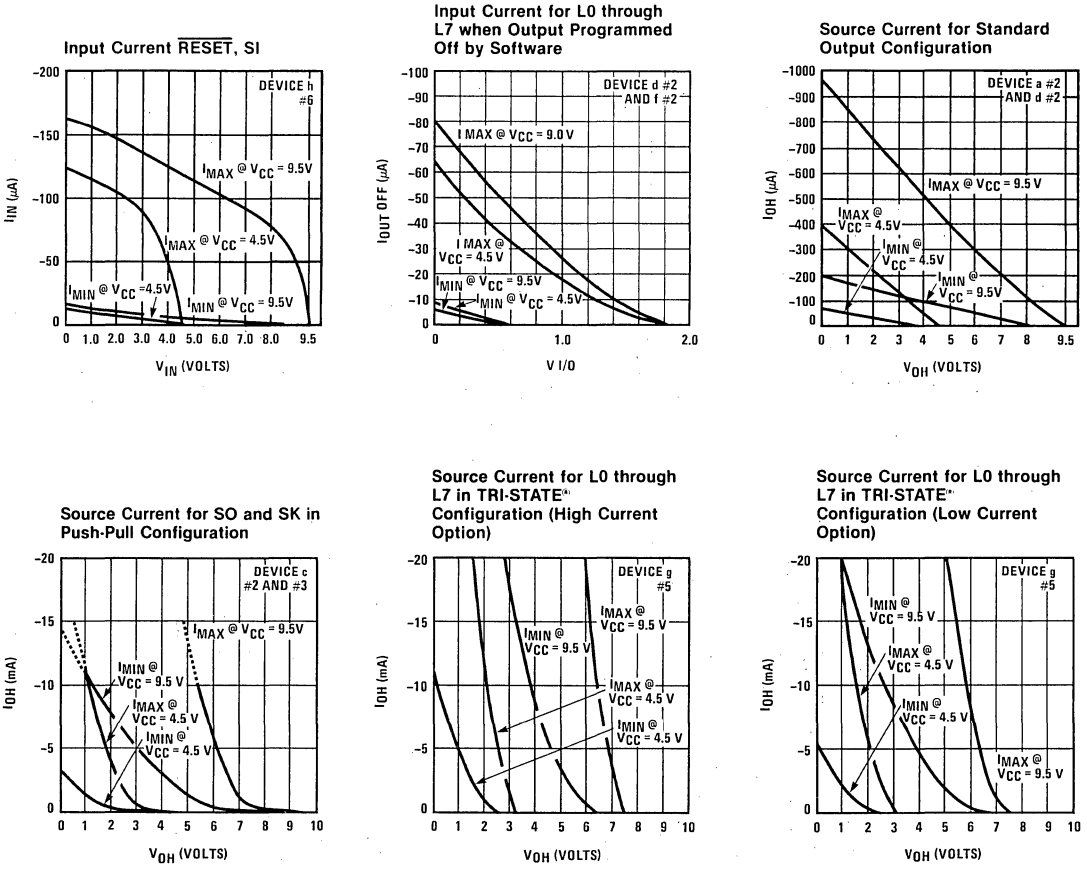
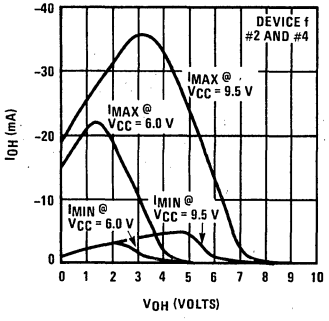
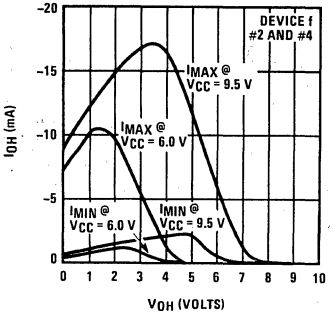



Figure 8. I/O DC Current Characteristics

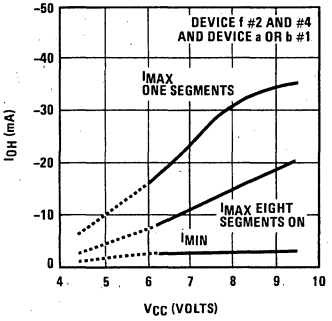
LED Output Source Current
(for High Current LED Option)



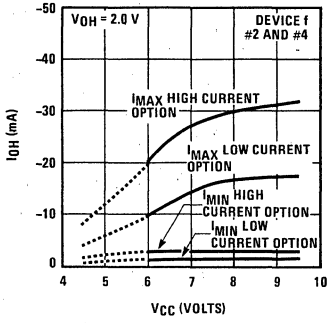

LED Output Source Current
(for Low Current LED Option)



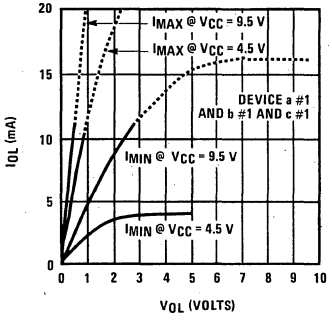

LED Output Direct Segment
and Digit Drive
High Current Options on
L0-L7
Very High Current Options on
D0-D3



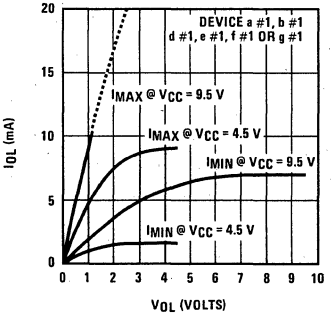

LED Output Direct Segment Drive



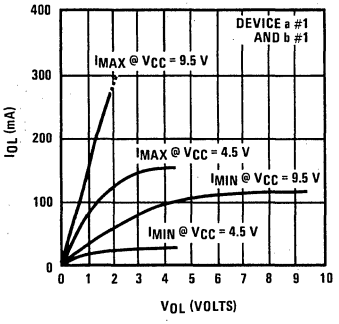

Output Sink Current for SO
and SK




Output Sink Current for L0-L7
and Standard Drive Option for
D0-D3 and G0-G3




Output Sink Current for D0-D3
with Very High Current Option




Output Sink Current for
D0-D3 (for High Current
Option)

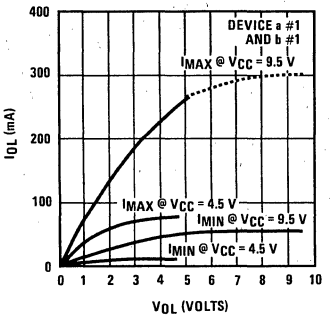



Figure 8. I/O DC Current Characteristics

## COP410L/411L INSTRUCTION SET

Table 2 is a symbol table providing internal architecture, instruction operand and operational symbols used in the instruction set table.

Table 3 provides the mnemonic, operand, machine code, data flow, skip conditions and description associated with each instruction in the COP410L/411L instruction set.

**Table 2. COP410L/411L Instruction Set Table Symbols**

| Symbol | Definition |
|---|---|
| **INTERNAL ARCHITECTURE SYMBOLS** | |
| A | 4-bit Accumulator |
| B | 6-bit RAM Address Register |
| Br | Upper 2 bits of B (register address) |
| Bd | Lower 4 bits of B (digit address) |
| C | 1-bit Carry Register |
| D | 4-bit Data Output Port |
| EN | 4-bit Enable Register |
| G | 4-bit Register to latch data for G I/O Port |
| L | 8-bit TRI-STATE I/O Port |
| M | 4-bit contents of RAM Memory pointed to by B Register |
| PC | 9-bit ROM Address Register (program counter) |
| Q | 8-bit Register to latch data for L I/O Port |
| SA | 9-bit Subroutine Save Register A |
| SB | 9-bit Subroutine Save Register B |
| SIO | 4-bit Shift Register and Counter |
| SK | Logic-Controlled Clock Output |

| Symbol | Definition |
|---|---|
| **INSTRUCTION OPERAND SYMBOLS** | |
| d | 4-bit Operand Field, 0-15 binary (RAM Digit Select) |
| r | 2-bit Operand Field, 0-3 binary (RAM Register Select) |
| a | 9-bit Operand Field, 0-511 binary (ROM Address) |
| y | 4-bit Operand Field, 0-15 binary (Immediate Data) |
| RAM(s) | Contents of RAM location addressed by s |
| ROM(t) | Contents of ROM location addressed by t |
| **OPERATIONAL SYMBOLS** | |
| + | Plus |
| − | Minus |
| → | Replaces |
| ⟷ | Is exchanged with |
| = | Is equal to |
| $\overline{A}$ | The one's complement of A |
| ⊕ | Exclusive-OR |
| : | Range of values |

**Table 3. COP410L/411L Instruction Set**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| ARITHMETIC INSTRUCTIONS | | | | | | |
| ASC | | 30 | 0011\|0000 | A + C + RAM(B) → A<br>Carry → C | Carry | Add with Carry, Skip on Carry |
| ADD | | 31 | 0011\|0001 | A + RAM(B) → A | None | Add RAM to A |
| AISC | y | 5- | 0101\| y | A + y → A | Carry | Add Immediate, Skip on Carry (y ≠ 0) |
| CLRA | | 00 | 0000\|0000 | 0 → A | None | Clear A |
| COMP | | 40 | 0100\|0000 | $\overline{A}$ → A | None | One's complement of A to A |
| NOP | | 44 | 0100\|0100 | None | None | No Operation |
| RC | | 32 | 0011\|0010 | "0" → C | None | Reset C |
| SC | | 22 | 0010\|0010 | "1" → C | None | Set C |
| XOR | | 02 | 0000\|0010 | A ⊕ RAM(B) → A | None | Exclusive-OR RAM with A |

Table 3. COP410L/411L Instruction Set (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **TRANSFER OF CONTROL INSTRUCTIONS** | | | | | | |
| JID | | FF | 1111 1111 | ROM ($PC_8$,A,M) → $PC_{7:0}$ | None | Jump Indirect (Note 2) |
| JMP | a | 6- | 0110 000 $a_8$ | a → PC | None | Jump |
| | | -- | $a_{7:0}$ | | | |
| JP | a | -- | 1 $a_{6:0}$ (pages 2,3 only) | a → $PC_{6:0}$ | None | Jump within Page (Note 3) |
| | | | or | | | |
| | | -- | 11 $a_{5:0}$ (all other pages) | a → $PC_{5:0}$ | | |
| JSRP | a | -- | 10 $a_{5:0}$ | PC + 1 → SA → SB<br>010 → $PC_{8:6}$<br>a → $PC_{5:0}$ | None | Jump to Subroutine Page (Note 4) |
| JSR | a | 6- | 0110 100 $a_8$ | PC + 1 → SA → SB<br>a → PC | None | Jump to Subroutine |
| | | -- | $a_{7:0}$ | | | |
| RET | | 48 | 0100 1000 | SB → SA → PC | None | Return from Subroutine |
| RETSK | | 49 | 0100 1001 | SB → SA → PC | Always Skip on Return | Return from Subroutine then Skip |
| **MEMORY REFERENCE INSTRUCTIONS** | | | | | | |
| CAMQ | | 33 | 0011 0011 | A → $Q_{7:4}$<br>RAM(B) → $Q_{3:0}$ | None | Copy A, RAM to Q |
| | | 3C | 0011 1100 | | | |
| LD | r | -5 | 00 r 0101 | RAM(B) → A<br>Br ⊕ r → Br | None | Load RAM into A, Exclusive-OR Br with r |
| LQID | | BF | 1011 1111 | ROM($PC_8$,A,M) → Q<br>SA → SB | None | Load Q Indirect (Note 2) |
| RMB | 0 | 4C | 0100 1100 | 0 → $RAM(B)_0$ | None | Reset RAM Bit |
| | 1 | 45 | 0100 0101 | 0 → $RAM(B)_1$ | | |
| | 2 | 42 | 0100 0010 | 0 → $RAM(B)_2$ | | |
| | 3 | 43 | 0100 0011 | 0 → $RAM(B)_3$ | | |
| SMB | 0 | 4D | 0100 1101 | 1 → $RAM(B)_0$ | None | Set RAM Bit |
| | 1 | 47 | 0100 0111 | 1 → $RAM(B)_1$ | | |
| | 2 | 46 | 0100 0110 | 1 → $RAM(B)_2$ | | |
| | 3 | 4B | 0100 1011 | 1 → $RAM(B)_3$ | | |
| STII | y | 7- | 0111 y | y → RAM(B)<br>Bd + 1 → Bd | None | Store Memory Immediate and Increment Bd |
| X | r | -6 | 00 r 0110 | RAM(B) ⟷ A<br>Br ⊕ r → Br | None | Exchange RAM with A, Exclusive-OR Br with r |
| XAD | 3,15 | 23 | 0010 0011 | RAM(3,15) ⟷ A | None | Exchange A with RAM (3,15) |
| | | BF | 1011 1111 | | | |
| XDS | r | -7 | 00 r 0111 | RAM(B) ⟷ A<br>Bd − 1 → Bd<br>Br ⊕ r → Br | Bd decrements past 0 | Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r |
| XIS | r | -4 | 00 r 0100 | RAM(B) ⟷ A<br>Bd + 1 → Bd<br>Br ⊕ r → Br | Bd increments past 15 | Exchange RAM with A and Increment Bd, Exclusive-OR Br with r |

Table 3. COP410L/411L Instruction Set (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **REGISTER REFERENCE INSTRUCTIONS** | | | | | | |
| CAB | | 50 | 0 1 0 1 \| 0 0 0 0 | A → Bd | None | Copy A to Bd |
| CBA | | 4E | 0 1 0 0 \| 1 1 1 0 | Bd → A | None | Copy Bd to A |
| LBI | r,d | -- | 0 0 \| r \| (d − 1) | r,d → B | Skip until not a LBI | Load B Immediate with r,d (Note 5) |
| | | | (d = 0, 9:15) | | | |
| LEI | y | 33 | 0 0 1 1 \| 0 0 1 1 | y → EN | None | Load EN Immediate (Note 6) |
| | | 6- | 0 1 1 0 \| y | | | |
| **TEST INSTRUCTIONS** | | | | | | |
| SKC | | 20 | 0 0 1 0 \| 0 0 0 0 | | C = "1" | Skip if C is True |
| SKE | | 21 | 0 0 1 0 \| 0 0 0 1 | | A = RAM(B) | Skip if A Equals RAM |
| SKGZ | | 33 | 0 0 1 1 \| 0 0 1 1 | | $G_{3:0}$ = 0 | Skip if G is Zero (all 4 bits) |
| | | 21 | 0 0 1 0 \| 0 0 0 1 | | | |
| SKGBZ | | 33 | 0 0 1 1 \| 0 0 1 1 | 1st byte | | Skip if G Bit is Zero |
| | 0 | 01 | 0 0 0 0 \| 0 0 0 1 | 2nd byte | $G_0$ = 0 | |
| | 1 | 11 | 0 0 0 1 \| 0 0 0 1 | | $G_1$ = 0 | |
| | 2 | 03 | 0 0 0 0 \| 0 0 1 1 | | $G_2$ = 0 | |
| | 3 | 13 | 0 0 0 1 \| 0 0 1 1 | | $G_3$ = 0 | |
| SKMBZ | 0 | 01 | 0 0 0 0 \| 0 0 0 1 | | $RAM(B)_0$ = 0 | Skip if RAM Bit is Zero |
| | 1 | 11 | 0 0 0 1 \| 0 0 0 1 | | $RAM(B)_1$ = 0 | |
| | 2 | 03 | 0 0 0 0 \| 0 0 1 1 | | $RAM(B)_2$ = 0 | |
| | 3 | 13 | 0 0 0 1 \| 0 0 1 1 | | $RAM(B)_3$ = 0 | |
| **INPUT/OUTPUT INSTRUCTIONS** | | | | | | |
| ING | | 33 | 0 0 1 1 \| 0 0 1 1 | G → A | None | Input G Ports to A |
| | | 2A | 0 0 1 0 \| 1 0 1 0 | | | |
| INL | | 33 | 0 0 1 1 \| 0 0 1 1 | $L_{7:4}$ → RAM(B)<br>$L_{3:0}$ → A | None | Input L Ports to RAM, A |
| | | 2E | 0 0 1 0 \| 1 1 1 0 | | | |
| OBD | | 33 | 0 0 1 1 \| 0 0 1 1 | Bd → D | None | Output Bd to D Outputs |
| | | 3E | 0 0 1 1 \| 1 1 1 0 | | | |
| OMG | | 33 | 0 0 1 1 \| 0 0 1 1 | RAM(B) → G | None | Output RAM to G Ports |
| | | 3A | 0 0 1 1 \| 1 0 1 0 | | | |
| XAS | | 4F | 0 1 0 0 \| 1 1 1 1 | A ⟷ SIO, C → SKL | None | Exchange A with SIO (Note 2) |

**Note 1:** All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, $A_3$ indicates the most significant (left-most) bit of the 4-bit A register.

**Note 2:** For additional information on the operation of the XAS, JID, and LQID instructions, see below.

**Note 3:** The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

**Note 4:** A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

**Note 5:** The machine code for the lower 4 bits of the LBI instruction equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 ($1001_2$), the lower 4 bits of the LBI instruction equal 8 ($1000_2$). To load 0, the lower 4 bits of the LBI instruction should equal 15 ($1111_2$).

**Note 6:** Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register.)

The following information is provided to assist the user in understanding the operation of several unique instructions and to provide notes useful to programmers in writing COP410L/411L programs.

### XAS Instruction

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register. The contents of SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output. (See Functional Description, EN Register, above.) If SIO is selected as a shift register, an XAS instruction must be performed once every 4 instruction cycles to effect a continuous data stream.

### JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM addressed by the 9-bit word, $PC_8$, A, M. $PC_8$ is not affected by this instruction.

Note that JID requires 2 instruction cycles.

### LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 9-bit word $PC_8$, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB) and replaces the least significant 8 bits of PC as follows: A → $PC_{7:4}$, RAM(B) → $PC_{3:0}$, leaving $PC_8$ unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped" (SB → SA → PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SA → SB, the previous contents of SB are lost. Also, when LQID pops the stack, the previously pushed contents of SA are left in SB. The net result is that the contents of SA are placed in SB (SA → SB). Note that LQID takes two instruction cycle times to execute.

### Instruction Set Notes

a. The first word of a COP410L/411L program (ROM address 0) must be a CLRA (Clear A) instruction.

b. Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths take the same number of cycle times whether instructions are skipped or executed.

c. The ROM is organized into 8 pages of 64 words each. The Program Counter is a 9-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example: a JP located in the last word of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3 or 7 will access data in the next group of 4 pages.

### OPTION LIST

The COP410L/411L mask-programmable options are assigned numbers which correspond with the COP410L pins.

The following is a list of COP410L options. When specifying a COP411L chip, Option 2 must be set to 3, Options 20, 21, and 22 to 0. The options are programmed at the same time as the ROM pattern to provide the user with the hardware flexibility to interface to various I/O components using little or no external circuitry.

Option 1 = 0: Ground Pin — no options available

Option 2: CKO Output (no option available for COP411L)
- = 0: clock output to ceramic resonator
- = 1: pin is RAM power supply ($V_R$) input
- = 2: multi-COP SYNC input
- = 3: No Connection

Option 3: CKI Input
- = 0: oscillator input divided by 8 (500 kHz max)
- = 1: single-pin RC controlled oscillator divided by 4

Option 4: RESET Input
- = 0: load device to $V_{CC}$
- = 1: Hi-Z input

Option 5: $L_7$ Driver
- = 0: Standard output
- = 1: Open-drain output
- = 2: High current LED direct segment drive output
- = 3: High current TRI-STATE® push-pull output
- = 4: Low-current LED direct segment drive output
- = 5: Low-current TRI-STATE® push-pull output

Option 6: $L_6$ Driver
same as Option 5

Option 7: $L_5$ Driver
same as Option 5

Option 8: $L_4$ Driver
same as Option 5

Option 9: $V_{CC}$ Pin
- = 0: 4.5V to 6.3V operation
- = 1: 4.5V to 9.5V operation

Option 10: $L_3$ Driver
same as Option 5

Option 11: $L_2$ Driver
same as Option 5

Option 12: $L_1$ Driver
same as Option 5

Option 13: $L_0$ Driver
same as Option 5

Option 14: SI Input
- = 0: load device to $V_{CC}$
- = 1: HI-Z input

Option 15: SO Driver
= 0: standard output
= 1: open-drain output
= 2: push-pull output

Option 16: SK Driver
same as Option 15

Option 17: $G_0$ I/O Port
= 0: standard output
= 1: open-drain output

Option 18: $G_1$ I/O Port
same as Option 17

Option 19: $G_2$ I/O Port
same as Option 17

Option 20: $G_3$ I/O Port (no option available for COP411L)
same as Option 17

Option 21: $D_3$ Output (no option available for COP411L)
= 0: very-high sink current standard output
= 1: very-high sink current open-drain output
= 2: high sink current standard output
= 3: high sink current open-drain output
= 4: standard LSTTL output (fanout = 1)
= 5: open-drain LSTTL output (fanout = 1)

Option 22: $D_2$ Output (no option available for COP411L)
same as Option 21

Option 23: $D_1$ Output
same as Option 21

Option 24: $D_0$ Output
same as Option 21

Option 25: L Input Levels
= 0: standard TTL input levels ("0" = 0.8V, "1" = 2.0V)
= 1: higher voltage input levels ("0" = 1.2V, "1" = 3.6V)

Option 26: G Input Levels
same as Option 25

Option 27: SI Input Levels
same as Option 25

Option 28: COP Bonding
= 0: COP410L (24-pin device)
= 1: COP411L (20-pin device)

National Semiconductor

# COP420/COP421 AND COP320/321 Single-Chip N-Channel Microcontrollers

## General Description

The COP420, COP421, COP320, COP321 Single-Chip N-Channel Microcontrollers are members of the COPS™ family, fabricated using N-channel, silicon gate MOS technology. They are complete microcomputers containing all system timing, internal logic, ROM, RAM and I/O necessary to implement dedicated control functions in a variety of applications. Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture and I/O scheme designed to facilitate keyboard input, display output and BCD data manipulation. The COP421 is identical to the COP420, except with 19 I/O lines instead of 23. They are an appropriate choice for use in numerous human interface control environments. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized Controller Oriented Processor at a low end-product cost.

The COP320 is the extended temperature range version of the COP420 (likewise the COP321 is the extended temperature range version of the COP421). The COP320/321 are exact functional equivalents of the COP420/421.

## Features

- Low cost
- Powerful instruction set
- 1k × 8 ROM, 64 × 4 RAM
- 23 I/O lines (COP420, COP320)
- True vectored interrupt, plus restart
- Three-level subroutine stack
- 4.0 μs instruction time
- Single supply operation
- Internal time-base counter for real-time processing
- Internal binary counter register with MICROWIRE™ compatible serial I/O
- General purpose and TRI-STATE® outputs
- TTL/CMOS compatible in and out
- LED direct drive outputs
- MICROBUS™ compatible
- Software/hardware compatible with other members of COP400 family
- Extended temperature range device COP320/COP321 (−40 °C to +85 °C)

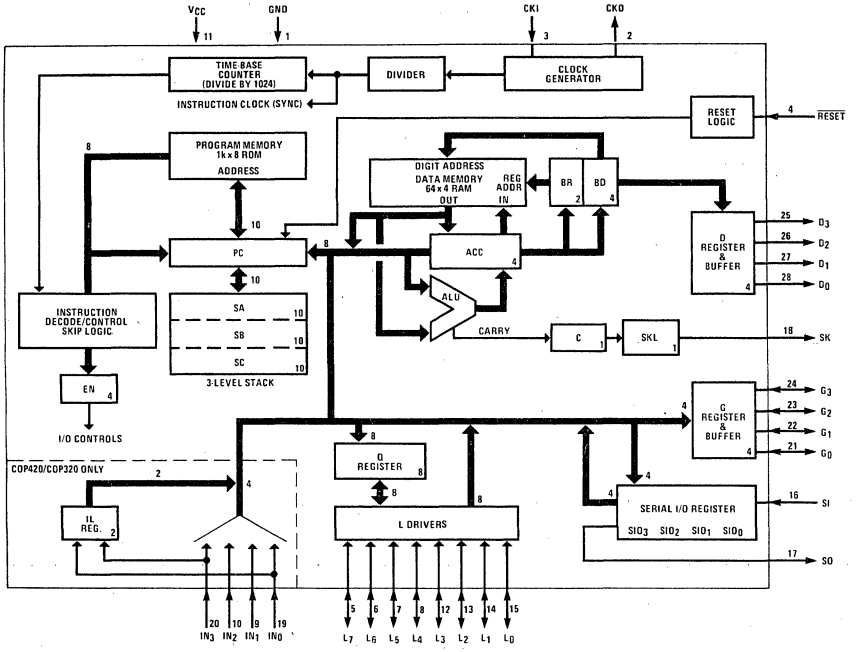



**Figure 1. COP420/COP421, COP320/COP321 Block Diagram**

## COP420/COP421

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Relative to GND | −0.5V to +7V |
| Ambient Operating Temperature | 0°C to +70°C |
| Ambient Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |
| Power Dissipation | 0.75 Watt at 25°C<br>0.4 Watt at 70°C |

*Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.*

## DC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 6.3V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Operating Voltage ($V_{CC}$) | | 4.5 | 6.3 | V |
| Operating Supply Current | Note 3 (all outputs open) | | 30 | mA |
| Input Voltage Levels | | | | |
| CKI Input Levels | | | | |
| Crystal Input | | | | |
| Logic High ($V_{IH}$) | | 2.0 | | V |
| Logic Low ($V_{IL}$) | | | 0.4 | V |
| TTL Input | $V_{CC} = 5V \pm 5\%$ | | | |
| Logic High ($V_{IH}$) | | 2.0 | | V |
| Logic Low ($V_{IL}$) | | −0.3 | 0.8 | V |
| Schmitt Trigger Input (÷4) | | | | |
| Logic High ($V_{IH}$) | | 0.7 $V_{CC}$ | | V |
| Logic Low ($V_{IL}$) | | −0.3 | 0.6 | V |
| RESET Input Levels | | | | |
| Logic High | | 0.7 $V_{CC}$ | | V |
| Logic Low | | −0.3 | 0.6 | V |
| RESET Hysteresis | | 1.0 | | V |
| SO Input Level (Test mode) | | 2.0 | 3.0 | V |
| All Other Inputs | | | | |
| Logic High | $V_{CC} = max$ | 3.0 | | V |
| Logic High | $V_{CC} = 5V \pm 5\%$ | 2.0 | | V |
| Logic Low | | −0.3 | 0.8 | V |
| Input Levels High Trip Option | | | | |
| Logic High | | 3.6 | | V |
| Logic Low | | −0.3 | 1.2 | V |
| Input Capacitance | | | 7 | pF |
| Hi Z Input Leakage | | −1 | +1 | μA |
| Output Voltage Levels | | | | |
| Standard Output | | | | |
| TTL Operation | $V_{CC} = 5V \pm 5\%$ | | | |
| Logic High ($V_{OH}$) | $I_{OH} = -100\mu A$ | 2.4 | | V |
| Logic Low ($V_{OL}$) | $I_{OL} = 1.6mA$ | −0.3 | 0.4 | V |
| CMOS Operation | | | | |
| Logic High ($V_{OH}$) | $I_{OH} = -10\mu A$ | $V_{CC} - 1$ | | V |
| Logic Low ($V_{OL}$) | $I_{OL} = 10\mu A$ | −0.3 | 0.2 | V |
| Output Current Levels | | | | |
| LED Direct Drive Output | $V_{CC} = 6V$ | | | |
| Logic High ($I_{OH}$) | $V_{OH} = 2.0V$ | 2.5 | 14 | mA |
| TRI-STATE® Output Leakage Current | | −10 | +10 | μA |
| CKO Output | | | | |
| $V_R$ Power Saving Option Power Requirements | $V_R = 3.3V$ | | 3 | mA |

## COP420/COP421

## AC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 6.3V$ unless otherwise stated.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Instruction Cycle Time – $t_C$ | figure 3 | 4 | 10 | μs |
| CKI Using Crystal (figure 8a) | | | | |
| Input Frequency – $f_I$ | ÷16 mode | 1.6 | 4 | MHz |
| | ÷8 mode | 0.8 | 2 | MHz |
| Duty Cycle (Note 1) | figure 3a | 30 | 55 | % |
| CKI Using External Clock (figure 8b) | | | | |
| Input Frequency | ÷16 mode | 1.6 | 4 | MHz |
| | ÷8 mode | 0.8 | 2 | MHz |
| Duty Cycle (Note 1) | | 30 | 60 | % |
| Rise Time | $f_I = 4MHz$ | | 60 | ns |
| Fall Time | $F_I = 4MHz$ | | 40 | ns |
| CKI Using RC (figure 8c) | ÷4 mode | | | |
| Frequency | $R = 15k \pm 5\%$, $C = 100pF \pm 10\%$ | 0.5 | 1.0 | MHz |
| Instruction Cycle Time | | 4 | 8 | μs |
| CKO as SYNC Input (figure 8d) | | | | |
| $t_{SYNO}$ | figure 3a | 50 | | ns |
| INPUTS: (figure 3) | | | | |
| $IN_3$-$IN_0$, $G_3$-$G_0$, $L_7$-$L_0$, CKO as Input | | | | |
| $t_{SETUP}$ | | 1.7 | | μs |
| $t_{HOLD}$ | | 300 | | ns |
| SI | | | | |
| $t_{SETUP}$ | | 0.3 | | μs |
| $t_{HOLD}$ | | 250 | | ns |
| OUTPUTS: | | | | |
| COP TO CMOS PROPOGATION DELAY | $4.5V \leq V_{CC} \leq 6.3V$, $C_L = 50pF$, $V_{OH} = 0.7V_{CC}$, $V_{OL} = 0.3V_{CC}$ | | | |
| SK as a Logic-Controlled Clock | | | | |
| $t_{PD1}$ | | | 1.1 | μs |
| $t_{PD0}$ | | | 0.3 | μs |
| SO, SK as a Data Output | | | | |
| $t_{PD1}$ | | | 1.4 | μs |
| $t_{PD0}$ | | | 0.3 | μs |
| $t_{PD1}$ | $V_{OH} = 2V$ | | 0.7 | μs |
| $D_3$-$D_0$, $G_3$-$G_0$ | | | | |
| $t_{PD1}$ | | | 1.6 | μs |
| $t_{PD0}$ | | | 0.6 | μs |
| $L_7$-$L_0$ (Standard) | | | | |
| $t_{PD1}$ | | | 1.4 | μs |
| $t_{PD0}$ | | | 0.3 | μs |
| $L_7$-$L_0$ (LED Direct Drive) | | | | |
| $t_{PD1}$ | $V_{OH} = 2V$ | | 2.4 | μs |
| $t_{PD0}$ | | | 0.4 | μs |

## COP420/COP421

## AC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 6.3V$ unless otherwise stated.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| OUTPUTS (cont.): | | | | |
| COP TO TTL PROPOGATION DELAY | fanout = 1 Standard TTL Load<br>$V_{CC} = 5V \pm 5\%$, $C_L = 50pF$<br>$V_{OH} = 2.4V$, $V_{OL} = 0.4V$ | | | |
| SK as a Logic-Controlled Clock | | | | |
| $t_{PD1}$ | | | 0.8 | μs |
| $t_{PD0}$ | | | 0.8 | μs |
| SK as a Data Output, SO | | | | |
| $T_{PD1}$ | | | 1.0 | μs |
| $T_{PD0}$ | | | 1.0 | μs |
| $D_3$-$D_0$, $G_3$-$G_0$ | | | | |
| $t_{PD1}$ | | | 1.3 | μs |
| $t_{PD0}$ | | | 1.3 | μs |
| $L_7$-$L_0$ | | | | |
| $t_{PD1}$ | | | 1.4 | μs |
| $t_{PD0}$ | | | 0.4 | μs |
| $L_7$-$L_0$ (Push-Pull) | | | | |
| $t_{PD1}$ | | | 0.4 | μs |
| $t_{PD0}$ | | | 0.3 | μs |
| CKO (figure 3b) | | | | |
| $t_{PD1}$ | | | 0.2 | μs |
| $t_{PD0}$ | | | 0.2 | μs |
| MICROBUS™ TIMING | $C_L = 50pF$, $V_{CC} = 5V \pm 5\%$ | | | |
| A. Read Operation (figure 4) | | | | |
| Chip Select Stable Before $\overline{RD}$ – $t_{CSR}$ | | 50 | | ns |
| Chip Select Hold Time for $\overline{RD}$ – $t_{RCS}$ | | 5 | | ns |
| $\overline{RD}$ Pulse Width – $t_{RR}$ | | 400 | | ns |
| Data Delay from $\overline{RD}$ – $t_{RD}$ | | | 300 | ns |
| $\overline{RD}$ to Data Floating – $t_{DF}$ | | | 200 | ns |
| Write Operation (figure 5) | | | | |
| Chip Select Stable Before $\overline{WR}$ – $t_{CSW}$ | | 50 | | ns |
| Chip Select Hold Time for $\overline{WR}$ – $t_{WCS}$ | | 30 | | ns |
| $\overline{WR}$ Pulse Width – $t_{WW}$ | | 350 | | ns |
| Data Set-Up Time for $\overline{WR}$ – $t_{DW}$ | | 300 | | ns |
| Data Hold Time for $\overline{WR}$ – $t_{WD}$ | | 40 | | ns |
| INTR Transition Time from $\overline{WR}$ – $t_{WI}$ | | | 700 | ns |

**Note 1:** Duty Cycle = $t_{WI}/(t_{WI} + t_{WO})$.
**Note 2:** See figure 9 for additional I/O Characteristics
**Note 3:** $V_{CC}$ voltage change must be less than 0.5V/ms to maintain proper operation.

## COP320/COP321

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Relative to GND | −0.5V to +7V |
| Ambient Operating Temperature | −40°C to +85°C |
| Ambient Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |
| Power Dissipation | 0.75 Watt at 25°C<br>0.25 Watt at 85°C |

*Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.*

## DC Electrical Characteristics

$-40°C \leq T_A \leq +85°C$, $4.5V \leq V_{CC} \leq 5.5V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Operating Voltage ($V_{CC}$) | Note 4 | 4.5 | 5.5 | V |
| Operating Supply Current | (all outputs open) | | 40 | mA |
| Input Voltage Levels | | | | |
| CKI Input Levels | | | | |
| Crystal Input | | | | |
| Logic High ($V_{IH}$) | | 2.2 | | V |
| Logic Low ($V_{IL}$) | | | 0.3 | V |
| TTL Input | | | | |
| Logic High ($V_{IH}$) | | 2.2 | | V |
| Logic Low ($V_{IL}$) | | −0.3 | 0.6 | V |
| Schmitt Trigger Input (÷4) | | | | |
| Logic High ($V_{IH}$) | | 0.7 $V_{CC}$ | | V |
| Logic Low ($V_{IL}$) | | | 0.4 | V |
| $\overline{\text{RESET}}$ Input Levels | | | | |
| Logic High | | 0.7 $V_{CC}$ | | V |
| Logic Low | | −0.3 | 0.4 | V |
| $\overline{\text{RESET}}$ Hysteresis | | 0.5 | | V |
| SO Input Level (Test mode) | | 2.2 | 3.0 | V |
| All Other Inputs | | | | |
| Logic High | | 2.2 | | V |
| Logic Low | | −0.3 | 0.6 | V |
| Input Levels High Trip Option | | | | |
| Logic High | | 3.6 | | V |
| Logic Low | | −0.3 | 1.2 | V |
| Input Capacitance | | | 7 | pF |
| Hi Z Input Leakage | | −1 | +1 | μA |
| Output Voltage Levels | | | | |
| Standard Output | | | | |
| TTL Operation | $V_{CC} = 5V \pm 5\%$ | | | |
| Logic High ($V_{OH}$) | $I_{OH} = -75\mu A$ | 2.4 | | V |
| Logic Low ($V_{OL}$) | $I_{OL} = 1.6mA$ | −0.3 | 0.4 | V |
| CMOS Operation | | | | |
| Logic High ($V_{OH}$) | $I_{OH} = -10\mu A$ | $V_{CC} - 1$ | | V |
| Logic Low ($V_{OL}$) | $I_{OL} = 10\mu A$ | −0.3 | 0.2 | V |
| Output Current Levels | | | | |
| LED Direct Drive Output | $V_{CC} = 5V$ (Note 1) | | | |
| Logic High ($I_{OH}$) | $V_{OH} = 2.0V$ | 1.0 | 12 | mA |
| TRI-STATE® Output Leakage Current | | −10 | +10 | μA |
| CKO Output | | | | |
| $V_R$ Power Saving Option Power Requirements | $V_R = 3.3V$ to 5.5V | | 4 | mA |

## COP320/COP321

## AC Electrical Characteristics $-40°C \leq T_A \leq +85°C$, $4.5V \leq V_{CC} \leq 5.5V$ unless otherwise stated.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Instruction Cycle Time – $t_C$ | figure 3 | 4 | 10 | μs |
| CKI Using Crystal (figure 8a) | | | | |
| Input Frequency – $f_I$ | ÷16 mode | 1.6 | 4 | MHz |
| | ÷8 mode | 0.8 | 2 | MHz |
| Duty Cycle (Note 2) | figure 3a | 40 | 55 | % |
| CKI Using External Clock (figure 8b) | | | | |
| Input Frequency | ÷16 mode | 1.6 | 4 | MHz |
| | ÷8 mode | 0.8 | 2 | MHz |
| Duty Cycle (Note 2) | | 40 | 60 | % |
| Rise Time | $f_I = 4$ MHz | | 60 | ns |
| Fall Time | $F_I = 4$ MHz | | 40 | ns |
| CKI Using RC (figure 8c) | ÷4 mode | | | |
| Frequency | $R = 15k \pm 5\%$, $C = 100pF \pm 10\%$ | 0.5 | 1.0 | MHz |
| Instruction Cycle Time | | 4 | 8 | μs |
| CKO as SYNC Input (figure 8d) | | | | |
| $t_{SYNO}$ | figure 3a | 50 | | ns |
| INPUTS: (figure 3) | | | | |
| $IN_3$-$IN_0$, $G_3$-$G_0$, $L_7$-$L_0$, CKO as Input | | | | |
| $t_{SETUP}$ | | 1.7 | | μs |
| $t_{HOLD}$ | | 300 | | ns |
| SI | | | | |
| $t_{SETUP}$ | | 0.3 | | μs |
| $t_{HOLD}$ | | 250 | | ns |
| OUTPUTS: | | | | |
| COP TO CMOS PROPOGATION DELAY | $C_L = 50pF$, $V_{OH} = 0.7V_{CC}$, $V_{OL} = 0.3V_{CC}$ | | | |
| SK as a Logic-Controlled Clock | | | | |
| $t_{PD1}$ | | | 1.3 | μs |
| $t_{PD0}$ | | | 0.4 | μs |
| SO, SK as a Data Output | | | | |
| $t_{PD1}$ | | | 1.6 | μs |
| $t_{PD0}$ | | | 0.4 | μs |
| $t_{PD1}$ | $V_{OH} = 2V$ | | 0.75 | μs |
| $D_3$-$D_0$, $G_3$-$G_0$ | | | | |
| $t_{PD1}$ | | | 2.0 | μs |
| $t_{PD0}$ | | | 1.0 | μs |
| $L_7$-$L_0$ (Standard) | | | | |
| $t_{PD1}$ | | | 2.0 | μs |
| $t_{PD0}$ | | | 1.0 | μs |
| $L_7$-$L_0$ (LED Direct Drive) | | | | |
| $t_{PD1}$ | $V_{OH} = 2V$ | | 3.0 | μs |
| $t_{PD0}$ | | | 1.0 | μs |

## COP320/COP321

## AC Electrical Characteristics $-40°C \leq T_A \leq +85°C$, $4.5V \leq V_{CC} \leq 5.5V$ unless otherwise stated.

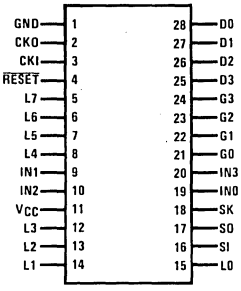
| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| OUTPUTS (cont.): | | | | |
| COP TO TTL PROPOGATION DELAY | fanout = 1 Standard TTL Load<br>$V_{CC} = 5V \pm 5\%$, $C_L = 50pF$<br>$V_{OH} = 2.4V$, $V_{OL} = 0.4V$ | | | |
| SK as a Logic-Controlled Clock | | | | |
| $t_{PD1}$ | | | 1.0 | μs |
| $t_{PD0}$ | | | 1.0 | μs |
| SK as a Data Output, SO | | | | |
| $T_{PD1}$ | | | 1.2 | μs |
| $T_{PD0}$ | | | 1.2 | μs |
| $D_3$-$D_0$, $G_3$-$G_0$ | | | | |
| $t_{PD1}$ | | | 1.5 | μs |
| $t_{PD0}$ | | | 1.5 | μs |
| $L_7$-$L_0$ | | | | |
| $t_{PD1}$ | | | 1.6 | μs |
| $t_{PD0}$ | | | 0.5 | μs |
| $L_7$-$L_0$ (Push-Pull) | | | | |
| $t_{PD1}$ | | | 0.5 | μs |
| $t_{PD0}$ | | | 0.5 | μs |
| CKO (figure 3b) | | | | |
| $t_{PD1}$ | | | 0.25 | μs |
| $t_{PD0}$ | | | 0.25 | μs |
| MICROBUS™ TIMING | $C_L = 50pF$, $V_{CC} = 5V \pm 5\%$ | | | |
| A. Read Operation (figure 4) | | | | |
| Chip Select Stable Before $\overline{RD}$ – $t_{CSR}$ | | 60 | | ns |
| Chip Select Hold Time for $\overline{RD}$ – $t_{RCS}$ | | 10 | | ns |
| $\overline{RD}$ Pulse Width – $t_{RR}$ | | 400 | | ns |
| Data Delay from $\overline{RD}$ – $t_{RD}$ | | | 350 | ns |
| $\overline{RD}$ to Data Floating – $t_{DF}$ | | | 250 | ns |
| Write Operation (figure 5) | | | | |
| Chip Select Stable Before $\overline{WR}$ – $t_{CSW}$ | | 100 | | ns |
| Chip Select Hold Time for $\overline{WR}$ – $t_{WCS}$ | | 50 | | ns |
| $\overline{WR}$ Pulse Width – $t_{WW}$ | | 400 | | ns |
| Data Set-Up Time for $\overline{WR}$ – $t_{DW}$ | | 350 | | ns |
| Data Hold Time for $\overline{WR}$ – $t_{WD}$ | | 50 | | ns |
| INTR Transition Time from $\overline{WR}$ – $t_{WI}$ | | | 800 | ns |

**Note 1:** Exercise great care not to exceed maximum device power dissipation limits when direct-driving LEDs (or sourcing similiar loads) at high temperature.

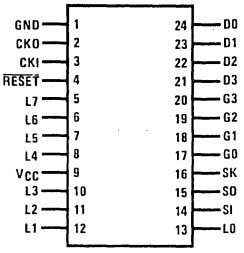
**Note 2:** Duty Cycle = $t_{WI}/(t_{WI} + t_{WO})$.

**Note 3:** See figure 9 for additional I/O Characteristics.

**Note 4:** $V_{CC}$ voltage change must be less than 0.5V/ms.




COP420, COP320




COP421, COP321

Figure 2. Connection Diagrams

| Pin | Description |
|---|---|
| $L_7$-$L_0$ | 8 bidirectional I/O ports with TRI-STATE® |
| $G_3$-$G_0$ | 4 bidirectional I/O ports |
| $D_3$-$D_0$ | 4 general purpose outputs |
| $IN_3$-$IN_0$ | 4 general purpose inputs (COP420/320 only) |
| SI | Serial input (or counter input) |
| SO | Serial output (or general purpose output) |
| SK | Logic-controlled clock (or general purpose output) |

| Pin | Description |
|---|---|
| CKI | System oscillator input |
| CKO | System oscillator output (or general purpose input or RAM power supply) |
| $\overline{\text{RESET}}$ | System reset input |
| $V_{CC}$ | Power supply |
| GND | Ground |

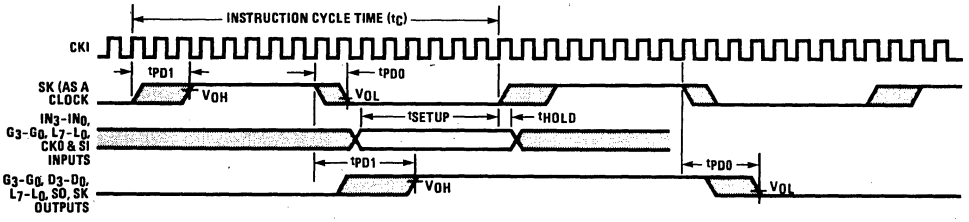



Figure 3. Input/Output Timing Diagrams (crystal divide by 16 mode)

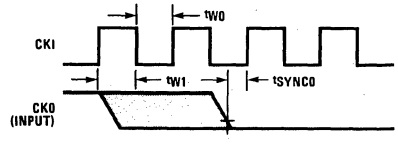



Figure 3A. Synchronization Timing

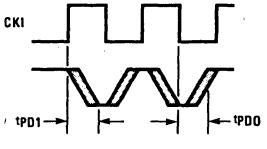



Figure 3B. CKO Output Timing

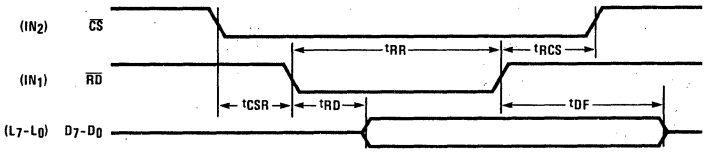



Figure 4. MICROBUS™ Read Operation Timing

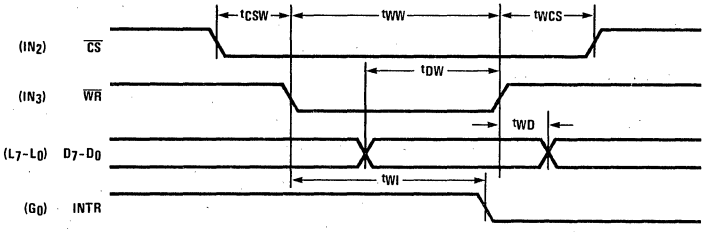



Figure 5. MICROBUS™ Write Operation Timing

## FUNCTIONAL DESCRIPTION COP420/COP421/COP320/COP321

For ease of reading this description, only COP420 and/or COP421 are referenced; however, all such references apply equally to COP320 and/or COP321, respectively.

A block diagram of the COP420 is given in figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1" (greater than 2 volts). When a bit is reset, it is a logic "0" (less than 0.8 volts).

### Program Memory

Program Memory consists of a 1,024 byte ROM. As can be seen by an examination of the COP420/421 instruction set, these words may be program instructions, program data or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID and LQID instructions, ROM must often be thought of as being orgainzed into 16 pages of 64 words each.

ROM addressing is accomplished by a 10-bit PC register. Its binary value selects one of the 1,024 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 10-bit binary count value. Three levels of subroutine nesting are implemented by the 10-bit subroutine save registers, SA, SB and SC, providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the Instruction Decode, Control and Skip Logic circuitry.

### Data Memory

Data memory consists of a 256-bit RAM, organized as 4 data registers of 16 4-bit digits. RAM addressing is implemented by a 6-bit **B register** whose upper 2 bits (Br) select 1 of 4 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) is usually loaded into or from, or exchanged with, the A register (accumulator), it may also be loaded into or from the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the 6-bit contents of the operand field of these instructions. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

### Internal Logic

The 4-bit **A register** (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd portions of the B register, to load and input 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

A **4-bit adder** performs the arithmetic and logic functions of the COP420/421, storing its results in A. It also outputs a carry bit to the 1-bit **C register**, most often employed to indicate arithmetic overflow. The C register, in conjunction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SK or can enable SK to be a sync clock each instruction cycle time. (See XAS instruction and EN register description, below.)

Four **general-purpose inputs, $IN_3$-$IN_0$**, are provided; $IN_1$, $IN_2$ and $IN_3$ may be selected, by a mask-programmable option, as Read Strobe, Chip Select and Write Strobe inputs, respectively, for use in MICROBUS™ applications.

The **D register** provides 4 general-purpose outputs and is used as the destination register for the 4-bit contents of Bd.

The **G register** contents are outputs to 4 general-purpose bidirectional I/O ports. $G_0$ may be mask-programmed as an output for MICROBUS™ applications.

The **Q register** is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control. (See LEI instruction). With the MICROBUS™ option selected, Q can also be loaded with the 8-bit contents of the L I/O ports upon the occurence of a write strobe from the host CPU.

The **8 L drivers**,when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. As explained above, the MICROBUS™ option allows L I/O port data to be latched into the Q register. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the LED Direct Drive output configuration option) with Q data being outputted to the Sa-Sg and decimal point segments of the display.

The **SIO register** functions as a 4-bit serial-in/serial-out shift register or as a binary counter depending on the contents of the EN register. (See EN register description, below.) Its contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O by connecting SO to external serial-in/parallel-out shift registers. For example of additional parallel output cap- see **Application #2.**

The XAS instruction copies C into the **SKL latch.** In the counter mode, SK is the output of SKL; in the shift register mode, SK outputs SKL ANDed with the clock.

The **EN register** is an internal 4-bit register loaded under program control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register ($EN_3$-$EN_0$).

1. The least significant bit of the enable register, $EN_0$, selects the SIO register as either a 4-bit shift register or a 4-bit binary counter. With $EN_0$ set, SIO is an asynchronous binary counter, *decrementing* its value by one upon each low-going pulse ("1" to "0") ocurring on the SI input. Each pulse must be at least two instruction cycles wide. SK outputs the value of SKL. The SO output is equal to the value of $EN_3$. With $EN_0$ reset, SIO is a serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time. (See 4 below.) The SK output becomes a logic-controlled clock.
2. With $EN_1$ set the $IN_1$ input is enabled as an interrupt input. Immediately following an interrupt, $EN_1$ is reset to disable further interrupts.
3. With $EN_2$ set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting $EN_2$ disables the L drivers, placing the L I/O ports in a high-impedance input state.
4. $EN_3$, in conjunction with $EN_0$, affects the SO output. With $EN_0$ set (binary counter option selected) SO will output the value loaded into $EN_3$. With $EN_0$ reset (serial shift register option selected), setting $EN_3$ enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting $EN_3$ with the serial shift register option selected disables SO as the shift register output; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0." The table below provides a summary of the modes associated with $EN_3$ and $EN_0$.

**Enable Register Modes — Bits $EN_3$ and $EN_0$**

| $EN_3$ | $EN_0$ | SIO | SI | SO | SK |
|---|---|---|---|---|---|
| 0 | 0 | Shift Register | Input to Shift Register | 0 | If SKL = 1, SK = CLOCK<br>If SKL = 0, SK = 0 |
| 1 | 0 | Shift Register | Input to Shift Register | Serial Out | If SKL = 1, SK = CLOCK<br>If SKL = 0, SK = 0 |
| 0 | 1 | Binary Counter | Input to Binary Counter | 0 | If SKL = 1, SK = 1<br>If SKL = 0, SK = 0 |
| 1 | 1 | Binary Counter | Input to Binary Counter | 1 | If SKL = 1, SK = 1<br>If SKL = 0, SK = 0 |

## Interrupt

The following features are associated with the $IN_1$ interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

a. The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1) onto the stack, pushing in turn the contents of the other subroutine-save registers to the next lower level (PC + 1 → SA → SB → SC). Any previous contents of SC are lost. The program counter is set to hex address OFF (the last word of page 3) and $EN_1$ is reset.

b. An interrupt will be acknowledged only after the following conditions are met:
   1. $EN_1$ has been set.
   2. A low-going pulse ("1" to "0") at least two instruction cycles wide occurs on the $IN_1$ input.
   3. A currently executing instruction has been completed.
   4. All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction, the interrupt will not be acknowledged until the second JP instruction has been executed.

c. Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon popping of the stack. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skip logic status is saved and program control is transferred to the interrupt servicing routine at hex address OFF. At the *end* of the interrupt routine, a RET instruction is executed to "pop" the stack and return program control to the instruction following the original ASC. *At this time*, the skip logic is enabled and skips this instruction because of the previous ASC carry. Subroutines and LQID instructions should not be nested within the interrupt service routine, since their popping the stack will enable any previously saved main program skips, interfering with the orderly execution of the interrupt routine.

d. The first instruction of the interrupt routine at hex address OFF must be a NOP.

e. A LEI instruction can be put immediately before the RET to re-enable interrupts.

## Microbus™ Interface

The COP420 has an option which allows it to be used as a peripheral microprocessor device, inputting and outputting data from and to a host microprocessor (μP). $IN_1$, $IN_2$ and $IN_3$ general purpose inputs become **MICROBUS™ compatible** read-strobe, chip-select, and write-strobe lines, respectively. $IN_1$ becomes $\overline{RD}$ — a logic "0" on this input will cause Q latch data to be enabled to the L ports for input to the μP. $IN_2$ becomes $\overline{CS}$ — a logic "0" on this line selects the COP420 as the μP peripheral device by enabling the operation of the $\overline{RD}$ and $\overline{WR}$ lines and allows for the selection of one of several peripheral components. $IN_3$ becomes $\overline{WR}$ — a logic "0" on this line will write bus data from the L ports to the Q latches for input to the COP420. $G_0$ becomes INTR a "ready" output, reset by a write pulse from the μP on the $\overline{WR}$ line, providing the "handshaking capability necessary for asynchronous data transfer between the host CPU and the COP420.

This option has been designed for compatibility with National's MICROBUS™ — a standard interconnect system for 8-bit parallel data transfer between MOS/LSI CPUs and interfacing devices. (See MICROBUS™ National Publication.) The functioning and timing relationships between the COP420 signal lines affected by this opotion are as specified for the MICROBUS™ interface, and are given in the AC electrical characteristics and shown in the timing diagrams (figures 4 and 5). Connection of the COP420 to the MICROBUS™ is shown in figure 6.

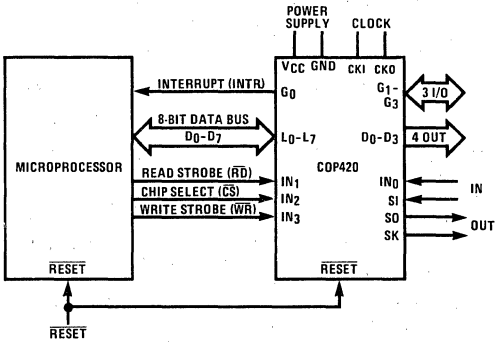



Figure 6. MICROBUS™ Option Interconnect

## Initialization

The Reset Logic, internal to the COP420/421, will initialize (clear) the device upon power-up if the power supply rise time is less than 1ms and greater than 1μs. If the power supply rise time is greater than 1ms, the user must provide an external RC network and diode to the $\overline{RESET}$ pin as shown below. The $\overline{RESET}$ pin is configured as a Schmitt trigger input. If not used it should be connected to $V_{CC}$. Initialization will occur whenever a logic "0" is applied to the $\overline{RESET}$ input, provided it stays low for at least three instruction cycle times.

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, and G registers are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. *Data Memory (RAM) must be cleared by the user's program.* The first instruction at address 0 must be a CLRA.

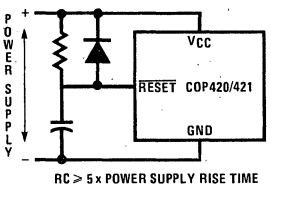



Figure 7. Power-Up Clear Circuit

### Oscillator

There are four basic clock oscillator configurations available as shown by figure 8.

a. **Crystal Controlled Oscillator.** CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency divided by 16 (optional by 8).

b. **External Oscillator.** CKI is an external clock input signal. The external frequency is divided by 16 (optional by 8) to give the instruction cycle time. CKO is now available to be used as the RAM power supply ($V_R$) or as a general purpose input.

c. **RC Controlled Oscillator.** CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 4. CKO is available for non-timing functions.

d. **Externally Synchronized Oscillator.** Intended for use in multi-COP systems, CKO is programmed to function as an input connected to the SK output of another COP420/421 with CKI connected as shown. In this configuration, the SK output connected to CKO must provide a SYNC (instruction cycle) signal to CKO, thereby allowing synchronous data transfer between the COPs using only the SI and SO serial I/O pins in conjunction with the XAS instruction. Note that on power-up SK is automatically enabled as a SYNC output (See Functional Description, Initialization, above).

### CKO Pin Options

In a crystal controlled oscillator system, CKO is used as an output to the crystal network. As an option CKO can be a SYNC input as described above. As another option CKO can be a general purpose input, read into bit 2 of A (accumulator) upon execution of an INIL instruction. As another option, CKO can be a RAM power supply pin ($V_R$), allowing its connection to a standby/backup power supply to maintain the integrity of RAM data with minimum power drain when the main supply is inoperative or shut down to conserve power. Using either option is appropriate in applications where the COP420/421 system timing configuration does not require use of the CKO pin.

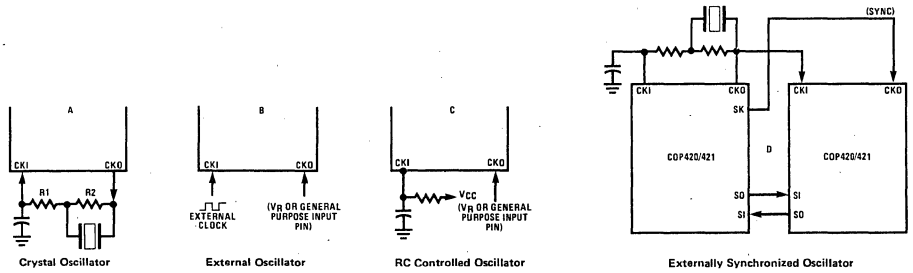
### RAM Keep-Alive Option

Selecting CKO as the RAM power supply ($V_R$) allows the user to shut off the chip power supply ($V_{CC}$) and maintain data in the RAM. To insure that RAM data integrity is maintained, the following conditions must be met:

1. $\overline{\text{RESET}}$ must go low before $V_{CC}$ goes below spec during power-off; $V_{CC}$ must be within spec before $\overline{\text{RESET}}$ goes high on power-up.
2. $V_R$ must be within the operating range of the chip, and equal to $V_{CC} \pm 1V$ during normal operation.
3. $V_R$ must be $\geq 3.3V$ with $V_{CC}$ off.

### CKO Pin Options

In a crystal controlled oscillator system, CKO is used as an output to the crystal network. As an option CKO can be a SYNC input as described above. As another option CKO can be a general purpose input, read into bit 2 of A (accumulator) upon execution of an INIL instruction. As another option, CKO can be a RAM power supply pin ($V_R$), allowing its connection to a standby/backup power supply to maintain the integrity of RAM data with minimum power drain when the main supply is inoperative or shut down to conserve power. Using either option is appropriate in applications where the COP420/421 system timing configuration does not require use of the CKO pin.




**Crystal Oscillator**

| Crystal Value | Component Values | | |
|---|---|---|---|
| | R1 | R2 | C |
| 4 MHz | 1k | 1M | 27 pF |
| 3.58 MHz | 1k | 1M | 27 pF |
| 2.09 MHz | 1k | 1M | 56 pF |

**RC Controlled Oscillator**

| R (kΩ) | C (pF) | Instruction Cycle Time in μs |
|---|---|---|
| 12 | 100 | 5 ± 20% |
| 6.8 | 220 | 5.3 ± 23% |
| 8.2 | 300 | 8 ± 29% |
| 22 | 100 | 8.6 ± 16% |

**Note: 50k ≥ R ≥ 5k**
**360 pF ≥ C ≥ 50 pF**

**Figure 8. COP420/421/COP320/321 Oscillator**

## I/O Options

COP420/421 outputs have the following optional configurations, illustrated in figure 9a:

**a. Standard** — an enhancement mode device to ground in conjunction with a depletion-mode device to $V_{CC}$, compatible with TTL and CMOS input requirements. Available on SO, SK, and all D and G outputs.

**b. Open-Drain** — an enhancement-mode device to ground only, allowing external pull-up as required by the user's application. Available on SO, SK, and all D and G outputs.

**c. Push-Pull** — An enhancement-mode device to ground in conjunction with a depletion-mode device paralleled by an enhancement-mode device to $V_{CC}$. This configuration has been provided to allow for fast rise and fall times when driving capacitive loads. Available on SO and SK outputs only.

**d. Standard L** — same as **a.**, but may be disabled. Available on L outputs only.

**e. Open Drain L** — same as **b.**, but may be disabled. Available on L outputs only.

**f. LED Direct Drive** — an enhancement-mode device to ground and to $V_{CC}$, meeting the typical current sourcing requirements of the segments of an LED display. the sourcing device is clamped to limit current flow. These devices may be turned off under program control (See Functional Description, EN Register), placing the outputs in a high-impedance state to provide required LED segment blanking for a multiplexed display.

**g. TRI-STATE® Push-Pull** — an enhancement-mode device to ground and $V_{CC}$. These outputs are TRI-STATE outputs, allowing for connection of these outputs to a data bus shared by other bus drivers.

COP420/COP421 inputs have the following optional configurations:

**h.** An on-chip depletion load device to $V_{CC}$.

**i.** A Hi-Z input which must be driven to a "1" or "0" by external components.

The above input and output configurations share common enhancement-mode and depletion-mode devices. Specifically, all configurations use one or more of six devices (numbered 1-6, respectively). Minimum and maximum current ($I_{OUT}$ and $V_{OUT}$) curves are given in figure 9b for each of these devices to allow the designer to effectively use these I/O configurations in designing a COP420/421 system.

The SO, SK outputs can be configured as shown in **a.**, **b.**, or **c.** The D and G outputs can be configured as shown in **a.** or **b.** Note that when inputting data to the G ports, the G outputs should be set to "1." The L outputs can be configured as in **d.**, **e.**, **f.** or **g.**

An important point to remember if using configuration **d.** or **f.** with the L drivers is that even when the L drivers are disabled, the depletion load device will source a small amount of current (see figure 9b, device 2); however, when the L lines are used as inputs, the disabled depletion device can *not* be relied on to source sufficient current to pull an input to logic "1".

## COP421

If the COP420 is bonded as a 24-pin device, it becomes the COP421, illustrated in figure 2, COP420/421 Connection Diagrams. Note that the COP421 does not contain the four general purpose IN inputs ($IN_3$-$IN_0$). Use of this option precludes, of course, use of the IN options, interrupt feature, and the MICROBUS™ option which uses $IN_1$-$IN_3$. All other options are available for the COP421.

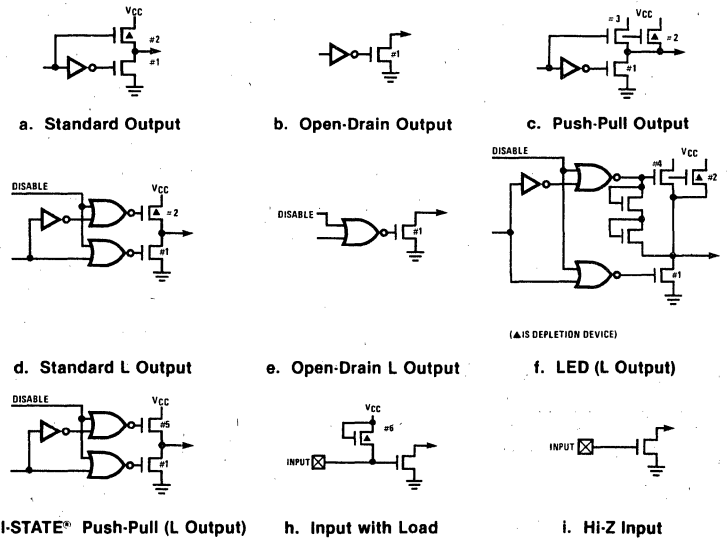



Figure 9a. Input/Output Configurations

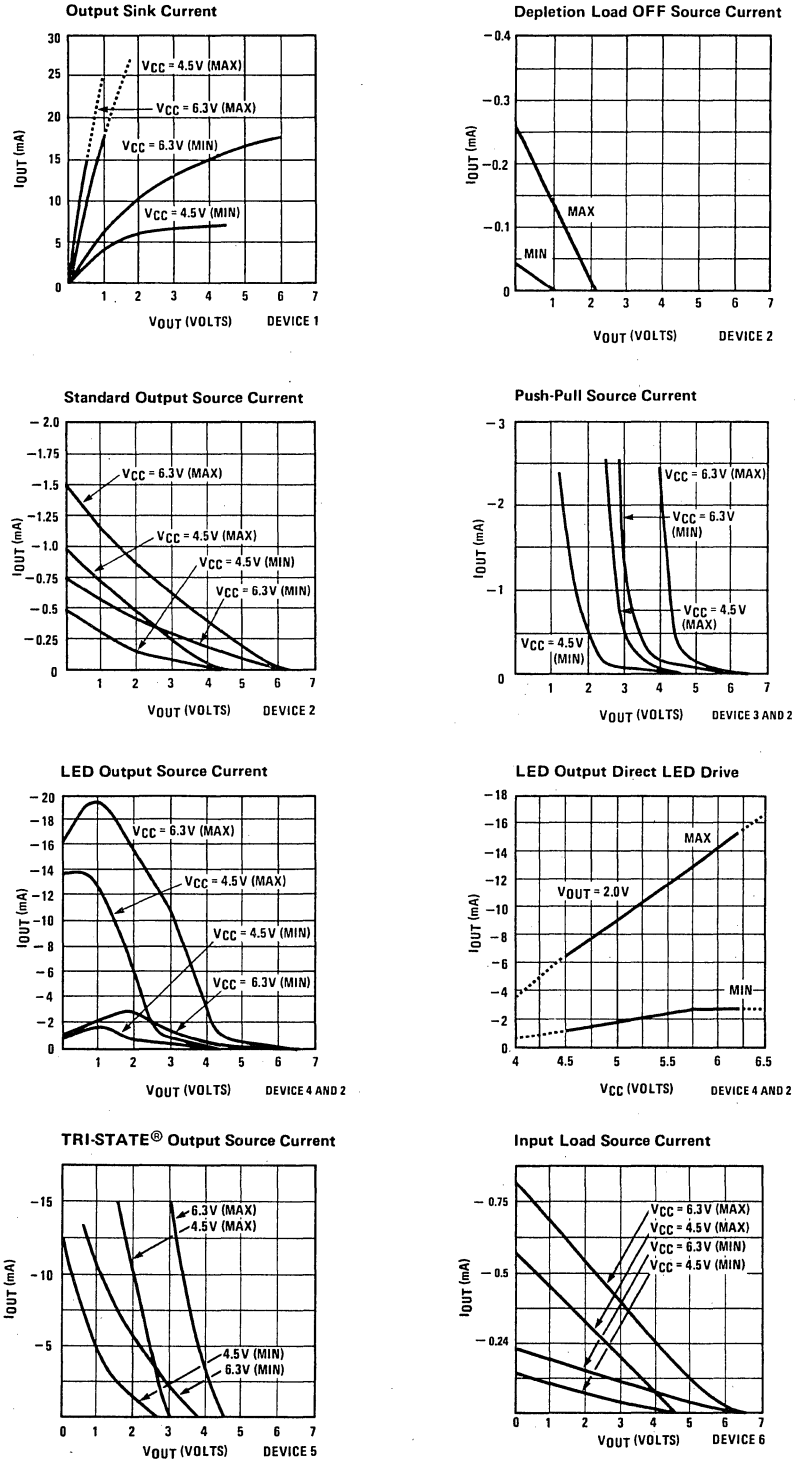



**Figure 9b. COP420/COP421 Input/Output Characteristics**

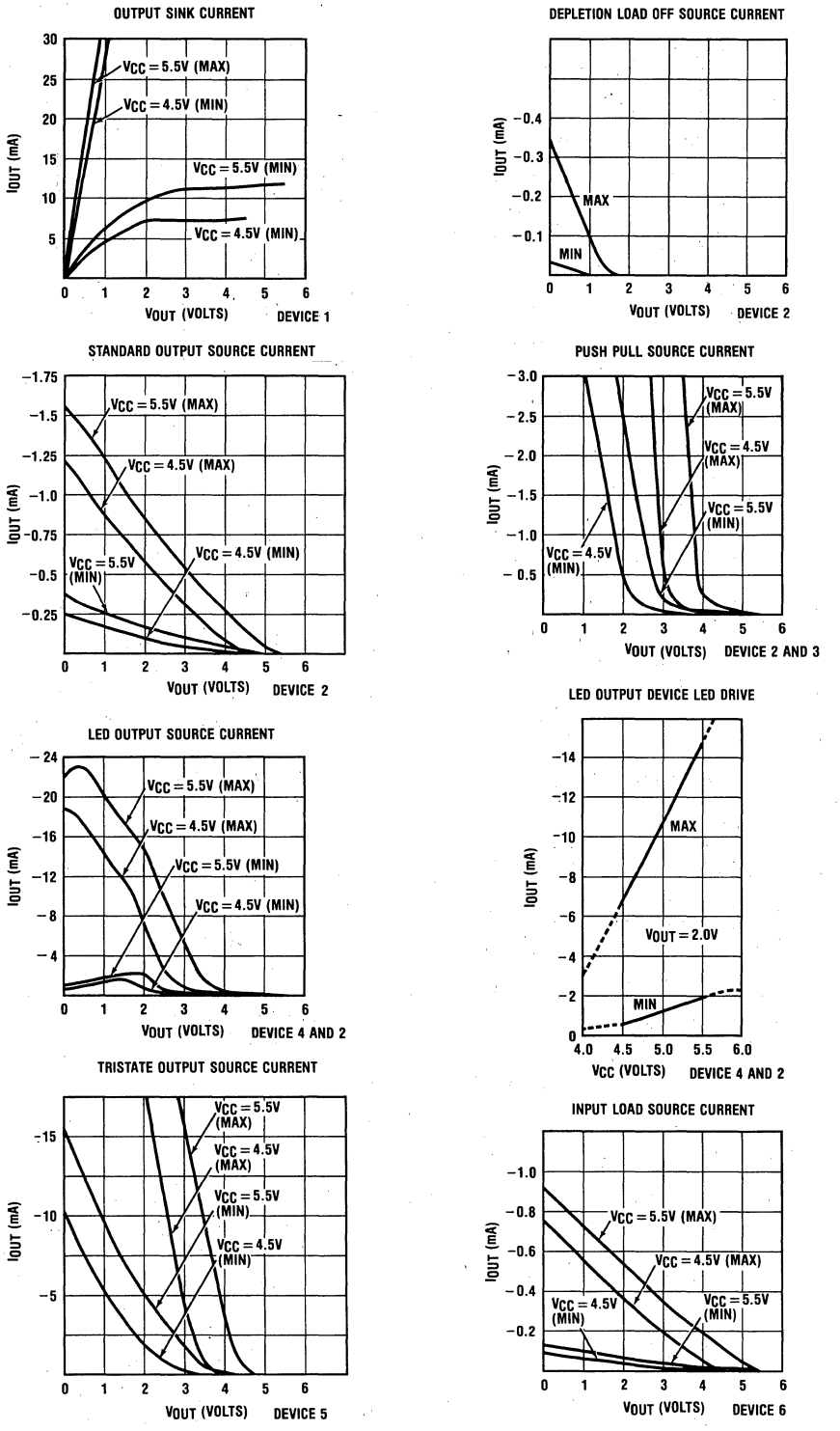



**Figure 9c. COP320/COP321 Input/Output Characteristics**

## COP420/421/320/321 INSTRUCTION SET

Table 1 is a symbol table providing internal architecture, instruction operand and operation symbols used in the instruction set table.

Table 2 provides the mnemonic, operand, machine code, data flow, skip conditions and description associated with each instruction in the COP420/421 instruction set.

**Table 1. COP420/421/320/321 Instruction Set Table Symbols**

| Symbol | Definition |
|---|---|
| INTERNAL ARCHITECTURE SYMBOLS | |
| A | 4-bit Accumulator |
| B | 6-bit RAM Address Register |
| Br | Upper 2 bits of B (register address) |
| Bd | Lower 4 bits of B (digit address) |
| C | 1-bit Carry Register |
| D | 4-bit Data Output Port |
| EN | 4-bit Enable Register |
| G | 4-bit Register to latch data for G I/O Port |
| IL | Two 1-bit Latches associated with the $IN_3$ or $IN_0$ Inputs |
| IN | 4-bit Input Port |
| L | 8-bit TRI-STATE I/O Port |
| M | 4-bit contents of RAM Memory pointed to by B Register |
| PC | 10-bit ROM Address Register (program counter) |
| Q | 8-bit Register to latch data for L I/O Port |
| SA | 10-bit Subroutine Save Register A |
| SB | 10-bit Subroutine Save Register B |
| SC | 10-bit Subroutine Save Register C |
| SIO | 4-bit Shift Register and Counter |
| SK | Logic-Controlled Clock Output |

| Symbol | Definition |
|---|---|
| INSTRUCTION OPERAND SYMBOLS | |
| d | 4-bit Operand Field, 0-15 binary (RAM Digit Select) |
| r | 2-bit Operand Field, 0-3 binary (RAM Register Select) |
| a | 10-bit Operand Field, 0-1023 binary (ROM Address) |
| y | 4-bit Operand Field, 0-15 binary (Immediate Data) |
| RAM(s) | Contents of RAM location addressed by s |
| ROM(t) | Contents of ROM location addressed by t |
| OPERATIONAL SYMBOLS | |
| + | Plus |
| − | Minus |
| → | Replaces |
| ⟷ | Is exchanged with |
| = | Is equal to |
| $\overline{A}$ | The ones complement of A |
| ⊕ | Exclusive-OR |
| : | Range of values |

**Table 2. COP420/421 Instruction Set**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| ARITHMETIC INSTRUCTIONS | | | | | | |
| ASC | | 30 | \|0 0 1 1\|0 0 0 0\| | A + C + RAM(B) → A<br>Carry → C | Carry | Add with Carry, Skip on Carry |
| ADD | | 31 | \|0 0 1 1\|0 0 0 1\| | A + RAM(B) → A | None | Add RAM to A |
| ADT | | 4A | \|0 1 0 0\|1 0 1 0\| | A + $10_{10}$ → A | None | Add Ten to A |
| AISC | y | 5- | \|0 1 0 1\| y \| | A + y → A | Carry | Add Immediate, Skip on Carry (y ≠ 0) |
| CASC | | 10 | \|0 0 0 1\|0 0 0 0\| | $\overline{A}$ + RAM(B) + C → A<br>Carry → C | Carry | Complement and Add with Carry, Skip on Carry |
| CLRA | | 00 | \|0 0 0 0\|0 0 0 0\| | 0 → A | None | Clear A |
| COMP | | 40 | \|0 1 0 0\|0 0 0 0\| | $\overline{A}$ → A | None | Ones complement of A to A |
| NOP | | 44 | \|0 1 0 0\|0 1 0 0\| | None | None | No Operation |
| RC | | 32 | \|0 0 1 1\|0 0 1 0\| | "0" → C | None | Reset C |
| SC | | 22 | \|0 0 1 0\|0 0 1 0\| | "1" → C | None | Set C |
| XOR | | 02 | \|0 0 0 0\|0 0 1 0\| | A ⊕ RAM(B) → A | None | Exclusive-OR RAM with A |

Table 2. COP420/421/320/321 Instruction Set (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **TRANSFER OF CONTROL INSTRUCTIONS** | | | | | | |
| JID | | FF | 1 1 1 1 \| 1 1 1 1 | ROM ($PC_{9:8}$,A,M) → $PC_{7:0}$ | None | Jump Indirect (Note 3) |
| JMP | a | 6- | 0 1 1 0 \| 0 0 \| $a_{9:8}$ | a → PC | None | Jump |
| | | -- | $a_{7:0}$ | | | |
| JP | a | -- | 1 \| $a_{6:0}$ (pages 2,3 only) | a → $PC_{6:0}$ | None | Jump within Page (Note 4) |
| | | | or | | | |
| | | -- | 1 1 \| $a_{5:0}$ (all other pages) | a → $PC_{5:0}$ | | |
| JSRP | a | -- | 1 0 \| $a_{5:0}$ | PC + 1 → SA → SB → SC<br>0010 → $PC_{9:6}$<br>a → $PC_{5:0}$ | None | Jump to Subroutine Page (Note 5 |
| JSR | a | 6- | 0 1 1 0 \| 1 0 $a_{9:8}$ | PC + 1 → SA → SB → SC<br>a → PC | None | Jump to Subroutine |
| | | -- | $a_{7:0}$ | | | |
| RET | | 48 | 0 1 0 0 \| 1 0 0 0 | SC → SB → SA → PC | None | Return from Subroutine |
| RETSK | | 49 | 0 1 0 0 \| 1 0 0 1 | SC → SB → SA → PC | Always Skip on Return | Return from Subroutine then Skip |
| **MEMORY REFERENCE INSTRUCTIONS** | | | | | | |
| CAMQ | | 33 | 0 0 1 1 \| 0 0 1 1 | A → $Q_{7:4}$ | None | Copy A, RAM to Q |
| | | 3C | 0 0 1 1 \| 1 1 0 0 | RAM(B) → $Q_{3:0}$ | | |
| CQMA | | 33 | 0 0 1 1 \| 0 0 1 1 | $Q_{7:4}$ → RAM(B) | None | Copy Q to RAM, A |
| | | 2C | 0 0 1 0 \| 1 1 0 0 | $Q_{3:0}$ → A | | |
| LD | r | -5 | 0 0 \| r \| 0 1 0 1 | RAM(B) → A | None<br>Br ⊕ r → Br | Load RAM into A, Exclusive-OR Br with r |
| LDD | r,d | 23 | 0 0 1 0 \| 0 0 1 1 | RAM(r,d) → A | None | Load A with RAM pointed to directly by r,d |
| | | -- | 0 0 \| r \| d | | | |
| LQID | | BF | 1 0 1 1 \| 1 1 1 1 | ROM($PC_{9:8}$,A,M) → Q<br>SB → SC | None | Load Q Indirect (Note 3) |
| RMB | 0 | 4C | 0 1 0 0 \| 1 1 0 0 | 0 → $RAM(B)_0$ | None | Reset RAM Bit |
| | 1 | 45 | 0 1 0 0 \| 0 1 0 1 | 0 → $RAM(B)_1$ | | |
| | 2 | 42 | 0 1 0 0 \| 0 0 1 0 | 0 → $RAM(B)_2$ | | |
| | 3 | 43 | 0 1 0 0 \| 0 0 1 1 | 0 → $RAM(B)_3$ | | |
| SMB | 0 | 4D | 0 1 0 0 \| 1 1 0 1 | 1 → $RAM(B)_0$ | None | Set RAM Bit |
| | 1 | 47 | 0 1 0 0 \| 0 1 1 1 | 1 → $RAM(B)_1$ | | |
| | 2 | 46 | 0 1 0 0 \| 0 1 1 0 | 1 → $RAM(B)_2$ | | |
| | 3 | 4B | 0 1 0 0 \| 1 0 1 1 | 1 → $RAM(B)_3$ | | |

Table 2. COP420/421/320/321 Instruction Set (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| MEMORY REFERENCE INSTRUCTIONS (continued) | | | | | | |
| STII | y | 7- | 0111 \| y | y → RAM(B)<br>Bd + 1 → Bd | None | Store Memory Immediate and Increment Bd |
| X | r | -6 | 00 \| r \| 0110 | RAM(B) ↔ A<br>Br ⊕ r → Br | None | Exchange RAM with A, Exclusive-OR Br with r |
| XAD | r,d | 23 | 0010 \| 0011 | RAM(r,d) ↔ A | None | Exchange A with RAM pointed to directly by r,d |
| | | -- | 10 \| r \| d | | | |
| XDS | r | -7 | 00 \| r \| 0111 | RAM(B) ↔ A<br>Bd – 1 → Bd<br>Br ⊕ r → Br | Bd decrements past 0 | Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r |
| XIS | r | -4 | 00 \| r \| 0100 | RAM(B) ↔ A<br>Bd + 1 → Bd<br>Br ⊕ r → Br | Bd increments past 15 | Exchange RAM with A and Increment Bd, Exclusive-OR Br with r |
| REGISTER REFERENCE INSTRUCTIONS | | | | | | |
| CAB | | 50 | 0101 \| 0000 | A → Bd | None | Copy A to Bd |
| CBA | | 4E | 0100 \| 1110 | Bd → A | None | Copy Bd to A |
| LBI | r,d | -- | 00 \| r \| (d – 1)<br>(d = 0, 9:15) | r,d → B | Skip until not a LBI | Load B Immediate with r,d (Note 6) |
| | | | or | | | |
| | | 33 | 0011 \| 0011 | | | |
| | | -- | 10 \| r \| d<br>(any d) | | | |
| LEI | y | 33 | 0011 \| 0011 | y → EN | None | Load EN Immediate (Note 7) |
| | | 6- | 0110 \| y | | | |
| XABR | | 12 | 0001 \| 0010 | A ↔ Br (0,0 → $A_3$,$A_2$) | None | Exchange A with Br |
| TEST INSTRUCTIONS | | | | | | |
| SKC | | 20 | 0010 \| 0000 | | C = "1" | Skip if C is True |
| SKE | | 21 | 0010 \| 0001 | | A = RAM(B) | Skip if A Equals RAM |
| SKGZ | | 33 | 0011 \| 0011 | | $G_{3:0} = 0$ | Skip if G is Zero (all 4 bits) |
| | | 21 | 0010 \| 0001 | | | |
| SKGBZ | | 33 | 0011 \| 0011 | 1st byte | | Skip if G Bit is Zero |
| | 0 | 01 | 0000 \| 0001 | 2nd byte | $G_0 = 0$ | |
| | 1 | 11 | 0001 \| 0001 | | $G_1 = 0$ | |
| | 2 | 03 | 0000 \| 0011 | | $G_2 = 0$ | |
| | 3 | 13 | 0001 \| 0011 | | $G_3 = 0$ | |
| SKMBZ | 0 | 01 | 0000 \| 0001 | | $RAM(B)_0 = 0$ | Skip if RAM Bit is Zero |
| | 1 | 11 | 0001 \| 0001 | | $RAM(B)_1 = 0$ | |
| | 2 | 03 | 0000 \| 0011 | | $RAM(B)_2 = 0$ | |
| | 3 | 13 | 0001 \| 0011 | | $RAM(B)_3 = 0$ | |
| SKT | | 41 | 0100 \| 0001 | | A time-base counter carry has occurred since last test | Skip on Timer (Note 3) |

**Table 2. COP420/421/320/321 Instruction Set (continued)**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| INPUT/OUTPUT INSTRUCTIONS | | | | | | |
| ING | | 33 | \|0 0 1 1\|0 0 1 1\| | G → A | None | Input G Ports to A |
| | | 2A | \|0 0 1 0\|1 0 1 0\| | | | |
| ININ | | 33 | \|0 0 1 1\|0 0 1 1\| | IN → A | None | Input IN Inputs to A (Note 2) |
| | | 28 | \|0 0 1 0\|1 0 0 0\| | | | |
| INIL | | 33 | \|0 0 1 1\|0 0 1 1\| | $IL_3$, CKO, "0", $IL_0$ → A | None | Input IL Latches to A (Note 3) |
| | | 29 | \|0 0 1 0\|1 0 0 1\| | | | |
| INL | | 33 | \|0 0 1 1\|0 0 1 1\| | $L_{7:4}$ → RAM(B)<br>$L_{3:0}$ → A | None | Input L Ports to RAM, A |
| | | 2E | \|0 0 1 0\|1 1 1 0\| | | | |
| OBD | | 33 | \|0 0 1 1\|0 0 1 1\| | Bd → D | None | Output Bd to D Outputs |
| | | 3E | \|0 0 1 1\|1 1 1 0\| | | | |
| OGI | y | 33 | \|0 0 1 1\|0 0 1 1\| | y → G | None | Output to G Ports Immediate |
| | | 5- | \|0 1 0 1\| y \| | | | |
| OMG | | 33 | \|0 0 1 1\|0 0 1 1\| | RAM(B) → G | None | Output RAM to G Ports |
| | | 3A | \|0 0 1 1\|1 0 1 0\| | | | |
| XAS | | 4F | \|0 1 0 0\|1 1 1 1\| | A ⟷ SIO, C → SKL | None | Exchange A with SIO (Note 3) |

**Note 1:** All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, $A_3$ indicates the most significant (left-most) bit of the 4-bit A register.

**Note 2:** The ININ instruction is not available on the 24-pin COP421/COP321 since this device does not contain the IN inputs.

**Note 3:** For additional information on the operation of the XAS, JID, LQID, INIL, and SKT instructions, see below.

**Note 4:** The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

**Note 5:** A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

**Note 6:** LBI is a single-byte instruction if d = 0, 9, 10, 11, 12, 13, 14, or 15. The machine code for the lower 4 bits equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 ($1001_2$), the lower 4 bits of the LBI instruction equal 8 ($1000_2$). To load 0, the lower 4 bits of the LBI instruction should equal 15 ($1111_2$).

**Note 7:** Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register.)

The following information is provided to assist the user in understanding the operation of several unique instructions and to provide notes useful to programmers in writing COP420/421 programs.

### XAS Instruction

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register. The contents of SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output. (See Functional Description, EN Register, above.) If SIO is selected as a shift register, an XAS instruction must be performed once every 4 instruction cycles to effect a continuous data stream.

### JID Instruction

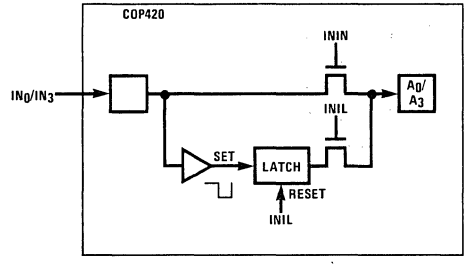
JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM addressed by the 10-bit word, $PC_{9:8}$, A, M. $PC_9$ and $PC_8$ are not affected by this instruction.

Note that JID requires 2 instruction cycles.

### INIL Instruction

INIL (Input IL Latches to A) inputs 2 latches, $IL_3$ and $IL_0$ (see figure 10) and CKO into A. The $IL_3$ and $IL_0$ latches are set if a low-going pulse ("1" to "0") has occurred on the $IN_3$ and $IN_0$ inputs since the last INIL instruction, provided the input pulse stays low for at least two instruction times. Execution of an INIL inputs $IL_3$ and $IL_0$ into A3 and A0 respectively, and resets these latches to allow them to respond to subsequent low-going pulses on the $IN_3$ and $IN_0$ lines. If CKO is mask programmed as a general purpose input, an INIL will input the state of CKO into A2. If CKO has not been so programmed, a "1" will be placed in A2. A "0" is always placed in A1 upon the execution of an INIL. The general purpose inputs $IN_3$–$IN_0$ are input to A upon execution of an ININ instruction. (See table 2, ININ instruction.) INIL is useful in recognizing pulses of short duration or pulses which occur too often to be read conveniently by an ININ instruction.

Note: IL latches are not cleared on reset.




### LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 10-bit word $PC_9$, $PC_8$, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB →SC) and replaces the least significant 8 bits of PC as follows: A → $PC_{7:4}$, RAM(B) → $PC_{3:0}$, leaving $PC_9$ and $PC_8$ unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped" (SC → SB → SA →PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SB → SC, the previous contents of SC are lost. Also, when LQID pops the stack, the previously pushed contents of SB are left in SC. The net result is that the contents of SB are placed in SC (SB → SC). Note that LQID takes two instruction cycle times to execute.

### SKT Instruction

The SKT (Skip On Timer) instruction tests the state of an internal 10-bit time-base counter. This counter divides the instruction cycle clock frequency by 1024 and provides a latched indication of counter overflow. The SKT instruction tests this latch, executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction, therefore, allow the COP420/421 to generate its own time-base for real-time processing rather than relying on an external input signal.

For example, using a 2.097 MHz crystal as the time-base to the clock generator, the instruction cycle clock frequency will be 131 kHz (crystal frequency ÷ 16) and the binary counter output pulse frequency will be 128 Hz. For time-of-day or similar real-time processing, the SKT instruction can call a routine which increments a "seconds" counter every 128 ticks.

### Instruction Set Notes

a. The first word of a COP420/421 program (ROM address 0) must be a CLRA (Clear A) instruction.

b. Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths take the same number of cycle times whether instructions are skipped or executed.

c. The ROM is organized into 16 pages of 64 words each. The Program Counter is an 10-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example: a JP located in the last word of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3, 7, 11 or 15 will access data in the next group of four pages.

## OPTION LIST

The COP420/421 mask-programmable options are assigned numbers which correspond with the COP*420* pins.

The following is a list of COP420 options. When specifying a COP421 chip, Options 9, 10, 19, 20 and 29 must all be set to zero. The options are programmed at the same time as the ROM pattern to provide the user with the hardware flexibility to interface to various I/O components using little or no external circuitry.

Option 1 = 0: Ground Pin — no options available

Option 2: CKO Pin
- = 0: clock generator output to crystal (0 not available if option 3 = 4 or 5)
- = 1: pin is RAM power supply ($V_R$) input
- = 2: general purpose input with load device
- = 3: multi-COP SYNC input
- = 4: general purpose Hi Z input

Option 3: CKI Input
- = 0: crystal input divided by 16
- = 1: crystal input divided by 8
- = 2: TTL external clock input divided by 16
- = 3: TTL external clock input divided by 8
- = 4: single-pin RC controlled oscillator (÷4)
- = 5: Schmitt trigger clock input (÷4)

Option 4: RESET Pin
- = 0: Load devices to $V_{CC}$
- = 1: Hi-Z input

Option 5: $L_7$ Driver
- = 0: Standard output (figure 9D)
- = 1: Open-Drain output (E)
- = 2: LED direct drive output (F)
- = 3: TRI-STATE® push-pull push-pull output (G)

Option 6: $L_6$ Driver
same as Option 5

Option 7: $L_5$ Driver
same as Option 5

Option 8: $L_4$ Driver
same as Option 5

Option 9: $IN_1$ Input
- = 0: load device to $V_{CC}$ (H)
- = 1: Hi-Z input (I)

Option 10: $IN_2$ Input
same as Option 9

Option 11 = 0: $V_{CC}$ Pin — no options available

Option 12: $L_3$ Driver
same as Option 5

Option 13: $L_2$ Driver
same as Option 5

Option 14: $L_1$ Driver
same as Option 5

Option 15: $L_0$ Driver
same as Option 5

Option 16: SI Input
same as Option 9

Option 17: SO Driver
- = 0: standard output (A)
- = 1: open-drain output (B)
- = 2: push-pull output (C)

Option 18: SK Driver
same as Option 17

Option 19: $IN_0$ Input
same as Option 9

Option 20: $IN_3$ Input
same as Option 9

Option 21: $G_0$ I/O Port
- = 0: Standard output (A)
- = 1: Open-Drain output (B)

Option 22: $G_1$ I/O Port
same as Option 21

Option 23: $G_2$ I/O Port
same as Option 21

Option 24: $G_3$ I/O Port
same as Option 21

Option 25: $D_3$ Output
- = 0: Standard output (A)
- = 1: Open-Drain output (B)

Option 26: $D_2$ Output
same as Option 25

Option 27: $D_1$ Output
same as Option 25

Option 28: $D_0$ Output
same as Option 25

Option 29: COP Function
- = 0: normal operation
- = 1: MICROBUS™ option

Option 30: COP Bonding
- = 0: COP420 (28-pin device)
- = 1: COP421 (24-pin device)

Option 31: IN Input Levels
- = 0: Normal input levels
- = 1: Higher voltage input levels ("0" = 1.2V, "1" = 3.6V)

Option 32: G Input Levels
same as Option 31

Option 33: L Input Levels
same as Option 31

Option 34: CKO Input Levels
same as Option 31

Option 35: SI Input Levels
same as Option 31

## TEST MODE (Non-Standard Operation

The SO output has been configured to provide for standard test procedures for the custom-programmed COP420. With SO forced to logic "1," two test modes are provided, depending upon the value of SI:

a. RAM and Internal Logic Test Mode (SI = 1)

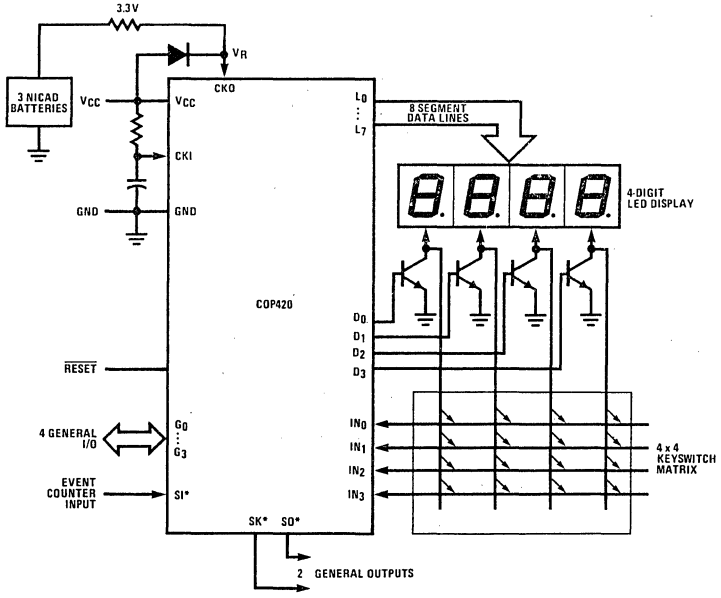
b. ROM Test Mode (SI = 0)

These special test modes should not be employed by the user; they are intended for manufacturing test only.

## APPLICATION #1: COP420 General Controller

Figure 8 shows an interconnect diagram for a COP420 used as a general controller. Operation of the system is as follows:

1. The $L_7$-$L_0$ outputs are configured as LED Direct Drive outputs, allowing direct connection to the segments of the display.
2. The $D_3$-$D_0$ outputs drive the digits of them multiplexed display directly and scan the columns of the 4 x 4 keyboard matrix.
3. The $IN_3$-$IN_0$ inputs are used to input the 4 rows of the keyboard matrix. Reading the IN lines in conjunction with the current value of the D outputs allows detection, debouncing, and decoding of any one of the 16 keyswitches.
4. CKI is configured as a single-pin oscillator input allowing system timing to be controlled by a single-pin RC network. CKO is therefore available for use as a $V_R$ RAM power supply pin. RAM data integrity is thereby assured when the main power supply is shut down (see RAM Keep-Alive Option description).
5. SI is selected as the input to a binary counter input. With SIO used as a binary counter, SO and SK can be used as general purpose outputs.
6. The 4 bidirectional G I/O ports ($G_3$-$G_0$) are available for use as required by the user's application.




*SI, SO and SK may also be used for serial I/O

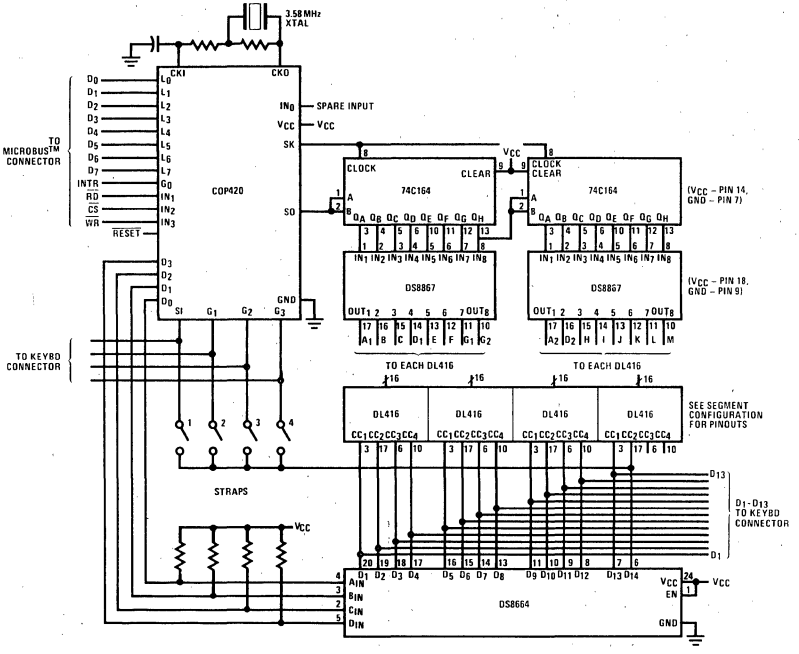
**Figure 11. COP420 Keyboard/Display Interface**

## APPLICATION #2:

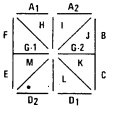
Figure 12 provides an interconnect diagram for a versatile application the COP420 as a keyboard/display interface to a microprocessor ($\mu$P). Generally, operation of the COP420 in this configuration is as follows:

1. The MICROBUS™ option has been selected.
2. System timing is provided by an external crystal. The time base for the real-time (counter and clock) modes is provided by the internal time-base counter, tested by the SKT instruction.
3. The SIO register is used as a serial-in/serial-out shift register. In this configuration, however, SI is shifted into SIO to be tested as one of the 4 row lines tied to the keyboard matrix. SO is used to output display segment data (loaded into SIO with an XAS instruction) to the cascaded 74C164s (8-bit parallel-out serial shift registers). SK functions as a logic-controlled clock, sending a SYNC signal to a clock serial data into the 74C164s.
4. The 16 bits of data shifted into the 74C164s are buffered through the DS8867s (8-segment LED drivers) to the 16 segments of the alpha-numeric LED displays.

5. The $D_0$-$D_1$ outputs are decoded by the DS8864 (14-digit decoder/driver) and used to select one of the 14 digits of the multiplexed display as well as to scan the 13 columns of the keyboard matrix and the strap switch scan line ($D_{14}$).
6. The $G_1$-$G_3$ lines together with SI are connected to the 4 rows of the keyboard matrix and the 4 strap switch lines to input key or strap switch data to the COP420. The strap switches can be used to select one of several ofthe system modes listed below.
7. The $L_0$-$L_7$ TRI-STATE® bidirectional I/O prots are connected to the microprocessor data bus to allow for input or output of data to and from the microprocessor and the COP420.
8. The various operations which can be performed by the system include the following "handshaking" and COP420 "stand-alone" modes:
   a. keyboard to μP (7-bit ASCII)
   b. μP to display
   c. display to μP
   d. display to μP
   e. μP to clock
   f. clock to μP
   g. keyboard to display
   h. clock to display




ALPHANUMERIC DISPLAY SEGMENT CONFIGURATION



| SEGMENT | PIN | SEGMENT | PIN |
|---|---|---|---|
| $A_1$ | 19 | $G_1$ | 1 |
| $A_2$ | 15 | $G_2$ | 14 |
| B | 18 | H | 21 |
| C | 20 | I | 16 |
| $D_1$ | 11 | J | 12 |
| $D_2$ | 4 | K | 9 |
| E | 7 | L | 5 |
| F | 22 | M | 8 |
| | | DP | 2 |

KEYBOARD LAYOUT

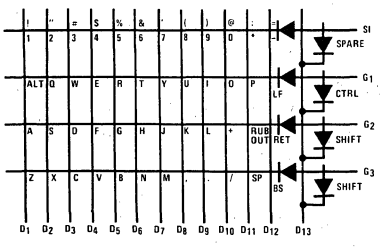



**Figure 12. COP420 Keyboard/Display Interface**

National Semiconductor

# COP420C/COP421C and COP320C/COP321C Single-Chip CMOS Microcontrollers

## General Description

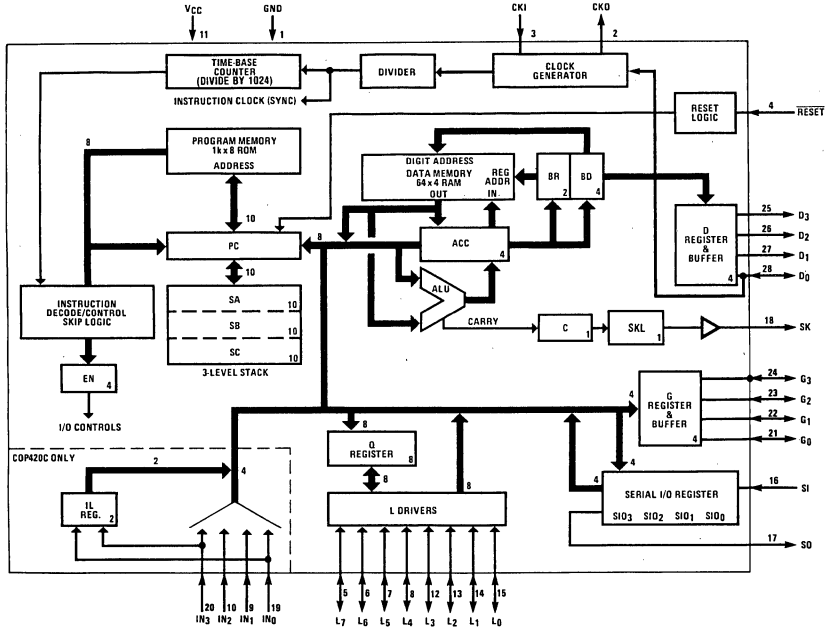
The COP420C, COP421C, COP320C, and COP321C Single-Chip CMOS Microcontrollers are members of the COPS™ family, fabricated using complementary MOS technology. They are complete microcomputers containing all system timing, internal logic, ROM, RAM and I/O necessary to implement dedicated control functions in a variety of applications. Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture and I/O scheme designed to facilitate keyboard input, display output and BCD and binary data manipulation. The COP421C is identical to the COP420C, except with 19 I/O lines instead of 23. They are an appropriate choice for use in numerous human interface control environments. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized Control Oriented Processor at a low end-product cost.

The COP320C is the extended temperature range version of the COP420C (likewise the COP321C is the extended temperature range version of the COP421C). The COP320C/321C are exact functional equivalents of the COP420C/421C.

## Features

- Lowest power dissipation (50 μW typical)
- Power saving "Idle" state
- Powerful instruction set
- 1k x 8 ROM, 64 x 4 RAM, 23 I/O lines (COP420C)
- True vectored interrupt, plus restart
- Three-level subroutine stack
- 15 μs instruction time, plus software selectable oscillators
- Single supply operation (2.4-6.0V)
- Internal time-base counter for real-time processing
- MICROWIRE™ compatible serial I/O
- General purpose and TRI-STATE® outputs
- TTL/CMOS compatible
- LED direct drive outputs
- MICROBUS™ compatible
- Software/hardware compatible with other members of COP400 family
- Extended temperature range device COP320C/COP321C (−40°C to +85°C)

## COP420C/421C Block Diagram



## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | −0.3V to $V_{CC}$ +0.3V |
| Operating Temperature Range | |
| COP420C/421C | 0°C to +70°C |
| COP320C/321C | −40°C to +85°C |
| Storage Temperature Range | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |
| Package Power Dissipation | 500mW |

## DC Electrical Characteristics

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Operation Voltage | | 2.4 | 6.0 | Volts |
| Supply Current (Note 1) | $V_{CC}$ = 2.4V $F_{IN}$ = 32kHz | | 35 | μA |
| | $V_{CC}$ = 5.0V $F_{IN}$ = 32kHz | | 100 | μA |
| | $V_{CC}$ = 5.0V $F_{IN}$ = 500kHz | | 800 | μA |
| | $V_{CC}$ = 5.0V $F_{IN}$ = 2.097MHz (÷32mode) | | 1200 | μA |
| Idle State Current | $V_{CC}$ = 2.4V $F_{IN}$ = 32kHz | | 15 | μA |
| | $V_{CC}$ = 5.0V $F_{IN}$ = 500kHz | | 250 | μA |
| Input Voltage Levels Schmitt Trigger Inputs RESET, CKI (as R/C) and DO (as Clock) | | | | |
| Logic High | | $0.9V_{CC}$ | | V |
| Logic Low | | | $0.1V_{CC}$ | V |
| All Other Inputs | | | | |
| Logic High | | $0.6V_{CC}$ | | V |
| Logic Low | | | $0.25V_{CC}$ | V |
| Output Voltage Levels | | | | |
| Standard Output TTL Operation | $V_{CC}$ = 5V ± 5% | | | |
| Logic High | $I_{OH}$ = −100μA | 2.4 | | V |
| Logic Low | $I_{OL}$ = 1.6mA | | 0.4 | V |
| CMOS Operation | | | | |
| Logic High | $I_{OH}$ = −10μA | $V_{CC}$ − 0.2 | | V |
| Logic Low | $I_{OL}$ = 10μA | | 0.2 | V |
| Output Current Levels | | | | |
| L Outputs with High Current Option | | | | |
| Logic High | $V_{CC}$ = 6V, $V_{OH}$ = 2.0V | −2.5 | −15 | mA |
| Logic Low | $V_{OL}$ = 0.4V | 1.6 | | mA |
| Hi-Z or Tri-State Input Current Levels | | −1.0 | 1.0 | μA |

## AC Electrical Characteristics $2.4V \leqslant V_{CC} \leqslant 6.0V$ unless otherwise stated

| Parameter | Condition | Min | Max | Units |
|---|---|---|---|---|
| Instruction Cycle Time | $V_{CC} \geqslant 4.5V$ | 15 | 245 | |
| COP420C/COP421C | $V_{CC} \geqslant 2.4V$ | 50 | 245 | μs |
| Operating CKI Frequency | ÷8 mode | 32 | 500 | kHz |
| COP420C/COP421C | ÷16 mode $V_{CC} \geqslant 4.5V$ | 64 | 1000 | kHz |
| | ÷32 mode | 128 | 2097 | kHz |
| | ÷8 mode | 32 | 160 | kHz |
| | ÷16 mode $V_{CC} \geqslant 2.4V$ | 64 | 320 | kHz |
| | ÷32 mode | 128 | 640 | kHz |
| Instruction Cycle Time | $V_{CC} \geqslant 4.5V$ | 15 | 125 | |
| COP320C/COP321C | $V_{CC} \geqslant 2.4V$ | 50 | 125 | μs |
| Operating CKI Frequency | ÷8 mode | 64 | 500 | kHz |
| COP320C/COP321C | ÷16 mode $V_{CC} \geqslant 4.5V$ | 128 | 1000 | kHz |
| | ÷32 mode | 256 | 2097 | kHz |
| | ÷8 mode | 64 | 160 | kHz |
| | ÷16 mode $V_{CC} \geqslant 2.4V$ | 128 | 320 | kHz |
| | ÷32 mode | 256 | 640 | kHz |
| Instruction Cycle Time | $R = 30k \pm 5\%$, $C = 100pF \pm 10\%$ | 15 | 25 | μs |
| D0 as Clock or CKI (R/C) | $V_{CC} = 5V$ | | | |
| CKO as SYNC Input | | | | |
| $t_{SYNC}$ | | 400 | | ns |
| INPUTS: | | | | |
| $t_{SETUP}$ | | 2 | | μs |
| $t_{HOLD}$ | | 0.6 | | μs |
| OUTPUTS: | $V_{OH} = 0.7V_{CC}$, $V_{OL} = 0.2V_{CC}$ | | | |
| COP to CMOS | $C_L = 50pF$ | | | |
| $t_{PD1}$ | | 4 | | μs |
| $t_{PD0}$ | | 2 | | μs |
| COP to TTL | Fanout = 1 Standard TTL Load. | | | |
| | $C_L = 50pF$, $V_{CC} = 5V \pm 5\%$ | | | |
| $t_{PD1}$ | $V_{OH} = 2.4V$ | 3 | | μs |
| $t_{PD0}$ | $V_{OL} = 0.4V$ | 3 | | μs |
| CKO Output | | | | |
| $t_{PD1}$ | | 0.4 | | μs |
| $t_{PD0}$ | | 0.4 | | μs |
| MICROBUS™ TIMING | $C_L = 50pF$, $V_{CC} = 5V \pm 5\%$ | | | |
| Read Operation (figure 4) | | | | |
| Chip Select Stable Before RD – $t_{CSR}$ | | 50 | | ns |
| Chip Select Hold Time for RD – $t_{RCS}$ | | 5 | | ns |
| RD Pulse Width – $t_{RR}$ | | 400 | | ns |
| Data Delay from RD – $t_{RD}$ | | | 300 | ns |
| RD to Data Floating – $t_{DI}$ | | | 200 | ns |
| Write Operation (figure 5) | | | | |
| Chip Select Stable Before WR – $t_{CSW}$ | | 50 | | ns |
| Chip Select Hold Time for WR – $t_{WCS}$ | | 30 | | ns |
| WR Pulse Width – $t_{WW}$ | | 350 | | ns |
| Data Set-Up Time for WR – $t_{DW}$ | | 300 | | ns |
| Data Hold Time for WR – $t_{WD}$ | | 40 | | ns |
| INTR Transition Time from WR – $t_{WI}$ | | | 700 | ns |

**Note 1** — Supply current is measured with a squarewave clock, all inputs at $V_{CC}$, and all outputs open while the COP420C is running.

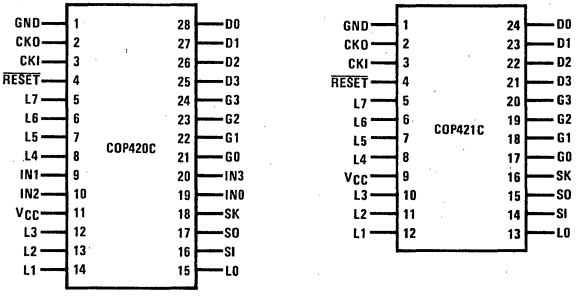



**Figure 2. Connection Diagrams**

## Pin Description

| | |
|---|---|
| $L_7$-$L_0$ | 8 bidirectional I/O ports with TRI-STATE® |
| $G_3$-$G_0$ | 4 bidirectional I/O ports |
| $D_3$-$D_1$ | 3 general purpose outputs |
| $D_0$ | General purpose output or oscillator input |
| $IN_3$-$IN_0$ | 4 general purpose inputs (COP420C only) |
| SI | Serial input |
| SO | Serial output |
| SK | Logic-controlled clock |
| CKI | System oscillator input |
| CKO | System oscillator output (or general purpose input) |
| $\overline{RESET}$ | System reset input |
| $V_{CC}$ | Power supply |
| GND | Ground |

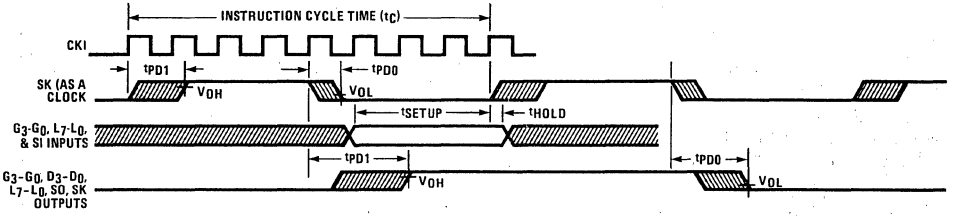



**Figure 3. Input/Output Timing Diagrams (divide by 8 mode)**

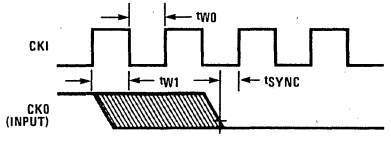



**Figure 3A. Synchronization Timing**

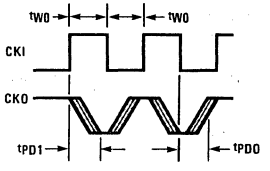



**Figure 3B. CKO Output Timing**

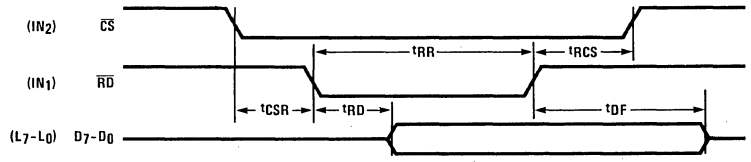



Figure 4. MICROBUS™ Read Operation Timing

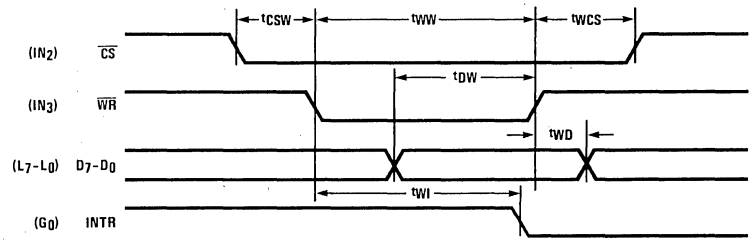



Figure 5. MICROBUS™ Write Operation Timing

## FUNCTIONAL DESCRIPTION

For ease of reading this description, only COP420C and/or COP421C are referenced; however, all such references apply equally to COP320C and/or COP321C, respectively.

A block diagram of the COP420C is given in Figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1". When a bit is reset, it is a logic "0"

### Program Memory

Program Memory consists of a 1,024-byte ROM. As can be seen by an examination of the COP420C/421C instruction set, these words may be program instructions, program data or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID and LQID instructions, ROM must often be thought of as being organized into 16 pages of 64 words each.

ROM addressing is accomplished by a 10-bit **PC register**. Its binary value selects one of the 1,024 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential **10-bit binary count** value. Three levels of subroutine nesting are implemented by the 10-bit binary subroutine save registers, SA, SB and SC, providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the Instruction Decode, Control and Skip Logic circuitry.

### Data Memory

Data Memory consists of a 256-bit RAM, organized as 4 data registers of 16 4-bit digits. RAM addressing is implemented by a 6-bit **B-register** whose upper 2 bits (Br) select 1 of 4 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) is usually loaded into or from, or exchanged with, the A register (accumulator), it may also be loaded into or from the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the 6-bit contents of the operand field of these instructions. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

### Internal Logic

The 4-bit **A register** (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd portions of the B register, to load and input 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

A **4-bit adder** performs the arithmetic and logic functions of the COP420C/421C, storing its results in A. It also outputs a carry bit to the 1-bit **C register**, most often employed to indicate arithmetic overflow. The C register, in conjuction with the XAS instruction and the EN register, also serves to control the SK output, C can be outputted directly to SK or can enable the SK to be a sync clock each instruction cycle time. (See XAS instruction and EN register description, below.)

Four **general-purpose inputs, $IN_3$-$IN_0$**, are provided; $IN_1$, $IN_2$ and $IN_3$ may be selected, by a mask-programmable option, as Read Strobe, Chip Select and Write Strobe inputs, respectively, for use in MICROBUS™ applications.

The **D register** provides 4 general purpose outputs and is used as the destination register for the 4-bit contents of Bd. In the dual clock mode, D-register bit 0 controls the clock selection (see dual oscillator below).

The **G register** contents are outputs to 4 general-purpose bidirectional I/O ports. $G_0$ may be mask-programmed as an output for MICROBUS™ applications.

The **Q register** is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control (see LEI instruction). With the MICROBUS™ option selected, Q can also be loaded with the 8-bit contents of the L I/O ports upon the occurence of a write strobe from the host CPU.

The **8 L drivers**, when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. As explained above, the MICROBUS™ option allows L I/O port data to be latched into the Q register. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the LED Direct Drive output configuration option) with Q data being outputted to the Sa-Sg and decimal point segments of the display.

The **SIO register** functions as a 4-bit serial-in/serial-out serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time (see 4 below). The SK output becomes a logic-controlled clock. The SIO contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O by connecting SO to external serial-in/parallel-out shift registers.

The XAS instruction copies C into the **SKL Latch**. SK outputs SKL ANDed with the internal instruction cycle clock.

The **EN Register** is an internal 4-bit register loaded under program control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register ($EN_3$-$EN_0$).

1. The least significant bit of the enable register, $EN_0$, must be set at 0.
2. With $EN_1$ set the $IN_1$ input is enabled as an interrupt input. Immediately following an interrupt, $EN_1$ is reset to disable further interrupts.
3. With $EN_2$ set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting $EN_2$ disables the L drivers, placing the L I/O ports in a high-impedance input state. If the MICROBUS™ option is being used, $EN_2$ does not affect the L drivers.
4. $EN_3$ affects the SO output. Setting $EN_3$ enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting $EN_3$ disables SO as the shift register output: data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0."

## COP420C/421C INSTRUCTION SET

Table 1 is a symbol table providing internal architecture, instruction operand and operational symbols used in the instruction set table.

Table 2 provides the mnemonic, operand, machine code, data flow, skip conditions and description associated with each instruction in the COP420C/421C instruction set.

**Table 1. COP420C/421C Instruction Set Table Symbols**

| Symbol | Definition |
|---|---|
| **INTERNAL ARCHITECTURE SYMBOLS** | |
| A | 4-bit Accumulator |
| B | 6-bit RAM Address Register |
| Br | Upper 2 bits of B (register address) |
| Bd | Lower 4 bits of B (digit address) |
| C | 1-bit Carry Register |
| D | 4-bit Data Output Port |
| EN | 4-bit Enable Register |
| G | 4-bit Register to latch data for G I/O Port |
| IL | Two 1-bit Latches associated with the $IN_3$ or $IN_0$ Inputs |
| IN | 4-bit Input Port |
| L | 8-bit TRI-STATE I/O Port |
| M | 4-bit contents of RAM Memory pointed to by B Register |
| PC | 10-bit ROM Address Register (program counter) |
| Q | 8-bit Register to latch data for L I/O Port |
| SA | 10-bit Subroutine Save Register A |
| SB | 10-bit Subroutine Save Register B |
| SC | 10-bit Subroutine Save Register C |
| SIO | 4-bit Shift Register |
| SK | Logic-Controlled Clock Output |
| **INSTRUCTION OPERAND SYMBOLS** | |
| d | 4-bit Operand Field, 0-15 binary (RAM Digit Select) |
| r | 2-bit Operand Field, 0-3 binary (RAM Register Select) |
| a | 10-bit Operand Field, 0-1023 binary (ROM Address) |
| y | 4-bit Operand Field, 0-15 binary (Immediate Data) |
| RAM(s) | Contents of RAM location addressed by s |
| ROM(t) | Contents of ROM location addressed by t |
| **OPERATIONAL SYMBOLS** | |
| + | Plus |
| − | Minus |
| → | Replaces |
| ↔ | Is exchanged with |
| = | Is equal to |
| $\overline{A}$ | The ones complement of A |
| ⊕ | Exclusive-OR |
| : | Range of values |

**Table 2. COP420C/421C Instruction Set Table** (Note 1)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| ARITHMETIC INSTRUCTIONS | | | | | | |
| ASC | | 30 | 0011\|0000 | A + C + RAM(B) → A<br>Carry → C | Carry | Add with Carry, Skip on Carry |
| ADD | | 31 | 0011\|0001 | A + RAM(B) → A | None | Add RAM to A |
| ADT | | 4A | 0100\|1010 | A + $10_{10}$ → A | None | Add Ten to A |
| AISC | y | 5- | 0101\| y | A + y → A | Carry | Add Immediate, Skip on Carry (y ≠ 0) |
| CASC | | 10 | 0001\|0000 | $\overline{A}$ + RAM(B) + C → A<br>Carry → C | Carry | Complement and Add with Carry, Skip on Carry |
| CLRA | | 00 | 0000\|0000 | 0 → A | None | Clear A |
| COMP | | 40 | 0100\|0000 | $\overline{A}$ → A | None | Ones complement of A to A |
| NOP | | 44 | 0100\|0100 | None | None | No Operation |
| RC | | 32 | 0011\|0010 | "0" → C | None | Reset C |
| SC | | 22 | 0010\|0010 | "1" → C | None | Set C |
| XOR | | 02 | 0000\|0010 | A ⊕ RAM(B) → A | None | Exclusive-OR A with RAM |

Table 2. COP420C/421C Instruction Set Table (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **TRANSFER OF CONTROL INSTRUCTIONS** | | | | | | |
| JID | | FF | 1111 1111 | ROM ($PC_{9:8}$,A,M) → $PC_{7:0}$ | None | Jump Indirect (Note 3) |
| JMP | a | 6- | 0110 00 $a_{9:8}$ | a → PC | None | Jump |
| | | -- | $a_{7:0}$ | | | |
| JP | a | -- | 1 $a_{6:0}$ (pages 2,3 only) | a → $PC_{6:0}$ | None | Jump within Page (Note 4) |
| | | | or | | | |
| | | -- | 11 $a_{5:0}$ (all other pages) | a → $PC_{5:0}$ | | |
| JSRP | a | -- | 10 $a_{5:0}$ | PC + 1 → SA → SB → SC<br>0010 → $PC_{9:6}$<br>a → $PC_{5:0}$ | None | Jump to Subroutine Page (Note 5) |
| JSR | a | 6- | 0110 10 $a_{9:8}$ | PC + 1 → SA → SB → a → PC | None | Jump to Subroutine |
| | | -- | $a_{7:0}$ | | | |
| RET | | 48 | 0100 1000 | SC → SB → SA → PC | None | Return from Subroutine |
| RETSK | | 49 | 0100 1001 | SC → SB → SA → PC | Always Skip on Return | Return from Subroutine then Skip |
| IT | | 33 | 0011 0011 | PC → PC | | Idle till Timer overflows then continue |
| | | 39 | 0011 1001 | | | |
| **MEMORY REFERENCE INSTRUCTIONS** | | | | | | |
| CAMQ | | 33 | 0011 0011 | A → $Q_{7:4}$<br>RAM(B) → $Q_{3:0}$ | None | Copy A, RAM to Q |
| | | 3C | 0011 1100 | | | |
| CQMA | | 33 | 0011 0011 | $Q_{7:4}$ → RAM(B)<br>$Q_{3:0}$ → A | None | Copy Q to RAM, A |
| | | 2C | 0010 1100 | | | |
| LD | r | -5 | 00 r 0101 | RAM(B) → A<br>Br ⊕ r → Br | None | Load RAM into A, Exclusive-OR Br with r |
| LDD | r,d | 23 | 0010 0011 | RAM(r,d) → A | None | Load A with RAM pointed to directly by r,d |
| | | -- | 00 r d | | | |
| LQID | | BF | 1011 1111 | ROM($PC_{9:8}$,A,M) → Q<br>SB → SC | None | Load Q Indirect (Note 3) |
| RMB | 0 | 4C | 0100 1100 | 0 → $RAM(B)_0$ | None | Reset RAM Bit |
| | 1 | 45 | 0100 0101 | 0 → $RAM(B)_1$ | | |
| | 2 | 42 | 0100 0010 | 0 → $RAM(B)_2$ | | |
| | 3 | 43 | 0100 0011 | 0 → $RAM(B)_3$ | | |
| SMB | 0 | 4D | 0100 1101 | 1 → $RAM(B)_0$ | None | Set RAM Bit |
| | 1 | 47 | 0100 0111 | 1 → $RAM(B)_1$ | | |
| | 2 | 46 | 0100 0110 | 1 → $RAM(B)_2$ | | |
| | 3 | 4B | 0100 1011 | 1 → $RAM(B)_3$ | | |

**Table 2. COP420C/421C Instruction Set (continued)**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| MEMORY REFERENCE INSTRUCTIONS (continued) | | | | | | |
| STII | y | 7- | 0 1 1 1 \| y | y → RAM(B)<br>Bd + 1 → Bd | None | Store Memory Immediate and Increment Bd |
| X | r | -6 | 0 0 \| r \| 0 1 1 0 | RAM(B) ⟷ A<br>Br ⊕ r → Br | None | Exchange RAM with A, Exclusive-OR Br with r |
| XAD | r,d | 23 | 0 0 1 0 \| 0 0 1 1 | RAM(r,d) ⟷ A | None | Exchange A with RAM pointed to directly by r,d |
| | | -- | 1 0 \| r \| d | | | |
| XDS | r | -7 | 0 0 \| r \| 0 1 1 1 | RAM(B) ⟷ A<br>Bd − 1 → Bd<br>Br ⊕ r → Br | Bd decrements past 0 | Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r |
| XIS | r | -4 | 0 0 \| r \| 0 1 0 0 | RAM(B) ⟷ A<br>Bd + 1 → Bd<br>Br ⊕ r → Br | Bd increments past 15 | Exchange RAM with A and Increment Bd, Exclusive-OR Br with r |
| REGISTER REFERENCE INSTRUCTIONS | | | | | | |
| CAB | | 50 | 0 1 0 1 \| 0 0 0 0 | A → Bd | None | Copy A to Bd |
| CBA | | 4E | 0 1 0 0 \| 1 1 1 0 | Bd → A | None | Copy Bd to A |
| LBI | r,d | -- | 0 0 \| r \| (d − 1)<br>(d = 0, 9:15) | r,d → B | Skip until not a LBI | Load B Immediate with r,d (Note 6) |
| | | | or | | | |
| | | 33 | 0 0 1 1 \| 0 0 1 1 | | | |
| | | -- | 1 0 \| r \| d<br>(any d) | | | |
| LEI | y | 33 | 0 0 1 1 \| 0 0 1 1 | y → EN | None | Load EN Immediate (Note 7) |
| | | 6- | 0 1 1 0 \| y | | | |
| XABR | | 12 | 0 0 0 1 \| 0 0 1 0 | A ⟷ Br (0,0 → $A_3$,$A_2$) | None | Exchange A with Br |
| TEST INSTRUCTIONS | | | | | | |
| SKC | | 20 | 0 0 1 0 \| 0 0 0 0 | | C = "1" | Skip if C is True |
| SKE | | 21 | 0 0 1 0 \| 0 0 0 1 | | A = RAM(B) | Skip if A Equals RAM |
| SKGZ | | 33 | 0 0 1 1 \| 0 0 1 1 | | $G_{3:0}$ = 0 | Skip if G is Zero (all 4 bits) |
| | | 21 | 0 0 1 0 \| 0 0 0 1 | | | |
| SKGBZ | | 33 | 0 0 1 1 \| 0 0 1 1 | 1st byte | | Skip if G Bit is Zero |
| | 0 | 01 | 0 0 0 0 \| 0 0 0 1 | 2nd byte | $G_0$ = 0 | |
| | 1 | 11 | 0 0 0 1 \| 0 0 0 1 | | $G_1$ = 0 | |
| | 2 | 03 | 0 0 0 0 \| 0 0 1 1 | | $G_2$ = 0 | |
| | 3 | 13 | 0 0 0 1 \| 0 0 1 1 | | $G_3$ = 0 | |
| SKMBZ | 0 | 01 | 0 0 0 0 \| 0 0 0 1 | | $RAM(B)_0$ = 0 | Skip if RAM Bit is Zero |
| | 1 | 11 | 0 0 0 1 \| 0 0 0 1 | | $RAM(B)_1$ = 0 | |
| | 2 | 03 | 0 0 0 0 \| 0 0 1 1 | | $RAM(B)_2$ = 0 | |
| | 3 | 13 | 0 0 0 1 \| 0 0 1 1 | | $RAM(B)_3$ = 0 | |
| SKT | | 41 | 0 1 0 0 \| 0 0 0 1 | | A time-base counter overflow has occurred since last test | Skip on Timer (Note 3) |

**Table 2. COP420C/421C Instruction Set (continued)**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| INPUT/OUTPUT INSTRUCTIONS | | | | | | |
| ING | | 33 | 0 0 1 1 \| 0 0 1 1 | G → A | None | Input G Ports to A |
| | | 2A | 0 0 1 0 \| 1 0 1 0 | | | |
| ININ | | 33 | 0 0 1 1 \| 0 0 1 1 | IN → A | None | Input IN Inputs to A (Note 2) |
| | | 28 | 0 0 1 0 \| 1 0 0 0 | | | |
| INIL | | 33 | 0 0 1 1 \| 0 0 1 1 | $IL_3$,"1","0",$IL_0$ → A | None | Input IL Latches to A (Note 3) |
| | | 29 | 0 0 1 0 \| 1 0 0 1 | | | |
| INL | | 33 | 0 0 1 1 \| 0 0 1 1 | $L_{7:4}$ → RAM(B)<br>$L_{3:0}$ → A | None | Input L Ports to RAM,A |
| | | 2E | 0 0 1 0 \| 1 1 1 0 | | | |
| OBD | | 33 | 0 0 1 1 \| 0 0 1 1 | Bd → D | None | Output Bd to D Outputs |
| | | 3E | 0 0 1 1 \| 1 1 1 0 | | | |
| OGI | y | 33 | 0 0 1 1 \| 0 0 1 1 | y → G | None | Output to G Ports Immediate |
| | | 5– | 0 1 0 1 \| y | | | |
| OMG | | 33 | 0 0 1 1 \| 0 0 1 1 | RAM(B) → G | None | Output RAM to G Ports |
| | | 3A | 0 0 1 1 \| 1 0 1 0 | | | |
| XAS | | 4F | 0 1 0 0 \| 1 1 1 1 | A ⟷ SIO, C → SKL | None | Exchange A with SIO (Note 3) |

**Note 1:** All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, $A_3$ indicates the most significant (left-most) bit of the 4-bit A register.

**Note 2:** The ININ instruction is not available on the 24-pin COP421C since this device does not contain the IN inputs.

**Note 3:** For additional information on the operation of the XAS, JID, LQID, INIL, IT and SKT instructions, see below.

**Note 4:** The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

**Note 5:** A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

**Note 6:** LBI is a single-byte instruction if d = 0, 9, 10, 11, 12, 13, 14, or 15. The machine code for the lower 4 bits equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 ($1001_2$), the lower 4 bits of the LBI instruction equal 8 ($1000_2$). To load 0, the lower 4 bits of the LBI instruction should equal 15 ($1111_2$).

**Note 7:** Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register.)

### Interrupt

The following features are associated with the $IN_1$ interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

a. The interupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1 → SA → SB → SC). Any previous contents of SC are lost. The program counter is set to hex address OFF (the last word of page 3) and $EN_1$ is reset.

b. An interrupt will be acknowledged only after the following conditions are met:
   1) $EN_1$ has been set.
   2) A low-going pulse ("1" to "0") of at least two instruction cycles wide occurs on the $IN_1$ input.
   3) A currently executing instruction has been completed.
   4) All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction the interrupt will not be acknowledged until the second JP instruction has been executed.

c. Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon the popping of the stack. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skip logic status is saved and program control is transferred to the interrupt servicing routine at hex address OFF. At the *end* of the interrupt routine, a RET instruction is executed to "pop" the stack and return program control to the instruction following the original ASC. *At this time*, the skip logic is enabled and skips this instruction because of the previous ASC carry. Subroutines and the LQID instruction should not be nested within the interrupt servicing routine since their popping of the stack enables any previously saved main program skips, interfering with the orderly execution of the interrupt routine.

d. The first instruction of the interrupt routine at hex address 0FF must be a NOP.

e. A LEI instruction can be put immediately before the RET to re-enable interrupts.

### MICROBUS™ Interface

The COP420C has an option which allows it to be used as a peripheral microprocessor device, inputting and outputting data from and to a host microprocessor (μP). $IN_1$, $IN_2$, and $IN_3$ general purpose inputs become **MICROBUS™ compatible** read-strobe, chip-select, and write-strobe lines, respectively. $IN_1$ becomes $\overline{RD}$ — a logic "0" on this input will cause Q latch data to be enabled to the L ports for input to the μP. $IN_2$ becomes $\overline{CS}$ — a logic 0 selects the COP420C as a μP peripheral device and allows for the selection of one of several peripheral components. $IN_3$ becomes $\overline{WR}$ — a logic "0" on this line will write bus data from the L ports to the Q latches for input to the COP420C. $G_0$ becomes INTR a "ready" output, reset by a write pulse from the μP on the $\overline{WR}$ line, providing the "handshaking" capability necessary for asynchronous data transfer between the host CPU and the COP420C.

This option has been designed for compatibility with National's MICROBUS™ — a standard interconnect system for 8-bit parallel data transfer between MOS/LSI CPUs and interfacing devices. (See MICROBUS™, National Publication.) The functioning and timing relationships between the COP420C signal lines affected by this option are as specified for the MICROBUS™ interface, and are given in the AC electrical characteristics and shown in the timing diagrams (Figures 4 and 5). Connection of the COP420C to the MICROBUS™ is shown in Figure 6.

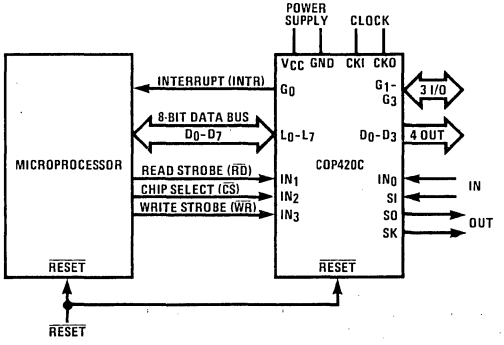



**Figure 6. MICROBUS™ Option Interconnect**

### Initialization

The Reset Logic, internal to the COP420C/421C, will initialize (clear) the device upon power-up if the power supply rise time is less than 1ms and greater than 1μs. If the power supply rise time is greater than 1ms, the user must provide an external RC network and diode to the $\overline{RESET}$ pin as shown below. The $\overline{RESET}$ pin is configured as a Schmitt trigger input. If not used it should be connected to $V_{CC}$. Initialization will occur whenever a logic "0" is applied to the $\overline{RESET}$ input, provided it stays low for at least two instruction cycle times.

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, and G registers are cleared. The SK output is enabled as a SYNC clock providing a pulse each instruction cycle time. *Data Memory (RAM) must be cleared by the user's program.* The first instruction at address 0 must be a CLRA.

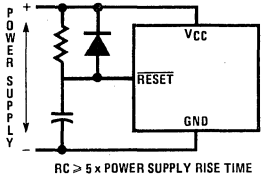



**Figure 7. Power-Up Clear Circuit**

The following information is provided to assist the user in understanding the operation of several unique instructions and to provide notes useful to programmers in writing COP420C/421C programs.

### XAS Instruction

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register. The contents of SIO will contain serial-in/serial-out shift register data. An XAS instruction will also affect the SK output, providing a logic controlled clock. An XAS instruction must be performed once every 4 instruction cycles to effect a continuous data stream.

### JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM addressed by the 10-bit word, $PC_{9:8}$, A, M. $PC_9$ and $PC_8$ are not affected by this instruction.

Note that JID requires 2 instruction cycles.

### INIL Instruction

INIL (Input IL Latches to A) inputs 2 latches, $IL_3$ and $IL_0$ (see figure 10) and CKO into A. The $IL_3$ and $IL_0$ latches are set if a low-going pulse ("1" to "0") has occurred on the $IN_3$ and $IN_0$ inputs since the last INIL instruction, provided the input pulse stays low for at least two instruction times. Execution of an INIL inputs $IL_3$ and $IL_0$ into $A_3$ and $A_0$ respectively, and resets these latches to allow them to respond to subsequent low-going pulses on the $IN_3$ and $IN_0$ lines. If CKO is mask programmed as a general purpose input, an INIL will input the state of CKO into $A_2$. If CKO has not been so programmed, a "1" will be placed in $A_2$. A "0" is always placed in $A_1$ upon the execution of an INIL. The general purpose inputs $IN_3$-$IN_0$ are input to A upon execution of an ININ instruction. (See table 2, ININ instruction.) INIL is useful in recognizing pulses of short duration or pulses which occur too often to be read conveniently by an ININ instruction. Note that IL latches are not cleared on reset. IL latches are not available on the COP421C

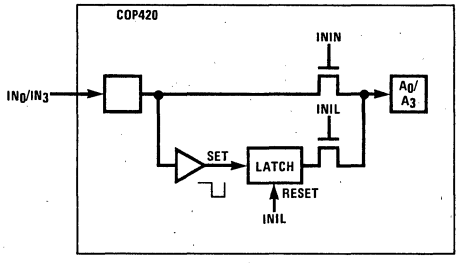



**Figure 8. INIL Hardware Implementation**

### LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 10-bit word $PC_9$, $PC_8$, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB →SC) and replaces the least significant 8 bits of PC as follows: A → $PC_{7:4}$, RAM(B) → $PC_{3:0}$, leaving $PC_9$ and $PC_8$ unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped" (SC → SB → SA →PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SB → SC, the previous contents of SC are lost. Also, when LQID pops the stack, the previously pushed contents of SB are left in SC. The net result is that the contents of SB are placed in SC (SB → SC). Note that LQID takes two instruction cycle times to execute.

### SKT Instruction

The SKT (Skip On Timer) instruction tests the state of an internal 10-bit time base counter. This counter divides the instruction cycle clock frequency by 1024 and provides a latched indication of counter overflow. The SKT instruction tests this overflow latch, executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction, therefore, allow the COP420C/421C to generate its own time base for real-time processing rather than relying on an external input signal.

For example, using a 32 kHz watch crystal for the oscillator, the counter pulse frequency will be 4 Hz. For time-of-day or similar real-time processing, the SKT instruction can call a routine which increments a "seconds" counter every 4 ticks.

#### IT Instruction

The user may choose to use the IT function instead of the SKT function. The IT (Idle till Timer) instruction halts the processor and puts it in an idle state until the time base counter overflows. This idle state reduces current drain since all logic (except the oscillator and time base counter) is stopped.

The time base counter always divides CKI by 8192 regardless of the divide-by option selected (see figures 10 and 11). Therefore, if using a 2.097 MHz crystal with CKI/32 option, the processor will come out of the idle state 256 times a second. If using the dual clock feature, the user *must* switch the processor to the CKI oscillator (D0 = 0) before executing the IT instruction.

Note: If using the dual clock feature or the IT instruction, contact the factory for emulation assistance.

### Oscillator

There are five basic clock oscillator configurations available as shown by figure 9.

a. **Crystal Controlled Oscillator.** CKI and CKO are connected to an external crystal (or resonator). The instruction cycle time equals the crystal frequency divided by 32, 16 or 8.

b. **External Oscillator.** CKI is connected to an external clock input signal. CKO is now available to be used as a general purpose input.

c. **RC Controlled Oscillator.** CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 8. CKO is now used as a general purpose input.

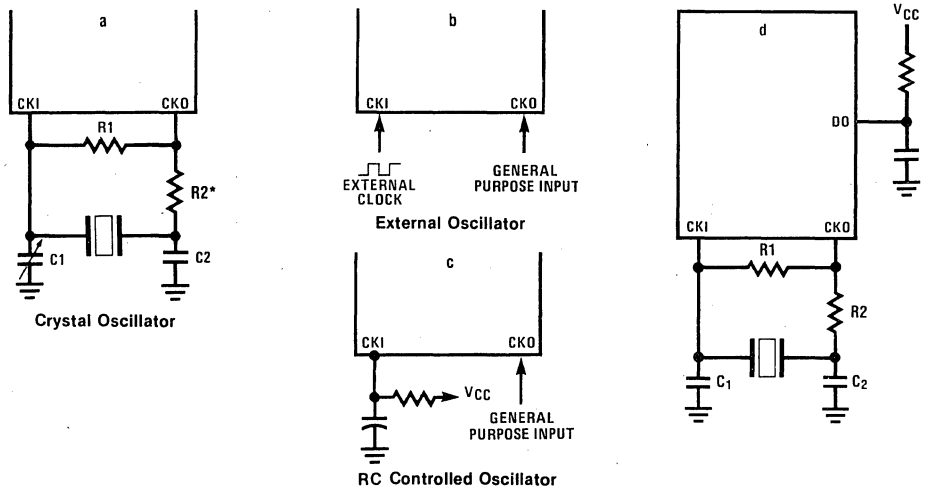
d. **Dual Oscillator.** By selecting the dual clock option, pin D0 is now a clock input. The user may connect a 32 kHz watch crystal to CKI and CKO and up to a 500 kHz RC circuit to D0; he may then software select between the RC oscillator for faster processing (D0 = 1) or the crystal for minimum current drain (D0 = 0). The time base counter continues even when the user selects D0 as the clock. Thus, a real time clock can be maintained by the IT instruction even when running off the RC oscillator. The SKT instruction is not available when using the dual clock feature.

Some features and options are exclusive of each other. The chart below shows which features may be used coincidentally.

| Xtal/ Ext. Osc. Input | RC Osc. ÷8 | SKT Instr. | IT Instr. | Dual Clock-RC | Dual Clock-Ext. |
|---|---|---|---|---|---|
| X | | X | | | |
| | X | X | | | |
| X | | | X | | |
| | X | | X | | |
| X | | | X | X | |
| X | | | X | | X |

### CKO Pin Options

In a crystal controlled oscillator system, CKO is used as an output to the crystal network. As an option CKO can be a general purpose input, read into bit 2 of A (accumulator) upon execution of an INIL instruction.




Crystal Oscillator

| Crystal Value | Component Values | | | |
|---|---|---|---|---|
| | R1 | R2 | C1 | C2 |
| 2.097MHz | 20M | 1K* | 5-36pF | 30pF |
| 32kHz | 20M | 220K* | 5-36pF | 30pF |
| 500kHz | 20M | 4K* | 40pF | 80pF |

*Selected based on Crystal used.

Figure 9.

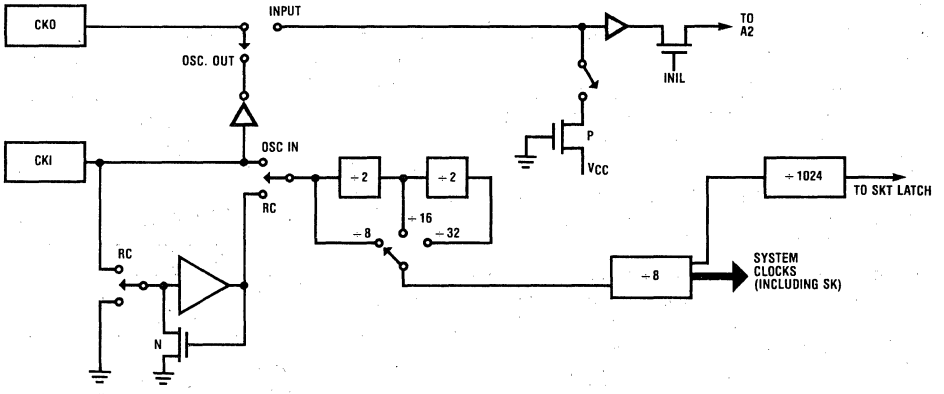



**Figure 10a. Oscillator Options Block Diagram Using SKT Instruction**

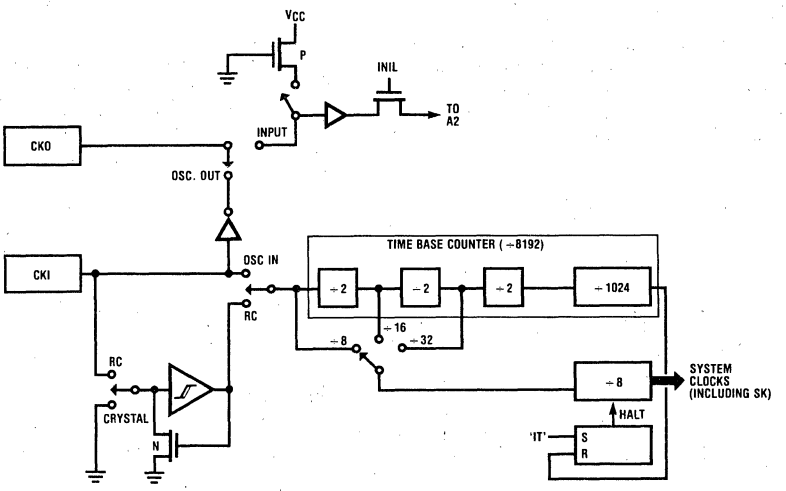



**Figure 10b. Oscillator Options Block Diagram Using IT Instruction**

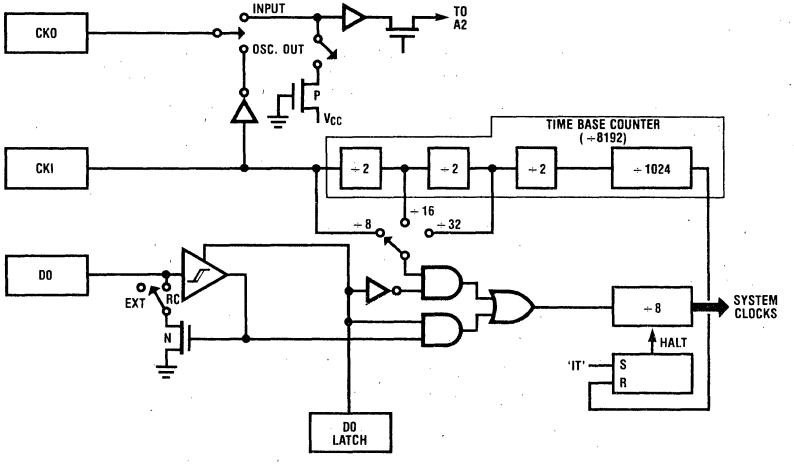



**Figure 11. Dual Clock Option Block Diagram**

### I/O Options

COP420C/421C outputs have the following optional configurations, illustrated in figure 13:

**a. Standard** — An N channel device to ground in conjunction with a P channel device to $V_{CC}$, compatible with CMOS and TTL.

**b. Open Drain** — An N channel device to ground only, allowing external pull-up as required by the user's application.

**c. TRI-STATE® L Output** — A CMOS output buffer which may be disabled by program control. These outputs meet the requirements associated with the MICROBUS™ option. These outputs are also capable of meeting the current sourcing requirements of the segments of a small LED display.

**d. Standard L Output** — This is the same configuration as c. above except that the sourcing current is standard.

**e. Open Drain L Output** — This has the N channel device to ground only.

COP420C/421C inputs have the following options:

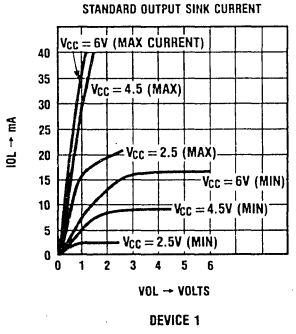
**f.** An on chip pullup load device to $V_{CC}$.

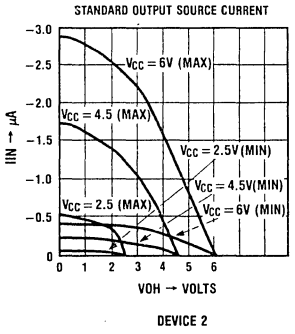
**g.** A HiZ input which must be driven by user logic.

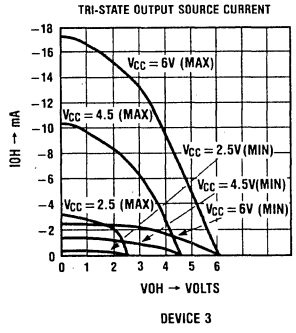
The above input and output configurations share common devices. Specifically, all configurations use one or more of four devices (numbered 1-4, respectively). Minimum and maximum current ($I_{OUT}$ and $V_{OUT}$) curves are given in figure 12 for each of these devices to allow the designer to effectively use these I/O configurations.

### COP421C

If the COP420C is bonded as a 24-pin device, it becomes the COP421C, illustrated in figure 2, COP420C/421C Connection Diagrams. Note that the COP421C does not contain the four general purpose IN inputs ($IN_3$-$IN_0$). Use of this option precludes, of course, use of the IN options, interrupt feature, and the MICROBUS™ option which uses $IN_1$-$IN_3$. All other options are available for the COP421C.



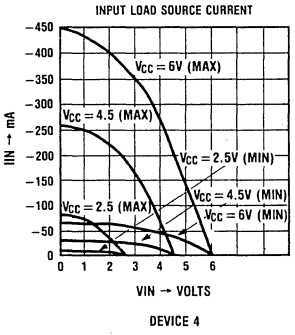




**Figure 12. I/O Characteristics**

## Instruction Set Notes

a. The first word of a program (ROM address 0) must be a CLRA (Clear A) instruction.

b. Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths take the same number of cycle times whether instructions are skipped or executed.

c. The ROM is organized into 16 pages of 64 words each. The Program Counter is an 10-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example: a JP located in the last word of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3, 7, 11, or 15 will access data in the next group of 4 pages.

## COP420C Power Dissipation

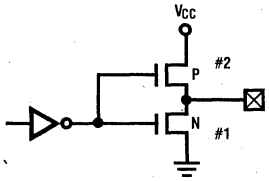
The lowest power configuration is at minimum voltage and lowest frequency. The user should take care that all inputs swing to full supply levels to insure that there are no DC current paths on inputs. An external square wave oscillator will use less current than a crystal or resonator since an input from a crystal is slow to transcend logic levels. For example: at 500 kHz, a crystal (or resonator) will typically cause the 420C to draw 100 $\mu$A more than with a square wave oscillator input. Power will increase with loading capacitance and frequency of the outputs.

The lowest possible current drain is when the processor is in the idle mode (see IT instruction).

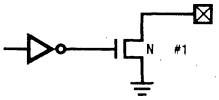
Another method to reduce power is to use the dual clock option. The overall current drain will be an average of the low frequency current and the high frequency current, based on the amount of time spent at each frequency.

## COP420C TTL Interface

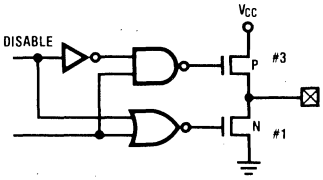
The COP420C outputs can directly drive one standard TTL load. A pull up device should be selected on inputs driven by TTL in order to bring the input signal up to the required logic "1" level.



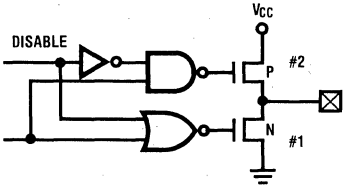

a. Standard



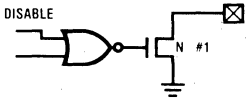

b. Open Drain



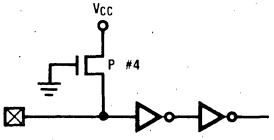

c. TRI STATE™ L Output



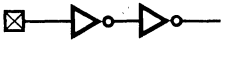

d. Standard L Output




e. Open Drain L Output




f. Input with Load



g. Hi Z Input

Figure 13. I/O Configurations

## OPTION LIST

The COP420C/432C mask-programmable options are assigned numbers which correspond with the COP420C pins.

The following is a list of COP420C options. When specifying a COP421C chip, Options 9, 10, 19, and 20 must all be set to zero. The options are programmed at the same time as the ROM pattern to provide the user with the hardware flexibility to interface to various I/O components using little or no external circuitry.

Option 1 = 0: Ground Pin
Not an option

Option 2: CKO Output
00 = Oscillator Output
02 = General Input, $V_{CC}$ Load
04 = General Input, Hi-Z

Option 3: CKI Input
00 = Oscillator IN (÷16)
01 = Oscillator IN (÷8)
02 = Oscillator IN (÷32)

Option 4: RESET Input
00 = Load $V_{CC}$
01 = Hi-Z

Option 5: $L_7$ Driver
00 = Standard Output
01 = Open Drain
02 = High Current TRI-STATE

Option 6: $L_6$ Driver
same as Option 5

Option 7: $L_5$ Driver
same as Option 5

Option 8: $L_4$ Driver
same as Option 5

Option 9: $IN_1$ Input
00 = Load $V_{CC}$
01 = Hi-Z

Option 10: $IN_2$ Input
same as Option 9

Option 11: $V_{CC}$ pin
not an option

Option 12: $L_3$ Driver
same as Option 5

Option 13: $L_2$ Driver
same as Option 5

Option 14: $L_1$ Driver
same as Option 5

Option 15: $L_0$ Driver
same as Option 5

Option 16: SI Input
same as Option 9

Option 17: SO Driver
00 = Standard Output
01 = Open Drain

Option 18: SK Driver
same as Option 17

Option 19: $IN_0$ Input
same as Option 9

Option 20: $IN_3$ Input
same as Option 9

Option 21: $G_0$ I/O Port
same as Option 17

Option 22: $G_1$ I/O Port
same as Option 17

Option 23: $G_2$ I/O Port
same as Option 17

Option 24: $G_3$ I/O Port
same as Option 17

Option 25: $D_3$ Output
same as Option 17

Option 26: $D_2$ Output
same as Option 17

Option 27: $D_1$ Output
same as Option 17

Option 28: $D_0$ Output
00 = Standard Output
01 = Open Drain (or Dual Clock)

Option 29: COP Function
00 = Normal
01 = MICROBUS

Option 30: COP Bonding
00 = COP420C (28-pin package)
01 = COP421C (24-pin package)
02 = COP420C and COP421C, same ROM (same die purchased in both 24 and 28 pin versions)

Option 31: Clock/Timer Mode
02 = Xtal/Ext. Osc, in; SKT instruction enabled; no IT
03 = RC Oscillator ÷ 8; SKT enabled; no IT
04* = Xtal/Ext. Osc. in; IT instruction enabled; no SKT
05* = RC Oscillator ÷ 8; IT enabled; no SKT
06* = Xtal/Ext. Osc. in; Dual Clock (RC); IT enabled no SKT
07* = Xtal/Ext. Osc. in; Dual Clock (Ext.) IT enabled; no SKT

*Contact factory for emulation assistance.

National Semiconductor

# COP420L/COP421L Single-Chip N-Channel Microcontrollers

## General Description

The COP420L and COP421L Single-Chip N-Channel Microcontrollers are members of the COPS™ family, fabricated using N-channel, silicon gate MOS technology. These controller oriented processors are complete microcomputers containing all system timing, internal logic, ROM, RAM and I/O necessary to implement dedicated control functions in a variety of applications. Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture and I/O scheme designed to facilitate keyboard input, display output and BCD data manipulation. The COP421L is identical to the COP420L, but with 19 I/O lines instead of 23. They are an appropriate choice for use in numerous human interface control environments. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized controller oriented processor at low end-product cost.

## Features

- Low cost
- Powerful instruction set
- 1k x 8 ROM, 64 x 4 RAM
- 23 I/O lines (COP420L)
- True vectored interrupt, plus restart
- Three-level subroutine stack
- 16 $\mu$s instruction time
- Single wide-range supply (4.5 - 9.5V)
- Low current drain (8mA max @ 5V)
- Internal time-base counter for real-time processing
- Internal binary counter register with serial I/O capability
- General purpose and TRI-STATE® outputs
- LSTTL/CMOS compatible in and out
- Direct drive of LED digit and segment lines
- Software/hardware compatible with other members of COP400 family
- Extended temperature range device available (-40°C to +85°C)

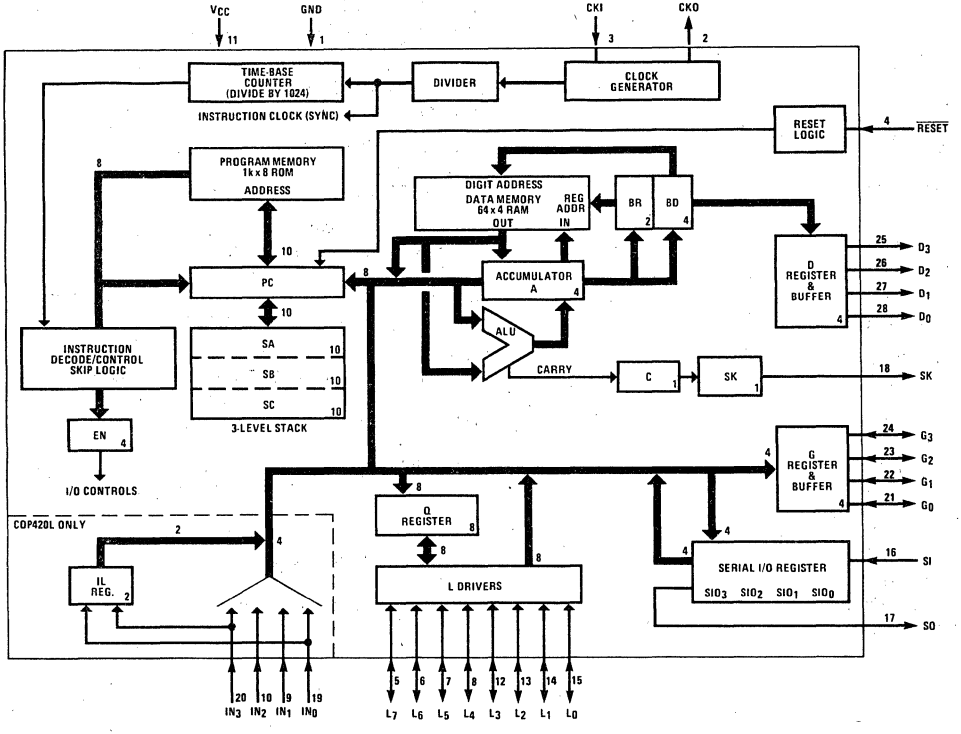



**Figure 1. COP420L/421L Block Diagram**

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Relative to GND | −0.5V to +10V |
| Ambient Operating Temperature | 0°C to +70°C |
| Ambient Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |
| Power Dissipation | 0.75 Watt at 25°C<br>0.4 Watt at 70°C |

*Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.*

## DC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Operating Voltage ($V_{CC}$) | | 4.5 | 9.5 | V |
| Operating Supply Current | $V_{CC} = 5V$, $T_A = 25°C$<br>(all inputs and outputs open) | | 8 | mA |
| Input Voltage Levels | | | | |
| CKI Input Levels | | | | |
| Crystal Input | | | | |
| Logic High ($V_{IH}$) | | 2.0 | | V |
| Logic Low ($V_{IL}$) | | | 0.4 | V |
| Schmitt Trigger Input | | | | |
| Logic High ($V_{CC}$) | | 0.7 $V_{CC}$ | | V |
| Logic Low ($V_{IL}$) | | | 0.6 | V |
| $\overline{\text{RESET}}$ Input Levels | | | | |
| Logic High | | 0.7 $V_{CC}$ | | V |
| Logic Low | | | 0.6 | V |
| $\overline{\text{RESET}}$ Hysteresis | | 1.0 | | V |
| SO Input Level (Test mode) | | 2.0 | 3.0 | V |
| All Other Inputs | | | | |
| Logic High | with TTL trip level options | 2.0 | | V |
| Logic Low | selected | | 0.8 | V |
| Logic High | with high trip level options | 3.6 | | V |
| Logic Low | selected | | 1.2 | V |
| Output Voltage Levels | | | | |
| LSTTL Operation | $V_{CC} = 5V \pm 5\%$ | | | |
| Logic High ($V_{OH}$) | $I_{OH} = -25\mu A$ | 2.7 | | V |
| Logic Low ($V_{OL}$) | $I_{OL} = 0.36mA$ | | 0.4 | V |

## DC Electrical Characteristics (continued) $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Output Current Levels | | | | |
| Output Sink Current | | | | |
| SO and SK Outputs ($I_{OL}$) | $V_{CC}=9.5V$, $V_{OL}=1.0V$ | 4.5 | 22 | mA |
| | $V_{CC}=4.5V$, $V_{OL}=1.0V$ | 2.2 | 11 | mA |
| $L_0$-$L_7$ Outputs and Standard Size $G_0$-$G_3$ and $D_0$-$D_3$ Outputs ($I_{OL}$) | $V_{CC}=9.5V$, $V_{OL}=1.0V$ | 2.0 | 9.0 | mA |
| | $V_{CC}=4.5V$, $V_{OL}=1.0V$ | 1.0 | 4.5 | mA |
| $G_0$-$G_3$ and $D_0$-$D_3$ Outputs with High Current Options ($I_{OL}$) | $V_{CC}=9.5V$, $V_{OL}=1.0V$ | 15 | 75 | mA |
| | $V_{CC}=4.5V$, $V_{OL}=1.0V$ | 7.0 | 35 | mA |
| $G_0$-$G_3$ and $D_0$-$D_3$ Outputs with Very High Current Options ($I_{OL}$) | $V_{CC}=9.5V$, $V_{OL}=1.0V$ | 30 | 150 | mA |
| | $V_{CC}=4.5V$, $V_{OL}=1.0V$ | 15 | 75 | mA |
| Output Source Current: | | | | |
| Standard Configuration, All Outputs ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −70 | −450 | μA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −26 | −190 | μA |
| Push-Pull Configuration, SO and SK Outputs ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −1.45 | −15.5 | mA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −0.07 | −2.8 | mA |
| LED Configuration, $L_0$-$L_7$ Outputs, Low Current Driver Option ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=2.0V$ | −1.5 | −15 | mA |
| | $V_{CC}=6.0V$, $V_{OH}=2.0V$ | −1.5 | −9.0 | mA |
| LED Configuration, $L_0$-$L_7$ Outputs, High Current Driver Option ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=2.0V$ | −3.0 | −30 | mA |
| | $V_{CC}=6.0V$, $V_{OH}=2.0V$ | −3.0 | −20 | mA |
| TRI-STATE Configuration, $L_0$-$L_7$ Outputs, Low Current Driver Option ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −2.4 | −24.5 | mA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −0.06 | −3.8 | mA |
| TRI-STATE Configuration, $L_0$-$L_7$ Outputs, High Current Driver Option ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −4.9 | −47.5 | mA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −0.12 | −8.1 | mA |
| CKO Output | | | | |
| RAM Power Supply Option Power Requirement | $V_R=3.3V$ | | 3.0 | mA |
| TRI-STATE® Output Leakage Current | | −10 | +10 | μA |

## AC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise specified.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Instruction Cycle Time — $t_C$ | | 15 | 40 | μs |
| CKI Using Crystal | | | | |
| Input Frequency — $f_I$ | ÷32 mode | 0.8 | 2.097 | MHz |
| | ÷16 mode | 0.4 | 1.0 | MHz |
| | ÷8 mode | 0.2 | 0.5 | MHz |
| Duty Cycle | | 30 | 55 | % |
| CKI Using External Clock | | | | |
| Input Frequency — $f_I$ | ÷32 mode | 0.8 | 2.0 | MHz |
| | ÷16 mode | 0.4 | 1.0 | MHz |
| | ÷8 mode | 0.2 | 0.5 | MHz |
| Duty Cycle | | 30 | 60 | % |
| Rise Time | | | 120 | ns |
| Fall Time | | | 80 | ns |
| CKI Using RC | $R = 51k\Omega \pm 5\%$<br>$C = 100pF \pm 10\%$ | | | |
| Instruction Cycle Time | | 15 | 25 | μs |
| CKO as SYNC Input | | | | |
| $t_{SYNC}$ | | 400 | | ns |
| INPUTS: | | | | |
| $IN_3$-$IN_0$, $G_3$-$G_0$, $L_7$-$L_0$ | | | | |
| $t_{SETUP}$ | | | 8 | μs |
| $t_{HOLD}$ | | | 600 | ns |
| SI | | | | |
| $t_{SETUP}$ | | | 2 | μs |
| $t_{HOLD}$ | | | 600 | ns |
| OUTPUTS: | | | | |
| COP TO CMOS PROPAGATION DELAY | $V_{OH} = 0.7\ V_{CC}$, $V_{OL} = 0.3\ V_{CC}$,<br>$C_L$ = 50pF | | | |
| All Standard Output Configurations | | | | |
| $t_{PD1}$ | | | 6.5 | μs |
| SO, SK Outputs | | | | |
| $t_{PD1}$ (push-pull) | | | 4.0 | μs |
| $t_{PD0}$ | | | 1.2 | μs |
| $D_3$-$D_0$, $G_3$-$G_0$ | | | | |
| $t_{PD0}$ (standard size) | | | 2.7 | μs |
| $t_{PD0}$ (high current) | | | 2.5 | μs |
| $t_{PD0}$ (very high current) | | | 2.4 | μs |
| $L_7$-$L_0$ | | | | |
| $t_{PD0}$ | | | 2.7 | μs |
| $t_{PD1}$ (standard size push-pull) | | | 3.0 | μs |
| $t_{PD1}$ (high current push-pull) | | | 2.5 | μs |
| $L_7$-$L_0$ LED Direct Drive Outputs | $6.0V \leq V_{CC} \leq 9.5V$, $V_{OH} = 2.0V$ | | | |
| $t_{PD1}$ (standard size) | $C_L = 50pF$ | 5.0 | μS | |
| $t_{PD1}$ (high current) | | 4.5 | μS | |

## AC Electrical Characteristics (continued)

$0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise specified.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| COP TO LSTTL PROPAGATION DELAY | $V_{CC} = 5V \pm 5\%$, $V_{OH} = 2.7V$<br>$V_{OL} = 0.4V$, $C_L = 50pF$ | | | |
| SO, SK Outputs | | | | |
| $t_{PD1}$ (standard) | | | 5 | μs |
| $t_{PD1}$ (push-pull) | | | 3.5 | μs |
| $t_{PD0}$ | | | 3 | μs |
| $L_7$-$L_0$ Outputs | | | | |
| $t_{PD1}$ (push-pull) | | | 1.5 | μs |
| $L_7$-$L_0$, $G_3$-$G_0$, $D_3$-$D_0$ Outputs | | | | |
| $t_{PD1}$ (standard) | | | 5.0 | μs |
| $t_{PD0}$ | | | 2.0 | μs |
| CKO (figure 3b) | | | | |
| $t_{PD1}$ | | | 0.4 | μs |
| $t_{PD0}$ | | | 0.4 | μs |

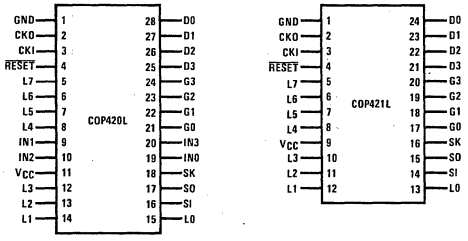



**Figure 2. Connection Diagrams**

| Pin | Description |
|---|---|
| $L_7$-$L_0$ | 8 bidirectional I/O ports with TRI-STATE® |
| $G_3$-$G_0$ | 4 bidirectional I/O ports |
| $D_3$-$D_0$ | 4 general purpose outputs |
| $IN_3$-$IN_0$ | 4 general purpose inputs (COP420L only) |
| SI | Serial input (or counter input) |
| SO | Serial output (or general purpose output) |
| SK | Logic-controlled clock (or general purpose output) |

| Pin | Description |
|---|---|
| CKI | System oscillator input |
| CKO | System oscillator output (or general purpose input, RAM power supply or SYNC input)) |
| $\overline{RESET}$ | System reset input |
| $V_{CC}$ | Power supply |
| GND | Ground |

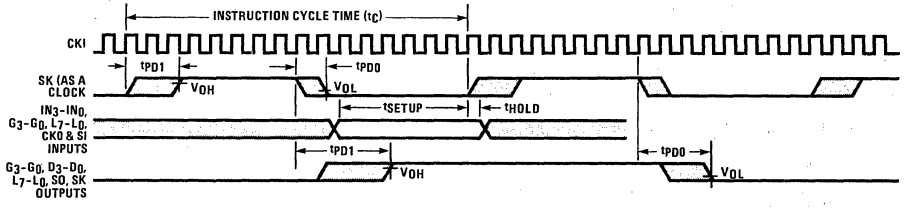



**Figure 3. Input/Output Timing Diagrams (Crystal Divide-by-16 Mode)**

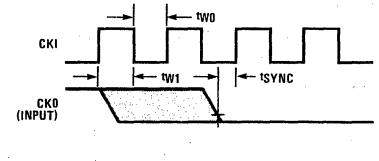



**Figure 3a. Synchronization Timing**

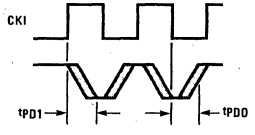



**Figure 3b. CKO Output Timing**

## FUNCTIONAL DESCRIPTION

A block diagram of the COP420L is given in figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1" (greater than 2 volts). When a bit is reset, it is a logic "0" (less than 0.8 volts).

### Program Memory

Program Memory consists of a 1,024-byte ROM. As can be seen by an examination of the COP420L/421L instruction set, these words may be program instructions, program data or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID and LQID instructions, ROM must often be thought of as being organized into 16 pages of 64 words each.

ROM addressing is accomplished by a 10-bit PC register. Its binary value selects one of the 1,024 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 10-bit binary count value. Three levels of subroutine nesting are implemented by the 10-bit subroutine save registers, SA, SB and SC, providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the Instruction Decode, Control and Skip Logic circuitry.

### Data Memory

Data memory consists of a 256-bit RAM, organized as 4 data registers of 16 4-bit digits. RAM addressing is implemented by a 6-bit B register whose upper 2 bits (Br) select 1 of 4 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) is usually loaded into or from, or exchanged with, the A register (accumulator), it may also be loaded into or from the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the 6-bit contents of the operand field of these instructions. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

### Internal Logic

The 4-bit A register (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd portions of the B register, to load and input 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

A 4-bit adder performs the arithmetic and logic functions of the COP420L/421L, storing its results in A. It also outputs a carry bit to the 1-bit C register, most often employed to indicate arithmetic overflow. The C register, in conjunction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SK or can enable SK to be a sync clock each instruction cycle time. (See XAS instruction and EN register description, below.)

Four general-purpose inputs, $IN_3$-$IN_0$, are provided.

The D register provides 4 general-purpose outputs and is used as the destination register for the 4-bit contents of Bd. The D outputs can be directly connected to the digits of a multiplexed LED display.

The G register contents are outputs to 4 general-purpose bidirectional I/O ports. G I/O ports can be directly connected to the digits of a multiplexed LED display.

The Q register is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control. (See LEI instruction.)

The 8 L drivers, when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the LED Direct Drive output configuration option) with Q data being outputted to the Sa-Sg and decimal point segments of the display.

The SIO register functions as a 4-bit serial-in/serial-out shift register or as a binary counter depending on the contents of the EN register. (See EN register description, below.) Its contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O by connecting SO to external serial-in/parallel-out shift registers. For example of additional parallel output capacity see Application #2.

The EN register is an internal 4-bit register loaded under program control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register ($EN_3$-$EN_0$).

1. The least significant bit of the enable register, $EN_0$, selects the SIO register as either a 4-bit shift register or a 4-bit binary counter. With $EN_0$ set, SIO is an asynchronous binary counter, *decrementing* its value by one upon each low-going pulse ("1" to "0") occurring on the SI input. Each pulse must be at least two instruction cycles wide. SK outputs the value of C upon execution of XAS and remains the same until the execution of another XAS instruction. The SO output is equal to the value of $EN_3$. With $EN_0$ reset, SIO is a serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time. (See 4 below.) The SK output becomes a logic-controlled clock, providing a SYNC signal each instruction time. It will start outputting a SYNC pulse upon the execution of an XAS instruction with C = 1, stopping upon the execution of a subsequent XAS with C = 0.

2. With $EN_1$ set the $IN_1$ input is enabled as an interrupt input. Immediately following an interrupt, $EN_1$ is reset to disable further interrupts.
3. With $EN_2$ set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting $EN_2$ disables the L drivers, placing the L I/O ports in a high-impedance input state.
4. $EN_3$, in conjunction with $EN_0$, affects the SO output. With $EN_0$ set (binary counter option selected) SO will output the value loaded into $EN_3$.

With $EN_0$ reset (serial shift register option selected), setting $EN_3$ enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting $EN_3$ with the serial shift register option selected disables SO as the shift register output; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0." The table below provides a summary of the modes associated with $EN_3$ and $EN_0$.

**Enable Register Modes — Bits $EN_3$ and $EN_0$**

| $EN_3$ | $EN_0$ | SIO | SI | SO | SK after XAS |
|---|---|---|---|---|---|
| 0 | 0 | Shift Register | Input to Shift Register | 0 | If C = 1, SK = SYNC<br>If C = 0, SK = 0 |
| 1 | 0 | Shift Register | Input to Shift Register | Serial Out | If C = 1, SK = SYNC<br>If C = 0, SK = 0 |
| 0 | 1 | Binary Counter | Input to Binary Counter | 0 | If C = 1, Sk = 1<br>If C = 0, SK = 0 |
| 1 | 1 | Binary Counter | Input to Binary Counter | 1 | If C = 1, SK = 1<br>If C = 0, SK = 0 |

**Interrupt**

The following features are associated with the $IN_1$ interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

a. The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1) onto the stack, pushing in turn the contents of the other subroutine-save registers to the next lower level (PC + 1 → SA → SB → SC). Any previous contents of SC are lost. The program counter is set to hex address 0FF (the last word of page 3) and $EN_1$ is reset.

b. An interrupt will be acknowledged only after the following conditions are met:
   1. $EN_1$ has been set.
   2. A low-going pulse ("1" to "0") at least two instruction cycles wide occurs on the $IN_1$ input.
   3. A currently executing instruction has been completed.
   4. All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction, the interrupt will not be acknowledged until the second JP instruction has been executed.

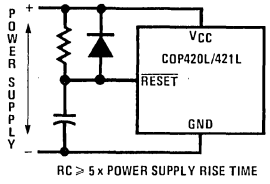
c. Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon the execution of a subsequent RET instruction. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skip logic status is saved and program control is transferred to the interrupt servicing routine at hex address 0FF. At the *end* of the interrupt routine, a RET instruction is executed to "pop" the stack and return program control to the instruction following the original ASC. *At this time*, the skip logic is enabled and skips this instruction because of the previous ASC carry. Since, as explained above, it is the RET instruction which enables the previously saved status of the skip logic, subroutines should not be nested within the interrupt servicing routine since their RET instruction will enable any previously saved main program skips, interfering with the orderly execution of the interrupt routine.

d. The first instruction of the interrupt routine at hex address 0FF must be a NOP.

e. A LEI instruction can be put immediately before the RET to re-enable interrupts.

### Initialization

The Reset Logic will initialize (clear) the device upon power-up if the power supply rise time is less than 1ms and greater than 1μs. If the power supply rise time is greater than 1ms, the user must provide an external RC network and diode to the RESET pin as shown below. The RESET pin is configured as a Schmitt trigger input. If not used it should be connected to $V_{CC}$. Initialization will occur whenever a logic "0" is applied to the RESET input, provided it stays low for at least two instruction cycle times.




**Power-Up Clear Circuit**

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, and G registers are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. *Data Memory (RAM) must be cleared by the user's program.* The first instruction at address 0 must be a CLRA.

### Oscillator

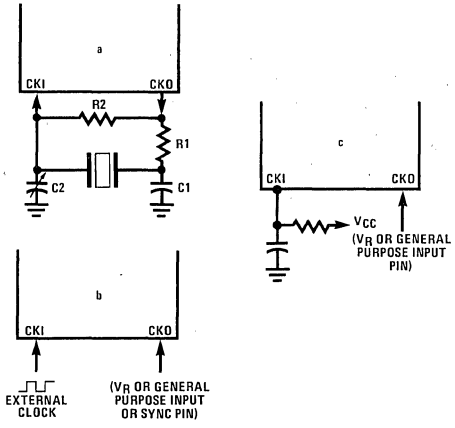
There are four basic clock oscillator configurations available as shown by figure 4.

a. **Crystal Controlled Oscillator.** CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency divided by 32 (optional by 16 or 8).

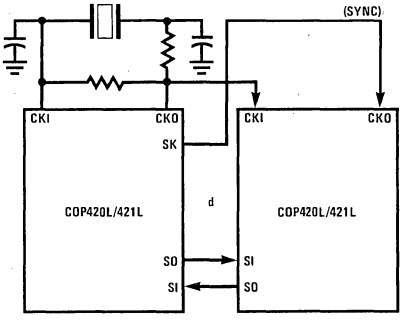
b. **External Oscillator.** CKI is an external clock input signal. The external frequency is divided by 32 (optional by 16 or 8) to give the instruction cycle time. CKO is now available to be used as the RAM power supply ($V_R$), as a general purpose input, or as a SYNC input.

c. **RC Controlled Oscillator.** CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 4. CKO is available as the RAM power supply ($V_R$) or as a general purpose input.

d. **Externally Synchronized Oscillator.** Intended for use in multi-COP systems, CKO is programmed to function as an input connected to the SK output of another COP chip operating at the same frequency (COP chip with L or C suffix) with CKI connected as shown. In this configuration, the SK output connected to CKO must provide a SYNC (instruction cycle) signal to CKO, thereby allowing synchronous data transfer between the COPs using only the SI and SO serial I/O pins in conjunction with the XAS instruction. Note that on power-up SK is automatically enabled as a SYNC output (see Functional Description, Initialization, above).




**Crystal Oscillator**




**RC Controlled Oscillator**

| Crystal Value | Component Values | | | |
|---|---|---|---|---|
| | R1 (Ω) | R2 (Ω) | C1 (pF) | C2 (pF) |
| 455 kHz | 16k | 1M | 80 | 80 |
| 2.097 MHz | 1k | 1M | 56 | 6-36 |

| R (kΩ) | C (pF) | Instruction Cycle Time in μs) |
|---|---|---|
| 51 | 100 | 19 ± 15% |
| 82 | 56 | 19 ± 13% |

**Figure 4. COP420L/421L Oscillator**

### CKO Pin Options

In a crystal controlled oscillator system, CKO is used as an output to the crystal network. As an option CKO can be a SYNC input as described above. As another option CKO can be a general purpose input, read into bit 2 of A (accumulator) upon execution of an INIL instruction. As another option, CKO can be a RAM power supply pin ($V_R$), allowing its connection to a standby/backup power supply to maintain the integrity of RAM data with minimum power drain when the main supply is inoperative or shut down to conserve power. Using either option is appropriate in applications where the COP420L/421L system timing configuration does not require use of the CKO pin.

### I/O Options

COP420L/421L outputs have the following optional configurations, illustrated in figure 5:

**a. Standard** — an enhancement-mode device to ground in conjunction with a depletion-mode device to $V_{CC}$, compatible with LSTTL and CMOS input requirements. Available on SO, SK, and all D and G outputs.

**b. Open-Drain** — an enhancement-mode device to ground only, allowing external pull-up as required by the user's application. Available on SO, SK, and all D and G outputs.

**c. Push-Pull** — an enhancement-mode device to ground in conjunction with a depletion-mode device paralleled by an enhancement-mode device to $V_{CC}$. This configuration has been provided to allow for fast rise and fall times when driving capacitive loads. Available on SO and SK outputs only.

**d. Standard L** — same as **a.**, but may be disabled. Available on L outputs only.

**e. Open Drain L** — same as **b.**, but may be disabled. Available on L outputs only.

**f. LED Direct Drive** — an enhancement mode device to ground and to $V_{CC}$, meeting the typical current sourcing requirements of the segments of an LED display. The sourcing device is clamped to limit current flow. These devices may be turned off under program control (see Functional Description, EN Register), placing the outputs in a high-impedance state to provide required LED segment blanking for a multiplexed display. Available on L outputs only.

**g. TRI-STATE® Push-Pull** — an enhancement-mode device to ground and $V_{CC}$. These outputs are TRI-STATE outputs, allowing for connection of these outputs to a data bus shared by other bus drivers. Available on L outputs only.

COP420L/421L inputs have the following optional configurations:

**h.** An on-chip depletion load device to $V_{CC}$.

**i.** A Hi-Z input which must be driven to a "1" or "0" by external components.

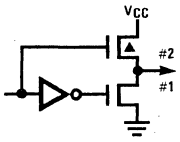
The above input and output configurations share common enhancement-mode and depletion-mode devices. Specifically, all configurations use one or more of six devices (numbered 1-6, respectively). Minimum and maximum current ($I_{OUT}$ and $V_{OUT}$) curves are given in figure 6 for each of these devices to allow the designer to effectively use these I/O configurations in designing a COP420L/421L system.

The SO,SK outputs can be configured as shown in **a.**, **b.**, or **c.** The D and G outputs can be configured as shown in **a.** or **b.** Note that when inputting data to the G ports, the G outputs should be set to "1." The L outputs can be configured as in **d.**, **e.**, **f** or **g.**

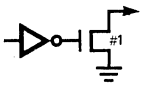
An important point to remember if using configuration **d.** or **f.** with the L drivers is that even when the L drivers are disabled, the depletion load device will source a small amount of current (see figure 6, device 2).

### COP421L

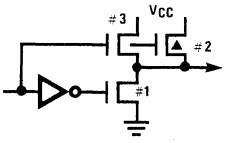
If the COP420L is bonded as a 24-pin device, it becomes the COP421L, illustrated in figure 2, COP420L/421L Connection Diagrams. Note that the COP421L does not contain the four general purpose IN inputs ($IN_3$-$IN_0$). Use of this option precludes, of course, use of the IN options and the interrupt feature, which uses $IN_1$-$IN_3$. All other options are available for the COP421L.



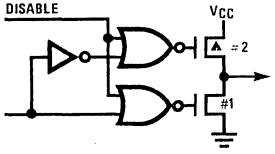

a. Standard Output



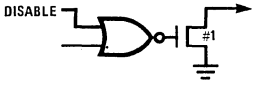

b. Open-Drain Output



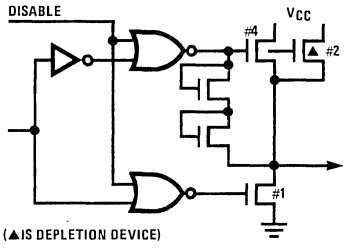

c. Push-Pull Output



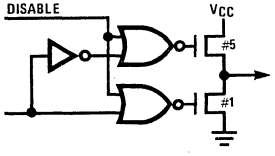

d. Standard L Output



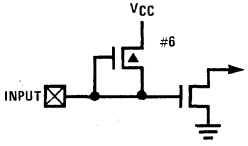

e. Open-Drain L Output



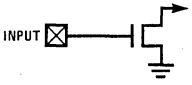

f. LED (L Output)




g. TRI-STATE® Push-Pull (L Output)



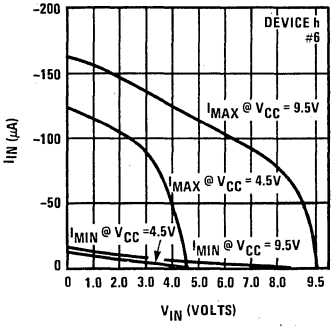

h. Input with Load



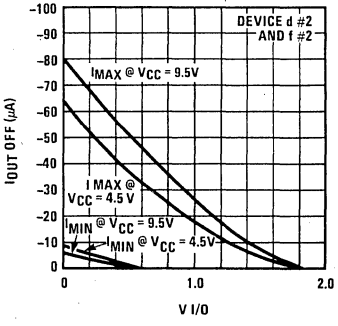

i. Hi-Z Input

Figure 5. Output Configurations

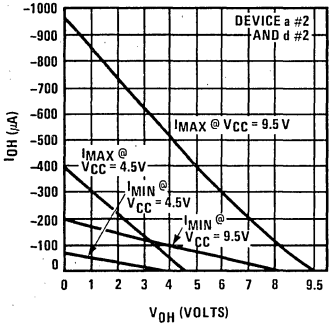
**Input Current IN0-IN3**



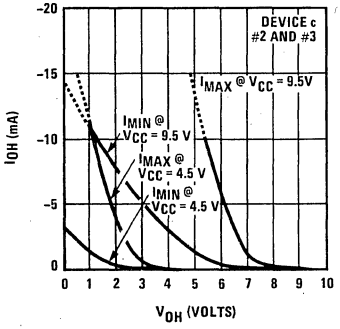

**Input Current for L0 through L7 when Output Programmed Off by Software**



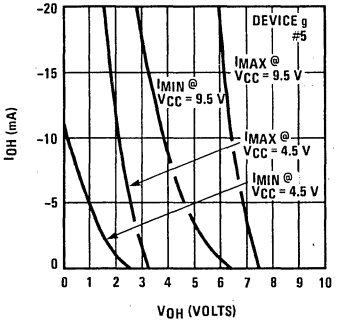

**Source Current for Standard Output Configuration**



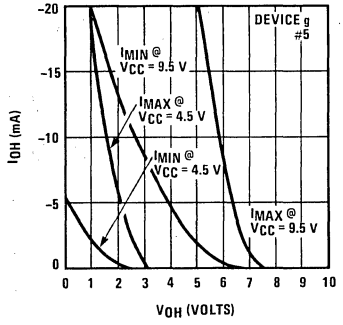

**Source Current for SO and SK in Push-Pull Configuration**



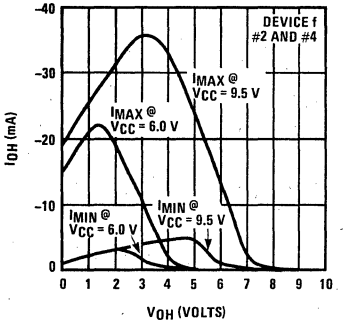

**Source Current for L0 through L7 in TRI-STATE® Configuration (High Current Option)**



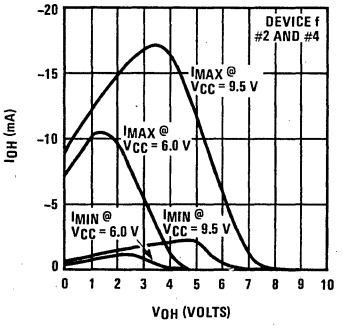

**Source Current for L0 through L7 in TRI-STATE® Configuration (Low Current Option)**



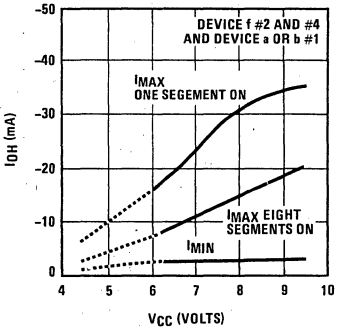
LED Output Source Current
(for High Current LED Option)



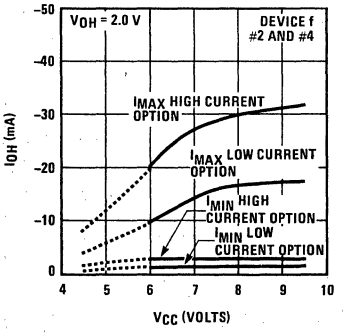

LED Output Source Current
(for Low Current LED Option)



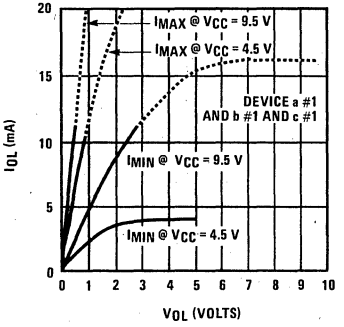

LED Output Direct Segment and Digit Drive
High Current Options on L0-L7
Very High Current Options on D0-D3 or G0-G3



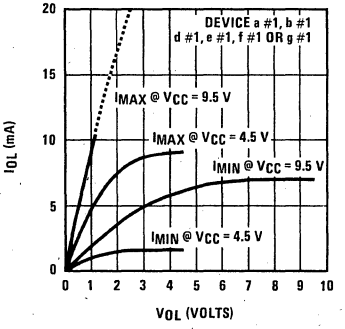

LED Output Direct Segment Drive



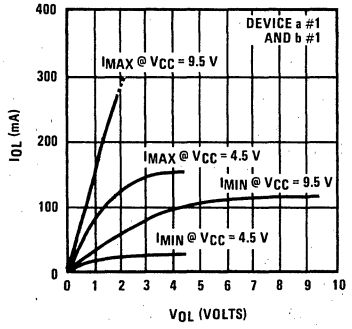

Output Sink Current for SO and SK



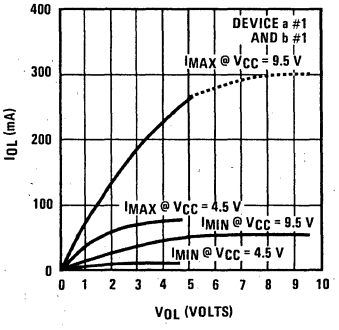

Output Sink Current for L0-L7 and Standard Drive Option for D0-D3 and G0-G3




Output Sink Current
G0-G3 and D0-D3 with Very High Current Option




Output Sink Current for G0-G3 and D0-D3 (for High Current Option)



## COP420L/421L INSTRUCTION SET

Table 1 is a symbol table providing internal architecture, instruction operand and operational symbols used in the instruction set table.

Table 2 provides the mnemonic, operand, machine code, data flow, skip conditions and description associated with each instruction in the COP420L/421L instruction set.

**Table 1. COP420L/421L Instruction Set Table Symbols**

| Symbol | Definition |
|---|---|
| **INTERNAL ARCHITECTURE SYMBOLS** | |
| A | 4-bit Accumulator |
| B | 6-bit RAM Address Register |
| Br | Upper 2 bits of B (register address) |
| Bd | Lower 4 bits of B (digit address) |
| C | 1-bit Carry Register |
| D | 4-bit Data Output Port |
| EN | 4-bit Enable Register |
| G | 4-bit Register to latch data for G I/O Port |
| IL | Two 1-bit Latches associated with the $IN_3$ or $IN_0$ Inputs |
| IN | 4-bit Input Port |
| L | 8-bit TRI-STATE I/O Port |
| M | 4-bit contents of RAM Memory pointed to by B Register |
| PC | 10-bit ROM Address Register (program counter) |
| Q | 8-bit Register to latch data for L I/O Port |
| SA | 10-bit Subroutine Save Register A |
| SB | 10-bit Subroutine Save Register B |
| SC | 10-bit Subroutine Save Register C |
| SIO | 4-bit Shift Register and Counter |
| SK | Logic-Controlled Clock Output |

| Symbol | Definition |
|---|---|
| **INSTRUCTION OPERAND SYMBOLS** | |
| d | 4-bit Operand Field, 0-15 binary (RAM Digit Select) |
| r | 2-bit Operand Field, 0-3 binary (RAM Register Select) |
| a | 10-bit Operand Field, 0-1024 binary (ROM Address) |
| y | 4-bit Operand Field, 0-15 binary (Immediate Data) |
| RAM(s) | Contents of RAM location addressed by s |
| ROM(t) | Contents of ROM location addressed by t |

| | |
|---|---|
| **OPERATIONAL SYMBOLS** | |
| + | Plus |
| − | Minus |
| → | Replaces |
| ⟷ | Is exchanged with |
| = | Is equal to |
| $\overline{A}$ | The ones complement of A |
| ⊕ | Exclusive-OR |
| : | Range of values |

Table 2. COP420L/421L Instruction Set

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| ARITHMETIC INSTRUCTIONS | | | | | | |
| ASC | | 30 | 0 0 1 1 \| 0 0 0 0 | A + C + RAM(B) → A<br>Carry → C | Carry | Add with Carry, Skip on Carry |
| ADD | | 31 | 0 0 1 1 \| 0 0 0 1 | A + RAM(B) → A | None | Add RAM to A |
| ADT | | 4A | 0 1 0 0 \| 1 0 1 0 | A + $10_{10}$ → A | None | Add Ten to A |
| AISC | y | 5- | 0 1 0 1 \| y | A + y → A | Carry | Add Immediate, Skip on Carry (y ≠ 0) |
| CASC | | 10 | 0 0 0 1 \| 0 0 0 0 | $\overline{A}$ + RAM(B) + C → A<br>Carry → C | Carry | Complement and Add with Carry, Skip on Carry |
| CLRA | | 00 | 0 0 0 0 \| 0 0 0 0 | 0 → A | None | Clear A |
| COMP | | 40 | 0 1 0 0 \| 0 0 0 0 | $\overline{A}$ → A | None | Ones complement of A to A |
| NOP | | 44 | 0 1 0 0 \| 0 1 0 0 | None | None | No Operation |
| RC | | 32 | 0 0 1 1 \| 0 0 1 0 | "0" → C | None | Reset C |
| SC | | 22 | 0 0 1 0 \| 0 0 1 0 | "1" → C | None | Set C |
| XOR | | 02 | 0 0 0 0 \| 0 0 1 0 | A ⊕ RAM(B) → A | None | Exclusive-OR RAM with A |
| TRANSFER OF CONTROL INSTRUCTIONS | | | | | | |
| JID | | FF | 1 1 1 1 \| 1 1 1 1 | ROM ($PC_{9:8}$,A,M) → $PC_{7:0}$ | None | Jump Indirect (Note 3) |
| JMP | a | 6-<br>-- | 0 1 1 0 \| 0 0 \| $a_{9:8}$<br>$a_{7:0}$ | a → PC | None | Jump |
| JP | a | -- | 1 \| $a_{6:0}$<br>(pages 2,3 only)<br>or | a → $PC_{6:0}$ | None | Jump within Page (Note 4) |
| | | -- | 1 1 \| $a_{5:0}$<br>(all other pages) | a → $PC_{5:0}$ | | |
| JSRP | a | -- | 1 0 \| $a_{5:0}$ | PC + 1 → SA → SB → SC<br>0010 → $PC_{9:6}$<br>a → $PC_{5:0}$ | None | Jump to Subroutine Page (Note 5) |
| JSR | a | 6-<br>-- | 0 1 1 0 \| 1 0 \| $a_{9:8}$<br>$a_{7:0}$ | PC + 1 → SA → SB → SC<br>a → PC | None | Jump to Subroutine |
| RET | | 48 | 0 1 0 0 \| 1 0 0 0 | SC → SB → SA → PC | None | Return from Subroutine |
| RETSK | | 49 | 0 1 0 0 \| 1 0 0 1 | SC → SB → SA → PC | Always Skip on Return | Return from Subroutine then Skip |

Table 2. COP420L/421L Instruction Set (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **MEMORY REFERENCE INSTRUCTIONS** | | | | | | |
| CAMQ | | 33 | 0 0 1 1 \| 0 0 1 1 | $A \rightarrow Q_{7:4}$<br>$RAM(B) \rightarrow Q_{3:0}$ | None | Copy A, RAM to Q |
| | | 3C | 0 0 1 1 \| 1 1 0 0 | | | |
| CQMA | | 33 | 0 0 1 1 \| 0 0 1 1 | $Q_{7:4} \rightarrow RAM(B)$<br>$Q_{3:0} \rightarrow A$ | None | Copy Q to RAM, A |
| | | 2C | 0 0 1 0 \| 1 1 0 0 | | | |
| LD | r | -5 | 0 0 \| r \| 0 1 0 1 | $RAM(B) \rightarrow A$<br>$Br \oplus r \rightarrow Br$ | None | Load RAM into A, Exclusive-OR Br with r |
| LDD | r,d | 23 | 0 0 1 0 \| 0 0 1 1 | $RAM(r,d) \rightarrow A$ | None | Load A with RAM pointed to directly by r,d |
| | | -- | 0 0 \| r \| d | | | |
| LQID | | BF | 1 0 1 1 \| 1 1 1 1 | $ROM(PC_{9:8},A,M) \rightarrow Q$<br>$SB \rightarrow SC$ | None | Load Q Indirect (Note 3) |
| RMB | 0 | 4C | 0 1 0 0 \| 1 1 0 0 | $0 \rightarrow RAM(B)_0$ | None | Reset RAM Bit |
| | 1 | 45 | 0 1 0 0 \| 0 1 0 1 | $0 \rightarrow RAM(B)_1$ | | |
| | 2 | 42 | 0 1 0 0 \| 0 0 1 0 | $0 \rightarrow RAM(B)_2$ | | |
| | 3 | 43 | 0 1 0 0 \| 0 0 1 1 | $0 \rightarrow RAM(B)_3$ | | |
| SMB | 0 | 4D | 0 1 0 0 \| 1 1 0 1 | $1 \rightarrow RAM(B)_0$ | None | Set RAM Bit |
| | 1 | 47 | 0 1 0 0 \| 0 1 1 1 | $1 \rightarrow RAM(B)_1$ | | |
| | 2 | 46 | 0 1 0 0 \| 0 1 1 0 | $1 \rightarrow RAM(B)_2$ | | |
| | 3 | 4B | 0 1 0 0 \| 1 0 1 1 | $1 \rightarrow RAM(B)_3$ | | |
| STII | y | 7- | 0 1 1 1 \| y | $y \rightarrow RAM(B)$<br>$Bd + 1 \rightarrow Bd$ | None | Store Memory Immediate and Increment Bd |
| X | r | -6 | 0 0 \| r \| 0 1 1 0 | $RAM(B) \longleftrightarrow A$<br>$Br \oplus r \rightarrow Br$ | None | Exchange RAM with A, Exclusive-OR Br with r |
| XAD | r,d | 23 | 0 0 1 0 \| 0 0 1 1 | $RAM(r,d) \longleftrightarrow A$ | None | Exchange A with RAM pointed to directly by r,d |
| | | -- | 1 0 \| r \| d | | | |
| XDS | r | -7 | 0 0 \| r \| 0 1 1 1 | $RAM(B) \longleftrightarrow A$<br>$Bd - 1 \rightarrow Bd$<br>$Br \oplus r \rightarrow Br$ | Bd decrements past 0 | Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r |
| XIS | r | -4 | 0 0 \| r \| 0 1 0 0 | $RAM(B) \longleftrightarrow A$<br>$Bd + 1 \rightarrow Bd$<br>$Br \oplus r \rightarrow Br$ | Bd increments past 15 | Exchange RAM with A and Increment Bd, Exclusive-OR Br with r |
| **REGISTER REFERENCE INSTRUCTIONS** | | | | | | |
| CAB | | 50 | 0 1 0 1 \| 0 0 0 0 | $A \rightarrow Bd$ | None | Copy A to Bd |
| CBA | | 4E | 0 1 0 0 \| 1 1 1 0 | $Bd \rightarrow A$ | None | Copy Bd to A |
| LBI | r,d | -- | 0 0 \| r \| (d – 1)<br>(d = 0, 9:15) | $r,d \rightarrow B$ | Skip until not a LBI | Load B Immediate with r,d (Note 6) |
| | | | or | | | |
| | | 33 | 0 0 1 1 0 0 1 1 | | | |
| | | -- | 1 0 \| r \| d<br>(any d) | | | |
| LEI | y | 33 | 0 0 1 1 \| 0 0 1 1 | $y \rightarrow EN$ | None | Load EN Immediate (Note 7) |
| | | 6- | 0 1 1 0 \| y | | | |
| XABR | | 12 | 0 0 0 1 \| 0 0 1 0 | $A \longleftrightarrow Br\ (0,0 \rightarrow A_3,A_2)$ | None | Exchange A with Br |

Table 2. COP420L/421L Instruction Set (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **TEST INSTRUCTIONS** | | | | | | |
| SKC | | 20 | 0 0 1 0 \| 0 0 0 0 | | C = "1" | Skip if C is True |
| SKE | | 21 | 0 0 1 0 \| 0 0 0 1 | | A = RAM(B) | Skip if A Equals RAM |
| SKGZ | | 33 | 0 0 1 1 \| 0 0 1 1 | | $G_{3:0}$ = 0 | Skip if G is Zero (all 4 bits) |
| | | 21 | 0 0 1 0 \| 0 0 0 1 | | | |
| SKGBZ | | 33 | 0 0 1 1 \| 0 0 1 1 | 1st byte | | Skip if G Bit is Zero |
| | 0 | 01 | 0 0 0 0 \| 0 0 0 1 | 2nd byte | $G_0$ = 0 | |
| | 1 | 11 | 0 0 0 1 \| 0 0 0 1 | | $G_1$ = 0 | |
| | 2 | 03 | 0 0 0 0 \| 0 0 1 1 | | $G_2$ = 0 | |
| | 3 | 13 | 0 0 0 1 \| 0 0 1 1 | | $G_3$ = 0 | |
| SKMBZ | 0 | 01 | 0 0 0 0 \| 0 0 0 1 | | $RAM(B)_0$ = 0 | Skip if RAM Bit is Zero |
| | 1 | 11 | 0 0 0 1 \| 0 0 0 1 | | $RAM(B)_1$ = 0 | |
| | 2 | 03 | 0 0 0 0 \| 0 0 1 1 | | $RAM(B)_2$ = 0 | |
| | 3 | 13 | 0 0 0 1 \| 0 0 1 1 | | $RAM(B)_3$ = 0 | |
| SKT | | 41 | 0 1 0 0 \| 0 0 0 1 | | A time-base counter carry has occurred since last test | Skip on Timer (Note 3) |
| **INPUT/OUTPUT INSTRUCTIONS** | | | | | | |
| ING | | 33 | 0 0 1 1 \| 0 0 1 1 | G → A | None | Input G Ports to A |
| | | 2A | 0 0 1 0 \| 1 0 1 0 | | | |
| ININ | | 33 | 0 0 1 1 \| 0 0 1 1 | IN → A | None | Input IN Inputs to A (Note 2) |
| | | 28 | 0 0 1 0 \| 1 0 0 0 | | | |
| INIL | | 33 | 0 0 1 1 \| 0 0 1 1 | $IL_3$,CKO,"0",$IL_0$ → A | None | Input IL Latches to A (Note 3) |
| | | 29 | 0 0 1 0 \| 1 0 0 1 | | | |
| INL | | 33 | 0 0 1 1 \| 0 0 1 1 | $L_{7:4}$ → RAM(B) | None | Input L Ports to RAM,A |
| | | 2E | 0 0 1 0 \| 1 1 1 0 | $L_{3:0}$ → A | | |
| OBD | | 33 | 0 0 1 1 \| 0 0 1 1 | Bd → D | None | Output Bd to D Outputs |
| | | 3E | 0 0 1 1 \| 1 1 1 0 | | | |
| OGI | y | 33 | 0 0 1 1 \| 0 0 1 1 | y → G | None | Output to G Ports Immediate |
| | | 5- | 0 1 0 1 \| y | | | |
| OMG | | 33 | 0 0 1 1 \| 0 0 1 1 | RAM(B) → G | None | Output RAM to G Ports |
| | | 3A | 0 0 1 1 \| 1 0 1 0 | | | |
| XAS | | 4F | 0 1 0 0 \| 1 1 1 1 | A ⟷ SIO, C → SK | None | Exchange A with SIO (Note 3) |

**Note 1:** All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, $A_3$ indicates the most significant (left-most) bit of the 4-bit A register.

**Note 2:** The INI instruction is not available on the 24-pin COP421L since this device does not contain the IN inputs.

**Note 3:** For additional information on the operation of the XAS, JID, LQID, INIL, and SKT instructions, see below.

**Note 4:** The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

**Note 5:** A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

**Note 6:** LBI is a single-byte instruction if d = 0, 9, 10, 11, 12, 13, 14, or 15. The machine code for the lower 4 bits equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 ($1001_2$), the lower 4 bits of the LBI instruction equal 8 ($1000_2$). To load 0, the lower 4 bits of the LBI instruction should equal 15 ($1111_2$).

**Note 7:** Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register.)

The following information is provided to assist the user in understanding the operation of several unique instructions and to provide notes useful to programmers in writing COP420L/421L programs.

### XAS Instruction

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register. The contents of SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output, providing a logic controlled clock if SIO is selected as a shift register or C → SK if SIO is selected as a binary counter. (See Functional Description, EN Register, above.) If SIO is selected as a shift register, an XAS instruction must be performed once every 4 instruction cycles to effect a continuous data stream.

### JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM addressed by the 10-bit word, $PC_{9:8}$, A, M. $PC_9$ and $PC_8$ are not affected by this instruction.

Note that JID requires 2 instruction cycles.

### INIL Instruction

INIL (Input IL Latches to A) inputs 2 latches, $IL_3$ and $IL_0$ (see figure 11) and CKO into A. The $IL_3$ and $IL_0$ latches are set if a low-going pulse ("1" to "0") has occurred on the $IN_3$ and $IN_0$ inputs since the last INIL instruction, provided the input pulse stays low for at least two instruction times. Execution of an INIL inputs $IL_3$ and $IL_0$ into A3 and A0 respectively, and resets these latches to allow them to respond to subsequent low-going pulses on the $IN_3$ and $IN_0$ lines. If CKO is mask programmed as a general purpose input, an INIL will input the state of CKO into A2. If CKO has not been so programmed, a "1" will be placed in A2. A "0" is always placed in A1 upon the execution of an INIL. The general purpose inputs $IN_3$-$IN_0$ are input to A upon the execution of an ININ instruction. (See table 2, ININ Instruction.) INIL is useful in recognizing pulses of short duration or pulses which occur too often to be read conveniently by an ININ instruction. Available on COP420L only.

### LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 10-bit word $PC_9$, $PC_8$, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB → SC) and replaces the least significant 8 bits of PC as follows: A → $PC_{7:4}$, RAM(B) → $PC_{3:0}$, leaving $PC_9$ and $PC_8$ unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped"(SC → SB → SA → PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SB → SC, the previous contents of SC are lost. Also, when LQID pops the stack, the previously pushed contents of SB are left in SC. The net result is that the contents of SB are placed in SC (SB → SC). Note that LQID takes two instruction cycle times to execute.

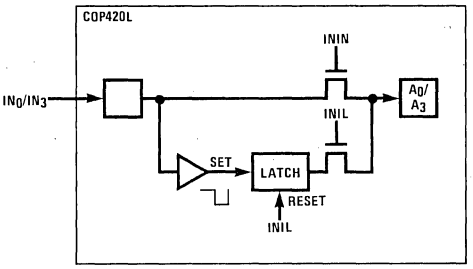



**Figure 10. INIL Hardware Implementation**

### SKT Instruction

The SKT (Skip on Timer) instruction tests the state of an internal 10-bit time-base counter. This counter divides the instruction cycle clock frequency by 1024 and provides a latched indication of counter overflow. The SKT instruction tests this latch, executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction, therefore, allow the COP420L/421L to generate its own time-base for real-time processing rather than relying on an external input signal.

For example, using a 2.097 MHz crystal as the time-base to the clock generator, the instruction cycle clock frequency will be 65 kHz (crystal frequency ÷ 32) and the binary counter output pulse frequency will be 64 Hz. For time-of-day or similar real-time processing, the SKT instruction can call a routine which increments a "seconds" counter every 64 ticks.

### Instruction Set Notes

a. The first word of a COP420L/421L program (ROM address 0) must be a CLRA (Clear A) instruction.

b. Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths take the same number of cycle times whether instructions are skipped or executed.

c. The ROM is organized into 16 pages of 64 words each. The Program Counter is a 10-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example: a JP located in the last word of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3, 7, 11, or 15 will access data in the next group of 4 pages.

## OPTION LIST

The COP420L/421L mask-programmable options are assigned numbers which correspond with the COP*420L* pins.

The following is a list of COP420L options. When specifying a COP421L chip, Options 9, 10, 19, and 20 must all be set to zero. The options are programmed at the same time as the ROM pattern to provide the user with the hardware flexibility to interface to various I/O components using little or no external circuitry.

Option 1 = 0: Ground Pin — no options available

Option 2: CKO Output
- = 0: clock generator output to crystal/resonator (0 not allowable value if Option 3 = 3)
- = 1: pin is RAM power supply ($V_R$) input
- = 2: general purpose input, load device to $V_{CC}$
- = 3: general purpose input, high-Z
- = 4: multi-COP SYNC input (CKI ÷ 32, CKI ÷ 16)
- = 5: multi-COP SYNC input (CKI ÷ 8)

Option 3: CKI Input
- = 0: oscillator input divided by 32 (2 MHz max)
- = 1: oscillator input divided by 16 (1 MHz max)
- = 2: oscillator input divided by 8 (500 kHz max)
- = 3: single-pin RC controlled oscillator divided by 4

Option 4: RESET Input
- = 0: load device to $V_{CC}$
- = 1: Hi-Z input

Option 5: $L_7$ Driver
- = 0: Standard output
- = 1: Open-drain output
- = 2: High current LED direct segment drive output
- = 3: High current TRI-STATE push-pull output
- = 4: Low-current LED direct segment drive output
- = 5: Low-current TRI-STATE push-pull output

Option 6: $L_6$ Driver
same as Option 5

Option 7: $L_5$ Driver
same as Option 5

Option 8: $L_4$ Driver
same as Option 5

Option 9: $IN_1$ Input
- = 0: load device to $V_{CC}$
- = 1: Hi-Z input

Option 10: $IN_2$ Input
same as Option 9

Option 11: $V_{CC}$ pin
- = 0: 4.5V to 6.3V operation
- = 1: 4.5V to 9.5V operation

Option 12: $L_3$ Driver
same as Option 5

Option 13: $L_2$ Driver
same as Option 5

Option 14: $L_1$ Driver
same as Option 5

Option 15: $L_0$ Driver
same as Option 5

Option 16: SI Input
same as Option 9

Option 17: SO Driver
- = 0: standard output
- = 1: open-drain output
- = 2: push-pull output

Option 18: SK Driver
same as Option 17

Option 19: $IN_0$ Input
same as Option 9

Option 20: $IN_3$ Input
same as Option 9

Option 21: $G_0$ I/O Port
- = 0: very-high current standard output
- = 1: very-high current open-drain output
- = 2: high current standard output
- = 3: high current open-drain output
- = 4: standard LSTTL output (fanout = 1)
- = 5: open-drain LSTTL output (fanout = 1)

Option 22: $G_1$ I/O Port
same as Option 21

Option 23: $G_2$ I/O Port
same as Option 21

Option 24: $G_3$ I/O Port
same as Option 21

Option 25: $D_3$ Output
same as Option 21

Option 26: $D_2$ Output
same as Option 21

Option 27: $D_1$ Output
same as Option 21

Option 28: $D_0$ Output
same as Option 21

Option 29: L Input Levels
- = 0: standard TTL input levels ("0" = 0.8V, "1" = 2.0V)
- = 1: higher voltage input levels ("0" = 1.2V, "1" = 3.6V)

Option 30: IN Input Levels
same as Option 29

Option 31: G Input Levels
same as Option 29

Option 32: SI Input Levels
same as Option 29

Option 33: RESET Input
= 0: Schmitt trigger input
= 1: standard TTL input levels
= 2: higher voltage input levels

Option 34: CKO Input Levels (CKO = input; Option 2 = 2,3)
same as Option 29

Option 35: COP Bonding
= 0: COP420L (28-pin device)
= 1: COP421L (24-pin device)

**TEST MODE (Non-Standard Operation)**

The SO output has been configured to provide for standard test procedures for the custom-programmed COP420L. With SO forced to logic "1," two test modes are provided, depending upon the value of SI:

a. RAM and Internal Logic Test Mode (SI = 1)

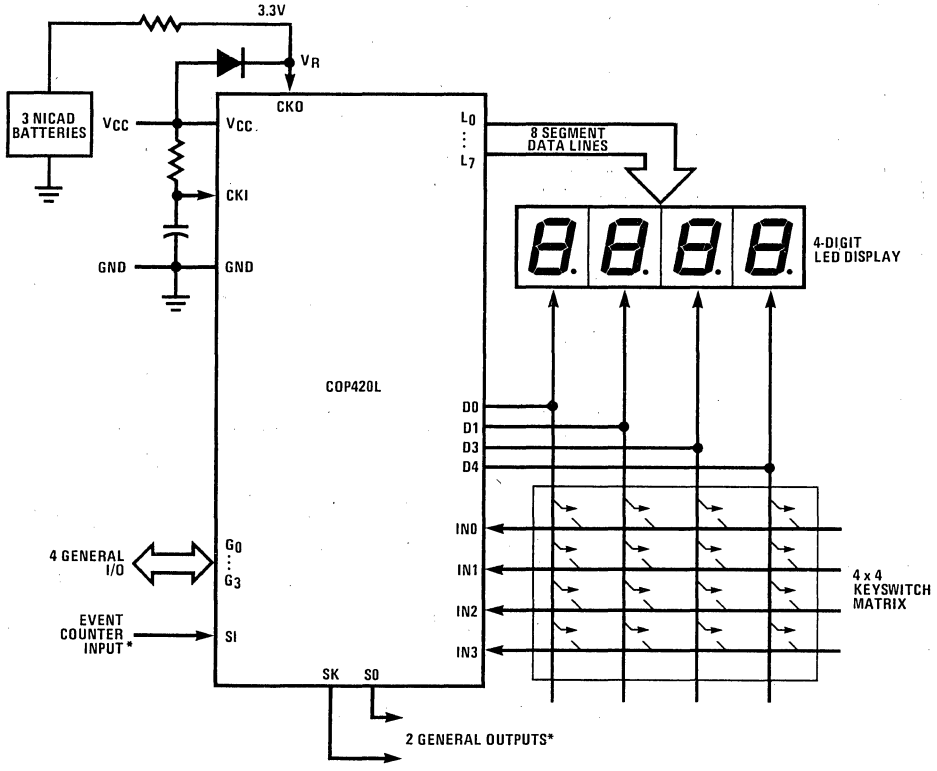
b. ROM Test Mode (SI = 0)

These special test modes should not be employed by the user; they are intended for manufacturing test only.

**APPLICATION #1: COP420L General Controller**

Figure 7 shows an interconnect diagram for a COP420L used as a general controller. Operation of the system is as follows:

1. The $L_7$-$L_0$ outputs are configured as LED Direct Drive outputs, allowing direct connection to the segments of the display.
2. The $D_3$-$D_0$ outputs drive the digits of the multiplexed display directly and scan the columns of the 4 x 4 keyboard matrix.
3. The $IN_3$-$IN_0$ inputs are used to input the 4 rows of the keyboard matrix. Reading the IN lines in conjunction with the current value of the D outputs allows detection, debouncing, and decoding of any one of the 16 keyswitches.
4. CKI is configured as a single-pin oscillator input allowing system timing to be controlled by a single-pin RC network. CKO is therefore available for use as a $V_R$ RAM power supply pin. RAM data integrity is thereby assured when the main power supply is shut down.
5. SI is selected as the input to a binary counter input. With SIO used as a binary counter, SO and SK can be used as general purpose outputs.
6. The 4 bidirectional G I/O ports ($G_3$-$G_0$) are available for use as required by the user's application.




* SO, SI, SK MAY ALSO BE USED FOR SERIAL I/O

**Figure 7. COP420L Keyboard/Display Interface**

**APPLICATION #2: Digitally Tuned Automotive Radio Controller and Clock**

Figure 8 shows the COP420L interconnect diagram for a digitally tuned AM/FM car radio with digital clock LED display and 4 x 6 keyboard for storage and recall of station, search up and search down or scan up and scan down of stations, AM/FM select and time setting and display. Operation of the system is as follows:

1. The DS8907 uses a 4.0 MHz crystal to provide the time base for frequency synthesis and the 500 kHz time base for operation of the COP420L and the 50 Hz signal for the timekeeping function.
2. An unswitched 5V supply goes to the $V_{CCM}$ pin of the DS8907 for the operation of the oscillator and divide-down for the 500 kHz and 50 Hz signals. It also provides $V_{CC}$ for the COP420L so the time-keeping channel storage and last station selected data are not lost when the ignition is off.
3. A switched 5V supply that goes high when the radio is turned on goes to the $V_{CC}$ pin of the DS8907 for the frequency generating circuitry and to the G1 I/O pin of the COP420L.
4. $L_1$ through $L_6$ are outputs to the keyboard (push-pull options selected) and $IN_0$ through $IN_4$ are the keyboard inputs (pullup to $V_{CC}$ and high trip levels selected).
5. SK provides the clock and SO provides the data to the MM5450 display driver with serial input and to the DS8907 PLL synthesizer. $L_7$ is the enable pin for the MM5450 and $G_0$ (standard option selected) is the enable line for the DS8907.
6. In the search up and search down operations, $G_2$ informs the COP420L when a station has been detected.

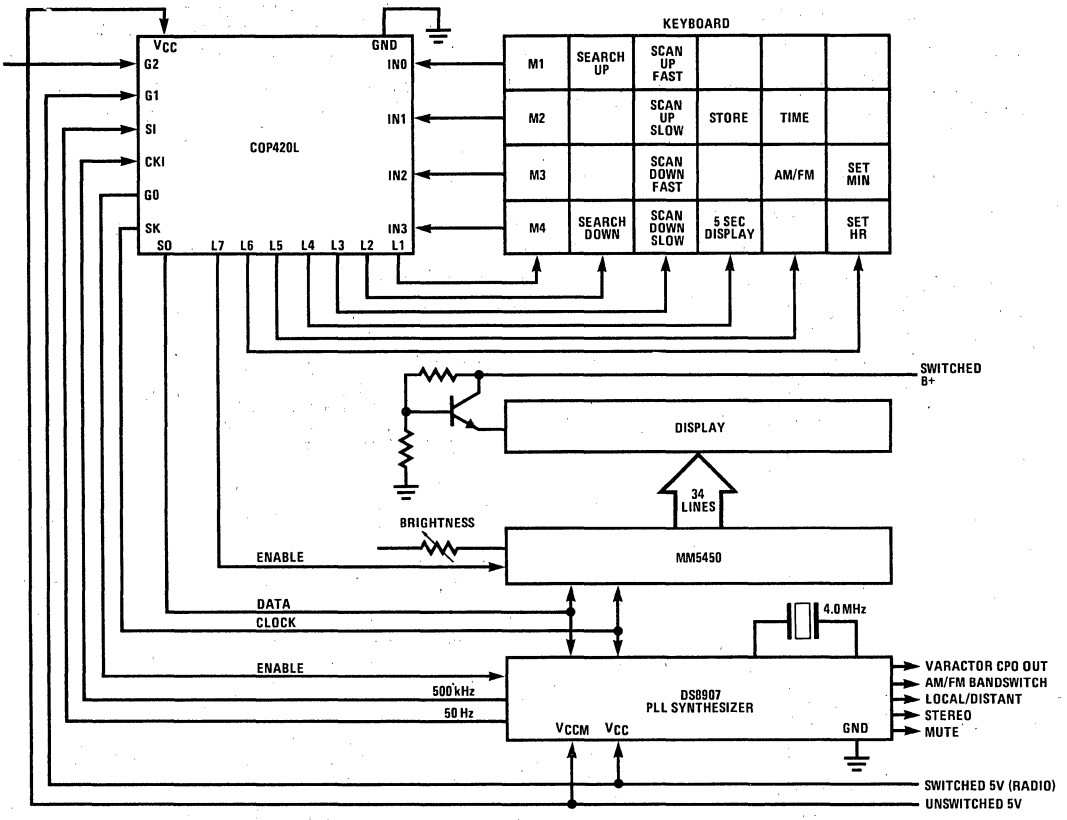



**Figure 8. Electronically Tuned Radio System**

National Semiconductor

PRELIMINARY

# COP444L/445L Single-Chip N-Channel Microcontrollers

## General Description

The COP444L and COP445L Single-Chip N-Channel Microcontrollers are members of the COPS™ family, fabricated using N-channel, silicon gate MOS technology. These controller oriented processors are complete microcomputers containing all system timing, internal logic, ROM, RAM and I/O necessary to implement dedicated control functions in a variety of applications. Features include single supply operation, a variety of output configuration options, with an instruction set, internal architecture and I/O scheme designed to facilitate keyboard input, display output and BCD data manipulation. The COP445L is identical to the COP444L, but with 19 I/O lines instead of 23. They are an appropriate choice for use in numerous human interface control environments. Standard test procedures and reliable high-density fabrication techniques provide the medium to large volume customers with a customized controller oriented processor at a low end-product cost.

## Features

- Low cost
- Powerful instruction set
- 2k × 8 ROM, 128 × 4 RAM
- 23 I/O lines (COP444L)
- True vectored interrupt, plus restart
- Three-level subroutine stack
- 15μs instruction time
- Single supply operation (4.5-6.3V)
- Low current drain (11mA max @ 5V)
- Internal time-base counter for real-time procesing
- Internal binary counter register with MICROWIRE™ serial I/O capability
- General purpose and TRI-STATE® outputs
- LSTTL/CMOS compatible in and out
- Direct drive of LED digit and segment lines
- Software/hardware compatible with other members of COP400 family
- Extended temperature range device to be available (−40°C to +85°C)
- Wider supply range (4.5-9.5V) optionally available

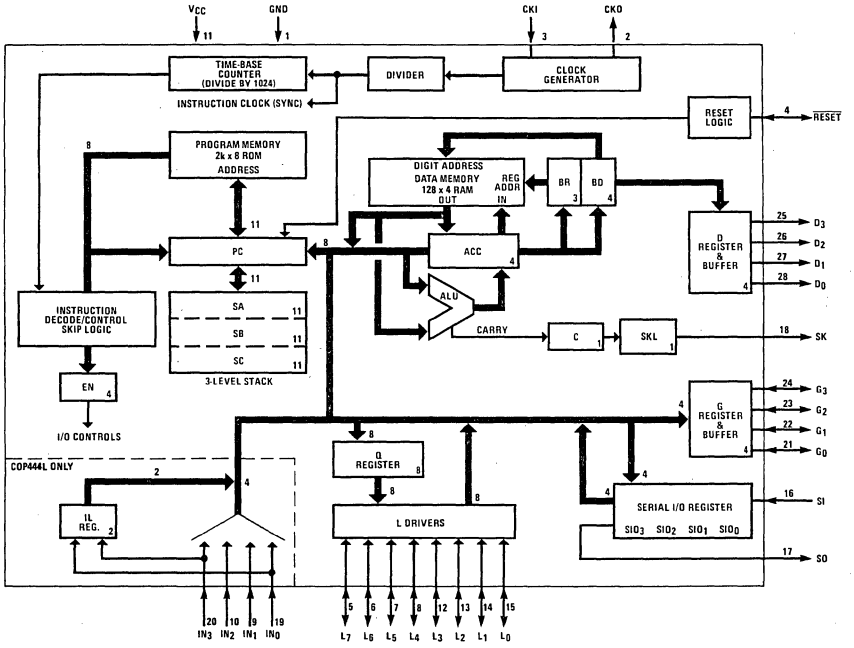



**Figure 1. COP444L/445L Block Diagram**

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Relative to GND | −0.3V to +10V |
| Ambient Operating Temperature | 0°C to +70°C |
| Ambient Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |
| Power Dissipation | 0.75 Watt at 25°C<br>0.4 Watt at 70°C |

*Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.*

## DC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Operating Voltage ($V_{CC}$) | Note 1 | 4.5 | 9.5 | V |
| Operating Supply Current | $V_{CC} = 5V$, $T_A = 25°C$<br>(all inputs and outputs open) | | 11 | mA |
| Input Voltage Levels | | | | |
| CKI Input Levels | | | | |
| Crystal Input | | | | |
| Logic High ($V_{IH}$) | $V_{CC} = 9.5V$ | 3.0 | | V |
| Logic High ($V_{IH}$) | $V_{CC} = 5V \pm 10\%$ | 2.0 | | V |
| Logic Low ($V_{IL}$) | | | 0.4 | V |
| Schmitt Trigger Input (÷4) | | | | |
| Logic High ($V_{IH}$) | | 0.7 $V_{CC}$ | | V |
| Logic Low ($V_{IL}$) | | | 0.6 | V |
| RESET Input Levels | | | | |
| Logic High | | 0.7 $V_{CC}$ | | V |
| Logic Low | | | 0.6 | V |
| RESET Hysteresis | | 1.0 | | V |
| SO Input Level (Test mode) | | 2.0 | 3.0 | V |
| All Other Inputs | | | | |
| Logic High | $V_{CC} = 9.5V$ | 3.0 | | V |
| Logic High | with TTL trip level options | 2.0 | | V |
| Logic Low | selected, $V_{CC} = 5V \pm 10\%$ | | 0.8 | V |
| Logic High | with high trip level options | 3.6 | | V |
| Logic Low | selected | | 1.2 | V |
| Output Voltage Levels | | | | |
| LSTTL Operation | $V_{CC} = 5V \pm 5\%$ | | | |
| Logic High ($V_{OH}$) | $I_{OH} = -25\mu A$ | 2.7 | | V |
| Logic Low ($V_{OL}$) | $I_{OL} = 0.36mA$ | | 0.4 | V |
| CMOS Operation | | | | |
| Logic High | $I_{OH} = -10\mu A$ | $V_{CC} - 1$ | | V |
| Logic Low | $I_{OL} = 10\mu A$ | | 0.2 | V |

**Note 1:** $V_{CC}$ voltage change must be less than 0.5V/ms to maintain proper operation.

## DC Electrical Characteristics (continued) $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Output Current Levels | | | | |
| Output Sink Current | | | | |
| SO and SK Outputs ($I_{OL}$) | $V_{CC}=9.5V$, $V_{OL}=1.0V$ | 4.5 | 22 | mA |
| | $V_{CC}=4.5V$, $V_{OL}=1.0V$ | 2.2 | 11 | mA |
| $L_0$-$L_7$ Outputs and Standard Size $G_0$-$G_3$ and $D_0$-$D_3$ Outputs ($I_{OL}$) | $V_{CC}=9.5V$, $V_{OL}=1.0V$ | 2.0 | 9.0 | mA |
| | $V_{CC}=4.5V$, $V_{OL}=1.0V$ | 1.0 | 4.5 | mA |
| $G_0$-$G_3$ and $D_0$-$D_3$ Outputs with High Current Options ($I_{OL}$) | $V_{CC}=9.5V$, $V_{OL}=1.0V$ | 15 | 75 | mA |
| | $V_{CC}=4.5V$, $V_{OL}=1.0V$ | 7.0 | 35 | mA |
| $G_0$-$G_3$ and $D_0$-$D_3$ Outputs with Very High Current Options ($I_{OL}$) | $V_{CC}=9.5V$, $V_{OL}=1.0V$ | 30 | 150 | mA |
| | $V_{CC}=4.5V$, $V_{OL}=1.0V$ | 15 | 75 | mA |
| Output Source Current: | | | | |
| Standard Configuration, All Outputs ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −70 | −450 | μA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −26 | −190 | μA |
| Push-Pull Configuration, SO and SK Outputs ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −1.45 | −15.5 | mA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −0.07 | −2.8 | mA |
| LED Configuration, $L_0$-$L_7$ Outputs, Low Current Driver Option ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −1.5 | −15 | mA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −1.5 | −9.0 | mA |
| LED Configuration, $L_0$-$L_7$ Outputs, High Current Driver Option ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=2.0V$ | −3.0 | −30 | mA |
| | $V_{CC}=6.0V$, $V_{OH}=2.0V$ | −3.0 | −20 | mA |
| TRI-STATE Configuration, $L_0$-$L_7$ Outputs, Low Current Driver Option ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −2.4 | 24.5 | mA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −0.06 | 3.8 | mA |
| TRI-STATE Configuration, $L_0$-$L_7$ Outputs, High Current Driver Option ($I_{OH}$) | $V_{CC}=9.5V$, $V_{OH}=4.75V$ | −4.9 | −47.5 | mA |
| | $V_{CC}=4.5V$, $V_{OH}=2.25V$ | −0.12 | −8.1 | mA |
| CKO Output | | | | |
| RAM Power Supply Option Power Requirement | $V_R=3.3V$ | | 3.0 | mA |
| TRI-STATE® Output Leakage Current | | −10 | +10 | μA |

## AC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise specified.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Instruction Cycle Time — $t_C$ | | 15 | 40 | μs |
| CKI Using Crystal | | | | |
| Input Frequency — $f_I$ | ÷32 mode | 0.8 | 2.097 | MHz |
| | ÷16 mode | 0.4 | 1.0 | MHz |
| | ÷8 mode | 0.2 | 0.5 | MHz |
| Duty Cycle | | 30 | 55 | % |
| CKI Using External Clock | | | | |
| Input Frequency — $f_I$ | ÷32 mode | 0.8 | 2.097 | MHz |
| | ÷16 mode | 0.4 | 1.0 | MHz |
| | ÷8 mode | 0.2 | 0.5 | MHz |
| Duty Cycle | | 30 | 60 | % |
| Rise Time | $f_I = 2.097$ MHz | | 120 | ns |
| Fall Time | | | 80 | ns |
| CKI Using RC (Option 3 = 3) | $R = 51k\Omega \pm 5\%$, $C = 100pF \pm 10\%$ ÷4 mode | | | |
| Instruction Cycle Time | | 15 | 25 | μs |
| CKO as SYNC Input | | | | |
| $t_{SYNC}$ | | 400 | | ns |
| INPUTS: | | | | |
| $IN_3$-$IN_0$, $G_3$-$G_0$, $L_7$-$L_0$ | | | | |
| $t_{SETUP}$ | | 8 | | μs |
| $t_{HOLD}$ | | 600 | | ns |
| SI | | | | |
| $t_{SETUP}$ | | 2 | | μs |
| $t_{HOLD}$ | | 600 | | ns |
| OUTPUTS: | | | | |
| COP TO CMOS PROPAGATION DELAY | $V_{OH} = 0.7\ V_{CC}$, $V_{OL} = 0.3\ V_{CC}$, $C_L = 50pF$ | | | |
| All Standard Output Configurations | | | | |
| $t_{PD1}$ | | | 6.5 | μs |
| SO, SK Outputs | | | | |
| $t_{PD1}$ (push-pull) | | | 4.0 | μs |
| $t_{PD0}$ | | | 1.2 | μs |
| $D_3$-$D_0$, $G_3$-$G_0$ | | | | |
| $t_{PD0}$ | | | 2.7 | μs |
| $L_7$-$L_0$ | | | | |
| $t_{PD0}$ | | | 2.7 | μs |
| $t_{PD1}$ (standard size push-pull) | | | 3.0 | μs |
| $t_{PD1}$ (high current push-pull) | | | 2.5 | μs |
| $L_7$-$L_0$ LED Direct Drive Outputs | $6.0V \leq V_{CC} \leq 9.5V$, $V_{OH} = 2.0V$ | | | |
| $t_{PD1}$ (standard size) | $C_L = 50pF$ | | 5.0 | μs |
| $t_{PD1}$ (high current) | | | 4.5 | μs |

## AC Electrical Characteristics (continued)

$0°C \leq T_A \leq +70°C$, $4.5V \leq V_{CC} \leq 9.5V$ unless otherwise specified.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| COP to LSTTL Propagation Delay | $V_{CC} = 5V \pm 5\%$, $V_{OH} = 2.7V$<br>$V_{OL} = 0.4V$, $C_L = 50pF$ | | | |
| SO, SK Outputs | | | | |
| $t_{PD1}$ (standard) | | | 5 | μs |
| $t_{PD1}$ (push-pull) | | | 3.5 | μs |
| $t_{PD0}$ | | | 3 | μs |
| $L_7$-$L_0$ Outputs | | | | |
| $t_{PD1}$ (push-pull) | | | 1.5 | μs |
| $L_7$-$L_0$, $G_3$-$G_0$, $D_3$-$D_0$ Outputs | | | | |
| $t_{PD1}$ (standard) | | | 5.0 | μs |
| $t_{PD0}$ | | | 5.0 | μs |

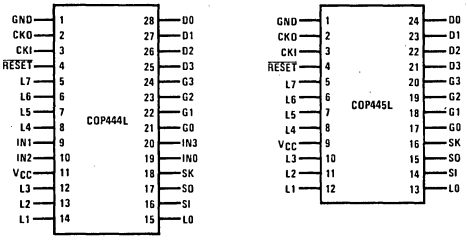



**Figure 2. Connection Diagrams**

| Pin | Description |
|---|---|
| $L_7$-$L_0$ | 8 bidirectional I/O ports with TRI-STATE® |
| $G_3$-$G_0$ | 4 bidirectional I/O ports |
| $D_3$-$D_0$ | 4 general purpose outputs |
| $IN_3$-$IN_0$ | 4 general purpose inputs (COP420L only) |
| SI | Serial input (or counter input) |
| SO | Serial output (or general purpose output) |
| SK | Logic-controlled clock (or general purpose output) |

| Pin | Description |
|---|---|
| CKI | System oscillator input |
| CKO | System oscillator output (or general purpose input, RAM power supply or SYNC input)) |
| $\overline{RESET}$ | System reset input |
| $V_{CC}$ | Power supply |
| GND | Ground |

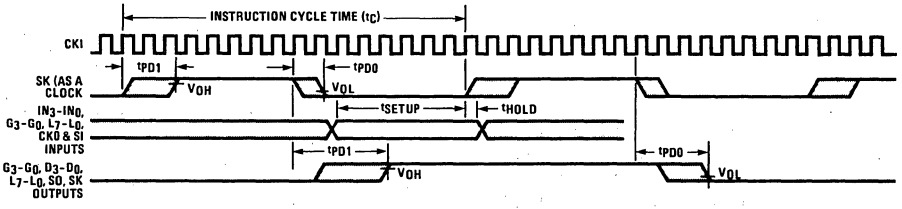



**Figure 3. Input/Output Timing Diagrams (Crystal Divide-by-16 Mode)**

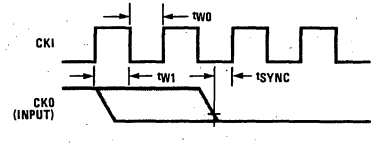



**Figure 3a. Synchronization Timing**

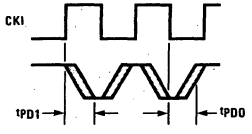



**Figure 3b. CKO Output Timing**

## FUNCTIONAL DESCRIPTION

A block diagram of the COP444L is given in figure 1. Data paths are illustrated in simplified form to depict how the various logic elements communicate with each other in implementing the instruction set of the device. Positive logic is used. When a bit is set, it is a logic "1" (greater than 2 volts). When a bit is reset, it is a logic "0" (less than 0.8 volts).

### Program Memory

Program Memory consists of a 2048 byte ROM. As can be seen by an examination of the COP444L/445L instruction set, these words may be program instructions, program data or ROM addressing data. Because of the special characteristics associated with the JP, JSRP, JID and LQID instructions, ROM must often be thought of as being organized into 32 pages of 64 words each.

ROM addressing is accomplished by a 11-bit PC register. Its binary value selects one of the 2048 8-bit words contained in ROM. A new address is loaded into the PC register during each instruction cycle. Unless the instruction is a transfer of control instruction, the PC register is loaded with the next sequential 11-bit binary count value. Three levels of subroutine nesting are implemented by the 11-bit subroutine save registers, SA, SB and SC, providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the Instruction Decode, Control and Skip Logic circuitry.

### Data Memory

Data memory consists of a 512-bit RAM, organized as 8 data registers of 16 4-bit digits. RAM addressing is implemented by a 7-bit B register whose upper 3 bits (Br) select 1 of 8 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) is usually loaded into or from, or exchanged with, the A register (accumulator), it may also be loaded into or from the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the 7-bit contents of the operand field of these instructions. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

### Internal Logic

The 4-bit A register (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd portions of the B register, to load and input 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

A 4-bit adder performs the arithmetic and logic functions, storing its results in A. It also outputs a carry bit to the 1-bit C register, most often employed to indicate arithmetic overflow. The C register, in conjunction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SK or can enable SK to be a sync clock each instruction cycle time. (See XAS instruction and EN register description, below.)

Four general-purpose inputs, $IN_3$-$IN_0$, are provided.

The D register provides 4 general-purpose outputs and is used as the destination register for the 4-bit contents of Bd. The D outputs can be directly connected to the digits of a multiplexed LED display.

The G register contents are outputs to 4 general-purpose bidirectional I/O ports. G I/O ports can be directly connected to the digits of a multiplexed LED display.

The Q register is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control. (See LEI instruction.)

The 8 L drivers,when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the LED Direct Drive output configuration option) with Q data being outputted to the Sa-Sg and decimal point segments of the display.

The SIO register functions as a 4-bit serial-in/serial-out shift register or as a binary counter depending on the contents of the EN register. (See EN register description, below.) Its contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O by connecting SO to external serial-in/parallel-out shift registers.

The XAS instruction copies C into the SKL latch. In the counter mode, SK is the output of SKL; in the shift register mode, SK outputs SKL ANDed with the clock.

The EN register is an internal 4-bit register loaded under program contol by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register ($EN_3$-$EN_0$).

1. The least significant bit of the enable register, $EN_0$, selects the SIO register as either a 4-bit shift register or a 4-bit binary counter. With $EN_0$ set, SIO is an asynchronous binary counter, *decrementing* its value by one upon each low-going pulse ("1" to "0") ocurring on the SI input. Each pulse must be at least two instruction cycles wide. SK outputs the value of SKL. The SO output is equal to the value of $EN_3$. With $EN_0$ reset, SIO is a serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time. (See 4 below.) The SK output becomes a logic-controlled clock.
2. With $EN_1$ set the $IN_1$ input is enabled as an interrupt input. Immediately following an interrupt, $EN_1$ is reset to disable further interrupts.

3. With $EN_2$ set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting $EN_2$ disables the L drivers, placing the L I/O ports in a high-impedance input state.
4. $EN_3$, in conjunction with $EN_0$, affects the SO output. With $EN_0$ set (binary counter option selected) SO will output the value loaded into $EN_3$. With $EN_0$ reset (serial shift register option selected), setting $EN_3$ enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting $EN_3$ with the serial shift register option selected disables SO as the shift register output; data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0." The table below provides a summary of the modes associated with $EN_3$ and $EN_0$.

c. Upon acknowledgement of an interrupt, the skip logic status is saved and later restored upon popping of the stack. For example, if an interrupt occurs during the execution of ASC (Add with Carry, Skip on Carry) instruction which results in carry, the skip logic status is saved and program control is transferred to the interrupt servicing routine at hex address OFF. At the *end* of the interrupt routine, a RET instruction is executed to "pop" the stack and return program control to the instruction following the original ASC. *At this time*, the skip logic is enabled and skips this instruction because of the previous ASC carry. Subroutines and LQID instructions should not be nested within the interrupt service routine, since their popping the stack will enable any previously saved main program skips, interfering with the orderly execution of the interrupt routine.

**Enable Register Modes — Bits $EN_3$ and $EN_0$**

| $EN_3$ | $EN_0$ | SIO | SI | SO | SK |
|---|---|---|---|---|---|
| 0 | 0 | Shift Register | Input to Shift Register | 0 | If SKL = 1, SK = CLOCK<br>If SKL = 0, SK = 0 |
| 1 | 0 | Shift Register | Input to Shift Register | Serial Out | If SKL = 1, SK = CLOCK<br>If SKL = 0, SK = 0 |
| 0 | 1 | Binary Counter | Input to Binary Counter | 0 | If SKL = 1, SK = 1<br>If SKL = 0, SK = 0 |
| 1 | 1 | Binary Counter | Input to Binary Counter | 1 | If SKL = 1, SK = 1<br>If SKL = 0, SK = 0 |

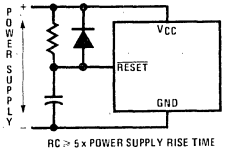
## Interrupt

The following features are associated with the $IN_1$ interrupt procedure and protocol and must be considered by the programmer when utilizing interrupts.

a. The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (PC + 1) onto the stack, pushing in turn the contents of the other subroutine-save registers to the next lower level (PC + 1 → SA → SB → SC). Any previous contents of SC are lost. The program counter is set to hex address OFF (the last word of page 3) and $EN_1$ is reset.
b. An interrupt will be acknowledged only after the following conditions are met:
   1. $EN_1$ has been set.
   2. A low-going pulse ("1" to "0") at least two instruction cycles wide occurs on the $IN_1$ input.
   3. A currently executing instruction has been completed.
   4. All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction, the interrupt will not be acknowledged until the second JP instruction has been executed.
d. The first instruction of the interrupt routine at hex address OFF must be a NOP.
e. A LEI instruction can be put immediately before the RET to re-enable interrupts.

## Initialization

The Reset Logic will initialize (clear) the device upon power-up if the power supply rise time is less than 1ms and greater than 1μs. If the power supply rise time is greater than 1ms, the user must provide an external RC network and diode to the $\overline{RESET}$ pin as shown below. The $\overline{RESET}$ pin is configured as a Schmitt trigger input. If not used it should be connected to $V_{CC}$. Initialization will occur whenever a logic "0" is applied to the $\overline{RESET}$ input, provided it stays low for at least three instruction cycle times.




**Note: If power-low failsafe option is selected, the RC and diode circuit is not used.**

**Power-Up Clear Circuit**

Upon initialization, the PC register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, and G registers are cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. *Data Memory (RAM) is not cleared upon initialization.* The first instruction at address 0 must be a CLRA.

### Oscillator

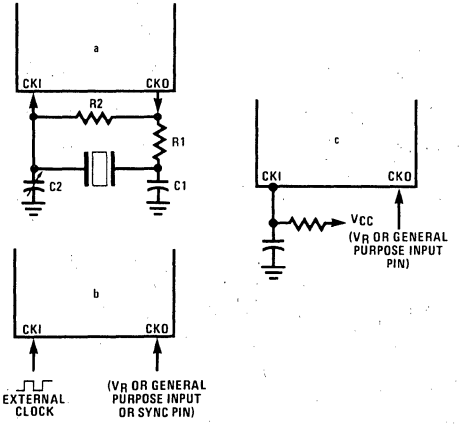
There are four basic clock oscillator configurations available as shown by figure 4.

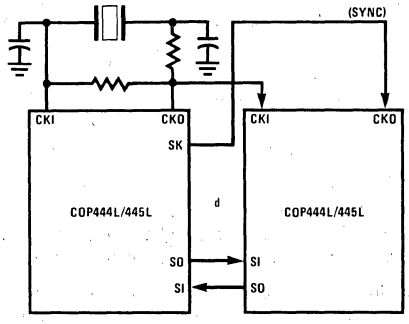
a. **Crystal Controlled Oscillator.** CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency divided by 32 (optional by 16 or 8).

b. **External Oscillator.** CKI is an external clock input signal. The external frequency is divided by 32 (optional by 16 or 8) to give the instruction cycle time. CKO is now available to be used as the RAM power supply ($V_R$), as a general purpose input, or as a SYNC input.

c. **RC Controlled Oscillator.** CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 4. CKO is available as the RAM power supply ($V_R$) or as a general purpose input.

d. **Externally Synchronized Oscillator.** Intended for use in multi-COP systems, CKO is programmed to function as an input connected to the SK output of another COP chip operating at the same frequency (COP chip with L or C suffix) with CKI connected as shown. In this configuration, the SK output connected to CKO must provide a SYNC (instruction cycle) signal to CKO, thereby allowing synchronous data transfer between the COPs using only the SI and SO serial I/O pins in conjunction with the XAS instruction. Note that on power-up SK is automatically enabled as a SYNC output (See Functional Description, Initialization, above).




**Crystal Oscillator**

| Crystal Value | Component Values | | | |
|---|---|---|---|---|
| | R1 | R2 | C1 (pF) | C2 (pF) |
| 455 kHz | 16k | 1M | 80 | 80 |
| 2.097 mHz | 1k | 1M | 56 | 6-36 |

**RC Controlled Oscillator**

| R (kΩ) | C (pF) | Instruction Cycle Time in μs) |
|---|---|---|
| 51 | 100 | 19 ± 15% |
| 82 | 56 | 19 ± 13% |

**Figure 4. COP444L/445L Oscillator**

### CKO Pin Options

In a crystal controlled oscillator system, CKO is used as an output to the crystal network. As an option CKO can be a SYNC input as described above. As another option CKO can be a general purpose input, read into bit 2 of A (accumulator) upon execution of an INIL instruction. As another option, CKO can be a RAM power supply pin ($V_R$), allowing its connection to a standby/backup power supply to maintain the integrity of RAM data with minimum power drain when the main supply is inoperative or shut down to conserve power. Using either option is appropriate in applications where the COP444L/445L system timing configuration does not require use of the CKO pin.

### I/O Options

COP444L/445L outputs have the following optional configurations, illustrated in figure 5:

**a. Standard** — an enhancement mode device to ground in conjunction with a depletion-mode device to $V_{CC}$, compatible with LSTTL and CMOS input requirements. Available on SO, SK, and all D and G outputs.

**b. Open-Drain** — an enhancement-mode device to ground only, allowing external pull-up as required by the user's application. Available on SO, SK, and all D and G outputs.

**c. Push-Pull** — An enhancement-mode device to ground in conjunction with a depletion-mode device paralleled by an enhancement-mode device to $V_{CC}$. This configuration has been provided to allow for fast rise and fall times when driving capacitive loads. Available on SO and SK outputs only.

**d. Standard L** — same as **a.**, but may be disabled. Available on L outputs only.

**e. Open Drain L** — same as **b.**, but may be disabled. Available on L outputs only.

**f. LED Direct Drive** — an enhancement-mode device to ground and to $V_{CC}$, meeting the typical current sourcing requirements of the segments of an LED display. The sourcing device is clamped to limit current flow. These devices may be turned off under program control (See Functional Description, EN Register), placing the outputs in a high-impedance state to provide required LED segment blanking for a multiplexed display. Available on L outputs only.

**g. TRI-STATE® Push-Pull** — an enhancement-mode device to ground and $V_{CC}$. These outputs are TRI-STATE outputs, allowing for connection of these outputs to a data bus shared by other bus drivers. Available on L outputs only.

COP444L/COP445L inputs have the following optional configurations:

**h.** An on-chip depletion load device to $V_{CC}$.

**i.** A Hi-Z input which must be driven to a "1" or "0" by external components.

The above input and output configurations share common enhancement-mode and depletion-mode devices. Specifically, all configurations use one or more of six devices (numbered 1-6, respectively). Minimum and maximum current ($I_{OUT}$ and $V_{OUT}$) curves are given in figure 6 for each of these devices to allow the designer to effectively use these I/O configurations in designing a COP444L/445L system.

The SO, SK outputs can be configured as shown in **a.**, **b.**, or **c.** The D and G outputs can be configured as shown in **a.** or **b.** Note that when inputting data to the G ports, the G outputs should be set to "1." The L outputs can be configured as in **d.**, **e.**, **f.** or **g.**

An important point to remember if using configuration **d.** or **f.** with the L drivers is that even when the L drivers are disabled, the depletion load device will source a small amount of current (see figure 6, device 2); however, when the L-lines are used as inputs, the disabled depletion device can *not* be relied on to source sufficient current to pull an input to logic "1".

### Power-Low Failsafe Option

If this option is selected, an on-chip level detection circuit will force the RESET pin low and reset the chip while the power supply is still within the operating range. Reset will occur with $V_{CC}$ between 4.5 and 7.5 volts, allowing normal system operation between 7.5 and 9.5 volts.

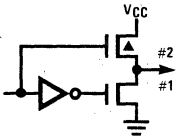
### RAM Keep-Aive Option

Selecting CKO as the RAM power supply ($V_R$) allows the user to shut off the chip power supply ($V_{CC}$) and maintain data in the lower four (r = 0,1,2,3) registers of RAM. To insure that RAM data integrity is maintained, the following conditions *must* be met:

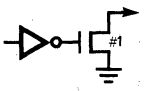
1. RESET must go low before $V_{CC}$ goes low during power off; $V_{CC}$ must go high before RESET goes high on power-up.
2. $V_R$ must be within the operating range of the chip, and equal to $V_{CC} \pm 1V$ during normal operation.
3. $V_R$ must be $\geq 3.3V$ with $V_{CC}$ off.

### COP445L

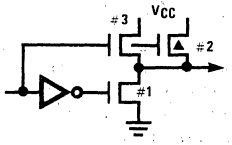
If the COP444L is bonded as a 24-pin device, it becomes the COP445L, illustrated in figure 2, COP444L/445L Connection Diagrams. Note that the COP445L does not contain the four general purpose IN inputs ($IN_3$-$IN_0$). Use of this option precludes, of course, use of the IN options and the interrupt feature, which uses $IN_1$. All other options are available for the COP445L.



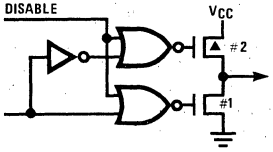

a. Standard Output



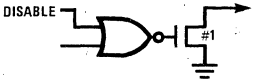

b. Open-Drain Output



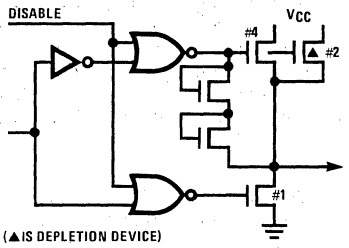

c. Push-Pull Output



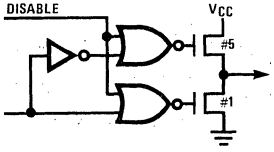

d. Standard L Output



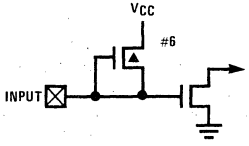

e. Open-Drain L Output



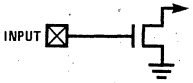

f. LED (L Output)




g. TRI-STATE® Push-Pull (L Output)



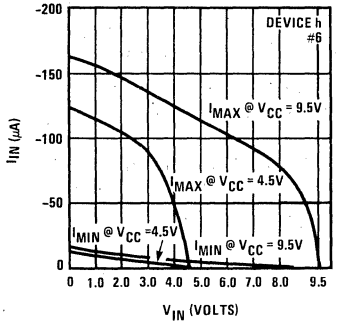

h. Input with Load



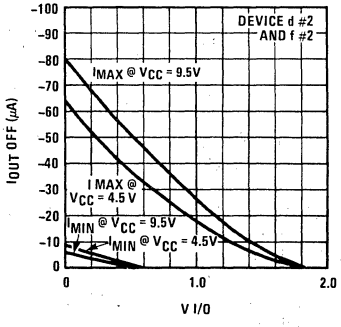

i. Hi-Z Input

Figure 5. Output Configurations

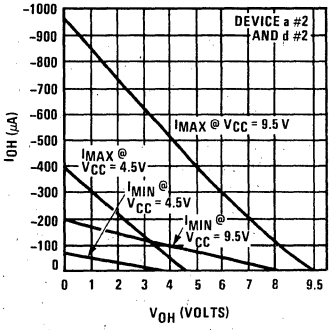
Current for Inputs with Load Device



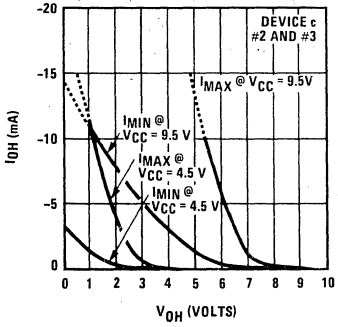

Input Current for $L_0$ through $L_7$ when Output Programmed Off by Software



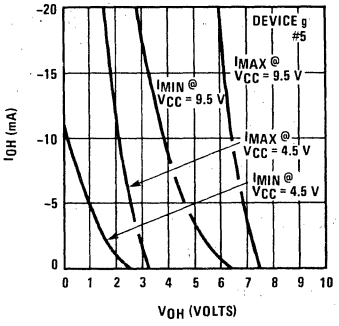

Source Current for Standard Output Configuration



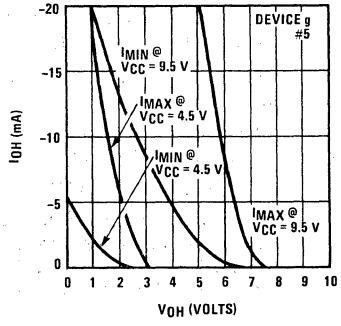

Source Current for SO and SK in Push-Pull Configuration



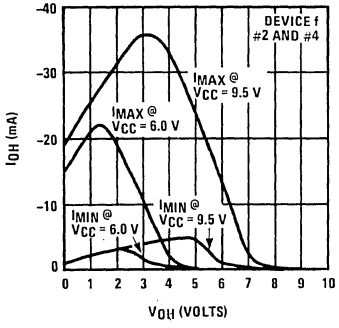

Source Current for $L_0$ through $L_7$ in TRI-STATE® Configuration (High Current Option)



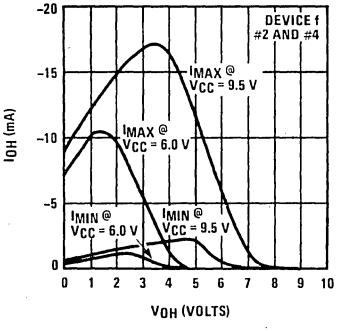

Source Current for $L_0$ through $L_7$ in TRI-STATE® Configuration (Low Current Option)



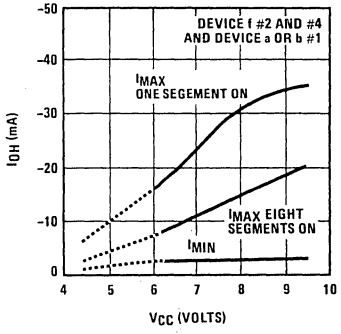
LED Output Source Current
(for High Current LED Option)



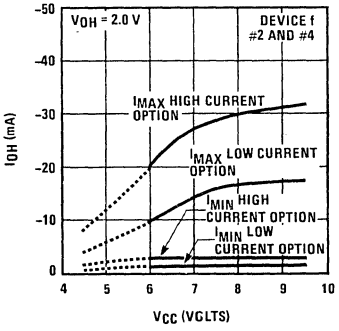

LED Output Source Current
(for Low Current LED Option)



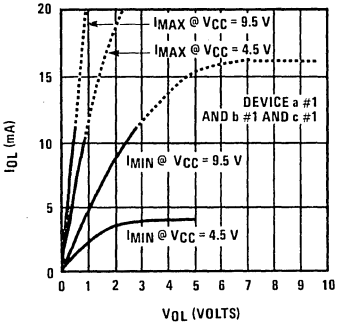

LED Output Direct Segment
Drive
High Current Options on $L_0$-$L_7$
Very High Current Options on
on $D_0$-$D_3$ or $G_0$-$G_3$



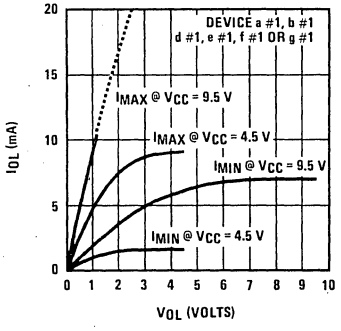

LED Output Direct Segment
Drive



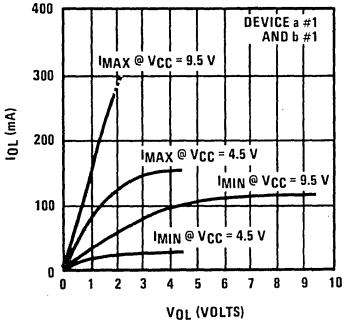

Output Sink Current for SO
and SK




Output Sink Current for $L_0$-$L_7$
and Standard Drive Option for
$D_0$-$D_3$ and $G_0$-$G_3$




Output Sink Current
$G_0$-$G_3$ and $D_0$-$D_3$ with Very
High Current Option




Output Sink Current for $G_0$-$G_3$
and $D_0$-$D_3$ (for High Current
Option)

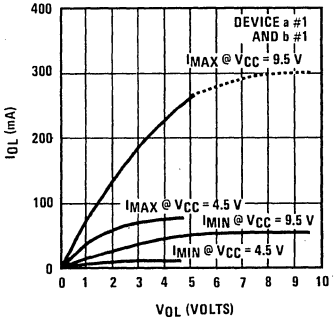



Figure 6. I/O Characteristics

## COP444L/445L INSTRUCTION SET

Table 1 is a symbol table providing internal architecture, instruction operand and operation symbols used in the instruction set table.

Table 2 provides the mnemonic, operand, machine code, data flow, skip conditions and description associated with each instruction in the COP444L/445L instruction set.

**Table 1. COP444L/445L Instruction Set Table Symbols**

| Symbol | Definition |
|---|---|
| **INTERNAL ARCHITECTURE SYMBOLS** | |
| A | 4-bit Accumulator |
| B | 6-bit RAM Address Register |
| Br | Upper 3 bits of B (register address) |
| Bd | Lower 4 bits of B (digit address) |
| C | 1-bit Carry Register |
| D | 4-bit Data Output Port |
| EN | 4-bit Enable Register |
| G | 4-bit Register to latch data for G I/O Port |
| IL | Two 1-bit Latches associated with the $IN_3$ or $IN_0$ Inputs |
| IN | 4-bit Input Port |
| L | 8-bit TRI-STATE I/O Port |
| M | 4-bit contents of RAM Memory pointed to by B Register |
| PC | 11-bit ROM Address Register (program counter) |
| Q | 8-bit Register to latch data for L I/O Port |
| SA | 11-bit Subroutine Save Register A |
| SB | 11-bit Subroutine Save Register B |
| SC | 11-bit Subroutine Save Register C |
| SIO | 4-bit Shift Register and Counter |
| SK | Logic-Controlled Clock Output |

| Symbol | Definition |
|---|---|
| **INSTRUCTION OPERAND SYMBOLS** | |
| d | 4-bit Operand Field, 0-15 binary (RAM Digit Select) |
| r | 3-bit Operand Field, 0-7 binary (RAM Register Select) |
| a | 11-bit Operand Field, 0-2047 binary (ROM Address) |
| y | 4-bit Operand Field, 0-15 binary (Immediate Data) |
| RAM(s) | Contents of RAM location addressed by s |
| ROM(t) | Contents of ROM location addressed by t |
| **OPERATIONAL SYMBOLS** | |
| + | Plus |
| − | Minus |
| → | Replaces |
| ↔ | Is exchanged with |
| = | Is equal to |
| $\overline{A}$ | The ones complement of A |
| ⊕ | Exclusive-OR |
| : | Range of values |

**Table 2. COP444L/445L Instruction Set**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| ARITHMETIC INSTRUCTIONS | | | | | | |
| ASC | | 30 | \|0011\|0000\| | A + C + RAM(B) → A<br>Carry → C | Carry | Add with Carry, Skip on Carry |
| ADD | | 31 | \|0011\|0001\| | A + RAM(B) → A | None | Add RAM to A |
| ADT | | 4A | \|0100\|1010\| | $A + 10_{10} \rightarrow A$ | None | Add Ten to A |
| AISC | y | 5- | \|0101\| y \| | A + y → A | Carry | Add Immediate, Skip on Carry (y ≠ 0) |
| CASC | | 10 | \|0001\|0000\| | $\overline{A}$ + RAM(B) + C → A<br>Carry → C | Carry | Complement and Add with Carry, Skip on Carry |
| CLRA | | 00 | \|0000\|0000\| | 0 → A | None | Clear A |
| COMP | | 40 | \|0100\|0000\| | $\overline{A}$ → A | None | Ones complement of A to A |
| NOP | | 44 | \|0100\|0100\| | None | None | No Operation |
| RC | | 32 | \|0011\|0010\| | "0" → C | None | Reset C |
| SC | | 22 | \|0010\|0010\| | "1" → C | None | Set C |
| XOR | | 02 | \|0000\|0010\| | A ⊕ RAM(B) → A | None | Exclusive-OR RAM with A |
| TRANSFER OF CONTROL INSTRUCTIONS | | | | | | |
| JID | | FF | \|1111\|1111\| | ROM ($PC_{10:8}$, A,M) → $PC_{7:0}$ | None | Jump Indirect (Note 3) |
| JMP | a | 6- | \|0110\|0\|$a_{10:8}$\| | a → PC | None | Jump |
| | | -- | \|$a_{7:0}$\| | | | |
| JP | a | -- | \|1\| $a_{6:0}$ \|<br>(pages 2,3 only)<br>or | a → $PC_{6:0}$ | None | Jump within Page (Note 4) |
| | | -- | \|11\| $a_{5:0}$ \|<br>(all other pages) | a → $PC_{5:0}$ | | |
| JSRP | a | -- | \|10\| $a_{5:0}$ \| | PC + 1 → SA → SB → SC<br>00010 → $PC_{10:6}$<br>a → $PC_{5:0}$ | None | Jump to Subroutine Page (Note 5) |
| JSR | a | 6- | \|0110\|1\|$a_{10:8}$\| | PC + 1 → SA → SB → SC<br>a → PC | None | Jump to Subroutine |
| | | -- | \|$a_{7:0}$\| | | | |
| RET | | 48 | \|0100\|1000\| | SC → SB → SA → PC | None | Return from Subroutine |
| RETSK | | 49 | \|0100\|1001\| | SC → SB → SA → PC | Always Skip on Return | Return from Subroutine then Skip |

Table 2. COP444L/445L Instruction Set (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **MEMORY REFERENCE INSTRUCTIONS** | | | | | | |
| CAMQ | | 33 | 0011\|0011 | $A \rightarrow Q_{7:4}$<br>$RAM(B) \rightarrow Q_{3:0}$ | None | Copy A, RAM to Q |
| | | 3C | 0011\|1100 | | | |
| CQMA | | 33 | 0011\|0011 | $Q_{7:4} \rightarrow RAM(B)$<br>$Q_{3:0} \rightarrow A$ | None | Copy Q to RAM, A |
| | | 2C | 0010\|1100 | | | |
| LD | r | -5 | 00\| r \|0101<br>(r = 0:3) | $RAM(B) \rightarrow A$<br>$Br \oplus r \rightarrow Br$ | None | Load RAM into A,<br>Exclusive-OR Br with r |
| LDD | r,d | 23 | 0010\|0011 | $RAM(r,d) \rightarrow A$ | None | Load A with RAM pointed to directly by r,d |
| | | -- | 0\| r \| d | | | |
| LQID | | BF | 1011\|1111 | $ROM(PC_{10:8},A,M) \rightarrow Q$<br>$SB \rightarrow SC$ | None | Load Q Indirect (Note 3) |
| RMB | 0 | 4C | 0100\|1100 | $0 \rightarrow RAM(B)_0$ | None | Reset RAM Bit |
| | 1 | 45 | 0100\|0101 | $0 \rightarrow RAM(B)_1$ | | |
| | 2 | 42 | 0100\|0010 | $0 \rightarrow RAM(B)_2$ | | |
| | 3 | 43 | 0100\|0011 | $0 \rightarrow RAM(B)_3$ | | |
| SMB | 0 | 4D | 0100\|1101 | $1 \rightarrow RAM(B)_0$ | None | Set RAM Bit |
| | 1 | 47 | 0100\|1101 | $1 \rightarrow RAM(B)_1$ | | |
| | 2 | 46 | 0100\|0110 | $1 \rightarrow RAM(B)_2$ | | |
| | 3 | 4B | 0100\|1011 | $1 \rightarrow RAM(B)_3$ | | |
| STII | y | 7- | 0111\| y | $y \rightarrow RAM(B)$<br>$Bd + 1 \rightarrow Bd$ | None | Store Memory Immediate and Increment Bd |
| X | r | -6 | 00\| r \|0110<br>(r = 0:3) | $RAM(B) \leftrightarrow A$<br>$Br \oplus r \rightarrow Br$ | None | Exchange RAM with A,<br>Exclusive-OR Br with r |
| XAD | r,d | 23 | 0010\|0011 | $RAM(r,d) \leftrightarrow A$ | None | Exchange A with RAM pointed to directly by r,d |
| | | -- | 1\| r \| d | | | |
| XDS | r | -7 | 00\| r \|0111<br>(r = 0:3) | $RAM(B) \leftrightarrow A$<br>$Bd - 1 \rightarrow Bd$<br>$Br \oplus r \rightarrow Br$ | Bd decrements past 0 | Exchange RAM with A and Decrement Bd,<br>Exclusive-OR Br with r |
| XIS | r | -4 | 00\| r \|0100<br>(r = 0:3) | $RAM(B) \leftrightarrow A$<br>$Bd + 1 \rightarrow Bd$<br>$Br \oplus r \rightarrow Br$ | Bd increments past 15 | Exchange RAM with A and Increment Bd,<br>Exclusive-OR Br with r |
| **REGISTER REFERENCE INSTRUCTIONS** | | | | | | |
| CAB | | 50 | 0101\|0000 | $A \rightarrow Bd$ | None | Copy A to Bd |
| CBA | | 4E | 0100\|1110 | $Bd \rightarrow A$ | None | Copy Bd to A |
| LBI | r,d | -- | 00\| r \|(d - 1)<br>(r = 0:3;<br>d = 0, 9:15)<br>or | r,d → B | Skip until not a LBI | Load B Immediate with r,d (Note 6) |
| | | 33 | 0011\|0011 | | | |
| | | -- | 1\| r \| d<br>(any r, any d) | | | |
| LEI | y | 33 | 0011\|0001 | $y \rightarrow EN$ | None | Load EN Immediate (Note 7) |
| | | 6- | 0110\| y | | | |
| XABR | | 12 | 0001\|0010 | $A \leftrightarrow Br$ ($0 \rightarrow A_3$) | None | Exchange A with Br |

Table 2. COP444L/445L Instruction Set (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **TEST INSTRUCTIONS** | | | | | | |
| SKC | | 20 | 0 0 1 0 \| 0 0 0 0 | | C = "1" | Skip if C is True |
| SKE | | 21 | 0 0 1 0 \| 0 0 0 1 | | A = RAM(B) | Skip if A Equals RAM |
| SKGZ | | 33 | 0 0 1 1 \| 0 0 1 1 | | $G_{3:0}$ = 0 | Skip if G is Zero (all 4 bits) |
| | | 21 | 0 0 1 0 \| 0 0 0 1 | | | |
| SKGBZ | | 33 | 0 0 1 1 \| 0 0 1 1 | 1st byte | | Skip if G Bit is Zero |
| | 0 | 01 | 0 0 0 0 \| 0 0 0 1 | 2nd byte | $G_0$ = 0 | |
| | 1 | 11 | 0 0 0 1 \| 0 0 0 1 | | $G_1$ = 0 | |
| | 2 | 03 | 0 0 0 0 \| 0 0 1 1 | | $G_2$ = 0 | |
| | 3 | 13 | 0 0 0 1 \| 0 0 1 1 | | $G_3$ = 0 | |
| SKMBZ | 0 | 01 | 0 0 0 0 \| 0 0 0 1 | | $RAM(B)_0$ = 0 | Skip if RAM Bit is Zero |
| | 1 | 11 | 0 0 0 1 \| 0 0 0 1 | | $RAM(B)_1$ = 0 | |
| | 2 | 03 | 0 0 0 0 \| 0 0 1 1 | | $RAM(B)_2$ = 0 | |
| | 3 | 13 | 0 0 0 1 \| 0 0 1 1 | | $RAM(B)_3$ = 0 | |
| SKT | | 41 | 0 1 0 0 \| 0 0 0 1 | | A time-base counter carry has occurred since last test | Skip on Timer (Note 3) |
| **INPUT/OUTPUT INSTRUCTIONS** | | | | | | |
| ING | | 33 | 0 0 1 1 \| 0 0 1 1 | G → A | None | Input G Ports to A |
| | | 2A | 0 0 1 0 \| 1 0 1 0 | | | |
| ININ | | 33 | 0 0 1 1 \| 0 0 1 1 | IN → A | None | Input IN Inputs to A (Note 2) |
| | | 28 | 0 0 1 0 \| 1 0 0 0 | | | |
| INIL | | 33 | 0 0 1 1 \| 0 0 1 1 | $IL_3$,"1","0",$IL_0$ → A | None | Input IL Latches to A (Note 3) |
| | | 29 | 0 0 1 0 \| 1 0 0 1 | | | |
| INL | | 33 | 0 0 1 1 \| 0 0 1 1 | $L_{7:4}$ → RAM(B) | None | Input L Ports to RAM, A |
| | | 2E | 0 0 1 0 \| 1 1 1 0 | $L_{3:0}$ → A | | |
| OBD | | 33 | 0 0 1 1 \| 0 0 1 1 | Bd → D | None | Output Bd to D Outputs |
| | | 3E | 0 0 1 1 \| 1 1 1 0 | | | |
| OGI | y | 33 | 0 0 1 1 \| 0 0 1 1 | y → G | None | Output to G Ports Immediate |
| | | 5- | 0 1 0 1 \| y | | | |
| OMG | | 33 | 0 0 1 1 \| 0 0 1 1 | RAM(B) → G | None | Output RAM to G Ports |
| | | 3A | 0 0 1 1 \| 1 0 1 0 | | | |
| XAS | | 4F | 0 1 0 0 \| 1 1 1 1 | A ⟷ SIO, C → SKL | None | Exchange A with SIO (Note 3) |

**Note 1:** All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant bit (low-order, right-most bit). For example, $A_3$ indicates the most significant (left-most) bit of the 4-bit A register.

**Note 2:** The ININ instruction is not available on the 24-pin COP445L since this device does not contain the IN inputs.

**Note 3:** For additional information on the operation of the XAS, JID, LQID, INIL, and SKT instructions, see below.

**Note 4:** The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

**Note 5:** A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

**Note 6:** LBI is a single-byte instruction if d = 0, 9, 10, 11, 12, 13, 14, or 15. The machine code for the lower 4 bits equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 ($1001_2$), the lower 4 bits of the LBI instruction equal 8 ($1000_2$). To load 0, the lower 4 bits of the LBI instruction should equal 15 ($1111_2$).

**Note 7:** Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register.)

The following information is provided to assist the user in understanding the operation of several unique instructions and to provide notes useful to programmers in writing COP444L/445L programs.

### XAS Instruction

XAS (Exchange A with SIO) exchanges the 4-bit contents of the accumulator with the 4-bit contents of the SIO register. The contents of SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output. (See Functional Description, EN Register, above.) If SIO is selected as a shift register, an XAS instruction must be performed once every 4 instruction cycles to effect a continuous data stream.

### JID Instruction

JID (Jump Indirect) is an indirect addressing instruction, transferring program control to a new ROM location pointed to indirectly by A and M. It loads the lower 8 bits of the ROM address register PC with the *contents* of ROM addressed by the 11-bit word, $PC_{10:8}$, A, M. $PC_{10}$, $PC_9$ and $PC_8$ are not affected by this instruction.

Note that JID requires 2 instruction cycles.

### INIL Instruction

INIL (Input IL Latches to A) inputs 2 latches, $IL_3$ and $IL_0$ (see figure 7) and CKO into A. The $IL_3$ and $IL_0$ latches are set if a low-going pulse ("1" to "0") has occurred on the $IN_3$ and $IN_0$ inputs since the last INIL instruction, provided the input pulse stays low for at least two instruction times. Execution of an INIL inputs $IL_3$ and $IL_0$ into A3 and A0 respectively, and resets these latches to allow them to respond to subsequent low-going pulses on the $IN_3$ and $IN_0$ lines. If CKO is mask programmed as a general purpose input, an INIL will input the state of CKO into A2. If CKO has not been so programmed, a "1" will be placed in A2. A "0" is always placed in A1 upon the execution of an INIL. The general purpose inputs $IN_3$-$IN_0$ are input to A upon execution of an ININ instruction. (See table 2, ININ instruction.) INIL is useful in recognizing pulses of short duration or pulses which occur too often to be read conveniently by an ININ instruction.

Note: IL latches are not cleared on reset; $IL_3$ and $IL_0$ *not* input on 445L.

### LQID Instruction

LQID (Load Q Indirect) loads the 8-bit Q register with the contents of ROM pointed to by the 11-bit word $PC_{10}$, $PC_9$, $PC_8$, A, M. LQID can be used for table lookup or code conversion such as BCD to seven-segment. The LQID instruction "pushes" the stack (PC + 1 → SA → SB → SC) and replaces the least significant 8 bits of PC as follows: A → $PC_{7:4}$, RAM(B) → $PC_{3:0}$, leaving $PC_{10}$, $PC_9$ and $PC_8$ unchanged. The ROM data pointed to by the new address is fetched and loaded into the Q latches. Next, the stack is "popped" (SC → SB → SA → PC), restoring the saved value of PC to continue sequential program execution. Since LQID pushes SB → SC, the previous contents of SC are lost. Also, when LQID pops the stack, the previously pushed contents of SB are left in SC. The net result is that the contents of SB are placed in SC (SB → SC). Note that LQID takes two instruction cycle times to execute.

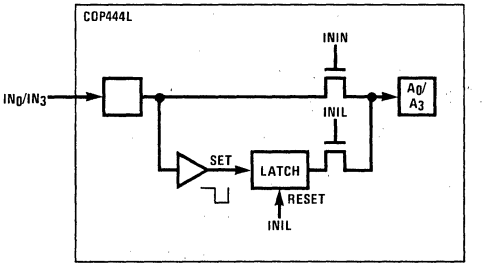



**Figure 7. INIL Hardware Implementation**

### SKT Instruction

The SKT (Skip On Timer) instruction tests the state of an internal 10-bit time-base counter. This counter divides the instruction cycle clock frequency by 1024 and provides a latched indication of counter overflow. The SKT instruction tests this latch, executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction, therefore, allow the COP444L/445L to generate its own time-base for real-time processing rather than relying on an external input signal.

For example, using a 2.097 MHz crystal as the time-base to the clock generator, the instruction cycle clock frequency will be 65kHz (crystal frequency ÷ 32) and the binary counter output pulse frequency will be 64Hz. For time-of-day or similar real-time processing, the SKT instruction can call a routine which increments a "seconds" counter every 64 ticks.

### Instruction Set Notes

a. The first word of a COP444L/445L program (ROM address 0) must be a CLRA (Clear A) instruction.

b. Although skipped instructions are not executed, one instruction cycle time is devoted to skipping each byte of the skipped instruction. Thus all program paths take the same number of cycle times whether instructions are skipped or executed.

c. The ROM is organized into 32 pages of 64 words each. The Program Counter is an 11-bit binary counter, and will count through page boundaries. If a JP, JSRP, JID or LQID instruction is located in the last word of a page, the instruction operates as if it were in the next page. For example: a JP located in the last word of a page will jump to a location in the next page. Also, a LQID or JID located in the last word of page 3, 7, 11, 15, 19, 23 or 27 will access data in the next group of four pages.

## OPTION LIST

The COP444L/445L mask-programmable options are assigned numbers which correspond with the COP*444L* pins.

The following is a list of COP444L options. When specifying a COP445L chip, Options 9, 10, 19, and 20 must all be set to zero. The options are programmed at the same time as the ROM pattern to provide the user with the hardware flexibility to interface to various I/O components using little or no external circuitry.

Option 1 = 0: Ground Pin — no options available

Option 2: CKO Output
- = 0: clock generator output to crystal/resonator (0 not allowable value if option3 = 3)
- = 1: pin is RAM power supply ($V_R$) input
- = 2: general purpose input, load device to $V_{CC}$
- = 3: general purpose input, high-Z
- = 4: multi-COP SYNC input (CKI ÷ 32, CKI ÷ 16)
- = 5: multi-COP SYNC input (CKI ÷ 8)

Option 3: CKI Input
- = 0: oscillator input divided by 32 (2 MHz max)
- = 1: oscillator input divided by 16 (1 MHz max)
- = 2: oscillator input divided by 8 (500 kHz max)
- = 3: single-pin RC controlled oscillator divided by 4
- = 4: oscillator input divide by 4 (Schmitt)

Option 4: $\overline{\text{RESET}}$ Input
- = 0: load device to $V_{CC}$
- = 1: Hi-Z input

Option 5: $L_7$ Driver
- = 0: Standard output
- = 1: Open-drain output
- = 2: High current LED direct segment drive output
- = 3: High current TRI-STATE® push-pull output
- = 4: Low-current LED direct segment drive output
- = 5: Low-current TRI-STATE® push-pull output

Option 6: $L_6$ Driver
same as Option 5

Option 7: $L_5$ Driver
same as Option 5

Option 8: $L_4$ Driver
same as Option 5

Option 9: $IN_1$ Input
- = 0: load device to $V_{CC}$
- = 1: Hi-Z input

Option 10: $IN_2$ Input
same as Option 9

Option 11: $V_{CC}$ pin
- = 0: 4.5V to 6.3V operation
- = 1: 4.5V to 9.5V operation

Option 12: $L_3$ Driver
same as Option 5

Option 13: $L_2$ Driver
same as Option 5

Option 14: $L_1$ Driver
same as Option 5

Option 15: $L_0$ Driver
same as Option 5

Option 16: SI Input
same as Option 9

Option 17: SO Driver
- = 0: standard output
- = 1: open-drain output
- = 2: push-pull output

Option 18: SK Driver
same as Option 17

Option 19: $IN_0$ Input
same as Option 9

Option 20: $IN_3$ Input
same as Option 9

Option 21: $G_0$ I/O Port
- = 0: very-high current standard output
- = 1: very-high current open-drain output
- = 2: high current standard output
- = 3: high current open-drain output
- = 4: standard LSTTL output (fanout = 1)
- = 5: open-drain LSTTL output (fanout = 1)

Option 22: $G_1$ I/O Port
same as Option 21

Option 23: $G_2$ I/O Port
same as Option 21

Option 24: $G_3$ I/O Port
same as Option 21

Option 25: $D_3$ Output
same as Option 21

Option 26: $D_2$ Output
same as Option 21

Option 27: $D_1$ Output
same as Option 21

Option 28: $D_0$ Output
same as Option 21

Option 29: L Input Levels
- = 0: standard TTL input levels ("0" = 0.8V, "1" = 2.0V)
- = 1: higher voltage input levels ("0" = 1.2V, "1" = 3.6V)

Option 30: IN Input Levels
same as Option 29

Option 31: G Input Levels
same as Option 29

Option 32: SI Input Levels
same as Option 29

Option 33: RESET Input
- = 0: Schmitt trigger input
- = 1: standard TTL input levels
- = 2: higher voltage input levels

Option 34: CKO Input Levels (CKO = input; Option 2 = 2,3)
same as Option 29

Option 35 COP Bonding
- = 0: COP444L (28-pin device)
- = 1: COP445L (24-pin device)

Option 36: Power-Low Failsafe
- = 0: normal operation (with $\overline{\text{RESET}}$)
- = 1: power-low failsafe enabled

### TEST MODE (Non-Standard Operation)

The SO output has been configured to provide for standard test procedures for the custom-programmed COP444L. With SO forced to logic "1," two test modes are provided, depending upon the value of SI:

a. RAM and Internal Logic Test Mode (SI = 1)

b. ROM Test Mode (SI = 0)

These special test modes should not be employed by the user; they are intended for manufacturing test only.

### APPLICATION #1: COP444L General Controller

Figure 8 shows an interconnect diagram for a COP444L used as a general controller. Operation of the system is as follows:

1. The $L_7$-$L_0$ outputs are configured as LED Direct Drive outputs, allowing direct connection to the segments of the display.
2. The $D_3$-$D_0$ outputs drive the digits of the multiplexed display directly and scan the columns of the 4 x 4 keyboard matrix.
3. The $IN_3$-$IN_0$ inputs are used to input the 4 rows of the keyboard matrix. Reading the IN lines in conjunction with the current value of the D outputs allows detection, debouncing, and decoding of any one of the 16 keyswitches.
4. CKI is configured as a single-pin oscillator input allowing system timing to be controlled by a single-pin RC network. CKO is therefore available for use as a general-purpose input.
5. SI is selected as the input to a binary counter input. With SIO used as a binary counter, SO and SK can be used as general purpose outputs.
6. The 4 bidirectional G I/O ports ($G_3$-$G_0$) are available for use as required by the user's application.
7. Normal reset operation is selected.

## COP444L Evaluation

The 444L-EVAL is a pre-programmed COP444L, containing several routines which facilitate user familiarization and evaluation of the COP444L operating characteristics. It may be used as an up/down counter or timer, interfacing to any combination of (1) an LED digit or lamps, (2) 4-digit LED Display Controller, (3) a 4-digit VF Display Controller, and/or (4) a 4-digit LCD Display Controller; alternatively, it may be used as a simple music synthesizer.

### Sample Circuits

1. By making only the oscillator, power supply and "L7" connections, (Fig. 9) an approximate 1Hz square wave will be produced at output "D1." This output may be observed with an oscilloscope, or connected to additional TTL or CMOS circuitry.
2. By making the indicated connections to a small LED digit (NSA1541A, NSA1166, or equiv. — larger digits will be proportionately dimmer), the counter actions may be observed. Place the "up/down" switch in the "up" (open) position and apply a TTL-compatible signal at the "counter-input." Placing the "up/down" switch in the "down" (closed) position causes the count to decrement on each high-to-low input transition.
3. All 4 digits of the counter may be displayed by connecting a standard display controller (COP470 for VF, COP472 for LCD, MM5450 for LED) as shown in Fig. 9.

   Any combination of the single LED digit and display controllers may be used simultaneously, and will display the same data.
4. The simple counter described above becomes a timer when the 1Hz output is connected to the "counter input." Up or down counting may be used with input frequencies up to 1kHz. Improved

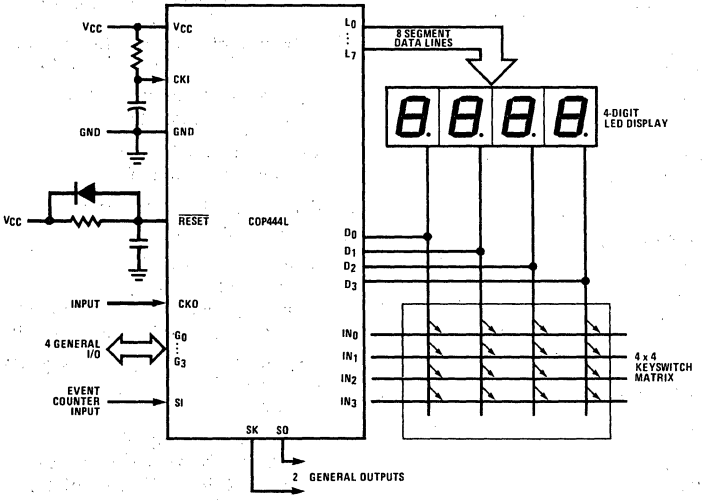



**Figure 8. COP444L Keyboard/Display Interface**

timing accuracies may be obtained by substituting the 2.097MHz crystal oscillator circuit of Figure 4a for the RC Network shown in Figure 9, or by connecting a more stable external frequency to the "counter input" in place of the 1Hz signal.

5. An "entertaining" use of the 444L-EVAL is as a simple music synthesizer (or electronic organ). By attaching a simple switch matrix (or keyboard), a speaker or piezo-ceramic transducer, and grounding "L7", the user can play "music" (Figure 10). Three modes of operation are available: Play a note, play one of four stored tunes, or record a tune for subsequent replay.

   a . *Play A Note*

   Twelve keys, representing the 12 notes in one octave, are labeled "C" through "B"; depressing a key causes a square wave of the corresponding frequency to be output to the speaker. Depressing "L Shift" or "U Shift" causes the next note to be shifted to the next lower octave (one-half frequency) or the next upper octave (double frequency), respectively.

   b . *Play Stored Tune*

   Depressing "Play" followed by "1/8", "1/4", "1/2", or "1" will cause one of 4 stored tunes to be played.

   c . *Record Tune*

   Any combination of notes and rests up to a total of 48 may be stored in RAM for later replay. To store a note, press the appropriate note key, followed by the duration of the note (1/8-note, 1/4-note, 1/2-note, whole (1)-note, followed by "Store;" a rest is stored by selecting the duration and pressing "Store." When the tune is complete, press "Play" followed by "Store;" the tune will be played for immediate audition. Subsequent depression of "Play" and "Store" will replay the last stored tune.

Note: The accuracy of the tones produced are a function of the oscillator accuracy and stability; the crystal oscillator is recommended.

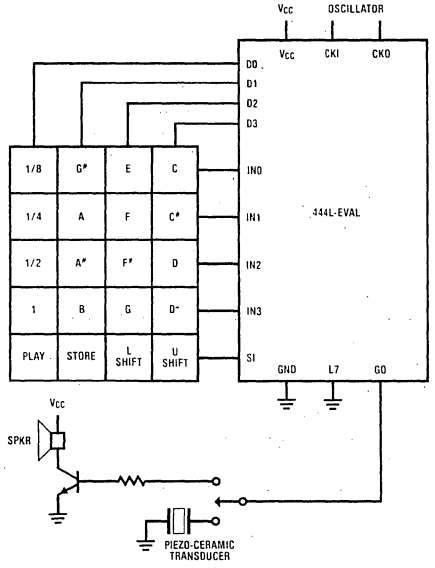



**Figure 10. Music Synthesizer**

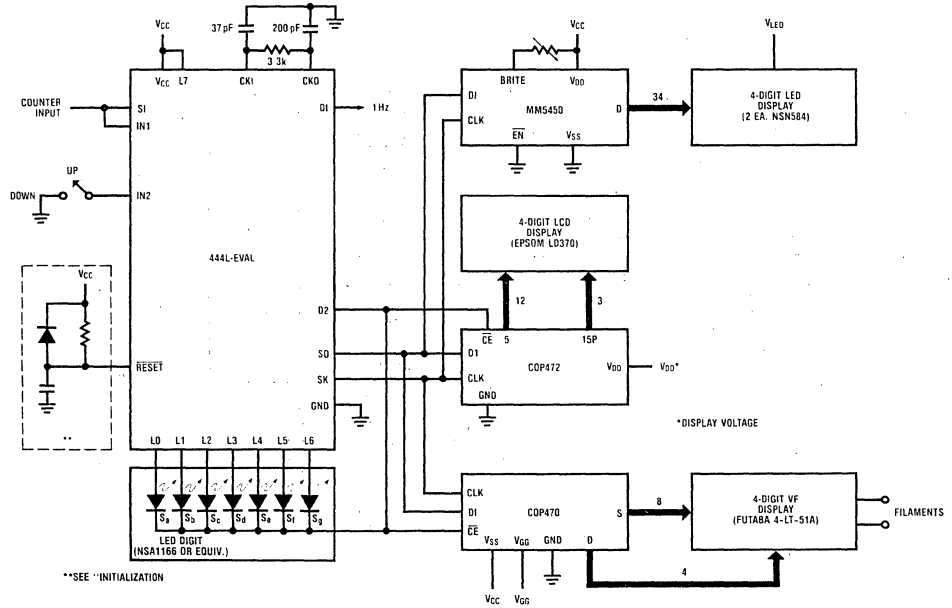



**Figure 9. Counter/Timer**

National Semiconductor

COPS™ Peripheral
Product Brief
PRELIMINARY

# COP430/COP431/COP432 A to D Converters

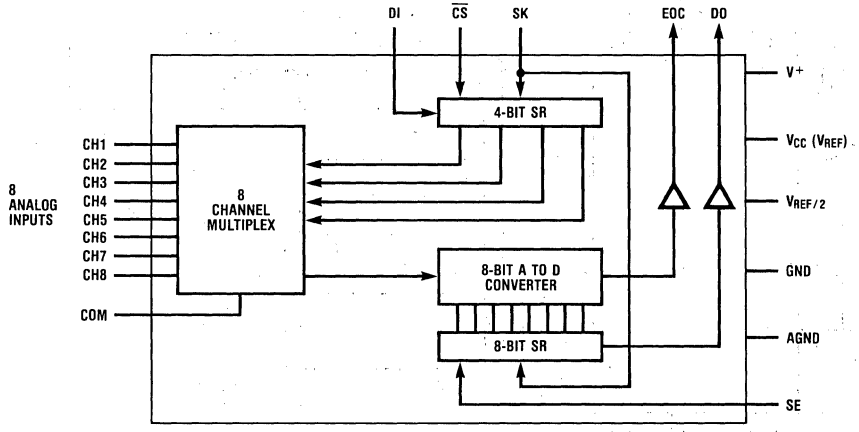
## General Description

The COP430, COP431 and COP432 A to D Converters are peripheral members of the COPS™ family, fabricated using CMOS technology. They include an 8-bit A to D converter, up to an 8 channel multiplexer and COP400 MICROWIRE™ compatible serial I/O. In the COP430, 8 channel multiplexer can be configured to directly access one of eight single-ended analog signals. It can also be configured to directly access up to 4 pairs of differential analog inputs which could be used to measure ratiometric transducers such as potentiometers, thermistor bridges, strain gauges, etc. The COP A to D device is available with an accuracy of 8 bits ± ¼ LSB. Also available is a lower cost version with an accuracy of 6 bits (±40mV error with 5.00V full scale). Full scale can be either $V_{CC}$ or twice $V_{REF/2}$.

The COP431 is a 14-pin version of the converter, and allows only 4 single-ended (or 2 differential) inputs. The COP432 is the lowest-cost version, packaged in an 8-pin mini-DIP. It allows a single differential input.

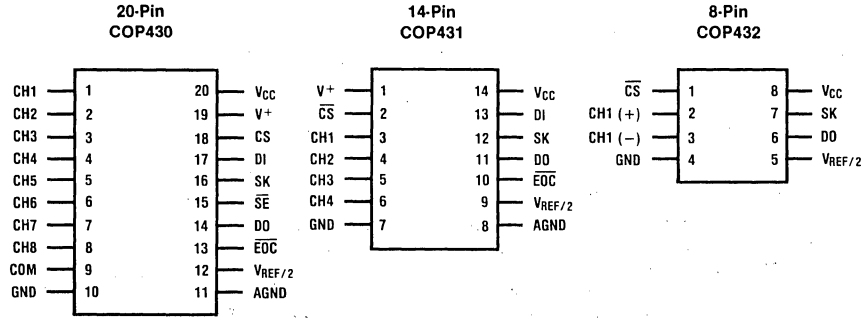
## Features

- 8-bit A to D Conversion
- Compatible with all COP400 processors
- 8-bit accuracy available
- Up to 8 analog inputs
- Single ended or differential inputs
- Low power dissipation
- Low cost 8, 14, 20 pin dual-in-line package
- Fast conversion time (25μsec)
- On chip timing for A to D conversion
- Interfaces directly to 9.5V COP controllers
- MICROWIRE compatible serial I/O
- Operates ratiometric or absolute with precision reference

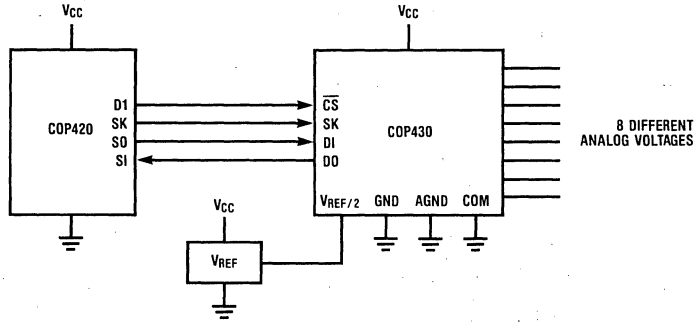



**COP430 Block Diagram**

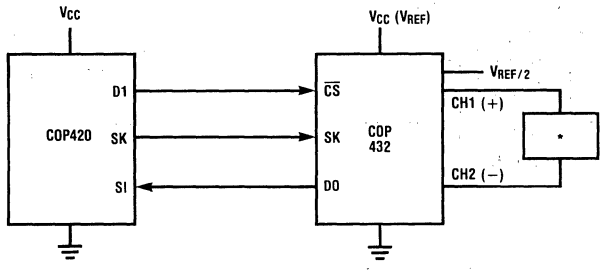
The COP A to D device is available in three different packages: 8 pin, 14 pin, or 20 pin depending on the number of channels required.




| Pin | Description | Pin | Description |
|---|---|---|---|
| $V_{CC}$ | Positive power supply 4.5 to 5.5V | $\overline{EOC}$ | End of conversion output |
| $V^+$ | Positive supply if greater than 5.5V | $\overline{SE}$ | Shift enable input |
| GND | Ground (most negative supply) | DI | Serial data input |
| AGND | Analog Ground | DO | Serial data output |
| CH1~CH8 | Analog voltage inputs | SK | Serial clock input |
| COM | Common input to 8 channels | $\overline{CS}$ | Chip select input |
| $V_{REF/2}$ | Voltage reference input | | |




System Diagram




*RATIOMETRIC TRANSDUCER OR A SINGLE VOLTAGE INPUT

Low Cost System Using 8-Pin Version

# COP450/COP451 PROM-RAM INTERFACE CHIP

## General Description

The COP450 and COP451 are peripheral members of the COPS™ family, fabricated using CMOS technology.

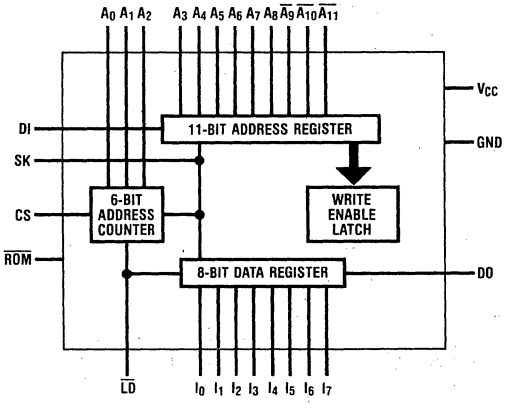
The COP450 is in a 28-lead package and allows any COPS microcontroller to communicate with PROM, ROM, or RAM through its MICROWIRE™ serial I/O port. The COP451 is a 20-lead version of the COP450 and allows a COPS microcontroller to communicate with RAM. Control pin $\overline{ROM}$ determines whether the COP450 interfaces with a RAM or a PROM/ROM. ($\overline{ROM}$ is internally bonded to $V_{CC}$ in the COP451.)

Address and Read/Write commands are entered serially into DI clocked by SK when CS is high. Data is read and written in 64-bit groups. A write enable latch — not reset by CS — is set or reset by a write enable or write disable command, so that incorrect data at SK and DI (which might occur while the COPS microcontroller is powering up or down) cannot change data stored in RAM.

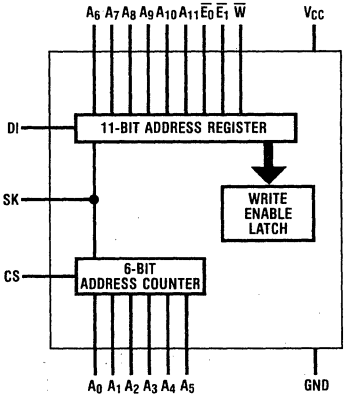
The COP450/COP451 and a CMOS RAM may be used for low power backup memory in a COPS system. The COP450 and a ROM or PROM may be used for large look-up tables.

## Features

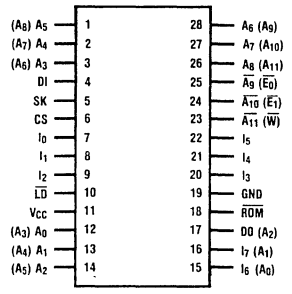
- Interfaces directly with standard ROMs, PROMs, and RAMs (NMOS, CMOS, PMOS)
- Low power
- Low cost
- Directly interfaces up to 8k RAM or 32k ROM
- Single supply operation (2.5V-6V)
- Protects external RAM data when processor power turned off
- Compatible with all COP400 processors (processor $V_{CC} \leq 9.5V$)
- Small (20-pin) dual-in-line package for COP451
- MICROWIRE compatible serial I/O




COP450 Block Diagram

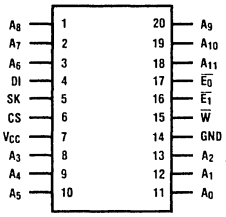



COP451 Block Diagram




**COP450**

(Pin names in parenthesis are used if the COP450 is accessing RAM)




**COP451**

| Pin | Description | | |
|---|---|---|---|
| $V_{CC}$ | Positive power supply | CS | Chip select input |
| GND | Negative supply pin | $A_0 \sim A_{11}$ | Memory address outputs |
| $\overline{ROM}$ | ROM operation select input | $I_0 \sim I_7$ | ROM data inputs |
| DI | Serial data input | $\overline{LD}$ | Dynamic ROM address load output |
| SK | Serial clock input | $\overline{E}_0$, $\overline{E}_1$ | Enable RAM outputs |
| DO | Serial data output | $\overline{W}$ | Write RAM output |

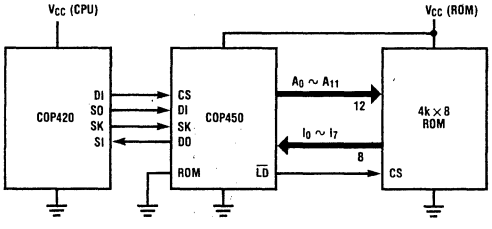
**Instruction Set**

($\overline{ROM} = V_{CC}$)

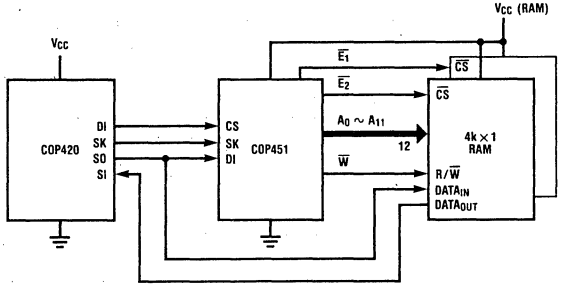
| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 1 | $E_1$ | $E_0$ | $A_{11}$ | $A_{10}$ | $A_9$ | $A_8$ | $A_7$ | $A_6$ | Write to RAM |
| 0 | 1 | 1 | 0 | $E_1$ | $E_0$ | $A_{11}$ | $A_{10}$ | $A_9$ | $A_8$ | $A_7$ | $A_6$ | Read RAM |
| 0 | 1 | 0 | 1 | X | X | X | X | X | X | X | X | Write Enable |
| 0 | 1 | 0 | 0 | X | X | X | X | X | X | X | X | Write Disable (Protect) |

($\overline{ROM} = GND$)

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | $A_{11}$ | $A_{10}$ | $A_9$ | $A_8$ | $A_7$ | $A_6$ | $A_5$ | $A_4$ | $A_3$ | Read from ROM |




**COP450 System Diagram**




**COP451 System Diagram**

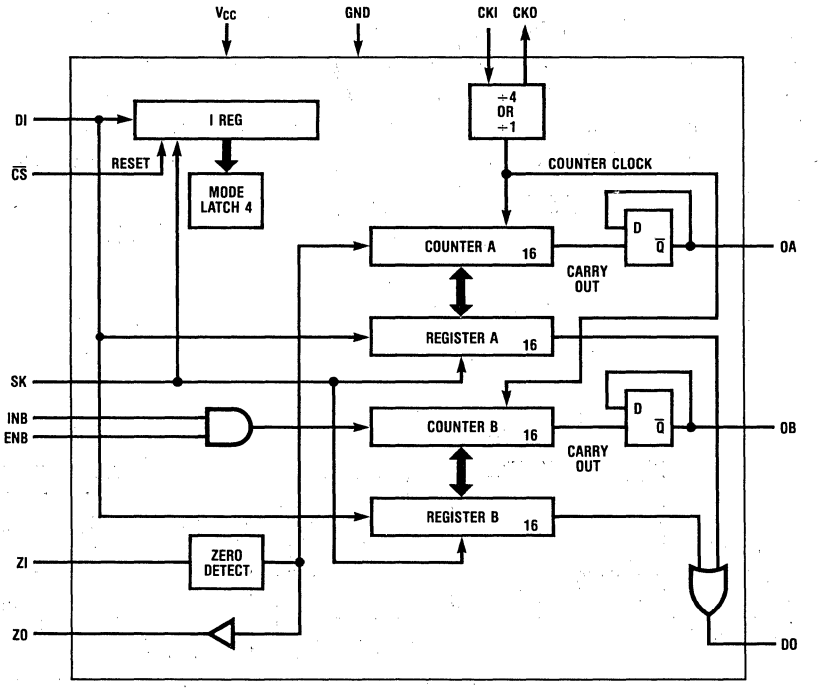
# COP452 Frequency/Counter Peripheral

## General Description

The COP452 is a peripheral member of the COPS™ family fabricated using N-channel silicon-gate MOS technology. Containing 2 independent 16-bit counter/register pairs, it is well suited to a wide variety of tasks involving the measurement and/or generation of times and/or frequencies. Included are multiple tones, precise duty cycles, event counting, waveform measurement, "white noise" generation, and A-D/D-A conversions. An on-chip zero-crossing detector can trigger a pulse with a programmed delay and duration.

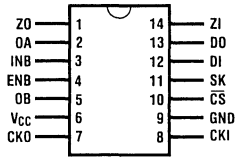
## Features

- Compatible with all COP400 processors
- MICROWIRE™ compatible serial I/O
- 14-pin package
- Single supply operation (4.5-6.3V)
- Low Cost
- Crystal or external clock (25 kHz to 4.4 MHz)
- TTL compatible
- User programmable
- True zero crossing detect
- 17 stage pseudo random white noise generator
- Wider supply range (4.5-9.5V) optionally available
- Extended temperature range device to be available (−40°C to +85°C)




COP452 Block Diagram

**COP452**
**Connection Diagram**




| Pin | Description |
|---|---|
| DI | Serial data input |
| SK | Serial clock input |
| $\overline{CS}$ | Chip select input |
| DO | Serial data output |
| $V_{CC}$ | Power supply |
| GND | Ground |
| CKI | Crystal input |
| CKO | Crystal output |
| OA | Output from counter A |
| OB | Output from counter B |
| INB | External input to counter B |
| ENB | Enable input INB |
| ZI | AC waveform input |
| ZO | Square wave output of ZI |

**COP452 Instruction Set**

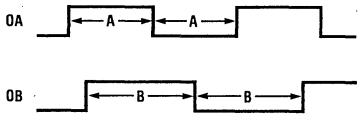
| | | |
|---|---|---|
| 0 0 0 0 0 | LDRB | Load Register B from DI |
| 0 0 0 0 1 | LDRA | Load Register A from DI |
| 0 0 0 1 0 | RDRB | Read Register B |
| 0 0 0 1 1 | RDRA | Read Register A |
| 0 0 1 0 0 | LDCB | Load Counter B from Register B |
| 0 0 1 0 1 | LDCA | Load Counter A from Register A |
| 0 0 1 1 0 | RDCB | Read Counter B |
| 0 0 1 1 1 | RDCA | Read Counter A |
| 0 1 0 0 0 | CK4 | CKI Divide By Four |
| 0 1 0 0 1 | CK1 | CKI Divide By One |
| 1 X X X X | LDM | Load Mode Latches |

## Mode Description

- Dual Frequency

  OA outputs a square wave of width A

  OB outputs a square wave of width B




  A = contents Counter A
  B = contents Counter B

- Frequency and Count

  OA outputs a square wave of width A

  Counter B counts external pulses on INB

- Dual Count

  Counter A counts pulses on ZI

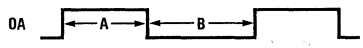
  Counter B counts pulses on INB

- Number of Pulses

  OA outputs a square wave of width A for B number of pulses

- Duty Cycle

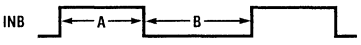
  OA outputs a duty cycle wave form of width high = A and width low = B




- Pulse Measurement

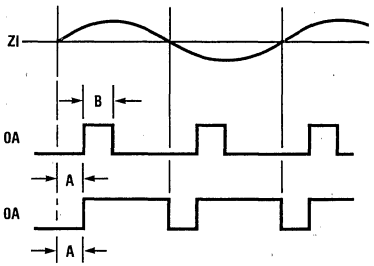
  Counter A counts the pulse width high on INB

  Counter B counts the pulse width low on INB




- Triggered Pulse

  OA outputs a pulse of width B triggered by ZI crossing zero delayed by A.




- Triggered Pulse and Count

  OA outputs a pulse triggered by ZI delayed by A.

  Counter B counts INB

- White Noise and Frequency

  OA outputs white noise

  OB outputs a square wave of width B

- White Noise and Duration

  OA outputs white noise for duration B

- RESET

National Semiconductor

# COP470 V.F. Display Driver

## General Description

The COP470 is a peripheral member of National's COPS™ Microcontroller family. It is designed to directly drive a multiplexed Vacuum Fluorescent display. Data is loaded serially and held in internal latches. The COP470 has an on-chip oscillator to multiplex four digits of eight segment display, and may be cascaded and/or stacked to drive more digits, more segments, or both.

With the addition of external drivers, the COP470 also provides a convenient means of interfacing to a large-digit LED display.

## Features

- Directly interfaces to multiplexed 4 digit by 8 segment Vacuum Fluorescent displays
- Expandable to drive 8 digits and/or 16 segments
- Compatible with all COP400 processors
- Needs no refresh from processor
- Internal or external oscillator
- No "glitches" on outputs when loading data
- Drives large and small displays
- Programmable display brightness
- Small (20 pin) dual-in-line package
- Operates from 4.5V to 9.5V
- Outputs switch 35 volts and require no external resistors
- Static latches
- Microwire™ compatible serial I/O

## Connection and Block Diagrams

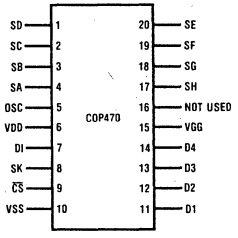



**Figure 1. COP470 Pin Connection**

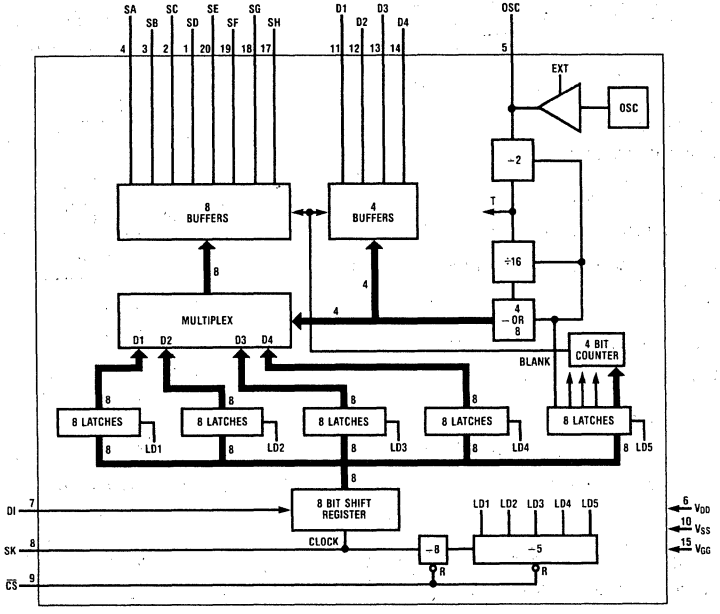



**Figure 2. COP470 Block Diagram**

## Absolute Maximum Ratings ($V_{SS} = 0$)

Voltage at Display Outputs ..... +0.3V to −35V
Voltage at All Other Pins ..... +0.3V to −20V
Operating Temperature ..... 0°C to +70°C
Storage Temperature ..... −65°C to +150°C
Lead Temperature (10 Seconds) ..... 300°C

## Electrical Characteristics

$V_{SS} = 0$, $V_{DD} = -4.5V$ to $-9.5V$, $V_{GG} = -30V$ to $-35V$, $T_A = 0$ to 70°C unless otherwise specified.

## D. C. Electrical Specifications

| | Min. | Max. | Unit |
|---|---|---|---|
| Power Supply Voltage | | | |
| $V_{DD}$ | −9.5 | −4.5 | Volts |
| $V_{GG}$ | −35 | $V_{DD}$ | Volts |
| Power Supply Current | | | |
| $I_{DD}$ | | 5 | mA |
| $I_{GG}$ (Display Blanked) | | 1 | mA |
| Input Levels | | | |
| $V_{IH}$ | −1.5 | +0.3 | V |
| $V_{IL}$ | −10.0 | −4.0 | V |
| Output Drive | | | |
| $I_{OH}$ @ $V_{OH} = V_{SS} - 3V$ | 10 | | mA |
| $I_{OH}$ @ $V_{OH} = V_{SS} - 2V$ | 7 | | mA |
| $I_{OL}$ @ $V_{OL} = V_{GG} + 2V$ | 10 | | μA |
| Output Drive @ $V_{GG} = V_{DD} = V_{SS} - 5V$ | | | |
| $I_{OH}$ @ $V_{OH} = V_{SS} - 2V$ | 1 | | mA |

## A. C. Electrical Specifications

| | | | |
|---|---|---|---|
| OSC Period (internal or external) | 4 | 20 | μSec |
| OSC Pulse Width | 1.5 | | μSec |
| Clock Period T (twice OSC period) | 8 | 40 | μSec |
| Display Frequency | | | |
| 4 digits = 1/64T | 390 | 2000 | Hz |
| 8 digits = 1/128T | 190 | 1000 | Hz |
| SK Clock Frequency | 0 | 250 | kHz |
| SK Clock Width | 1.5 | | μSec |
| Data Set-up and Hold Time | | | |
| t set-up | 1.0 | | μSec |
| t hold | 50 | | nSec |
| CS Set-up and Hold Time | | | |
| t set-up | 1.0 | | μSec |
| t hold | 1.0 | | μSec |
| Duty Cycle | | | |
| 4 digits | 1/64 | 15/64 | |
| 8 digits | 1/128 | 15/128 | |
| Input Capacitance | | 7 | pF |
| Input Leakage | | 1 | μA |

## Timing Diagram

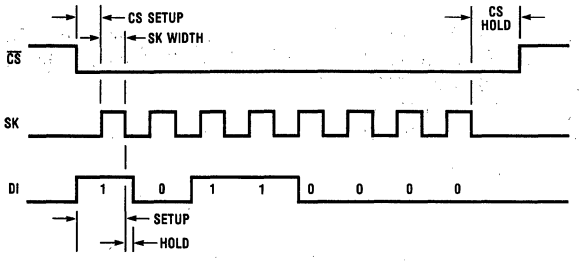
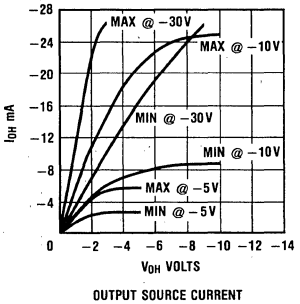



Figure 3. Serial Load Timing Diagram

## Typical Performance Characteristics




## Functional Description

### Segment Data Bits

Data is loaded in serially in sets. Each set of segment data is in the following format:

| SA | SB | SC | SD | SE | SF | SG | SH |
|---|---|---|---|---|---|---|---|

Data is shifted into an eight bit shift register. The first bit of the data is for segment H, digit 1. The eighth bit is segment A, digit 1.

A set of eight bits is shifted in and then loaded into the digit one latches. The second set of 8 bits is loaded into digit two latches. The third set into digit three latches and the fourth set is loaded into digit four latches.

### Display on Time and Control Bits

The fifth set of 8 data bits contains blank time data and control data in the following format:

Display Digits

Sync Ext. Right Left ← On Time →

| | Osc. | 4 of 8 | 4 of 8 | LSB | | | MSB |
|---|---|---|---|---|---|---|---|

the first four bits shifted in contain the on time. This is used to control display brightness. The brightness is a function of the on time of each segment divided by the total time (duty cycle). The on time is programmable from 0 to 15 and the total time is 64. For example, if the on time is 15, the duty cycle is 15/64 which is maximum brightness. If on time is 8, the duty cycle is 8/64, about 1/2 brightness. There are 16 levels of brightness from 15/64 to 0/64 (off).

The fifth and sixth bits control the multiplex digits. To enable the COP470 to drive a 4 digit multiplex display, set both bits to one. If two COP470s are used to drive an 8 digit display, bit five is set on the left COP470 and bit six is set on the right COP470 (see Fig. 6). In the eight digit mode, the display duty cycle is on time/128.

The seventh bit selects internal or external oscillator. The OSC pin of the COP470 is either an output of the internal oscillator (bit 7 = 0) or is an input allowing the COP470 to run from an external oscillator (bit 7 = 1).

The eighth bit is set to synchronize two COP470s. For example, to set the COP470 to internal osc, 4 digits, and maximum brightness, send out six ones and two zeros.

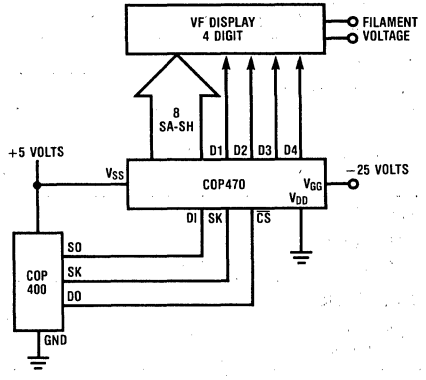



Figure 4. System Diagram — 4 Digit Display

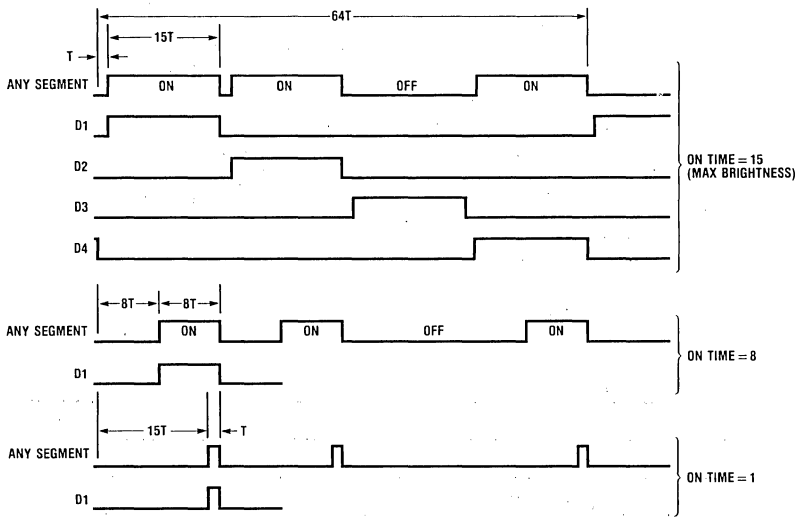



**Figure 5. Segment and Digit Output Timing Diagram**



### Loading Sequence:

Step

1 Turn $\overline{CS}$ Low.
2 Clock in 8 bits of data for digit 1.
3 Clock in 8 bits of data for digit 2.
4 Clock in 8 bits of data for digit 3.
5 Clock in 8 bits of data for digit 4.
6 Clock in 8 bits of data for on time and control bits.
7 Turn $\overline{CS}$ high.

Note: $\overline{CS}$ may be turned high after any step. For example, to load only 2 digits of data do steps 1, 2, 3, and 7. $\overline{CS}$ must make a high to low transition before loading data in order to reset internal counters.

### 8 Digit Displays

Two COP470s may be tied together in order to drive an eight digit multiplexed display. This is shown in Figure 6. The following is the loading sequence to drive an eight digit display using two COP470s.

1. Turn $\overline{CS}$ low on both COP470s.
2. Shift in 32 bits of data for the right 4 digits.
3. Shift in 4 bits of on time, a zero and three ones. This synchronizes both chips, sets to external oscillator, and to right four of eight digits. Thus both chips are synchronized and the oscillator is stopped.
4. Turn $\overline{CS}$ high to both chips.
5. Turn $\overline{CS}$ low to the left COP470.
6. Shift in 32 bits of data for the left 4 digits.
7. Shift in 4 bits of on time, a one and three zeros. This sets this COP470 to internal oscillator and to left four of eight digits. Now both chips start and run off the same oscillator.
8. Turn $\overline{CS}$ high.

The chips are now synchronized and driving eight digits of display. To load new data simply load each chip separately in the normal manner.

### 16 Segment Display

Two COP470s may be tied together in order to drive a sixteen segment display. This is shown in Figure 8. To do this, both chips must be synchronized, one must run off external oscillator while the other runs off its internal oscillator outputting to the other. Similarly, four COP470s could be tied together to drive eight digits of sixteen segments.

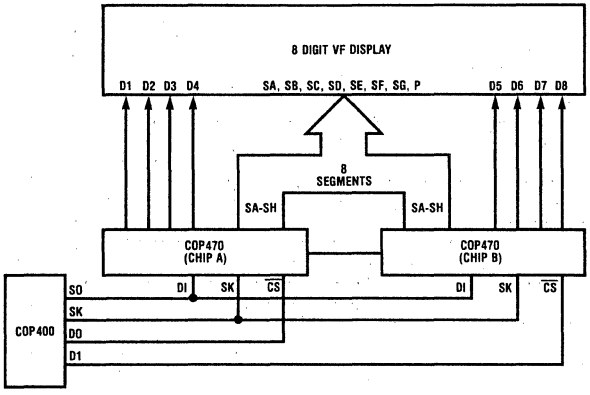



**Figure 6. System Diagram 8 Digit Display**

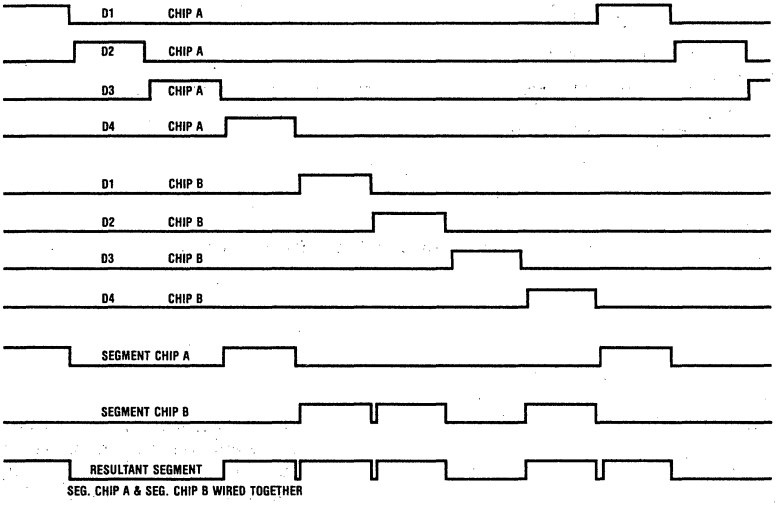



**Figure 7. Segment and Digit Output Timing Diagram for 8 Digits**

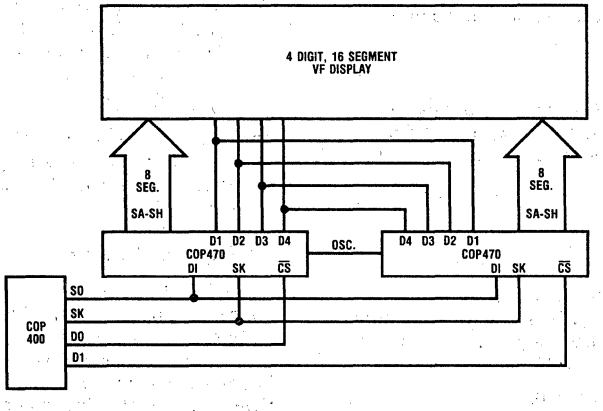



**Figure 8. System Diagram for 16 Segment Display**

## LED Display

The COP470 may be used to drive LED displays. The COP470 can drive the segments directly on small, low current LED displays as shown in Figure 9. By adding display drivers, large, high current LED displays can be driven as shown in Figure 10.

**Example:**

**COP420 Code to Load COP470**

**(Display Data is in Memory 0, 12 — 0, 15)**

```
        LBI 0,12        ; Point to first display data
        OBD             ; Turn CS low (DO)
LOOP:   CLRA
        LQID            : Look up segment data
        CQMA            ; Copy data from Q to M & A
        SC              ; Set C to turn on SK
        XAS             ; Output lower 4 bits of data
        NOP             ; Delay
        NOP             ; Delay
        LD              ; Load A with upper 4 bits
        XAS             ; Output 4 bits of data
        NOP             ; Delay
        NOP             ; Delay
        RC              ; Reset C
        XAS             ; Turn off SK clock
        XIS             ; Increment B for next data
        JP LOOP         ; Skip this jump after last digit
        SC              ; Set C
        CLRA            ;
        AISC 15         ; 15 to A
        XAS             ; Output on time (max brightness)
        NOP             ;
        CLRA            ;
        AISC 12         ; 12 to A
        XAS             ; Output control bits
        NOP             ;
        LBI 0,15        ; 15 to B
        RC              ; Reset C
        XAS             ; Turn off SK
        OBD             ; Turn CS high (DO)
```

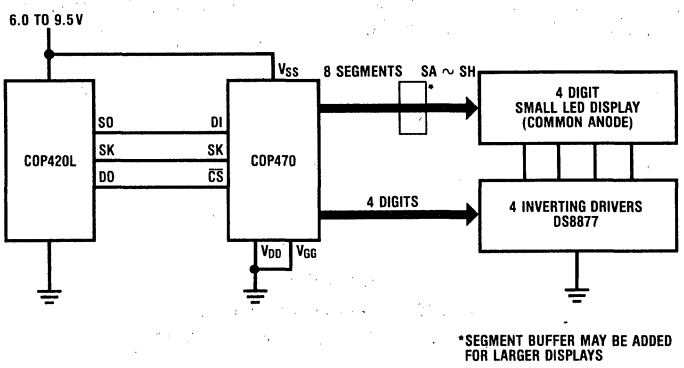



Figure 9. LED Display

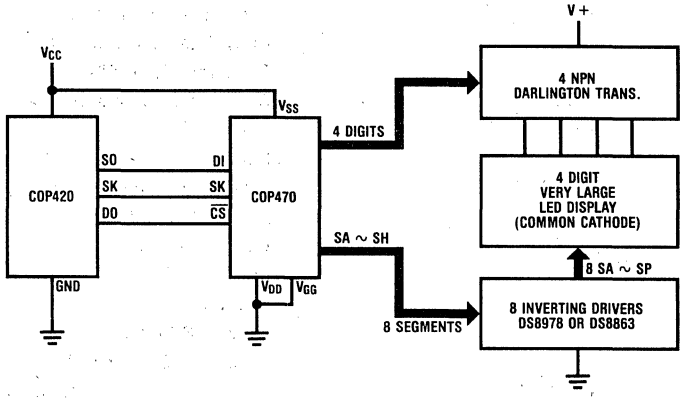



Figure 10. Large LED Display

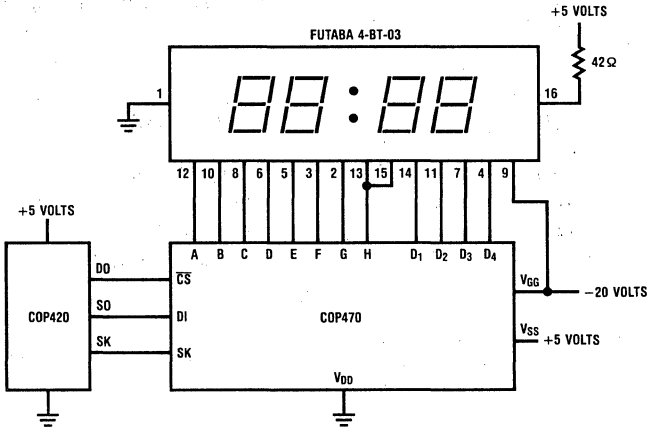



Figure 11. Sample V.F. System

National Semiconductor

PRELIMINARY

# COP472 Liquid Crystal Display Controller

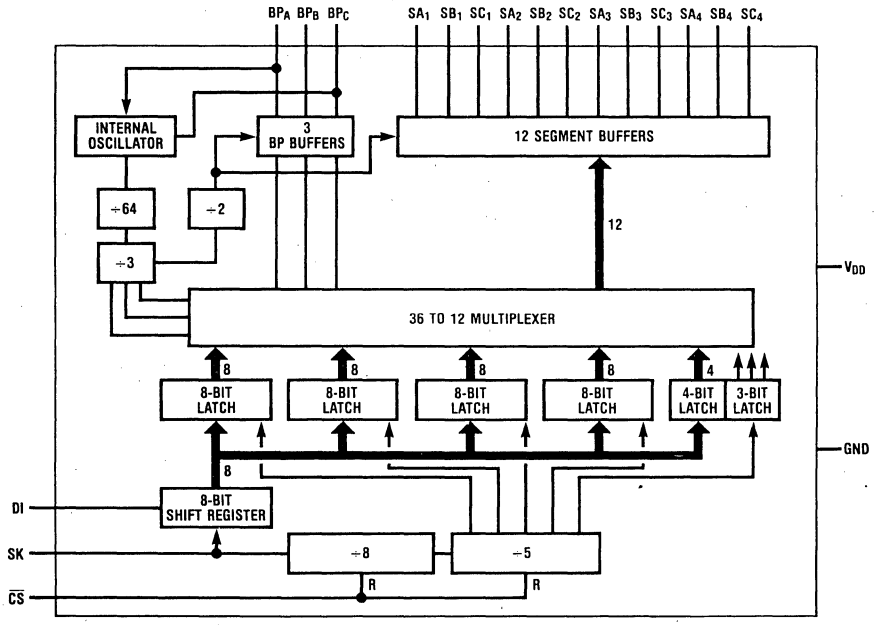
## General Description

The COP472 Liquid Crystal Display (LCD) Controller is a peripheral member of the COPS™ family, fabricated using CMOS technology. The COP472 drives a multiplexed liquid crystal display directly. Data is loaded serially and is held in internal latches. The COP472 contains an on-chip oscillator and generates all the multi-level waveforms for backplanes and segment outputs on a triplex display. One COP472 can drive 36 segments multiplexed as 3 x 12 (4½ digit display). Two COP472 devices can be used together to drive 72 segments (3 x 24) which could be an 8½ digit display.

## Features

- Direct interface to TRIPLEX LCD
- Low power dissipation (100μW typ.)
- Low cost
- Compatible with all COP400 processors
- Needs no refresh from processor
- On-chip oscillator and latches
- Expandable to longer displays
- Software compatible with COP470 V.F. Display Driver chip
- Operates from display voltage
- MICROWIRE™ compatible serial I/O
- 20-pin dual-in-line package






**COP472 Block Diagram**

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at CS, DI, SK pins | −0.3V to +9.5V |
| Voltage at all other Pins | −0.3V to $V_{DD}$ + 0.3V |
| Operating Temperature Range | 0°C to 70°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 Seconds) | 300°C |

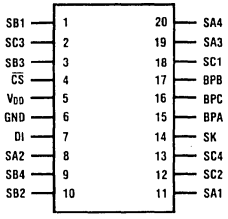
## Electrical Characteristics

GND = 0V, $V_{DD}$ = 2.4V to 5.5V,
$T_A$ = 0°C to 70°C (depends on display characteristics)

### D.C. Electrical Specifications

| Parameter | Conditions | Min. | Max. | Units |
|---|---|---|---|---|
| Power Supply Voltage, $V_{DD}$ | | 2.4 | 5.5 | Volts |
| Power Supply Current, $I_{DD}$ | | 30 | 60 | μA |
| Input Levels | | | | |
| DI, SK, CS | | | | |
| $V_{IL}$ | | | 0.8 | Volts |
| $V_{IH}$ | | 0.7 $V_{DD}$ | 9.5 | Volts |
| BPA (as Osc. In) | | | | |
| $V_{IL}$ | | | 0.6 | Volts |
| $V_{IH}$ | | $V_{DD}$ − 0.6 | $V_{DD}$ | Volts |
| Output Levels, BPC (as Osc. Out) | | | | |
| $V_{OL}$ | | | 0.4 | Volts |
| $V_{OH}$ | | $V_{DD}$ − 0.4 | $V_{DD}$ | Volts |
| Backplane Outputs (BPA, BPB, BPC) | | | | |
| $V_{BPA, BPB, BPC}$ ON | During | $V_{DD}$ − 0.1 | $V_{DD}$ | Volts |
| $V_{BPA, BPB, BPC}$ OFF | $BP^+$ Time | ⅓ $V_{DD}$ − 0.1 | ⅓ $V_{DD}$ + 0.1 | Volts |
| $V_{BPA, BPB, BPC}$ ON | During | 0 | 0.1 | Volts |
| $V_{BPA, BPB, BPC}$ OFF | $BP^-$ Time | ⅔ $V_{DD}$ − 0.1 | ⅔ $V_{DD}$ + 0.1 | Volts |
| Segment Outputs ($SA_1 \sim SA_4$) | | | | |
| $V_{SEG}$ ON | During | 0 | 0.1 | Volts |
| $V_{SEG}$ OFF | $BP^+$ Time | ⅔ $V_{DD}$ − 0.1 | ⅔ $V_{DD}$ + 0.1 | Volts |
| $V_{SEG}$ ON | During | $V_{DD}$ − 0.1 | $V_{DD}$ | Volts |
| $V_{SEG}$ OFF | $BP^-$ Time | ⅓ $V_{DD}$ − 0.1 | ⅓ $V_{DD}$ + 0.1 | Volts |
| Internal Oscillator Frequency | | 60 | 110 | kHz |
| Frame Time (Int. Osc. ÷ 384) | | 3.4 | 6.4 | ms |
| SK Clock Frequency | | 4 | 250 | kHz |
| SK Width | | 1.7 | | μs |
| DI | | | | |
| Data Setup, $t_{SETUP}$ | | 1.0 | | μs |
| Data Hold, $t_{HOLD}$ | | 100 | | ns |
| $\overline{CS}$ | | | | |
| $t_{SETUP}$ | | 1.0 | | μs |
| $t_{HOLD}$ | | 1.0 | | μs |
| Output Loading Capacitance | | | 100 | pF |

**COP472**
**Connection Diagram**




| Pin | Description |
|---|---|
| $\overline{CS}$ | Chip select |
| $V_{DD}$ | Power supply (display voltage) |
| GND | Ground |
| DI | Serial data input |
| SK | Serial clock input |
| $BP_A$ | Display backplane A(or oscillator in) |
| $BP_B$ | Display backplane B |
| $BP_C$ | Display backplane C (or oscillator out) |
| SA1~SC4 | 12 multiplexed outputs |

**Figure 2. Connection Diagram**

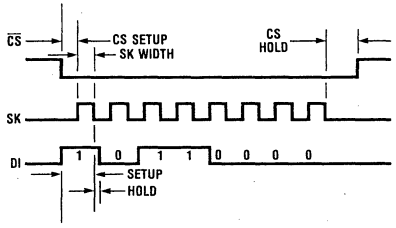



**Figure 3. Serial Load Timing Diagram**

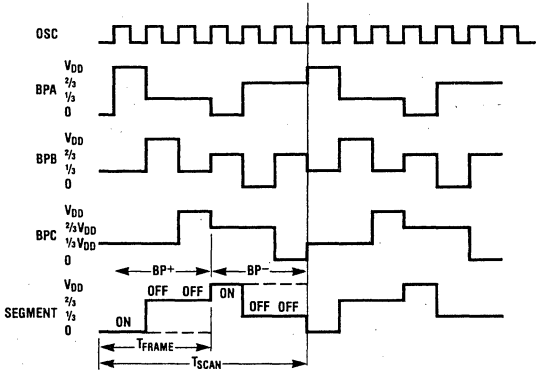
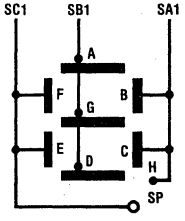



**Figure 4. Backplane and Segment Waveforms**



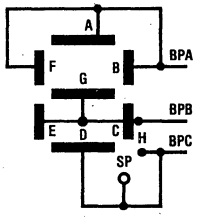




**Figure 5. Typical Display Internal Connections Epson LD-370**

## Functional Description

The COP472 drives 36 bits of display information organized as twelve segments and three backplanes. The COP472 requires 40 information bits: 36 data and 4 control. The function of each control bit is described below. Display information format is a function of the LCD interconnections. A typical segment/backplane configuration is illustrated in Figure 5, with this configuration the COP472 will drive 4 digits of 9 segments.

To adapt the COP472 to any LCD display configuration, the segment/backplane multiplex scheme is illustated in Table 1.

Two or more COP472 chips can be cascaded to drive additional segments. There is no limit to the number of COP472's that can be used as long as the output loading capacitance does not exceed specification.

**Table 1. COP472 Segment/Backplane Multiplex Scheme**

| Bit Number | Segment, Backplane | Data to Numeric Display | |
|---|---|---|---|
| 1 | SA1, BPC | SH | |
| 2 | SB1, BPB | SG | |
| 3 | SC1, BPA | SF | |
| 4 | SC1, BPB | SE | |
| 5 | SB1, BPC | SD | Digit 1 |
| 6 | SA1, BPB | SC | |
| 7 | SA1, BPA | SB | |
| 8 | SB1, BPA | SA | |
| 9 | SA2, BPC | SH | |
| 10 | SB2, BPB | SG | |
| 11 | SC2, BPA | SF | |
| 12 | SC2, BPB | SE | |
| 13 | SB2, BPC | SD | Digit 2 |
| 14 | SA2, BPB | SC | |
| 15 | SA2, BPA | SB | |
| 16 | SB2, BPA | SA | |
| 17 | SA3, BPC | SH | |
| 18 | SB3, BPB | SG | |
| 19 | SC3, BPA | SF | |
| 20 | SC3, BPB | SE | |
| 21 | SB3, BPC | SD | Digit 3 |
| 22 | SA3, BPB | SC | |
| 23 | SA3, BPA | SB | |
| 24 | SB3, BPA | SA | |
| 25 | SA4, BPC | SH | |
| 26 | SB4, BPB | SG | |
| 27 | SC4, BPA | SF | |
| 28 | SC4, BPB | SE | |
| 29 | SB4, BPC | SD | Digit 4 |
| 30 | SA4, BPB | SC | |
| 31 | SA4, BPA | SB | |
| 32 | SB4, BPA | SA | |
| 33 | SC1, BPC | SP1 | Digit 1 |
| 34 | SC2, BPC | SP2 | Digit 2 |
| 35 | SC3, BPC | SP3 | Digit 3 |
| 36 | SC4, BPC | SP4 | Digit 4 |
| 37 | not used | | |
| 38 | Q6 | | |
| 39 | Q7 | | |
| 40 | SYNC | | |

### Segment Data bits

Data is loaded in serially, in sets of eight bits. Each set of segment data is in the following format:

| SA | SB | SC | SD | SE | SF | SG | SH |

Data is shifted into an eight bit shift register. The first bit of the data is for segment H, digit 1. The eighth bit is segment A, digit 1. A set of eight bits is shifted in and then loaded into the digit one latches. The second set of 8 bits is loaded into digit two latches. The third set into digit three latches, and the fourth set is loaded into digit four latches.

### Control Bits

The fifth set of 8 data bits contains special segment data and control data in the following format:

| SYNC | Q7 | Q6 | X | SP4 | SP3 | SP2 | SP1 |

The first four bits shifted in contain the special character segment data. The fifth bit is not used. The sixth and seventh bits program the COP472 as a stand alone LCD driver or as a master or slave for cascading COP472's. BPC of the master is conected to BPA of each slave. The following table summarizes the function of bits six and seven:

| Q7 | Q6 | Function | BPC Output | BPA Output |
|---|---|---|---|---|
| 1 | 1 | Slave | Backplane Output | Oscillator Input |
| 0 | 1 | Stand Alone | Backplane Output | Backplane Output |
| 1 | 0 | Not Used | Internal Osc. Output | Oscillator Input |
| 0 | 0 | Master | Internal Osc. Output | Backplane Output |

The eighth bit is used to synchronize two COP472's to drive an 8½-digit display.

### Loading Sequence to Drive a 4½-Digit Display

Steps:

1. Turn $\overline{CS}$ low.
2. Clock in 8 bits of data for digit 1.
3. Clock in 8 bits of data for digit 2.
4. Clock in 8 bits of data for digit 3.
5. Clock in 8 bits of data for digit 4.
6. Clock in 8 bits of data for special segment and control function of BPC and BPA.

| 0 | 0 | 1 | 1 | SP4 | SP3 | SP2 | SP1 |

7. Turn $\overline{CS}$ high.

**Note:** $\overline{CS}$ may be turned high after any step. For example to load only 2 digits of data, do steps 1, 2, 3, and 7.

$\overline{CS}$ must make a high to low transition before loading data in order to reset internal counters.

### Loading Sequence to Drive an 8½-Digit Display

Two or more COP472's may be connected together to drive additional segments. An eight digit multiplexed display is shown in Figure 7. The following is the loading sequence to drive an eight digit display using two COP472's. The right chip is the master and the left the slave.

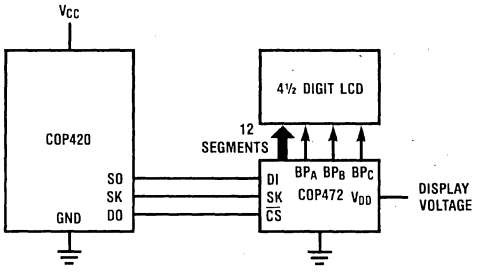



**Figure 6. System Diagram — 4½ Digit Display**

Steps:

1. Turn $\overline{CS}$ low on both COP472's.
2. Shift in 32 bits of data for for the slave's four digits.
3. Shift in 4 bits of special segment data: a zero and three ones.

| 1 | 1 | 1 | 0 | SP4 | SP3 | SP2 | SP1 |
|---|---|---|---|---|---|---|---|

This synchronizes both the chips and BPA is oscillator input. Both chips are now stopped.

4. Turn CS high to both chips.
5. Turn CS low to master COP472.
6. Shift in 32 bits of data for the master's 4 digits.
7. Shift in four bits of special segment data, a one and three zeros.

| 0 | 0 | 0 | 1 | SP4 | SP3 | SP2 | SP1 |
|---|---|---|---|---|---|---|---|

This sets the master COP472 to BPA as a normal backplane output and BPC as oscillator output. Now both the chips start and run off the same oscillator.

8. Turn $\overline{CS}$ high.

The chips are now synchronized and driving 8 digits of display. To load new data simply load each chip separately in the normal manner, keeping the correct status bits to each COP472.

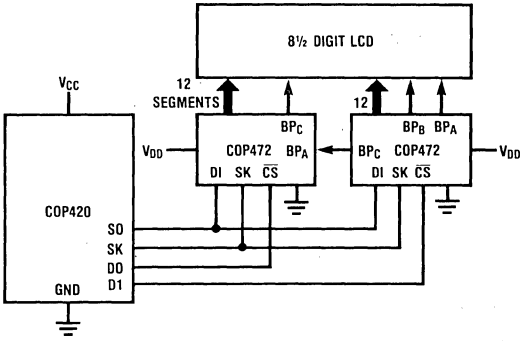



**Figure 7. System Diagram — 8½ Digit Display**

## Example Software

### Example 1.

COP420 Code to load a COP472 [Display data is in M(0, 12)-M(0, 15), special segment data is in M(0, 0)]

```
        LBI 0, 12       ; POINT TO FIRST DISPLAY DATA
        OBD             ; TURN CS LOW (DO)
LOOP:   CLRA
        LQID            ; LOOK UP SEGMENT DATA
        CQMA            ; COPY DATA FROM Q TO M & A
        SC              ; SET C TO TURN ON SK
        XAS             ; OUTPUT LOWER 4 BITS OF DATA
        NOP             ; DELAY
        NOP             ; DELAY
        LD              ; LOAD A WITH UPPER 4 BITS
        XAS             ; OUTPUT 4 BITS OF DATA
        NOP             ; DELAY
        NOP             ; DELAY
        RC              ; RESET C
        XAS             ; TURN OFF SK CLOCK
        XIS             ; INCREMENT B FOR NEXT DATA
        JP LOOP         ; SKIP THIS JUMP AFTER LAST DIGIT
        SC              ; SET C
        LBI 0, 0        ; ADDRESS SPECIAL SEGMENTS
        LD              ; LOAD INTO A
        XAS             ; OUTPUT SPECIAL SEGMENTS
        NOP             ;
        CLRA            ;
        AISC 12         ; 12 to A
        XAS             ; OUTPUT CONTROL BITS
        NOP             ;
        LBI 0, 15       ; 15 to B
        RC              ; RESET C
        XAS             ; TURN OFF SK
        OBD             ; TURN CS HIGH (DO)
```

**Example 2**

COP420 Code to load two COP472 parts [display data is in M(0, 12)-M(0,15) and M(1, 12)-M(1, 15), special segment data is in M(0, 0) and M(1, 0)]

```
INIT:   LBI     0, 15
        OBD                     ; TURN BOTH CS'S HIGH
        LEI     8               ; ENABLE SO OUT OF S. R.
        RC
        XAS                     ; TURN OFF SK CLOCK
        LBI     3, 15           ; USE M(3, 15) FOR CONTROL BITS
        STII    7               ; STORE 7 TO SYNC BOTH CHIPS
        LBI     0, 12           ; SET B TO TURN BOTH CS'S LOW
        JSR     OUT             ; CALL OUTPUT SUBROUTINE
```

**MAIN DISPLAY SEQUENCE**

```
DISPLAY: LBI    3, 15
        STII    8               ; SET CONTROL BITS FOR SLAVE
        LBI     0, 13           ; SET B TO TURN SLAVE CS LOW
        JSR     OUT             ; OUTPUT DATA FROM REG. 0
        LBI     3, 15
        STII    6               ; SET CONTROL BITS FOR MASTER
        LBI     1, 14           ; SET B TO TURN MASTER CS LOW
        JSR     OUT             ; OUTPUT DATA FROM REG. 1
```

**OUTPUT SUBROUTINE**

```
OUT:    OBD                     ; OUTPUT B TO CS'S
        CLRA
        AISC    12              ; 12 TO A
        CAB                     ; POINT TO DISPLAY DIGIT (BD=12)
LOOP:   CLRA
        LQID                    ; LOOK UP SEGMENT DATA
        CQMA                    ; COPY DATA FROM Q TO M & A
        SC
        XAS                     ; OUTPUT LOWER 4 BITS OF DATA
        NOP                     ; DELAY
        NOP                     ; DELAY
        LD                      ; LOAD A WITH UPPER 4 BITS
        XAS                     ; OUTPUT 4 BITS OF DATA
        NOP                     ; DELAY
        NOP                     ; DELAY
        RC                      ; RESET C
        XAS                     ; TURN OFF SK
        XIS                     ; INCREMENT B FOR NEXT DISPLAY DIGIT
        JP      LOOP            ; SKIP THIS JUMP AFTER LAST DIGIT
        SC                      ; SET C
        NOP
        LD                      ; LOAD SPECIAL SEGS. TO A (BD=0)
        XAS                     ; OUTPUT SPECIAL SEGMENTS
        NOP
        LBI     3, 15
        LD                      ; LOAD A
        XAS                     ; OUTPUT CONTROL BITS
        NOP
        NOP
        RC
        XAS                     ; TURN OFF SK
        OBD                     ; TURN CS'S HIGH (BD=15)
        RET
```

# COP498 Low Power CMOS RAM and Timer

## General Description

The COP498 low power CMOS Random-Access Memory and Timer is a peripheral member of the COPS™ family, fabricated using CMOS technology. It is an external memory and timer chip with the simple MICROWIRE™ serial interface. The device contains 256 bits of read/write memory divided into 4 registers of 64 bits each. Each register can be serially loaded or read by a COP400 controller. The COP498 also contains a crystal-based timer for timekeeping purposes, and can provide a "wake-up" signal to turn on a COPS controller.

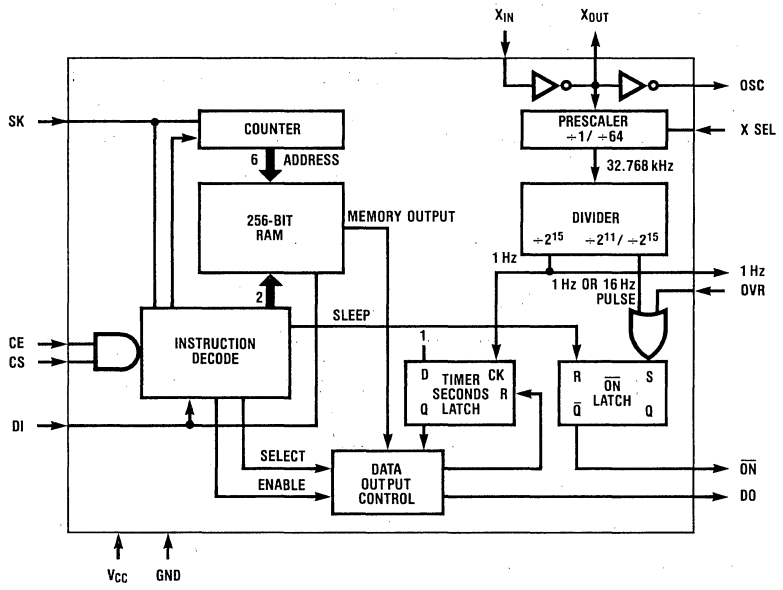
The COP498 can be used for low power standby memory and can also be used for low power operation by turning the controller off and on, on a duty cycle basis.

A COP400 N-channel controller coupled with the COP498 RAM/Timer offers a user the low-power advantages of a CMOS system and the low-cost advantage of an NMOS system. This type of system solution is ideally suited to a wide variety of automotive and instrumentation applications.

## Features

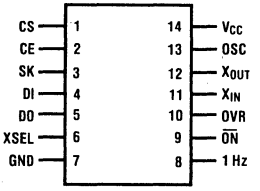
- Low power dissipation
- Low cost
- Single supply operation (2.5V-6.0V)
- CMOS compatible I/O
- 4 × 64 serial read/write memory
- Selectable crystal-based timer (2.097152 MHz or 32.768 kHz)
- Software selectable 1 Hz or 16 Hz "wake-up" signal for COPS controller
- External override
- Compatible with all COP400 processors (processor $V_{CC} \leq 9.5V$)
- MICROWIRE compatible serial I/O
- 14-pin dual-in-line package






**COP498 Block Diagram**

**COP498**
**Connection Diagram**

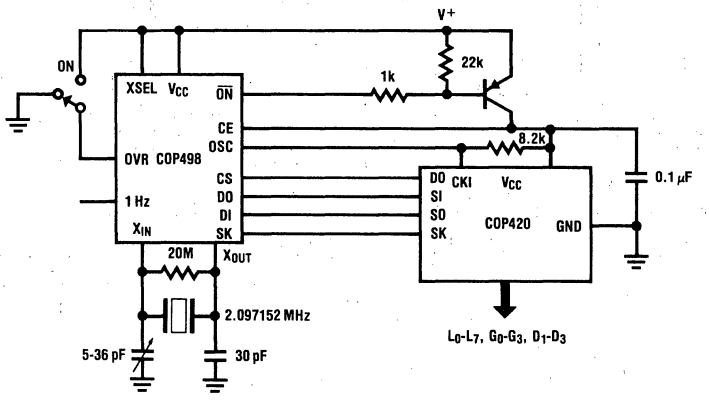



| Pin | Description |
|---|---|
| CS | Chip Select |
| CE | Chip Enable |
| SK | Serial Data Clock |
| DI | Serial Data Input |
| DO | Serial Data Output |
| XSEL | Crystal Option Select |
| XIN | Crystal Oscillator Input |
| XOUT | Crystal Oscillator Output |
| 1 Hz | 1 Hz square wave output |
| $\overline{ON}$ | Active low wake-up signal to COPS controller |
| OVR | External override wake-up for COPS controller |
| OSC | Open drain oscillator output |
| $V_{CC}$ | Power Supply |
| GND | Ground |

**COP498 Instruction Set**

| | | |
|---|---|---|
| WRITE | 1 s 1 $r_1$ $r_0$ | s = $\overline{ON}$ (wake up signal) frequency select |
| READ | 1 1 0 $r_1$ $r_0$ | 1 = 16 Hz; 0 = 1 Hz |
| | | $r_1$ $r_0$ = register number (00, 01, 10, 11) |
| WREN | 1 0 0 1 1 | Write Enable |
| WRDS | 1 0 0 0 0 | Write Disable |
| TSEC | 1 0 0 1 0 | Test timer seconds latch |
| SLEEP | 1 0 0 0 1 | Put COP controller to sleep ($\overline{ON}$ goes high) |

The instruction setup and chip select/chip enable structure is organized so as to provide maximum protection to the read/write memory while the COPS™ controller is powered up and down.




**Typical System Diagram**

# COP499 Low Power CMOS Memory

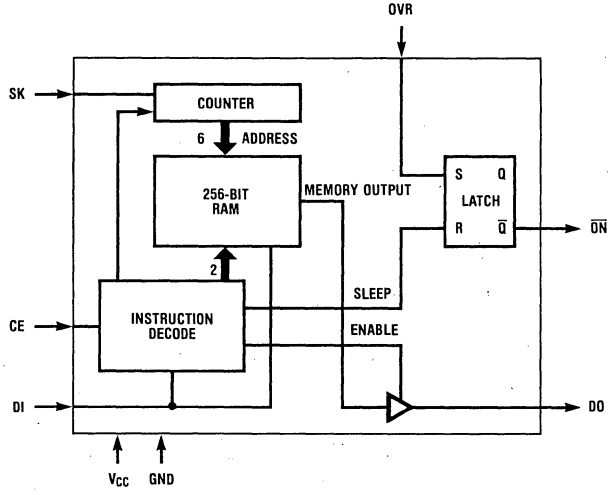
## General Description

The COP499 Low Power CMOS Random-Access Memory is a peripheral member of the COPS™ family, fabricated using CMOS technology. It is an external memory and switch chip with the simple MICROWIRE™ serial interface. The device contains 256 bits of read/write memory divided into 4 registers of 64 bits each. Each register can be serially loaded or read by a COP400 controller. The COP499 also contains circuitry that enables the user to turn a controller on and off while maintaining the integrity of the memory.

## Features

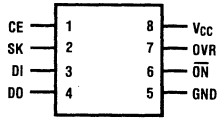
- Low power dissipation
- Low cost
- Single supply operation (2.5V-6.0V)
- CMOS compatible I/O
- 4 x 64 serial read/write memory
- External "wake-up" signal for COPS controller
- Compatible with all COP400 processors (processor $V_{CC} \leq 9.5V$)
- MICROWIRE compatible serial I/O
- 8-pin mini-DIP






COP499 Block Diagram

**COP499**
**Connection Diagram**

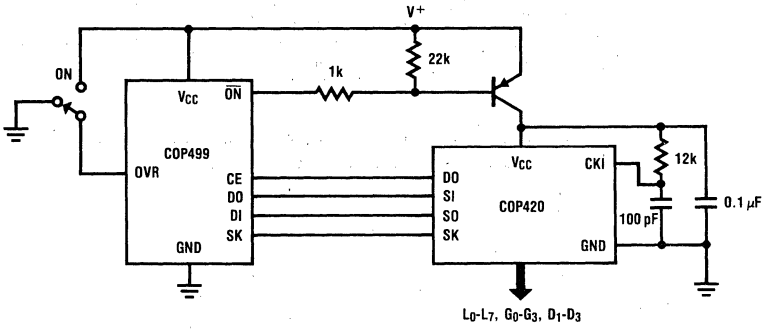



| Pin | Description |
|---|---|
| CE | Chip enable |
| SK | Serial data clock |
| DI | Serial data input |
| DO | Serial data output |
| $V_{CC}$ | Power supply |
| GND | Ground |
| $\overline{ON}$ | Active low wake-up signal to COPS controller |
| OVR | External wake-up signal |

**COP499 Instruction Set**

| | | |
|---|---|---|
| WRITE | 1 0 1 $r_1$ $r_0$ | Write to memory $r_1$ $r_0$ = register number (00, 01, 10, 11) |
| READ | 1 1 0 $r_1$ $r_0$ | Read from memory |
| WREN | 1 0 0 1 1 | Write Enable |
| WRDS | 1 0 0 0 0 | Write Disable |
| SLEEP | 1 0 0 0 1 | Put COP controller to sleep (ON goes high) |

The instruction setup and chip select/chip enable structure is organized so as to provide maximum protection to the read/write memory while the COPS™ controller is powered up and down.




**Typical System Diagram**

Section 2

# COPS Application Information

# COP400 Microcontroller Family

## COPS™ Family User's Guide

2

National Semiconductor

420305785-001

# Table of Contents


| Section | Description | Page |
|---|---|---|
| | **Chapter 1. Introduction to COP400 Microcontrollers** | |
| 1.1 | Summary of COP400 Microcontroller Features | 2-8 |
| | **Chapter 2. COP400 Architecture** | |
| 2.1 | COP420/COP421 Architecture | 2-10 |
| 2.2 | COP420/COP421 Functional Description | 2-12 |
| 2.3 | Initialization | 2-13 |
| 2.4 | COP420/COP421 Mask Programmable Options | 2-14 |
| 2.5 | COP420/COP421 Option List | 2-18 |
| 2.6 | COP420L/COP421L Description | 2-19 |
| 2.7 | COP420L/COP421L Mask Programmable Options | 2-19 |
| 2.8 | COP420L/COP421L Option List | 2-21 |
| 2.9 | COP420C Description | 2-22 |
| 2.10 | COP444L Description | 2-23 |
| 2.11 | COP402 and COP402M ROM-less Part Description | 2-23 |
| 2.12 | COP404L ROM-less Part Description | 2-23 |
| 2.13 | COP410L/COP411L Architecture | 2-23 |
| 2.14 | COP410L/COP411L Functional Description | 2-25 |
| 2.15 | COP410L/COP411L Mask Programmable Options | 2-26 |
| 2.16 | COP410L/COP411L Option List | 2-26 |
| | **Chapter 3. COP400 Instruction Sets** | |
| 3.1 | COP420-Series/COP444L Instruction Set | 2-29 |
| 3.2 | COP420-Series/COP444L Instruction Set Description | 2-33 |
| 3.3 | COP421-Series Instruction Set Differences | 2-40 |
| 3.4 | COP410L/COP411L Instruction Set | 2-40 |
| 3.5 | COP410L/COP411L Instruction Set Differences | 2-43 |

# Table of Contents


| Section | Description | Page |
|---|---|---|
| | **Chapter 4. COP400 Programming Techniques** | |
| 4.1 | Program Memory Allocation | 2-48 |
| 4.2 | Data Memory Allocation and Manipulation | 2-51 |
| 4.3 | Subroutine Techniques | 2-52 |
| 4.4 | Utility Routines | 2-53 |
| 4.5 | Timing Considerations | 2-54 |
| 4.6 | BCD Arithmetic Routines | 2-55 |
| 4.7 | Simple Display Loop Routine | 2-57 |
| 4.8 | Interrupt Service Routine | 2-59 |
| 4.9 | Timekeeping Routine | 2-59 |
| 4.10 | String Search Routine | 2-62 |
| 4.11 | Programming Techniques for the COP421-Series, COP410L and 411L | 2-63 |
| | **Chapter 5. COP400 I/O Techniques** | |
| 5.1 | Hardware Interfacing Techniques | 2-64 |
| 5.2 | Software I/O Techniques | 2-69 |
| 5.3 | Keyboard/Display Interface | 2-70 |
| 5.4 | SIO (Serial) Input/Output | 2-81 |
| 5.5 | Add-on RAM | 2-82 |
| 5.6 | $IN_3/IN_0$ Inputs | 2-83 |
| | **Appendices — COP400 Data Sheets** | |

# List of Figures


| Figure | Description | Page |
|---|---|---|
| 2.1 | COP420/COP421 Block Diagram | 2-10 |
| 2.2 | COP420/COP421 Connection Diagrams | 2-11 |
| 2.3 | COP420/COP421 Pin Descriptions | 2-11 |
| 2.4 | Power-Clear Circuit | 2-11 |
| 2.5 | COP420/COP421 Clock Oscillator Configurations | 2-15 |
| 2.6 | COP420/COP421 Input/Output Configurations | 2-16 |
| 2.7 | COP420/COP421 Input/Output Characteristics | 2-17 |
| 2.8 | COP420L/COP421L Oscillator Configurations | 2-20 |
| 2.9 | COP410L/COP411L Block Diagram | 2-24 |
| 2.10 | COP410L/COP411L Connection Diagrams | 2-25 |
| 2.11 | COP410L/COP411L Pin Description | 2-25 |
| 2.12 | COP410L/COP411L Oscillator Configurations | 2-27 |
| 3.1 | INIL Hardware Implementation | 2-34 |
| 3.2 | Enable Register Features — Bits $EN_3$ and $EN_0$ | 2-39 |
| 4.1 | COP420 Data Memory Map | 2-51 |
| 4.2 | Flowchart for Multiply Routine | 2-56 |
| 4.3 | Flowchart for Timekeeping Routine | 2-60 |
| 5.1 | COP420 I/O Lines | 2-65 |
| 5.2 | COP420 I/O Options | 2-65 |
| 5.3 | COP420 Standard Output Characteristics | 2-65 |
| 5.4 | COP420 I/O Interconnect Examples | 2-66 |
| 5.5 | COP420 IN Input Characteristics | 2-66 |
| 5.6 | D and G Port Characteristics | 2-67 |
| 5.7 | COP420L I/O Port Characteristics | 2-67 |
| 5.8 | COP420 SI, SO, SK Characteristics | 2-68 |
| 5.9 | COP420 CKO, CKI, RESET Characteristics | 2-68 |
| 5.10 | COP420 I/O Expansion | 2-69 |
| 5.11 | COP420 LED Display System | 2-69 |
| 5.12 | COP420 VF Display System | 2-69 |
| 5.13 | COP420 MICROBUS™ Interconnect | 2-69 |
| 5.14 | COP420 Add-On RAM | 2-69 |
| 5.15 | Display/Keyboard Interconnect | 2-70 |
| 5.16 | Flowchart for Display/Keyboard Debounce Routine | 2-71 |
| 5.17 | Display Timing Diagram | 2-72 |
| 5.18 | Display/Keyboard Interface Source Code | 2-77 |
| 5.19 | Key-Decode Routine Assembler Output Listing | 2-80 |
| 5.20 | Additional I/O Using SI and SO | 2-81 |
| 5.21 | Multi-COP420 System | 2-81 |
| 5.22 | Add-On RAM Interconnect | 2-83 |

# List of Tables


| Table | Description | Page |
|---|---|---|
| 1.1 | COP400 Device Features | 2-9 |
| 2.1 | Enable Register Functions — Bits $EN_3$ and $EN_0$ | 2-14 |
| 3.1 | COP420/COP421 Instruction Set Table | 2-30 |
| 3.2 | COP420/COP421 Instruction Set Symbols | 2-33 |
| 3.3 | COP410L/COP411L Instruction Set Table | 2-40 |
| 3.4 | COP410L/COP411L Instruction Set Symbols | 2-43 |
| 3.5 | Alphabetical Mnemonic Index of COP420/COP421 Instructions | 2-44 |
| 3.6 | COP420/COP421 Instructions Listed by Hex Opcodes | 2-44 |
| 3.7 | Alphabetical Mnemonic Index of COP410L/COP411L Instructions | 2-46 |
| 3.8 | COP410L/COP411L Instructions Listed by Hex Opcodes | 2-47 |
| 4.1 | Page to Hexadecimal Address | 2-48 |
| 5.1 | COP400 I/O Comparison Chart | 2-64 |
| 5.2 | Seven-Segment Decode Values | 2-74 |
| 5.3 | JID Pointer Table for Display/Keyboard Routine | 2-79 |

# 1 Introduction to the COP400 Microcontrollers

This manual provides information on the COP400 series of National's single-chip microcontrollers. The material contained in this manual is intended to assist the reader in understanding the internal architecture, instruction set, programming techniques, and hardware and software I/O techniques pertaining to the COP400 family of microcontroller devices.

The primary focus of this manual is the COP420 — at the time of this printing the most inclusive device, on a hardware and software level, of the COP400 family. Other members of the COP400 family are discussed primarily in terms of the less inclusive features of these other parts (i.e., the COP421, COP410L, COP411L). This approach should not result in a lack of understanding in terms of the operation and programming of these parts since they are "subset" devices of the COP420, distinguished, for the most part, by deleted hardware and software features. For further information on these other devices and on future COP400 devices the reader should consult the data sheets appropriate to particular COP400 devices.

## 1.1 Summary of COP400 Microcontroller Features

COP400 Microcontrollers are fabricated using CMOS or N-channel, silicon gate MOS technology. They are complete microcomputers containing all system timing, internal logic, ROM, RAM, and I/O necessary to implement dedicated control functions in a variety of applications. Features of the COP400 devices include an instruction set, internal architecture, and I/O scheme designed to facilitate keyboard input, display output, and efficient BCD data manipulation.

The various members of the COP400 family allow the user to specify a microcontroller best suited for use in a particular dedicated application. Specifically, COP400 devices offer a choice among single-chip parts with differing amounts of ROM, RAM, I/O capability, and number of instructions. Additionally, many parts have different versions which allow a choice of electrical characteristics while retaining the basic architecture and instruction set of the basic device. (For example, the COP420L and COP420C are available as low-power and CMOS versions, respectively, of the standard COP420 device.) Finally, each part contains a number of clock, I/O and other options, mask-programmed into the part at the same time as the user's program; this allows even greater flexibility in matching the COP400 Microcontroller to the user's specifications, reducing the need for external interface logic.

All COP400 devices feature single-supply operation and fast, standardized, "in-house" test procedures which verify the internal logic and user program (ROM code) mask-programmed into the device. Several COP400 controllers are available in ROM-less versions for use in prototyping a COP400 system (using the COP400 Development System) or for low-volume applications.

Table 1.1 provides a list of COP400 devices currently available or in design, together with a summary of the basic features of each device. Refer to this manual and data sheets of particular devices for further information on these parts. Future members of the COP400 family will include more powerful hardware and software capabilities, alternative electrical specification devices (low power, CMOS versions) and peripheral devices suitable for use in many applications.

The flexible I/O configuration of COP400 Microcontrollers allows them to interface with and drive a wide range of devices using minimal external parts. Typical peripheral devices include:

1. Keyboards and displays (direct segment and digit drive possible for several devices).
2. External data memories.
3. Printers.
4. Other COPS™ devices.
5. A/D and D/A converters.
6. Power control devices (SCRs, TRIACs).
7. Mechanical actuators.
8. General purpose microprocessors (communication with host CPUs over National's MICROBUS™ for several COP400 devices).
9. Shift registers.
10. External ROM data storage devices.

In conclusion, National's COP400 series of Microcontrollers provides low-cost solutions to low-end computing and control problems. Proven applications include:

1. Clocks, timers.
2. Laboratory instruments.
3. Radio controllers.
4. Appliance controllers.
5. Programmable sequencers.
6. Scales, cash registers.
7. Calculators.
8. Microcontroller computational elements.
9. Toys and games.
10. Automotive computers.

**Table 1.1 COP400 Device Features**

| | Specifications \ COP: | ROMless Devices | | | Single-Chip Microcontrollers | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 402 | 402M | 404L | 410L | 411L | 420 | 420L | 420C | 421 | 421L | 421C | 444L | 445L |
| MEMORY | ROM x 8 | up to 1024 ext. | | up to 2048 ext. | 512 | | 1024 | | | 1024 | | | 2048 | |
| | RAM x 4 | 64 | | 128 | 32* | | 64* | | | 64* | | | 128* | |
| INPUTS/OUTPUTS | Inputs | 4 | | | 0 | | 4 | | | 0 | | | 4 | 0 |
| | Bidirectional TRI-STATE™ I/O | 8 | | | 8 | | 8 | | | 8 | | | 8 | |
| | Bidirectional I/O | 4 | | | 4 | 3 | 4 | | | 4 | | | 4 | |
| | Outputs | 4 | | | 4 | 2 | 4 | | | 4 | | | 4 | |
| | Serial I/O and External Event Counter | Yes | | | Yes | | Yes | | SIO | Yes | | SIO | Yes | |
| GENERAL | Interrupt | Yes | No | Yes | No | | Yes | | | No | | | Yes | No |
| | Stack Levels | 3 | | | 2 | | 3 | | | 3 | | | 3 | |
| | MICROBUS™ Option | No | Yes | No | No | | Yes | No | Yes | No | | | No | |
| | Instruction Cycle (μs) | 4 | | 16 | 16 | | 4 | 16 | | 4 | 16 | | 16 | |
| POWER/PKG | Supply Voltage | 4.5-6.3 | | 4.5-9.5 | 4.5-6.3*** | | 4.5-6.3 | 4.5-6.3*** | 2.4-6.0 | 4.5-6.3 | 4.5-6.3*** | 2.4-6.0 | 4.5-6.3*** | |
| | Supply Current (mA) | 30 | | 15 | 5 | | 30 | 8 | ** | 30 | 8 | ** | 11 | |
| | Package Size (pins) | 40 | | | 24 | 20 | 28 | | | 24 | | | 28 | 24 |

*RAM keep-alive option (except 420C/421C) **Fast: 800μA, Slow: 35μA, Sleep: 15μA ***4.5-9.5V optionally available

# 2 COP400 Architecture

This chapter provides information on the architecture of the COP400 Microcontrollers. Consistent with the general approach of this manual, the COP420 is primarily discussed with the COP421 treated in terms of differences with respect to the COP420. The COP410L, COP411L and COP444L are similarly treated. The text, therefore, primarily discusses the internal architecture of the COP420, with differences noted for the other devices. Also briefly discussed are different versions of each primary device (e.g., for the COP420, the COP420L and COP420C). As these additional devices, as well as the most inclusive COP400 device, the COP440, become available, further information will be provided in data sheets for each part.

## 2.1 COP420/COP421 Architecture

Figure 2.1 provides a block diagram of the COP420/COP421. It is intended to acquaint the user with the functions of, and interconnections among, the various logic blocks within the processor. Data paths are illustrated in simplified form to depict how the logic elements communicate with each other in implementing the instruction set of the devices. Note that the $IN_3$-$IN_0$ general purpose inputs are not available on the COP421, nor are the two internal IL latches associated with $IN_3$ and $IN_0$.

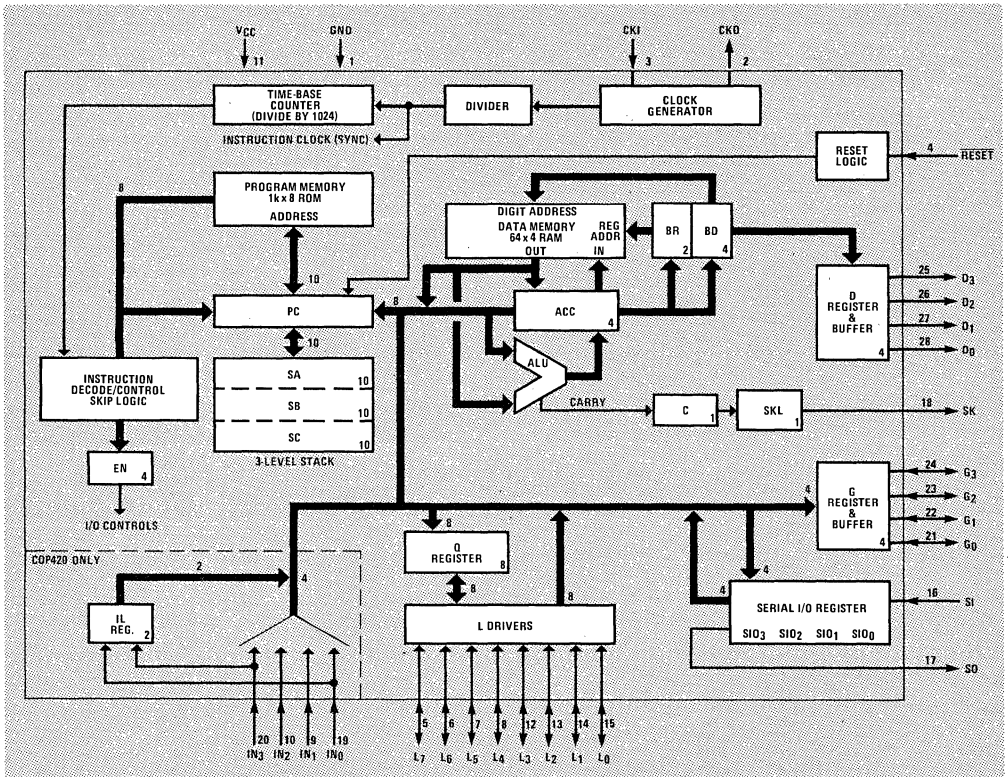



Figure 2.1 COP420/COP421 Series Block Diagram

Figure 2.2 shows the connection diagrams for the 28-pin COP420 and the 24-pin COP421. Figure 2.3 provides a pin description for the COP420/COP421 devices.

One should consult the COP420/COP421 data sheet for maximum ratings, DC and AC electrical characteristics for these devices.

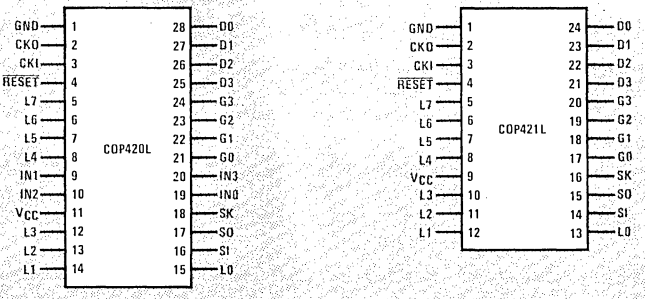



Figure 2.2 COP420/COP421 Connection Diagrams

| | |
|---|---|
| $L_7$-$L_0$ | 8 bidirectional I/O ports with TRI-STATE® |
| $G_3$-$G_0$ | 4 bidirectional I/O ports |
| $D_3$-$D_0$ | 4 general purpose outputs |
| $IN_3$-$IN_0$ | 4 general purpose inputs (COP420 only) |
| SI | Serial input (or counter input) |
| SO | Serial output (or general purpose output) |
| SK | Logic-controlled clock (or general purpose output) |
| CKI | System oscillator input |
| CKO | System oscillator output (or general purpose input or RAM power supply) |
| $\overline{\text{RESET}}$ | System reset input |
| $V_{CC}$ | Power Supply |
| GND | Ground |

**Figure 2.3 COP420/COP421 Pin Description**

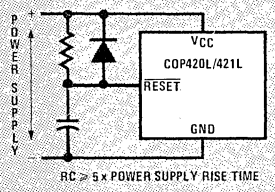



Figure 2.4 Power-Up Clear Circuit

## 2.2 COP420/COP421 Functional Description

The following text provides a functional description of the logic elements depicted in the COP420/COP421 block diagram.

### Program Memory

Program memory consists of a 1,024-byte ROM. ROM words may be program instructions, program data or ROM address pointers. Due to the special characteristics associated with the JP and JSRP instructions, ROM must often be conceived of as organized into 16 pages of 64 words (bytes) each. Also, because of the unique operations performed by the LQID and JID instructions, ROM pages must often be thought of as organized into four consecutive blocks of four ROM pages. (For further information on the paging characteristics of these instructions, see Section 4.1.)

ROM addressing is accomplished by the 10-bit P register. Its binary value selects one of the 1,024 8-bit words ($I_7$-$I_0$) contained in ROM. The value of P is automatically incremented by 1 *prior* to the execution of the current instruction to point to the next sequential ROM location, unless the current instruction is a transfer of control instruction. In the latter case, P is loaded with the appropriate non-sequential value to implement the transfer of control operation performed by the instruction. It should be noted that P will automatically "roll-over" to point to the next page of program memory. This feature has particular significance for transfer of control instructions with paging restrictions, i.e., JP, JSRP, JID and LQID. Since P is incremented to roll-over to the next ROM page *prior* to executing these instructions, they will be treated as residing on the *next* ROM page if they reside in the last word of a ROM page. Further information is provided in Section 4.1.

Three levels of subroutine are implemented by the 10-bit subroutine save registers, SA, SB and SC, providing a last-in, first-out (LIFO) hardware subroutine stack.

ROM instruction words are fetched, decoded and executed by the Instruction Decode, Control and Skip Logic circuitry.

### Data Memory

Data memory consists of a 256-bit RAM, organized as 4 data registers of 16 4-bit digits. RAM addressing is implemented by a 6-bit B register whose upper 2 bits (Br) select 1 of 4 data registers and lower 4 bits (Bd) select 1 of 16 4-bit digits in the selected data register. While the 4-bit contents of the selected RAM digit (M) are usually loaded into or from, or exchanged with, the A register (accumulator), they may also be loaded into or from the Q latches or loaded from the L ports. RAM addressing may also be performed directly by the LDD and XAD instructions based upon the 6-bit contents of the operand field of these instructions. The Bd register also serves as a source register for 4-bit data sent directly to the D outputs.

### Internal Logic

The 4-bit A register (accumulator) is the source and destination register for most I/O, arithmetic, logic and data memory access operations. It can also be used to load the Br and Bd portions of the B register, to load and input 4 bits of the 8-bit Q latch data, to input 4 bits of the 8-bit L I/O port data and to perform data exchanges with the SIO register.

A 4-bit adder performs the arithmetic and logic functions of the COP420, storing results in A. It also outputs a carry bit to the 1-bit C register, most often employed to indicate arithmetic overflow. The C register, in conjunction with the XAS instruction and the EN register, also serves to control the SK output. C can be outputted directly to SKL or can enable SKL to be a SYNC pulse, providing a clock each instruction cycle time. (See XAS instruction, Table 3.1, and EN register description, below.)

Four general-purpose inputs, $IN_3$-$IN_0$, are provided for the COP420: $IN_1$, $IN_2$ and $IN_3$ may be selected, by a mask-programmable option, as Read Strobe, Chip Select and Write Strobe inputs, respectively, for use in MICROBUS™ applications.

The COP421 does not contain the $IN_3$-$IN_0$ inputs and, therefore, must use the 4 bidirectional G I/O ports or 8 bidirectional L I/O ports as input pins to the device. Use of National's MICROBUS is inappropriate with the COP421.

The D register provides 4 general purpose outputs and is used as the destination register for the 4-bit contents of Bd.

The G register contents are output to 4 general-purpose bidirectional I/O ports. The COP420 $G_0$ pin may be mask-programmed as a "ready" output for MICROBUS applications.

The Q register is an internal, latched, 8-bit register, used to hold data loaded to or from M and A, as well as 8-bit program data from ROM. Its contents are output to the L I/O ports when the L drivers are enabled under program control (via an LEI instruction). The COP420 may use the MICROBUS option to write L I/O port data into Q upon the occurrence of a WS pulse from the host CPU.

The 8 L drivers, when enabled, output the contents of latched Q data to the L I/O ports. Also, the contents of L may be read directly into A and M. As explained above, the COP420 MICROBUS option allows L I/O port data to be latched into the Q

register. L I/O ports can be directly connected to the segments of a multiplexed LED display (using the TRI-STATE® LED Direct Drive output configuration option) with Q data being outputted to the Sa-Sg and decimal point segments of the display.

The SIO register functions as a 4-bit serial-in/ serial-out shift register or as a binary counter depending on the contents of the EN register. (See EN register description, below.) Its contents can be exchanged with A, allowing it to input or output a continuous serial data stream. SIO may also be used to provide additional parallel I/O when used as a shift register with its input or output connected to external serial-in/parallel-out shift registers.

The 10-bit time base counter divides the instruction cycle frequency by 1,024, providing a pulse upon overflow. The COP420 SKT instruction tests for the occurrence of this pulse, allowing the programmer to rely on this internal time-base rather than external inputs (e.g., 50/60 Hz signals) to implement "real-time" routines.

The EN register is an internal 4-bit register loaded under program control by the LEI instruction. The state of each bit of this register selects or deselects the particular feature associated with each bit of the EN register ($EN_3$-$EN_0$).

1. The least significant bit of the enable register, $EN_0$, selects the SIO register as either a 4-bit shift register or a 4-bit binary counter. With $EN_0$ set, SIO is an asynchronous binary counter, decrementing its value by one upon each low-going pulse ("1" to "0") occurring on the SI input (count-down counter). Each pulse must be at least two instruction cycles wide. SK outputs the value of C upon execution of XAS and remains latched until the execution of another XAS instruction. The SO output is equal to the value of $EN_3$. With $EN_0$ reset, SIO is a serial shift register shifting left each instruction cycle time. The data present at SI goes into the least significant bit of SIO. SO can be enabled to output the most significant bit of SIO each cycle time. (See Table 2.2 below.) The SK output becomes a logic-controlled clock, providing a SYNC signal each instruction time. It will start outputting a SYNC pulse upon the execution of an XAS instruction with C = 1, stopping upon the execution of a subsequent XAS with C = 0.
2. With $EN_1$ set, the COP420 $IN_1$ input is enabled as an interrupt input. Immediately following an interrupt, $EN_1$ is reset to disable further interrupts. Note that this interrupt feature associated with $IN_1$ is unavailable on the COP421 since it lacks the IN inputs. Bit 1 ($EN_1$) of the Enable Register is, therefore, a "don't care" bit for the COP421: setting or resetting this bit via an LEI instruction will have no effect on the operation of the COP421. (For further information on the procedure and protocol of this COP420 interrupt feature, see Section 3.2, LEI instruction description.)
3. With $EN_2$ set, the L drivers are enabled to output the data in Q to the L I/O ports. Resetting $EN_2$ disables the L drivers, placing the L I/O ports in a high-impedance input state. If the COP420 MICROBUS™ option is being used, $EN_2$ does not affect the L drivers.
4. $EN_3$, in conjunction with $EN_0$, affects the SO output. With $EN_0$ set (binary counter option selected), SO will output the value loaded into $EN_3$. With $EN_0$ reset (serial shift register option selected), setting $EN_3$ enables SO as the output of the SIO shift register, outputting serial shifted data each instruction time. Resetting $EN_3$ with the serial shift register option selected disables SO as the shift register output: data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0." Table 2.1 provides a summary of the options and features associated with $EN_3$ and $EN_0$.

### 2.3 Initialization

Upon initialization of the COP420/COP421 as described below, the P register is cleared to 0 (ROM address 0) and the A, B, C, D, EN, and G registers are cleared. The $IN_0$ and $IN_3$ latches are not cleared. The SK output is enabled as a SYNC output, providing a pulse each instruction cycle time. *Data memory (RAM) can only be cleared by the user's program. The first instruction at address 0 must be a CLRA.*

The Reset Logic, internal to the COP420/COP421, will initialize (clear) the device upon power-up if the power supply rise time is less than 1 ms and greater than 1 μs. If the power supply rise time is greater than 1 ms, the user must provide an external RC network and diode to the $\overline{\text{RESET}}$ pin as shown in Figure 2.4 below. The $\overline{\text{RESET}}$ pin is configured as a Schmitt trigger input. If not used, it should be connected to $V_{CC}$. Initialization will occur whenever a logic "0" is applied to the $\overline{\text{RESET}}$ input, provided it stays low for at least three instruction cycle times. In order to reset the Time Base Counter, a $\overline{\text{RESET}}$ pulse ten instruction cycle times wide must be applied; note that the counter will overflow and generate an output pulse.

Table 2.1 Enable Register Modes — Bits $EN_3$ and $EN_0$

| $EN_3$ | $EN_0$ | SIO | SI | SO | SK after XAS |
|---|---|---|---|---|---|
| 0 | 0 | Shift Register | Input to Shift Register | 0 | If C = 1, SK = SYNC<br>If C = 0, SK = 0 |
| 1 | 0 | Shift Register | Input to Shift Register | Serial Out | If C = 1, SK = SYNC<br>If C = 0, SK = 0 |
| 0 | 1 | Binary Counter | Input to Binary Counter | 0 | If C = 1, SK = 1<br>If C = 0, SK = 0 |
| 1 | 1 | Binary Counter | Input to Binary Counter | 1 | If C = 1, SK = 1<br>If C = 0, SK = 0 |

## 2.4 COP420/COP421 Mask Programmable Options

To allow even greater flexibility in specifying a COP400 device appropriate to the user's application, all COP400 microcontrollers have specific clock configuration, I/O and other mask-programmable options associated with them. These options are masked into the part simultaneously with the masking of the user's program in ROM and have been chosen to offer the user a wide range of options which encompasses design options most frequently employed in dedicated, small system applications.

The following text summarizes the COP420/COP421 options according to the various functions (oscillator, I/O, etc.) with which they are associated.

### Clock Oscillator Options

There are four basic COP420/COP421 clock oscillator configurations avilable as shown by Figure 2.5 (a-d):

a. Crystal Controlled Oscillator. CKI and CKO are connected to an external crystal. The instruction cycle time equals the crystal frequency (4 MHz maximum) divided by 16 (optional by 8).

b. External Oscillator. CKI is configured as a TTL compatible input accepting an external clock signal. The external frequency (4 MHz maximum) is divided by 16 (optional by 8) to derive the instruction cycle time. CKO is now available to be used as the RAM power supply ($V_R$) pin, as a general purpose input, or as a synchronizing input.

c. RC Controlled Oscillator. CKI is configured as a single-pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 4. CKO is available for non-timing functions as in b above.

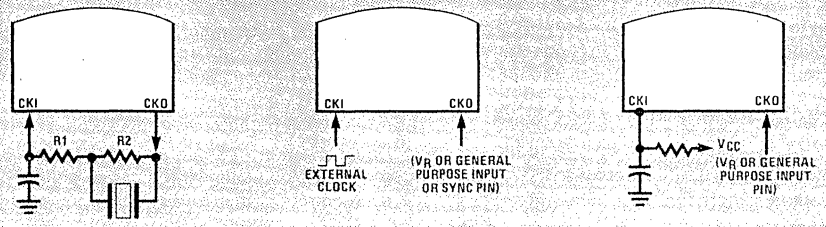
d. Externally Synchronized Oscillator. Intended for use in multi-COP systems, CKO is programmed to function as an input connected to the SK output of another COP420/COP421 with CKI connected as shown. In this configuration, the SK output connected to CKO must provide a SYNC (instruction cycle) signal to CKO, thereby allowing synchronous data transfer between the COPs using only the SI and SO serial I/O pins in conjunction with the XAS instruction. Note that on power-up SK is automatically enabled as a SYNC output. (See Initialization, above.)

The lower portion of Figure 2.5 provides component values for several instruction cycle times and crystal values associated with the RC controlled and Crystal Oscillator options, respectively.

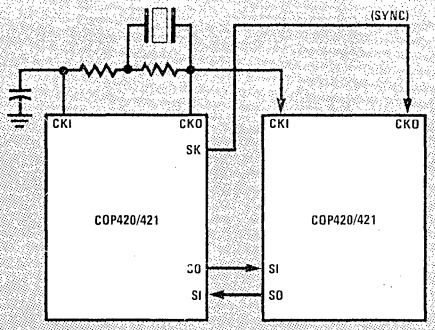
### CKO Non-Timing Options

In a crystal controlled or multi-COP oscillator system, CKO is used as an output to the crystal network. In the other two configurations (external clock or RC controlled oscillator), CKO may be mask-programmed to perform one of two available options. Specifically, CKO may be mask-programmed as a general purpose input, read into bit 1 of the accumulator ($A_2$) upon the execution of an INIL instruction.

As another option (for both the COP420 and COP421), CKO can be a RAM power supply pin ($V_R$), allowing its connection to a standby/backup power supply to maintain the integrity of RAM data with minimum power drain when the main supply is inoperative or shut down to conserve power. Use of this options should include external circuitry to detect loss of $V_{CC}$ power and force $\overline{\text{RESET}}$ low before $V_{CC}$ drops below spec.




a. Crystal Oscillator b. External Oscillator c. RC Controlled Oscillator




d. Externally Synchronized Oscillator

**Crystal Oscillator**

| Crystal Value | Component Values | | |
|---|---|---|---|
| | R1 | R2 | C |
| 4 MHz | 1k | 1M | 27 pF |
| 3.58 MHz | 1k | 1M | 27 pF |
| 2.09 MHz | 1k | 1M | 56 pF |

**RC Controlled Oscillator**

| R (kΩ) | C (pF) | Instruction Cycle Time (μs) |
|---|---|---|
| 12 | 100 | 5 ±20% |
| 6.8 | 220 | 5.3 ±23% |
| 8.2 | 300 | 8 ±29% |
| 22 | 100 | 8.6 ±16% |

Figure 2.5 COP420/COP421 Oscillator Configurations

### MICROBUS™ Option

The COP420 has an option which allows it to be used as a peripheral microprocessor device, inputting and outputting data from and to a host microprocessor ($\mu$P). $IN_1$, $IN_2$, and $IN_3$ general purpose inputs become MICROBUS compatible read-strobe, chip-select, and write-strobe lines, respectively. $IN_1$ becomes $\overline{RD}$ — a logic "0" on this input will cause Q latch data to be enabled to the L ports for input to the $\mu$P. $IN_2$ becomes $\overline{CS}$ — a logic "0" on this line selects the COP420 as the $\mu$P peripheral device by enabling the operation of the $\overline{RD}$ and $\overline{WR}$ lines and allows for the selection of one of several peripheral components. $IN_3$ becomes $\overline{WR}$ — a logic "0" on this line will write bus data from the L ports to the Q latches for input to the COP420. $G_0$ becomes a "ready" output, reset by a write pulse from the $\mu$P on the $\overline{WR}$ line, providing the "handshaking" capability necessary for asynchronous data transfer between the host CPU and the COP420.

This option has been designed for compatibility with National's MICROBUS — a standard interconnect system for 8-bit parallel data transfer between MOS/LSI CPUs and interfacing devices. (See *MICROBUS™*, National Publication.) The functioning and timing relationships between the COP420 signal lines affected by this option are as specified for the MICROBUS interface. Connection of the COP420 to the MICROBUS is shown in Figure 5.13.

### I/O Options

COP420/421 outputs have the following optional configurations, illustrated in figure 2.6:

**a. Standard —** an enhancement mode device to ground in conjunction with a depletion-mode device to $V_{CC}$, compatible with TTL and CMOS input requirements. Available on SO, SK, and all D and G outputs.

**b. Open-Drain —** an enhancement-mode device to ground only, allowing external pull-up as required by the user's application. Available on SO, SK, and all D and G outputs.

**c. Push-Pull —** An enhancement-mode device to ground in conjunction with a depletion-mode device paralleled by an enhancement-mode device to $V_{CC}$. This configuration has been provided to allow for fast rise and fall times when driving capacitive loads. Available on SO and SK outputs only.

**d. Standard L —** same as **a.**, but may be disabled. Available on L outputs only.

**e. Open Drain L —** same as **b.**, but may be disabled. Available on L outputs only.

**f. LED Direct Drive —** an enhancement-mode device to ground and to $V_{CC}$, meeting the typical current sourcing requirements of the segments of an LED display. the sourcing device is clamped to limit current flow. These devices may be turned off under program control (See Functional Description, EN Register), placing the outputs in a high-impedance state to provide required LED segment blanking for a multiplexed display.

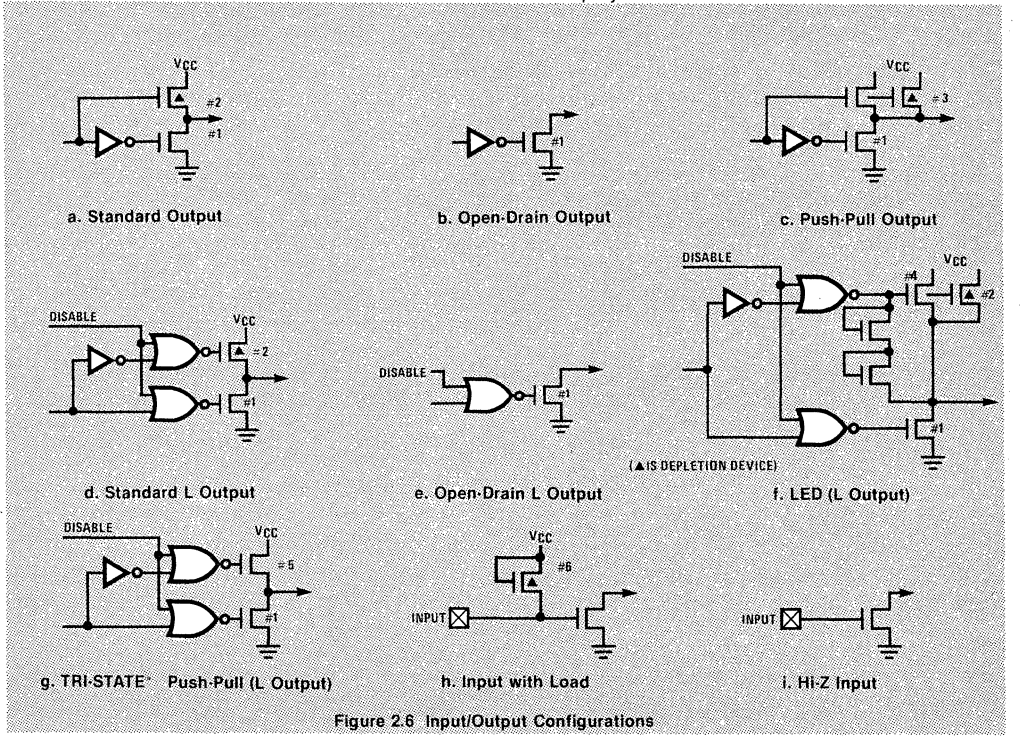



Figure 2.6 Input/Output Configurations

**g. TRI-STATE® Push-Pull** — an enhancement-mode device to ground and $V_{CC}$. These outputs are TRI-STATE outputs, allowing for connection of these outputs to a data bus shared by other bus drivers.

COP420/COP421 inputs have the following optional configurations:

**h.** An on-chip depletion load device to $V_{CC}$.

**i.** A Hi-Z input which must be driven to a "1" or "0" by external components.

The above input and output configurations share common enhancement-mode and depletion-mode devices. Specifically, all configurations use one or more of six devices (numbered 1-6, respectively). Minimum and maximum current ($I_{OUT}$ and $V_{OUT}$) curves are given in figure 2.7 for each of these devices to allow the designer to effectively use these I/O configurations in designing a COP420/421 system.

The SO, SK outputs can be configured as shown in **a.**, **b.**, or **c.** The D and G outputs can be configured as shown in **a.** or **b.** Note that when inputting data to the G ports, the G outputs should be set to "1." The L outputs can be configured as in **d.**, **e.**, **f.** or **g.**

An important point to remember if using configuration **d.** or **f.** with the L drivers is that even when the L drivers are disabled, the depletion load device will source a small amount of current (see figure 2.7, device 2); however, when the L lines are used as inputs, the disabled depletion device can *not* be relied on to source sufficient current to pull an input to logic "1".

*All* of the L driver options are TRI-STATE® -able. Therefore, the L drivers have TRI-STATE-able Standard and Open-Drain output options as well as the TRI-STATE LED Direct Drive and Push-Pull output options. Since the device to $V_{CC}$ in the Standard output configuration is a depletion-mode

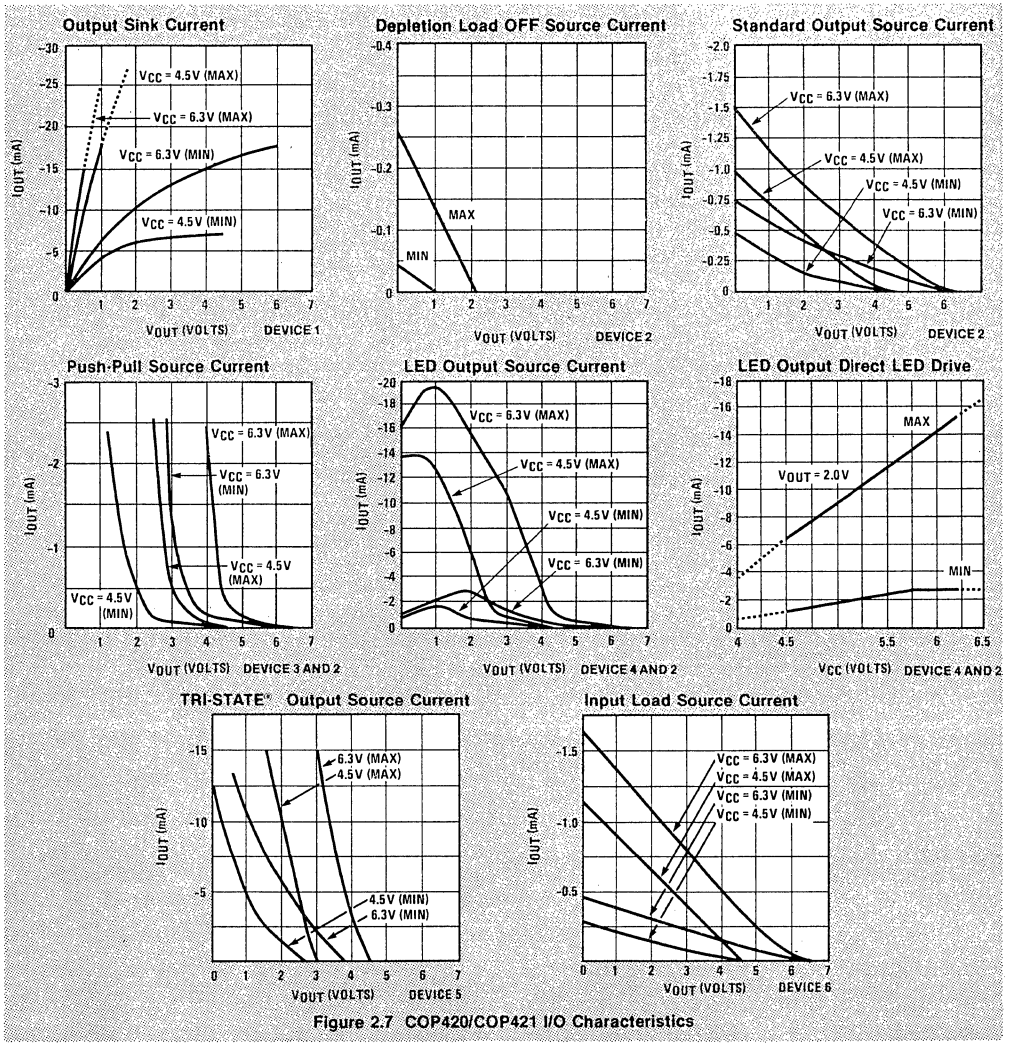



**Figure 2.7 COP420/COP421 I/O Characteristics**

device, it will source up to 0.125 mA when this output is "turned off" in the TRI-STATE mode. This is not a worst case input for a logic "1" level on these inputs and will not be sufficient for an input level without previously enabling Q to L with $(Q) = FF_{16}$.

**Bonding Option**

The COP421 is a bonding option of the COP420: if the COP420 is bonded as a 24-pin device (without the 4 IN inputs), it becomes the COP421. Note that since it lacks the IN inputs, use of the COP421 bonding option precludes use of the IN input options; the MICROBUS™ option which would otherwise affect $IN_3$-$IN_1$ and $G_0$: use of the $IN_1$ hardware interrupt pin and the use of the $IL_3$ and $IL_0$ latches associated with the $IN_3$ and $IN_0$ pins. All other options are available. The COP421 is pin-compatible with the COP410L.

## 2.5 COP420/COP421 Option List

The COP420/COP421 mask-programmable options are assigned numbers which correspond with the *COP420* pins.

The following is a list of COP420 options. When specifying a COP421 chip, Options 9, 10, 19, 20, and 29 must all be set to zero. The options are programmed at the same time as the ROM pattern in order to provide the user with the hardware flexibility to interface the COP420/COP421 to various I/O components using little or no external circuitry.

Option 1 = 0: Ground Pin — no options available

Option 2: CKO Pin
- = 0: clock generator output to crystal
- = 1: pin is RAM power supply ($V_R$) input
- = 2: general purpose input with load device to $V_{CC}$
- = 3: multi-COP SYNC input
- = 4: general purpose hi-Z input

Option 3: CKI Input
- = 0: crystal input divided by 16 (crystal = 4 MHz maximum)
- = 1: crystal input divided by 8 (crystal = 2 MHz maximum)
- = 2: TTL external clock input divided by 16 (input = 4 MHz maximum)
- = 3: TTL external clock input divided by 8 (input = 2 MHz maximum)
- = 4: single-pin RC controlled oscillator

Option 4: $\overline{\text{RESET}}$ Pin
- = 0: load device to $V_{CC}$
- = 1: hi-Z input

Option 5: $L_7$ Driver
- = 0: standard output (Figure 2.6a)
- = 1: open-drain output (Figure 2.6b)
- = 2: LED direct drive output (Figure 2.6d)
- = 3: TRI-STATE® push-pull output (Figure 2.6e)

Option 6: $L_6$ Driver
same as Option 5

Option 7: $L_5$ Driver
same as Option 5

Option 8: $L_4$ Driver
same as Option 5

Option 9: $IN_1$ Input
- = 0: load device to $V_{CC}$ (Figure 2.6f), std. TTL input levels ("0" = 0.8V, "1" = 2.0V) (mandatory value for COP421)
- = 1: hi-Z input (Figure 2.6g), std. TTL input levels ("0" = 0.8V, "1" = 2.0V)
- = 3: hi-Z input (2.6g), higher trip levels ("0" = 1.2V, "1" = 3.6V) — use for all or none of IN inputs

Option 10: $IN_2$ Input
same as Option 9

Option 11 = 0: $V_{CC}$ pin — no options available

Option 12: $L_3$ Driver
same as Option 5

Option 13: $L_2$ Driver
same as Option 5

Option 14: $L_1$ Driver
same as Option 5

Option 15: $L_0$ Driver
same as Option 5

Option 16: SI Input
- = 0: load device to $V_{CC}$ (2.6f)
- = 1: hi-Z input (2.6g)

Option 17: SO Driver
- = 0: standard output (Figure 2.6a)
- = 1: open-drain output (Figure 2.6b)
- = 2: push-pull output (Figure 2.6c)

Option 18: SK Driver
same as Option 17

Option 19: $IN_0$ Input
same as Option 9

Option 20: $IN_3$ Input
same as Option 9

Option 21: $G_0$ I/O Port
- = 0: standard output (Figure 2.6a)
- = 1: open-drain output (Figure 2.6b)
- = 2: standard output, small driver (⅓ current)
- = 3: open-drain output, small driver (⅓ current)

Option 22: $G_1$ I/O Port
same as Option 21

Option 23: $G_2$ I/O Port
same as Option 21

Option 24: $G_3$ I/O Port
same as Option 21

Option 25: $D_3$ Output
= 0: standard output (Figure 2.6a)
= 1: open-drain output (Figure 2.6b)

Option 26: $D_2$ Output
same as Option 25

Option 27: $D_1$ Output
same as Option 25

Option 28: $D_0$ Output
same as Option 25

Option 29: COP Function
= 0: normal operation (mandatory value for COP421)
= 1: MICROBUS™ option

Option 30: COP Bonding
= 0: COP420 (28-pin device)
= 1: COP421 (24-pin device)

Option 31: IN Input Levels — no option available
(see Options 9, 10, 19, 20)

Option 32: G Input Levels
= 0: standard TTL input levels ("0" = 0.8V, "1" = 2.0V)
= 1: higher trip levels ("0" = 1.2V, "1" = 3.6V)

Option 33: L Input Levels
same as Option 32

Option 34: CKO Input Level
(CKO = input; Option 2 = 2, 3, 4)
same as Option 32

Option 35: SI Input Level
same as Option 32

## 2.6 COP420L/COP421L Description

The COP420L/COP421L are low power versions of the COP420/COP421 containing the *same* internal logic elements and instruction set as the COP420/COP421, with *electrical* characteristics which are similar to the COP410L. The major differences between the COP420L/COP421L and COP420/COP421 are the following:

- Wider operating voltage range of 4.5 to 9.5V optionally available.
- Operating supply current less than 8mA @ $V_{CC}$ = 5V.
- Minimum instruction cycle time of 15μs.
- Divide-by-32 crystal clock option (2MHz XTAL divided by 32 = 15μs instruction cycle time).
- D and G outputs have direct LED digit drive option (sink 30mA).
- Other outputs will drive 1 LSTTL or 2 LPTTL loads ($I_{OL}$ = 360μA at 0.4V; $I_{OH}$ = 40μA at 2.4V).
- No MICROBUS™ option available.

The COP421L is simply a COP420L packaged in a 24-pin dual-in-line package. As a result, the IN inputs are not available on the COP421L, so that the COP421L is pin-compatible with the COP410L.

For further information, see the COP420L/COP421L data sheet.

## 2.7 COP420L/COP421L Mask Programmable Options

Since the COP420L/COP421L are frequently used in battery-operated and/or hand-held consumer-type products, an even greater array of system-cost-reducing options is available. The following text summarizes these options.

### Clock Oscillator Options

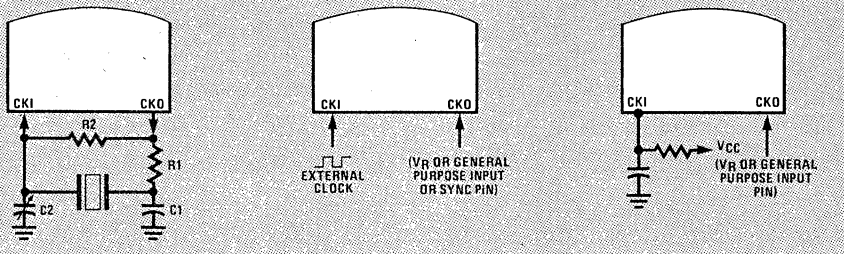
There are four basic COP420L/COP421L clock oscillator configurations available as shown in Figure 2.8 (a-d):

a. Crystal/Resonator Controlled Oscillator. CKI and CKO are connected to an external crystal or ceramic resonator. The instruction cycle time equals the crystal/resonator frequency (2.097MHz maximum) divided by 32 (optional by 16 or 8).

b. External Oscillator. CKI is configured as a CMOS compatible input accepting an external clock signal. The external frequency (2MHz maximum) is divided by 32 (optional by 16, 8 or 4) to derive the instruction cycle time. CKO is now available to be used as the RAM power supply ($V_R$) pin, as a COP420L general purpose input, or as a synchronizing input.

c. RC Controlled Oscillator. CKI is configured as a single-pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillation frequency divided by 4. CKO is available for non-timing functions as in b above.

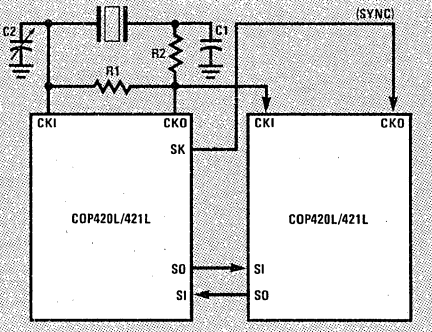
d. Externally Synchronized Oscillator. Intended for use in multi-COP systems, CKO is programmed to function as an input connected to the SK output of another COP420L/COP421L with CKI connected as shown. In this configuration, the SK output connected to CKO must provide a SYNC (instruction cycle) signal to CKO, thereby allowing synchronous data transfer between the COPs using only the SI and SO serial I/O pins in conjunction with the XAS instruction. Note that on power-up SK is automatically enabled as a SYNC output.




a. Crystal Oscillator

b. External Oscillator

c. RC Controlled Oscillator




d. Externally Synchronized Oscillator

**Crystal Oscillator**

| Crystal Value | Component Values | | | |
|---|---|---|---|---|
| | R1 | R2 | C1 | C2 |
| 455 kHz (Resonator) | 16k | 1M | 80 pF | 80 pF |
| 2.09 MHz | 1k | 1M | 56 pF | 6-36 pF |

**RC Controlled Oscillator**

| R (kΩ) | C (pF) | Instruction Cycle Time (μs) |
|---|---|---|
| 51 | 100 | 19 ± 15% |
| 82 | 56 | 19 ± 15% |

Figure 2.8 COP420L/COP421L Oscillator Configurations

The lower portion of Figure 2.8 provides component values for several instruction cycle times and crystal values associated with the RC controlled and crystal controlled oscillator options, respectively.

### CKO Non-Timing Options

In a crystal controlled or multi-COP oscillator system, CKO is used as an output to the crystal network. In the other two configurations (external clock or RC controlled oscillator), CKO may be mask-programmed to perform one of two available options. Specifically, CKO may be mask-programmed as a general purpose COP420L input, read into bit 1 of the accumulator ($A_2$) upon the execution of an INIL instruction.

As another option (for both the COP420L and COP421L), CKO can be a RAM power supply pin ($V_R$), allowing its connection to a standby/backup power supply to maintain the integrity of RAM data with minimum power drain when the main supply is inoperative or shut down to conserve power.

### I/O Options

While the COP420L/COP421L has capabilities to directly drive LED displays through increased voltage and current specs, the circuit configurations are identical to those of the COP420 in Figure 2.6. Increased current sink and source values are a result of changing device sizes (within the bounds of the same circuit configuration). When emulating the COP420L with the COP402, one might use the typical values of the 402 as worst case COP420L drive parameters. An alternative is the use of the COP404L to emulate the drive of the COP420L.

For detailed electrical characteristics, refer to the COP420L/COP421L data sheet.

The SO and SK outputs can be configured as shown in Figure 2.6, a, b, or c. The D and G outputs can be configured as shown in a or b. Note that when inputting data to the G ports, the G outputs should be set to "1." The L outputs can be configured as shown in d, e, f, or g.

An important point to remember is that *all* of the L driver options are TRI-STATE® -able. Therefore, the L drivers have TRI-STATE-able Standard and Open-Drain output options as well as the TRI-STATE LED Direct Drive and Push-Pull output options. Since the device to $V_{CC}$ in the Standard output configuration is a depletion-mode device, it will source up to 0.125 mA when this output is "turned off" in the TRI-STATE mode,which is insufficient to guarantee a logic "1" input level.

### Bonding Option

The COP421L is a bonding option of the COP420L: if the COP420L is bonded as a 24-pin device (without the 4 IN inputs), it becomes the COP421L. The COP421L is pin-compatible with the COP410L.

## 2.8 COP420L/COP421L Option List

The COP420L/COP421L mask-programmable options are assigned numbers which correspond with the *COP420L* pins.

The following is a list of COP420L options. When specifying a COP421L chip, Options 9, 10, 19, and 20 must all be set to zero. The options are programmed at the same time as the ROM pattern in order to provide the user with the hardware flexibility to interface the COP420L/COP421L to various I/O components using little or no external circuitry.

Option 1 = 0: Ground Pin — no options available

Option 2: CKO Output
- = 0: clock generator output to crystal/resonator
- = 1: pin is RAM power supply ($V_R$) input
- = 2: general purpose input with load device to $V_{CC}$
- = 3: general purpose hi-Z input
- = 4: multi-COP SYNC input (CKI ÷ 32, CKI ÷ 16)
- = 5: multi-COP SYNC input (CKI ÷ 8)

Option 3: CKI Input
- = 0: oscillator input divided by 32 (2 MHz maximum)
- = 1: oscillator input divided by 16 (1 MHz maximum)
- = 2: oscillator input divided by 8 (500 kHz maximum)
- = 3: single-pin RC controlled oscillator divided by 4

Option 4: $\overline{\text{RESET}}$ Input
- = 0: load device to $V_{CC}$
- = 1: hi-Z input

Option 5: $L_7$ Driver
- = 0: standard output (Figure 2.6d)
- = 1: open-drain output (Figure 2.6e)
- = 2: high current LED direct segment drive output (Figure 2.6f)
- = 3: high current TRI-STATE® push-pull output (Figure 2.6g)
- = 4: low current LED direct segment drive output (Figure 2.6f)
- = 5: low current TRI-STATE push-pull output (Figure 2.6g)

Option 6: $L_6$ Driver
same as Option 5

Option 7: $L_5$ Driver
same as Option 5

Option 8: $L_4$ Driver
same as Option 5

Option 9: $IN_1$ Input
= 0: load device to $V_{CC}$ (Figure 2.6h) (mandatory value for COP421L)
= 1: hi-Z input (Figure 2.6i)

Option 10: $IN_2$ Input
same as Option 9

Option 11 = 0: $V_{CC}$ pin
= 0: 4.5V to 6.3V operation
= 1: 4.5V to 9.5V operation (extra cost option)

Option 12: $L_3$ Driver
same as Option 5

Option 13: $L_2$ Driver
same as Option 5

Option 14: $L_1$ Driver
same as Option 5

Option 15: $L_0$ Driver
same as Option 5

Option 16: SI Input
same as Option 9

Option 17: SO Driver
= 0: standard output (Figure 2.6a)
= 1: open-drain output (Figure 2.6b)
= 2: push-pull output (Figure 2.6c)

Option 18: SK Driver
same as Option 17

Option 19: $IN_0$ Input
same as Option 9

Option 20: $IN_3$ Input
same as Option 9

Option 21: $G_0$ I/O Port
= 0: very high current standard output (Figure 2.6a)
= 1: very high current open-drain output (Figure 2.6b)
= 2: high current standard output
= 3: high current open-drain output
= 4: standard LSTTL output (fanout = 1)
= 5: open-drain LSTTL output (fanout = 1)

Option 22: $G_1$ I/O Port
same as Option 21

Option 23: $G_2$ I/O Port
same as Option 21

Option 24: $G_3$ I/O Port
same as Option 21

Option 25: $D_3$ Output
same as Option 21

Option 26: $D_2$ Output
same as Option 21

Option 27: $D_1$ Output
same as Option 21

Option 28: $D_0$ Output
same as Option 21

Option 29: L Input Levels
= 0: standard TTL input levels ("0" = 0.8V, "1" = 2.0V)
= 1: higher voltage input levels ("0" = 1.2V, "1" = 3.6V)

Option 30: IN Input Levels
same as Option 29

Option 31: G Input Levels
same as Option 29

Option 32: SI Input Levels
same as Option 29

Option 33: $\overline{\text{RESET}}$ Input
= 0: Schmitt trigger input
= 1: standard TTL input levels
= 2: higher voltage input levels

Option 34: CKO Input Levels
(CKO = input; Option 2 = 2, 3)
same as Option 29

Option 35: COP Bonding
= 0: COP420L (28-pin device)
= 1: COP421L (24-pin device)

## 2.9 COP420C Description

The COP420C is a CMOS version of the COP420. It differs from the COP420 primarily in electrical specifications; however, it also features a dual clock mode option for operation at low speed (typically 244 $\mu$s instruction cycle time) with low power consumption (25 $\mu$A with $V_{CC}$ = 2.4V) or high speed (15 $\mu$s instruction cycle time) when necessary to perform internal data computations at a faster rate. The COP420C has the same output drive characteristics as the COP420 (TTL/CMOS compatible) and retains the MICROBUS™ option. The following are the major differences between the COP420C and the COP420:

- Operating voltage of 2.4V to 6.0V.
- Low power consumption at 244 $\mu$s instruction cycle time (inexpensive 32 kHz XTAL ÷ 8) = 25 $\mu$A at $V_{CC}$ = 2.4V.

- Dual clock mode option allowing operation at 16μs instruction cycle time (using external RC network) for internal data computation operations.
- "Fast" clock mode entered under program control.

For further information, see the COP420C data sheet.

### 2.10 COP444L/COP445L Description

The COP444L/COP445L are expanded-memory versions of the COP420L containing the same internal logic elements and instruction set as the COP420 and COP420L, but with twice the amounts of ROM and RAM. The major differences between the COP444L/COP445Land the COP420L/COP421L are the following:

- Operating supply current less than 11mA at $V_{CC} = 5V$.
- 2048 × 8 ROM.
- 128 × 4 RAM.

The COP445L is simply a COP444L in a 24-pin dual-in-line package. As a result, the IN inputs are not available on the COP445L, so that the COP445L is pin-compatible with the COP421L and COP410L.

These devices are emulated using the COP404L.

For further information, see the COP444L/445L and/or COP404L data sheets.

### 2.11 COP402 and COP402M ROM-Less Parts Description

The COP402 and COP402M are ROM-less versions of the COP420. They are packaged in 40-pin packages and are available for prototyping a COP420 system using the COP400 Development System (PDS) or, in quantity, for small volume applications using external ROM.

The COP402 has been mask programmed with options suitable for use as a general controller. COP402 inputs have load devices to $V_{CC}$, the various outputs have the fullest drive capability associated with them (L outputs = LED direct drive; G and D outputs = standard; SO, SK outputs = push-pull). The COP402 has been programmed for use with an external crystal network, using CKI and CKO, with an instruction cycle time equal to the crystal frequency divided by 16.

The COP402M is the MICROBUS™ compatible version of the COP402. It features the same options as the COP402 with the single exception that the MICROBUS option has been selected. It is, of course, intended for use in prototyping systems or small volume applications which use the microcontroller as a CPU peripheral component, with communication over National's MICROBUS.

### 2.12 COP404L ROM-Less Part Description

The COP404L is a ROM-less version of the COP444L. It is packaged in a 40-pin package and may be used to prototype all low-power COP400 devices (COP411L, COP410L, COP420L, COP421L, COP444L).

### 2.13 COP410L/COP411L Architecture

Figure 2.9 provides a block diagram of the COP410L/COP411L. As with the COP420/COP421 block diagram, it depicts the internal logic and interconnects of the device in simplified form. Note that the COP410L is functionally a subset of the 24-pin COP421L. As with the COP421L, it lacks the COP420L IN inputs and the internal IL latches associated with two of these deleted input pins. These and other architectural differences are discussed in the Functional Description, below.

Figure 2.10 shows the Connection Diagrams for the 24-pin COP410L and the 20-pin COP411L. Figure 2.11 provides a pin description for the COP410L/COP411L devices.

See data sheet for the electrical specifications of the COP410L/COP411L, showing maximum ratings plus DC and AC characteristics for these devices.

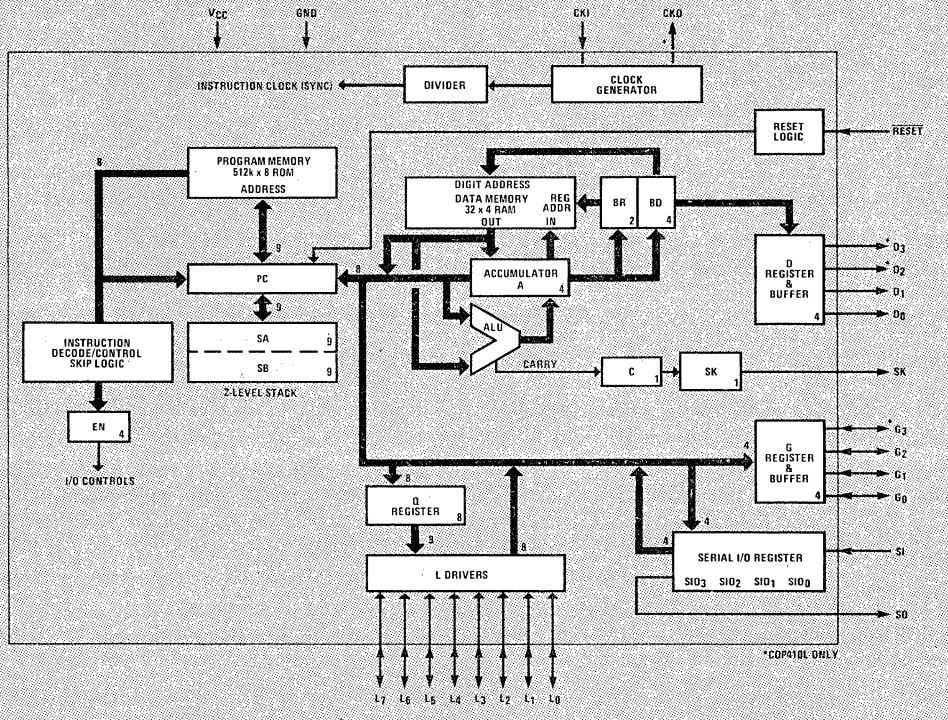



Figure 2.9 COP410L/COP411L Block Diagram

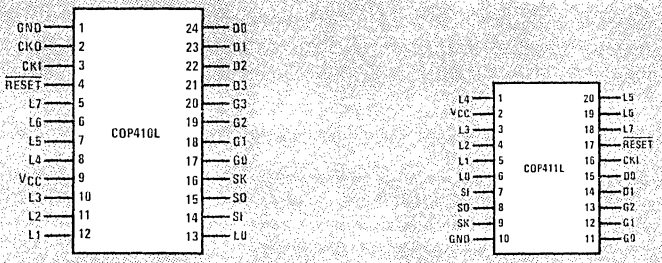



Figure 2.10 COP410L/COP411L Connection Diagrams

| | | | |
|---|---|---|---|
| $L_7$-$L_0$ | 8 bidirectional I/O ports with TRI-STATE® | CKI | System oscillator input |
| $G_3$-$G_0$ | 4 bidirectional I/O ports | CKO | System oscillator output (or RAM power supply) |
| $D_3$-$D_0$ | 4 general purpose outputs | $\overline{\text{RESET}}$ | System reset input |
| SI | Serial input (or counter input) | $V_{CC}$ | Power supply |
| SO | Serial output (or general purpose output) | GND | Ground |
| SK | Logic-controlled clock (or general purpose output) | | |

Figure 2.11 COP410L/COP411L Pin Description

## 2.14 COP410L/COP411L Functional Description

The following text provides a functional description of the differences which exist between the internal architecture of the COP420, covered in detail in Section 2.2, and that of the COP410L and COP411L. Consequently, for information on logic elements not discussed below which appear in Figure 2.10, COP410L/COP411L Block Diagram, refer to Section 2.2. Where appropriate, differences between the COP410L and its smaller version, the COP411L, are noted in the following text.

### Program Memory

Program memory consists of a 512-byte ROM. The same paging characteristics apply to the COP410L/COP411L when allocating program memory instruction code as those which apply to the COP420 (see Section 4.1) except that ROM consists of *8* (0-7) pages of 64 (0-63) words each.

ROM addressing is accomplished by a *9-bit* P register. The auto increment-before-execution and page-rollover features of the COP420 apply to the COP410L/COP411L.

Since the COP410L/COP411L have *2* 9-bit subroutine-save registers, SA and SB, subroutine nesting is allowable to two levels (only one level when executing a LQID instruction since this instruction pushes the stack).

### Data Memory

Data memory consists of a 128-bit RAM organized as 4 (0-3) data registers of 8 4-bit digits. Digit addressing is valid only for digits 0, 9-15 in a particular register. (The COP410L/COP411L will, however, treat digit addresses of 1-7 as valid digit values of 9-15, respectively.) As with the COP420, RAM addressing is accomplished by a 6-bit B register whose upper 2 bits (Br) select 1 of 4 data registers and lower 3 bits (Bd) select 1 of *8* 4-bit digits.

A direct access to data memory, without using the B register, is only permissible with respect to M(3, 15) by using an XAD 3, 15 instruction. All other XAD and all LDD instructions have been deleted from the COP410L/COP411L instruction set. Consequently, all other RAM locations must be accessed by loading the B register with the address of data memory to be accessed.

As with the COP420, Bd also may be used as a source register to output its 4-bit contents directly to the D outputs via an OBD instruction.

The Q register functions in a similar manner as the COP420 Q register with the following exceptions:

1. Its contents must be read with the INL instruction, since the CQMA instruction has been deleted.
2. It cannot be loaded with the contents of the L I/O ports since this function is associated with the deleted MICROBUS™ option.

The COP410L/COP411L does not contain the COP420 internal divide-by-1024 time-base counter; hence, the SKT instruction has been deleted. "Real-time" program counters must, therefore, rely on an external time-base input (e.g., 50/60 Hz square wave) to derive a program "clock" for such applications, rather than on the COP410L/COP411L instruction cycle clock itself.

Bit 1 of the EN register ($EN_1$) is a "don't care" bit, as explained above, due to the lack of a COP410L/COP411L $IN_1$ input. (The COP420 uses the $EN_1$ bit to enable $IN_1$ as an interrupt signal.)

The CASC, ADT and OGI instructions have been deleted. See Section 3.4 for hints on performing these functions.

## 2.15 COP410L/COP411L Mask Programmable Options

The following text describes the differences which exist between the COP420L mask programmable options and those which are available for the COP410L and COP411L devices.

Available clock oscillator configurations are as follows:

a. Ceramic Resonator Controlled Oscillator. CKI and CKO are connected to an external ceramic resonator. The instruction cycle time equals the resonator frequency (500 kHz maximum) divided by 8. This configuration and its associated options are not available on the 20-pin COP411L since it lacks the CKO pin.

b. External Oscillator. CKI is configured as a Schmitt trigger input (not TTL compatible), accepting an external clock signal. The external frequency (500 kHz maximum) is divided by 8 to derive the instruction cycle time. This option applies to both the COP410L and the COP411L. For the COP410L, moreover, this configuration allows CKO to be used for a RAM power supply ($V_R$).

c. RC Controlled Oscillator. CKI is configured as a single pin RC controlled Schmitt trigger oscillator. The instruction cycle equals the oscillator (RC time-constant) frequency divided by 4.

d. Externally Synchronized Oscillator. CKO is configured as a synchronizing input from the SK output of another COP400 device. CKI is an external oscillator (divide by 8).

The lower portion of Figure 2.12 provides component values associated with the RC controlled oscillator option.

### COP410L CKO Non-Timing Options

In the COP410L resonator controlled configuration, CKO is used as an output to the resonator network. In the other two configurations (external clock and RC controlled), CKO may be mask-programmed as a RAM power supply pin ($V_R$), allowing its connection to a standby battery backup power supply to maintain the integrity of RAM data with minimum power drain when the main supply is inoperative or shut down to conserve power.

### COP410L/COP411L I/O Options

COP410L/COP411L *inputs* and *outputs* have the same optional configurations as the COP420L/COP421L; see Section 2.7

The input and output configurations share common enhancement-mode and depletion-mode devices. For detailed electrical characteristics on these devices, refer to the COP410L and COP421L data sheets.

The SO and SK outputs can be configured as shown in Figure 2.6, a, b, or c. The D and G outputs can be configured as shown in a or b. Note that when inputting data to the G ports, the G outputs should be set to "1." The L outputs can be configured as shown in d, e, f, or g.

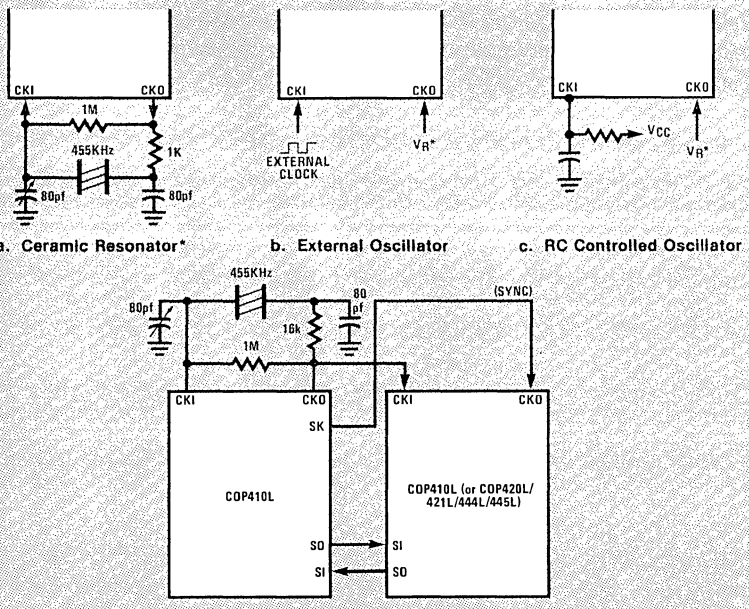
An important point to remember is that *all* of the L driver options are TRI-STATE® -able. Therefore, the L drivers have TRI-STATE-able Standard and Open-Drain output options as well as the TRI-STATE LED Direct Drive and Push-Pull output options. Since the device to $V_{CC}$ in the Standard output configuration is a depletion-mode device, it will source up to 0.125 mA when this output is "turned off" in the TRI-STATE mode, which is insufficient to guarantee a logic "1" input level.

### Bonding Option

The COP411L is a bonding option of the COP410L: if the COP410L is bonded as a 20-pin device (without CKO, $D_2$, $D_3$, and $G_3$), it becomes the COP411L. Use of output options associated with these deleted pins are, of course, precluded. All other COP410L options are available.

## 2.16 COP410L/COP411L Option List

The COP410L/COP411L mask-programmable options are assigned numbers which correspond with the *COP410L* pins.




RC Controlled Oscillator

| R (kΩ) | C (pF) | Instruction Cycle Time (μs) |
|---|---|---|
| 51 | 100 | 19 ± 15% |
| 82 | 56 | 19 ± 15% |

Figure 2.12 COP410L/COP411L Oscillator Configurations

When specifying a COP411L device, Option 2 must be set to a value of "3," and Options 20, 21, and 22 must be set to a value of "0," since the COP411L does not include these pins.

The following is a list of COP410L/COP411L options:

Option 1 = 0: Ground Pin — no options available

Option 2: CKO Output
- = 0: clock generator output to crystal/resonator
- = 1: pin is RAM power supply ($V_R$) input
- = 2: multi-COP SYNC input (CKI ÷ 8)
- = 3: no connection (COP411L)

Option 3: CKI Input
- = 0: oscillator input divided by 8 (500 kHz maximum)
- = 1: single-pin RC controlled oscillator divided by 4

Option 4: $\overline{\text{RESET}}$ Input
- = 0: load device to $V_{CC}$
- = 1: hi-Z input

Option 5: $L_7$ Driver
- = 0: standard output (Figure 2.6d)
- = 1: open-drain output (Figure 2.6e)
- = 2: high current LED direct segment drive output (Figure 2.6f)
- = 3: high current TRI-STATE® push-pull output (Figure 2.6g)
- = 4: low current LED direct drive output (Figure 2.6g)
- = 5: low current TRI-STATE® push-pull output (Figure 2.9f)

Option 6: $L_6$ Driver
same as Option 5

Option 7: $L_5$ Driver
same as Option 5

Option 8: $L_4$ Driver
same as Option 5

Option 9: $V_{CC}$ Pin
= 0: 4.5-6.3V operation
= 1: 4.5-9.5V operation (extra cost option)

Option 10: $L_3$ Driver
same as Option 5

Option 11: $L_2$ Driver
same as Option 5

Option 12: $L_1$ Driver
same as Option 5

Option 13: $L_0$ Driver
same as Option 5

Option 14: SI Input
same as Option 4

Option 15: SO Driver
= 0: standard output (Figure 2.6a)
= 1: open-drain output (Figure 2.6b)
= 2: push-pull output (Figure 2.6c)

Option 16: SK Driver
same as Option 15

Option 17: $G_0$ I/O Port
= 0: standard output (Figure 2.6a)
= 1: open-drain output (Figure 2.6b)

Option 18: $G_1$ I/O Port
same as Option 17

Option 19: $G_2$ I/O Port
same as Option 17

Option 20: $G_3$ I/O Port
same as Option 17

Option 21: $D_3$ Output
= 0: very high current standard output
= 1: very high current open-drain output
= 2: high current standard output
= 3: high current open-drain output
= 4: standard LSTTL output (fanout = 1)
= 5: open-drain LSTTL output (fanout = 1)

Option 22: $D_2$ Output
same as Option 21

Option 23: $D_1$ Output
same as Option 21

Option 24: $D_0$ Output
same as Option 21

Option 25: L Input Levels
= 0: standard TTL input levels ("0" = 0.8V, "1" = 2.0V)
= 1: higher voltage input levels ("0" = 1.2V, "1" = 3.6V)

Option 26: G Input Levels
same as Option 25

Option 27: SI Input Levels
same as Option 25

Option 28: COP Bonding
= 0: COP410L
= 1: COP411L

# 3 COP400 Instruction Sets

This chapter provides information on the instruction sets of the COP400 microcontrollers. As with the architecture of the different devices in the COP400 family, the instruction sets of the various devices allow the user to choose among several devices to provide only as much software capability as is needed for a particular application. Specifically, the instruction sets of the various devices are, generally, subsets of the most inclusive instruction set of the COP440. This chapter will discuss the COP420-series (includes COP421, COP421L, COP421C), COP444L, COP410L, and COP411L, respectively. Users of the COP440 should refer to the COP440 data sheet (when the device becomes available) for information on the additional instructions associated with the COP440 instruction set.

This chapter primarily provides information on the machine operations associated with the instruction set of COP400 devices. However, where appropriate, short examples indicating typical usage of particular instructions are provided. For a detailed treatment on using COP400 instructions to write COP400 assembly language programs, see Chapter 4 of this manual.

## 3.1 COP420-Series/COP444L Instruction Set

Table 3.1 provides the mnemonic, operand, machine code, data flow, skip conditions and description associated with each instruction in the COP420-series/COP444L instruction set. As indicated, an asterisk in the description column signifies a double-byte instruction. Also, notes are provided following this table which describe or refer to additional information relevant to particular instructions. As indicated by Note 3, the INI and INIL instructions are not included in the COP421 instruction set, due to its lack of IN inputs and the $IL_3$ and $IL_0$ latches associated with two of the IN inputs ($IN_3$ and $IN_0$, respectively).

Note that the COP420-series/COP444L set, as with all COP400 instruction sets, is divided into the following categories: Arithmetic Operations, Input/Output Instructions, Transfer of Control Instructions, Memory Reference Instructions, Register Reference Instructions, and Test Instructions.

Table 3.1 COP420 Series/COP444L Instruction Set

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **ARITHMETIC INSTRUCTIONS** | | | | | | |
| ASC | | 30 | 0 0 1 1 \| 0 0 0 0 | A + C + RAM(B) → A<br>Carry → C | Carry | Add with Carry, Skip on Carry |
| ADD | | 31 | 0 0 1 1 \| 0 0 0 1 | A + RAM(B) → A | None | Add RAM to A |
| ADT | | 4A | 0 1 0 0 \| 1 0 1 0 | A + $10_{10}$ → A | None | Add Ten to A |
| AISC | y | 5– | 0 1 0 1 \| y | A + y → A | Carry | Add Immediate, Skip on Carry (y ≠ 0) |
| CASC | | 10 | 0 0 0 1 \| 0 0 0 0 | $\overline{A}$ + RAM(B) + C → A<br>Carry → C | Carry | Complement and Add with Carry, Skip on Carry |
| CLRA | | 00 | 0 0 0 0 \| 0 0 0 0 | 0 → A | None | Clear A |
| COMP | | 40 | 0 1 0 0 \| 0 0 0 0 | $\overline{A}$ → A | None | Ones complement of A to A |
| NOP | | 44 | 0 1 0 0 \| 0 1 0 0 | None | None | No Operation |
| RC | | 32 | 0 0 1 1 \| 0 0 1 0 | "0" → C | None | Reset C |
| SC | | 22 | 0 0 1 0 \| 0 0 1 0 | "1" → C | None | Set C |
| XOR | | 02 | 0 0 0 0 \| 0 0 1 0 | A ⊕ RAM(B) → A | None | Exclusive-OR RAM with A |
| **TRANSFER OF CONTROL INSTRUCTIONS** | | | | | | |
| JID | | FF | 1 1 1 1 \| 1 1 1 1 | ROM ($PC_{9:8}$,A,M) → $PC_{7:0}$ | None | Jump Indirect (Note 3) |
| JMP | a | 6–<br>– – | 0 1 1 0 \| 0 0 \| $a_{9:8}$<br>$a_{7:0}$ | a → PC | None | • Jump |
| JP | a | – – | 1 \| $a_{6:0}$<br>(pages 2,3 only)<br>or | a → $PC_{6:0}$ | None | Jump within Page (Note 4) |
| | | – – | 1 1 \| $a_{5:0}$<br>(all other pages) | a → $PC_{5:0}$ | | |
| JSRP | a | – – | 1 0 \| $a_{5:0}$ | PC+1 → SA → SB → SC<br>0010 → $PC_{9:6}$<br>a → $PC_{5:0}$ | None | Jump to Subroutine Page (Note 5) |
| JSR | a | 6–<br>– – | 0 1 1 0 \| 1 0 \| $a_{9:8}$<br>$a_{7:0}$ | PC+1 → SA → SB → SC<br>a → PC | None | • Jump to Subroutine |
| RET | | 48 | 0 1 0 0 \| 1 0 0 0 | SC → SB → SA → PC | None | Return from Subroutine |
| RETSK | | 49 | 0 1 0 0 \| 1 0 0 1 | SC → SB → SA → PC | Always Skip on Return | Return from Subroutine then Skip |

Table 3.1 COP420 Series/COP444L Instruction Set (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **MEMORY REFERENCE INSTRUCTIONS** | | | | | | |
| CAMQ | | 33 | 0 0 1 1 \| 0 0 1 1 | $A \rightarrow Q_{7:4}$ | None | • Copy A, RAM to Q |
| | | 3C | 0 0 1 1 \| 1 1 0 0 | $RAM(B) \rightarrow Q_{3:0}$ | | |
| CQMA | | 33 | 0 0 1 1 \| 0 0 1 1 | $Q_{7:4} \rightarrow RAM(B)$ | None | • Copy Q to RAM, A |
| | | 2C | 0 0 1 0 \| 1 1 0 0 | $Q_{3:0} \rightarrow A$ | | |
| LD | r | -5 | 0 0 \| r \| 0 1 0 1 | $RAM(B) \rightarrow A$<br>$Br \oplus r \rightarrow Br$ | None | Load RAM into A, Exclusive-OR Br with r |
| LDD | r,d | 23 | 0 0 \| 1 0 \| 0 0 1 1 | $RAM(r,d) \rightarrow A$ | None | • Load A with RAM pointed to directly by r,d |
| | | -- | 0 0 \| r \| d | | | |
| LQID | | BF | 1 0 1 1 \| 1 1 1 1 | $ROM(PC_{9:8},A,M) \rightarrow Q$<br>$SB \rightarrow SC$ | None | Load Q Indirect (Note 3) |
| RMB | 0 | 4C | 0 1 0 0 \| 1 1 0 0 | $0 \rightarrow RAM(B)_0$ | None | Reset RAM Bit |
| | 1 | 45 | 0 1 0 0 \| 0 1 0 1 | $0 \rightarrow RAM(B)_1$ | | |
| | 2 | 42 | 0 1 0 0 \| 0 0 1 0 | $0 \rightarrow RAM(B)_2$ | | |
| | 3 | 43 | 0 1 0 0 \| 0 0 1 1 | $0 \rightarrow RAM(B)_3$ | | |
| SMB | 0 | 4D | 0 1 0 0 \| 1 1 0 1 | $1 \rightarrow RAM(B)_0$ | None | Set RAM Bit |
| | 1 | 47 | 0 1 0 0 \| 0 1 1 1 | $1 \rightarrow RAM(B)_1$ | | |
| | 2 | 46 | 0 1 0 0 \| 0 1 1 0 | $1 \rightarrow RAM(B)_2$ | | |
| | 3 | 4B | 0 1 0 0 \| 1 0 1 1 | $1 \rightarrow RAM(B)_3$ | | |
| STII | y | 7- | 0 1 1 1 \| y | $y \rightarrow RAM(B)$<br>$Bd + 1 \rightarrow Bd$ | None | Store Memory Immediate and Increment Bd |
| X | r | -6 | 0 0 \| r \| 0 1 1 0 | $RAM(B) \longleftrightarrow A$<br>$Br \oplus r \rightarrow Br$ | None | Exchange RAM with A, Exclusive-OR Br with r |
| XAD | r,d | 23 | 0 0 1 0 \| 0 0 1 1 | $RAM(r,d) \longleftrightarrow A$ | None | • Exchange A with RAM pointed to directly by r,d |
| | | -- | 1 0 \| r \| d | | | |
| XDS | r | -7 | 0 0 \| r \| 0 1 1 1 | $RAM(B) \longleftrightarrow A$<br>$Bd - 1 \rightarrow Bd$<br>$Br \oplus r \rightarrow Br$ | Bd decrements past 0 | Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r |
| XIS | r | -4 | 0 0 \| r \| 0 1 0 0 | $RAM(B) \longleftrightarrow A$<br>$Bd + 1 \rightarrow Bd$<br>$Br \oplus r \rightarrow Br$ | Bd increments past 15 | Exchange RAM with A and Increment Bd, Exclusive-OR Br with r |
| **REGISTER REFERENCE INSTRUCTIONS** | | | | | | |
| CAB | | 50 | 0 1 0 1 \| 0 0 0 0 | $A \rightarrow Bd$ | None | Copy A to Bd |
| CBA | | 4E | 0 1 0 0 \| 1 1 1 0 | $Bd \rightarrow A$ | None | Copy Bd to A |
| LBI | r,d | -- | 0 0 \| r \| (d - 1)<br>(d = 0, 9:15) | $r,d \rightarrow B$ | Skip until not a LBI | Load B Immediate with r,d (Note 6) |
| | | | or | | | |
| | | 33 | 0 0 1 1 \| 0 0 1 1 | | | |
| | | -- | 1 0 \| r \| d<br>(any d) | | | |
| LEI | y | 33 | 0 0 1 1 \| 0 0 1 1 | $y \rightarrow EN$ | None | • Load EN Immediate (Note 7) |
| | | 6- | 0 1 1 0 \| y | | | |
| XABR | | 12 | 0 0 0 1 \| 0 0 1 0 | $A \longleftrightarrow Br\ (0,0 \rightarrow A_3,A_2)$ | None | Exchange A with Br |

Table 3.1 COP420 Series/COP444L Instruction Set (continued)

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **TEST INSTRUCTIONS** | | | | | | |
| SKC | | 20 | 0 0 1 0 \| 0 0 0 0 | | C = "1" | Skip if C is True |
| SKE | | 21 | 0 0 1 0 \| 0 0 0 1 | | A = RAM(B) | Skip if A Equals RAM |
| SKGZ | | 33 | 0 0 1 1 \| 0 0 1 1 | | $G_{3:0} = 0$ | * Skip if G is Zero (all 4 bits) |
| | | 21 | 0 0 1 0 \| 0 0 0 1 | | | |
| SKGBZ | | 33 | 0 0 1 1 \| 0 0 1 1 | 1st byte | | * Skip if G Bit is Zero |
| | 0 | 01 | 0 0 0 0 \| 0 0 0 1 | 2nd byte | $G_0 = 0$ | |
| | 1 | 11 | 0 0 0 1 \| 0 0 0 1 | | $G_1 = 0$ | |
| | 2 | 03 | 0 0 0 0 \| 0 0 1 1 | | $G_2 = 0$ | |
| | 3 | 13 | 0 0 0 1 \| 0 0 1 1 | | $G_3 = 0$ | |
| SKMBZ | 0 | 01 | 0 0 0 0 \| 0 0 0 1 | | $RAM(B)_0 = 0$ | Skip if RAM Bit is Zero |
| | 1 | 11 | 0 0 0 1 \| 0 0 0 1 | | $RAM(B)_1 = 0$ | |
| | 2 | 03 | 0 0 0 0 \| 0 0 1 1 | | $RAM(B)_2 = 0$ | |
| | 3 | 13 | 0 0 0 1 \| 0 0 1 1 | | $RAM(B)_3 = 0$ | |
| SKT | | 41 | 0 1 0 0 \| 0 0 0 1 | | A time-base counter carry has occurred since last test | Skip on Timer (Note 3) |
| **INPUT/OUTPUT INSTRUCTIONS** | | | | | | |
| ING | | 33 | 0 0 1 1 \| 0 0 1 1 | G → A | None | * Input G Ports to A |
| | | 2A | 0 0 1 0 \| 1 0 1 0 | | | |
| ININ | | 33 | 0 0 1 1 \| 0 0 1 1 | IN → A | None | * Input IN Inputs to A (Note 2) |
| | | 28 | 0 0 1 0 \| 1 0 0 0 | | | |
| INIL | | 33 | 0 0 1 1 \| 0 0 1 1 | $IL_3$,"1","0",$IL_0$ → A | None | * Input IL Latches to A (Note 3) |
| | | 29 | 0 0 1 0 \| 1 0 0 1 | | | |
| INL | | 33 | 0 0 1 1 \| 0 0 1 1 | $L_{7:4}$ → RAM(B) | None | * Input L Ports to RAM, A |
| | | 2E | 0 0 1 0 \| 1 1 1 0 | $L_{3:0}$ → A | | |
| OBD | | 33 | 0 0 1 1 \| 0 0 1 1 | Bd → D | None | * Output Bd to D Outputs |
| | | 3E | 0 0 1 1 \| 1 1 1 0 | | | |
| OGI | y | 33 | 0 0 1 1 \| 0 0 1 1 | y → G | None | * Output to G Ports Immediate |
| | | 5– | 0 1 0 1 \| y | | | |
| OMG | | 33 | 0 0 1 1 \| 0 0 1 1 | RAM(B) → G | None | * Output RAM to G Ports |
| | | 3A | 0 0 1 1 \| 1 0 1 0 | | | |
| XAS | | 4F | 0 1 0 0 \| 1 1 1 1 | A ⟷ SIO, C → SK | None | Exchange A with SIO (Note 3) |

**Note 1:** All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant (low-order, right-most bit). For example, $A_3$ indicates the most significant (left-most) bit of the 4-bit A register.

**Note 2:** The ININ instruction is not available on the 24-pin COP421 since this device does not contain the IN inputs.

**Note 3:** For additional information on the operation of the XAS, JID, LQID, INIL, and SKT instructions, see Section 3.2.

**Note 4:** The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

**Note 5:** A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

**Note 6:** LBI is a single-byte instruction if d = 0, 9, 10, 11, 12, 13, 14, or 15. The machine code for the lower 4 bits equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 ($1001_2$), the lower 4 bits of the LBI instruction equal 8 ($1000_2$). To load 0, the lower 4 bits of the LBI instruction should equal 15 ($1111_2$).

**Note 7:** Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register.)

Table 3.2 provides a list of internal architecture, instruction operand and operational symbols used in the COP420-series/COP444L Instruction Set Table. Table 3.5 shows an alphabetical mnemonic index of COP420-series/COP444L instructions, indicating the hexadecimal opcode and description associated with each instruction. Table 3.6 is a list of COP420-series/COP444L instructions arranged in order of their hexadecimal opcodes.

The following text gives a description of each COP420-series/COP444L instruction, explaining the machine operations performed by each instruction and, where appropriate, providing short examples illustrating typical usage of particular instructions.

**Table 3.2 COP420-Series/COP444L Instruction Set Table Symbols**

| Symbol | Definition |
|---|---|
| INTERNAL ARCHITECTURE SYMBOLS | |
| A | 4-bit Accumulator |
| B | 6-bit RAM Address Register |
| Br | Upper 2 bits of B (register address) |
| Bd | Lower 4 bits of B (digit address) |
| C | 1-bit Carry Register |
| D | 4-bit Data Output Port |
| EN | 4-bit Enable Register |
| G | 4-bit Register to latch data for G I/O Port |
| IL | Two 1-bit Latches associated with the $IN_3$ or $IN_0$ inputs |
| IN | 4-bit Input Port |
| L | 8-bit TRI-STATE I/O Port |
| M | 4-bit contents of RAM Memory pointed to by B Register |
| PC | 10-bit ROM Address Register (program counter) |
| Q | 8-bit Register to latch data for L I/O Port |
| SA | 10-bit Subroutine Save Register A |
| SB | 10-bit Subroutine Save Register B |
| SC | 10-bit Subroutine Save Register C |
| SIO | 4-bit Shift Register and Counter |
| SK | Logic-Controlled Clock Output |
| **Symbol** | **Definition** |
| INSTRUCTION OPERAND SYMBOLS | |
| d | 4-bit Operand Field, 0-15 binary (RAM Digit Select) |
| r | 2-bit Operand Field, 0-3 binary (RAM Register Select) |
| a | 10-bit Operand Field, 0-1023 binary (ROM Address) |
| y | 4-bit Operand Field, 0-15 binary (Immediate Data) |
| RAM(s) | Contents of RAM location addressed by s |
| ROM(t) | Contents of ROM location addressed by t |
| OPERATIONAL SYMBOLS | |
| + | Plus |
| − | Minus |
| → | Replaces |
| ⟷ | Is exchanged with |
| = | Is equal to |
| $\overline{A}$ | The ones complement of A |
| ⊕ | Exclusive-OR |
| : | Range of values |

## 3.2 COP420-Series/COP444L Instruction Set Description

### Arithmetic Instructions

**ASC** (Add with carry, Skip on Carry) performs a binary addition of A, C (Carry bit), and M, placing the result in A and C. If a carry occurs, the next program instruction is skipped.

**ADD** (ADD) performs binary addition. The 4-bit addends are A and M. The 4-bit sum is placed in A. ADD does not affect the carry or skip.

**ADT** (ADd Ten to A) adds ten ($1010_2$) to A and, like ADD, does not affect the carry or skip. It is intended to facilitate Binary Coded Decimal (BCD) arithmetic. For example, the following sequence of instructions will perform a single-digit BCD add of the contents of A and M [the carry is assumed set when entering this routine if addition of the previous least significant digits produced an overflow (A > 9)]:

```
AISC 6
ASC
ADT
```

The AISC 6 instruction adds a BCD correction factor (i.e., 6) to the digit in the accumulator. (See AISC instruction.) Since the accumulator contains a BCD digit (⩽ 9) no carry will occur and the next instruction, ASC, will always be executed. The ASC instruction adds the carry and memory digit to A, as explained above. If the result does *not* produce a carry, signifying that the previous AISC 6 (correction factor) instruction was unnecessary, the ADT instruction is executed, readjusting the accumulator to the proper BCD result. (Remember: ADT neither affects the carry nor skips.)

If the ASC result does produce a carry, C is set for propagation to the addition of the next most significant digits and, since no readjustment of the result is necessary, the ADT instruction is skipped.

**AISC** (Add Immediate, Skip on Carry) adds the instruction operand constant "y" (1-15) to A, skipping the next instruction if a carry out occurs (C is *not* changed). This instruction finds frequent use in BCD add and subtract routines (see ADT and CASC descriptions) as well as in testing the value of A. (If A is greater than 12, for instance, an AISC 5 will skip the next instruction.)

**CASC** (Complement and Add, Skip on Carry) performs a binary subtraction of A from M by summing the complement of A ($\overline{A}$) with C and M, placing the result in A and C. If no carry out occurs, indicating a borrow, C is reset and the next instruction is executed. If a carry occurs, indicating no borrow, C is set and the next instruction is skipped.

A single BCD digit binary subtraction of A from M may be performed as follows. (The carry bit is assumed set upon initial entry to the routine.)

```
CASC
ADT
```

The CASC instruction will set C and skip the ADT instruction if the subtraction does not result in a borrow (A > M). If a borrow occurs, the ADT instruction is executed, readjusting the result to the proper BCD value, leaving C reset for propagation of the borrow in the subtraction of the next most significant BCD digits. CASC is functionally equivalent to a COMP instruction followed by an ASC.

**CLRA** (CLeaR A) clears the accumulator by placing zeros in each of the 4 bits of A.

This instruction is often required prior to loading A equal to a desired value with an AISC instruction if the previous contents of A are unknown. For instance, to load A = 11, the following sequence may be used:

```
CLRA
AISC 11
```

The skip features associated with AISC need not be considered in this example. (A carry will never occur.)

**COMP** (COMPlement A) changes the state of each of 4 bits of A with ones becoming zeros and zeros becoming ones. It has the effect of, and may be used to perform, a binary (one's complement) subtraction of A from 15 ($1111_2$), e.g., complementing A = 6 ($0110_2$) will yield 9 ($1001_2$).

**NOP** (No OPeration) does not perform any operation. It is useful, however, for simple single instruction time delays or to defeat the skip conditions associated with particular instructions.

**SC** (Set Carry) and **RC** (Reset Carry) set C and reset C, respectively. SC and RC are most often employed to initialize C prior to entering arithmetic routines. They also allow C to be used as a general-purpose (testable) flag, as long as subsequent instructions do not inadvertently affect the C register.

**XOR** (eXclusive-OR A with M) performs a logical EXCLUSIVE-OR operation of each bit of A with each corresponding bit of M, *placing the result in A*. This operation can be used to change the state of any bit in M, if the corresponding (equally weighted) bit of A is set. This follows from the EXCLUSIVE-OR truth table where a X + "1" = $\overline{X}$, and a X + "0" = X, assuming the "X" bits to be one of the 4 bits in M, and the "1" and "0" to be equally weighted bits in A. This instruction, therefore, allows the selective complementing or toggling of one or more bits of M. Example: to change the state of bit 2 of M, set A = 0100, perform an XOR, then exchange A into M with an X instruction.

**Input/Output Instructions**

**ING** (INput G ports to A) transfers the 4-bit contents of the IN ports ($IN_3$-$IN_0$) to A.

**ININ** (INput IN inputs to A) transfers the 4-bit contents of the IN ports ($IN_3$-$IN_0$) to A.

**INIL** (INput IL latches to A) is a special purpose instruction which inputs the two latches $IL_3$ and $IL_0$ (see Figure 3.1 below) and, if the appropriate option is selected, a general-purpose input, CKO, to the accumulator — the unused bit/bits of A are reset. Specifically, INIL places $IL_3 \rightarrow A_3$, CKO $\rightarrow A_2$, "0" $\rightarrow A_1$, $IL_0 \rightarrow A_0$. $IL_3$ and $IL_0$ are the outputs of latches associated with the $IN_3$ and $IN_0$ inputs. (The *general purpose* inputs, $IN_3$-$IN_0$, are input to A upon the execution of an ININ instruction. (See ININ Instruction.) The $IL_3$ and $IL_0$ latches are set if a low-going pulse ("1" to "0") has occurred on the $IN_3$ and $IN_0$ inputs, respectively, since the last INIL instruction, provided the input pulse stays low for at least two instruction times. Execution of an INIL inputs $IL_3$ and $IL_0$ into $A_3$ and $A_0$ respectively, and resets these latches to allow them to respond to subsequent low-going pulses on the $IN_3$ and $IN_0$ lines.

If CKO is mask-programmed as a general-purpose input, an INIL will input the state of CKO into $A_2$. If CKO has not been so programmed, a "1" will be placed in $A_2$. A "0" is always placed in $A_1$ upon the execution of an INIL.

INIL is useful in recognizing and capturing pulses of short duration or which can't be read conveniently by an ININ instruction.

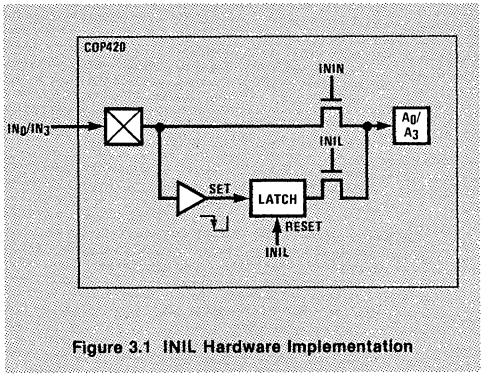



**Figure 3.1 INIL Hardware Implementation**

**INL** (INput L ports to M, A) transfers the 8-bit contents of the bidirectional TRI-STATE® I/O ports to M, A. $L_7$-$L_4$ are placed in $M_3$-$M_0$ (the memory digit pointed to by the B register); $L_3$-$L_0$ are placed in $A_3$-$A_0$.

**OBD** (Output Bd to D outputs) transfers the 4-bit contents of Bd (lower 4 bits of the B register) to the D output ports ($D_3$-$D_0$). Since, in many applications, the D outputs are connected to a digit decoder, the direct output of Bd allows for a standard interconnect to the binary inputs of the decoder/driver device.

**OGI** (Output to G ports Immediate) transfers the four bits specified in the "y" operand field of this instruction (0-15, binary) to $G_3$-$G_0$.

**OMG** (Output M to G ports) transfers the 4-bit contents of M ($M_3$-$M_0$) to $G_3$-$G_0$.

**XAS** (eXchange A with SIO) exchanges the 4-bit contents of A ($A_3$-$A_0$) with the 4-bit contents of the SIO register ($SIO_3$-$SIO_0$). SIO will contain serial-in/serial-out shift register or binary counter data, depending on the value of the EN register. An XAS instruction will also affect the SK output, providing a logic controlled clock if SIO is selected as a shift register or C → SK if SIO is selected as a binary counter.

For further information on the EN register and its relationship to the XAS instruction, see LEI Instruction, below. If SIO is selected as a shift register, an XAS instruction must be performed once every 4 instruction cycle times to effect a continuous serial-in or serial-out data stream.

### Transfer of Control Instructions

**JID** (Jump InDirect) is an indirect addressing instruction, transferring program control to a new ROM location addrssed by the *contents* of the ROM location pointed to by A and M. Specifically, it loads the lower 8 bits of the ROM address register P with the *contents* of ROM pointed to by the 10-bit word $P_9P_8A_3A_2A_1A_0M_3M_2M_1M_0$. The contents of the selected ROM location ($I_7$-$I_0$) are, therefore, loaded into $P_7$-$P_0$, changing the lower 8 bits of P to transfer program control to the new ROM location.

$P_9$ and $P_8$ remain unchanged throughout the execution of the JID instruction. JID, therefore, may only jump to a ROM location within the current 4-page ROM "block" (pages 0-3, 4-7, 8-11 or 12-15). For further information regarding the "paging" restrictions associated with the JID instruction, see Section 4.1.

JID can be useful in keyboard-decode routines when the values associated with the row and column of a particular key closure are placed in A and M for a jump indirect to the contents of ROM which point to the starting address of the appropriate routine associated with that particular key closure. For an example of use of the JID instruction to access a keyboard-decode ROM pointer table, see Display/Keyboard Program, Section 5.3, #16.

**JMP** (JuMP) transfers program control to any word in the ROM as specified by the "a" field of this instruction. The 10-bit "a" field is placed in $P_9$-$P_0$. JMP is used to transfer program control from one page to *another* page (if in page 2 or 3, the more efficient single-byte JP instruction may be used) or to transfer control to the *last* word of the current page — an invalid transfer for the JP instruction.

**JP** (Jump within Page) transfers program control to the ROM address specified in the operand field of this instruction. The machine code and operand field of this instruction have two formats. If program execution is currently within page 2 or 3 (subroutine pages) a 7-bit "a" field is specified, transferring program control to a word within either of the two subroutine pages. Otherwise, only a 6-bit "a" field is specified, transferring program control to a particular word within the *current* 64-word ROM page.

Specifically, this instruction places $a_6$-$a_0$ in $P_6$-$P_0$ if the program is currently in subroutine page 2 or 3. If in any other page, it places $a_5$-$a_0$ in $P_5$-$P_0$.

The restrictions associated with the JP instruction, therefore, are that a 7-bit "a" field may be used only when in pages 2 or 3. Otherwise, a JP may be used only to jump within the current page by specifying a 6-bit "a" field in the operand of this instruction. An additional restriction associated with the JP instruction, in either of the above two formats, is that a JP to the last word of any page is invalid, i.e., "a" may not equal all 1s. A transfer of program control to last word on a page may be effected by using a JMP instruction. (See JMP Instruction, above.)

**JSRP** (Jump to SubRoutine Page) is used to transfer program control from a page other than 2 or 3 to a word within page 2. It accomplishes this by placing a 2 ($0010_2$) in $P_9$-$P_6$, and the word address specified in the 6-bit "a" field of the instruction into $P_5$-$P_0$. Designed to transfer control to subroutines, it *pushes the stack* to save the subroutine return address — the address of the next program instruction is saved in SA and the other subroutine-save registers are likewise pushed (P + 1 → SA → SB → SC). Any previous contents of SC are lost, since SC is the last of the three subroutine-save registers. Subroutine nesting, therefore, is permitted to three levels. JSRP is used in conjunction with the RET or RETSK instructions which "pop" the stack at the end of subroutine to return program control to the main program. As with the JP instruction, JSRP may not transfer program control to the last word of page 2: "a" may not equal all "1s." A JSR may be used to jump to the last word of a subroutine beginning at the last word of page 2. (See JSR, below.) As mentioned above, a further restriction is that a

JSRP may not be used when in subroutine pages 2 or 3. To transfer program control to a subroutine in page 2 when in pages 2 or 3, the double-byte JSR should be used, or, if it is not necessary to push the stack, a JP instruction may be used.

**JSR** (Jump to SubRoutine) transfers program control to a subroutine located at a particular word address in *any* ROM page. It modifies the entire P register with the value of the "a" operand of this instruction, as follows: $a_9$-$a_0 \rightarrow P_9$-$P_0$. As with the JSRP instruction, JSR pushes the stack ($P+1 \rightarrow SA \rightarrow SB \rightarrow SC$), saving the next program instruction for a return from the subroutine to the main program via a RET or RETSK instruction. JSR may be used to overcome the restrictions associated with the JSRP instruction: to jump to a subroutine and push the stack when in pages 2 or 3, or to jump to a subroutine located at the last word of page 2.

**RET** (RETurn from subroutine) is used to return program control to the main program following a JSR or JSRP instruction. RET "pops" the stack ($SC \rightarrow SB \rightarrow SA \rightarrow P$): the next main program instruction address ($P+1$) saved in SA is loaded into P, the contents of SB are loaded into SA and the contents of SC are loaded into SB. (The contents of SC are also retained in SC.) Program control, therefore, is returned to the instruction immediately following the previous subroutine call.

**RETSK** (RETurn from subroutine then SKip), as with the RET instruction above, pops the stack ($SC \rightarrow SB \rightarrow SA \rightarrow P$), restoring program control to the main program following a subroutine call. It, however, *always* skips the first instruction encountered when it returns to the main program. This instruction, therefore, provides the programmer with an alternate return from subroutines, either via a RET or RETSK, based upon tests made within the subroutine itself.

**CAMQ** (Copy A, M to Q) transfers the 8-bit contents of A and M to the Q latches. $A_3$-$A_0$ are output to $Q_7$-$Q_4$; $M_3$-$M_0$ are output to $Q_3$-$Q_0$. Note that CAMQ is the inverse of CQMA (see CQMA Instruction, below) with respect to the 4 bits of Q with which A and M communicate. Therefore, the input and processing of Q must often be followed by an X (Exchange M with A) instruction before final output to Q in order to maintain the proper bit-weights of the Q data. For example, the following instructions read Q to M, A, set $Q_7$ and perform the necessary exchange before execution of the CAMQ instruction:

```
CQMA      ; Q TO M, A
SMB  3    ; SET Q7 BIT LOCATED IN M3
X         ; EXCHANGE M WITH A
CAMQ      ; A, M TO Q
```

**CQMA** (Copy Q to M, A) transfers the 8-bit contents of the Q latches to M and A. $Q_7$-$Q_4$ are placed in $M_3$-$M_0$; $Q_3$-$Q_0$ are placed in $A_3$-$A_0$. CQMA can be employed after an LQID (Load Q InDirect) instruction to input or alter the value of lookup data. CQMA is also an essential instruction when the COP420 is employed as a MICROBUS™ peripheral component. In such applications, $IN_3$ is used by the control microprocessor to write bus data from the L ports to the Q latches. (See Section 2.4, MICROBUS™ option.) A CQMA will then input this data to M, A as explained above for processing by the COP420 program.

### Memory Reference Instructions

**LD** (LoaD M into A) loads M (the 4-bit contents of RAM pointed to by the B register: $M_3$-$M_0$) into $A_3$-$A_0$. After M is loaded into A, the 2-bit "r" operand field is EXCLUSIVE-ORed with the contents of Br (upper 2 bits of B — RAM register select) to point to a new RAM register for successive memory reference operations. Since the properties of the EXCLUSIVE-OR logic operation are such that a $1 \oplus X$ equals the complement of X, use of the "r" field allows the programmer to switch between any one of the 4 RAM registers by complementing the appropriate bit/bits of the current contents of the Br register. Of course, if "r" = 0, the contents of Br will remain unchanged after the execution of a LD instruction.

For example, if the assembly language instruction LD 3 ("r" = $11_2$) is executed with Br = 2 ($10_2$) and Bd = 12 ($1100_2$), the contents of RAM register 2, digit 12 will be loaded to A and Br will be changed to ($11_2 + 10_2 = 01_2$), with B pointing to RAM register 1, digit 12. For assembly language programming use of an EXCLUSIVE-OR "r" operand field with memory reference instructions which use this field is optional — if not specified, an "0" operand is assumed. For further information on allocating RAM map locations for optimum use of the EXCLUSIVE-OR feature associated with this and other memory reference instructions and for sample routines utilizing this feature, refer to Sections 4.2 and 4.4.

**SMB** (Set Memory Bit) and **RMB** (Reset Memory Bit) set and reset, respectively, a bit in M as specified by the operand field of these instructions. (Remember: M is the 4-bit RAM digit pointed to by the B register.) The operand field is specified according to the bit number (0-3, left-most to right-most bit) of the particular bit to be set or reset, e.g., an SMB 3 would set the most significant bit of M. These instructions are useful in operating upon program status flags located in RAM.

**STII** (Store Memory Immediate and Increment Bd) loads the 4-bit contents specified by the "y"

operand field of the instruction into the RAM memory digit pointed to by the B register, $M_3$-$M_0$. It is important to note that the value of Bd (RAM digit-select) is *incremented* (as with the XIS instruction) after the "y" data is stored in M.

**LDD** (LoaD A with M Directly) loads the 4-bit contents of the RAM memory location pointed to directly by the "r" and "d" operand fields (register and digit select, respectively) of the instruction, $M_3$-$M_0$, into $A_3$-$A_0$. Note that this instruction and the XAD instruction differ from other memory reference instructions in that the operand of the instruction, not the B register, is used to point to the appropriate RAM digit location to be accessed — the B register is unaffected by these instructions. This instruction is useful in accessing RAM counters, status and flag digits, etc., within routines or loops without destroying the previous value of B, allowing the latter to be used for sequential memory access operations and for other reiterative purposes.

**LQID** (Load Q InDirect) is, in effect, a ROM data "lookup" instruction. It transfers the 8-bit contents of ROM, $I_7$-$I_0$, pointed to by the 10-bit word $P_9P_8$AM to $Q_7$-$Q_0$, respectively. It does this by pushing the stack (P + 1 → SA → SB → SC) and replacing the least significant 8 bits of P as follows: $A_3$-$A_0$ → $P_7$-$P_4$; $M_3$-$M_0$ → $P_3$-$P_0$, leaving the two most significant bits of P unchanged. The ROM data pointed to by the new P address is fetched and loaded into the Q latches, $Q_7$-$Q_0$. Next, the stack is popped (SC → SB → SA → P), restoring the previous pushed value of P (P + 1) to continue sequential program execution. Since LQID pushes SB → SC, the previous contents of SC are lost. Also, when LQID pops the stack, the previously pushed contents of SB are left in SC as well as loaded back into SB. The net result, therefore, of an LQID instruction upon the subroutine-save stack is that the contents of SB are placed in SC (SB → SC). Since it pushes the stack, a LQID should not be executed when *three* levels of subroutine nesting are currently in effect. (The last return address in SC will be lost.)

Since, as with the JID instruction, LQID affects only the lower 8 bits of P ($P_9$ and $P_8$ are unchanged), it may only access ROM data located within the current 4-page ROM "block" (pages 0-3, 4-7, 8-11 or 12-15). For further information on the use of the LQID instruction, see Section 4.1.

**X** (eXchange M with A) exchanges the 4-bit contents of RAM pointed to by the B register, $M_3$-$M_0$, with $A_3$-$A_0$. The "r" operand field of the instruction is EXCLUSIVE-ORed with the contents of Br after the exchange to provide a new Br RAM register select value as explained in the LD instruction above.

**XAD** (eXchange A with M Directly) exchanges the 4-bit contents of the RAM memory location pointed to directly by the "r" and "d" operand fields of the instruction, $M_3$-$M_0$, with $A_3$-$A_0$. It has the same characteristics and utility as the LDD instruction above, e.g., the B register is not affected.

**XDS** (eXchange M with A, Decrement Bd and Skip on borrow) performs the same operation as the X instruction above, and also decrements the value of the Bd register (RAM digit-select) *after* the exchange. Use of an "r" operand field will, therefore, result in both an altered RAM *digit*-select value and a new RAM *register* select value in B. XDS skips the next program instruction when Bd is decremented *past* 0 (after the contents of RAM digit 0 have been exchanged with A and XDS decrements Bd to 15). Repeated XDSs will "walk down" through the digits of a RAM register before skipping. XDS together with X instructions can be used to operate upon the corresponding digits of different RAM registers in successive fashion. (See Section 4.2.)

**XIS** (eXchange M with A, Increment Bd, and Skip on carry) performs the same operation as the XDS instruction except that it *increments* Bd *after* the exchange and skips the next program instruction after Bd increments *past* 15 (after the contents of RAM digit 15 have been exchanged with A and XIS increments Bd to 0). Consequently, successive XISs "walk up" through the digits of a RAM register before skipping.

**Register Reference Instructions**

**CAB** (Copy A to Bd) transfers the 4-bit contents of A, $A_3$-$A_0$, to Bd (the RAM digit-select register). This instruction allows the loading of a new RAM digit-select value via the accumulator, a useful operation in many memory-digit access loops.

**CBA** (Copy Bd to A) transfers the 4-bit contents of Bd (RAM digit select) to $A_3$-$A_0$. It is the functional complement of the CAB instruction and finds similar use in memory-digit access loops.

**LBI** (Load B Immediate) loads the B register with the 6-bit value specified by the "r" (2-bit) and "d" (4-bit) fields of the instruction. Its purpose is to directly load a new RAM register and digit select value into B and, unlike CAB, CBA or XABR, does not require use of the accumulator. A further distinction with respect to CAB and CBA is its ability to alter the Br register (RAM register-select).

The LBI instruction is coded or assembled into machine language as *either* a single- or a double-byte instruction, depending on the value of the "d" field. If the "d" field value equals 0 or 9 through 15, the instruction is coded as a single-byte instruction with the lower 6 bits equal to the value of "d" *minus* 1. If the "d" field equals 1 through 8 (1-8), the instruction is coded as a double-byte instruction, with the lower 6 bits of the second byte equal to the value of "d." (See LBI Instruction, Table 3.1, and Note 6 of Table 3.1.)

To take advantage of the more efficient single-byte LBI format, frequently used program data (counters, flags, etc.) should be placed within RAM digit locations accessible by the LBI single-byte "d" field (d = 0, 9-15). (See Section 4.2 for further information.)

An important characteristic of the LBI instruction is that it will skip all subsequent LBI instructions until it encounters an instruction which is not an LBI. This feature accommodates it for use in multiple-entry subroutines. (For example, see Adjacent Memory Move Routine, Section 4.4.)

**LEI** (Load EN Immediate) loads the enable register with the value contained in the "y" operand field of this instruction (0-15, binary). Its function is to select or deselect a particular software selectable feature associated with each of the four bits of the enable register ($EN_3$-$EN_0$). These features and the corresponding bit-weights and values associated with each feature are as follows:

1. The least significant bit of the enable register, $EN_0$, selects the SIO register as either a 4-bit shift register or a 4-bit binary counter.

   With $EN_0$ set, SIO is an asynchronous binary counter, decrementing its value by one upon each low-going pulse ("1" to "0") occurring on the SI input. Each pulse must remain at each logic level at least two instruction cycles. SK outputs the value of the C upon the execution of an XAS and remains latched until the execution of another XAS instruction. The SO output is equal to the value of $EN_3$.

   With $EN_0$ reset, SIO is a serial shift register, shifting continuously left each instruction cycle time. The data present at SI goes into the least significant bit of SIO; SO can be enabled to output the most significant bit of SIO each cycle time. SK output becomes a logic-controlled clock, providing a SYNC signal each instruction time. It will start outputting a SYNC pulse upon the execution of an XAS instruction with C = "1," stopping upon the execution of a subsequent XAS with C = "0."

   If $EN_0$ is changed from "1" to "0" ("0" to "1"), the SK output will change from "1" to SYNC (SYNC to "1") *without* the execution of an XAS instruction.

2. With $EN_1$ set, the $IN_1$ input is enabled as an interrupt input. Upon the occurrence of a negative pulse on $IN_1$, program control is transferred to the last word of page 3 (address $0FF_{16}$). Immediately following an interrupt, $EN_1$ is reset to disable further interrupts until later set by an LEI instruction (usually at the end of the interrupt service routine or later within the main program).

   The following features are associated with the $IN_1$ interrupt procedure and protocol and must be considered by the programmer when utilizing this software-selectable feature of the COP420-series. (Interrupt is unavailable on the COP421-series since it does not have the $IN_3$-$IN_0$ inputs.)

   a. The interrupt, once acknowledged as explained below, pushes the next sequential program counter address (P + 1) onto the stack, pushing in turn the contents of the other subroutine-save registers to the next lower level (P + 1 → SA → SB → SC). Any previous contents of SC are lost. The program counter is set to address $0FF_{16}$ (the last word of page 3) and $EN_1$ is reset.

   b. An interrupt will be acknowledged only after the following conditions are met:

      1) $EN_1$ has been set;

      2) A low-going pulse ("1" to "0") at least two instruction cycles in width has occurred on the $IN_1$ input;

      3) A currently executing instruction has been completed;

      4) All successive transfer of control instructions and successive LBIs have been completed (e.g., if the main program is executing a JP instruction which transfers program control to another JP instruction, the interrupt will not be acknowledged until the second JP instruction has been executed).

   c. Upon acknowledgement of an interrupt, the skip logic status is saved and implemented upon the execution of a subsequent RET instruction. For example, if an interrupt occurs during the execution of ASC (Add with carry, Skip on Carry) instruction which results in a carry, the next instruction (which would normally be skipped) is *not* skipped; instead, its address is pushed onto the stack, the skip logic status is saved and program control is transferred to the interrupt servicing routine at location $0FF_{16}$. At the *end* of the interrupt routine, a RET instruction is executed to pop the stack and return program control to the instruction following the original ACS. *At this time,* the skip logic is enabled and skips this instruction because of the previous ASC carry. Since, as explained above, it is the RET instruction which enables the previously saved status of the skip logic, subroutines should not be nested within the interrupt service routine since their RET instruction will enable any previously saved main program skips, interfering with the orderly execution of the interrupt routine.

   d. The first instruction of the interrupt routine at address $0FF_{16}$ must be NOP.

3. With $EN_2$ set, the L drivers are enabled, loading data previously latched into Q to the L I/O ports. Resetting $EN_2$ disables the L drivers, placing the L I/O ports in a high-impedance state. When the L I/O ports are used as segment drivers to an LED display, the setting and resetting of $EN_2$ results in the outputting and blanking, respectively, of segment data to the display. When using the MICROBUS™ option $EN_2$ does not affect the L drivers.

4. $EN_3$, in conjunction with $EN_0$, affects the SO output. With $EN_0$ set (binary counter option selected) SO will output the value loaded into $EN_3$. With $EN_0$ reset (serial shift register feature selected), setting $EN_3$ enables SO as the output of the SIO shift register, outputting serial shifted data (the most significant bit of SIO) each instruction time as explained above. Resetting $EN_3$ with the serial shift register feature selected disables SO as the shift register output: data continues to be shifted through SIO and can be exchanged with A via an XAS instruction but SO remains reset to "0." Figure 3.2 below provides a summary of the features associated with $EN_3$ and $EN_0$.

| $EN_3$ | $EN_0$ | SIO | SI | SO | SK after XAS |
|---|---|---|---|---|---|
| 0 | 0 | Shift Register | Input to Shift Register | 0 | If C = 1, SK = SYNC<br>If C = 0, SK = 0 |
| 1 | 0 | Shift Register | Input to Shift Register | Serial Out | If C = 1, SK = SYNC<br>If C = 0, SK = 0 |
| 0 | 1 | Binary Counter | Negative Edge Sensitive Input to Binary Counter | 0 | If C = 1, SK = 1<br>If C = 0, SK = 0 |
| 1 | 1 | Binary Counter | Negative Edge Sensitive Input to Binary Counter | 1 | If C = 1, SK = 1<br>If C = 0, SK = 0 |

Figure 3.2 Enable Register Features — Bits $EN_3$ and $EN_0$

**XABR** (eXchange A with Br) exchanges Br (upper 2 bits of B: RAM register-select) with A. Since Br contains only 2 bits, only the lower two bits of A, $A_1$-$A_0$, are placed in Br. Similarly, the 2 bits of Br are placed in $A_1$-$A_0$ with "0s" being loaded into the upper 2 bits of A, $A_3$-$A_2$. XABR is an efficient means of loading the Br register via the accumulator — a direct load of the Br register must otherwise be accomplished by an LBI instruction which also affects the Bd portion of the B register.

**Test Instructions**

**SKC** (SKip on Carry) skips the next program instruction if the carry bit is equal to "1." When used in conjunction with the RC and SC instructions, it allows C to be used as a 1-bit testable flag.

**SKE** (SKip if A Equals M) compares all 4 bits of A with M, skipping the next instruction if the value of A is equal to the value of M. SKE can be used to compare A with a status or counter digit in M, skipping to an instruction which transfers program control to another routine if equality exists.

**SKGBZ** (SKip if G Bit is Zero) is a double-byte instruction. It tests the state of *one* of the four G lines ($G_3$-$G_0$) as specified by the "n" operand of the instruction, skipping the next program instruction if the specified G line is equal to "0."

**SKGZ** (SKip if G is Zero) is a double-byte instruction. It tests the state of all *four* of the G lines, skipping the next program instruction if $G_3$-$G_0$ are all equal to "0."

**SKMBZ** (SKip on Memory Bit Zero) skips the next program instruction if the RAM memory bit specified by the "n" field of the instruction (0-3, right-most to left-most M bit) is equal to "0." This instruction, together with the SMB and RMB instructions, allow for the testing and manipulation of single-bit flags contained within RAM digit locations.

**SKT** (SKip on Timer) instruction tests the state of an internal 10-bit time-base counter. This counter divides the instruction cycle clock frequency by 1024 and provides a latched indication of counter overflow. The SKT instruction tests this latch, executing the next program instruction if the latch is not set. If the latch has been set since the previous test, the next program instruction is skipped and the latch is reset. The features associated with this instruction, therefore, allow the controller to generate its own time-base for real-time processing rather than relying on an external input signal.

For example, using a 2.097 MHz crystal as the time-base to the clock generator, the instruction cycle clock frequency will be 131 kHz (crystal frequency ÷ 16) and the binary counter output pulse frequency will be 128 Hz. For time-of-day or similar real-time processing, the SKT instruction can call a routine which increments a "seconds" counter every 128 ticks.

### 3.3 COP421-Series Instruction Set Differences

The **ININ** instruction has been deleted. This is due to the lack of the IN inputs.

The **INIL** instruction has been substantially modified due to the lack of IN inputs and $IL_3/IL_0$ latches. If an INIL instruction is executed on a COP421-series device, it will input only the state of CKO, providing CKO has been programmed as a general-purpose input ($0 \rightarrow A_3$, $A_1$, $A_0$; $CKO \rightarrow A_2$). If CKO has not been programmed as a general-purpose input, the INIL instruction is non-functional on the COP421-series.

### 3.4 COP410L/COP411L Instruction Set

The COP410L and COP411L instruction sets are subsets of the COP421-series instruction set.

Table 3.3 provides the mnemonic, operand, machine code, data flow, skip conditions and description associated with each instruction in the COP410L and COP411L instruction sets. An asterisk in the description column indicates the double-byte instruction. Notes are provided, following this table, which include additional information relevant to particular instructions.

Table 3.4 provides a list of internal architecture, instruction operand and operational symbols used in the COP410L/COP411L Instruction Set Table. Table 3.7 provides an alphabetical mnemonic index of COP410L/COP411L instructions, indicating the hexadecimal opcode and description associated with each instruction. Table 3.8 is a list of COP410L/COP411L instructions arranged in order of their hexadecimal opcodes.

The following text discusses the differences which exist between the COP410L and COP411L instruction sets and that of the COP420-series. The COP410L is specifically discussed with differences between it and the COP411L noted. All other instructions perform the same machine operations and have the same typical usage as discussed in Section 3.2. For a treatment of the significance of those differences when writing programs for the COP410L and COP411L, see Section 3.5, COP410L/COP411L Instruction Set Differences, and Section 4.11, COP410L/COP411L Programming.

**Table 3.3 COP410L/COP411L Instruction Set**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| ARITHMETIC INSTRUCTIONS | | | | | | |
| ASC | | 30 | 0 0 1 1 0 0 0 0 | A + C + RAM(B) → A<br>Carry → C | Carry | Add with Carry, Skip on Carry |
| ADD | | 31 | 0 0 1 1 0 0 0 1 | A + RAM(B) → A | None | Add RAM to A |
| AISC | y | 5- | 0 1 0 1 y | A + y → A | Carry | Add Immediate, Skip on Carry (y ≠ 0) |
| CLRA | | 00 | 0 0 0 0 0 0 0 0 | 0 → A | None | Clear A |
| COMP | | 40 | 0 1 0 0 0 0 0 0 | $\overline{A} \rightarrow A$ | None | Ones complement of A to A |
| NOP | | 44 | 0 1 0 0 0 1 0 0 | None | None | No Operation |
| RC | | 32 | 0 0 1 1 0 0 1 0 | "0" → C | None | Reset C |
| SC | | 22 | 0 0 1 0 0 0 1 0 | "1" → C | None | Set C |
| XOR | | 02 | 0 0 0 0 0 0 1 0 | A ⊕ RAM(B) → A | None | Exclusive-OR RAM with A |

**Table 3.3 COP410L/COP411L Instruction Set**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **TRANSFER OF CONTROL INSTRUCTIONS** | | | | | | |
| JID | | FF | 1111\|1111 | ROM ($PC_8$,A,M) → $PC_{7:0}$ | None | Jump Indirect (Note 2) |
| JMP | a | 6- | 0110\|000\|$a_8$ | a → PC | None | Jump |
| | | -- | $a_{7:0}$ | | | |
| JP | a | -- | 1\| $a_{6:0}$ (pages 2,3 only) | a → $PC_{6:0}$ | None | Jump within Page (Note 3) |
| | | | or | | | |
| | | -- | 11\| $a_{5:0}$ (all other pages) | a → $PC_{5:0}$ | | |
| JSRP | a | -- | 10\| $a_{5:0}$ | PC + 1 → SA → SB | None | Jump to Subroutine Page (Note 4) |
| | | | | 010 → $PC_{8:6}$<br>a → $PC_{5:0}$ | | |
| JSR | a | 6- | 0110\|100\|$a_8$ | PC + 1 → SA → SB | None | Jump to Subroutine |
| | | -- | $a_{7:0}$ | a → PC | | |
| RET | | 48 | 0100\|1000 | SB → SA → PC | None | Return from Subroutine |
| RETSK | | 49 | 0100\|1001 | SB → SA → PC | Always Skip on Return | Return from Subroutine then Skip |
| **MEMORY REFERENCE INSTRUCTIONS** | | | | | | |
| CAMQ | | 33 | 0011\|0011 | A → $Q_{7:4}$ | None | Copy A, RAM to Q |
| | | 3C | 0011\|1100 | RAM(B) → $Q_{3:0}$ | | |
| LD | r | -5 | 00\|r\|0101 | RAM(B) → A<br>Br ⊕ r → Br | None | Load RAM into A, Exclusive-OR Br with r |
| LQID | | BF | 1011\|1111 | ROM($PC_8$,A,M) → Q<br>SA → SB | None | Load Q Indirect (Note 2) |
| RMB | 0 | 4C | 0100\|1100 | 0 → $RAM(B)_0$ | None | Reset RAM Bit |
| | 1 | 45 | 0100\|0101 | 0 → $RAM(B)_1$ | | |
| | 2 | 42 | 0100\|0010 | 0 → $RAM(B)_2$ | | |
| | 3 | 43 | 0100\|0011 | 0 → $RAM(B)_3$ | | |
| SMB | 0 | 4D | 0100\|1101 | 1 → $RAM(B)_0$ | None | Set RAM Bit |
| | 1 | 47 | 0100\|0111 | 1 → $RAM(B)_1$ | | |
| | 2 | 46 | 0100\|0110 | 1 → $RAM(B)_2$ | | |
| | 3 | 4B | 0100\|1011 | 1 → $RAM(B)_3$ | | |
| STII | y | 7- | 0111\| y | y → RAM(B)<br>Bd + 1 → Bd | None | Store Memory Immediate and Increment Bd |
| X | r | -6 | 00\|r\|0110 | RAM(B) ⟷ A<br>Br ⊕ r → Br | None | Exchange RAM with A, Exclusive-OR Br with r |
| XAD | 3,15 | 23 | 0010\|0011 | RAM(3,15) ⟷ A | None | * Exchange A with RAM (3,15) |
| | | BF | 1011\|1111 | | | |
| XDS | r | -7 | 00\|r\|0111 | RAM(B) ⟷ A<br>Bd − 1 → Bd<br>Br ⊕ r → Br | Bd decrements past 0 | Exchange RAM with A and Decrement Bd, Exclusive-OR Br with r |
| XIS | r | -4 | 00\|r\|0100 | RAM(B) ⟷ A<br>Bd + 1 → Bd<br>Br ⊕ r → Br | Bd increments past 15<br>Exclusive-OR Br with r | Exchange RAM with A and Increment Bd, |

**Table 3.3 COP410L/COP411L Instruction Set**

| Mnemonic | Operand | Hex Code | Machine Language Code (Binary) | Data Flow | Skip Conditions | Description |
|---|---|---|---|---|---|---|
| **REGISTER REFERENCE INSTRUCTIONS** | | | | | | |
| CAB | | 50 | 0 1 0 1 \| 0 0 0 0 | A → Bd | None | Copy A to Bd |
| CBA | | 4E | 0 1 0 0 \| 1 1 1 0 | Bd → A | None | Copy Bd to A |
| LBI | r,d | -- | 0 0 \| r \| (d – 1) (d = 0, 9:15) | r,d → B | None | Load B Immediate with r,d (Note 5) |
| LEI | y | 33<br>6- | 0 0 1 1 \| 0 0 1 1<br>0 1 1 0 \| y | y → EN | None | • Load EN Immediate (Note 6) |
| **TEST INSTRUCTIONS** | | | | | | |
| SKC | | 20 | 0 0 1 0 \| 0 0 0 0 | | C = "1" | Skip if C is True |
| SKE | | 21 | 0 0 1 0 \| 0 0 0 1 | | A = RAM(B) | Skip if A Equals RAM |
| SKGZ | | 33<br>21 | 0 0 1 1 \| 0 0 1 1<br>0 0 1 0 \| 0 0 0 1 | | $G_{3:0}$ = 0 | Skip if G is Zero (all 4 bits) |
| SKGBZ | | 33 | 0 0 1 1 \| 0 0 1 1 | 1st byte | | • Skip if G Bit is Zero |
| | 0 | 01 | 0 0 0 0 \| 0 0 0 1 | 2nd byte | $G_0$ = 0 | |
| | 1 | 11 | 0 0 0 1 \| 0 0 0 1 | | $G_1$ = 0 | |
| | 2 | 03 | 0 0 0 0 \| 0 0 1 1 | | $G_2$ = 0 | |
| | 3 | 13 | 0 0 0 1 \| 0 0 1 1 | | $G_3$ = 0 | |
| SKMBZ | 0 | 01 | 0 0 0 0 \| 0 0 0 1 | | $RAM(B)_0$ = 0 | Skip if RAM Bit is Zero |
| | 1 | 11 | 0 0 0 1 \| 0 0 0 1 | | $RAM(B)_1$ = 0 | |
| | 2 | 03 | 0 0 0 0 \| 0 0 1 1 | | $RAM(B)_2$ = 0 | |
| | 3 | 13 | 0 0 0 1 \| 0 0 1 1 | | $RAM(B)_3$ = 0 | |
| **INPUT/OUTPUT INSTRUCTIONS** | | | | | | |
| ING | | 33<br>2A | 0 0 1 1 \| 0 0 1 1<br>0 0 1 0 \| 1 0 1 0 | G → A | None | • Input G Ports to A |
| INL | | 33<br>2E | 0 0 1 1 \| 0 0 1 1<br>0 0 1 0 \| 1 1 1 0 | $L_{7:4}$ → RAM(B)<br>$L_{3:0}$ → A | None | • Input L Ports to RAM, A |
| OBD | | 33<br>3E | 0 0 1 1 \| 0 0 1 1<br>0 0 1 1 \| 1 1 1 0 | Bd → D | None | • Output Bd to D Outputs |
| OMG | | 33<br>3A | 0 0 1 1 \| 0 0 1 1<br>0 0 1 1 \| 1 0 1 0 | RAM(B) → G | None | • Output RAM to G Ports |
| XAS | | 4F | 0 1 0 0 \| 1 1 1 1 | A ←→ SIO, C → SK | None | Exchange A with SIO (Note 2) |

**Note 1:** All subscripts for alphabetical symbols indicate bit numbers unless explicitly defined (e.g., Br and Bd are explicitly defined). Bits are numbered 0 to N where 0 signifies the least significant (low-order, right-most bit). For example, $A_3$ indicates the most significant (left-most) bit of the 4-bit A register.

**Note 2:** For additional information on the operation of the XAS, JID, and LQID instructions, see Section 3.2.

**Note 3:** The JP instruction allows a jump, while in subroutine pages 2 or 3, to any ROM location within the two-page boundary of pages 2 or 3. The JP instruction, otherwise, permits a jump to a ROM location within the current 64-word page. JP may not jump to the last word of a page.

**Note 4:** A JSRP transfers program control to subroutine page 2 (0010 is loaded into the upper 4 bits of P). A JSRP may not be used when in pages 2 or 3. JSRP may not jump to the last word in page 2.

**Note 5:** LBI is a single-byte instruction if d = 0, 9, 10, 11, 12, 13, 14, or 15. The machine code for the lower 4 bits equals the binary value of the "d" data *minus 1*, e.g., to load the lower four bits of B (Bd) with the value 9 ($1001_2$), the lower 4 bits of the LBI instruction equal 8 ($1000_2$). To load 0, the lower 4 bits of the LBI instruction should equal 15 ($1111_2$).

**Note 6:** Machine code for operand field y for LEI instruction should equal the binary value to be latched into EN, where a "1" or "0" in each bit of EN corresponds with the selection or deselection of a particular function associated with each bit. (See Functional Description, EN Register.)

Table 3.4 COP410L/411L Instruction Set Table Symbols

| Symbol | Definition |
|---|---|
| INTERNAL ARCHITECTURE SYMBOLS | |
| A | 4-bit Accumulator |
| B | 6-bit RAM Address Register |
| Br | Upper 2 bits of B (register address) |
| Bd | Lower 4 bits of B (digit address) |
| C | 1-bit Carry Register |
| D | 4-bit Data Output Port |
| EN | 4-bit Enable Register |
| G | 4-bit Register to latch data for G I/O Port |
| L | 8-bit TRI-STATE I/O Port |
| M | 4-bit contents of RAM Memory pointed to by B Register |
| PC | 9-bit ROM Address Register (program counter) |
| Q | 8-bit Register to latch data for L I/O Port |
| SA | 9-bit Subroutine Save Register A |
| SB | 9-bit Subroutine Save Register B |
| SIO | 4-bit Shift Register and Counter |
| SK | Logic-Controlled Clock Output |

| Symbol | Definition |
|---|---|
| INSTRUCTION OPERAND SYMBOLS | |
| d | 4-bit Operand Field, 0-15 binary (RAM Digit Select) |
| r | 2-bit Operand Field, 0-3 binary (RAM Register Select) |
| a | 9-bit Operand Field, 0-511 binary (ROM Address) |
| y | 4-bit Operand Field, 0-15 binary (Immediate Data) |
| RAM(s) | Contents of RAM location addressed by s |
| ROM(t) | Contents of ROM location addressed by t |
| OPERATIONAL SYMBOLS | |
| + | Plus |
| − | Minus |
| → | Replaces |
| ←→ | Is exchanged with |
| = | Is equal to |
| A | The ones complement of A |
| ⊕ | Exclusive-OR |
| : | Range of values |

## 3.5 COP410L/COP411L Instruction Set Differences

### Arithmetic Instructions

**ADT** has been deleted. To perform a similar operation an AISC 10 followed by a NOP to defeat the skip condition (carry) may be used.

**CASC** has been deleted. A COMP instruction followed by an ASC will achieve the same result (subtraction of A from M).

### Input/Output Instructions

**ININ** has been deleted due to the COP410L's lack of IN inputs.

**OGI** has been deleted. A loading of data to the G ports must be accomplished via M by first loading M and then outputting its contents to G via an OMG instruction.

### Memory Reference Instructions

**CQMA** has been deleted. Since no MICROBUS™ option is provided for the COP410L, Q is used in the COP410L primarily for output operations. An input of the L I/O ports, therefore, will effectively function as the equivalent of a CQMA; this is accomplished by the execution of an INL instruction.

**LDD** has been deleted. To load the contents of a data memory digit location into A, the usual procedure of loading B via an LBI to point to a particular RAM location followed by an LD instruction must be used.

**XAD** has been altered to reference *one* data memory location only; specifically, M(3,15). "Scratch-pad" data to be exchanged with A without affecting the B register should be placed, therefore, in M(3,15) and accessed by the XAD 3,15 instruction.

### Register Reference Instructions

**LBI** has been altered to correspond to the data memory configuration of the COP410L. Specifically, it may only be used to access valid RAM locations, namely digits 9 through 15 and 0 in registers 0-3. The LBI "d" field, therefore, is limited to "d" values of 9-15 and 0, resulting in *all LBIs* being coded as *single-byte* instructions. Remember, the *machine code* for the "d" operand field is the binary value of "d" minus 1.

**XABR** has been deleted. To load Br, the entire B register must be loaded via an LBI. Altering Br may also be accomplished by using the EXCLUSIVE-OR "r" field associated with the memory reference instructions LD, X, XDS, and XIS.

### Test Instructions

**SKT** has been deleted since the COP410L does not contain an internal divide-by-1024 time-base counter.

**Table 3.5 Alphabetical Mnemonic Index of COP420/COP421-Series Instructions**

| Instruction | Hexadecimal Opcode | Description |
|---|---|---|
| ADD | 31 | ADD RAM to A |
| ADT | 4A | ADd Ten to A |
| AISC 1-15 | 51-5F | Add Immediate, Skip on Carry |
| ASC | 30 | Add with carry, Skip on Carry |
| CAB | 50 | Copy A to Bd |
| CAMQ* | 33/3C | Copy A, RAM to Q |
| CASC | 10 | Complement and Add with carry, Skip on Carry |
| CBA | 4E | Copy Bd to A |
| CLRA | 00 | CLeaR A |
| COMP | 40 | COMPlement A |
| CQMA* | 33/2C | Copy Q to RAM, A |
| ING* | 33/2A | INput G ports to A |
| INIL* | 33/29 | INput IL latches to A** |
| ININ* | 33/28 | INput IN inputs to A** |
| INL* | 33/2E | INput L ports to RAM, A |
| JID | FF | Jump InDirect |
| JMP* | 60-63/00-FF | JuMP |
| JP | 80-BE,C0-CE | Jump within Page |
| JSR* | 68-6B/00-FF | Jump to SubRoutine |
| JSRP | 80-BE | Jump to SubRoutine Page |
| LBI 0,9-15,0 | 08-0F | Load Bd Immediate (single-byte) |
| LBI 1,9-15,0 | 18-1F | |
| LBI 2,9-15,0 | 28-2F | |
| LBI 3,9-15,0 | 38-3F | |
| LBI* 0,1-8 | 33/81-88 | Load Bd Immediate (double-byte) |
| LBI* 1,1-8 | 33/91-98 | |
| LBI* 2,1-8 | 33/A1-A8 | |
| LBI* 3,1-8 | 33/B1-B8 | |
| LD 0,1,2,3 | 05,15,25,35 | LoaD RAM into A |
| LDD* 0-3,0-15 | 23/00-3F | LoaD A with RAM, Directly |
| LEI* 0-15 | 33/60-6F | Load EN Immediate |
| LQID | BF | Load Q InDirect |
| NOP | 44 | No OPeration |
| OBD* | 33/3E | Output Bd to D outputs |
| OGI* | 33/50-5F | Output to G ports Immediate |
| OMG* | 33/3A | Output RAM to G ports |
| RC | 32 | Reset Carry |
| RET | 48 | RETurn |
| RETSK | 49 | RETurn then SKip |
| RMB 0,1,2,3 | 4C,45,42,43 | Reset Memory Bit |
| SC | 22 | Set Carry |
| SMB 0,1,2,3 | 4D,47,46,4B | Set Memory Bit |
| SKC | 20 | SKip if Carry is true |
| SKE | 21 | SKip if A Equals RAM |
| SKGBZ* 0,1,2,3 | 33/01,11,03,13 | SKip if G Bit is Zero |
| SKGZ* | 33/21 | SKip if G equals Zero (all 4 bits) |
| SKMBZ 0,1,2,3 | 01,11,03,13 | SKip if Memory Bit is Zero |
| SKT | 41 | SKip on Timer |
| STII | 70-7F | STore memory Immediate and Increment Bd |
| X 0,1,2,3 | 6,16,26,36 | eXchange RAM with A, exclusive-OR r with Br |
| XABR | 12 | eXchange A with Br |
| XAD* 0-3,0-15 | 23/80-BF | eXchange A with RAM Directly |
| XAS | 4F | eXchange A with SIO (serial I/O) |
| XDS 0,1,2,3 | 07,17,27,37 | eXchange RAM with A and Decrement Bd |
| XIS 0,1,2,3 | 04,14,24,34 | eXchange RAM with A and Increment Bd |
| XOR | 02 | eXclusive-OR RAM with A |

*Double-Byte Instruction: first byte/second byte (or first byte range/second byte range).

**Instruction not available or has different features on COP421-series.

**Table 3.6 Table of COP420/COP421-Series Instructions Listed by Opcodes (Hexadecimal)**

| | | | |
|---|---|---|---|
| 00 | CLRA | 26 | X 2 |
| 01 | SKMBZ 0 | 27 | XDS 2 |
| 02 | XOR | 28 | LBI 2,9 |
| 03 | SKMBZ 2 | 29 | LBI 2,10 |
| 04 | XIS 0 | 2A | LBI 2,11 |
| 05 | LD 0 | 2B | LBI 2,12 |
| 06 | X 0 | 2C | LBI 2,13 |
| 07 | XDS 0 | 2D | LBI 2,14 |
| 08 | LBI 0,9 | 2E | LBI 2,15 |
| 09 | LBI 0,10 | 2F | LBI 2,0 |
| 0A | LBI 0,11 | 30 | ASC |
| 0B | LBI 0,12 | 31 | ADD |
| 0C | LBI 0,13 | 32 | RC |
| 0D | LBI 0,14 | 33 | TWO WORD* (except LDD, XAD, JMP, JSR) |
| 0E | LBI 0,15 | | |
| 0F | LBI 0,0 | | |
| 10 | CASC | 34 | XIS 3 |
| 11 | SKMBZ 1 | 35 | LD 3 |
| 12 | XABR | 36 | X 3 |
| 13 | SKMBZ 3 | 37 | XDS 3 |
| 14 | XIS 0 | 38 | LBI 3,9 |
| 15 | LD 1 | 39 | LBI 3,10 |
| 16 | X 1 | 3A | LBI 3,11 |
| 17 | XDS 1 | 3B | LBI 3,12 |
| 18 | LBI 1,9 | 3C | LBI 3,13 |
| 19 | LBI 1,10 | 3D | LBI 3,14 |
| 1A | LBI 1,11 | 3E | LBI 3,15 |
| 1B | LBI 1,12 | 3F | LBI 3,0 |
| 1C | LBI 1,13 | 40 | COMP |
| 1D | LBI 1,14 | 41 | SKT |
| 1E | LBI 1,15 | 42 | RMB 2 |
| 1F | LBI 1,0 | 43 | RMB 3 |
| 20 | SKC | 44 | NOP |
| 21 | SKE | 45 | RMB 1 |
| 22 | SC | 46 | SMB 2 |
| 23 | LDD/XAD** | 47 | SMB 1 |
| 24 | XIS 2 | 48 | RET |
| 25 | LD 2 | 49 | RETSK |

**Table 3.6 Table of COP420/COP421-Series Instructions Listed by Opcodes (Hexadecimal) (continued)**

| | |
|---|---|
| 4A | ADT |
| 4B | SMB 3 |
| 4C | RMB 0 |
| 4D | SMB 0 |
| 4E | CBA |
| 4F | XAS |
| 50 | CAB |
| 51 | AISC 1 |
| 52 | AISC 2 |
| 53 | AISC 3 |
| 54 | AISC 4 |
| 55 | AISC 5 |
| 56 | AISC 6 |
| 57 | AISC 7 |
| 58 | AISC 8 |
| 59 | AISC 9 |
| 5A | AISC 10 |
| 5B | AISC 11 |
| 5C | AISC 12 |
| 5D | AISC 13 |
| 5E | AISC 14 |
| 5F | AISC 15 |
| 60 | JMP*** to Page 0, 1, 2, or 3 |
| 61 | JMP*** to Page 4, 5, 6, or 7 |
| 62 | JMP*** to Page 8, 9, 10, or 11 |
| 63 | JMP*** to Page 12, 13, 14, or 15 |
| 64 | invalid |
| 65 | invalid |
| 66 | invalid |
| 67 | invalid |
| 68 | JSR*** to Page 0, 1, 2, or 3 |
| 69 | JSR*** to Page 4, 5, 6, or 7 |
| 6A | JSR*** to Page 8, 9, 10, or 11 |
| 6B | JSR*** to Page 12, 13, 14, or 15 |
| 6C | invalid |
| 6D | invalid |
| 6E | invalid |
| 6F | invalid |
| 70 | STII 0 |
| 71 | STII 1 |
| 72 | STII 2 |
| 73 | STII 3 |
| 74 | STII 4 |
| 75 | STII 5 |
| 76 | STII 6 |
| 77 | STII 7 |
| 78 | STII 8 |
| 79 | STII 9 |
| 7A | STII 10 |
| 7B | STII 11 |
| 7C | STII 12 |
| 7D | STII 13 |
| 7E | STII 14 |
| 7F | STII 15 |
| 80-BE | JP to word XX ($0\text{-}3F_{16}$) or JSRP to page 2, word XX ($0\text{-}3F_{16}$): opcode = 80 + XX |
| BF | LQID |
| C0-CE | JP to word XX ($0\text{-}3F_{16}$): opcode = C0 + XX |
| FF | JID |
| | **Two Word Instructions, Second Word:** |
| *00 | INIL (different features for COP421) |
| 01 | SKGBZ 0 |
| 03 | SKGBZ 2 |
| 11 | SKGBZ 1 |
| 13 | SKGBZ 3 |
| 21 | SKGZ |
| 28 | ININ (Invalid for COP421) |
| 2A | ING |
| 2C | CQMA |
| 2E | INL |
| 3A | OMG |
| 3C | CAMQ |
| 3E | OBD |
| 50 | OGI 0 |
| 51 | OGI 1 |
| 52 | OGI 2 |
| 53 | OGI 3 |
| 54 | OGI 4 |
| 55 | OGI 5 |
| 56 | OGI 6 |
| 57 | OGI 7 |
| 58 | OGI 8 |
| 59 | OGI 9 |
| 5A | OGI 10 |
| 5B | OGI 11 |
| 5C | OGI 12 |
| 5D | OGI 13 |
| 5E | OGI 14 |
| 5F | OGI 15 |
| 60 | LEI 0 |
| 61 | LEI 1 |
| 62 | LEI 2 |
| 63 | LEI 3 |
| 64 | LEI 4 |
| 65 | LEI 5 |
| 66 | LEI 6 |
| 67 | LEI 7 |
| 68 | LEI 8 |
| 69 | LEI 9 |
| 6A | LEI 10 |
| 6B | LEI 11 |
| 6C | LEI 12 |
| 6D | LEI 13 |
| 6E | LEI 14 |
| 6F | LEI 15 |
| 81 | LBI 0,1 |
| 82 | LBI 0,2 |
| 83 | LBI 0,3 |
| 84 | LBI 0,4 |
| 85 | LBI 0,5 |
| 86 | LBI 0,6 |
| 87 | LBI 0,7 |
| 88 | LBI 0,8 |
| 91 | LBI 1,1 |
| 92 | LBI 1,2 |
| 93 | LBI 1,3 |
| 94 | LBI 1,4 |
| 95 | LBI 1,5 |
| 96 | LBI 1,6 |
| 97 | LBI 1,7 |
| 98 | LBI 1,8 |
| A1 | LBI 2,1 |
| A2 | LBI 2,2 |
| A3 | LBI 2,3 |
| A4 | LBI 2,4 |
| A5 | LBI 2,5 |
| A6 | LBI 2,6 |
| A7 | LBI 2,7 |
| A8 | LBI 2,8 |
| B1 | LBI 3,1 |
| B2 | LBI 3,2 |
| B3 | LBI 3,3 |
| B4 | LBI 3,4 |
| B5 | LBI 3,5 |
| B6 | LBI 3,6 |
| B7 | LBI 3,7 |
| B8 | LBI 3,8 |
| **00 | LDD 0,0 |
| 01 | LDD 0,1 |
| 02 | LDD 0,2 |
| 03 | LDD 0,3 |
| 04 | LDD 0,4 |
| 05 | LDD 0,5 |
| 06 | LDD 0,6 |
| 07 | LDD 0,7 |
| 08 | LDD 0,8 |
| 09 | LDD 0,9 |
| 0A | LDD 0,10 |
| 0B | LDD 0,11 |
| 0C | LDD 0,12 |
| 0D | LDD 0,13 |
| 0E | LDD 0,14 |
| 0F | LDD 0,15 |
| 10 | LDD 1,0 |
| 11 | LDD 1,1 |
| 12 | LDD 1,2 |
| 13 | LDD 1,3 |
| 14 | LDD 1,4 |
| 15 | LDD 1,5 |
| 16 | LDD 1,6 |
| 17 | LDD 1,7 |
| 18 | LDD 1,8 |
| 19 | LDD 1,9 |
| 1A | LDD 1,10 |
| 1B | LDD 1,11 |
| 1C | LDD 1,12 |
| 1D | LDD 1,13 |
| 1E | LDD 1,14 |
| 1F | LDD 1,15 |
| 20 | LDD 2,0 |
| 21 | LDD 2,1 |
| 22 | LDD 2,2 |
| 23 | LDD 2,3 |
| 24 | LDD 2,4 |
| 25 | LDD 2,5 |
| 26 | LDD 2,6 |
| 27 | LDD 2,7 |
| 28 | LDD 2,8 |
| 29 | LDD 2,9 |
| 2A | LDD 2,10 |
| 2B | LDD 2,11 |
| 2C | LDD 2,12 |
| 2D | LDD 2,13 |
| 2E | LDD 2,14 |
| 2F | LDD 2,15 |
| 30 | LDD 3,0 |
| 31 | LDD 3,1 |
| 32 | LDD 3,2 |
| 33 | LDD 3,3 |
| 34 | LDD 3,4 |
| 35 | LDD 3,5 |
| 36 | LDD 3,6 |
| 37 | LDD 3,7 |
| 38 | LDD 3,8 |
| 39 | LDD 3,9 |
| 3A | LDD 3,10 |
| 3B | LDD 3,11 |
| 3C | LDD 3,12 |
| 3D | LDD 3,13 |
| 3E | LDD 3,14 |
| 3F | LDD 3,15 |
| 80 | XAD 0,0 |
| 81 | XAD 0,1 |
| 82 | XAD 0,2 |
| 83 | XAD 0,3 |
| 84 | XAD 0,4 |
| 85 | XAD 0,5 |
| 86 | XAD 0,6 |
| 87 | XAD 0,7 |
| 88 | XAD 0,8 |
| 89 | XAD 0,9 |
| 8A | XAD 0,10 |
| 8B | XAD 0,11 |

**Table 3.6 Table of COP420/COP421-Series Instructions Listed by Opcodes (Hexadecimal) (continued)**

| | |
|---|---|
| 8C | XAD 0,12 |
| 8D | XAD 0,13 |
| 8E | XAD 0,14 |
| 8F | XAD 0,15 |
| 90 | XAD 1,0 |
| 91 | XAD 1,1 |
| 92 | XAD 1,2 |
| 93 | XAD 1,3 |
| 94 | XAD 1,4 |
| 95 | XAD 1,5 |
| 96 | XAD 1,6 |
| 97 | XAD 1,7 |
| 98 | XAD 1,8 |
| 99 | XAD 1,9 |
| 9A | XAD 1,10 |
| 9B | XAD 1,11 |
| 9C | XAD 1,12 |
| 9D | XAD 1,13 |
| 9E | XAD 1,14 |
| 9F | XAD 1,15 |
| A0 | XAD 2,0 |
| A1 | XAD 2,1 |
| A2 | XAD 2,2 |
| A3 | XAD 2,3 |
| A4 | XAD 2,4 |
| A5 | XAD 2,5 |
| A6 | XAD 2,6 |
| A7 | XAD 2,7 |
| A8 | XAD 2,8 |
| A9 | XAD 2,9 |
| AA | XAD 2,10 |
| AB | XAD 2,11 |
| AC | XAD 2,12 |
| AD | XAD 2,13 |
| AE | XAD 2,14 |
| AF | XAD 2,15 |
| B0 | XAD 3,0 |
| B1 | XAD 3,1 |
| B2 | XAD 3,2 |
| B3 | XAD 3,3 |
| B4 | XAD 3,4 |
| B5 | XAD 3,5 |
| B6 | XAD 3,6 |
| B7 | XAD 3,7 |
| B8 | XAD 3,8 |
| B9 | XAD 3,9 |
| BA | XAD 3,10 |
| BB | XAD 3,11 |
| BC | XAD 3,12 |
| BD | XAD 3,13 |
| BE | XAD 3,14 |
| BF | XAD 3,15 |

***00 + XX JSR or JMP to page 0, 4, 10, or 14, word XX ($03F_{16}$): 0-3F
40 + XX JSR or JMP to page 1, 5, 11, or 15, word XX ($0-3F_{16}$):40-7F
80 + XX JSR or JMP to page 2, 6, 12, or 16, word XX ($0-3F_{16}$):80-BF
C0 + XX JSR or JMP to page 3, 7, 13, or 17, word XX ($0-3F_{16}$):C0-FF

**Table 3.7 Alphabetical Mnemonic Index of COP410L/COP411L-Series Instructions**

| Instruction | Hexadecimal Opcode | Description |
|---|---|---|
| ADD | 31 | ADD RAM to A |
| AISC 1-15 | 51-5F | Add Immediate, Skip on Carry |
| ASC | 30 | Add with carry, Skip on Carry |
| CAB | 50 | Copy A to Bd |
| CAMQ | 33/3C | Copy A, RAM to Q |
| CBA | 4E | Copy Bd to A |
| CLRA | 00 | CLeaR A |
| COMP | 40 | COMPlement A |
| ING* | 33/2A | INput G ports to A |
| INL* | 33/2E | INput L ports to RAM, A |
| JID | FF | Jump InDirect |
| JMP* | 60-61/00-FF | JuMP |
| JP | 80-BE,C0-CE | Jump within Page |
| JSR* | 68-69/00-FF | Jump to SubRoutine |
| JSRP | 80-BE | Jump to SubRoutine Page |
| LBI 0;9-15,0 | 08-0F | Load Bd Immediate (single-byte) |
| LBI 1;9-15,0 | 18-1F | |
| LBI 2;9-15,0 | 28-2F | |
| LBI 3;9-15,0 | 38-3F | |
| LD 0,1,2,3 | 05,15,25,35 | LoaD RAM into A |
| LEI* 0-15 | 33/60-6F | Load EN Immediate |
| LQID | BF | Load Q inDirect |
| NOP | 44 | No OPeration |
| OBD* | 33/3E | Output Bd to D outputs |
| OMG* | 33/3A | Output RAM to G ports |
| RC | 32 | Reset Carry |
| RET | 48 | RETurn |
| RETSK | 49 | RETurn then SKip |
| RMB 0,1,2,3 | 4C,45,42,43 | Reset Memory Bit |
| SC | 22 | Set Carry |
| SMB 0,1,2,3 | 4D,47,46,4B | Set Memory Bit |
| SKC | 20 | SKip if Carry is true |
| SKE | 21 | SKip if A Equals RAM |
| SKGBZ* 0,1,2,3 | 33/01,11,03,13 | SKip if G Bit is Zero |
| SKGZ* | 33/21 | SKip if G equals Zero (all 4 bits) |
| SKMBZ 0,1,2,3 | 01,11,03,13 | SKip if Memory Bit is Zero |
| STII | 70-7F | STore memory Immediate and Increment Bd |
| X 0,1,2,3 | 6,16,26,36 | eXchange RAM with A |
| XAD* 3,15 | 23-BF | eXchange A with RAM Directly |
| XAS | 4F | eXchange A with SIO (serial I/O) |
| XDS 0,1,2,3 | 07,17,27,37 | eXchange RAM with A and Decrement Bd |
| XIS 0,1,2,3 | 04,14,24,34 | eXchange RAM with A and Increment Bd |
| XOR | 02 | eXclusive-OR RAM with A |

*Double-Byte Instruction: first byte/second byte (or first byte range/second byte range)
**Instruction not available or has different features on COP421-series

**Table 3.8 Table of COP410L/COP411L-Series Instructions Listed by Opcodes (Hexadecimal) (continued)**

| | |
|---|---|
| 00 | CLRA |
| 01 | SKMBZ 0 |
| 02 | * XOR |
| 03 | SKMBZ 2 |
| 04 | XIS 0 |
| 05 | LD 0 |
| 06 | X 0 |
| 07 | XDS 0 |
| 08 | LBI 0,9 |
| 09 | LBI 0,10 |
| 0A | LBI 0,11 |
| 0B | LBI 0,12 |
| 0C | LBI 0,13 |
| 0D | LBI 0,14 |
| 0E | LBI 0,15 |
| 0F | LBI 0,0 |
| 10 | invalid |
| 11 | SKMBZ 1 |
| 12 | invalid |
| 13 | SKMBZ 3 |
| 14 | XIS 0 |
| 15 | LD 1 |
| 16 | X 1 |
| 17 | XDS 1 |
| 18 | LBI 1,9 |
| 19 | LBI 1,10 |
| 1A | LBI 1,11 |
| 1B | LBI 1,12 |
| 1C | LBI 1,13 |
| 1D | LBI 1,14 |
| 1E | LBI 1,15 |
| 1F | LBI 1,0 |
| 20 | SKC |
| 21 | SKE |
| 22 | SC |
| 23 | XAD** |
| 24 | XIS 2 |
| 25 | LD 2 |
| 26 | X 2 |
| 27 | XDS 2 |
| 28 | LBI 2,9 |
| 29 | LBI 2,10 |
| 2A | LBI 2,11 |
| 2B | LBI 2,12 |
| 2C | LBI 2,13 |
| 2D | LBI 2,14 |
| 2E | LBI 2,15 |
| 2F | LBI 2,0 |
| 30 | ASC |
| 31 | ADD |
| 32 | RC |
| 33 | TWO WORD* (except XAD, JMP, JSR) |
| 34 | XIS 3 |
| 35 | LD 3 |
| 36 | X 3 |
| 37 | XDS 3 |
| 38 | LBI 3,9 |
| 39 | LBI 3,10 |
| 3A | LBI 3,11 |
| 3B | LBI 3,12 |
| 3C | LBI 3,13 |
| 3D | LBI 3,14 |
| 3E | LBI 3,15 |
| 3F | LBI 3,0 |
| 40 | COMP |
| 41 | invalid |
| 42 | RMB 2 |
| 43 | RMB 3 |
| 44 | NOP |
| 45 | RMB 1 |
| 46 | SMB 2 |
| 47 | SMB 1 |
| 48 | RET |
| 49 | RETSK |
| 4A | invalid |
| 4B | SMB 3 |
| 4C | RMB 0 |
| 4D | SMB 0 |
| 4E | CBA |
| 4F | XAS |
| 50 | CAB |
| 51 | AISC 1 |
| 52 | AISC 2 |
| 53 | AISC 3 |
| 54 | AISC 4 |
| 55 | AISC 5 |
| 56 | AISC 6 |
| 57 | AISC 7 |
| 58 | AISC 8 |
| 59 | AISC 9 |
| 5A | AISC 10 |
| 5B | AISC 11 |
| 5C | AISC 12 |
| 5D | AISC 13 |
| 5E | AISC 14 |
| 5F | AISC 15 |
| 60 | JMP*** to Page 0, 1, 2, or 3 |
| 61 | JMP*** to Page 4, 5, 6, or 7 |
| 64 | invalid |
| 65 | invalid |
| 66 | invalid |
| 67 | invalid |
| 68 | JSR*** to Page 0, 1, 2, or 3 |
| 69 | JSR*** to Page 4, 5, 6, or 7 |
| 6C | invalid |
| 6D | invalid |
| 6E | invalid |
| 6F | invalid |
| 70 | STII 0 |
| 71 | STII 1 |
| 72 | STII 2 |
| 73 | STII 3 |
| 74 | STII 4 |
| 75 | STII 5 |
| 76 | STII 6 |
| 77 | STII 7 |
| 78 | STII 8 |
| 79 | STII 9 |
| 7A | STII 10 |
| 7B | STII 11 |
| 7C | STII 12 |
| 7D | STII 13 |
| 7E | STII 14 |
| 7F | STII 15 |
| 80-BE | JP to word XX ($0\text{-}3F_{16}$) or JSRP to page 2, word XX ($0\text{-}3F_{16}$): opcode = 80 + XX |
| BF | LQID |
| C0-CE | JP to word XX ($0\text{-}3F_{16}$): opcode = C0 + XX |
| FF | JID |

**Two Word Instructions, Second Word:**

| | |
|---|---|
| *00 | invalid |
| 01 | SKGBZ 0 |
| 03 | SKGBZ 2 |
| 11 | SKGBZ 1 |
| 13 | SKGBZ 3 |
| 21 | SKGZ |
| 28 | invalid |
| 2A | ING |
| 2C | invalid |
| 2E | INL |
| 3A | OMG |
| 3C | CAMQ |
| 3E | OBD |
| 50-5F | invalid |
| 60 | LEI 0 |
| 61 | LEI 1 |
| 62 | LEI 2 |
| 63 | LEI 3 |
| 64 | LEI 4 |
| 65 | LEI 5 |
| 66 | LEI 6 |
| 67 | LEI 7 |
| 68 | LEI 8 |
| 69 | LEI 9 |
| 6A | LEI 10 |
| 6B | LEI 11 |
| 6C | LEI 12 |
| 6D | LEI 13 |
| 6E | LEI 14 |
| 6F | LEI 15 |
| 81-88 | invalid |
| 91-98 | invalid |
| A1-A8 | invalid |
| B1-B8 | invalid |

**00-BE Invalid BF → XAD 3,15

***00 + XX JSR or JMP to page 0 or 4 word XX ($03F_{16}$): 0-3F
40 + XX JSR or JMP to page 1 or 5 word XX ($0\text{-}3F_{16}$):40-7F
80 + XX JSR or JMP to page 2 or 6 word XX ($0\text{-}3F_{16}$):80-BF
C0 + XX JSR or JMP to page 3 or 7 word XX ($0\text{-}3F_{16}$):C0-FF

# 4 COP400 Programming Techniques

This chapter provides several examples of programming techniques for COP400 devices. The COP420-series/COP444L instruction set is assumed since it falls between the smaller and larger instruction sets, respectively, of the COP410L and the COP440. For users of the COP410L/COP411L, Section 3.5 provides information on use of multiple COP410L instructions to simulate the function of COP420 instructions not provided for the COP410L. Users of the COP440 will find all examples relevant since this device contains all COP420 instructions as well as several additional instructions.

All examples are given in COPS™ Cross Assembler language, using COP400 assembler instruction mnemonics and operand statements. Although, in the following examples, instruction operands and ROM page numbers are written using decimal notation, the programmer may specify these expressions in hexadecimal notation — the assembler accepts either format (e.g., AISC 13 = AISC X'C, Page X'A = Page 10). On occasion, source code examples contain non-instruction statements, such as assembler directives which convey information to the assembler necessary for proper program address allocation and similar assembler related tasks. For further information on the COPS Cross Assembler and its use see *PDS User's Manual,* Chapter 8.

## 4.1 Program Memory Allocation

Generally, COP420-series program memory may be thought of as one area of 1024 bytes of ROM with an address range of 0 to 3FF (hexadecimal). However, while this concept is convenient in writing, assembling and debugging major portions of COP420-series programs, it is necessary, with respect to a few instructions, to conceptualize program memory on a 64-word "page" basis.

Specifically, because of the characteristics and restrictions associated with the JP, JSRP, JID, and LQID instructions, the programmer must conceive of program memory as 1024 bytes or words, organized as sixteen pages, numbered 0-15 respectively. The following discussion provides information and examples relating to the "page" characteristics of each of these unique instructions. For information on the machine code and operations performed by these instructions, see Section 3.2. Table 4.1 provides a conversion chart indicating the hexadecimal address equivalents for each of the 16 "pages" of ROM. Note — each page consists of 0 through $3F_{16}$ words.

**Table 4.1 Page to Hexadecimal Address Table**

| Page | Hexadecimal Address Range |
|---|---|
| 0 | 000-03F |
| 1 | 040-07F |
| 2 | 080-0BF |
| 3 | 0C0-0FF |
| 4 | 100-13F |
| 5 | 140-17F |
| 6 | 180-1BF |
| 7 | 1C0-1FF |
| 8 | 200-23F |
| 9 | 240-27F |
| 10 | 280-2BF |
| 11 | 2C0-2FF |
| 12 | 300-33F |
| 13 | 340-37F |
| 14 | 380-3BF |
| 15 | 3C0-3FF |

### JP Instruction

The JP instruction is used to transfer program control to a ROM location within a page or within a two-page boundary consisting of "subroutine pages" 2 or 3.

The following page restrictions apply to the JP instruction:

- When used in any page other than page 2 or 3, it can only jump to a word within the *current* page.
- When used in page 2 or 3, it may jump to a word within page 2 *or* 3.
- In all cases, it cannot jump to the last word of a page (word $03F_{16}$).

The JP instruction assembly operand normally consists of a program label or expression specifying the address of the word to be jumped to. To specify page boundaries and to ensure correct placement of the JP and other page-oriented

instructions, the assembler .PAGE directive is used to specify the beginning of new page boundaries for program code placement. (See *PDS User's Manual*, Chapter 8.) The following are examples of use of the JP instruction when used outside subroutine pages 2 and 3:

```
        .PAGE 0             ; PLACE FOLLOWING CODE IN
                            ; PAGE 0
LABEL1: .
        JP      LABEL2      ; LEGAL JUMP WITHIN PAGE
        .
        .
LABEL2: .
        JP      LABEL3      ; ILLEGAL JUMP TO LAST
        .                   ; WORD OF PAGE
        JP      LABEL4      ; ILLEGAL JUMP TO ANOTHER
        .                   ; PAGE
        .
LABEL3: ; THIS INSTRUCTION IN LAST
        ; WORD OF PAGE 0
        ; PLACE FOLLOWING CODE
        ; ON PAGE 1*
        .
        PAGE 1
        .
        .
        .
LABEL4: .
        .
        .
        .
```

*Note: The .PAGE 1 directive is not necessary — the PDS Assembler automatically places code in successive memory locations. After a particular page is full, code is automatically placed in successive locations on the following page.

The following examples illustrate use of the JP instruction when in subroutine pages 2 and 3:

```
        .PAGE 2             ; START OF "SUBROUTINE"
LABEL1: .                   ; PAGE 2 CODE
        JP      LABEL3      ; LEGAL JUMP TO PAGE 3
        .                   ; LOCATION
        JP      LABEL2      ; ILLEGAL JUMP TO LAST
        .                   ; WORD OF PAGE
        .
LABEL2: .                   ; LAST WORD OF PAGE 2
        .PAGE 3             ; START OF PAGE 3 CODE
        .
        .
        JP      LABEL4      ; ILLEGAL JUMP TO PAGE
        .                   ; OUTSIDE PAGE 2 OR 3
        .
LABEL3: .
        JP      LABEL1      ; LEGAL JUMP TO PAGE 2
        .                   ; LOCATION
        JP      LABEL3      ; LEGAL JUMP WITHIN PAGE
        .
        .
        .PAGE 4             ; START OF PAGE 4 CODE
        .
        .
LABEL4: .
        .
        .
        JP      LABEL1      ; ILLEGAL JUMP TO PAGE 2
        .                   ; (MAY ONLY BE DONE WHEN
        .                   ; IN PAGE 2 OR 3)
```

### JSRP Instruction

The JSRP instruction is another page-oriented instruction which transfers program control to a word located within "subroutine" page 2 *only*. Its primary purpose is to allow a single-byte jump to a subroutine in page 2 from any program location other than from page 2 or 3. As explained in Section 3.2, JSRP pushes the subroutine-save stack to allow a return to the next program instruction following the subroutine call. The restrictions with the JSRP instruction are as follows:

- JSRP cannot be used to jump to a subroutine *when in* pages 2 or 3. (The double-byte JSR instruction can be used for this purpose.)
- JSRP cannot be used to jump to a subroutine located at the last word of page 2. (A JSR can also be used for this purpose.)

Examples of use of the JSRP instruction:

```
        .PAGE 0
        .
        .
LABEL1: .                   ; PAGE 0 SUBROUTINE
        .
        RET                 ; RETURN FROM SUBROUTINE
        .
        JSRP    ADD         ; LEGAL CALL TO PAGE 2
        .
        JSRP    SUB         ; ILLEGAL CALL TO PAGE 3
        .
        .PAGE 2             ; START OF PAGE 2 CODE
        .
        .
ADD:    .                   ; START OF ADD SUBROUTINE
        .
        RET
        .
        JSRP    LABEL1      ; ILLEGAL CALL FROM PAGE 2
        .
        .PAGE 3             ; START OF PAGE 3 CODE
        .
        .
SUB:    .                   ; SUBTRACT SUBROUTINE
        .
        RET
        .
        .
```

### Subroutine Pages 2 and 3

The special characteristics of the JP and JSRP instructions facilitate the use of pages 2 and 3 as subroutine pages. Programmers should consider dedicating these pages to the recursive program subroutine for the following reasons:

- A single-byte JSRP can be used to transfer program control to a page 2 subroutine.
- When in pages 2 or 3, a single-byte JP can be used to jump to either of these pages.

The following code exemplifies the use of the JP and JSRP instructions to transfer program control to and within pages 2 and 3 as follows. Note that in this example the ADD subroutine jumps to MEMOVE (Memory Move) routine before returning.

Thus, subroutines may share a common "return" subroutine, jumped to from page 2 or 3 with a single-byte JP instruction.

```
        .PAGE 0
        .
        .
        JSRP    ADD         ; CALL ADD SUBROUTINE
        .
        .
        .PAGE 2             ; START OF PAGE 2 CODE
        .
        .
ADD:    .                   ; ADD SUBROUTINE
        .
        .
        JP      MEMOVE      ; JUMP TO MEMOVE
        .                   ; "RETURN" ROUTINE (NO
        .                   ; "PUSH" OF STACK)
        .PAGE 3             ; START OF PAGE 3 CODE
        .
        .
MEMOVE: .                   ; MEMORY MOVE ROUTINE
        .
        .
        RET                 ; RETURN TO MAIN PROGRAM
        .                   ; (POP STACK)
```

**JID Instruction**

The JID (Jump Indirect) instruction is another page-oriented instruction. For a machine operation description, see Section 3.2. JID is an *indirect* ROM addressing instruction which transfers program control to a new ROM location based upon the contents of a ROM "pointer." The paging features and restrictions associated with the JID instruction are as follows:

- JID first jumps to a *ROM pointer* based upon the contents of A and RAM.
- JID then transfers program control to the ROM word specified by the *contents* of the ROM pointer.
- The ROM pointer *and* the indirect address jumped to must be within the same 4-page ROM "block" as the JID instruction. Specifically, for purposes of this instruction, the sixteen pages of ROM are divided into 4 blocks as follows:

| Block | Pages |
|---|---|
| 1 | 0-3 |
| 2 | 4-7 |
| 3 | 8-11 |
| 4 | 12-15 |

For example, if the JID instruction is located in page 5, the ROM pointer and the indirect address to which program control is transferred must be within block 2 (pages 4-7). For an example of the use of the JID instruction in a simple keyboard decode routine, see Section 5.3.

**LQID Instruction**

The LQID instruction is an *indirect* data output instruction. It loads the 8-bit Q register with the 8-bit contents of a particular ROM location pointed to by A and RAM. For an explanation of the machine operations associated with this instruction, see Section 3.2. The paging restrictions associated with this instruction are similar to those associated with the JID instruction, as follows:

- For purposes of the LQID instruction as with the JID instruction, ROM is divided into 4-page ROM "blocks" (pages 0-3, 4-7, 8-11 and 12-15).
- The ROM location containing the LQID "lookup" data must be within the same ROM block as the LQID instruction.

For example, a LQID instruction located in page 9 must access ROM data located in pages 8 through 11.

**Additional Restrictions Associated with JP, JSRP, JID and LQID Instructions**

As already mentioned, the ROM address register (P) increments its value when executing an instruction to point to the next memory instruction, automatically "rolling over" to the next page after executing an instruction located in the last word of a page. It is important to realize, however, that P is incremented *prior to the execution of the current instruction.* This characteristic has important consequences for JP, JSR, JID and LQID instructions *which are located in the last word of a page.* Specifically, these instructions will operate on the incremented value of P which, because of the increment-before-execution COP feature, will point to the first word of the next page. Consequently, if any of these instructions are placed in the last word of a page, the program will treat them as residing on the first word of the following page. Given the paging restrictions associated with these instructions, the following operations and restrictions are associated with the following placements of these instructions:

- A JP in the last word of a page will go to any location in the following page (except the last word). A JP in the last word of page 1 will be able to go to any location (except the last word) of page 2 or 3 since it is treated as a JP in page 2. Furthermore, a JP in the last word of page 3 *will not go* to a location within page 2 or 3, but, instead, will go to a location within page 4.
- A JSRP instruction is not allowed to reside in the last word of page 1, since it will be treated as an illegal use of JSRP in page 2. A JSRP in the last word of page 3, however, is allowed, since it will be treated as a JSRP outside of pages 2 or 3, namely in page 4.
- A LQID or JID instruction located in the last word of the last page of a particular ROM block (last word of page 3, 7, 11 or 15) will lookup data or transfer program control, respectively, to a location within the *next* 4 page ROM block.

As is evident from the above, these characteristics are not necessarily restrictions, provided the programmer intentionally uses these instructions to operate in the above manner. For example, a JP on the last word of page 1, unlike other page 1 JP instructions, will be able to transfer program control to the two-page subroutine pages 2 or 3, provided the operand specifies a location within page 2 or 3. Similarly, a LQID or JID located in the last word of the last page of a ROM block will allow data lookups on or indirect program control transfers to locations within the next ROM block, provided the lookup data or address pointers are placed in the appropriate locations within the *next* ROM block.

### Use of Assembler .PAGE Directive

Because of the above paging restrictions, programmers are advised to place .PAGE assembler directives at the beginning of each page of code. Although portions of the program may not contain page-related instructions, this practice will facilitate placement of program "patches" or other modifications required during the program debug phase, these often involving page-related instructions. This practice is also a convenient, if not necessary, documentation tool, dividing the assembler output listing into a COPS™ page format. Finally, since the COPS Cross Assembler places program memory words into successive locations without regard to COPS pages, the use of a .PAGE directive is a simple means of reserving program memory space at the end of a page during initial program code generation, often used later for program additions. An alternative means of reserving program memory space anywhere within a page is by use of an assembler *assignment* statement which references the assembler location pointer — the pointer is referenced by a period ("."). For more information on the assignment statement, see *PDS User's Manual,* Section 8.4. An example and explanation of its use in referencing the assembler location counter (".") is contained in Section 4.5 of this manual.

## 4.2 Data Memory Allocation and Manipulation

An important step which should occur prior to writing a COPS™ program is the allocation of program data (registers, flags, counters, etc.) to specific areas of program memory (RAM). This process is referred to as "creating a RAM map" and, although the map will undoubtedly change as programming continues, construction of an initial RAM map will make the ensuing programming process significantly easier.

As explained in Section 2.8, the COP420-series has 4 data memory registers, numbered 0 through 3, consisting of 16 4-bit digits. Frequently accessed data should be stored in locations which are able to be pointed to by loading the B register with a single-byte LBI instruction. These locations consist of digit numbers 0 and 9 through 15 in any data memory register. These areas are indicated by the diagonal-lined areas of Figure 4.1. It requires a double-byte LBI instruction to load the B register to access the other digits in data memory registers, thus requiring an extra program memory word. Single-bit flags and digit counters should be located in these diagonal-lined regions since they tend to be frequently accessed in most programs.

The memory reference instructions LD, X, XDS, and XIS allow the programmer to modify the data memory *register* address without using an LBI instruction. All of these instructions may modify the upper two bits of B (Br — RAM register-select) by specifying an "r" operand field which is exclusive-ORed with the current value of Br. This feature allows the programmer to toggle back and forth between any of the four COP420 data memory registers. For example, data located within the data memory locations marked with shaded boxes in Figure 4.1 can be easily swapped back and forth using the LD and X instructions. They can also be added to or subtracted from each other easily.



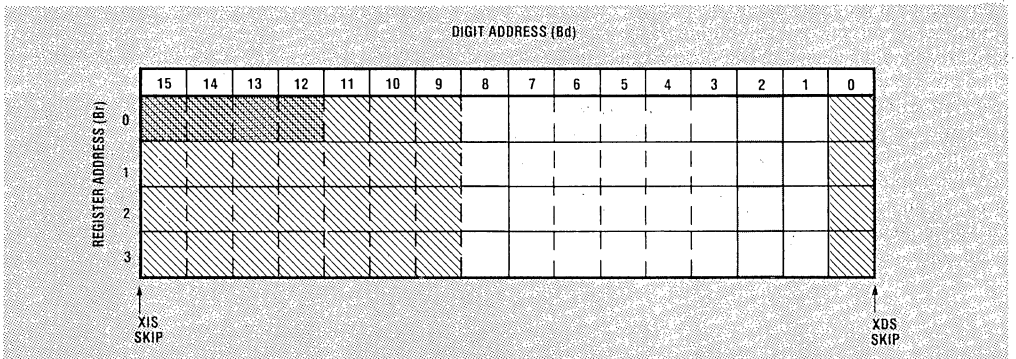



Figure 4.1 COP420 Data Memory Map

The automatic data memory *digit* address increment and decrement features associated with the XIS and XDS instructions and their skip condition features facilitate the shifting, adding, and subtracting of the contents of data memory. Data that needs to be shifted should be located in adjacent digit locations (for example, the dotted-box locations in Figure 4.1). Data that needs to be added, subtracted, or shifted should be located in areas adjacent to the XIS or XDS skip boundaries. The dotted locations in Figure 4.1 are against the XIS boundary at digit 15. This allows the programmer to take advantage of the skip feature of the XIS instruction.

The following examples illustrate several of the principles discussed above. The notation $M(N_1,N_2)$ indicates a particular data memory digit M, where $N_1$ = register number and $N_2$ = digit number.

```
; MOVE M(3,0) TO M(1,0)

        LBI     3,0     ; 3 TO BR; 0 TO BD (SINGLE-BYTE
                        ;  LBI: D = 0)
        LD      2       ; M(3,0) TO A; 1 TO BR 3 ⊕ 2 = 1)
        X               ; A TO M(1,0)

; MOVE MEMORY REGISTER 1 TO MEMORY REGISTER 0
; M(1,15) - M(1,0) TO M(0,15) - M(0,0)

        LBI     1,15    ; 1 TO BR, 15 TO BD (SINGLE-BYTE
                        ;  LBI)
MV1:    LD      1       ; M(1,15) TO A; 0 TO BR
        XDS     1       ; A TO M(0,15); 1 TO BR; BD - 1 TO
                        ;  BD; CONTINUE TO MOVE NEXT
                        ;  LOWER DIGIT UNTIL BD GOES
                        ;  PAST 0 AND SKIPS
        JP      MV1     ; HERE IF NO SKIP

; LEFT SHIFT DOTTED AREAS OF FIGURE 4.1
; 0 TO M(0,12) → M(0,12) → M(0,13) → M(0,14) → M(0,15) TO A

        CLRA            ; 0 TO A
        LBI     0,12    ; 0 TO BR; 12 TO BD
LSHFT   XIS             ; M(0,12) TO A; 0 TO M(0,12)
        JP      LSHFT   ; EXCHANGE A INTO BD, LEFT
                        ;  SHIFT NEXT HIGHER DIGIT UNTIL
                        ;  "BD" GOES PAST 15 AND SKIPS
```

### 4.3 Subroutine Techniques

Any section of program code used repeatedly within the main program should be coded as a subroutine, preferably on "subroutine pages" 2 or 3 for the reasons discussed above. Subroutines are jumped to or "called" by the JSRP or JSR (double-byte) instruction, both of which "push" the stack, saving the next memory location address after the subroutine call in the SA subroutine-save *register.* The other subroutine-save registers are correspondingly pushed. Subroutine nesting on the COP420-series is permitted to 3 levels, since this device contains 3 subroutine-save registers.

Subroutines should terminate with a RET or RETSK instruction, both of which "pop" the subroutine stack, with the program return address in SA being placed in the program counter register. The other subroutine-save registers are also popped. The contents of SC, which is the bottom-most subroutine-save register, are retained in SC in addition to being placed in SB.

It is convenient to think of a subroutine as a program module. The programmer should make its interface to the calling program as clearly defined and as simple as possible. *The interface* (including data memory registers, entry points, etc., used by the subroutine) *should be documented fully by comments to the code.*

Subroutine examples presented in this chapter often use the double-byte JSR instruction to call subroutines since no restrictions are associated directly with its use. When writing an actual program, programmers should use the more efficient single-byte JSRP instruction as well as use the double-page boundaries of subroutine pages 2 and 3 for placement of subroutine code (as discussed above) for efficient single-byte jumps while in these pages using the JP instruction.

It is often useful to define multiple-entry points for a single subroutine. The successive-skip feature of the LBI instruction often facilitates this technique. For example, see Register Move Routines, Section 4.4.

The RETSK instruction allows the programmer to use an alternate return to the main program (skipping the first program instruction encountered upon return) based upon tests or computations made within the subroutine itself. Example:

```
        .PAGE 0
        .
        .
        .
        JSRP    ADD     ; CALL ADD SUBROUTINE
        .               ; RETURN HERE IF RESULT ≤ 9
        .               ; RETURN HERE IF RESULT > 9
        .
        .PAGE 2         ; START PAGE 2 CODE
        .
        .
        .
ADD:    ADD             ; ADD SUBROUTINE — ADDS TWO
        .               ;  BCD DIGITS; RESULT TO A
        .
        AISC    7       ; OVERFLOW AND SKIP IF RESULT
        .               ;  > 9
        RET             ; RETURN WITHOUT SKIP (RESULT
        .               ;  ≤ 9)
        RETSK           ; RETURN THEN SKIP (RESULT > 9)
```

## 4.4 Utility Routines

Programmers often build a library of basic routines which are useful in numerous applications. This and the following sections provide examples of several such "utility" routines.

### Register Move Routine

It is often necessary to move data from one memory register to another. The following are examples of this type of routine. Note that the routines may be easily modified to perform moves in the opposite direction (e.g., from register 1 to 0) or to include a move of register 1 to 2.

ADJACENT MEMORY MOVE ROUTINE

```
; ADJACENT MEMORY REGISTER MOVE, MULTIPLE ENTRY POINT SUBROUTINE
; MOV0T1: MOVE MEMORY REGISTER 0 TO REGISTER 1 ENTRY POINT
; MOV2T3: MOVE MEMORY REGISTER 2 TO REGISTER 3 ENTRY POINT
; ROUTINE MOVES DIGITS 15 THROUGH 0
; PREVIOUS CONTENTS OF A AND B ARE LOST

MOV0T1:   LBI    0,15    ; POINT TO M(0,15)
MOV2T3:   LBI    2,15    ; NOTE LBI SUCCESSIVE SKIP FEATURE
MOV:      LD     1       ; TRANSFER M TO A; EXCLUSIVE-OR 1 WITH BR
          XDS    1       ; EXCHANGE A WITH M; EXCLUSIVE-OR 1 WITH BR; DECREMENT BD
          JP     MOV     ; JUMP TO "MOV" IF MORE DIGITS TO MOVE
          RET            ; RETURN WHEN XDS SKIPS (LAST DIGIT MOVED)
```

DATA MEMORY SHIFT AND ROTATE ROUTINES

```
; MULTIPLE ENTRY POINT SUBROUTINE TO RIGHT SHIFT MEMORY REGISTER 0, 1, 2, OR 3 ONE DIGIT POSITION
; ZEROS ARE SHIFTED INTO DIGIT 15
; PREVIOUS CONTENTS OF A AND B ARE LOST
; RSH0: RIGHT SHIFT REGISTER 0 ENTRY POINT
; RSH1: RIGHT SHIFT REGISTER 1 ENTRY POINT
; RSH2: RIGHT SHIFT REGISTER 2 ENTRY POINT
; RSH3: RIGHT SHIFT REGISTER 3 ENTRY POINT

RSH0:     LBI    0,15    ; POINT TO DIGIT 15 IN APPROPRIATE REGISTER
RSH1:     LBI    1,15    ; NOTE LBI SUCCESSIVE SKIP FEATURE
RSH2:     LBI    2,15
RSH3:     LBI    3,15
          CLRA           ; ZEROS IN FIRST DIGIT (DIGIT 15)
SHFTR:    XDS            ; SHIFT RIGHT*
          JP     SHFTR   ; CONTINUE UNTIL ENTIRE REGISTER SHIFTED
          RET            ; RETURN WHEN FINISHED ("XDS" SKIPS)
```

*NOTE THAT THE ABOVE ROUTINE CAN SHIFT THE REGISTERS ONE DIGIT TO THE *LEFT* USING THE "XIS" INSTRUCTION IN PLACE OF "XDS" AND STARTING AT DIGIT 0.

```
; MULTIPLE ENTRY POINT SUBROUTINE TO LEFT SHIFT THE BITS OF A MEMORY DIGIT
; UPON ENTRY, B MUST POINT TO THE DIGIT TO BE SHIFTED
; ZEROS ARE SHIFTED IN FROM THE RIGHT
; PREVIOUS CONTENTS OF A ARE LOST
; LEF1: SHIFT DIGIT LEFT 1 BIT ENTRY POINT
; LEF2: SHIFT DIGIT LEFT 2 BITS ENTRY POINT
; LEF3: SHIFT DIGIT LEFT 3 BITS ENTRY POINT

LEF3:     LD             ; DIGIT TO A
          ADD            ; ADD DIGIT TO ITSELF
          X              ; SHIFTED DIGIT TO MEMORY
LEF2:     LD
          ADD
          X
LEF1:     LD
          ADD
          X
          RET
```

```
; MULTIPLE ENTRY POINT SUBROUTINE TO LEFT ROTATE THE BITS OF A MEMORY DIGIT
; UPON ENTRY, B MUST POINT TO THE DIGIT TO BE ROTATED
; PREVIOUS CONTENTS OF A ARE LOST
; LR01: ROTATE DIGIT LEFT 1 BIT ENTRY POINT
; LR02: ROTATE DIGIT LEFT 2 BITS ENTRY POINT
; LR03: ROTATE DIGIT LEFT 3 BITS ENTRY POINT (SAME AS RIGHT ROTATE 1)

L0R3:   JSR     LR01    ; ROTATE 1, THEN 2 MORE
L0R2:   JSR     LR01
L0R1:   LD              ; DIGIT TO A
        ADD             ; ADD DIGIT TO ITSELF
        X               ; EXCHANGE M WITH A
        AISC    8       ; WAS MEMORY BIT3 ON?
        RET             ; NO, RETURN
        SMB     0       ; YES, WRAP AROUND BIT0
        RET
```

ACCUMULATOR SHIFT ROUTINE:

```
; SUBROUTINE TO LEFT SHIFT BITS OF A BY USING THE SIO REGISTER (SIO MUST BE ENABLED AS A SERIAL SHIFT REGISTER)
; SI MUST BE CONNECTED TO LOGIC "0" (GROUND)
; ZEROS ARE SHIFTED IN FROM THE RIGHT
; LFTA1: LEFT SHIFT A 1 BIT ENTRY POINT
; LFTA2: LEFT SHIFT A 2 BITS ENTRY POINT
; LFTA3: LEFT SHIFT A 3 BITS ENTRY POINT

LFTA1:  XAS             ; A TO SIO
LFT2:   XAS             ; SIO TO A (SIO SHIFT RIGHT 1 BIT)
        RET
LFTA2:  XAS             ; A TO SIO
LFT3:   JP      LFT2    ; DELAY 1 INSTRUCTION CYCLE TIME — SIO SHIFT RIGHT 1 MORE BIT
LFTA3:  XAS             ; A TO SIO
        JP      LFT3    ; DELAY 1 INSTRUCTION CYCLE TIME — SI SHIFT RIGHT 2 MORE BITS
```

CLEAR DATA MEMORY ROUTINE:

```
; SUBROUTINE TO CLEAR ALL RAM
; CLEAR REGISTERS 3 THROUGH 0 IN SUCCESSION, THEN RETURN

CLRAM:  LBI     3,15    ; START BY CLEARING REGISTER 3
CLR:    CLRA            ; 0 TO A
        XDS             ; EXCHANGE WITH DIGIT 15, DECREMENT DIGIT
        JP      CLR     ; CONTINUE UNTIL DIGIT 0 CLEARED
        XABR            ; BR TO A
        AISC    15      ; REGISTER 0 CLEARED?
        RET             ; YES, RETURN
        XABR            ; NO, REPLACE BR – 1 INTO BR
        JP      CLR     ; CLEAR NEXT REGISTER
```

## 4.5 Timing Considerations

Programmers must often synchronize programs with external events ("real-time" programming). Such programs must be balanced with respect to the execution times of the various branches taken by the program. To ensure equal execution times, program timing delays are added. There are numerous ways of introducing timing delays, the simplest but least efficient involving the use of NOPs. Obviously these are appropriate for only the shortest delays.

A counting loop, such as:

```
        CLRA
        AISC    1
        JP      .–1     ; ADD 1 TO A UNTIL A
CONTINUE: .             ;  OVERFLOWS*
          .
          .
```

is more efficient for longer delays, but destroys the previous contents of A. Another method is to use a "scratch-pad" counter in data memory using the XAD instruction. For example, assuming the use of a counter in M(3,15):

```
        XAD     3,15    ; COUNTER TO A; A TO M(3,15)
        AISC    1       ; ADD 1 TO COUNTER UNTIL IT
        JP      .–1     ;  OVERFLOWS*
        XAD             ; RESTORE A THEN CONTINUE
        .
        .
```

*Note: The above timing code example shows the use of a special assembler symbol in the operand of the JP instruction. Namely, the operand of the JP instruction, rather than using a program label, references the

assembler location counter (which equals the address of the current program address). The "." signifies the assembler location counter and the value of the operand equals the location counter minus the number of memory bytes to the right of the "." sign. Use of the "." location pointer symbol for transfer of control instructions facilitates coding in avoiding the need to create unique program labels to reference memory addresses.

Larger delays may be implemented by using multi-digit RAM counters. Another technique is calling unrelated subroutines which change registers or memory locations not currently in use or whose net effect on memory is null. An example of the latter technique is illustrated below.

```
JSR     LR03    ; LEFT ROTATE 3 BITS
JSR     LR01    ; LEFT ROTATE 1 MORE BIT
```

This combination of subroutines only affects A, while maintaining the integrity of data in the rotated memory digit.

## 4.6 BCD Arithmetic Routines

BCD data manipulation routines are essential in applications which interface with human operators of a microcomputer system. They are easily translated to and from codes used by decimal displays and keyboards. The COP400 series instruction set and internal architecture has been designed to perform BCD routines efficiently. The following routines are examples of simple BCD data manipulation routines.

### Unsigned BCD Integer Add and Subtract Routines

The following programs present unsigned BCD integer add and subtract subroutines. Data is stored in data memory registers 0 and 1 and is 13 digits long, occupying memory digits 0 through 12, respectively. The most significant BCD digit is in memory digit 12. The techniques used to manipulate the contents of memory address register B are common to many arithmetic routines. The LD and XIS instructions transfer data between memory and A. After the transfer they modify B. LD 1 causes a "1" to be exclusive-ORed with Br. Since, in these routines, Br is always equal to 1 when the LD 1 instruction operates upon it, Br is always changed to 0. (LD 1 causes Br to point to memory register 0.) Similarly, XIS 1 also changes Br to point to memory register 0, *as well as* incrementing the value of Bd to point to the next higher memory *digit*. Thus, Br "flip-flops" between registers 1 and 0 while Bd "walks-up" the digits of the registers.

```
; SUBROUTINE TO DO UNSIGNED BCD INTEGER ADD OF R1 AND R0, RESULT TO R0
; EACH INTEGER OCCUPIES MEMORY DIGITS 0 (LOW ORDER) THROUGH 12 (HIGH ORDER)
; ON RETURN, C = 1 INDICATES OVERFLOW
; PREVIOUS CONTENTS OF A AND B ARE LOST
; ENTRY POINT: BCDADD

BCDADD:  LBI     1,0     ; POINT TO LOW ORDER DIGIT, REGISTER 1
         RC              ; INITIALIZE C TO "0" (NO CARRY)
ADDL:    LD      1       ; MOVE R1 DIGIT TO A, POINT TO SAME DIGIT IN R0
         AISC    6       ; ADD BCD CORRECTION FACTOR OF 6 TO A
         ASC             ; ADD R0 DIGIT TO R1 DIGIT
         ADT             ; RESTORE BCD VALUE IF BCD CORRECTION NOT NECESSARY
         XIS     1       ; MOVE SUM DIGIT TO R0: POINT TO R1, NEXT HIGHER DIGIT
         CBA             ; BD TO A
         AISC    3       ; LAST DIGITS ADDED?
         JP      ADDL    ; NO, ADD NEXT HIGHER DIGITS
         RET             ; YES, RETURN

; SUBROUTINE TO DO UNSIGNED BCD INTEGER SUBTRACT
; MINUEND IS IN R0, SUBTRAHEND IS IN R1
; DIFFERENCE IS PLACED IN R0
; MINUEND, SUBTRAHEND AND DIFFERENCE DIGITS EACH OCCUPY MEMORY DIGITS 0 (LOW ORDER) THROUGH 12 (HIGH ORDER)
; ON RETURN: C = 1 INDICATES NO BORROW, C = 0 INDICATES BORROW
; PREVIOUS CONTENTS OF A AND B ARE LOST
; ENTRY POINT: BCDSUB

BCDSUB:  LBI     1,0     ; POINT TO LOW ORDER DIGIT IN R1
         SC              ; INITIALIZE C TO "1" (NO BORROW)
SUB:     LD      1       ; LOAD R1 DIGIT TO A, POINT TO SAME DIGIT IN R0
         CASC            ; SUBTRACT R1 DIGIT FROM R0 DIGIT
         ADT             ; BCD ADJUST IF BORROW (C = 0)
         XIS     1       ; PLACE DIFFERENCE DIGIT IN R0, POINT TO NEXT HIGHER DIGIT IN R1
         CBA             ; BD TO A
         AISC    3       ; HIGH ORDER DIGITS (12) SUBTRACTED?
         JP      SUB     ; NO, SUBTRACT NEXT HIGHER DIGITS
         RET             ; YES, RETURN
```

### BCD Integer Multiply Routine

This routine will multiply the contents of data memory register 2 with register 1, placing the result in register 2 (digits 0-12). It also calls the BCD add routine ("BCDADD") given above. Note that a loop-counter is contained in M(0,13) which causes the program to return after all 12 digits have been multiplied. Also note the alternate-return feature of page 3 subroutine TMZERO (Test Memory Digit = 0). A flowchart for the routine is given in Figure 4.2.

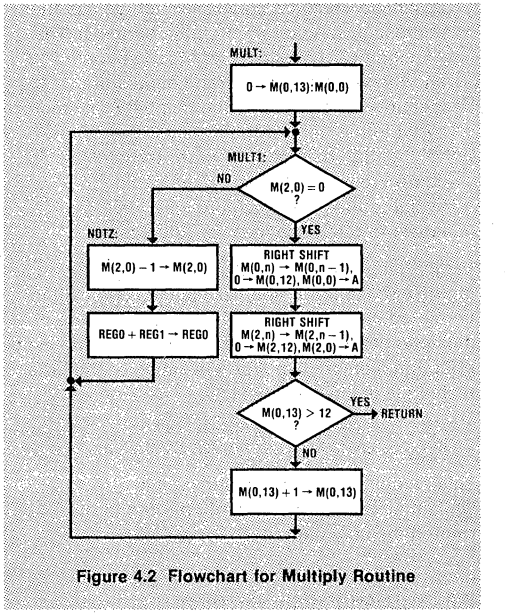



Figure 4.2 Flowchart for Multiply Routine

```
; TWO-LEVEL BCD INTEGER MULTIPLY SUBROUTINE
; 12 DIGIT BCD INTEGER CONTAINED IN REGISTER 1, DIGITS 0 - 12 (LOW ORDER TO HIGH ORDER) MULTIPLIED BY 12 DIGIT BCD
;  INTEGER CONTAINED IN REGISTER 2, DIGITS 0 - 12 (LOW ORDER TO HIGH ORDER), RESULT TO REGISTER 2
; MULTIPLICATION OF DIGITS PERFORMED BY MULTIPLE ADDITIONS OF REGISTER 1 ACCORDING TO VALUE OF REGISTER 2
;  DIGITS
; DIGIT ADDITION RESULTS TEMPORARILY STORED IN R0 AND CONSECUTIVELY RIGHT SHIFTED INTO RESULT REGISTER 2, HIGH
;  ORDER DIGIT
; ENTRY POINT: MULT
; SUBROUTINES CALLED: RSHR0, RSHR2, CLR, DEC 1, INC 1, TMZERO, BCDADD

MULT:   LBI     0,13        ; POINT TO M(0,13)
        JSR     CLR         ; CLEAR REGISTER 0, DIGITS 13 - 0
MULT1:  LBI     2,0         ; POINT TO M(2,0)
        JSR     TMZERO      ; IS M(2,0) = 0?
        JP      NOTZ        ; NO, JUMP TO NOTZ
        JSR     RSHR0       ; YES, RIGHT SHIFT REGISTER 0, DIGITS 12 - 0
        JSR     RSHR2       ; RIGHT SHIFT REGISTER 2, DIGITS 12 - 0
        LBI     0,13        ; POINT TO LOOP COUNTER
        LD                  ; LOOP COUNTER TO A
        AISC    3           ; IS COUNTER > 12
        JP      .+2         ; NO, CONTINUE
        RET                 ; YES, ALL DIGITS MULTIPLIED, RETURN
        JSR     INC1        ; CONTINUE, INCREMENT LOOP COUNTER DIGIT
        JP      MULT1       ; MULTIPLY NEXT HIGHER ORDER DIGITS
NOTZ:   JSR     DEC1        ; DECREMENT M(2,0)
        JSR     BCDADD      ; ADD R0, DIGITS 0 - 12, TO R1, DIGITS 0 - 12, RESULT TO R0
        JP      MULT1       ; JUMP BACK TO MULT 1

; MULTIPLE ENTRY POINT SUBROUTINE TO RIGHT SHIFT DIGITS 12 - 0 OF REGISTER 0 OR 2
; ON RETURN A CONTAINS LOW ORDER REGISTER DIGIT
; RSHR0: RIGHT SHIFT DIGITS OF REGISTER 0 ENTRY POINT
; RSHR2: RIGHT SHIFT DIGITS OF REGISTER 2 ENTRY POINT

RSHR0:  LBI     0,12        ; POINT TO HIGH ORDER DIGIT, REGISTER 0
RSHR2:  LBI     2,12        ; POINT TO HIGH ORDER DIGIT, REGISTER 2
RSH:    XDS                 ; SHIFT RIGHT DIGITS 12 - 0 IN REGISTER
        JP      RSH
        RET
```

```
; SUBROUTINE TO CLEAR ALL DIGITS TO THE RIGHT AND INCLUSIVE OF A HIGH-ORDER DIGIT OF A REGISTER
; ON ENTRY, B MUST POINT TO THE REGISTER AND HIGH ORDER DIGIT NUMBER

CLR:    CLRA
        XDS                     ; CLEAR REGISTER, STARTING WITH HIGH ORDER DIGIT
        JP          CLR
        RET                     ; RETURN WHEN DIGIT 0 CLEARED

; MULTIPLE ENTRY SUBROUTINE TO EITHER DECREMENT OR INCREMENT BY 1 THE VALUE OF A MEMORY DIGIT
; ON ENTRY, B MUST POINT TO THE DIGIT TO BE OPERATED UPON
; DEC1: ENTRY POINT TO DECREMENT A DIGIT
; INC1: ENTRY POINT TO INCREMENT A DIGIT

DEC1:   CLRA
        COMP                    ; 15 TO A
ADEX:   ADD                     ; ADD MEMORY DIGIT TO A
        X                       ; EXCHANGE BACK TO MEMORY
        RET                     ; RETURN
INC1:   CLRA
        AISC        1           ; 1 TO A
        JP          ADEX        ; ADD AND EXCHANGE WITH MEMORY DIGIT

; SUBROUTINE TO TEST MEMORY DIGIT EQUAL TO ZERO
; ON ENTRY, B MUST POINT TO MEMORY DIGIT TO BE TESTED
; ON RETURN, SKIP FIRST INSTRUCTION IF MEMORY DIGIT EQUAL TO ZERO
; NORMAL RETURN IF MEMORY DIGIT NOT EQUAL TO ZERO

TMZERO: CLRA                    ; 0 TO A
        SKE                     ; DIGIT = ZERO?
        RET                     ; NO, NORMAL RETURN
        RETSK                   ; YES, RETURN THEN SKIP
```

## 4.7 Simple Display Loop Routine

The following routine is a simple LED display loop routine. It illustrates the use of LEI and LQID instructions, both designed to facilitate the outputting of segment data to a multiplexed display. As explained in Section 3.2, LEI Instruction description, setting bit 2 of the EN register enables Q latch (segment) data to the L I/O ports; resetting $EN_2$ disables the L I/O ports, providing segment blanking for the LED display. $EN_2$ is set and reset, respectively, by the LEI 4 and LEI 0 instructions.

As explained in Sections 3.2 and 4.1, LQID loads the 8-bit Q register with the contents of a ROM location pointed to by A and M (ROM "lookup" data must be within the same 4-page ROM block as the LQID instruction). In this example, since A is always equal to 0 at the time of the LQID instruction, the ROM data accessed by this instruction must be within the first 16 words of the first page of the ROM block in which the LQID instruction is located as pointed to by the 4-bit contents of M ($P_9$ and $P_8$ remain the same, $P_7$-$P_4$ equal "0"). For example, if, as is the case for the following routine, LQID is in page 5, it will lookup data within one of the first 16 locations of page 4. The value of the contents of the memory digit pointed to by the B register at the time of the LQID instruction determines which one of the 16 words is accessed (e.g., if M = 2, word 2 is loaded into Q).

Due to these considerations, page 4, words 0-9 should equal the 8-bit, seven-segment decode lookup data for the BCD digits 0-9 respectively. (In this example the low-order bit — decimal point — of each lookup data word is reset, signifying that the decimal point is off.) ROM seven-segment decode lookup data is placed in ROM memory locations by the Assembler WORD directive. (See *PDS User's Manual*, Section 8.4.)

Another feature of this routine is the dual function of Bd. Its value may be output directly to the D outputs to select one of 16 digits of the multiplexed display (assuming the D outputs are connected to a 1-of-16 decoder/driver device). Also, its value is used to select one of 16 RAM digits whose contents are used by the LQID instruction to access the segment data to be output to the selected digit. To facilitate coding (by avoiding the need to change the value of Bd after its contents are output to D to select or display digit), RAM digit locations should correspond to the digit of the display. In other words, RAM digits 0-15 should contain, respectively, the LQID pointers to segment data for display digits 0-15. This technique, used below, allows Bd to first enable the appropriate display digit and then, without its value being changed, to point to the RAM digit used to access the segment data for the same display digit.

```
; SEVEN-SEGMENT DECODE DATA TABLE:
; ROM BITS I7 - I0 = SA - SG, D.P. (DECIMAL POINT) BITS, RESPECTIVELY

            .PAGE     4          ; PLACE LOOKUP DATA IN WORDS 0 - 9, PAGE 4
LOOKUP:     .WORD     X'FC       ; = 0 (SEVEN-SEGMENT DECODE HEX VALUES)
            .WORD     X'60       ; = 1
            .WORD     X'DA       ; = 2
            .WORD     X'F2       ; = 3
            .WORD     X'66       ; = 4
            .WORD     X'B6       ; = 5
            .WORD     X'BE       ; = 6
            .WORD     X'E0       ; = 7
            .WORD     X'F4       ; = 8
            .WORD     X'F6       ; = 9
            .                    ; NEXT FIVE LOCATIONS CAN BE USED FOR SPECIAL ALPHABETICAL DISPLAY
            .                    ;  CHARACTER DATA
            .

; BEGIN CODE FOR DISPLAY LOOP

            .PAGE     5          ; PLACE FOLLOWING CODE ON PAGE 5
DSPLY:      LBI       0,15       ; POINT TO HIGH ORDER RAM DIGIT, BD = 15
LOOP:       CLRA                 ; A = 0 FOR LOOKUP
            LEI       0          ; BLANK SEGMENTS (EN2 = 0)
            OBD                  ; OUTPUT DIGIT VALUE
            LQID                 ; LOOKUP DATA TO Q
            LEI       4          ; OUTPUT SEGMENT DATA (EN2 = 1)
            CBA                  ; BD TO A
            AISC      15         ; DECREMENT A
            JP        .+3        ; JUMP 3 WORDS WHEN FINISHED
            CAB                  ; A(BD - 1) TO BD
            JP        LOOP       ; DISPLAY NEXT LOWER DIGIT
            .                    ; CONTINUE WHEN FINISHED
            .
            .
```

## 4.8 Interrupt Service Routine

As explained in Section 3.2, LEI Instruction description, setting bit 1 of the EN register enables the COP420-series and COP444L $IN_1$ input as an interrupt input, responding to low going pulses. Upon the occurrence of an interrupt signal, the subroutine stack is pushed and program control is transferred to the last word of page 3 (address $0FF_{16}$). The following routine contains code which may be placed at the beginning and end of the interrupt service routine to save the contents of A, C and B, freeing them for use by the interrupt routine. At the end of the routine the previous contents of A, C and B are restored for use by the main program. It should be noted that the main program need only enable $IN_1$ as an interrupt input once; thereafter, the interrupt service routine, itself, re-enables interrupt servicing (LEI 1 instruction before return).

```
; INTERRUPT SERVICE ROUTINE TO SAVE AND RESTORE THE CONTENTS OF A, C AND B (BR AND BD) IN MEMORY REGISTER 0,
;  DIGITS 0 - 2.
; AUTOMATIC ENTRY TO LAST WORD OF PAGE 3
; ON RETURN, IN1 INPUT RE-ENABLED AS INTERRUPT INPUT

INTSER:  NOP                ; FIRST INTERRUPT ROUTINE INSTRUCTION MUST BE A NOP (LOCATION X'FF)
         XAD      0,0       ; SAVE A IN M(0,0)
         CBA                ; BD TO A
         XAD      0,1       ; SAVE BD IN M(0,1)
         XABR               ; BR TO A
         SKC                ; CARRY = 1?
         AISC     8         ; NO, SET A3
         XAD      0,2       ; SAVE C AND BR IN M(0,2)
         .                  ; PERFORM INTERRUPT ROUTINE
         .
         .
         LDD      0,2       ; M(0,2) (C AND BR) TO A
         RC                 ; RESET CARRY
         AISC     8         ; A3 SET (SAVED CARRY = 0)?
         SC                 ; NO, RESTORE CARRY = 1
         XABR               ; RESTORE BR
         LDD      0,1       ; M(0,1) (BD) TO A
         CAB                ; RESTORE BD
         LDD      0,0       ; M(0,0) TO A, RESTORE A
         LEI      1         ; ENABLE INTERRUPT (SET IN1)
         RET                ; RETURN FROM INTERRUPT SERVICE ROUTINE
```

## 4.9 Timekeeping Routine

The following multilevel subroutine counts time in a 12-hour format. It relies on the COP420 system oscillator, itself (controlled by an inexpensive 3.58 MHz color TV crystal), and the COP420 internal time-base counter for a real-time base, rather than on a 60 Hz external input. The subroutine is entered each time the SKT instruction skips, indicating time-base counter overflow. As explained in Section 3.2, SKT Instruction description, overflow frequency is dependent upon the frequency of the COPS™system oscillator. This frequency equals the oscillator frequency, first divided by 16 by the instruction cycle divider, then by 1024 by the internal 10-bit time-base counter. In this case the SKT overflow frequency will equal a fractional number: 218.478 Hz (3.58 MHz divided by 16, divided by 1024). Consequently, the timekeeping *calling* routine must execute a SKT instruction at least once approximately each 218 Hz to ensure that each SKT overflow is detected.

As indicated above, using an inexpensive TV crystal results in a fractional SKT frequency. Program compensation techniques, therefore, must be employed to derive an integer which may be used by the program in counting seconds, the basic timekeeping units.

This routine derives this integer and utilizes it to keep accurate time in the following manner:

- A 2-digit binary "SKT" counter in RAM is initialized to different values at different times during the course of an hour so that the total counts for the hour equal an integer which corresponds to the 218.478 Hz SKT frequency.
- Every odd second in the range of 0-59 seconds, the SKT counter is set to 218, decremented by 1 each time the SKT instruction skips. When decremented to 0, a 2-digit BCD "seconds" counter in RAM is incremented by 1. (The seconds counter overflows every 60 counts to a 2-digit BCD "minute" counter. The minutes counter overflows every 60 counts to a 1-digit "hours" counter.)
- Every even second in the range of 0-59 seconds, the SKT counter is set to 219 and decremented by 1, as above, each time the SKT pulse occurs.
- Every minute in the range of 0-59 minutes, the SKT counter is set to 218 and decremented as above.
- Every hour, the SKT counter is set to 199 and decremented as above.

The above compensation techniques result in a timekeeping routine which is accurate at the end of each hour. (During the hour, inaccuracy is extremely small.) The basis for the above compensation scheme is as follows:

- Using a 3.58 MHz crystal resulting in a 218.478 Hz SKT frequency, an SKT integer count of 786,521 is obtained each hour (218.478 × 3600 seconds/hour).
- Using the above compensation scheme, the same number of "SKT" counts (786,521) is required to increment the time by 1 hour. This follows since 392,400 counts are required by the "odd" seconds compensation (30 × 60 × 218 counts); 381,060 by the "even" seconds compensation (29 × 60 × 219 counts); 12,862 by the "minutes" compensation (59 × 218 counts) and 199 by the "hours" compensation — resulting in a total hours count of 786,521.

A flowchart and a RAM map for this routine are provided in Figure 4.3. Note that an assembler assignment statement is used in the assembler source code to equate the address of low order digits of the RAM SKT counter and seconds counter with the symbols "COUNT" and "SECS," respectively. This provides clearer documentation of the program since an instruction referencing the seconds counter, for instance, can use the word "SECS" instead of a numerical value in the operand field (i.e., LBI SECS). For further information on the assignment statement, see *PDS User's Manual,* Section 8.4. Also note that the program initializes the SKT counter to 218, 219 and 199, respectively, by loading its two digits with the following binary equivalent pairs (high-order value, low-order value): 13, 10; 13, 11; and 12, 7.

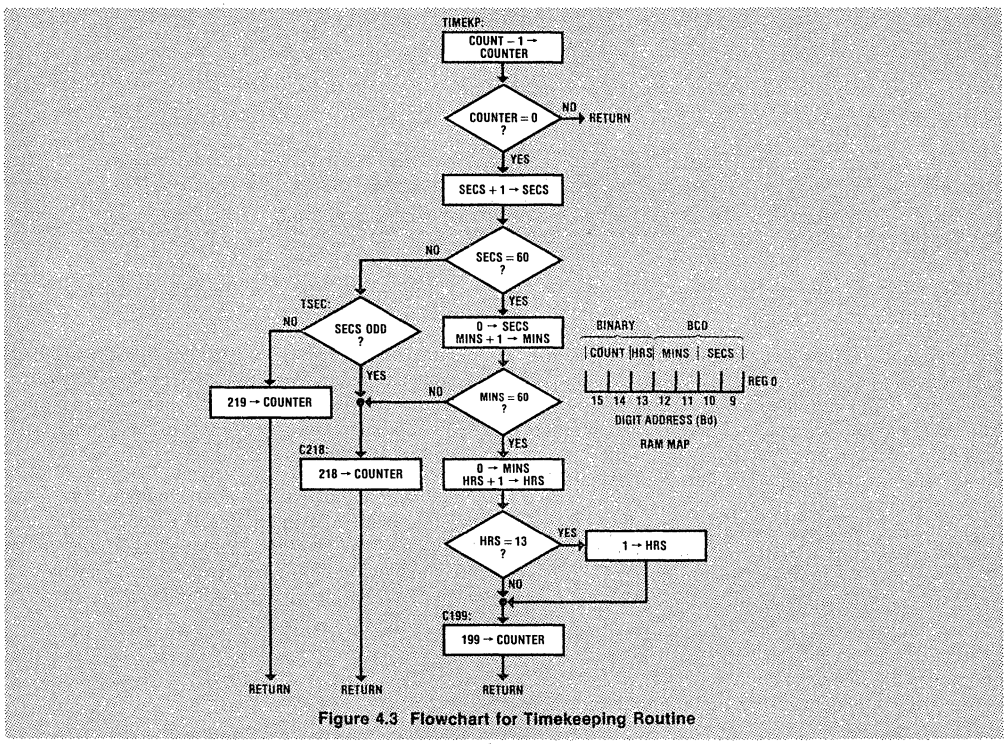



Figure 4.3 Flowchart for Timekeeping Routine

This subroutine is coded to reside on subroutine page 2. The source code provided below also illustrates the use of the PDS Assembler .LOCAL directive and local symbol labels. Specifically, the program begins and ends with a .LOCAL directive, making the memory addresses between them a local region. Within this local region, local symbols (labels whose first character is a "$") will be defined only within the local region — they will not conflict with labels appearing in other portions of program source code. This relieves the programmer from worry about duplicate label definitions, allowing the subroutine or other utility program to be included or added to different programs, regardless of the labels used by these other programs.

In effect, therefore, utility programs or commonly used subroutines may be coded in this manner and placed in separate "utility" files on a disk. They can then be added or included, when needed, to main programs at a later date. For an example of a program which includes this "TIMEKP" subroutine (using the assembler .INCLD directive), see Figure 5.18.

Local symbols must begin with a "$" and be unique within the particular local region in the first *4* characters following the "$." The programmer may, as is done in this example, use local labels with more than four characters for convenience and, although not "recognized" by the assembler, these extra characters will be printed out on the assembler output listing. Note: The label of the starting address of a local utility routine must be a *long* (regular) label, since it will be referenced by a portion of the program outside of the local region (e.g., "TIMEKP" is not a local label).

```
; PAGE 2 SUBROUTINE TO KEEP TIME IN A 12-HOUR FORMAT USING A 3.58 MHZ TV CRYSTAL
; 2-DIGIT "SKT" COUNTER CONTAINED IN M(2,15) - M(2,14): HIGH- TO LOW-ORDER
; 1-DIGIT BINARY HOURS COUNTER IN M(2,13)
; 2-DIGIT BCD MINUTES COUNTER IN M(2,12) - M(2,11): HIGH- TO LOW-ORDER
; 2-DIGIT BCD SECONDS COUNTER IN M(2,10) - M(2,9): HIGH- TO LOW-ORDER
; ENTRY POINT: TIMEKP; ENTRY UPON SKT INSTRUCTION OVERFLOW
; SUBROUTINES CALLED: INC2

            .PAGE     2          ; PAGE 2 SUBROUTINE
            .LOCAL               ; CREATE LOCAL REGION FOR LOCAL SYMBOLS
            $COUNT    =2,14      ; ASSIGN "COUNT" = ADDRESS OF LOW-ORDER SKT COUNTER DIGIT
            $SECS     =2,9       ; ASSIGN "SECS" = ADDRESS OF LOW-ORDER SECONDS COUNTER DIGIT
TIMEKP:
            LBI       $COUNT     ; POINT TO LOW-ORDER DIGIT OF SKT COUNTER
            LD                   ; LOAD DIGIT TO A
            AISC      15         ; DIGIT = 0? (A = DIGIT - 1)
            JP        $HIGHST    ; YES, TEST HIGH-ORDER DIGIT
            X                    ; NO, EXCHANGE DIGIT - 1 INTO M
            RET                  ; RETURN UNTIL NEXT SKT OVERFLOW
$HIGHTST:   XIS                  ; REPLACE DIGIT IN COUNTER, INCREMENT BD
            JP        TIMEKP+1   ; JUMP BACK AND TEST HIGH-ORDER DIGIT — IF ALREADY TESTED AND =0,
                                 ;  SKIP AND CONTINUE
            LBI       $SECS      ; POINT TO LOW-ORDER SECS DIGIT
            JSR       $INC2      ; INCREMENT SECS COUNTER
            JP        $TSEC      ; SECS < 60, TEST SECS FOR ODD OR EVEN
            STII      0          ; SECS = 60, 0 TO HIGH-ORDER DIGIT, POINT TO LOW-ORDER MINS DIGIT
            JSR       $INC2      ; INCREMENT MINS COUNTER
            JP        $C218      ; MINS < 60, SET COUNTER = 218
            STII      0          ; MINS = 60, 0 TO HIGH-ORDER DIGIT, POINT TO HOURS DIGIT
            LD                   ; LOAD HOURS DIGIT TO A
            AISC      1          ; INCREMENT HOURS
            X                    ; PLACE IN M, PREVIOUS HRS TO A
            AISC      4          ; HOURS > 12?
            JP        $C199      ; NO, SET COUNTER = 199
            STII      1          ; YES, SET HOURS = 1
$C199:      LBI       $COUNT     ; POINT TO LOW-ORDER COUNTER DIGIT
            STII      7          ; SET COUNTER = 199 (BINARY 12,7)
            STII      12
            RET                  ; RETURN UNTIL NEXT SKT OVERFLOW
$TSEC:      LBI       $SECS      ; POINT TO LOW-ORDER SECS DIGIT
            SKMBZ     0          ; SECS ODD?
            JP        $C218      ; YES, SET COUNTER = 218 (BINARY 13,10)
$C219:      LBI       $COUNT     ; NO, POINT TO LOW-ORDER COUNTER DIGIT
            STII      11         ; SET COUNTER = 219 (BINARY 13,11)
$C21X       STII13
            RET
$C218:      LBI       COUNT      ; POINT TO LOW-ORDER COUNTER DIGIT
            STII      10         ; SET COUNTER = 218
            JP        $C21X      ; JUMP TO "C21X" THEN RETURN
```

```
; SUBROUTINE TO INCREMENT A 2-DIGIT BCD RAM COUNTER
; ON ENTRY, B MUST POINT TO LOW-ORDER DIGIT OF COUNTER
; ENTRY POINT: INC2
; NORMAL RETURN IF 2-DIGIT VALUE LESS THAN 60
; RETURN THEN SKIP IF 2-DIGIT VALUE EQUAL TO 60
; BOTH RETURNS EXIT WITH B POINTING TO HIGH-ORDER DIGIT

$INC2:
        SC                      ; INITIALIZE C TO 1 TO ADD TO LOW-ORDER DIGIT
        CLRA                    ; ZERO TO A
        AISC    6               ; BCD ADJUST RESULT IF NECESSARY
        ASC                     ; IF RESULT > 9, LOW ORDER DIGIT = 0
        ADT
        XIS                     ; PLACE INCREMENTED DIGIT IN M, POINT TO HIGH-ORDER DIGIT
        CLRA                    ; ZERO TO A
        AISC    6               ; ADD CARRY, IF PROPAGATED FROM LOW-ORDER DIGIT TO HIGH-ORDER DIGIT
        ASC
        ADT                     ; BCD RESULT IF NECESSARY
        X                       ; REPLACE DIGIT IN M
        LD                      ; LOAD HIGH-ORDER DIGIT INTO A
        AISC    10              ; HIGH-ORDER DIGIT = 6 (COUNT = 60)?
        RET                     ; NO, NORMAL RETURN
        RETSK                   ; YES, RETURN THEN SKIP
        .LOCAL                  ; END LOCAL REGION
```

## 4.10 String Search Routine

It is often necessary to search data memory for a string of characters. The following routine searches register 0 for a match with three contiguous 4-bit characters, "X," "Y," and "Z." Note that a match with more than three characters is easily accommodated by providing for additional character tests, using the simple character test instructions provided below containing modified LDD instructions whose operands specify the additional characters to be matched. Also, the code may be easily modified to search through more than one RAM register for a match.

```
; SUBROUTINE TO SEARCH STRING OF DATA MEMORY CHARACTERS FOR A MATCH WITH "X," "Y," AND "Z" CONTIGUOUS
;  CHARACTERS
; 16 4-BIT CHARACTERS ASSUMED STORED IN M(0,15) THROUGH M(0,0)
; "X," "Y," AND "Z" CHARACTERS ASSUMED STORED IN AND ASSIGNED VALUES OF M(1,15) THROUGH M(1,13), RESPECTIVELY
; NORMAL RETURN IF NO MATCH
; RETURN THEN SKIP IF MATCH OCCURS WITH THE ACCUMULATOR CONTAINING THE DIGIT NUMBER OF "X"

        X = 1,15
        Y = 1,14
        Z = 1,13

SEARCH:
        LBI     0,15            ; POINT TO M(0,15)
LOOKX:
        LDD     X               ; X TO A
        SKE                     ; X FOUND?
        JP      NOX             ; NO, JUMP TO X
        XDS                     ; YES, POINT TO NEXT LOWER DIGIT
        JP      LOOKY           ; LOOK FOR Y MATCH, IF AT M(0,0) SKIP AND NORMAL RETURN — NO MATCH
NOX:
        LD
        XDS                     ; DECREMENT DIGIT POINTER
        JP      LOOKX           ; LOOK AGAIN FOR X MATCH, IF AT M(0,0), SKIP AND NORMAL RETURN — NO
        RET                     ;  MATCH
LOOKY:
        LDD     Y               ; Y TO A
        SKE                     ; Y FOUND?
        JP      LOOKX           ; NO, TRY AGAIN
        XDS                     ; YES, POINT TO NEXT LOWER DIGIT
        JP      LOOKX           ; LOOK FOR Z MATCH, IF AT M(0,0), SKIP AND NORMAL RETURN — NO MATCH
        RET
LOOKZ:
        LDD     Z               ; Z TO A
        SKE                     ; Z FOUND?
        JP      LOOKX           ; NO, TRY AGAIN
        OBA                     ; YES, MATCH COMPLETE, COPY Z DIGIT ADDRESS TO A
        AISC    2               ; ADD 2 TO A TO EQUAL X DIGIT ADDRESS
        RETSK                   ; RETURN THEN SKIP — MATCH FOUND
```

## 4.11 Programming Techniques for the COP421-Series, COP410L and COP411L

### COP421-Series Programming

Since the COP421-series differs from the COP420-series only in not having the $IN_3$-$IN_0$ inputs, the foregoing programming considerations and examples for the COP420-series are, for the most part, relevant to COP421-series programming. However, due to its lack of IN inputs, the COP421-series does not include the ININ instruction, and its INIL instruction inputs only CKO into A (when CKO is programmed as a general-purpose input). The following are the results of these COP421 differences:

1. MICROBUS™ interface programming is not available since $IN_3$-$IN_0$ cannot be mask-programmed as WR, CS, and RD, respectively. Also, $G_0$ cannot be mask-programmed as a "ready" output to facilitate "handshaking" with a host CPU over the MICROBUS™ bus. The COP421 may still, however, function as a CPU peripheral component, relying on more general, programmed I/O techniques.
2. Due to the lack of IN inputs, other bidirectional I/O pins must be used as general purpose input pins when implementing a programmed input operation.
3. A hardware interrupt utilizing $IN_1$ is not possible. (Setting $EN_1$ has no effect on the operation of any COP421.) Any interrupt servicing must be accomplished using software interrupt techniques. (The routine provided in Section 4.8 is inapplicable to the COP421-series.)
4. A software interrupt cannot rely on the inputting and testing of the $IL_3$ or $IL_0$ latches associated with $IN_3$ and $IN_0$ inputs. Software interrupts, therefore, require that the interrupt signal be tied to one of the non-latched input pins. As a result, the input interrupt signal must be input and tested at least once during each "low" and "high" pulse occurring during each period of the signal. For example, if the interrupt signal is a 50% duty cycle, 60 Hz square wave, it must be tested at least twice every 1/60 second.

### COP410L/COP411L Programming

Since the COP410L/COP411L, as with the COP421-series, does not have IN inputs, the above programming considerations relating to the COP421 apply as well as to COP410L/COP411L programming. Also, since, as discussed below, other hardware logic elements are not included in the architecture of the COP410L, the following additional considerations apply to COP410L programming:

1. The COP410L/COP411L has one-half the ROM and RAM of the COP420-series and COP421-series. ROM, therefore, consists of 512 x 8-bit words, limiting program code to eight pages (pages 0-7). RAM consists of a 32 x 4-bit RAM, organized as four RAM registers (0-3) consisting of *8* 4-bit digits (9-15,0). The LBI register reference instruction should, therefore, contain a "d" field equal to 9-15 or 0. Since all LBIs will reference RAM digits 9-15 or 0, all LBIs are single-byte instructions, occupying one word in program memory. A field restriction occurs with respect to the memory reference XAD instruction: only an XAD 3,15 instruction is valid, limiting its use to reference a RAM "scratch-pad" digit *contained in M(3,15) only.*
2. The COP410L/COP411L has 2 subroutine save registers, SA and SB. Only two levels of subroutine nesting, therefore, are allowed. The programmer should also realize that since LQID pushes and pops the stack in performing the operation associated with this instruction, only *1* level of subroutine nesting should be in effect at the time of the execution of this instruction. (Otherwise the second level of previous subroutine nesting will be disrupted — the previous contents of SB will be lost.)
3. Since the COP410L/COP411L does not have an internal divide-by-1024 time-base counter, the SKT instruction is not available. "Real-time" routines, such as 12-hour timekeeping and the like, must rely on *external* time-base inputs in order to derive a time-base for such routines (e.g., external 50/60 Hz input for time-of-day routines).
4. Certain deleted or altered instructions have already been mentioned: INIL, ININ, and SKT are not available; LBIs must have a "d" field equal to 9-15 or 0, and XAD's operand must equal 3,15. The following instructions have also been deleted from the COP410L/COP411L instruction set. To the right of each of the following deleted instructions, where appropriate, alternative COP410L/COP411L instructions are shown which, when executed in succession, will perform the same or similar operation as the deleted instruction:

| Deleted Instructions | Alternative COP410L/COP411L Instructions |
|---|---|
| LDD | LBI, LD |
| CASC | COMP, ASC |
| ADT | AISC 10, NOP |
| CQMA | INL |
| OGI | OMG |
| XABR | |
| SKT | |
| ININ | |
| INIL | |

For further information on deleted or altered COP410L/COP411L instructions and the operations performed by the alternative instructions given above, see Section 3.4.

# 5 COP400 I/O Techniques

This chapter provides information and examples pertaining to hardware and software interfacing techniques for the COP400 Microcontrollers. The information contained in this chapter is derived, in large part, from material already provided in previous chapters, particularly Chapter 2. The reader should refer to this chapter when reading the following material to obtain a complete picture of the COP400 series I/O characteristics and capability.

The following text provides I/O examples for the COP420 specifically. The I/O capability of the other members of the COP420-series (e.g., COP420L and COP420C), the COP444L and other, less inclusive devices, the COP410L and COP411L, are summarized in Table 5.1.

## 5.1 Hardware Interfacing Techniques

### COP420 I/O

Figure 5.1 depicts the I/O lines associated with the COP420. As indicated, there are 24 I/O lines. The following discussion provides information on the capabilities of the mask-programmable I/O options associated with the COP420. These optional configurations are shown in Figure 5.2.

### COP420 Inputs

COP420 inputs may be programmed either with a depletion-load device to $V_{CC}$ or floating (Hi-Z input). All inputs are TTL/CMOS compatible. Hi-Z inputs should not be left floating; they should be connected to the output of a "high" and "low" driving device if active or to $V_{CC}$ or ground if unused. Inputs may also be optionally programmed for higher trip levels for interfacing to non-TTL sources (e.g., keyboards, switches).

**Table 5.1 COP400 Comparison Chart**

| I/O Pins | Bits | COP420 | COP420C | COP420L | COP410L |
|---|---|---|---|---|---|
| $D_{OUT}$ | 4 | TTL | TTL | 20mA Sink | 20mA Sink |
| $G_{OUT}$ | 4 | TTL | TTL | 20mA Sink | LS TTL |
| $L_{OUT}$ | 8 | TTL or LED | TTL or LED | LS or LED | LS or LED |
| SO, SK | 2 | TTL | TTL | LS | LS |
| IN | 1 | 4 Inputs | 4 Inputs | 4 Inputs | No |
| SI | 1 | Shift Register or Counter Input | Shift Register or Counter Input | Shift Register or Counter Input | Shift Register or Counter Input |
| CKI | 1 | Oscillator Input | Oscillator Input | Oscillator Input | Oscillator Input |
| CKO | 1 | Oscillator Out or SYNC In or General In or RAM Supply | Oscillator Out or SYNC In or General In | Oscillator Out or SYNC In or General In or RAM Supply | Oscillator Out or SYNC In or RAM Supply |
| RESET | 1 | RESET Input | RESET Input | RESET Input | RESET Input |
| $V_{CC}$, GND | 2 | Power Supply | Power Supply | Power Supply | Power Supply |
| Oscillator Frequency Range | | 0.4 to 4MHz | 32kHz to 2MHz | 0.2 to 2MHz | 200 to 500kHz |
| Cycle Time | | 4 to 10μs | 15 to 250μs | 15 to 40μs | 15 to 40μs |
| $V_{CC}$ Supply | | 4.5 to 6.3V | 2.4 to 6.3V | 4.5 to 9.5V | 4.5 to 9.5V |
| $V_{CC}$ Current (max) | | 25mA | 1mA (25μA) | 8mA | 5mA |

### COP420 Outputs

**Standard Output:** The N-channel device to ground is good at sinking current and is compatible with the sinking requirements of 1 TTL load (1.6mA at 0.4V); it will meet the "low" voltage requirements of CMOS logic. All output options use this device (device #1), as illustrated in Figure 5.2, for current sinking. The depletion-load device to $V_{CC}$ provides low sourcing capability (100μA at 2.4V). While this device meets the sourcing requirements of TTL logic and will go to $V_{CC}$ to meet the "high" voltage requirements of CMOS logic, an external resistor to $V_{CC}$ may be required to interface to other external devices requiring higher sourcing capability. A standard output may be connected directly to the base of an external transistor for current sourcing since the depletion-load device's current capability is limited to a safe operating area. Figure 5.3 provides a summary of the characteristics of the COP420 Standard Output.

**Open-Drain Output:** The COP420 open-drain output uses the same enhancement mode device to ground as the standard output with the same current sinking capability. As its name implies, this output configuration does not contain a load device to $V_{CC}$, allowing various external pullup techniques as required by the user's application.

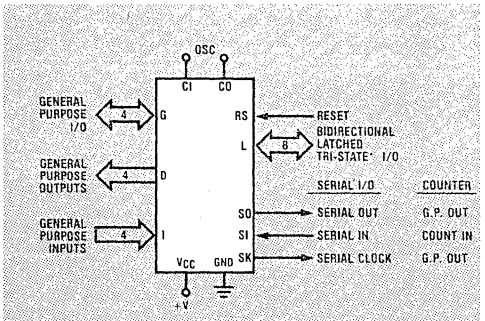



Figure 5.1 COP420 I/O Lines

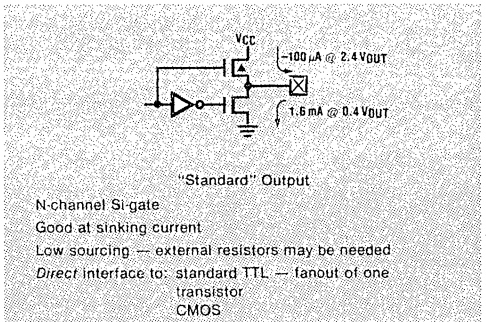



Figure 5.3 COP420 Standard Output Characteristics

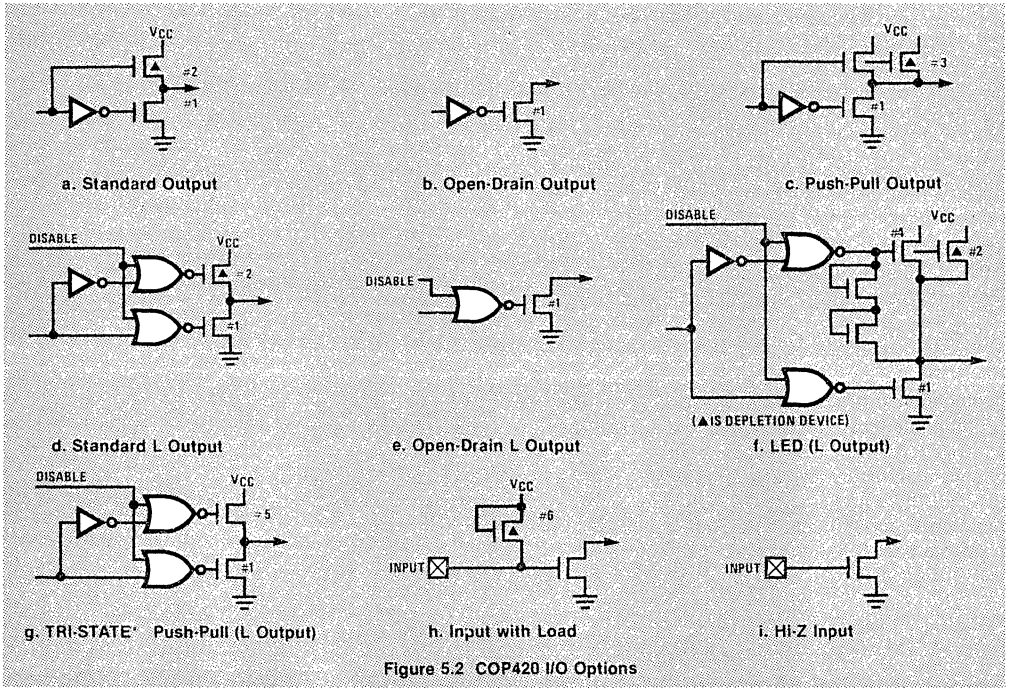



Figure 5.2 COP420 I/O Options

**Push-Pull Output:** The COP420 push-pull output differs from the standard output configuration in having an enhancement mode device in parallel with the depletion-load device to $V_{CC}$, providing greater current sourcing capability and faster rise and fall times when driving capacitive loads. This option is available for the COP420 SO and SK outputs, often tied to the highly capacitive clock lines of external shift registers to provide additional external I/O for the COP420. (For an example, see Figure 5.20.) If a push-pull output is interfaced to an external transistor, a limiting resistor must be placed in series with the base of the transistor to avoid excessive source current flow out of the push-pull output.

Figure 5.4 summarizes, in interconnect form, the information provided above relevant to the capabilities of the push-pull, open drain and standard outputs, as well as the Hi-Z and load device input configurations.

For an example of use of the SK output, configured as a push-pull output to drive the clock lines of an external shift register, see Figure 5.10.

**LED Direct Drive Output:** The COP420 LED direct drive output differs from the standard output configuration in two basic ways:

1. Its depletion-load device to $V_{CC}$ is paralleled by an enhancement mode device to $V_{CC}$ to allow for the greater current sourcing capacity required by the segments of an LED display. Source current is clamped to prevent excessive source current flow.
2. This configuration can be disabled under program control by resetting bit 2 ($EN_2$) of the enable register to provide simplified display segment blanking. However, while both enhancement mode devices are turned off in the disabled mode, the depletion-load device to $V_{CC}$ will still source up to 0.125mA when this output is turned off. (This is not a worst case pull-up for keyboard input loads).

For an example of use of the L I/O ports, using this option, to directly drive the segments of a LED and VF display, respectively, see Figures 5.11 and 5.12.

### TRI-STATE® Push-Pull Output

This COP420 output was designed to meet the specifications of National's MICROBUS™, outputting data over the data bus to a host CPU. It has TRI-STATE® logic to disable both enhancement mode devices to free the MICROBUS™ data lines for COP420 input operation. Figure 5.13 shows an interconnect between a host CPU and the COP420 over the MICROBUS™ using this L output option.

### COP420 I/O Summary

Figures 5.5 through 5.9 provide diagrams of the internal logic and a summary of the hardware and software features associated with the COP420 I/O ports.

### Interconnect Examples

Figures 5.10 through 5.14 provide interconnect diagrams illustrating several schemes for interconnecting the COP420 to external devices. Several of these interconnect diagrams, with minor variations, are used in providing software I/O techniques in the final sections of this chapter.

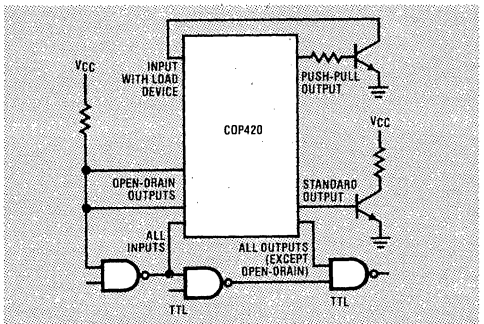
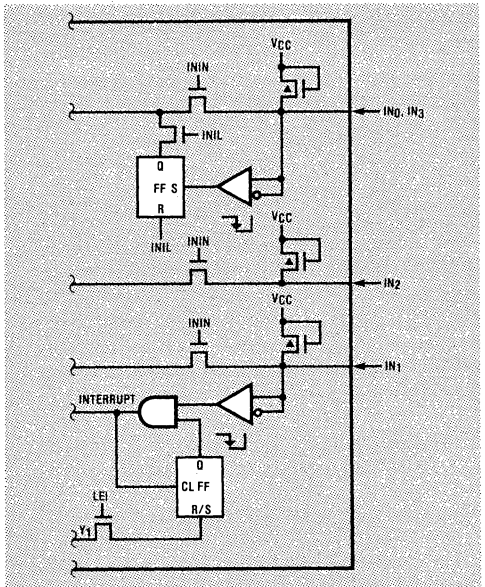


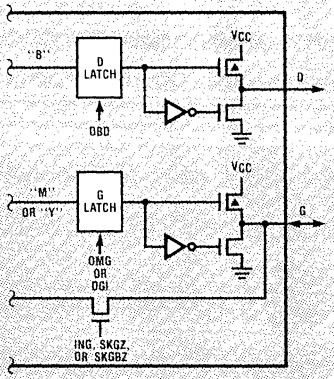

**Figure 5.4 COP420 I/O Interconnect Examples**




IN Inputs

1. Four general purpose inputs read directly into A (ININ instruction);
2. $IN_1$ can be enabled as an interrupt input (by setting $EN_1$);
3. $IN_0$ and $IN_3$ can "catch" low-going pulses, read into $A_0$ and $A_3$ (INIL instruction);
4. All inputs have optional pull-up load device to $V_{CC}$ (shown in diagram), or Hi-Z (floating) inputs.

**Figure 5.5 COP420 IN Port Characteristics**




D Outputs

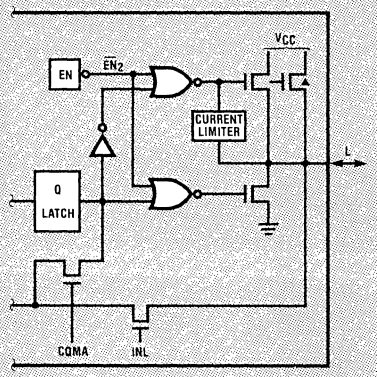
1. Four general purpose outputs loaded from B (OBD) instruction;
2. Standard (as shown) or open-drain outputs.

G Inputs

1. Four general purpose I/O lines loaded from memory (M) by OMG instruction or loaded with immediate data (Y) by OGI instruction;
2. Read inputs into accumulator (ING instruction), test individually (SKGBZ instruction), collectively (SKGZ instruction) for zero — seg G latch to "1" when using as input;
3. Standard (as shown) or open-drain outputs.

**Figure 5.6 COP420 D and G Port Characteristics**

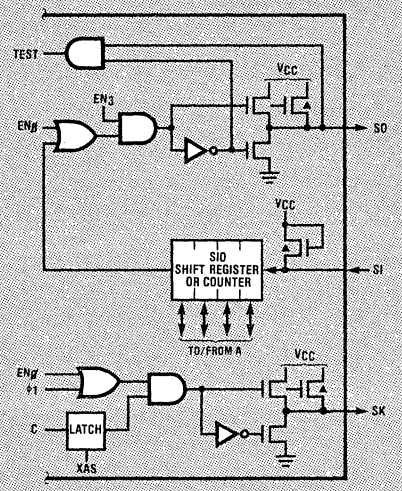





L TRI-STATE® Inputs/Outputs

1. Eight TRI-STATE inputs/outputs, loaded with Q latch data by setting $EN_2$ or direct input of L port data to M and A (INL instruction); Q latch loaded from A and M by CAMQ instruction and read into M and A by CQMA instruction;
2. L ports TRI-STATED with $EN_2 = 0$ (if output contains depletion-load device to $V_{CC}$, $I_{OL} \approx 0.2$mA @ 0V in);
3. All output options available:
   a. Standard
   b. Open-Drain
   c. Push-Pull
   d. LED Direct Drive (as shown)
   e. TRI-STATE Push-Pull

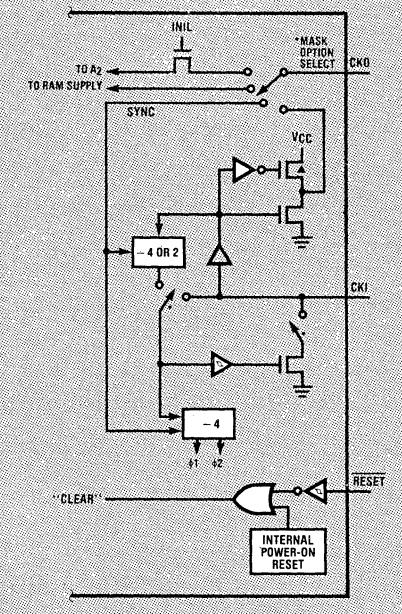
**Figure 5.7 COP420 L I/O Port Characteristics**




SI Input, SO, SK Outputs

1. SI is a single-pin input to the SIO register. SIO can be enabled as a 4-bit serial shift register or a 4-bit binary counter, selected by $EN_0$.
2. If SIO is selected as a counter, SO outputs the value of $EN_3$, SK outputs the value of C upon the execution of an XAS instruction.
3. If SIO is selected as a shift register, SO may be used as a serial data output and SK may be a logic controlled clock selected by $EN_3$.
4. The contents of SIO may be exchanged with A using an XAS instruction.
5. SI, SO and SK are also used for "in-house" standardized testing of the COP420.
6. SI may be configured with a load device to $V_{CC}$ (as shown) or as a Hi-Z input.
7. SO and SK may be configured as:
   a. Standard
   b. Open-Drain, or
   c. Push-Pull (as shown) outputs.

**Figure 5.8 SI, SO, SK Characteristics**




CKO, CKI and RESET Pins

1. The COP420 CKO pin has the following options:
   a. output to crystal oscillator;
   b. general purpose input (read into $A_2$ by an INIL instruction);
   c. synchronization (SYNC) input;
   d. RAM power supply pin.
2. CKI has the following options:
   a. crystal oscillator input;
   b. external oscillator input;
   c. RC controlled oscillator input.
3. RESET may be used as an external reset pin or, if the power supply rise time is greater than 1 ma, as a power-on clear input.

**Figure 5.9 COP420 CKO, CKI, RESET Characteristics**

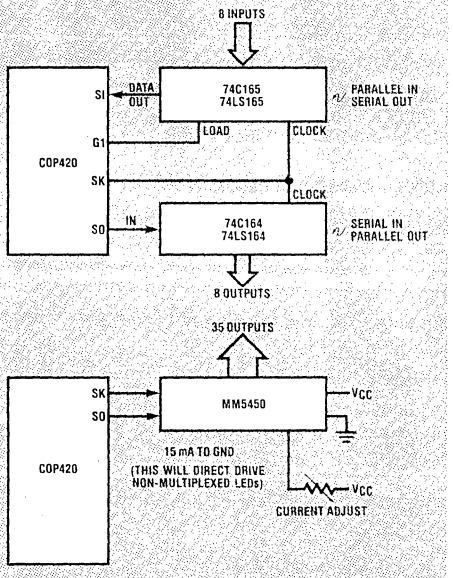



Figure 5.10 COP420 I/O Expansion

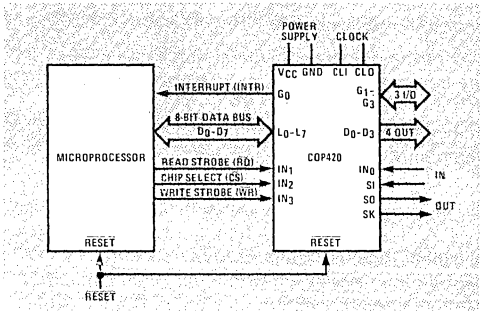



Figure 5.13 MICROBUS™ Interconnect

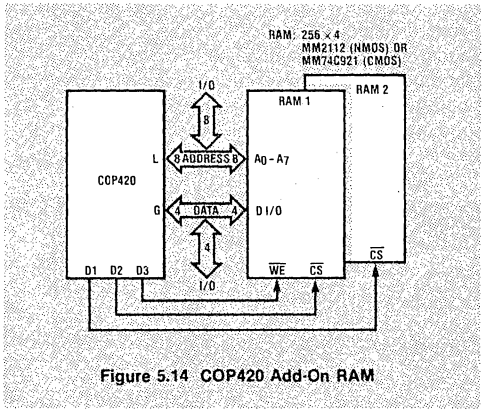



Figure 5.14 COP420 Add-On RAM

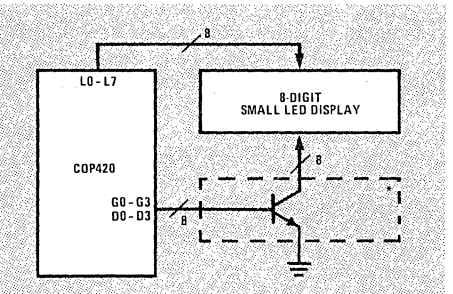



Figure 5.11 COP420 LED Display System

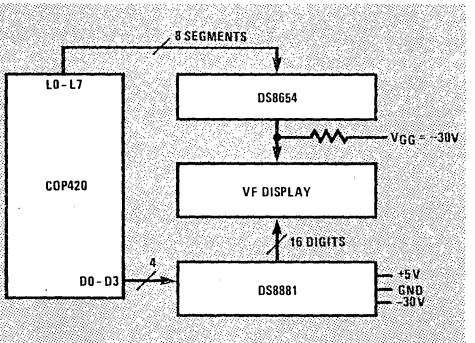



Figure 5.12 COP420 VF Display System

### COP400 I/O Comparison Table

Table 5.1 provides a comparison table of the I/O capabilities of COP400 series devices. It should be understood that this is a partial listing of COP400 devices, since more inclusive parts (the COP440 and its related devices) as well as other devices will be available in the near future. For complete information on the listed devices, as well as other members of the COP400 Microcontroller family, consult the appropriate data sheets.

## 5.2 Software I/O Techniques

The following sections of this chapter provide several software I/O examples and techniques for interfacing the COP420 to external I/O, including program code necessary to service these peripherals.

## 5.3 Keyboard/Display Interface

One of the primary considerations in the design of the internal architecture of the COP400 family was to allow for easy interface to keyboards and numeric displays, the input and output peripherals commonly associated with small system applications, using a minimum amount of external circuitry. To further aid in the implementation of such systems, the instruction set was carefully designed to service these peripherals and handle BCD data manipulation with a minimum amount of external circuitry and program code. The following sections describe a typical keyboard/display interface system to output BCD data stored in data memory (RAM) to a 14-digit LED display, and input keyswitch closure data entered from a 4 × 4 keyboard matrix. In addition, the sample program also makes provision for a timekeeping routine, another typical user application.

Figures 5.15 through 5.18, respectively, provide the hardware interconnect diagram, program flowchart, display timing diagram and assembly source code for the basic interface scheme. The general approach of the interface is common to most keyboard/display interfaces. It takes advantage of the fact that an image persists in the eye for a fraction of a second after the source is removed. It is not necessary, therefore, to have all display digits on simultaneously: the digits are sequentially enabled (multiplexed) at a rate fast enough to avoid noticeable flicker. Multiplexing greatly reduces the amount of interconnect and buffer hardware required.

The most common type of display consists of several seven-segment digits (see lower right section of Figure 5.15). Each light emitting diode segment has two terminals and conducts current in only one direction. Various combinations of segments are turned on to represent numbers and a few alphabetical characters. In our example, the cathodes of all segments (Sa-Sg, D.P.) *in a given digit* are connected together and the anodes of corresponding segments of the *different digits* are also connected together (common cathode display).

The cathode or digit lines are driven by a decoder/driver device, the DS8664, which provides a 4-to-14 buffered decode of the COP420 *D* outputs.

The anode or segment lines are driven directly by the COP420 *L* I/O ports, utilizing the L output *LED Direct Drive* output option. A given segment is turned on only if both its digit and segment lines are driven.

Each digit of the display is multiplexed, with each digit scanned in sequence by changing the binary output code at the D outputs. The DS8664 decoder/driver will set a corresponding D line to a low level to drive each cathode. At the same time the L outputs are set at a high level to correspond to the values necessary to turn on the segments associated with the numeric or alphabetical character to be displayed for the present digit. (To display a "3" at digit 5, segments Sa, Sb, Sc, Sd and Sg would be driven high when $D_5$ is driven low.)

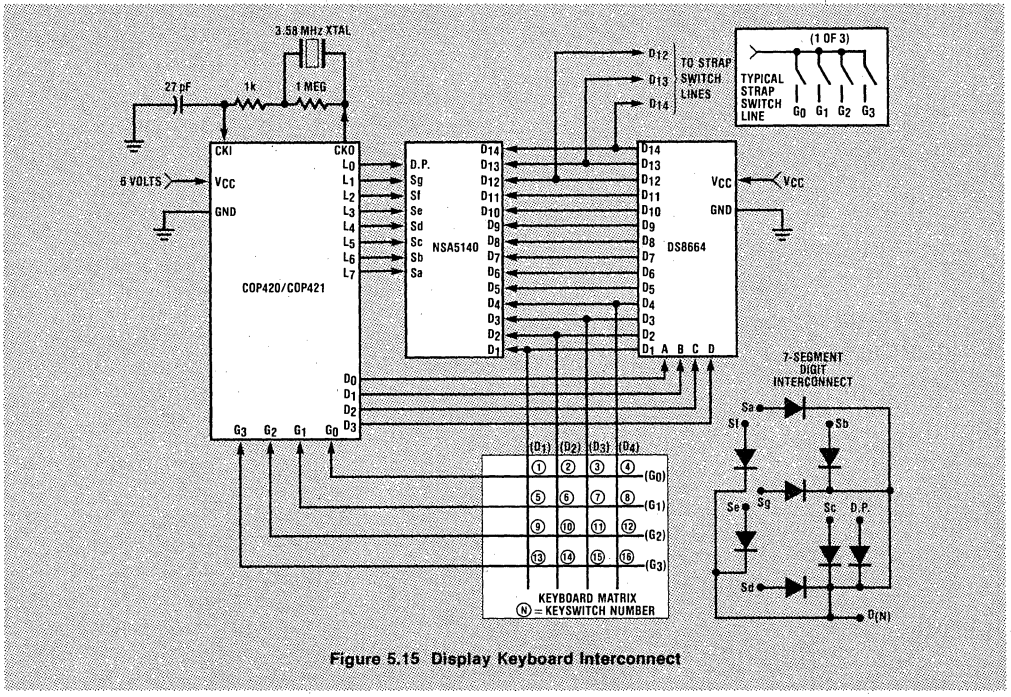



Figure 5.15 Display Keyboard Interconnect

Since people operate keyboards at a rate which is very slow compared to the COP420 instruction cycle time, it is possible to scan the keyboard as well as service the display and execute the timekeeping routine without missing a key closure. As with the display, the keys are connected in a matrix to minimize interconnect. Further economy is gained by sharing the D lines with the display. In fact, the program loop used to scan the display is also used to scan the keyboard. When the program addresses a display digit, it also addresses a column in the keyboard matrix. The program senses the closure of a particular key in that column by testing the *G* I/O ports which are tied to the rows of the keyboard matrix: each key is associated with the conjunction of one D line and one G I/O line.

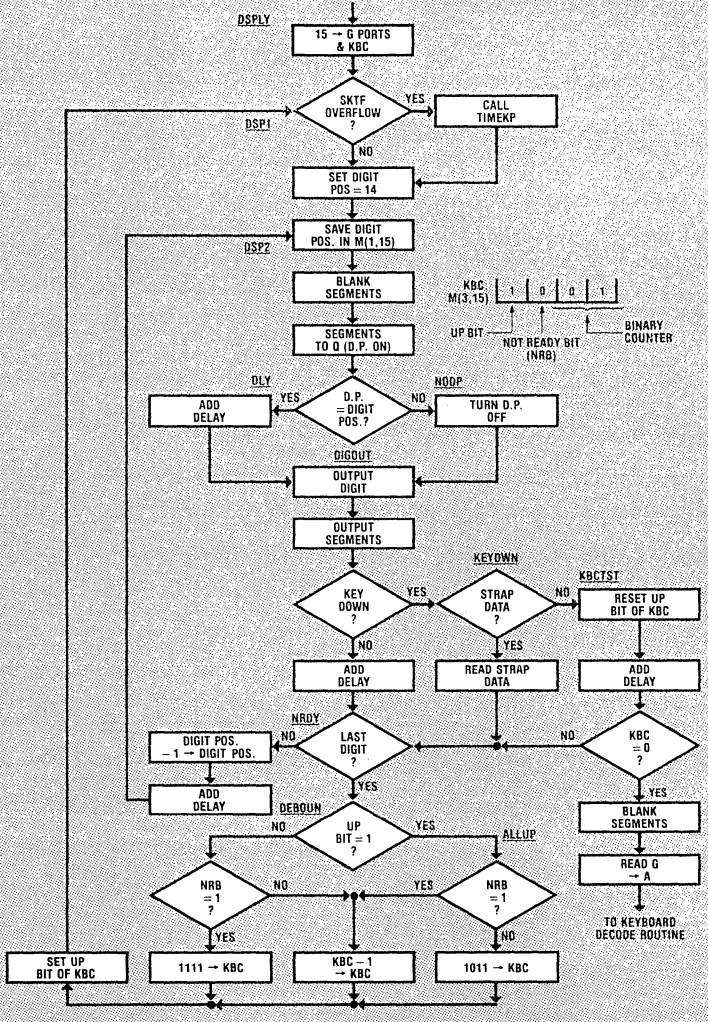
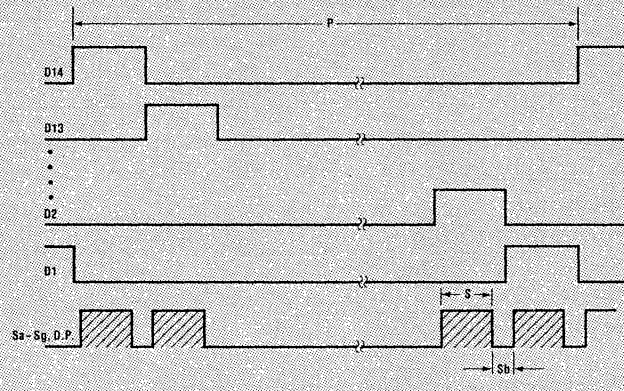



**Figure 5.16 Flowchart for Display/Keyboard Debounce Routine**

The following is a list of design criteria and considerations relevant to the sample keyboard/display interface:

1. With this design, if two keys on different *G* I/O lines are pressed simultaneously, key identity may be lost. After sensing a key closure, the program requires that the keyboard be clear (no keys pressed) for a short duration before it will input another key. "Rollover" and "shift-key" schemes may be implemented with more sophisticated designs.
2. Multiple key closures on the same *G* I/O line will allow segment current to flow through the keyboard causing display digits to be ANDed. Key closure is still detected, however, because the "on" driver presents a small resistance to GND compared to the resistance that the "off" driver and G port present to $V_{CC}$. The ANDing of display digits may be prevented by placing diodes on each digit line. If key identity must be maintained when more than two keys are closed, a diode must be placed in series with each keyswitch.
3. For this design, the G ports are configured as standard outputs (options 21-24 = 0). The program itself sets them each to "1" at the beginning and on each pass through the main program loop. When all keys in the associated matrix row are up, the port will read as a "1." When a key is closed, its corresponding D line will pull the associated G port low, with a "0," therefore signifying key closure.
4. The L ports are configured as *LED Direct Drive* outputs (options 5-8 and 12-15 = 2) to directly drive the segments of the LED display. An average L output source current capability of 8 mA is assumed, being midway between the minimum (2.5 mA) and maximum (14 mA) current sourcing specifications for this output configuration at $V_{CC} = 6V$.
5. To prevent flickering of the display, the display should be refreshed at a rate of at least 100 Hz (1/P in Figure 5.17).
6. The duty cycle (S/P in Figure 5.17) must be maintained to ensure adequate brightness. The L port segment current capability is assumed, as mentioned above, to be 8 mA and the NSA5140 requires 0.5 mA *average* current. Average current is determined by the segment duty cycle and should be the average display current requirement divided by the peak output current or 0.5 ÷ 8 = 1/16. Therefore, the program must be written to ensure a duty cycle of at least 1/16 for proper LED display brightness.
7. Each segment on time (S in Figure 5.17) must be the same width to ensure that all digits are *uniformly* bright.
8. Since keyswitches bounce, the program must debounce or filter the signals on the G lines. This is achieved by requiring that a key be held down for at least four display cycles before being accepted. A key must also be lifted for at least four display cycles before a new key can be accepted.
9. To prevent crosstalk or ghosting between display digits, a LED display requires segment blanking (Sb in Figure 5.17).




Period = P = 896 Instruction Times = 4ms @ 4.5μs/instruction
Display Rate = 1/P = 1/4ms = 250Hz
Segment Time = S = 56 Instruction Cycle Times
Segment Blanking = Sb = 13 Instruction Cycle Times
Duty Cycle = S/P = 56/896 = 1/16

**Figure 5.17 Display Timing Diagram**

10. The system clock oscillator is configured as a crystal controlled oscillator with the instruction cycle frequency derived by driving the crystal oscillator frequency by 16 (options 2 and 3 = 0). This interface scheme uses an inexpensive 3.58 MHz TV crystal to provide the clock oscillator frequency, divided by 16 to derive a 4.5 μs instruction cycle time. This also allows use of the "TIMEKP" (timekeeping) routine given in Section 4.9, which uses the internal COP420 Time-Base Counter and the SKT instruction, together with program compensation techniques, to provide a "real time base" for keeping time — eliminating the need for an external 60 Hz real-time input and associated external circuitry.

**Sample Display/Keyboard Debounce-Decode Program**

Figure 5.16 depicts the flowchart for the sample display/keyboard debounce routine. The actual assembly source code written to perform the flowchart operations is given in Figure 5.18.

Following the flowchart from top to bottom, and referring to the source code where appropriate, the following sequence of operations is performed:

1. The G port is set to 15 (each G line set to "1"). This allows them to be driven low when scanned by their associated D lines. If a keyswitch is closed, the associated G line will therefore become a "0," to be input and tested by the keyboard servicing routine.

2. The program initializes the KBC (Keyboard Debounce Counter) to 15 ($1111_2$). This counter name, as well as two other RAM status digit names, "DIGIT" and "STORE," are assigned the values of their RAM register and digit numbers by assembler assignment statements at the beginning of the source code. This allows these names to be substituted in the operand field of instructions which reference these RAM digits, providing more effective documentation of the source code program. For example, since the KBC is located in RAM register 3, digit 15 and since this value (3,15) must be contained in the operand field of an instruction referencing the KBC, an assignment statement of KBC = 3,15 is written at the beginning of the program. Thereafter, an instruction referencing the KBC may use its name, rather than its RAM value, in the operand field of the instruction (e.g., an LBI KBC will be interpreted by the assembler as an LBI 3,15).

   The contents of the KBC are depicted in the upper right hand corner of the flowchart. From left to right, the bits of the counter indicate the following status conditions: the "up" bit, set to "1" if all keys are up; the "not ready" bit (NRB), set to "1" if keyswitch data has not been debounced; two binary counter bits, both set to "1" at the beginning of the debounce sequence. As will be seen, the two leftmost bits of the KBC ("up" and "NRB") are tested during the debounce routine to determine which branch of the routine will be executed. The rightmost bits, the binary counter bits, provide a binary count of the number of times the program falls through the debounce routine.

3. The internal time-base counter is tested for overflow by an SKT instruction, calling the "TIMEKP" subroutine given in Section 4.9 to keep time if the SKT instruction tests "true."

4. The digit position is set to 14, the most significant digit of the display. As indicated by the source code, the digit position is set by loading Bd (the RAM digit-select register) with the digit position value with an LBI instruction. Bd is later output to the D ports using an OBD instruction, decoded by the DS8664 to enable the appropriate display digit line ($D_{14}$-$D_1$). Since, as mentioned, Bd also functions as a pointer to a particular RAM digit as well as being the source of a direct output of data to the D ports, loading Bd also is used to access the contents of a particular RAM digit, used later by an LQID instruction to obtain seven-segment decode data contained in a lookup table. Because of this dual function of Bd, the segment data for a particular *display* digit should be located in a numerically corresponding *RAM* digit of the RAM display register (register 0). For example, when Bd is set to 14 by an LBI 0,14 to later enable *display* digit 14, it will also be used to obtain the segment lookup data for that display digit located in *RAM* register 0, digit 14. Consequently the segment data pointers for *display* digits 14 through 1 are located in RAM register 0, digits 14 through 1, respectively.

   As will be seen below, the segment data contained in a particular RAM digit, although used by LQID to obtain the actual seven-segment data output to the display, will equal the binary equivalent of the numeral to be displayed (e.g., if a RAM register 0 digit contents = $0010_2$, the LQID instruction will access the seven-segment diode data for the numeral "2." RAM digit contents equal to 10-15 will be used to access special seven-segment alphabetical characters.

5. The value of the digit position loaded into Bd is saved in M(1,15), equated by an assembler statement, as explained in 1. above, to the symbol name "DIGIT." The digit value is saved for later manipulation by the display program (testing, decrementing).

6. The segments of the display are blanked, a requirement for LED multiplexed displays. This is accomplished by disabling the drivers from the Q latches (which contain the seven-segment decode display data) to the L ports by resetting bit 2 of the EN register with an LEI 0 instruction.

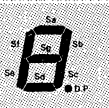
With the L drivers thus disabled, the L I/O ports are disabled, turning off the segments of the display.

7. Next, the program utilizes an LQID instruction to access and load seven-segment decode data contained in a lookup table into the Q latches. This is accomplished in the following manner: as explained in Section 3.2, LQID loads $Q_7$-$Q_0$ with the 8-bit contents of ROM ($I_7$-$I_0$) pointed to by $P_9$, $P_8$, A and M. In this example, LQID is located in page 0, with the result that, at the time of execution, $P_9, P_8 = 0,0$. The program sets A = 0100 with an AISC 4 instruction before execution of the LQID instruction so that $P_7, P_6 = 0,1$ and $P_5, P_4 = 0,0$. Since the upper 4 bits of P may be thought of a ROM "page-select" bits, selecting 1 of 16 pages (0-15) and, since these 4 bits will equal 0001 at the time of the execution of the LQID instruction, it will always "look to" page 1. The lowest 6 bits of P ($P_5$-$P_0$) may be thought of a ROM "word-select" bits, selecting 1 of 64 (0-63) words on a "looked-to" page. Moreover, $P_5$ and $P_4$, the upper 2 bits of these 6 word-select bits, may be thought of as ROM "sub-page-select" bits, selecting 1 of 4 (0-3) successive groups of 16 words on a 64-word ROM page. Since $P_5$ and $P_4$ will always equal 0,0 upon the execution of the LQID instruction, it will always look to one of the first 16 words located in page 1. Since the contents of M (the RAM digit pointed to by the B register), are loaded into the lowest 4 bits of P ($P_3$-$P_0$), it is the binary contents of M directly (0-15) which determine which of the first 16 words (0-15) on page 1 are "looked up" and placed in Q.

   In effect, M is the only variable involved in the LQID operation with its contents directly determining which one of the 16 words in page 1 (words 0-15) are loaded into Q. Of course, the seven-segment decode values have been placed in these locations. Also, as indicated above, the first 10 words (locations 0-9) have been loaded with the seven-segment decode values for the numerals 0-9, respectively. Consequently if M = 3 binary ($0011_2$), a LQID will place the seven-segment lookup data for a display numeral 3 into Q. If M = 10-15 binary, LQID will place the seven-segment decode values for the special alphabetical characters P, A, U, C, F and E, respectively, into Q, since page 1, locations 10-15, contain the decode values to display these characters on the display.

   The hexadecimal value of the seven-segment lookup data is placed in page 1, locations 0-15 with the assembler .WORD directive. Although operands of the .WORD may be concatenated (i.e., .WORD X'FD, X'1F, ...), each 8-bit segment decode value has been placed in successive memory locations with a separate .WORD directive. It should be noted, as indicated by the comments to the program, that ROM word bits $I_7$-$I_0$ (rightmost to leftmost) represent and are tied via the L ports to the Sa-Sg, D.P. segments of the display. A "1" bit for a particular segment means that that segment will be turned on. In all cases, each seven-segment decode word has the D.P. bit ($I_0$) seg; if not later reset by the program the decimal point segment of a particular digit will be turned on when that digit is serviced. See Table 5.2 for a representation of the interconnection of the seven-segments of a display digit and a list of binary and hex values associated with setting the segments of a digit to display the numerals 0-9.

**Table 5.2 Seven-Segment Decode Values**




| | Binary Values | | | | | | | | Hex Values | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | Sa·Sdp | Sdp·Sa |
| Display | Sa | Sb | Sc | Sd | Se | Sf | Sg | Sdp | →$I_7$·$I_0$ | →$I_7$·$I_0$ |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | FC | 3F |
| 0. | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | FD | BF |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 60 | 06 |
| 1. | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 61 | 86 |
| 2 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | DA | 5B |
| 2. | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | DB | DB |
| 3 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | F2 | 4F |
| 3. | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | F3 | CF |
| 4 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 66 | 66 |
| 4. | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 67 | E6 |
| 5 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | B6 | 6D |
| 5. | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | B7 | ED |
| 6 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | BE | 7D |
| 6. | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | BF | FD |
| 7 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | E0 | 07 |
| 7. | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | E1 | 87 |
| 8 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | FE | 7F |
| 8. | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | FF | FF |
| 9 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | E6 | 67 |
| 9. | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | E7 | E7 |

8. A comparison is made to see whether the decimal point position stored in RAM is equal to the digit position of the digit to be displayed during the present pass through the display loop. If the comparison result is "false," the program jumps to "NODP," which resets the least significant bit of Q to keep the decimal point segment of the current digit off when Q latch data is later output to the display via the L ports. Note that an X instruction must follow the CQMA and precede the CAMQ instruction to maintain the integrity (bit-weights) of the Q data, since these instructions perform opposite exchanges with respect to A and M. (See Section 3.2.)

9. If the comparison tests "true," the least significant bit of Q is left set to turn on the decimal point of the current digit and a delay is added to ensure that the program will require

the same amount of execution time whether or not the comparison tests "false" (goes to "NODP") or "true." This and other delays contained in the program ensure that the servicing of a particular display digit will always require the same number of instruction cycle times regardless of which branch of the program is executed during a pass through the program; this is necessary for equal segment-on time for each digit and uniform brightness among the various digits of the display.

10. Digit position data is output from Bd to the D outputs, decoded by the DS8664, enabling the appropriate digit of the display and scanning the corresponding D line (if connected) to the keyboard matrix column or strap switch line.

11. Segment data is output to the current digit by enabling the L drivers with an LEI 4 instruction, setting bit 2 of the EN register and outputting the 8-bit Q latch data to the L I/O ports, the latter connected directly to the segments of the display.

12. Having output data to one digit of the display, the program now begins to service the keyboard. A test is made to see whether any key closure has occurred. If so, the program jumps to "KEYDWN," first testing to see if the key closure occurred on a strap digit line. If this test result is true, the strap data is read into RAM and the program goes to "NRDY." If the key closure was associated with the keyboard matrix, the "up" bit of the KBC is reset and the KBC is tested for all 4 bits equal to 0. If the KBC equals 0, indicating a debounced keyswitch closure, the program blanks the display, inputs the G port (keyswitch row data) into A, and jumps to the keyboard decode routine. If the KBC did not equal 0, the program also goes to "NRDY" (with the KBC "up" bit reset to indicate a key closure).

    It should be noted that the "up" bit is not reset if the key closure was a strap data switch. As will be seen, this means the program will not treat this switch closure as a key depression (since the "up" bit remains set) and does not debounce this closure nor jump to decode a strap switch closure. Strap switches are of the on/off type not requiring debouncing as do the *momentary* on/off keyswitches. Also, a strap switch decode routine, in this example, is not necessary. The strap data bits read into RAM may be tested at any time for execution of a routine implementing the "mode" associated with a particular strap switch closure.

13. If the program jumps to "NRDY," a test is made to determine whether the digit position equals 1, indicating that all 14 digits have been displayed. If the last digit has not been displayed, the digit position is decremented by one and the program goes to "DSP2" to service the next digit. If the last digit has been displayed, the program falls through to "DEBOUN," the keyswitch debouncing portion of the program.

14. Debouncing begins at "DEBOUN" by testing to see whether the up bit has been reset, indicating a keyswitch closure. If not, the program takes the right branch to "ALLUP" and tests the not ready bit (NRB) of the KBC. If NRB is equal to 1, the KBC is decremented, the up bit remains set and the program goes back to "DSP1" to output data to all 14 digits again. If, on the first pass through the program, no key closure has occurred, the KBC will enter the debounce routine equal to 1111, exiting with a decremented value of 1110. Provided all keys remain up, it will take four passes through the right debounce branch before the KBC has been decremented to 1011, thereby resetting the not ready bit. If all keys remain up *after* four passes, the program will continue to fall through the NRB not equal to 1 (right) branch, keeping the KBC at 1011. The foregoing operations ensure that all keys remain up for at least four debounce passes before the not ready bit is reset to 0 (and a key closure will be accepted for *keydown*-debouncing).

15. If, upon entering the debounce routine, the up bit has been reset indicating a key closure, the program will take the left debounce branch. If the not ready bit has been reset to 0, indicating as explained above that all keys have previously remained up for at least four passes, the program will continue to decrement the KBC, exiting by setting the up bit and going back to "DSP2." Assuming that the right debounce branch has previously decremented the KBC to 1011, "DEBOUN" will be entered with the KBC equal to 0011. (A key closure resets the up bit.) If the key remains down for four passes, the left branch will decrement the KBC to 0000 and go back to "DSP1" with the KBC equal to 1000 (up bit reset). On the next pass, with the keyswitch still down, "KBCTST" will reset the up bit, the KBC will equal 0000 and the program will jump to the keyboard decode routine with the value of the current D line stored in RAM and the G port data in A.

    If the left branch of the debounce routine is entered without the keys having been up for at least four passes (NRB equal to 1), the program will set the KBC to 1111, continuing to do so until the key is lifted and remains up for four passes through the right branch of the debounce loop. Consequently, the program requires that a key be down, as well as up, for at least four debounce periods before keyboard data will be accepted and decoded. Since it takes 16 milliseconds to execute four program passes, ample time is provided to debounce even the most inexpensive keyboards.

16. Once a keyswitch closure has been debounced, the program exits to "KEYDEC" (keyboard decode routine). Upon entry to "KEYDEC," G port data is in the accumulator and represents the particular *row* of the keyboard matrix upon which a key closure has occurred. Data memory M(1,15) contains the value of the D line and represents the particular keyboard matrix *column* upon which a key closure has occurred. The conjunction of a particular D line value and the state of a particular G port bit, therefore, define one of sixteen key closures. Only two instructions are necessary to jump to the particular decode routine associated with each key closure based upon the contents of A and M(1,15): a COMP and a JID instruction.

    The COMP instruction is necessary to invert the contents of A since a particular key closure will result in one bit of G being driven to "0," with the remaining bits of G set to "1." Complementing A results in a "1" representing a key closure with the value of A equal to 0001, 0010, 0100, or 1000 (binary) if the key closure occurred on the $G_0$-$G_3$ row lines, respectively. D will equal 0001, 0010, 0011, or 0100 (binary) if the key closure occurred on the $D_1$-$D_4$ lines, respectively. The JID instruction can then use A and M without further manipulation to access key routine *pointers*, provided these pointers have been placed in appropriate ROM locations (those which the JID will access based upon the values of A and M associated with each key).

    The operation of the JID instruction is similar to that of the LQID instruction in that it accesses a ROM location based upon the current value of $P_9$, $P_8$, $A_3$, $A_2$, $A_1$, $A_0$, $M_3$, $M_2$, $M_1$, $M_0$. JID, however, then uses the contents of this ROM location as a pointer and transfers program control to this "pointed-to" address. The exact location of this address (first instruction of each decode routine) need not be of concern to the programmer provided it resides within the same ROM block as the JID instruction (see Section 4.1); in this example within ROM block 2 (pages 4-7).

    The location of each JID key decode routine pointer must correspond with the current value of $P_9$ and $P_8$, and with the value of A (G port data) and M (D line data) associated with each particular key closure. Table 5.3 depicts the various address values of $P_9$, $P_8$, A and M for each keyswitch closure. The programmer must place, *within these address locations*, the lower 8 bits of the address of the first instruction of each keydecode routine, to allow the JID instruction to automatically transfer program control to one of these instructions. This loading of ROM address pointers with the proper 8-bit data is easily accomplished using the assembler assignment statement and the .ADDR directive.

    First, the programmer must specify a label for the first instruction of each keyswitch decode routine — in this example labels "KEY1"-"KEY16" are given for the starting address of keyswitch number 1-16 decode routines, respectively. (No decode *servicing code* is given.) As already mentioned, these decode labels and the code for each decode routine must reside within the same ROM block as the JID instruction (ROM block 2, pages 4-7).

    Second, at each pointer address for each key closure as indicated in Table 5.3, an .ADDR directive must be used to place the lower 8 bits of the address of the beginning of each keyswitch decode routine within each pointer location. This is easily accomplished by moving the assembler location counter to the appropriate pointer address using an assignment statement which assigns the location counter (".") to the hexadecimal address of the appropriate JID pointer location. In this example, for instance, the "KEY1" pointer should be located at address X'111. The assignment statement, . = X'111, moves the assembler location counter to this address. The assembler will then generate code into successive memory locations starting at this location until the assembler location counter is again moved.

    After moving the assembler location counter to the proper JID pointer address, the 8-bit value of the address of each appropriate keyswitch decode label location is loaded into the pointer address by using an .ADDR directive with an operand specifying the *label* associated with the first instruction of each key decode routine. For example, to load the keyswitch number 1 decode routine starting address into its pointer location, an .ADDR KEY1 directive will place the lower 8 bits of the address of the KEY1 label into the ROM pointer location.

    As can be seen, once labels have been given to the beginning of each decode routine and the assembler location pointer has been moved to the proper JID pointer location, a simple .ADDR (label) statement for each label will automatically allow the JID instruction to transfer program control to the appropriate decode routine for each keyswitch immediately after exiting from the DISPLAY/KEYBOARD DEBOUNCE routine (after complementing G data as explained above). In this example, the assembler location pointer need only be moved four times, since each group of 4 JID pointers resides in successive memory locations. (See Table 5.3.)

    Of course, the gaps which exist between the JID pointer locations on pages 4-6 are available for use by other portions of program code. To aid the user in understanding the operations of the assignment statements and .ADDR directives in

this sample program, an *assembler output listing* of the program is provided in Figure 5.19, indicating in the leftmost columns the line numbers, memory addresses and 8-bit memory contents associated with the use of these assembler control statements.

For convenience, the "KEY1"-"KEY16" labels are placed in successive double-byte memory locations, jumping back to "DSP1." In a "real" program, each of these labels would be followed, respectively, by the code required to perform the program operations associated with each key closure. Alternatively, they might still be placed in successive double-byte memory locations if they used a JMP instruction to jump to any location within the 1K ROM area to a routine which serviced the appropriate keyswitch. For further information on the use of the PDS assembler, see Chapter 8, *PDS User's Manual.*

```
; COP420 DISPLAY/KEYBOARD DEBOUNCE/DECODE ROUTINE
; DISPLAYS 14 BCD DIGITS CONTAINED IN M(0,14) THROUGH M(0,1), HIGH-ORDER TO LOW-ORDER, RESPECTIVELY
; DECIMAL POINT POSITION VALUE CONTAINED IN M(0,15)
; DIGIT POSITION CONTAINED IN M(1,15)
; TEMPORARY STORAGE OF 4 BITS OF SEGMENT DATA IN M(1,14)
; KEYBOARD DEBOUNCE COUNTER (KBC) CONTAINED IN M(3,15)
; SEVEN-SEGMENT DECODE ROM LOOKUP DATA CONTAINED IN PAGE 4, WORDS 0 - F
; ROUTINE READS STRAP DATA SWITCHES TIED TO DIGIT LINES 12, 13 AND 14 INTO M(1,12) THROUGH M(1,14) RESPECTIVELY
; EXIT TO KEY DECODE ROUTINE AFTER DEBOUNCING KEYSWITCH CLOSURES WITH DIGIT VALUE IN M(1,15) AND G PORT DATA
;  IN A

            .PAGE     0
            DIGIT     =1,15         ; ASSIGN VALUE 1,15 TO "DIGIT"
            STORE     =1,14         ; ASSIGN VALUE 1,14 TO "STORE"
            KBC       =3,15         ; ASSIGN VALUE 1,13 TO "KBC"
            CLRA                    ; FIRST INSTRUCTION MUST BE A "CLRA"

DSPLY:      OGI       15            ; SET ALL G PORTS HIGH
            LBI       KBC           ; POINT TO M(3,15)
            STII      15            ; 15 TO KBC
DSP1:       SKT                     ; TIME-BASE COUNTER OVERFLOW?
            JP        NOCNT         ; NO COUNTER OVERFLOW
            JSR       TIMEKP        ; YES, CALL TIMEKEEPING SUBROUTINE
NOCNT:      LBI       0,14          ; START DISPLAY AT DIGIT 14
DSP2:       CBA                     ; DIGIT POSITION TO A
            XAD       DIGIT         ; STORE IN M(1,15)
            CLRA
            AISC      4             ; SET A2 TO FLIP TO PAGE 1 FOR LOOKUP
            LEI       0             ; BLANK SEGMENTS (RESET EN2)
            LQID                    ; LOOKUP TABLE SEGMENT DATA TO Q
            LBI       DIGIT         ; POINT TO DIGIT POSITION
            LD        1             ; DIGIT POSITION TO A, POINT TO DECIMAL POINT POSITION DIGIT

            SKE                     ; DECIMAL POINT = DIGIT POSITION?
            JMP       NODP          ; NO, RESET DECIMAL POINT BIT IN Q
            CLRA
            AISC      4
            JP        .-1           ; DELAY 9 INSTR. CYCLE TIMES
DIGOUT:     LBI       DIGIT         ; POINT TO DIGIT POSITION
            LD                      ; DIGIT POSITION TO A
            CAB                     ; DIGIT POSITION TO BD
            OBD                     ; OUTPUT DIGIT VALUE
            LEI       4             ; OUTPUT SEGMENT DATA (SET EN2)
            LBI       KBC           ; POINT TO KBC
            ING                     ; G PORTS TO A
            AISC      1             ; ALL G PORTS STILL HIGH (= 15)?
            JMP       KEYDWN        ; NO, JUMP TO "KEYDOWN" ROUTINE
            CLRA
            AISC      3             ; YES, DELAY 13 INSTR. CYCLE TIMES
            JP        .-1           ; BACK TO PREVIOUS INSTR. UNTIL SKIP
            LBI       KBC           ; POINT TO KBC
NRDY:       LDD       DIGIT         ; DIGIT POSITION TO A
            AISC      14            ; LAST DIGIT DONE (A = 1)?
            JMP       DEBOUN        ; YES, JUMP TO DEBOUNCE ROUTINE (A = 15)
            AISC      1             ; NO, DECREMENT DIGIT POSITION VALUE
            LBI       0,0           ; POINT TO DISPLAY REGISTER 0
            CAB                     ; DIGIT POSITION VALUE TO BD
```

**Figure 5.18 Display/Keyboard Interface Source Code**

```
            CLRA
            AISC    4               ; DELAY 9 INSTR. TIMES
            JP      .-1             ; REPEAT PREVIOUS INSTR. UNTIL SKIP
            JP      DSP2            ; DISPLAY NEXT DIGIT
            .PAGE   1
        ; WORDS 0 - F EQUAL SEVEN-SEGMENT DECODE LOOKUP DATA TABLE
        ; I(7) - I(0) = SA - SG, D.P. SENT UPON LOOKUP TO Q(7) - Q(0), RESPECTIVELY
        ; HEX VALUE FOR CHARACTERS 0 - 9, P, A, U, C, F, E PLACED IN SUCCESSIVE LOCATIONS BY ".WORD" DIRECTIVE
            .SPACE  5               ; LEAVE 5 BLANK LINES ON LISTING
            .WORD   X'FD            ; = 0 (SEVEN-SEGMENT DECODE HEX VALUES)
            .WORD   X'61            ; = 1
            .WORD   X'DB            ; = 2
            .WORD   X'F3            ; = 3
            .WORD   X'67            ; = 4
            .WORD   X'B7            ; = 5
            .WORD   X'BF            ; = 6
            .WORD   X'E1            ; = 7
            .WORD   X'FF            ; = 8
            .WORD   X'E7            ; = 9
            .WORD   X'CF            ; = P
            .WORD   X'EF            ; = A
            .WORD   X'7F            ; = U
            .WORD   X'90            ; = C
            .WORD   X'8F            ; = F
            .WORD   X'9F            ; = E
DEBOUN:     SKMBZ   3               ; UP BIT = 1?
            JP      ALLUP           ; YES
            SKMBZ   2               ; NO, NRB = 1?
            JP      STR             ; YES, A = 15 SO STORE IT IN KBC
DECKBC:     ADD                     ; DECREMENT KBC
STR:        X                       ; PLACE A IN KBC
            SMB     3               ; SET UP BIT OF KBC
            JMP     DSP1            ; DO DISPLAY LOOP OVER AGAIN
ALLUP:      SKMBZ   2               ; NRB = 1?
            JP      DECKBC          ; YES, DECREMENT KBC (A = 15)
            AISC    4               ; NO, SET KBC = 11
            NOP                     ; DEFEAT "AISC" SKIP
            JP      STR
KEYDWN:     LDD     DIGIT           ; DIGIT POSITION TO A
            AISC    4               ; DIGIT POSITION > 11 (STRAP DATA)?
            JP      KBCTST          ; NO, TEST KBC
            AISC    12              ; YES, RESTORE STRAP DIGIT VALUE
            CAB                     ; STRAP DIGIT POSITION TO BD
            CLRA
            AISC    1
            XABR                    ; 1 TO BR (POINT TO STRAP DATA REG. 1)
            ING                     ; STRAP DATATO A
            X                       ; PLACE IN APPROPRIATE DIGIT, REG. 1
            JMP     NRDY
KBCTST:     RMB     3               ; RESET UP BIT OF KBC
            CLRA
            AISC    8               ; DELAY 5 INSTR. CYCLE TIMES
            JP      .-1             ; REPEAT PREVIOUS INSTR. UNTIL SKIP
            CLRA                    ; 0 TO A
            SKE                     ; KBC = 0?
            JMP     NRDY            ; NO
            LEI     0               ; YES, BLANK SEGMENTS
            ING                     ; G PORTS TO A
            LBI     DIGIT           ; POINT TO DIGIT NUMBER
            JMP     KEYDEC          ; JUMP TO KEY DECODE ROUTINE
            .FORM                   ; FORM FEED
            .PAGE   2               ; SUBROUTINE PAGE 2 CODE
            .LIST   X'31            ; FULL MASTER LIST AND LIST OF INCLUDED "TIMEKP" CODE
            .INCLUD TIMEKP          ; INCLUDE "TIMEKP" SUBROUTINE CODE
            .PAGE   4
; FOLLOWING CODE USES CONTENTS OF A AND M, KEYSWITCH COLUMN AND ROW CLOSURE DATA, RESPECTIVELY, ON EXIT
; FROM DISPLAY ROUTINE, TO ACCESS ROM POINTERS TO JUMP TO KEY1 - KEY16 DECODE ROUTINES
; LABELS "KEY1" THROUGH "KEY16" MUST BE LOCATED WITHIN PAGES 4 THROUGH 7
            .SPACE  5               ; FIVE BLANK LINES ON LISTING
KEYDEC:     COMP                    ; COMPLEMENT A SO THAT BIT = 1 INDICATES KEY CLOSURE
```

**Figure 5.18 Display/Keyboard Interface Source Code (continued)**

```
        JID                         ; JUMP TO KEY DECODE ROUTINE FOR PARTICULAR KEY CLOSURE
        .=          X'111           ; MOVE ASSEMBLER LOCATION COUNTER TO KEY1 ROM POINTER ADDRESS

        .ADDR       KEY1            ; PLACE KEY1 POINTER IN ADDRESS X'111
        .ADDR       KEY2            ; PLACE KEY2 - KEY4 POINTERS IN NEXT ROM LOCATIONS
        .ADDR       KEY3
        .ADDR       KEY4

        .=          X'121           ; MOVE TO KEY5 POINTER LOCATION
        .ADDR       KEY5
        .ADDR       KEY6
        .ADDR       KEY7
        .ADDR       KEY8
        .=          X'141           ; MOVE TO KEY9 POINTER LOCATION (PAGE 5)
        .ADDR       KEY9
        .ADDR       KEY10
        .ADDR       KEY11
        .ADDR       KEY12
        .=          X'181           ; MOVE TO KEY13 POINTER LOCATION (PAGE 6)
        .ADDR       KEY13
        .ADDR       KEY14
        .ADDR       KEY15
        .ADDR       KEY16

NODP:   LBI         STORE           ; POINT TO M(2,15)
        CQMA                        ; SE - SG, D.P. TO A
        X                           ; EXCHANGE INTO M(2,15)
        RMB         0               ; RESET D.P. BIT (DECIMAL POINT OFF)
        CAMQ                        ; SEGMENT DATA BACK TO Q
        JMP         DIGOUT
```

**Figure 5.18 Display/Keyboard Interface Source Code (continued)**



**Table 5.3 JID Pointer Table for Display/Keyboard Routine**

| Key No. | Value at Time of JID P9 | P8 | Rows G3 A3 | G2 A2 | G1 A1 | G0 A0 | Keyboard Columns D3 M3 | D2 M2 | D1 M1 | D0 M0 | JID Pointer Hex (X') Address | JID Pointer Hex Contents |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | X'111 | 85 |
| 2 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | X'112 | 87 |
| 3 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | X'113 | 89 |
| 4 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | X'114 | 8B |
| 5 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | X'121 | 8D |
| 6 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | X'122 | 8F |
| 7 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | X'123 | 91 |
| 8 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | X'124 | 93 |
| 9 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | X'131 | 95 |
| 10 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | X'132 | 97 |
| 11 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | X'133 | 99 |
| 12 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | X'134 | 9B |
| 13 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | X'141 | 9D |
| 14 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | X'142 | 9F |
| 15 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | X'143 | A1 |
| 16 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | X'144 | A3 |

```
COP CROSS ASSEMBLER
COP420  DISPLAY

  127                           .FORM                   ; FORM FEED

  125           0100            .PAGE   4
  126                                                   ; FOLLOWING CODE USES CONTENTS OF A AND M, KEYSWITCH
  127                                                   ; COLUMN AND ROW CLOSURE DATA, RESPECTIVELY, ON EXIT FROM
  128                                                   ; DISPLAY ROUTINE, TO ACCESS ROM POINTERS TO JUMP TO
  129                                                   ; KEY1 - KEY16 DECODE ROUTINES.  LABELS "KEY1" THROUGH
  130                                                   ; "KEY16" MUST BE LOCATED WITHIN PAGES 4 THROUGH 7.
  131           0005            .SPACE  5               ; FIVE BLANK LINES ON LISTING


  132   100     40      KEYDEC: COMP                    ; COMPLEMENT A SO THAT BIT = 1 INDICATES KEY
  133                                                   ; CLOSURE
  134   101     FF              JID                     ; JUMP TO KEY DECODE ROUTINE FOR PARTICULAR
  135                                                   ; CLOSURE
  136           0111            =       X'111           ; MOVE ASSEMBLER LOCATION COUNTER TO KEY1 ROM
  137                                                   ; POINTER ADDRESS
  138   111     85              .ADDR   KEY1            ; PLACE KEY1 POINTER IN ADDRESS
  139   112     87              .ADDR   KEY2            ; PLACE KEY2 - KEY4 POINTERS IN ROM LOCATIONS
  140   113     89              .ADDR   KEY3
  141   114     8B              .ADDR   KEY4
  142           0121            . =     X'121           ; MOVE TO KEY5 POINTER LOCATION
  143   121     8D              .ADDR   KEY5
  144   122     8F              .ADDR   KEY6
  145   123     91              .ADDR   KEY7
  146   124     93              .ADDR   KEY8
  147           0141            . =     X'141           ; MOVE TO KEY9 POINTER LOCATION (PAGE 5)
  148   141     95              .ADDR   KEY9
  149   142     97              .ADDR   KEY10
  150   143     99              .ADDR   KEY11
  151   144     9B              .ADDR   KEY12
  152           0181            =       X'181           ; MOVE TO KEY13 POINTER LOCATION (PAGE 6)
  153   181     9D              .ADDR   KEY13
  154   182     9F              .ADDR   KEY14
  155   183     A1              .ADDR   KEY15
  156   184     A3              .ADDR   KEY16
  157   185     6002    KEY1:   JMP     DSPLY   ; G0, D1 KEY
  158   187     6002    KEY2:   JMP     DSPLY   ; G0, D2 KEY
  159   189     6002    KEY3:   JMP     DSPLY   ; G0, D3 KEY
  160   18B     6002    KEY4:   JMP     DSPLY   ; G0, D4 KEY
  161   18D     6002    KEY5:   JMP     DSPLY   ; G1, D1 KEY
  162   18F     6002    KEY6:   JMP     DSPLY   ; G1, D2 KEY
  163   191     6002    KEY7:   JMP     DSPLY   ; G1, D3 KEY
  164   193     6002    KEY8:   JMP     DSPLY   ; G1, D4 KEY
  165   195     6002    KEY9:   JMP     DSPLY   ; G2, D1 KEY
  166   197     6002    KEY10:  JMP     DSPLY   ; G2, D2 KEY
  167   199     6002    KEY11:  JMP     DSPLY   ; G2, D3 KEY
  168   19B     6002    KEY12:  JMP     DSPLY   ; G2, D4 KEY
  169   19D     6002    KEY13:  JMP     DSPLY   ; G3, D1 KEY
  170   19F     6002    KEY14:  JMP     DSPLY   ; G3, D2 KEY
  171   1A1     6002    KEY15:  JMP     DSPLY   ; G3, D3 KEY
  172   1A3     6002    KEY16:  JMP     DSPLY   ; G3, D4 KEY
  173   1A5     1D      NODP:   LBI     STORE           ; POINT TO M(2,15)
  174   1A6     332C            CQMA                    ; SE - SG, D.P. TO A
  175   1A8     06              X                       ; EXCHANGE INTO M(2,15)
  176   1A9     4C              RMB     0               ; RESET D.P. BIT (DECIMAL POINT)
  177   1AA     333C            CAMQ                    ; SEGMENT DATA BACK TO Q
  178   1AC     6019            JMP     DIGOUT
  179                           .END
```

**Figure 5.19 Key Decode Routine — Output Listing**

## 5.4 SIO Input/Output

SI and SO can be used to provide additional I/O capability for the COP400 family by connecting, for example, external 8-bit parallel-to-serial (MM74C165) and serial-to-parallel (MM74C164) shift registers, as shown in Figure 5.20. The following routine will output 8 bits of data serially using the SIO registers, at the same time inputting 8 bits serially. Data is output from and input to A and M. This program must be entered with the SIO register enabled as a serial shift register. The execution of an XAS instruction with C = "1" and "0" respectively will enable and disable SK as a SYNC output. (See Section 3.2, LEI instruction description.) With SK enabled as a SYNC output it will provide a clock pulse to the shift registers each instruction cycle time. Note that SI is simultaneously shifting 1 bit of serial data into SIO while SO is shifting 1 bit of serial data out. Since the 4-bit contents of SIO are continuously shifted each instruction cycle time, the routine is written to insure that SIO is exchanged with A every 4 instruction cycle times.

```
; ROUTINE TO OUTPUT 8 BITS OF DATA SERIALLY FROM M
; AND A WHILE INPUTTING 8 BITS OF SERIAL DATA INTO M
; AND A USING THE SIO REGISTER
; UPON ENTRY, SIO MUST BE ENABLED AS A SERIAL SHIFT
; REGISTER (EN0 = 0)

SERIO:
        SC      ; SET CARRY TO ENABLE SK AS A SYNC
                ; OUTPUT
        XAS     ; START SYNC, A TO SIO, START SHIFTING
                ; A OUT, SI DATA IN
        NOP     ; WAIT 4 INSTR. CYCLE TIMES
        NOP
        LD      ; M TO A
        XAS     ; FIRST 4 SI BITS TO A, A TO SIO,
                ; CONTINUE SHIFTING SI IN, SO OUT
        X       ; STORE FIRST 4 SI BITS IN M
        CLRA    ; CLEAR A (WAIT 4 INSTR. CYCLE TIMES)
        RC      ; RESET C TO DISABLE SK AS A SYNC
                ; OUTPUT
        XAS     ; STOP SYNC, LAST 4 SI BITS TO A
```

Figure 5.21 shows an example of a multi-COP420 system. As is indicated, data transfers between the two devices are done in a serial fashion, with one COP providing a SYNC pulse via the SK output to the CKO pin of the second COP. To ensure the validity of the data being transferred, both COPs must contain a routine which will synchronize the inputting and outputting of data between the two devices using the SIO register. The following code accomplishes this by providing that each COP receive and send a string of four "1s" (SIO = $1111_2$) before an SIO data transfer is effected.

```
; ROUTINE TO SYNCHRONIZE SERIAL DATA TRANSFERS
; BETWEEN TWO COP DEVICES (COPA AND COPB) USING
; THE SIO REGISTER
; SIO MUST HAVE BEEN PREVIOUSLY ENABLED AS A SERIAL
; SHIFT REGISTER

; COPA CODE:

BACK:   NOP             ; ADD 1 INSTR. CYCLE TIME FOR
                        ; RE-SYNC
        CLRA            ; ZERO TO A
        XAS             ; OUTPUT ZEROS, WAIT 4 INSTR.
        NOP             ; CYCLE TIMES
        CLRA
        COMP            ; 15 TO A
        XAS             ; OUTPUT 15 VIA SK, SI BITS TO A
        AISC    1       ; ARE INPUT BITS = 15?
        JP      BACK    ; NO, TRY AGAIN
        .               ; YES, DEVICES SYNCHRONIZED
        .
        .

; COPB CODE:

BACK:
        CLRA            ; OUTPUT ZEROS IN 4-CYCLE
                        ; LOOP
        XAS
        AISC    1       ; 15 FROM COPA?
        JP      BACK    ; NO, KEEP SENDING OUT ZEROS
        COMP            ; YES, OUTPUT 15 TO COPA
        XAS
        NOP             ; DEVICES SYNCHRONIZED
        NOP
        NOP             ; WAIT FOR COPA TO START
```

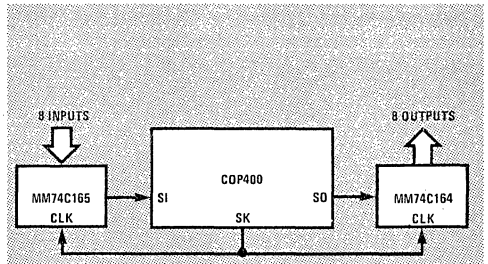



Figure 5.20 Additional I/O Using SI & SO

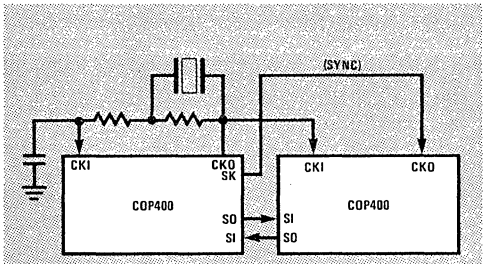



Figure 5.21 Multi-COP400 System

## 5.5 Add-On RAM

The following routine will interface the COP420 to an additional 2K bits (512 × 4) of RAM. The interconnect diagram (see Figure 5.22) shows the COP420 interfaced to two additional MM2112 (256 × 4) RAM devices, although CMOS equivalents (MM74C921s) may also be used where lower power consumption or RAM battery backup is desired. Up to four devices may be used by decoding the $D_0$ and $D_1$ lines (2-to-4 binary decoder). If all 4 bits of D are used, up to 16 additional RAM devices can be interfaced utilizing a 4-to-16 binary decoder (an additional 2K *bytes* of RAM).

The following routine treats the 1024 bits of external RAM as organized as 16 registers of 16 4-bit digits. It sequentially addresses digits 0 through 15 in a particular external RAM register (as determined by the 4-bit contents of *COP* RAM memory digit M(3,15). It then reads from or write I/O data into COP RAM memory, register 0, digits 0-15, respectively.

Note that two different operands for the LEI instruction are used to select or de-select specific operations associated with three of the four bits of the EN register. The LEI 13 instruction sets $EN_3$-$EN_0$ equal to 1101 with the result that $EN_3$ and $EN_0$ are equal to "1" and, therefore, SO will output a "1" to the WE pins of external RAM to perform a read operation. $EN_2$ is also set to "1" to enable the L drivers so that Q latch data will be output to the L I/O ports and, via the interconnect, to the RAM address lines. The LEI 5 instruction alters $EN_3$ to "0," resulting in SO being driven low, enabling a write operation into the external RAM device.

```
; SUBROUTINE TO READ FROM/WRITE TO ONE OF TWO EXTERNAL RAM DEVICES (256 x 4 BITS EACH)
; 16 4-BIT DIGITS OF I/O DATA READ FROM OR WRITTEN INTO COP RAM, REGISTER 0, DIGITS 0 - 15
; C = 0 INDICATES A READ OPERATION, C = 1 INDICATES A WRITE OPERATION
; 8-BIT RAM ADDRESS SPECIFIED BY A 4-BIT REGISTER NUMBER CONTAINED IN M(3,15), ASSIGNED TO SYMBOL "DIGIT"
; CHIP-SELECT NUMBER (1110 OR 1101 BINARY) CONTAINED IN M(2,15), ASSIGNED TO SYMBOL "CSEL"
; READ:  ENTRY POINT TO READ RAM
; WRITE:  ENTRY POINT TO WRITE RAM

        DIGIT   =1,15
        CSEL    =2,15
        REG     =3,15

READ:
        RC                      ; RESET CARRY FOR READ OPERATION
        JP      RW
WRITE:
        SC                      ; SET CARRY FOR WRITE OPERATION
RW:                             ; READ/WRITE CODE
        OGI     15              ; SET G3 - G0 HIGH
        LEI     13              ; SO = 1, ENABLE L DRIVERS
        LBI     CSEL
        OBD                     ; OUTPUT CHIP SELECT VALUE
        LBI     DIGIT           ; POINT TO DIGIT NUMBER
        CLRA                    ; START WITH DIGIT 0
RWL:
        X                       ; EXCHANGE A INTO DIGIT NUMBER IN M
        LDD     REG             ; REGISTER NUMBER TO A
        CAMQ                    ; OUTPUT REGISTER AND DIGIT NUMBER FOR RAM ADDRESS
        LD      1               ; DIGIT NUMBER TO A, POINT TO REGISTER 0
        CAB                     ; DIGIT NUMBER TO BD TO POINT TO I/O DATA IN M
        SKC                     ; IS CARRY EQUAL TO 1?
        JP      RR              ; NO, JUMP TO READ RAM
        LEI     5               ; YES, PERFORM WRITE OPERATION, DRIVE WRITE ENABLE LOW
        OMG                     ; OUTPUT DATA TO RAM
        LEI     13              ; SET WRITE ENABLE HIGH
        OGI     15              ; SET G3 - G0 HIGH
RWCONT:
        LBI     DIGIT           ; POINT TO DIGIT NUMBER
        LD                      ; DIGIT NUMBER TO A
        AISC    1               ; INCREMENT DIGIT NUMBER, IS DIGIT = 15?
        JP      RWL             ; NO, CONTINUE READ/WRITE
        OBD                     ; YES, DISABLE RAMS (CHIP SELECTS HIGH)
        RET                     ; RETURN
RR:
        ING                     ; READ RAM DATA
        X                       ; STORE IN I/O DIGIT IN M
        JP      RWCONT          ; CONTINUE
```

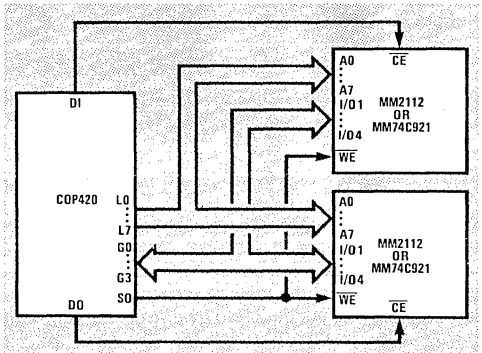



Figure 5.22 Typical Add-On RAM Interconnect

### 5.6 $IN_3/IN_0$ Inputs

Section 4.8 has already provided an example of an interrupt service routine utilizing the "hardware" interrupt capability of the $IN_1$ COP420 pin. It is also possible to implement a "software" interrupt, using either the COP420 $IN_3$ or $IN_0$ inputs, since they have testable input latches associated with them. These latches, $IL_3$ and $IL_0$, will be set if a low going pulse, at least two instruction cycles wide, has occurred on the $IN_3$ or $IN_0$ inputs, respectively. The INIL instruction inputs these latches to A, as explained in Section 3.2, to allow them to be tested as software interrupt flags ($A_3$ and $A_0$).

To accomplish a software interrupt, an INIL instruction must be executed often enough to respond to the requirements of the interrupt signal tied to $IN_3$ or $IN_0$. For example, in timekeeping applications, $IN_3$ or $IN_0$ may be connected to a 60 Hz square wave. The program must, in this case, execute an INIL instruction at least every 1/60 second.

If an interrupt input occurs irregularly, it will be more efficient to connect it to the hardware interrupt pin, $IN_1$, to insure that no interrupt is missed due to infrequent testing. Conversely, if an interrupt input occurs regularly and predictably (such as a 60 Hz signal) a software interrupt may be efficiently utilized by simply building into the program a sufficient test rate to insure that no inputs are missed.

---

### Technical Assistance

National Semiconductor will be pleased to provide technical assistance to aid a user in design and development. Inquiries may be directed to any of our Field Applications Engineers (FAEs) — located in every National sales office — or to our in-plant COPS™ Applications Group at (408) 737-5582.

---

# Analog to Digital Conversion Techniques With COPS™ Family Microcontrollers

National Semiconductor
Leonard A. Distaso
February 1980
COP Note 1

## Table of Contents

I. Introduction ..... 2-84
II. Simple Capacitor Charge Time Measurement ..... 2-84
A. Basic Approach ..... 2-84
B. Accuracy Improvements ..... 2-87
C. Conclusions ..... 2-88
III. Pulse Width Modulation (Duty Cycle) Technique ..... 2-89
A. Mathematical Analysis ..... 2-89
B. Basic Implementation ..... 2-90
C. Accuracy Improvements ..... 2-93
IV. Dual Slope Integration Techniques ..... 2-97
A. Mathematical Background ..... 2-97
B. Basic Dual Slope Technique ..... 2-98
C. Modified Dual Slope Technique ..... 2-100
V. Voltage to Frequency Converters, VCO's ..... 2-103
A. Basic Approach ..... 2-103
B. The LM131/LM231/LM331 ..... 2-105
C. Voltage Controlled Oscillators ..... 2-105
D. A Combined Approach ..... 2-105
VI. Successive Approximation ..... 2-107
A. Basic Approach ..... 2-107
B. Some Comments on Resistor Ladders ..... 2-109
VII. "Offboard" Techniques ..... 2-112
A. General Comments ..... 2-112
B. ADC0800 Interface ..... 2-112
C. Naked-8™ Interface ..... 2-113
D. The MM5407 as an A/D Converter ..... 2-115
VIII. Conclusion ..... 2-118
IX. References ..... 2-118

## I. Introduction

A variety of techniques for performing analog to digital conversion are presented. The COP420 microcontroller is used as the control element in all cases. However, any of the COPS™ family of microcontrollers could be used with only minor changes in some component values to allow for different instruction cycle times.

All indirect analog to digital converters are composed of three basic building blocks:

- D/A Converter
- Comparator
- Control logic

In a software driven system the D/A converter and comparator are present but the control logic is replaced by instruction sequences. There are a variety of software/hardware techniques for implementing A/D converters. They differ primarily in their approach to the included D/A. There are two primary approaches to the digital to analog conversion which can in turn be divided into a number of subcategories:

- D/A as a function of weighted closures
  - — R/2R ladder
  - — Binary weighted ladder
- D/A as a function of time
  - — RC exponential charge
  - — Linear charge/discharge (dual slope)
  - — Pulse width modulation

These techniques should be generally familiar to persons skilled in the electronic art. The objective here is to illustrate the application of these established methods to a low cost system with a COPS microcontroller as the intelligent control element. Circuit configurations are provided as well as the appropriate flow charts and code to implement the function.

Some mathematical and theoretical analysis is presented as an aid to understanding the various techniques and their limits. However, it is not the purpose here to provide a definitive theoretical analysis of the analog to digital conversion process or of the various techniques described.

## II. Simple Capacitor Charge Time Measurement

### A. BASIC APPROACH

#### A.1 General

Perhaps the simplest means to perform an analog to digital conversion is to charge a capacitor until the capacitor voltage is equal to the unknown voltage. The capacitor voltage and the unknown are compared by means of a standard analog comparator. The unknown is determined simply by counting, in the microcontroller, the amount of time it takes for the charge on the capacitor to reach a value equal to the unknown voltage. The capacitor voltage is given by the standard capacitor charge equation:

$$V_C = V0 + [V1 - V0][1 - e^{**}(-t/RC)]$$

where: $V_C$ = capacitor voltage
V0 = "discharge voltage" — low level voltage
V1 = high level voltage

The most obvious problem with this method, from the standpoint of software implementation, is the nonlinearity of the relationship. This can be circumvented in

several ways. First of all, a routine to calculate the exponential can be implemented. This, however, usually requires too much code if the exponential routine is not otherwise required in the program. Alternatively, the range of input voltages can be restricted so that only a portion of the capacitor charge curve — which can be approximated with a linear relationship or with some minor straight line curve fitting — is used. Finally, a look up table can be used which will effectively convert the measured time to the appropriate voltage. The look up table has the advantage that all the math can be built into the table, thereby simplifying matters significantly. If arithmetic routines are going to be used, it is clear that the relationship is simplified if V0 is 0 volts because it then drops out the equation.

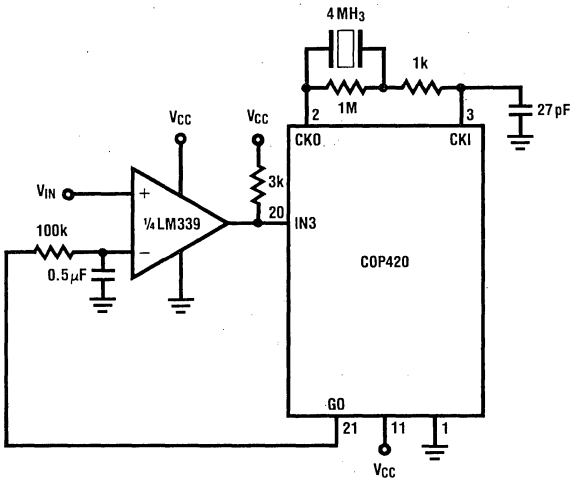
### A.2. Basic Circuit Implementation

The circuit in Figure 1 is the basic implementation of the capacitor charge method of A/D conversion. The selection of input and output used is arbitrary and is dictated by general system considerations. V0 is the "0" level of the G output and V1 is the "1" level of the output. The technique is basically to discharge the capacitor to V0 (which is ideally ground) and then to apply V1 and increment an internal counter until the comparator changes state. The flow chart and code for this implementation are shown in Figure 2.

### A.3 Accuracy Considerations

The levels reached by the microcontroller output constitute one of the more significant problems with this basic implementation. The levels of V1 and V0 are not $V_{CC}$ and ground as would be desired. The level is defined by the load on the output, the value of $V_{CC}$, and the device itself. Furthermore, these levels are likely to change from device to device and over temperature. To be sure, the output values will be at least those given in the data sheet, but it must be remembered that those values are minimum high voltages and maximum low voltages. Typically, the high value will be greater than the spec minimum and the low value will be lower than the spec maximum. In fact, with a light load the values will be close to $V_{CC}$ and ground. Therefore, in order to obtain any accurate result for a voltage measurement the exact values of V1 and V0 need to be measured and somehow stored in the microcontroller. Typical values of these voltages can be measured experimentally and an average could be used for a final implementation.

The other problem associated with the levels is that the capacitive load on the output line is substantial and far in excess of the values used when specifying the characteristics of the various COP420 outputs. The significant effect of this is that it will take longer than "normal" for the output to reach its maximum value. In addition, it is likely that there will be dips in the output as it rises to its maximum value since the capacitor will start to draw charging current from the output. All of this will be fast relative to the other system times. Still, it will affect the result since the level to which the capacitor is attempting to charge is not being applied uniformly and "instantaneously". It can be viewed as though the voltage V1 is bouncing before it stabilizes.




CRYSTAL OSCILLATOR VALUES CHOSEN TO GIVE 4 µS CYCLE TIME WITH DIVIDE BY 16 OPTION SELECTED ON COP 420 CKO/CKI PINS

$V_{CC} = +5V$

**Figure 1. Basic Capacitor Charge Technique**

```
                OGI     0       ;TURN OFF G TO DISCHARGE CAPACITOR
                ; INSERT SOME DELAY TO MAKE SURE CAPACITOR DISCHARGED
                ;USING 12 BIT COUNTER, BUT ONLY UPPER 8 USED IN TABLE
                ;LOOK UP DUE TO ACCURACY OF RC CHARGE METHOD.  THE OTHER
                ;BITS COULD BE USED BUT THE COMPLICATIONS ARE NOT WORTH
                ;THE EFFORT FOR THIS PARTICULAR TECHNIQUE.  ALSO, HERE THE
                ;INPUT RANGE IS RESTRICTED SO THAT THE TOP 3 BITS ARE ZERO

RCAD:   OGI     1       ;TURN ON THE G LINE
INCR:   LBI     2,13    ;BINARY INCREMENT OF 12 BIT COUNTER
BINPLS: SC              ;LOWER FOUR BITS WILL BE DISCARDED
BINPLI: CLRA            ;ONLY TOP BITS USED IN TABLE LOOK UP
        ASC             ;SPEED WOULD BE IMPROVED IF THE ADD WERE
        NOP             ;STRAIGHT LINE CODED-BUT COSTS MORE CODE
        XIS
        JP      BINPLI
        ININ            ;READ IN3 TO SEE IF COMPARATOR CHANGED
        AISC    8
        JP      END
        CLRA
        JP      INCR
END:    OGI     0       ;TURN OFF THE G LINE AND DISCHARGE C
        ;DO ARITHMETIC HERE OR LOOK UP TABLE OR WHATEVER IS
        ;REQUIRED--SAMPLE LOOK UP TABLE CONTROL INDICATED BELOW
        ;SAMPLE TABLE WRITTEN CORRECTING FOR THE EXPONENTIAL
        ;RELATIONSHIP.  THE TABLE ALSO INCORPORATES A CONVERSION
        ;TO BCD.  THE VALUE IN THE TABLE IS THE RATIO OF
        ;THE CAPACITOR VOLTAGE V TO THE MAXIMUM VOLTAGE VMAX.
        ;THE NUMBER IS A TWO DIGIT BCD FRACTION.  WE ARE USING
        ;A 5 BIT COUNT IN THIS EXAMPLE.  ADDRESSING ARBITRARILY
        ;SET UP ASSUMING THAT CONTROL CODE IS IN PAGE 0 (OTHER
        ;THAN AT ADDRESS 0) AND THAT THE TABLE THEREFORE IS IN
        ;PAGE 1 (STARTING AT HEX ADDRESS 040).
        ;
        LBI     2,15    ;POINT TO TOP 4 BITS
        XDS             ;TOP 4 IN A,POINTING TO LOWER 4 IN 2,14
        AISC    4       ;THIS MERELY ADJUSTING FOR ADDRESS--NO
                        ;OTHER FUNCTION
        LQID            ;DO THE LOOK UP
        CQMA            ;FETCH THE ADJUSTED VALUE FROM Q
         ; THE ADJUSTED VALUE IS NOW IN A AND M.  FROM THIS POINT MAY
         ;USE THE VALUE IN OTHER CALCULATIONS OR OUTPUT THE INFORMATION,
         ;OR WHATEVER MAY BE REQUIRED BY THE APPLICATION.
        LBI     2,13    ;CLEAR THE COUNTER
        STII    0
        STII    0
        STII    0
        JP      RCAD:   ;JUMP BACK AND REPEAT

        .=X'040         ;SET UP TABLE ADDRESS
        .WORD 000,003,006,008   ;SET UP THE TABLE VALUES
        .WORD 011,014,016,019   ;HERE, COMPENSATED FOR EXPONENTIAL
        .WORD 021,023,026,028   ;AND CONVERTED TO BCD FRACTION
        .WORD 030,032,034,036   ;TABLE VALUE IS RATIO V/VMAX
        .WORD 038,039,041,043
        .WORD 045,046,048,049
        .WORD 051,052,053,055
        .WORD 056,057,059,060
```

**Figure 2A. Typical RC Charge A/D Code**

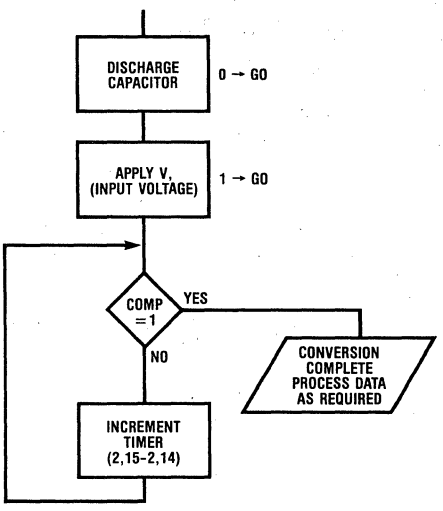



**Figure 2B. RC Charge Flow Chart**

A more general problem is that of the tolerance of RC time constant. The value of the voltage with respect to time is obviously related to the RC value. Therefore, a change in that value will result in a change in the voltage for a given time period t. The graph in Figure 3 illustrates the effect of a ±10% variation in the RC value upon the voltage measured for a given time t. If one cares to work out the math, it comes out that the error is an exponential relationship in much the same manner as the capacitor voltage itself. The maximum error induced for ±10% RC variation is ±3.9%.

Remember also that we are measuring time. Therefore variation in the RC value will have a direct, linear effect on the time required to measure a given voltage. It is also necessary that the time base for the COP420 be accurate. A variation in the accuracy in the operating frequency of the COP420 will have a direct impact on the accuracy of the result.

Given the errors mentioned so far and assuming that no changes are made in the hardware, the accuracy of the technique then is determined by the resolution of the time measurement. This is improved in two ways: increase the RC time constant so that there is a smaller change in capacitor voltage for a given time period or try to minimize the loop time required to increment the counter. Lengthening the RC time constant is easier but the cost is increased conversion time. The minimum time to increment a 5 to 8 bit binary counter and test an input is 13 cycle times. For a 9 to 12 bit binary counter this minimum time is 17 cycle times. Note also that the minimum time to perform the function does not necessarily correspond to the minimum number of code words required to implement the function. At a cycle time of 4 microseconds, the 13 cycle times correspond to 52 microseconds.

## B. ACCURACY IMPROVEMENTS

Several options are available if it is desired to improve the accuracy of this method. Three such improvements are shown in Figure 4. Figure 4A is the smallest change. Here a pullup resistor has been added to the G output line and the G line is run open drain internally, i.e., the internal pullup is removed. This improves the "bounce" problem mentioned earlier. The G line will go to the high state and remain there with this setup. However, the addition of the resistor does little more than eliminate the bounce. The degree of improvement is not great, but it is an easy way to eliminate a minor source of error.

Figure 4B is the next step. A 74C04 is used as a buffer. The 74C04 was chosen because of its symmetric output characteristics. Any CMOS gate with such characteristics could be used. The software can easily be adjusted to provide the proper polarity. The COP420 output drives a CMOS gate which in turn drives the RC network. This change does make significant improvements in accuracy. With a light load the CMOS gate will typically

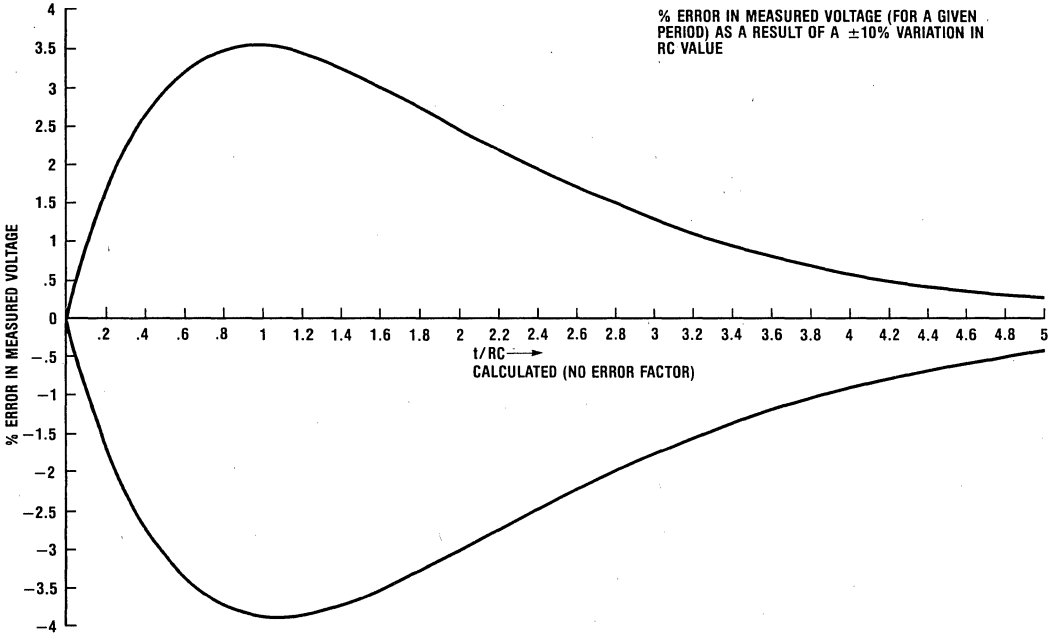



Figure 3

swing from ground to $V_{CC}$ and its output level is not as likely to be affected by the capacitor discharge.

Figure 4C is the best approach, but it involves the greatest component cost. Here two G outputs are controlling analog switches. Ground is connected to the RC network to discharge the capacitor, and a positive reference is used to charge the capacitor. This reference can be any suitable voltage source: zener diodes, $V_{CC}$, etc. The controlling voltage tolerance is now clearly the tolerance of the reference. Precise voltage references are readily obtainable. Figure 4C also shows an analog switch connected directly across the capacitor to speed up the capacitor discharge time. When using this version of the basic scheme, remember to include the 'on' resistance of the analog switch connected to $V_{REF}$ in the RC calculation. Failure to do so will introduce error into the result.

Note that the LM339 is a quad comparator. If these comparators are not otherwise needed in the system, they can be used in much the same manner as the CMOS gate mentioned above. They can be used to buffer the output of the COPS™ device and to reset the capacitor, or whatever other function is required. This has the advantage of fully utilizing the components in the system and eliminates the need to add another package to the system.

## C. CONCLUSIONS

This approach is an inexpensive way to perform an A/D conversion. However, it is not that accurate. With a 10% $V_{CC}$ supply and a 10% tolerance in the RC value and 10% variation in the oscillator frequency the best that can be hoped for is about 25% accuracy. If a 1% reference voltage is used, this accuracy becomes about 15%.

Under laboratory conditions — holding all variables constant and using precise measured values in the calculations — the configuration of Figure 2 yielded 5 bit accuracy over an input range of 0 to 3.5 volts. Over the same range and under the same conditions, the circuit of Figure 4B yield 7 to 8 bit accuracy. It must be emphasized that these accuracies were obtained under controlled conditions. All variables were held constant and actual measured values were used in all calculations. It is unlikely that the general situation will yield these accuracies unless adjustments are provided and a calibration procedure is used. This could defeat the low cost objective.

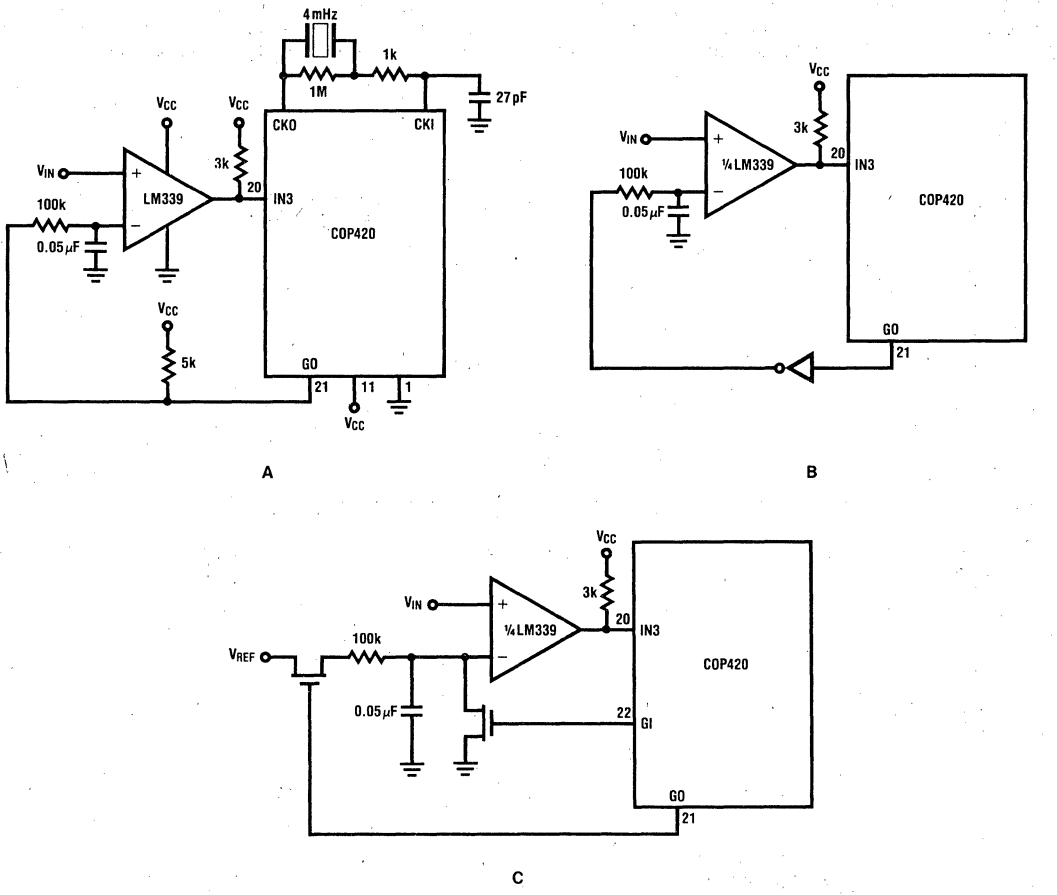



**Figure 4**

## III. Pulse Width Modulation (Duty Cycle) Technique

### A. MATHEMATICAL ANALYSIS

The pulse width modulation, or duty cycle, conversion technique is based on the fact that if a repetitive pulse waveform is applied to an RC network, the capacitor will charge to the average voltage of the waveform provided that the RC time constant is sufficiently large relative to the pulse period. See Figure 5.

In this technique, the capacitor voltage $V_C$ is compared to the voltage to be measured by means of an analog comparator. The duty cycle is then adjusted to cause $V_C$ to approach the input voltage. The COPS™ device reads the comparator output and then drives one of its outputs high or low depending on the result, i.e., if $V_C$ is lower than the input voltage, a positive voltage (V1) is applied to charge the capacitor; if $V_C$ is higher than the input voltage, a lower voltage (V0) is applied to discharge the capacitor. Thus the capacitor voltage will seek a point where it varies above and below the input voltage by a small amount. Figure 6 illustrates the capacitor voltage and the comparator output.

Some mathematical analysis here will be useful to help clarify the technique and to point out its restrictions. Referrring to Figure 6, we have the following:

$$V_A = V0 + [V_B - V0][e^{**}(-t1/RC)]$$
$$V_B = V_A + [V1 - V_A][1 - e^{**}(-t2/RC)]$$
$$= V1 + [V_A - V1][e^{**}(-t2/RC)]$$

solving for t1 and t2 we have:

$$t1 = -RC\ \ln[(V_A - V0)/(V_B - V0)]$$
$$t2 = -RC\ \ln[(V_B - V1)/(V_A - V1)]$$

let:

$$V_A = V_{IN} - d1$$
$$V_B = V_{IN} + d2$$

substituting the above, the equations for t1 and t2 become:

$$t1 = -RC\ \ln\{[1 - (d1/(V_{IN} - V0))]/[1 + (d2/(V_{IN} - V0))]\}$$
$$t2 = -RC\ \ln\{[1 - (d2/(V_{IN} - V1))]/[1 - (d1/(V_{IN} - V1))]\}$$

the equations reduce by means of the following assumptions:

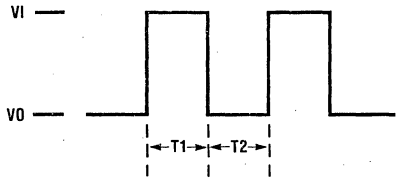
1. $d1 = d2 = d$
2. $|V_{IN} - V0| >> d$
   $|V_{IN} - V1| >> d$

applying these assumptions, we get the following:

$$t1 = -RC\ \ln[(1 + x)/(1 - x)] \text{ where } x = -d/(V_{IN} - V0)$$
$$t2 = -RC\ \ln[(1 + x)/(1 - y)] \text{ where } y = d/(V_{IN} - V1)$$

because of the assumptions above, the x and y terms in the preceding equations are less than 1, therefore the following expansion can be used:

$$\ln[(1 + z)/(1 - z)] = 2[z + (z^{**}3)/3 + (z^{**}5)/5 + \ldots]$$




$$V_C = \frac{(V1 - V0) \cdot T1}{T1 + T2}$$

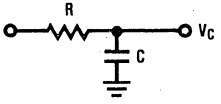



Figure 5

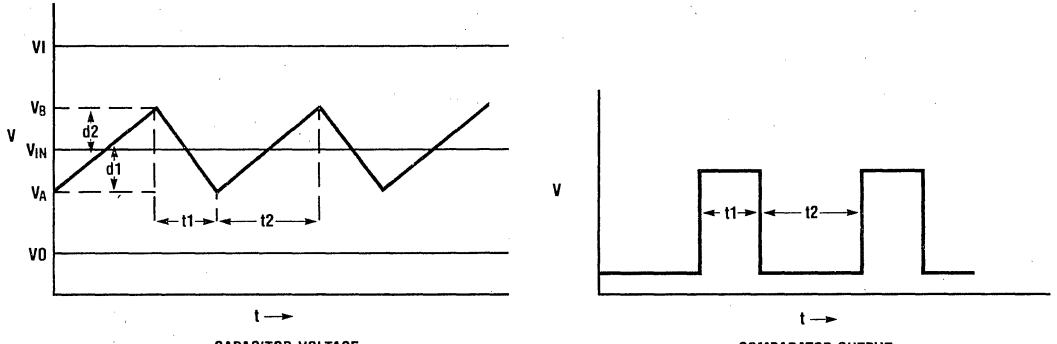



Figure 6

substituting we have:

$t1 = -2RC[x + (x^{**}3)/3 + \ldots]$

$t2 = -2RC[y + (y^{**}3)/3 + \ldots]$

under assumption 2 above, the linear term completely swamps the exponential terms yielding the following result (after substituting back into the equation):

$t1 = 2dRC/(V_{IN} - V0)$ $\quad$ $t2 = -2dRC/(V_{IN} - V1)$

therefore:

$t1/(t1 + t2) = (V1 - V_{IN})/(V1 - V0)$

$t2/(t1 + t2) = (V_{IN} - V0)/(V1 - V0)$

solving for $V_{IN}$:

$V_{IN} = [t2/(t1 + t2)][V1 - V0] + V0$

or $V_{IN} = V1 - [t1/(t1 + t2)][V1 - V0]$

It follows from the above results that by measuring the times t1 and t2, the input voltage can be accurately determined. As will be seen, the restrictions based upon the assumptions above do not cause any serious difficulty.

### A.2 General Accuracy Considerations

In the preceding calculations it was assumed that the differential output above and below the input voltage was the same. If the comparator output is checked at absolutely regular intervals, and if the intervals are kept as small as possible this assumption can be fairly easily guaranteed — at least to within the comparator offset which is only a few millivolts. As we shall see, this aspect of the technique presents few, if any, difficulties. In addition, there is an RC network at the input of the comparator. The time constant of this network must be long relative to the time between checks of the comparator output. This will insure that the capacitor voltage does not change very much between checks and thereby help to insure that the differences above and below the input voltage are the same.

The next major approximation has to do with the difference between the input voltage and either V1 or V0. We have relied on this difference being much greater than the amount the capacitor voltage changes above and below the input voltage. This approximation allows the nonlinear terms in the logarithmic expansion to be discarded. In practicality, the approximation means that the input voltage must not be "close" to either V1 or V0. Therefore, it becomes necessary to determine how closely the input voltage can approach V1 or V0. It is obvious that the smaller the difference d can be made, the closer the input voltage can approach either reference. The following calculations illustrate the method for determining that difference d. Note, using either V1 or V0 produces the same result. Thus $V = V1 = V0$.

For at least 1% accuracy

$x + (x^{**}3)/3 < 1.01x$

therefore $x < 0.173$

since $x = d/|(V_{IN} - V)|$ we have $d < 0.173\ |(V_{IN} - V)|$.

Using the same analysis for 0.1% accuracy in the approximation we get $d < 0.0548|(V_{IN} - V)|$. By applying this relationship, the RC time constant can be adjusted so that, within the time interval, the capacitor voltage does not change by more than d volts. The user may then select, within reason, how close to the references he can allow the input voltage to go.

The next consideration is really just one of simplification. It is clear that if V0 is zero, it drops out of the first equation and the relationship is simplified. Therefore, it is desireable to use zero volts as the V0 value. The equation then becomes:

$V_{IN} = V1t2/(t1 + t2)$.

It is obvious by now that the heart of the technique lies in accurately measuring the times t1 and t2. Clearly this requires that the time base of the COP420 be accurate. Short term variations in the COP420 time base will clearly impact the accuracy of the result. In addition to that there is a serious problem in being able to check the comparator output often enough to get any accuracy and resolution out of simply measuring the times t1 and t2. This problem is circumvented by measuring many periods of the waveform. Doing this gives a large average, which improves the accuracy and tends to eliminate any spurious changes. Of course, the trade off is increased time to do the conversion. However if the time is available, the technique becomes restricted only by the accuracy of the external components. Those of the comparator and the reference voltage are most critical.

It is clear from the equation above that the accuracy of the result is directly dependent upon the accuracy of the reference voltage V1. In other words, it is not possible to be more accurate than the reference voltage. If, however, all that is required is a ratio between the input voltage and the reference voltage, the accuracy of the reference will not be a controlling factor provided that the input voltage tracks the reference. This requires that the input voltage be generated from the reference voltage in some form, e.g., a voltage divider with $V_{IN}$ coming off a variable resistance.

Finally, we have noted that the difference d must be small. If the capacitor had to charge or discharge a long way toward $V_{IN}$, the nonlinearity of the capacitor charge curve would be significant. This therefore requires that the conversion begin with the capacitor voltage close to the input voltage.

Note that the RC value is not part of the equation. Therefore the accuracy of the time constant has no effect on the result as long as the time constant is long relative to the time between checks of the comparator output.

The final point is that the reference voltages, whatever they may be, must be hard sources. Should these voltages vary or drift at all, they will directly affect the result. In those configurations where the references are being switched in and out, the voltage should not change when it is switched into the circuit.

## B. BASIC IMPLEMENTATION

### B.1. General

The objective, then, is to measure the times t1 and t2. This is accomplished in the software by means of two counters. One of the two counters counts the t2 time; the other counter counts the total time $t1 + t2$.

It is necessary to check the comparator output at regular intervals. Thus the software must insure that

path lengths through the test and increment loops are equal in time. Further it is desirable to keep the time required to increment the counters as short as possible. A trade off usually comes into play here. The shortest loop in terms of code required to implement the function is rarely the shortest loop in terms of time required to execute the function. The user has to decide which implementation is best for him. The choice will frequently be governed by factors other than the A/D conversion limits.

It must be remembered that we are now dealing with analog signals. If significant accuracy is required, we are handling very small analog signals. This requires the user to take precautions that are normally required when working with linear circuits, e.g., power supply decoupling and bypassing, lead length restrictions, crosstalk, op amp and comparator stabilization and compensation, desired and undesired feedback, etc. As greater accuracy is sought these factors are more and more significant. It is suggested that the reader refer to the National Semiconductor Linear Applications Handbook and to the data sheets for the various components involved to see what specific precautions should be taken both in general and for a specific device.

### B.2 The Basic Circuit

Figure 7 shows the diagram for the basic circuit required to implement the duty cycle conversion scheme. The flow chart and code required to implement the function are shown in Figure 8. Note that the flow chart and code do not change — except for possible polarity change on output to allow for an inverting buffer — for any of the improvements in accuracy discussed later. The only exception to this is the technique illustrated in Figure 10 and the variations there are minor.

The code and flow chart in Figure 8 implement the technique as described above. The large averaging technique is used as it would be too difficult to measure the times t1 and t2 in a single period. The total time, t1 + t2, is the viewing window under complete control of the software. This window is a time equal to the total number of counts, determined by desired accuracy, multiplied by the loop time for a single count. A second counter is counting the t2 time. Special care is taken to insure that all paths through the code take the same length of time since the integrity of the time count is the essence of the technique. The full conversion scheme would use the subroutine in Figure 8. Normally the subroutine would be called first just to get the capacitor charged close to the input voltage. The result obtained here would be discarded. Then the routine would be called a second time and the result used as required.

In the configuration in Figure 7, there is an RC network in both input legs of the comparator. This is to balance the inputs of the device. For this reason, R1 = R2. C1 is the capacitor whose voltage is being varied by the pulse waveform. C2 is in the circuit only for stabilization and symmetry and is not significant in the result. The comparator tends to oscillate when the + and − inputs are nearly equal without capacitor C2 in the circuit.

As would be expected, the basic circuit has some difficulties. By far the most serious of these difficulties is the output level of the G line. To be sure of the high and low level of this output the levels should be measured. The "1" level will be between the spec minimum of 2.4V and $V_{CC}$ (here assumed to be 5 volts). The "0" level will be between the 0.4V spec maximum and ground. With light loads, these levels are likely to vary from device to device. Furthermore, we have the same "1" level problem that was mentioned in the simplest technique: the capacitive load is large and the capacitor is



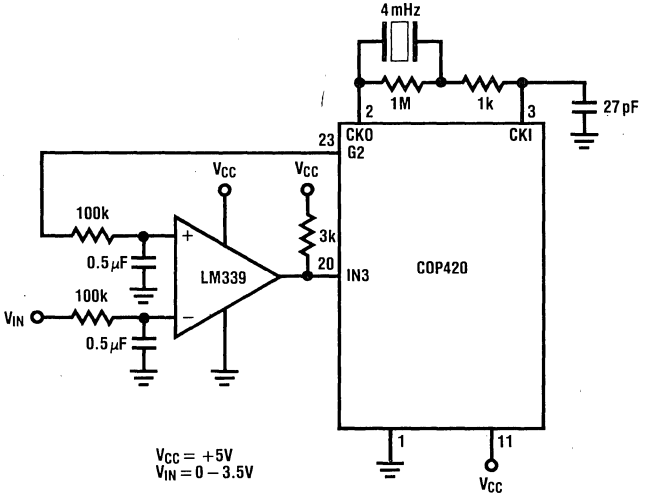



Figure 7. Basic Duty Cycle A/D

charging while the output is trying to go to the high level.

There is also a problem with the low level. When the output goes low, the capacitor begins to discharge through the output device of the COP420. This discharge current has the effect of raising the "0" level and thereby introducing error. Note that we are not talking about large changes in the voltages, especially the low level. Typically, the change will only be a few millivolts but that can translate into a loss of accuracy of several bits.

Under laboratory conditions — holding all variables constant and using precise measured values in the calculations — the circuit of Figure 7 yielded 5 bit ± 1 bit accuracy over the range of V0 (here measured to be 0.028 volts) to 3.5 volts (the maximum specified input voltage for the comparator with $V_S = 5$ volts). Increasing the number of total counts had very little effect on the result. In the general case, the basic scheme should not be relied upon for more than 4 bits of accuracy, especially if one assumes that $V1 = V_{CC}$ and $V0 = 0$. As shall be seen, it is not difficult to improve this accuracy considerably.

```
;ATOD IS THE FULL CONVERSION SCHEME WRITTEN AS A SUBROUTINE
ATOD:   LBI     1,10    ;MAKE SURE COUNTERS CLEARED
        JSRP    CLEAR
        LBI     2,10
        JSRP    CLEAR
        LBI     1,13    ;PRELOAD FOR TOTAL COUNT = 2048
        STII    0
        STII    0
        STII    8
ATOD1:  ININ            ;READ COMPARATOR--INPUT TO 420 = IN3
        AISC    8
        JP      SND01
SND1A:  LBI     3,0     ;USING OMG BELOW TO SAVE STATE OF OTHER G
                ;VALUES IF IT WAS NECESSARY TO DO SO,ELSE USE OGI
        SMB     2       ;VIN > Vc,DRIVE Vc HIGHER
        OMG             ;THIS CODE STRAIGHT LINED FOR SPEED
        SC              ;APPLY POSITIVE REFERENCE
        CLRA            ;INCREMENT THE SUB COUNTER
        LBI     2,13
        ASC
        NOP
        XIS
        CLRA
        ASC
        NOP             ;BINARY INCREMENT
        XIS             ;WOULD ELIMINATE THESE 4 WORDS IF 8 BIT
        CLRA            ;COUNTER OR LESS-HERE SET UP FOR UP TO 12 BIT
        ASC             ;COUNTER
        NOP
        X
        JP      TOTAL
SND01:  LBI     3,0
        RMB     2
        OMG
        CLRA
        AISC    10      ;THIS PART OF THE CODE MERELY INSURES THAT
        NOP             ;ALL PATHS THROUGH THE ROUTINE ARE EQUAL IN TI
DLY:    AISC    1
        JP      DLY
TOTAL:  CLRA
        LBI     1,13
        SC
        ASC             ;INCREMENT THE TOTAL LOOP COUNTER
        NOP             ;WHEN OVERFLOW,DONE SO EXIT
        XIS
        CLRA
        ASC
        NOP
        XIS
        CLRA
        ASC
        JP      ATOD2
        RET
ATOD2:  X
        JP      ATOD1
        PAGE    2
CLEAR:  CLRA
        XIS
        JP      CLEAR
        RET
```

**Figure 8A. Duty Cycle A/D Code**

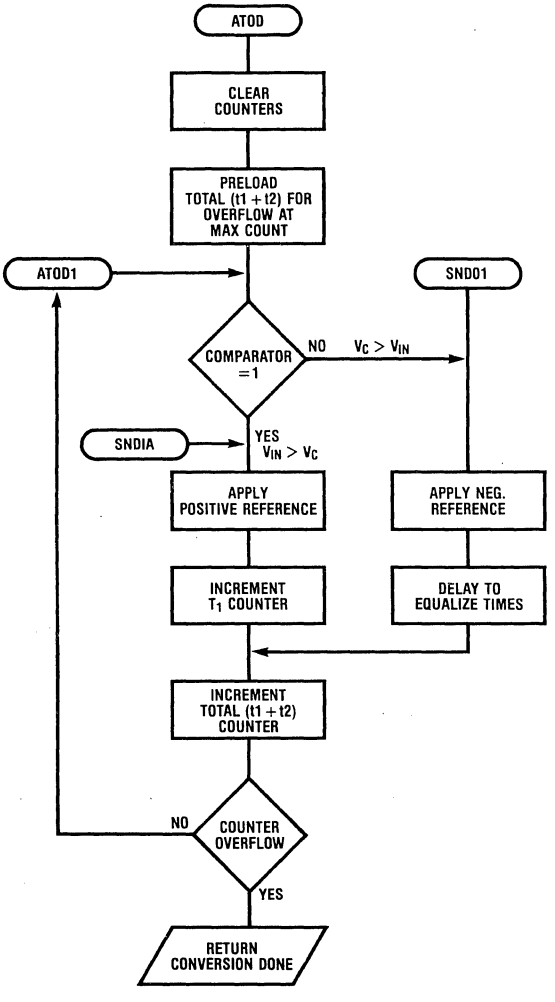



**Figure 8B. Duty Cycle A/D Flow Chart**

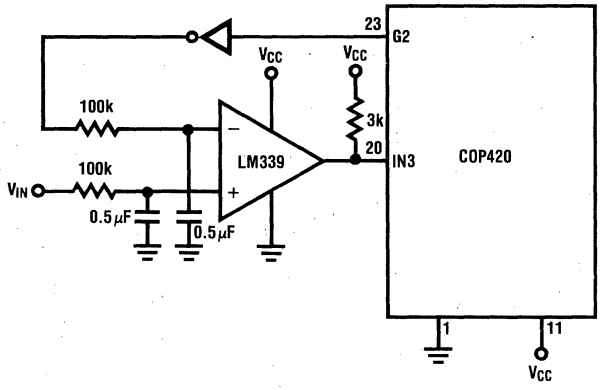
## C. ACCURACY IMPROVEMENTS

### C.1 General Improvements

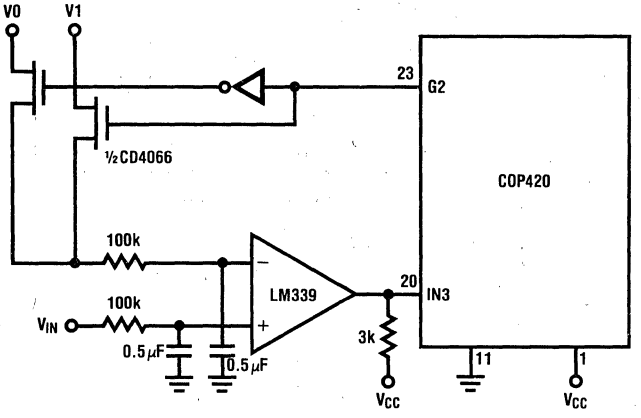
Figure 9 illustrates circuit changes that will make significant improvements in the accuracy of the technique. In Figure 9A a CMOS buffer is used to drive the RC network. The output of the COP420 drives the CMOS gate, which here is a 74C04 because of its output characteristics. The main thing that this technique does is to reduce the difficulties with the output levels. Typically, V0 is 0 volts and V1 is $V_{CC}$. We also have a "harder" source for the voltages — the levels don't change while the capacitor is charging or discharging. Now, even more clearly than before, the accuracy of $V_{CC}$ is the controlling voltage tolerance. The accuracy of the result will be no better than the accuracy of $V_{CC}$ (for a system requiring absolute accuracy).

Under laboratory conditions, the circuit of Figure 9A yielded the accuracies as indicated below for various total counts. The accuracy increased with the total count until the count exceeded 2048. There was no significant increase in accuracy with this circuit for counts in excess of 2048. (Remember that these results were obtained under controlled conditions). We may then view the results obtained with 2048 counts as the upper limit of accuracy with the circuit of Figure 9A. The results were as follows:

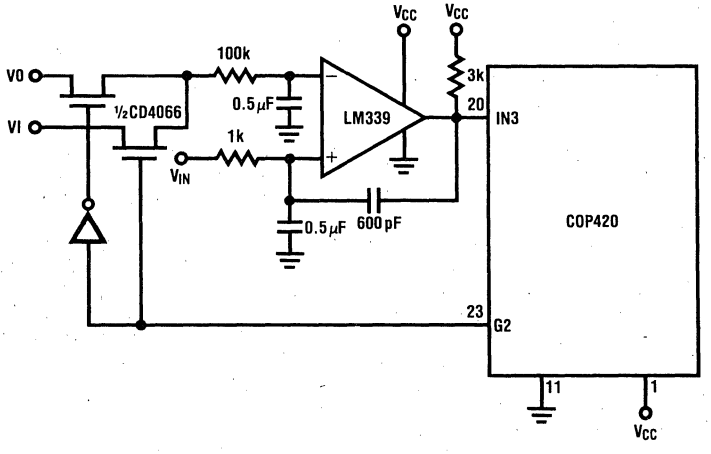
| Total Count | Resultant Accuracy |
|---|---|
| 512 | 8 ± 1/2 bits |
| 1024 | 9 ± 1 bits |
| 2048 | 9 ± 1/2 bits |
| 4096 | 9 ± 1/2 bits |




A




B




C

Figure 9. Improvements to Duty Cycle A/D

The circuit of Figure 9B makes a significant change to improve accuracy. Now the COP420 is controlling analog switches and switching in positive and negative references. Therefore the accuracy of the reference voltages is the controlling factor. Generally this will improve the accuracy over that obtained with Figure 9A. With the circuit of Figure 9B, with V0 = 1 volt (negative reference), and V1 = 3 volts (positive reference), 9 bit accuracy was achieved with a total count of 1024. V0 and V1 were arbitrarily chosen to place the input voltage approximately in the center of the allowable comparator input range with $V_S = 5$ volts. Remember, the accuracy of the references is controlling. The result can be no more accurate than the references. Furthermore, these references must be hard sources; i.e., they must not change when they are switched into the circuit as that contributes error into the result.

In Figure 9C, capacitive feedback was added to the comparator circuit and the series resistance to $V_{IN}$ was decreased. The feedback added hysteresis and forced the comparator to slew at its maximum rate (significant errors are introduced if the comparator does not change state in a time shorter than the cycle time of the controller). Both of these changes resulted in increased accuracy of the result. With V0 = 0, V1 = 5 volts ($V_{CC}$) and $V_{CC}$ held steady at 5.000 volts, an accuracy of 10 bits ± 1 bit was achieved over the input range of 0 to 3.5 volts.

It is obviously possible to use any combination of the configurations in Figure 9 for a given application. What is used will depend on the user and his specific requirements.

Figure 10 illustrates a further refinement of the basic approach. This configuration can be used if greater accuracies are needed. The major change is the addition of a summing amplifier to the circuit for the purpose of adding a fixed offset voltage to the input voltage. This has the effect of moving the input voltage away from the negative reference (which is 0 volts here). This offset voltage should be stable as the changes in it will directly affect the result. The offset voltage should be chosen so as to place the effective input voltage (the voltage at the comparator input) approximately in the center of the range between the two references. The precise value of the offset in not critical nor is its source. The forward voltage drop across a germanium diode is used as the offset in Figure 10, but this offset can be generated in any convenient manner. The forward voltage drop of the germanium diode is aproximately 0.3 volts. Given this and the negative reference of 0 volts and a positive reference of 2.5 volts, the input voltage is restricted to a range of 0 to 2 volts. Therefore, the effective input voltage (at the comparator input) is approximately 0.3 volts to 2.3 volts — well within the limits of the two references. The circuit also includes provision for an autozero self calibration procedure.

Note that the resistors in the summing amplifier should be matched. The absolute accuracy of these resistors is not significant, but their accuracy relative to one another can have a significant bearing on the result. The restriction is imposed so that the output of the summing amplifier is exactly the sum of the input voltage and the offset voltage. This requires unity gain

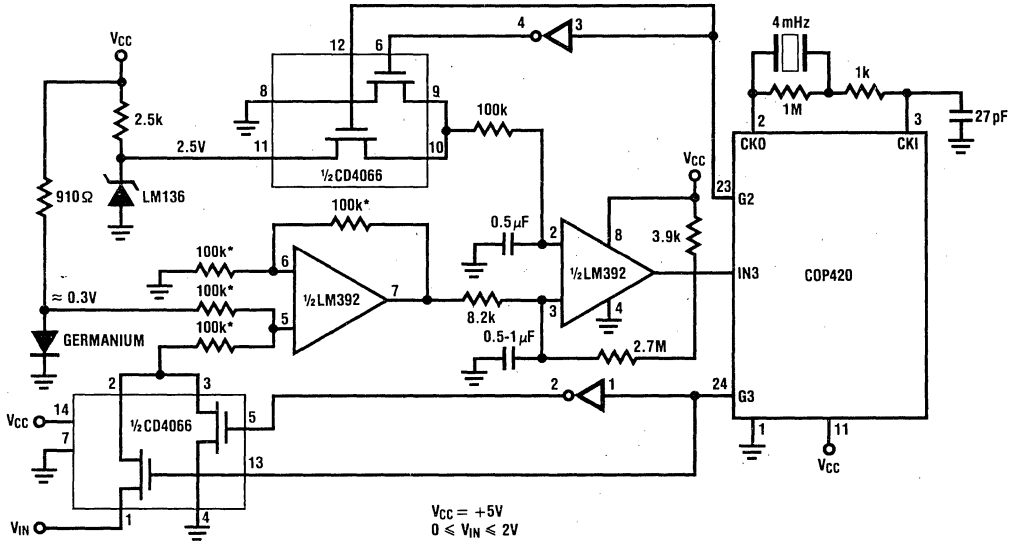



Figure 10. Improved Duty Cycle A/D with Autozero

through the amplifier and that the impedance in each summing leg be the same. These effects can become very serious if one is trying for significant accuracy — e.g., if 12 bit accuracy is being sought 1% matching of those resistors can introduce an error of 1% maximum. While 1% accurate is fairly good, it is significantly less than 12 bit accuracy. Related to this effect is a possible problem with the source impedance of the input voltage. If that impedance is significant in terms of its ratio to the summing resistor, errors are introduced just as if the resistors are mismatched. "Significant" is determined in terms of the desired system accuracy and the relative impedance values. The comparator section is using some feedback to provide hysteresis for stability and a low series resistance is used for the input to the comparator.

Most significantly, this configuration allows a true zeroing of the system. Through the additional analog switches shown, the COP420 can easily perform an autozero function by tying the input to ground and measuring the result. Thus the system offsets can be calculated, stored and subtracted from the result. This improves the accuracy and is also more forgiving on the choice of the comparator and op amp selected. Furthermore, the offset can be periodically recomputed by the COP420 thereby compensating for drift in system offsets. Nonetheless, the accuracy of the reference is the controlling factor. It is NOT possible to obtain an absolute (as opposed to ratiometric) accuracy of 12 bits without a reference that is accurate to 12 bits. The LM136 used in Figure 10 is a 1% reference. Although not inherently accurate to 12 bits, the voltage of the LM136 may be trimmed to an exact value by means of a variable resistor. The data sheet of the LM136 illustrates this connection. Under laboratory conditions, the circuit of Figure 1 yielded 11 bit ±1 bit accuracy with a total count of 4096 over the input range of 0 to 2 volts. Figure 11 indicates the flow chart and the code required to implement the technique of Figure 10.

```
        ;CODE FOR IMPROVED A TO D PULSE WIDTH METHOD
        ;SEE FIGURE 8A FOR CODE FOR ROUTINE ATOD
        ;
AUTZER: LBI     3,0     ;DO AUTO ZERO, 3,0 CONTAINS G STATUS
        RMB     3       ;SET UP TO GRND INPUT & MEASURE OFFSET
        JSR     ATOD    ;FIRST TIME IS TO GET CLOSE
        JSR     ATOD    ;MEASURE THE OFFSET
        LBI     2,13    ;NOW SAVE THE OFFSET VOLTAGE
XFER:   LD      1               ;SAVE THE OFFSET VALUE IN M3
        XIS     1
        JP      XFER
        LBI     0,0
        JP      INPUT
MEASUR:         ;NOW DO REAL MEASUR(1ST TIME IS OFFSET)
        JSR     ATOD    ;FIRST TIME TO GET CLOSE
        JSR     ATOD    ;NOW REAL MEASUREMENT
        JSRP    BINSUB  ;SUBTRACT THE OFFSET
        ;HAVE THE VALUE AT THIS POINT(IN BINARY)-NOW DO WHAT
        ;THE APPLICATION REQUIRES.  VALUE MUST BE MULTIPLIED
        ;BY (VREF+/TOTAL COUNT) TO GET FINAL VALUE IF SUCH IS
        ;DESIRED
        LBI     1,0     ;INCREMENT COUNTER FOR NEW OFFSET MEASURE
        LD
        AISC    1
        JP      SAVE
        X               ;IS 16TH TIME, MEASURE OFFSET AGAIN
        JP      AUTZER
SAVE:   X
        LBI     3,0
        SMB     3       ;SET BIT SO CAN MEASURE VIN
        JP      MEASUR
        .PAGE   2
BINSUB: LBI     3,13
        SC
BNSUB2: LD      1
        CASC
        NOP
        XIS     1
        JP      BNSUB2
        RET
```

**Figure 11A. Duty Cycle A to D, Improved Method**

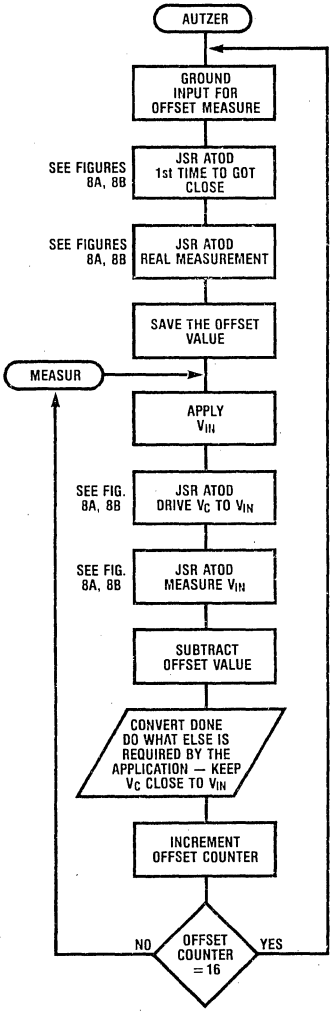



Figure 11B. Flow Chart for Improved Duty Cycle A/D

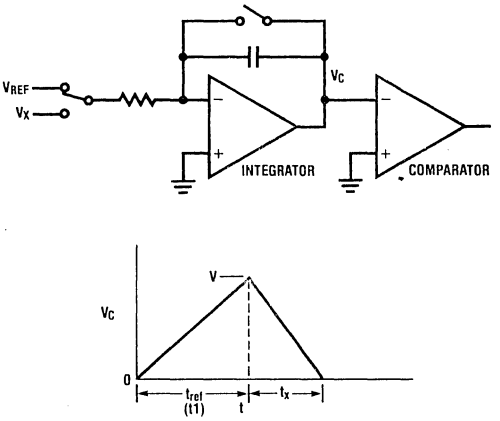



Figure 12. Dual Slope Integration — Basic Concept

$$I_X = C\frac{dV}{dt} = V_X/R$$

$$V_X = RC\frac{dv}{dt}$$

$$\int_0^{T1} V_X dt = \int_0^V RCdV$$

$$V_X T1 = RCV$$

$$V = V_X T1/RC = I_X T1/C$$

Similarly:

$$I_{REF} = C\frac{dV}{dt} = V_{REF}/R$$

$$V_{REF} = RC\frac{dV}{dt}$$

$$\int_{T1}^{T1+T_X} V_{REF} dt = \int_V^0 RCdV$$

$$V_{REF} T_X = -RCV$$

$$V = -V_{REF} T_X/RC$$

$$-V_{REF} T_X/RC = V_X T1/RC$$

$$V_X = -V_{REF} T_X/T1$$

Two important facts arise from the preceding mathematics. First of all, there is a linear relationship involved in determining the unknown voltage. Secondly, the negative sign in the final equation indicates that the reference and the unknown, relative to some point (which may be 0 volts or some bias voltage), have opposite polarity. Thus, if it is desired to measure 0 to +5 volts, the reference voltage must be −5 volts. If the input is restricted to 2.5 to 5 volts, the reference can be 0 volts as the integrator and comparator are biased at +2.5 volts (then the 0 volts is in fact −2.5 volts relative to the biasing voltage, and the input range is 0 to 2.5 volts relative to the same bias voltage).

There are some difficulties with dual polarity conversion using the dual slope method. It is clear from the math above that if the input voltage will be dual polarity, it is necessary to have two references — one of each polarity. The midrange biasing arrangement briefly

## IV. Dual Slope Integration Techniques

### A. MATHEMATICAL BACKGROUND

(Some of this background information is taken from National Semiconductor Linear Applications Note AN-155. The reader is referred to that document for other related general information.)

The basic approach of dual slope integration conversion techniques is to integrate a voltage across a capacitor for a fixed time, and then to integrate in the other direction with a known voltage until the starting point is reached. The ratio of the two times then represents the unknown voltage. Some of the math below in conjunction with Figure 12 will illustrate the approach.

described above eliminates the need for two different polarities but does not help very much since two references are still required — one at the positive value and one at the bias value. Ground is the other reference. Further, the need to select one of two references further complicates the circuitry involved to implement the approach. Also, the dual requirement brings up a difficulty with the bias currents of the integrator and comparator. They could add to the slope in one polarity and subtract in the other.

The only real operational difficulty in dual slope systems is establishing the initial conditions on the integrating capacitor. If this capacitor is not at the proper initial conditions, accuracy will be severely impaired. Figure 12 indicates a switch across the capacitor as a means of initializing it. In a software driven system, the initilization can be accomplished by doing two successive conversions. The result of the first conversion is discarded. It is performed only to initialize the capacitor. The second conversion produces the valid result. One need only insure that there is not significant time lapse between the two conversions. They should take place immediately after one another.

This approach obviously lengthens conversion time but it eliminates many problems. The alternative to this approach of two successive conversions is to take a great deal of care in insuring the initial state of the integrating capacitor and in selecting op amps and comparators with low offsets.

## B. THE BASIC DUAL SLOPE TECHNIQUE

Figure 13 indicates an implementation of the basic dual slope technique. This is a single polarity system and thus requires only the single reference voltage. The circuit of Figure 13 is perhaps not the cheapest way to implement such a scheme but it is representative and illustrates the factors that must be considered.

Consider first the means of initializing the integrating capacitor C1. The routine here connects the input to ground and does a conversion on zero volts as a means of initialization. Subsequently — and this is typical of the more usual technique — two conversions are performed. The first conversion is to initialize the capacitor. The second conversion yields the result. Some form of initialization or calibration prodcedure is required to achieve optimum accuracy from dual slope conversion schemes.

The comparator in this circuit is used in the inverting mode and has positive feedback as recommended in the LM111 data sheet. The voltage reference is the LH0070, which is a 0.01% reference. A resistive voltage divider on the LH0070 creates the 5 volt value. The use of the voltage divider brings up two difficulties (which can be overcome if the LH0070 is used at its full value, thus eliminating the divider, and the result properly scaled in the microcontroller or series integrating resistor increased). First, the impedance of the reference must be small relative to the series resistance used in the integrator. If this were not the case, the

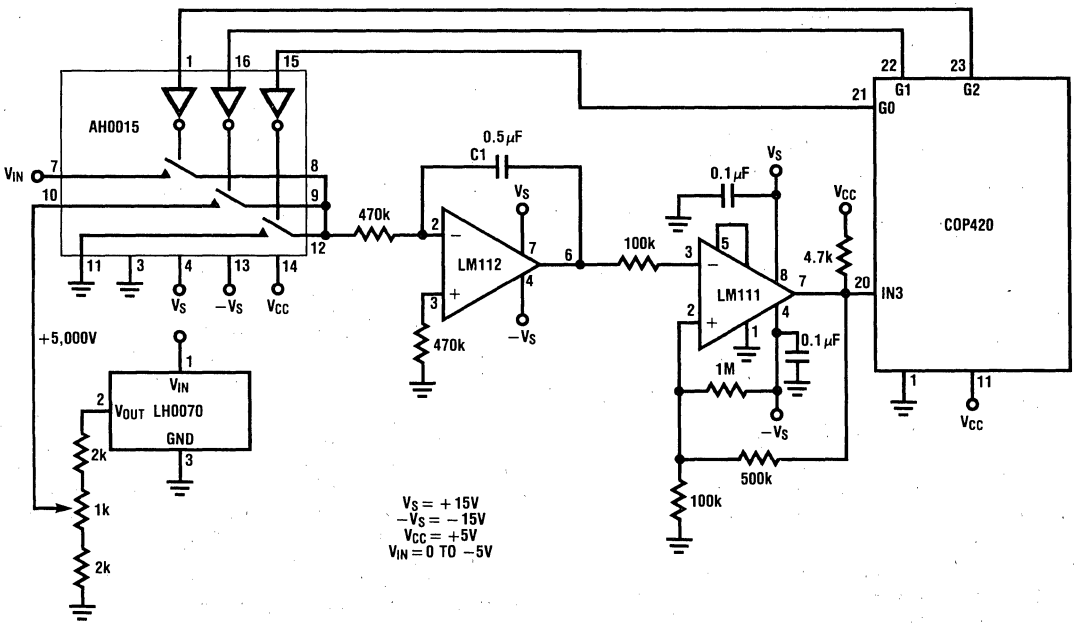



Figure 13. Basic Dual Slope Integration A/D Scheme

slopes would show an effect due to the difference in the R value between the applied reference voltage and the unknown input. (By the same token, the output impedance of the source supplying the unknown must also be small relative to that series integrating resistor). Secondly, the bias currents of the integrator may be such as to affect the reference voltage when it is coming from a simple resistor divider. Both problems are reduced if small resistor values are used in the divider. Note also that current mode switching would reduce the problem as well. It should be pointed out that the errors introduced by these problems are not gross deviations from the expected value. They are small errors that will not make much difference in the majority of applications. They are, however the kind of errors that can make the difference between a system accurate to 10 bits and one accurate to 12 bits (assuming all other factors the same).

Figure 14 shows the flow chart and code required to implement the basic dual slope technique as shown in Figure 13. Under laboratory conditions an accuracy of 12 bits ±1 bit was achieved. The method is slow, with the maximum conversion time equal to $2 \times T_{REF}$. Notice that the accuracy of $V_{CC}$ and that of the integrating resistor and capacitor are not involved in the accuracy of the result. The accuracy of $V_{REF}$ is, of course, controlling if absolute accuracy — rather than ratiometric accuracy — is desired. The absolute accuracy of the circuit can be no better than the accuracy of the reference. If ratiometric accuracy is all that is required, there is no particular problem. The accuracy is merely relative to the reference. The R and C values do not impact the accuracy because the integration in both directions is being done through the same R and C. Results would be quite different is a different value of R or C was used for one of the slopes.

```
DUALSLP: OGI     1          ;HOLD THE INPUT TO GROUND TO RESET THE
         LBI     2,11       ;INTEGRATING CAPACITOR
         JSRP    CLEAR      ;CLEAR THE COUNTER
         JSR     INCRA      ;TO GET US CLOSE, NEXT READING IS REAL
CLEAR2:  LBI     2,11       ;NOW CLEAR THE COUNTER
         JSRP    CLEAR      ;MAKE SURE COUNTER CLEARED TO ZERO
  ;J,15 = 0 AND START AT 1,13 FOR COUNT = 4096
  ;J,15 = 14 AND START AT 1,12 FOR COUNT = 8192
  ;J,15 = 12 AND START AT 1,12 FOR COUNT = 16384
  ;FOLLOW SAME PATTERN FOR OTHER COUNTS
  ;
MEASUR:  JSR     INCRA      ;RUN THRU THE INCREMENTS
         ; NOW HAVE THE BINARY VALUE, USE IT AS IS OR
         ; MULTIPLY BY (Vref/TOTAL COUNT) TO CREATE THE VOLTAGE
         ; RESULT--THEN CONTINUE WITH THE OPERATION
         LBI     2,11
         JSRP    CLEAR      ;CLEAR THE COUNTER
         JSR     INCRA      ;TO GET CAP CLOSE TO 0 AGAIN
         JP      CLEAR2
 ;FOLLOWING SUBROUTINE INCRA IS THE REAL PART OF THE ROUTINE
 ;CONCERNED WITH THE COUNTING FOR THE CONVERSION.
INCRA:   LBI     1,15       ;R1 IS CLEARED PRIOR TO START
         STII    15         ;PRESET THE COUNTER FOR 4096
         OGI     4          ;APPLY VIN
INCR:    LBI     1,12
         SC
BINAD1:  CLRA
         ASC
         NOP
         XIS
         JP      BINAD1
         NOP                ;2 NOPS TO EQUALIZE TIMES
         NOP
         SKC
         JP      INCR
         OGI     2          ;DONE, NOW APPLY VREF
INCR2:   LBI     2,12       ;COUNT UNTIL COMPARATOR CHANGES
         SC
BINAD2:  CLRA
         ASC
         NOP
         XIS
         JP      BINAD2     ;STRAIGHT LINE THE ADD FOR SPEED
         ININ               ;SAVE WORDS BY USING G
         AISC    8          ;SEE IF IN3=1
         JP      INCR2      ;IN3 IS 0, KEEP COUNTING
OUTPUT:  OGI     1          ;KEEP INPUT AT 0
         RET
```

**Figure 14A. Dual Slope A/D Code**

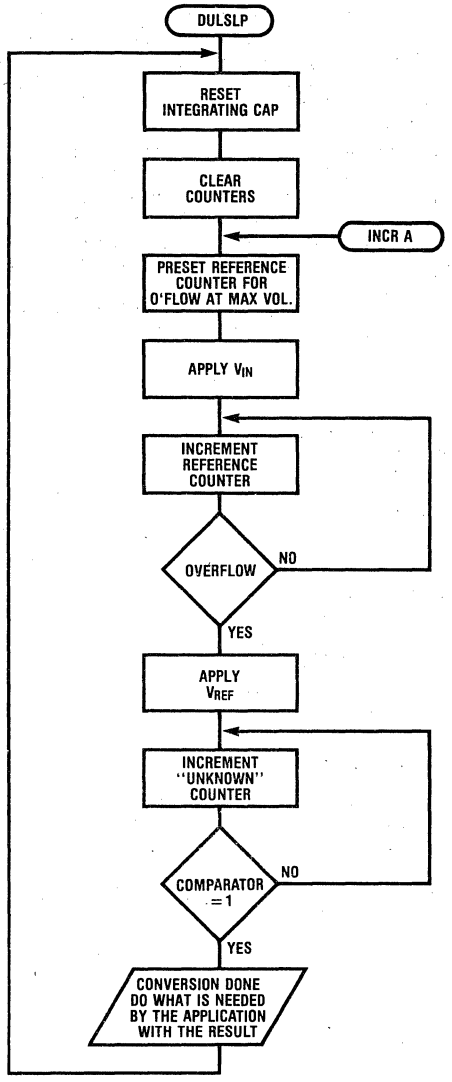



Figure 14B. Basic Dual Slope A/D Flow Chart

## C. MODIFIED DUAL SLOPE TECHNIQUE

### C.1 General

The basic idea of the modified dual slope technique is the same as that of the basic approach. The modified approach eliminates the need for dual polarity references and is also more forgiving in the selection of the op amp and comparator required. Figure 15 illustrates the basic idea

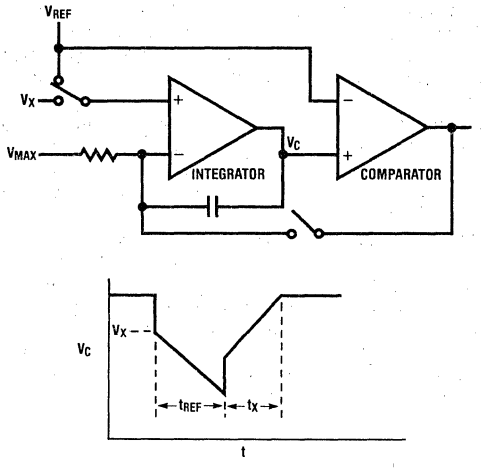



Figure 15. Modified Dual Slope — Basic Concept

The math analysis is much the same:

$$I_X = C\frac{dV}{dt} = (V_X - V_{MAX})/R$$

$$V_X - V_{MAX} = RC\frac{dV}{dt}$$

$$(V_X - V_{MAX})T1 = RC$$

$$V = (V_X - V_{MAX})T1/RC$$

Similarly:

$$I_{REF} = C\frac{dV}{dt} = (V_{REF} - V_{MAX})/R$$

$$(V_{REF} - V_{MAX})T_X = -VRC$$

$$V = -(V_{REF} - V_{MAX})T_X/RC$$

$$(V_{MAX} - V_{REF})T_X = (V_X - V_{MAX})T1$$

$$V_X = V_{MAX} + (V_{MAX} - V_{REF})T_X/T1$$

The main difference between this and the basic approach is the offset voltage $V_{MAX}$. The main restriction is that all input voltage values ($V_X$) are less than $V_{MAX}$. It is also apparent that the total count is proportional to the difference between $V_{MAX}$ and $V_X$. The only significant effect of this is, however, to slightly complicate the arithmetic required to arrive at a value for $V_X$.

Given that the input voltage $V_X$ is always less than $V_{MAX}$, the modified dual slope technique is automatic polarity. This fact comes straight out of the equation above. Thus dual polarity references are not required. However, two precise voltages are required: $V_{MAX}$ and $V_{REF}$. However, the $V_{MAX}$ value can be used for a zero adjust as indicated in Figure 16. This means that the $V_{MAX}$ value need not be so precise as it will be adjusted in a calibration procedure to produce a zero output. This adjustment amounts to a compensation for the bias currents and offsets. Thus the COP420 can use the supposed value of $V_{MAX}$ with $V_{MAX}$ later being "tweaked" to give the proper result at zero input. In addition, the initialization loop for the integrating capacitor includes the comparator. Thus the initial condition on the capacitor becomes not zero but the

sum of the offset voltages of the comparator and op amp. Thus the choice of these components is not critical in a modified dual slope approach.

### C.2 An Example of the Modified Dual Slope Approach

Figure 16 illustrates an implementation of the modified dual slope technique. The system is calibrated by holding $V_{IN}$ to ground and then adjusting $V_{MAX}$ for a "0" result. Capacitor C1 is the integrating capacitor. Capacitor C2 is used only to cause a rapid transition on the comparator output. C2 is especially useful if an op amp is being used as the comparator stage. Resistor R1 is just part of the capacitor initializing loop. An LH0070 is being used to generate the reference voltage and the $V_{MAX}$ value. The discussion previously about these being hard sources is equally relevant here. In fact, this problem was much more significant in this particular implementation and made the difference between a 10 and 12 bit system. As shown, the technique was accurate to 10 bits. Another bit was obtained when the $V_{MAX}$ and $V_{REF}$ values were buffered. It must be remembered that when trying to achieve accuracies of this magnitude board layout, parts placement, lead length, etc. become significant factors that must be specifically addressed by the user.

There are some other considerations in using this technique. The amount of time required to count the specified number of counts starts to become a significant factor. If it takes "too long" to do the counting, the capacitor can charge to either supply voltage depending on which direction it is integrating. This causes the wave shape shown in Figure 15 to flatten out. This effectively limits the input range for all accuracy is lost once that waveform flattens out. In fact, this was the limiting factor on the accuracy in Figure 16 as shown. Given the amount of time required for an increment of the counter for $T_{REF}$ (or $T_X$), it was not possible to reach the 4096 counts required for 12 bit accuracy before the waveform flattened out. Decreasing the total count solves the problem at the expense of accuracy. It is therefore desirable to keep the loop time required for an increment as fast as possible. The code to implement Figure 16 is shown in Figure 17 and reflects that concern. The other way to solve the problem is to use a large value for R and C. This is the easiest solution and preserves accuracy. Its cost is increased conversion time.

Both the basic and modified dual slope schemes can be very accurate and are commonly used. They tend to be relatively slow. In many applications, however, speed is not a factor and these approaches can serve very well. There are various approaches to dual slope analog to digital conversion which try to improve speed and/or accuracy. These are usually multiple ramping schemes of one form or another. The heart of the approach is the basic scheme described above. It is not the purpose here to delve into all the possible ways that dual slope conversion may be accomplished. The control software is not significantly different regardless of which particular variation is used. The basic ramping control is the same as that indicated here.

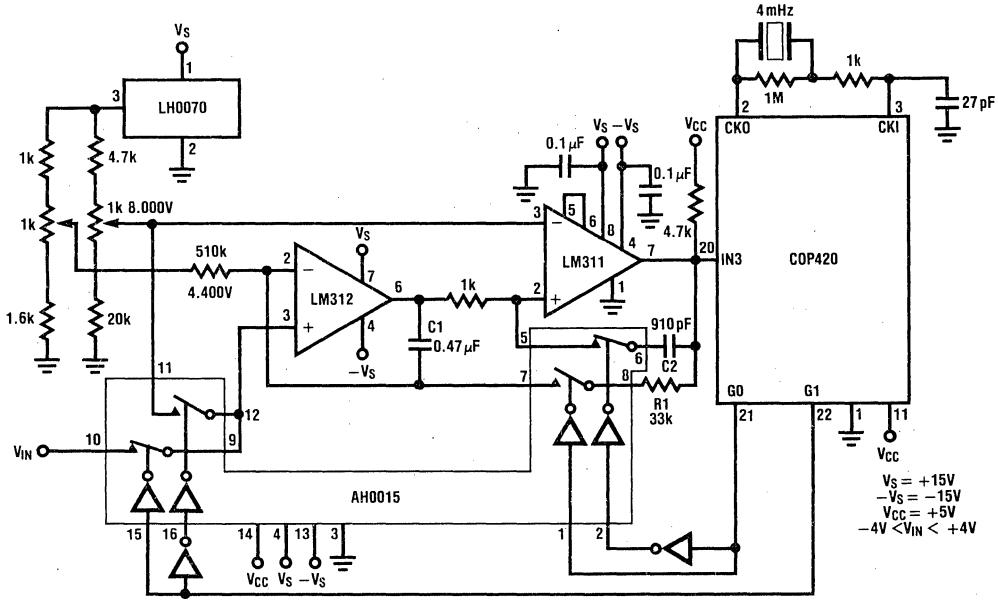



**Figure 16. Modified Dual Slope Integration**

The number of components required to implement a dual slope scheme is not related to the desired accuracy. The approach is generally tolerant as to the op amps and comparators used as long as proper care is given to the initialization of the integrating capacitor.

Precise references are not required if a ratiometric system is all that is required. Cheaper switches can be safely used. The dual slope scheme controlled by a COPS™ microcontroller can be a very cost effective solution to an analog to digital conversion problem.

```
CLRCAP: OGI     1       ;APPLY VREF AND ENABLE RESET PATH
CLEAR2: LBI     2,11    ;NOW CLEAR THE COUNTER
        JSRP    CLEAR
  ;1,15=15,1,14=4 AND START AT 1,12 FOR COUNT = 3072
  ;1,15 =15 AND START AT 1,12 FOR COUNT = 4096
  ;1,15 = 14 AND START AT 1,12 FOR COUNT = 8192
  ;1,15 = 12 AND START AT 1,12 FOR COUNT = 16384
  ;FOLLOW SAME PATTERN FOR OTHER COUNTS
  ;
MEASUR: JSR     INCRA   ;RUN THRU THE INCREMENTS
        ;HAVE THE VALUE AT THIS POINT, DO WHAT THE APPLICATION
        ;REQUIRES--REMEMBER,TO CREATE REAL VALUE MUST MULTIPLY
        ;RESULT BY (VREF-VMAX)/TOTAL COUNT AND THEN SUBTRACT
        ;THAT RESULT FROM VMAX--DO IT IN DECIMAL OR BINARY,WHICHEVER
        ;IS BEST FOR THE APPLICATION
        LBI     1,11    ;MAKE SURE SPACE IS CLEARED
        JSRP    CLEAR
        LBI     2,11
        JSRP    CLEAR
        JSR     INCRB    ;FOR TEST-KEEP IT CLOSE
        LBI     1,11     ;MAKE SURE COUNTER IS CLEARED
        JSRP    CLEAR
        JP      CLEAR2
INCRA:  LBI     1,14
        STII    4        ;PRESET HERE FOR SMALLER COUNT
        STII    15       ;PRESET THE COUNTER FOR 4096
INCRA1: OGI     2        ;APPLY VIN AND ENABLE FEEDBACK
INCR:   LBI     1,12
        SC
BINAD1: CLRA
        ASC
        NOP
        XIS
        JP      BINAD1
        NOP              ;2 NOPS TO EQUALIZE TIMES
        NOP
        SKC
        JP      INCR
        OGI     0        ;DONE,NOW APPLY VREF
INCR2:  LBI     2,12     ;COUNT UNTIL COMPARATOR CHANGES
        SC
BINAD2: CLRA
        ASC
        NOP
        XIS
        JP      BINAD2   ;STRAIGHT LINE THE ADD FOR SPEED
        ININ             ;SAVE WORDS BY USING G
        AISC    8        ;SEE IF IN3=1
        JP      INCR2    ;IN1 IS 0,KEEP COUNTING
OUTPUT: OGI     1        ;CLEAR THE CAPACITOR,APPLY VREF
        RET
INCRB:  LBI     1,14     ;MAKE THE PASS FOR CAP INIT SHORT
        STII    7
        STII    15
        JP      INCRA1
```

**Figure 17A. Modified Dual Slope Code**

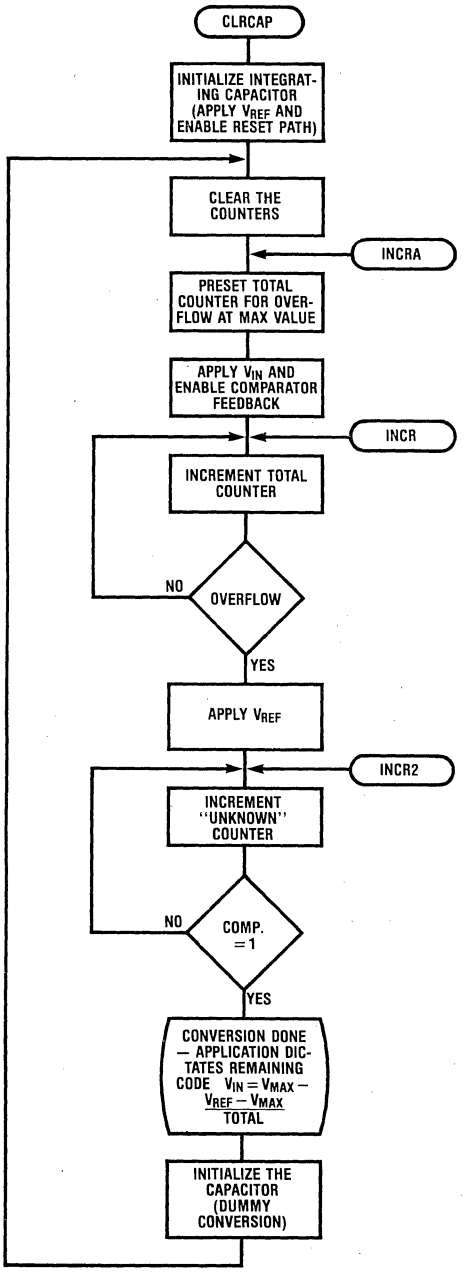



Figure 17B. Modified Dual Slope Flow Chart

# V. Voltage to Frequency Converters, VCO's

## A. BASIC APPROACH

The basic idea of this scheme is simply to use the COP420 to measure the frequency output of a voltage to frequency converter or VCO. This frequency is in direct relation to the input voltage by the very nature of such devices. There are really only two limiting factors involved. First of all, the maximum frequency that can be measured is defined in the microcontroller by the amount of time required to test an input and increment a counter of the proper length. With the COP420 this upper limit is typically 10 to 15kHz. The other limiting factor is simply the accuracy of the voltage to frequency converter or VCO. This accuracy will obviously affect the accuracy of the result.

Two basic implementations are possible and their code implementation is not significantly different. First, the number of pulses that occur within a given time period may be counted. This is straightforward and fairly simple to implement. The crucial factor is how long that given time period should be. To get the maximum accuracy from this implementation the time period should be one second. Such a time period would allow the distinction between the frequencies of 5000Hz and 5001Hz for example (assuming the V to F converter was that accurate or precise). Decreasing the amount of time will decrease the precision of the result. The alternate approach is to measure (by means of a counter) the amount of time between two successive pulses. This period measurement is only slightly more complicated than the pulse counting approach. The approach also makes it possible to do averaging of the measurement during conversion. This will smooth out any changes and add stability to the result. The time measurement technique is also faster than the pulse counting approach. Its accuracy is governed by how finely the time periods can be measured. The greater the count that can be achieved at the fastest input frequency — shortest period — the more accurate the result.

Figure 18 illustrates the basic concept. Figure 19 shows the flow charts and code implementation for both of the approaches discussed above. Note that whatever type of V to F converter is used, the code illustrated in Figure 19 is not significantly changed. In the code of Figure 19, the interrrupt is being used to test an input and thereby decreases the total time loop.

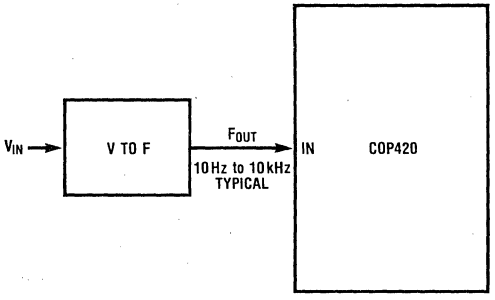



Figure 18. V to F Converter — Basic Concept

```
MEASUR:             MEASURE BY COUNTING PULSES OF V TO F

        LEI     2           ;ENABLE INTERRUPT
        LBI     1,14        ;PRESET TIME FOR 122 COUNTS
        STII    5           ;APPROX ONE HALF SECOND
        STII    8
TIME:   SKT                 ;USE INTERNAL TIMER TO FIND
        JP      TIME        ;THE 1/2 SECOND
BINPLS: LBI     1,14        ;HAVE GOT IT, INCREMENT COUNTER
        SC
BINADD: CLRA
        ASC
        NOP
        XIS
        JP      BINADD
        SKC                 ;NOW SEE IF DONE
        JP      TIME        ;NO COUNTER OVERFLOW, CONTINUE
        LEI     0           ;DONE, DISABLE INTERRUPT
FIN:    ;AT THIS POINT HAVE THE VALUE--CONVERT IT TO DECIMAL OR
        ;SEND IT OUT OR PROCESS IT FURTHER, WHATEVER IS REQUIRED
        ;BY THE APPLICATION.  ARITHMETIC IS REQUIRED TO CREATE THE
        ;VOLTAGE VALUE, USUALLY A SIMPLE MULTIPLY
        ;MAY HAVE TO DOUBLE THE RESULT TO COMPENSATE LOOKING FOR
        ;ONLY 1/2 SECOND IN THIS CASE
        ;
        JP      MEASUR      ;DO IT OVER AGAIN
        .=X'OFF             ;SET ADDRESS TO OFF FOR INTERRUPT
INTENT: NOP                 ;ADDRESS OFF MUST BE NOP FOR INTERRUPT
INTRPT: LBI     2,12        ;DO ADD OF THE VALUE FOR FREQ CNT
        SC
INTRI:  CLRA                ;STRAIGHT LINE THE CODE FOR SPEED
        ASC
        NOP
        XIS
        CLRA
        ASC
        NOP
        XIS
        CLRA
        ASC
        NOP
        XIS
        CLRA
        ASC
        NOP
        X
        LEI     2           ;ENABLE THE INTERRUPT AGAIN
        RET
```

Figure 19A. V to F By Counting Pulses

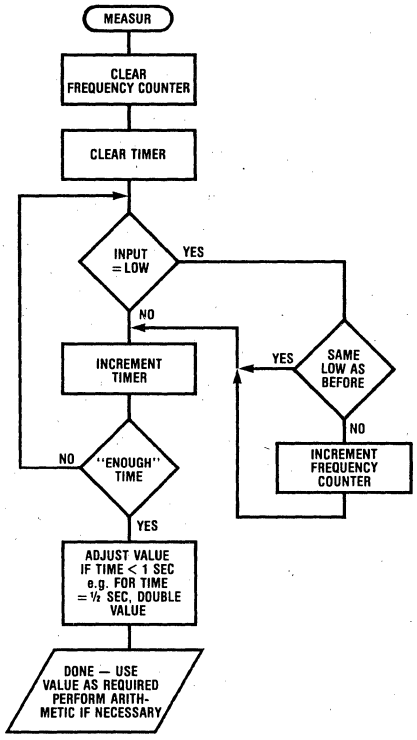



Figure 19B. V to F By Counting Pulses

```
                ;USE INTERRUPT FOR CATCHING THE PULSE EDGE

VPER:   LBI     0,12        ;CLEAR COUNTER SPACE AND FLAG
        STII    0
        STII    0
        STII    0
        STII    0
        LBI     0,12
        LEI     2           ;NOW ENABLE THE INTERRUPT
WAIT:   SC                  ;DUMMY WAIT LOOP, WAITING FOR SIGNAL TO
        LBI     0,12        ;INTERRUPT THE CONTROLLER
        JP      WAIT
        .=X'OFF             ;SET ADDRESS TO OFF--INTERRUPT ENTRY POINT
INTENT: NOP                 ;REQUIRED FOR INTERRUPT ENTRY
COUNT:  LBI     0,12        ;NOW CHECKING TO SEE IF SECOND INTERRUPT
        SKMBZ   0           ;I.E., ARE WE DONE?
        JP      DONE
        SMB     0           ;SET BIT FOR NEXT INTERRUPT
        LEI     2           ;ENABLE INTERRUPT AGAIN
PLUS1:  LBI     0,13        ;NOW START COUNTING
        SC
        CLRA                ;STRAIGHT LINE THE CODE FOR SPEED
        ASC
        NOP
        XIS
        CLRA
        ASC
        NOP
        XIS
        CLRA
        ASC
        NOP
        X
        JP      PLUS1
DONE:   ;FINISHED WHEN GET HERE--THE COUNT REPRESENTS THE PERIOD
        ;WITH ABOVE CODE, THE ACTUAL PERIOD IS THE COUNT MULTIPLIED
        ;BY 15(THE NUMBER OF WORDS TO INCREMENT BY 1) PLUS AN OVERHEAD
        ;OF 9 CYCLE TIMES =  24 CYCLE TIMES.  AT 4us THIS IS 96 us
        ;OR A FREQUENCY OF JUST OVER 10KHz.  MAX COUNT HERE IS 4095.
        ;THIS GIVES A MAXIMUM PERIOD = 61434 CYCLE TIMES(=245.736ms AT
        ;4us).  THIS CORRESPONDS TO A FREQUENCY OF JUST OVER 4Hz
        ;NOTE, THIS IS 12 BIT RESOLUTION
```

Figure 19C. A to D with VF Converter/VCO By Measuring Period

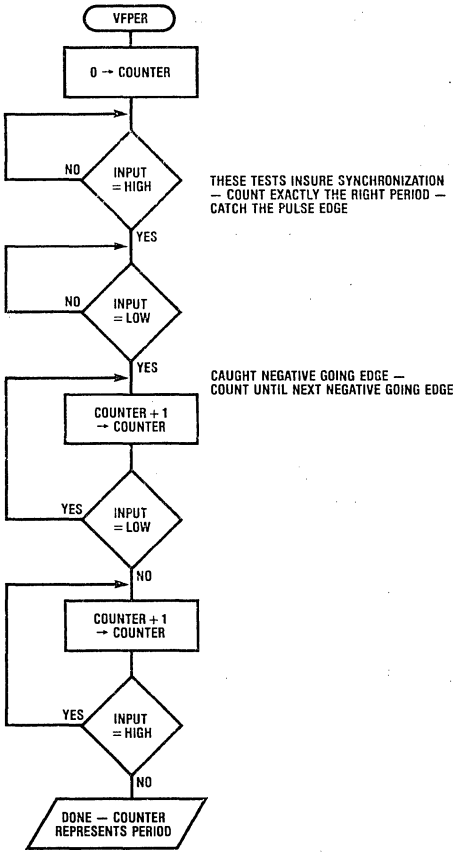



**Figure 19D. V to F — Measure Period**

## B. THE LM131/LM231/LM331

The LM131 is a standard product voltage to frequency converter with a linear relationship between the input voltage and the resultant frequency. The reader should refer to the data sheet for the LM131 for further information on the device itself and precautions that should be taken when using the device. Figure 20 is the basic circuit for using the LM131. Figure 21 represents improvements that increase the accuracy (by increasing the linearity) of the result. Note that these circuits have been taken from the data sheet of the LM131 and the user is referred there for a further discussion of their individual characteristics. With the LM131 the frequency output is given by the relationship:

$F_{OUT} = (V_{IN}/2.09)(1/R_TC_T)(R_S/RL)$

It is clear from the expression above that the accuracy of the result depends upon the accuracy of the external components. The circuit may be calibrated by means of a variable resistance in the $R_S$ term (a gain adjust) and an offset adjust. The offset adjust is optional but its inclusion in the circuit will allow maximum accuracy to be obtained. The standard calibration procedure is to trim the gain adjust ($R_S$) until the output frequency is correct near full scale. Then set the input ot 0.01 or 0.001 of full scale and trim the offset adjust to get $F_{OUT}$ to be correct at 0.01 or 0.001 of full scale. With that calibration, the circuit of Figure 20 is accurate to within ±0.03% typical and ±0.14% maximum. The circuit of Figure 21 attains the spec limit accuracy of ±0.01%.

## C. VOLTAGE CONTROLLED OSCILLATORS (VCO's)

A VCO is simply another form of voltage to frequency converter. It is an oscillator whose oscillation frequency is dependant upon the input voltage. Numerous designs for VCO's exist and the reader should refer to the data sheets and application notes for various op-amps and VCO devices. The code in Figure 19 is still applicable if a VCO is used. The only possible difficulty that might be encountered is if the relationship between frequency and input voltage is non-linear. This does not affect the basic code but would affect the processing to create the final result. A sample circuit, taken from the data sheet of the LM358, is shown in Figure 22. The accuracy of the VCO is the controlling factor.

## D. A COMBINED APPROACH

Elements of the period measurement and pulse counting techniques can be combined to produce a system with the advantages of both schemes and with few problems. Such a system is only slightly more complicated in terms of its software implementation than the approaches mentioned above. Note that in a microcontroller driven system, no additional hardware beyond the voltage to frequency converter is required to implement this approach. Basically, the microcontroller establishes a viewing window during which time the microcontroller is both measuring time and counting pulses. The result can be very precise if two conditions are met. First, when the microcontroller determines that it needs the conversion information, the microcontroller does not begin counting time or pulses until the first pulse is received from the VFC (first pulse after the microcontroller "ready"). Note, the COPS™ microcontroller could provide a "start conversion" pulse to enable the VFC if such an arrangement were desirable. The time would be counted for a fixed period and the number of pulses would be counted. After the fixed period of time the controller would wait for the next pulse from the VFC and continue to count time until that pulse is received. The ratio of the total time to the number of pulse is a very precise result provided that all the system times are slow enough that the microcontroller can do its job. The speed limits mentioned previously apply here. It is clear that the total time is not fixed. It is some basic time period plus some variable time. This is a little more complicated than simply using a fixed time, but it allows greater accuracies to be achieved. Also, the approach takes approximately the same amount of time for all conversions. It is also faster than the simple pulse counting scheme.

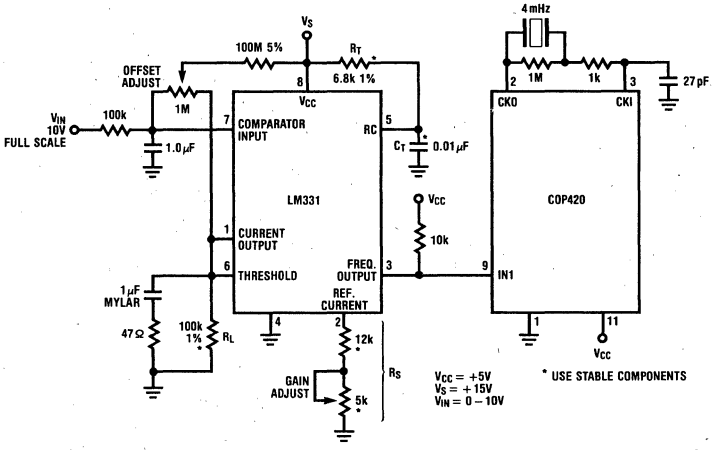



Figure 20. Basic LM331 Connection

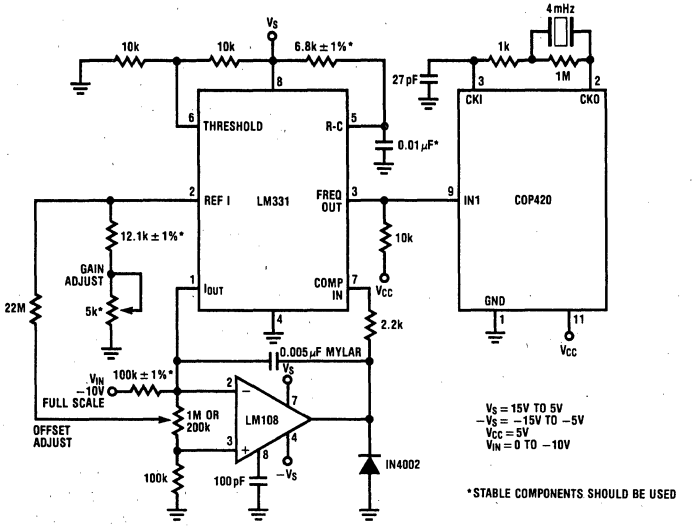



Figure 21. A to D with Precision Voltage to Frequency Converter

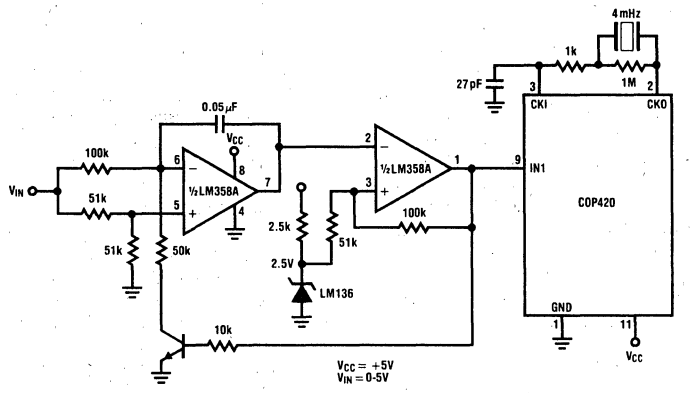



Figure 22. A to D with VCO

## VI. Successive Approximation

### A. BASIC APPROACH

The successive approximation technique is one of the more standard approaches in analog to digital conversion. It requires a counter or register (here provided by the COP420), a digital to analog converter, and a comparator. Figure 23 illustrates the basic idea with the COP420. In the most basic scheme, the counter is reset to zero and then incremented until the voltage from the digital to analog converter is equal to the input voltage. The equality is determined by means of the comparator. Figure 24 illustrates the flow chart and code for this most basic approach. The preferred approach is illustrated in Figure 25. This is the standard binary search method. The counter or register is set at the midpoint and the "delta" value set at one half the midpoint. The "delta" value is added or subtracted from the initial guess depending on the output of the comparator. The "delta" value is divided by 2 before the next increment or decrement. The method repeats until the desired resolution is achieved. While this approach is somewhat more complicated than the basic approach it has the advantage of always taking the same amount of time for the conversion regardless of the value of the input voltage. The conversion time for the basic approach increases with the input voltage. The preferred approach is almost always faster than the basic approach. The basic approach is faster only for those voltages near zero where it has only a few increments to perform.

The accuracy of the approach is governed by the accuracy of the digital to analog converter and the comparator. Thus, the result can be as accurate as one desires depending on the choice of those components. Digital to analog converters of various accuracies are readily available as standard parts. Their cost is usually in direct relation to their accuracy. The reader should refer to the National Semiconductor Data Acquisition Handbook for some possible candidates for digital to analog converters. It is not the purpose here to compare those parts. The COPS™ interface to these parts is generally straightforward and follows the basic schematics shown in Figure 23. The user should take note and make sure the input and output ports of the converter are compatible — in terms of voltages and currents — with the COPS device. This is generally not a problem as most of the parts are TTL compatible on input and output. The precautions and restrictions as to the use of any given device are governed by that device and are indicated in the respective data sheets.

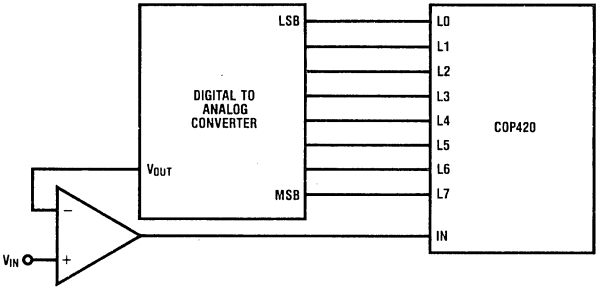



Figure 23A. Basic Parallel Implementation

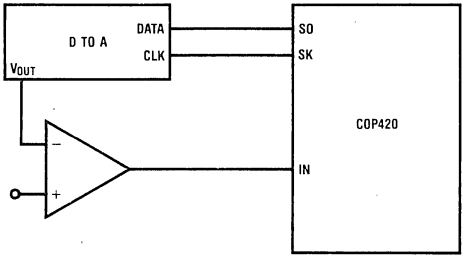



Figure 23B. Basic Serial Implementation

```
          ;8 BIT SUCCESSIVE APPROXIMATION--BASIC SCHEME
          ;COMPARATOR INPUT TO COP = IN3
          ;OUTPUTS TO D TO A ARE L7 THRU L0 WITH L7 = MSB, L0 = LSB

CONVRT:   LBI     2,14      ;SET THE RESULT VALUE TO ZERO
          STII    0
          STII    0
          LEI     4         ;ENABLE THE L PORT AS OUTPUTS
          JP      OUTPUT
INCR:     SC                ;ROUTINE FOR INCREMENTING THE RESULT VALUE
PLUS1:    CLRA
          LBI     2,14
          ASC
          NOP
          XIS
          JP      PLUS1
OUTPUT:   LBI     2,15      ;SEND THE RESULT VALUE, STORED IN 2,15-2,14 TO
          LD                ;Q AND THEREBY OUT THROUGH L
          XDS
          CAMQ
          JSR     DELAY     ;THIS IS ANY CONVENIENT ROUTINE TO MAKE SURE
                            ;THAT THE COP DOES NOT TEST THE COMPARATOR UNTIL
                            ;THE D TO A CONVERTER HAS HAD ENOUGH TIME TO DO
                            ;THE CONVERSION--THE AMOUNT OF TIME REQUIRED
                            ;IS CLEARLY DEPENDANT UPON THE D TO A CONVERTER
                            ;USED
          ININ              ;NOW READ THE COMPARATOR INPUT TO COP
          AISC    8         ;COULD SAVE A WORD IF USE G LINE AS INPUT
          JP      INCR      ;INPUT VOLTAGE STILL > CONVERTED ANALOG VOLTAGE

          ;CONVERSION DONE AT THIS POINT--THE COMPARATOR HAS CHANGED STATE
          ;HENCE, CONVERTED ANALOG VOLTAGE > INPUT VOLTAGE--SO STOP
```

Figure 24A. Code for Basic Approach of Successive Approximation

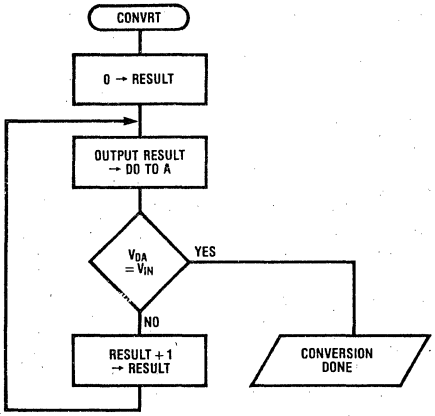



**Figure 24B. Basic Approach, Successive Approximation**

```
        ;8 BIT BINARY SEARCH SUCCESSIVE APPROXIMATION
        ;INPUT TO COP IS IN3,L BUS IS OUTPUT TO D TO A,L7=MSB,L0=LSB
        ;COMPARATOR=0 WHEN D TO A VOLTAGE > VIN, OTHERWISE = 1

BINSRH: LBI     3,14    ;SET INCREMENT = MAX VALUE/2(WILL BECOME
        STII    0       ;MAX VALUE/4 BEFORE FIRST USE)
        STII    8
        LBI     2,14    ;SET INITIAL VALUE OF RESULT TO MAX VALUE/2
        STII    0
        STII    8
        LEI     4       ;ENABLE THE L BUS AS OUTPUTS
        LBI     1,15    ;NOW SET UP THE BIT COUNTER-OVERFLOW WHEN 8 BITS
        CLRA
        AISC    9       ;DO IT THIS WAY FOR COMPATIBILITY WITH INCREMENT
OUTPUT: X       3       ;SAVE THE BIT COUNTER VALUE AND POINT TO RESULT
        LD
        XDS             ;SEND THE RESULT TO G AND HENCE TO L
        CAMQ
DIVIDE: LBI     3,15    ;DIVIDE THE INCREMENT VALUE BY 2, CAN BE DONE
DIVA:   LD              ;IN SEVERAL WAYS SINCE THIS IS A VERY SPECIAL
        AISC    8       ;PURPOSE DIVIDE FUNCTION
        JP      DIV1    ;ALSO,DO THE DIVIDE HERE TO GIVE THE D TO A TIME
        STII    4       ;TO DO THE DIGITAL TO ANALOG CONVERSION
        JP      TEST
DIV1:   AISC    4
        JP      DIV2
        STII    2
        JP      TEST
DIV2:   AISC    2
        JP      DIV3
        STII    1
        JP      TEST
DIV3:   LBI     3,14
        AISC    1
        JP      DIVA
        STII    8
        STII    0
        ;DEPENDING ON THE D TO A USED, MAY NEED MORE DELAY HERE
        ;MUST BE SURE THE RESULT IS STEADY BEFORE TEST THE COMPARATOR
TEST:   LBI     3,14
        ININ
        AISC    8       ;COULD SAVE A WORD IF USED G LINE AS INPUT
        JP      INCR
DECR:   SC              ;INPUT LESS THAN D TO A CONVERTED VOLTAGE
SUB:    LD      1       ;SUBTRACT THE INCREMENT VALUE FROM RESULT
        CASC
        NOP
        XIS     1
        JP      SUB
        JP      BITPL1
INCR:   RC              ;INPUT > D TO A CONVERTED VOLTAGE
ADD:    LD      1       ;ADD THE INCREMENT VALUE TO RESULT VALUE
        ASC
        NOP
        XIS     1
        JP      ADD
BITPL1: LBI     1,15    ;NOW INCREMENT BIT COUNTER TO SEE IF DONE
        LD
        AISC    1
        JP      OUTPUT
        ;CONVERSION DONE AT THIS POINT
```

**Figure 25A. Binary Search Successive Approximation Code**

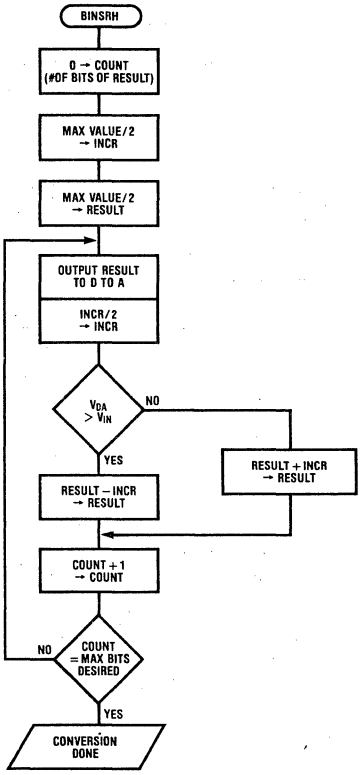



**Figure 25B. Binary Search Successive Approximation Flow Chart**

## B. SOME COMMENTS ON RESISTOR LADDERS

If the user does not wish to use one of the standard digital to analog converters, he can always build one of his own. One of the most standard methods of doing so is to use a resistor ladder network of some form. Figure 26 illustrates the basic forms of binary ladders for digital to analog converters. The figures also show the transition from the basic binary weighted ladder in Figure 26A to the standard R-2R ladder Figure 26C.

Consider Figure 26A. The choice of the terminating resistor is made by hypothesizing that the ladder were to go on ad infinitum. It can then be shown that the equivalent resistance at point X in that figure would be equal to 128R, the same value as the resistor to the least significant bit output. This fact is used to create the intermediate ladder of Figure 26B. This step is done because it is usually undesirable to have to find the multitude of resistor values required in the basic binary ladder. Thus, the modification in Figure 26B significantly reduces the number of resistor values required. As stated earlier, the resistance looking down the ladder at point X in Figure 2 is equal to the resistor connected to the binary output at that point; here the value is 2R. Remembering the objective is to minimize the number of different values required, if we simply use the same R-2R arrangement as before with a termination of 2R we get an effective resistance at point Y of Figure 26B or 0.5R. This means that a serial resistance of 1.5R is required to maintain the integrity of the ladder. If we carry this on through 8 bits, the circuit of Figure 26B results. From this it is only a small step to create the standard R-2R network. The analysis is the same as done previously.

There is absolutely no restriction that the ladders must be binary. A ladder for any type of code can be constructed with the same techniques. Ladders comparable to Figures 26A and 26B are shown in Figure 27 for a standard 8421 BCD code. With the BCD code, the input must be considered in groups of digits with four bits creating one digit. This is the direct analog of 1 binary digit per input. We need four inputs to create one decimal digit. Thus the resistor values in each decimal digit are 10 times the values in the previous decimal digit just as the resistor value for each successive binary digit was twice the value for the preceding binary digit. Note that this analysis can be easily extended to any code. The termination resistance is calculated in the same manner — assume the decimal digit groupings extend out to infinity. It can be shown that the resistance of the ladder at point X in Figure 27A is 480R. Thus Figure 27A represents the basic 8421 BCD ladder for three digit BCD number. This termination resistance will vary with where it is placed. Basically this resistance is equal to nine times (for a decimal ladder) the parallel resistance of the last digit implemented. (This relation can be shown mathematically if one desires, the multiplier is a function of the type of ladder used — multiplier = 1 for binary systems, 9 for decimal systems, etc.) Thus the termination resistance would be 48R if the network were terminated after the 2nd digit and 4.8R if the

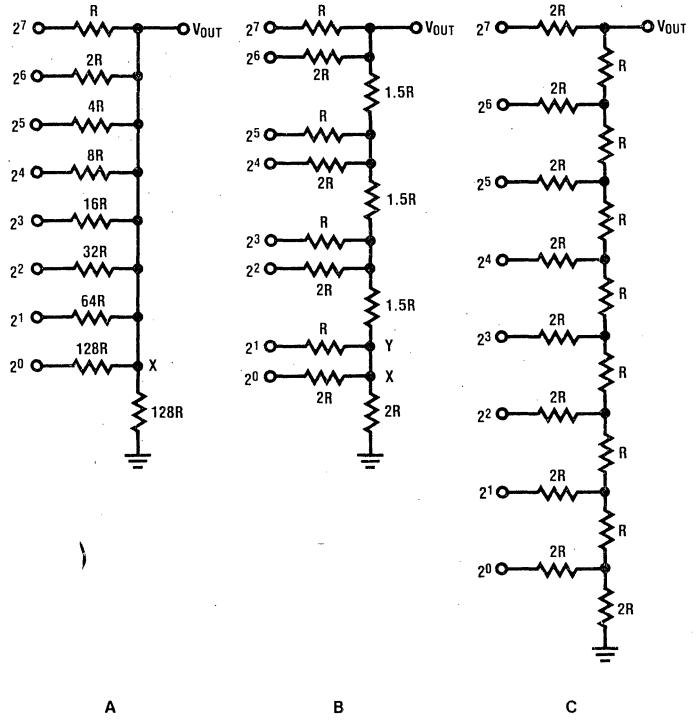



Figure 26. Binary Ladders

network were terminated after the 1st digit implemented. In Figure 27B we are attempting to use only the resistor values for one decimal digit. This means that the last terminating resistor must be a 4.8R by the analysis above. Thus at point X in Figure 27B we must have an equivalent of resistance of 4.8R. The equivalent resistance at point Y of Figure 27B, looking down from the ladder, is 0.48R. Thus the other series resistance must be 4.32R (4.8R – 0.48R). Thus the network of Figure 27B results.

Generally, ladders can be very effective tools when understood and used properly. They can be significantly more involved than indicated here. There are a number of texts and articles that cover the subject very nicely and the reader is referred to them if more information on ladder design, the use of ladders, and advanced techniques with ladders is desired.

One final note is of some interest. The ladders may be readily constructed for any type of code to create the analog voltage. Note that there is no restriction that the code, or the ladder network, be linear. Thus, effective use of ladder networks may significantly reduce system difficulties and complexities caused by the fact that the analog to digital conversion is being performed on a voltage source that changes nonlinearly, for example, a thermistor temperature probe. By using the properly designed ladder network, the nonlinearity can effectively be eliminated from consideration in the code implementation of the analog to digital conversion.

The accuracy of ladders is a direct function of the accuracy of the resistors and the accuracy of the voltage source inputs. This is obvious since the analog voltage is in fact created by means of equivalent voltage dividers created when the various inputs are on or off. It is also essential that the ladder sources be the precise same value at all inputs to the ladder network. If this is not the case, errors will be introduced. In addition, the output impedance of the voltage source should be as small as possible. The success of the ladder scheme depends on the ratios of the resistance values. Inaccuracies are introduced if those ratios are disturbed. Some possible implementations of the successive approximation approach with a ladder network used for the digital to analog conversion are

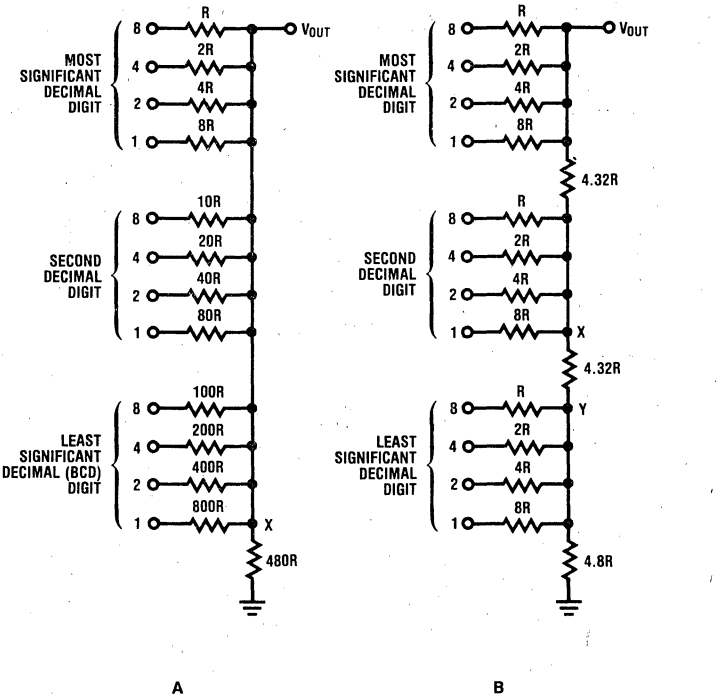
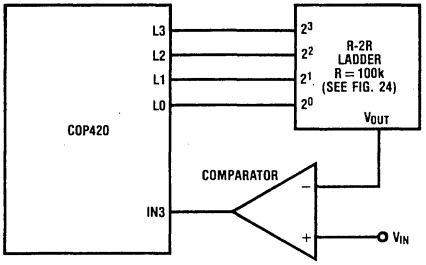
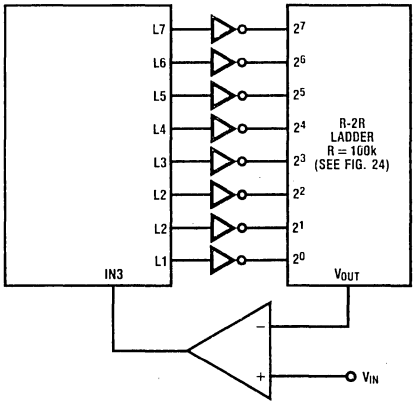



**Figure 27. 8421 BCD Ladders**

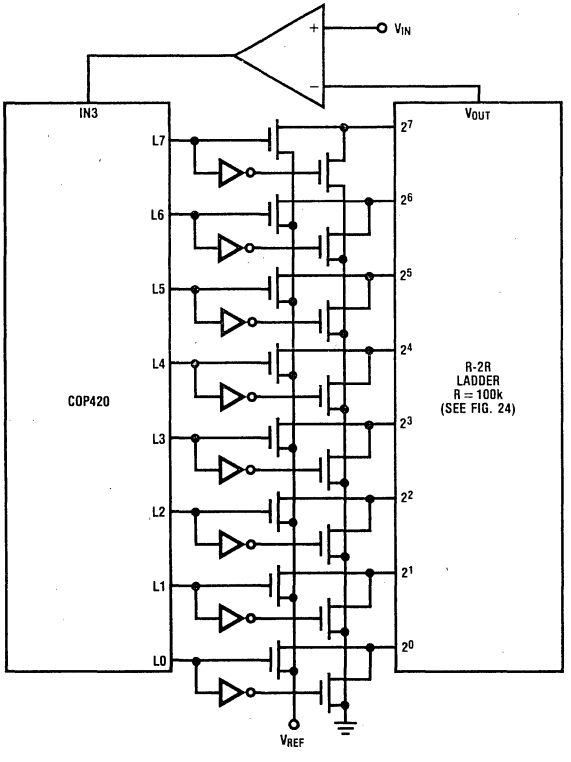
indicated in Figure 28. Note that these are functional diagrams. Feedback or hysteresis for comparator stabilization are not shown. The reader should be aware that his particular application may require that these factors be considered. Figure 28A is the simplest scheme and also the least accurate. With little or no load, the high output level of the L buffer should be very close to $V_{CC}$ and the low level close to ground. Also the output impedance of the buffers must be considered. Therefore, rather large resistor values are used — both to keep the load very small and to dwarf the effect of the output impedance. With the configuration in Figure 28A, four bit accuracy is about the best that can be achieved. By being extremely careful and using measured values, an additional bit of accuracy may be obtained but care must be used. However, the schematic of Figure 28A is very simple. Figure 28B represents the next step of improvement. Here we have placed CMOS buffers in the network. This eliminates the output impedance and reduces the level problems of the circuit of Figure 28A. The CMOS buffer will swing rail to rail, or nearly so. The accuracy of $V_{CC}$ and the




A




B




C

**Figure 28. Interfaces to Ladder Networks**

resistor network is then controlling. Using 1% resistors and holding $V_{CC}$ constant, the user should be able to achieve 7 to 8 bit accuracy without much difficulty. Remember, however, that $V_{CC}$ is one of the controlling factors. If $V_{CC}$ is ±5%, there is no point in using 1% resistors since the $V_{CC}$ tolerance swamps their effect. Figure 28C is the final and most accurate approach. Naturally enough, it is the most expensive. However, one can get as accurate as one desires. Here, an accurate reference is required. That reference is switched into the network by means of the analog switch. Alternately, ground may be connected to the input. Now the user need only consider the accuracy of the reference and the accuracy of the resistors. However, the on impedance of the switches must be considered. It is necessary to make this on impedance as low as possible so as not to alter the effective resistor values.

## VII. "Offboard" Techniques

### A. GENERAL COMMENTS

This section is devoted to a few illustrations of interfacing the COP420 to standard, stand alone analog to digital converters. These standard converters are used as peripherals to the COPS™ device. Whenever the microcontroller requires a new reading of some analog voltage, it simply initiates a read of the peripheral analog to digital converter. As a result, the accuracies and restrictions in using the converters are governed by those devices and not by the COPS device. These techniques are generally applicable to other A to D converters not mentioned here and the user should not have difficulty in applying these principles to other devices. It should be pointed out that in almost every instance, the choice of COP420 inputs and outputs is arbitrary. Obviously, when there is an 8-bit bus it is natural, and most efficient, to use the L port to interface to the bus. Generally, the G lines have been used as outputs rather than the D lines simply because the G lines are, in many instances, somewhat easier to control. The choice of input line is also free. If the interrupt is not otherwise being used, it may be possible to utilize this feature of IN1 for reading a return signal from the converter. However, this is by no means required. If there is a serial interface it is clearly more efficient to use the serial port of the COP420 as the interface. If a clock is required, SK is the natural choice.

### B. ADC0800 INTERFACE

The ADC0800 is an 8-bit analog to digital converter with an 8-bit parallel output port with complementary outputs. The ADC0800 requires a clock and a start convert pulse. It generates an end of conversion signal. There is an output enable which turns the outputs on in order to read the 8-bit result.

The reader is referred to the data sheet for the ADC0800 for more information on the device. The circuit of Figure 29 illustrates the basic implementation of a system with the ADC0800. The interface to the COP420 is straightforward. The appropriate timing restrictions on the control signals are easily met by the microcontroller.

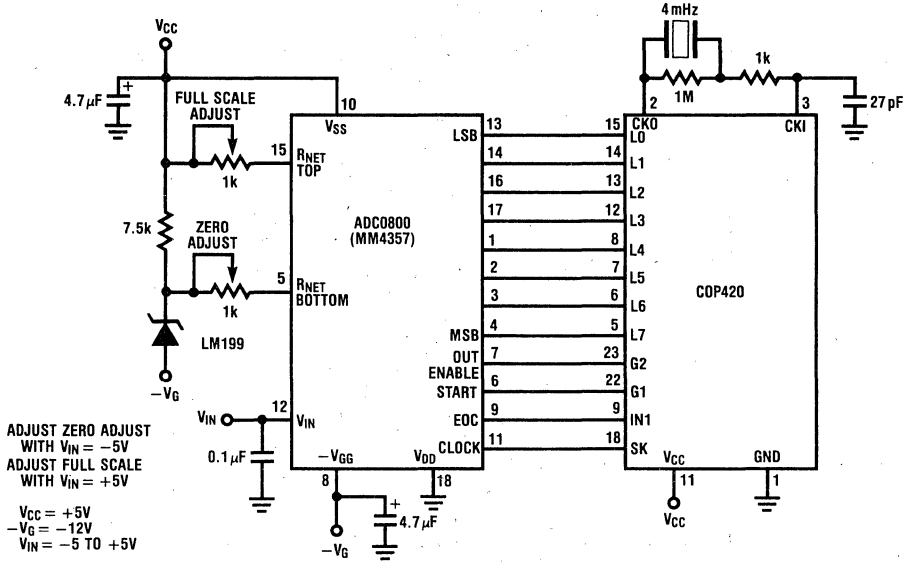



**Figure 29. Simple A/D with ADC0800**

Figure 30 is the flow chart and code required to do the interfacing. As can be seen, the overhead in the COP420 device is very small. The choice of inputs and outputs is arbitrary. The only pin that is more or less restricted is the use of SK as the clock for the converter. SK is clearly the output to use for that function as, when properly enabled, it provides pulses at the instruction cycle rate.

## C. NAKED-8™ INTERFACE

The Naked-8 family of analog to digital converters (ADC0801, ADC0802, ADC0803, ADC0804) is very easy to interface and is generally a very useful offboard converter. The interface is not significantly different from that of the ADC0800, but the Naked-8 is a much better device. The four control signals are somewhat different, although there are still four control lines. Here we have a chip select, a read, a write, and an interrupt signal. All are negative going signals. Start conversion is the anding of chip select and write. Output enable is the anding of chip select and read. The interrupt output is an end convert signal of sorts. The device may be clocked externally or an RC may be connected to it and it will generate its own clock for the conversion. In addition the device has differential inputs which allow

```
MEASUR: LEI     0       ;FLOAT THE L LINES
        SC
START2: CLRA            ;MAKE SURE SO STAYS ZERO
        XAS             ;MAKE SURE SK STAYS CLOCK
        OGI     2       ;SEND START PULSE
        OGI     0
        LBI     2,13
READ11: ININ
        AISC    14      ;WAIT FOR EOC SIGNAL
        JP      READ11
        OGI     4       ;HAVE EOC, ENABLE OUTPUTS
        INL             ;READ THE L LINES
        X
        COMP            ;CREATE PROPER POLARITY
        XDS
        COMP
        X
        OGI     0       ;DISABLE ADC0800 OUTPUT
        ;HAVE THE RESULT AT THIS POINT--USE IT IN WHATEVER
        ;MANNER IS REQUIRED BY THE APPLICATION
        LBI     2,10
        JSRP    CLRR
        JP      MEASUR
```

Figure 30A. A to D with ADC0800

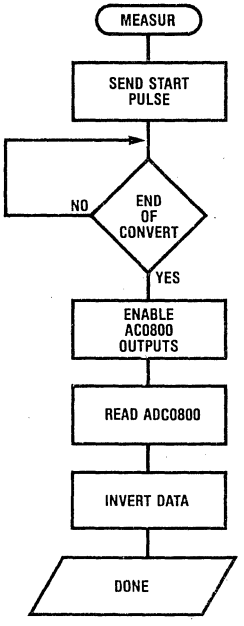



Figure 30B. ADC0800 Interface Flow

the 8-bit conversion to be performed over a given window or range of input voltages. The reader should refer to the Naked-8™ data sheet for more information. Figure 31 indicates a basic interface of the Naked-8 to the COP420. Again, the interface is simple and straightforward. The code required to interface to the device is minimal. Figure 32 illustrates the flow chart and code required to do the interface.

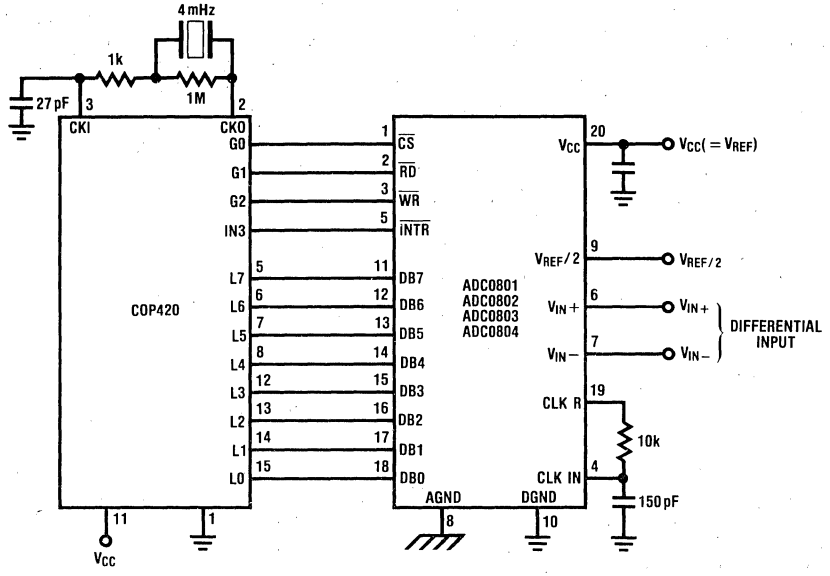



Figure 31. COP420 — Naked-8 Interface

```
          ;INTERFACE TO NAKED 8(TM)
          ;
NAKED:    OGI       15          ;SET ALL G LINES HIGH(USUALLY DONE AT
                                  ;POWER UP
          LEI       0           ;TRI STATE THE L LINES FOR READING
LOOP:     OGI       14          ;SEND CHIP SELECT LOW(CS BRACKETS OTHER SIGNAL)
          OGI       10          ;CS LOW AND WR LOW = START CONVERSION
          OGI       14          ;RAISE WR
          OGI       15          ;RAISE CS, NAKED 8 IS NOW CONVERTING
LOOP2:    ININ                  ;WAIT FOR THE INTR SIGNAL--COULD SAVE THIS TES
          AISC      8           ;IF USED IN1 AND THE INTERRUPT FEATURE OF COP4
          JP        READ        ;INTR IS LOW, DATA IS READY
          JP        LOOP2
READ:     LBI       0,0         ;SET UP RAM LOCATION FOR READ
          OGI       14          ;SEND CS
          OGI       12          ;SEND CS AND READ = OUTPUT ENABLE
          NOP                   ;WAIT-NEED WAIT ONLY 125NS, BUT 1 CYCLE IS MIN
                                  ;TIME WE CAN WAIT
          INL                   ;READ THE L LINES
          OGI       15          ;TURN OFF THE NAKED 8--CS AND RD HIGH
          ;
          ;DONE AT THIS POINT, DO WHATEVER IS REQUIRED  WITH THE RESULT
          ;
```

Figure 32A. COP420/Naked-8 Sample Interface Code

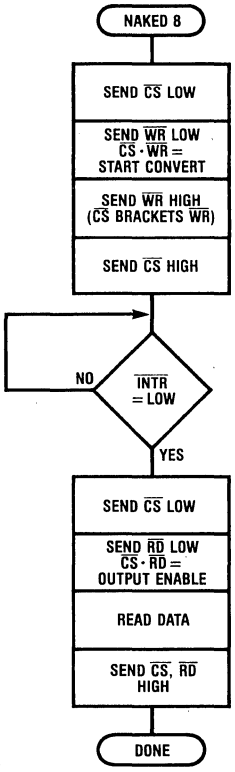



Figure 32B. COP420/Naked-8 Interface Flow

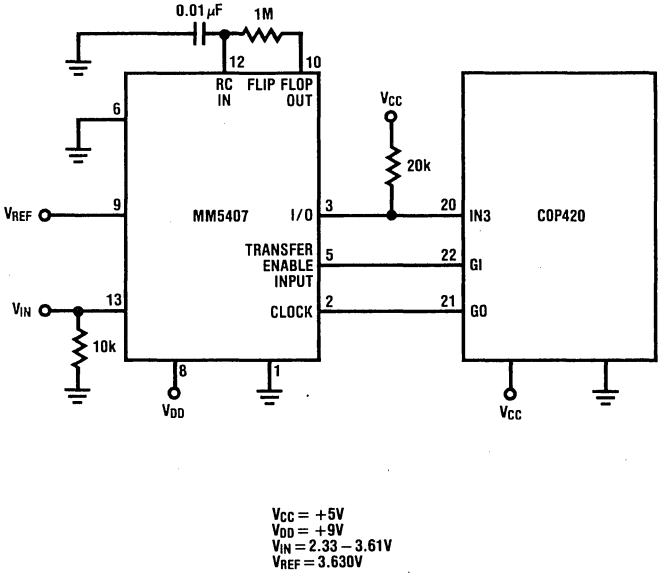



Figure 33. MM5407 Interface

## D. THE MM5407 AS AN A/D CONVERTER

The MM5407 is a digital thermometer usually used in conjunction with the MM5406 digital clock. However, the MM5407 can make a very effective analog to digital converter. The heart of the MM5407 is, in fact, an analog to digital converter. The device is designed to interface directly with the LM134 temperature transducer which produces an output voltage related to temperature. The relationship is 10mV per degree Kelvin. The MM5407 is specified to operate from −40°C to +88°C (233°K to 361°K). The device provides a serial output with the result in either centigrade or fahrenheit. The accuracy is ±2°F.

Now, translating all of this into the pertinent information that we need we get the following: The MM5407 will perform an analog to digital conversion for input voltages in the range of 2.33 volts to 3.61 volts. The result is accurate to about ±10 millivolts. This translates to an accuracy of 7 bits ±1 bit. The interface, as shown in Figure 33 is not complex. Note that here SK is not being used for the clock because SK is too fast. The clock input on the MM5407 has an upper limit of 10kHz. Also because of the speed, we are using IN3 rather than serial in as the input from the MM5407. Note also that the MM5407 is a nine volt device although the interface signals are TTL compatible. The COP420 is a 5 volt device. However, the COP420L will run at 9 volts and thereby remove a requirement for two power supplies. If the user system has dual supplies, the dual supply requirement is not serious.

Once the data is read into the COPS™ device, the processing required is simple. One need only add 273 to the number received (if the MM5407 is operated in the Centigrade mode) to create the proper voltage value. Obviously, if a different range is desired, it would be possible to do some scaling at the input of the MM5407 to create the proper voltage. The COPS device would then have to account for this scaling — generally a straightforward task.

```
        ;CODE FOR MM5407/COP420 AS A TO D CONVERTER
        ;G0 AND G1 ARE HIGH ON ENTRY TO THE ROUTINE

MM5407: CLRA            ;RUN A FEW CLOCKS TO DO THE CONVERSION
        AISC    8
        LBI     2,12
LOOP:   X
        JSRP    CLOCK2
        NOP
        LD
        AISC    1
        JP      LOOP
        STII    0       ;NOW CLEAR OUT THE MEMORY FOR READING
        STII    0
        STII    0
        STII    0       ;0 TO 2,12 THRU 2,15
START:  LBI     2,12    ;NOW SEND START TRANSMIT SIGNAL AND MAINTAIN
        JSRP    CLOCK1  ;TIMING
        NOP
        JSRP    CLOCK2
        NOP
        JSRP    CLOCK2
        NOP
        JSRP    CLOCK2
        NOP
READ:   JSRP    CLOCK2  ;NOW READY TO READ THE DATA(16 BITS)
        SMB     3       ;ALLOW FOR THE COMPLEMENT DATA ON THE READ
        JSRP    CLOCK2  ;I.E.,COMPLEMENT THE INFO. WHEN READING IT
        SMB     2
        JSRP    CLOCK2
        SMB     1
        JSRP    CLOCK2
        SMB     0
        LD              ;NOW TEST TO SEE IF DONE
        XIS
        JP      READ    ;NOT YET FINISHED
        LBI     2,13    ;NOW JUGGLE THE DATA TO PUT IT IN MORE DESIRAB
        CLRA            ;FORM--MINUS/BLANK,TENS,UNIT
        X               ;IGNORE 2,12 BECAUSE WE KNOW IS CENTIGRADE MODE
        LBI     2,15    ;REFER TO MM5407 DATA SHEET
        X               ;INFO WAS IN FORM: UNITS,TENS,MINUS/BLANK
        LBI     2,13
        X
        LBI     2,15    ;NOW TEST TO SEE IF IS MINUS
        X               ;ACCUMULATOR IS ZERO PRIOR TO THIS EXCHANGE
        AISC    5       ;TEST FOR THE MINUS CODE
        JP      ADD273
COMP1:  SC              ;IS MINUS,TAKE TENS COMPLEMENT OF NUMBER
        LBI     2,13    ;ALSO, ZERO IS IN MINUS POSITION
COMP2:  CLRA
        X
        CASC
        ADT
        XIS
        JP      COMP2
ADD273: LBI     1,13    ;NOW SET UP TO ADD 273 TO THE RESULT
        STII    3
        STII    7
        STII    2
```

Figure 34A. MM5407/COP420 A/D Interface Code

```
          RC
          LBI     1,13
ADDLP:    LD      3
          AISC    6
          ASC
          ADT
          XIS     3
          JP      ADDLP
;FINISHED AT THIS POINT, DO ANY REQUIRED SCALING,ETC. HERE
          RET
          .PAGE   2           ;THE REQUIRED SUBROUTINES HERE
CLOCK1:   CLRA
          OGI     0           ;SEND CLOCK AND START SIGNAL LOW
          JP      CLK
CLOCK2:   OGI     2           ;SEND CLOCK ONLY LOW
          CLRA
CLK:      AISC    3           ;MAKING SIMPLE TIMING LOOP-HERE ADJUSTING FOR
          JP      .-1         ;TOTAL PERIOD = 100us(25 CYCLE TIMES AT 4us
          AISC    4           ;INSTRUCTION CYCLE TIME)--HERE USING 13 CYCLE
          JP      .-1         ;TIMES ON, 12 CYCLE TIMES OFF
          OGI     3           ;SET CLOCK BACK HIGH
          NOP                 ;THESE NOP's FOR TIMING ONLY
          NOP
          NOP
          NOP
          ININ                ;READ THE INPUT LINE(I3)
          AISC    8
          RET
          RETSK
```

**Figure 34A. MM5407/COP420 A/D Interface Code, cont'd**

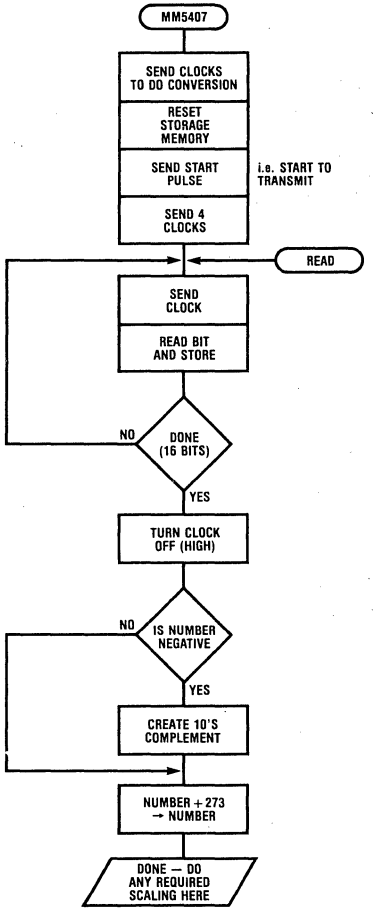



**Figure 34B. MM5407 as A/D Converter Flow Chart**

## VIII. Conclusion

Several analog to digital techniques using the COPS™ family have been presented. These are by no means the only techniques possible. The user is limited only by his imagination and whatever parts he can find. The COPS family of parts is extemely versatile and can readily be used to perform the analog to digital conversion in almost any method. Generally, those techniques where the COPS device is doing the counting or timekeeping are slow. However, those techniques are generally slow inherently. The fastest methods are those where the conversion is being done offboard and the COPS device is merely reading the result of the conversion when required. Also, an attempt has been made to illustrate the lower cost techniques of analog to digital conversion. This, by itself, restricts most of the techniques described to about 8-bits accuracy. As was mentioned several times, the greater the accuracy that is desired the more accurate the external circuits must be. Ten and twelve-bit accuracies, and more, require references that are accurate. These get very expensive very rapidly. There is nothing inherent in the COPS devices that prevents them from being used in accurate systems. The precautions are to be taken in the system regardless of the microcontroller. The only problem is that, in those accurate systems where the COPS device is doing the timekeeping and counting, this increased accuracy is paid for by increased time to perform the conversion.

Several devices have been used in conjunction with the COPS device in the previous sections. It is again recommended that the user refer to the specific data sheets of those devices when using any of those circuits. It must again be mentioned that the standard precautions when dealing with analog signals and circuits must be taken. These are described in the National Semiconductor Linear Applications Handbook and in the data sheets for the various linear devices. These precautions are especially significant when greater accuracy is desired.

The COPS family of microcontrollers has shown itself to be very versatile and powerful when used to perform analog to digital conversions. Most techniques are code efficient and the microcontroller itself is almost never the limiting factor. It is hoped that this document will provide some guidance when it is necessary to perform analog to digital conversion in a COPS system.

## IX. References


1. "Digital Voltmeters and the MM5330", National Semiconductor Application Note AN-155.
2. Walker, Monty, "Exploit Ladder Network Design Potential." Part One of two part article on ladder networks. Magazine and date unknown.
3. Wyland, David C., "VFC's give your ADC design high resolution and wide range." *EDN*, Feb. 5, 1978.
4. Redfern, Thomas P., "Pulse Modulation A/D Converter" *Society of Automotive Engineers Congress and Exposition Technical paper #780435,* March 1978.
5. National Semiconductor Linear Applications Handbook, 1978.
6. National Semiconductor Linear Databook, 1980.
7. National Semiconductor Data Acquisition Handbook, 1978.

# COPS™ Television Controller

National Semiconductor
COP Note 2
Brett K. Nelson
May 1980

## Introduction

As part of National Semiconductor's continuing effort to define and implement a full spectrum of COPS Television Controllers (CTCs), this document will describe progress made in programming a COP420 to serve as a prototype 'low-end' CTC. Used in conjunction with an MM5439 Phase Locked Loop (PLL) and an MM5450 display driver, this processor allows a television receiver to have the following functions:

1. Frequency Synthesis Tuning
2. Keyboard Scan and Decode
3. MM53126 Format Serial Decode
4. 64 Level Analog Outputs
5. Direct Channel Entry
6. Channel and Fine Tune Slewing
7. Analog Output Slewing
8. LED Channel Display
9. Last Channel Memory

## System Overview

Shown in Figure 1, the heart of the CTC prototype hardware is the COP420 itself. This particular member of National's COPS family of 4-bit microcontrollers has 1024 bytes of program memory, 64 digits of scratch-pad RAM, 24 input and output pins, and an efficient 49-member instruction set. It is the workhorse of the television tuning system and provides the processing power to scan the keyboard, decode the serial input, run the channel display, and control the PLL. System capabilities may be enhanced or scaled-down for different markets simply by changing the processor's algorithms. This flexibility combined with low-cost makes the COPS family, and in particular the COP420, a standout in the field of high-volume, low-to-medium range television controllers.

The MM5439 PLL is of next importance in the prototype system. Originally designed for the European Microprocessor Television Controller (MTC) market, the 5439 offers capabilities found in traditional PLL circuits as well as general purpose input and output pins and 6 pulse-width modulation D/A converters. This allows the COP420 to use it to band-switch the UHF and VHF tuners in addition to providing analog outputs for controlling television parameters such as volume, brightness, and color. The MM5439 operates with a 14-bit code and is capable of resolving the RF spectrum into 64 kHz steps; more than adequate for U.S. Television receivers.

The serial input of Figure 1 is generated by using an MM53126 infrared remote control circuit. The MM53126 scans and decodes a key closure and provides serial data to drive infrared transmitter diodes. At the receiving end, the infrared signal must be detected and amplified to provide a digital signal for the COP420. The COPS device provides the intelligence to receive the serial data

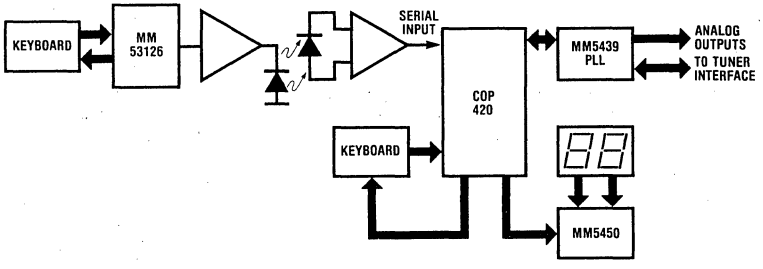



Figure 1. CTC Block Diagram

and to route program control just as if the key entry originated from the main keyboard.

The third circuit shown in Figure 1 is the MM5450 display driver. The 5450 is a direct drive, serial input, 35-segment LED driver. Due to its serial nature, it is best interfaced to the COPS' serial output port. The 5450 is gaining popularity because of its low-cost, adjustable high-current outputs, and low-noise non-multiplexed display format. Its sole duty in the system is to display the current channel number.

## Hardware Description

Utilizing the MM5439 as the system PLL dictated the basic structure of much of the prototype circuitry. The MTC series of components were designed to be MICROBUS™ compatible. That is, they were designed to connect to an 8-bit bi-directional data bus, address lines, and control strobes. The COPS™ family of processors does not possess a traditional bus structure, and to interface to a parallel bus device such as an MM5439 requires that COPS inputs and outputs emulate the data, address, and control bus functions. Figure 2 illustrates the use of the COPS L pins as the data bus, the G port for addressing, SK as a read strobe, SO as a write strobe, and DO as chip select.

Figure 2 also details the 5439 D/A, band-switching, and oscillator circuitry. The D/A interface is a simple capacitor integrator that requires a current source from within the receiver chassis. UHF/VHF band-switching is accomplished by using 3 general purpose open-collector outputs to drive dual transistor 24 volt buffers. The one transistor 4.0 MHz crystal oscillator also shown provides the stable reference needed by the PLL. In addition, it is used to generate a 4-microsecond instruction cycle within the COP420. This speed is necessary to insure that pulse-position-modulated (PPM) signals coming from the MM53126 are properly decoded.

The MM5439 and UHF/VHF tuner interface shown in Figure 2 is somewhat more complicated. By comparing the UHF/VHF local oscillator to the 4 MHz system clock, the 5439 generates two negative-going signals that are designed to raise or lower the varactor tuning voltage, and thus close the frequency synthesis loop. To accomplish this an LF351 is configured as a differential integrator to generate the tuning voltage. The single-pole filter on the output is to minimize transients. The PLL NMOS circuitry in the 5439 is not fast enough to handle the tuner local oscillator directly, so two counters are used to divide this frequency down. The SDA2001 ECL prescaler divides the frequency first by 64, and then the 74LS169 alternately divides by 15 or 16 under 5439 control.

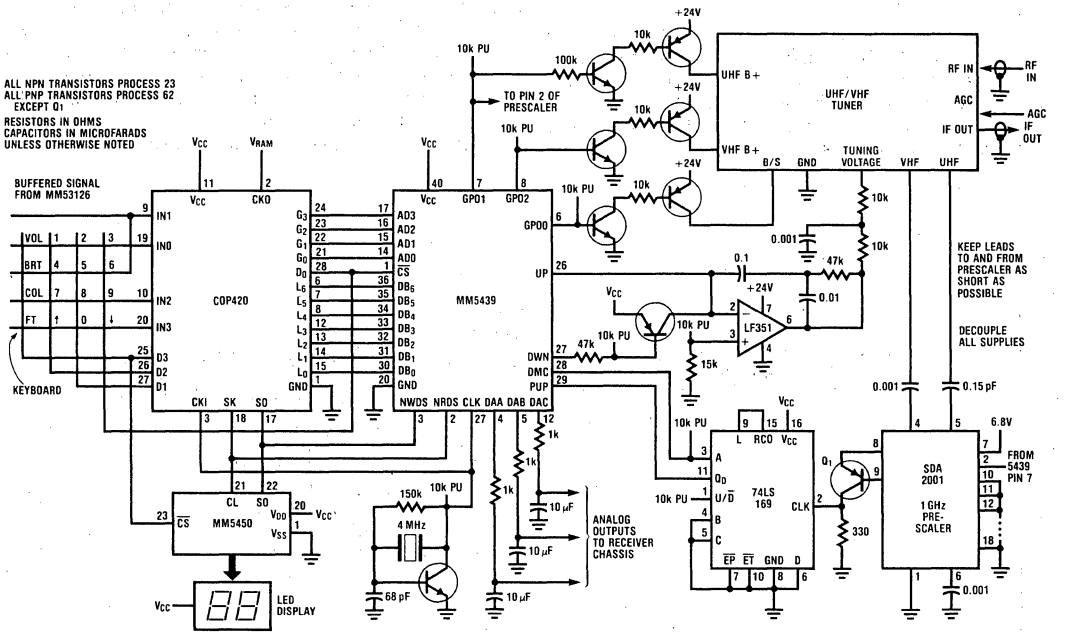



Figure 2. Low-End CTC Schematic

## Software Description

The major features of the software written for this low-end CTC implementation are described in the flowchart of Figure 3. Readily observable items of interest are the initialization, serial-input, delay, and instruction decode portions of the program. The function blocks comprising the PLL code calculations, serial processing, and display routines are less noticeable, but worthy of additional mention. They will now be summarized.

To successfully tune the television receiver a 14-bit code must be presented to the MM5439 PLL. This 14-bit binary code is calculated from current BCD channel number using the following equation:

PLL CODE = CHANNEL NUMBER * 6 MHz + BIAS

The variable marked BIAS is necessary because there are gaps between channel groups in the American television RF spectrum. BIAS will have different values for the channel ranges 2-4, 5-6, 7-13, and 14-83.

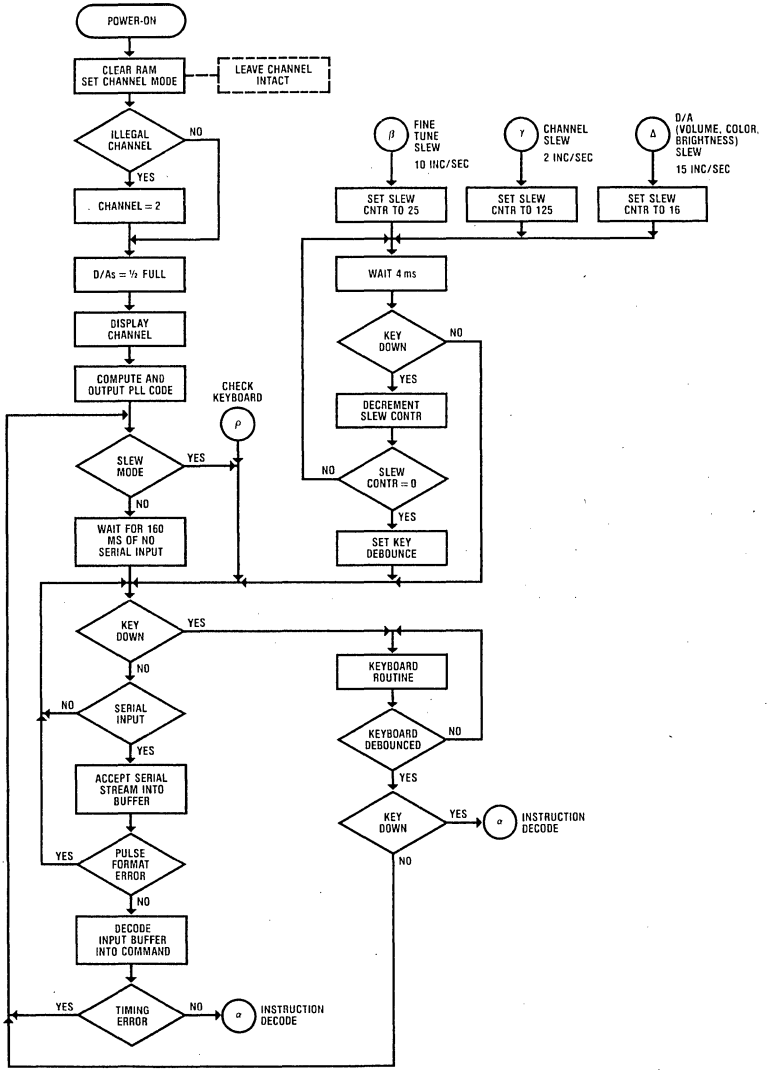



Figure 3. CTC Major Program Flow

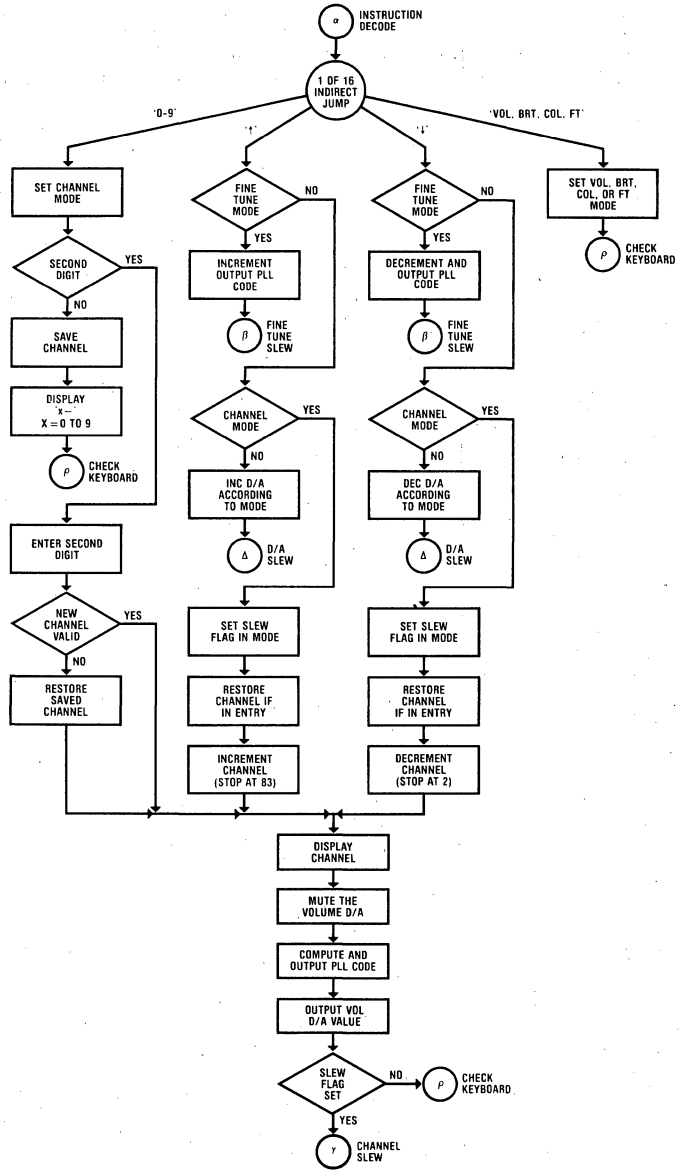



**Figure 3. CTC Major Program Flow (cont'd)**

The most time critical software operation encountered was processing the remote serial input stream. Speed considerations necessitated that this routine be broken into two portions, reading and decoding. Reading the stream required that the time between each pulse in the 14-bit code (counting start and stop bits) be saved in a unique memory location. Figures 4 and 5 illustrate the pulse timing and serial format. Only after all 14 bits were received could the timing be analyzed for validity and converted into a parallel code. Because the MM53126 generates a continuous stream of pulse packages during key depression, a form of debouncing was also needed on the input so only the first packet was decoded as an instruction.

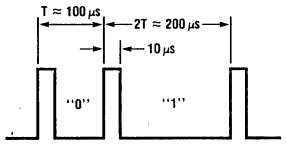



**Figure 4. Pulse-Position-Modulation (PPM) Timing**

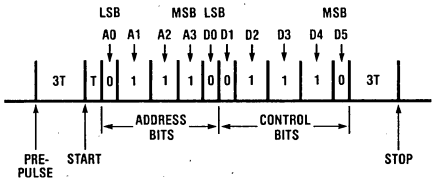



**Figure 5. Format of Remote Control Signal 0111001110**

The keyboard routine scans the key contacts by sweeping a logic low through the column outputs and checking for a resulting low on the row inputs. Once a key closure is sensed, it is converted into a unique 1 of 16 code and acted upon. It must then be released 64 milliseconds before a new key may be processed.

The last major routine shown only as a function block in the flowchart is the MM5450 display interface routine. In preparation to passing segment data to the 5450, the COP420 must first convert each digit of the channel number into its seven-segment display equivalent and place that information in a buffer. The final part of the display routine is simply serializing that buffer along with a start bit to the MM5450.

As previously stated, the COP420 has 64 digits of scratchpad RAM. Well designed data structures within this RAM will optimize overall program efficiency. With this in mind, the CTC structures were defined and assigned to particular positions in memory. Table 1 breaks down the program data structures and lists the number of 4-bit digits needed for each. RAM efficiency for this program was 39/64 or approximately 60 percent.

**Table 1. CTC RAM Allocation**

| Data Description | Digits Used |
|---|---|
| PLL Code and band data | 5 |
| Display and PLL word area | 5 |
| Remote input buffer | 13 |
| Remote command buffer | 3 |
| D/A mirror values | 6 |
| Current channel | 2 |
| Channel storage | 2 |
| Flags | 2 |
| Key decoding | 2 |
| Misc. | 2 |
| **Total** | **39** |

Listed in Table 2 are the major routines in the low-end CTC program and their respective ROM usage. ROM efficiency in this case would be 780/1024 or 76 percent.

**Table 2. CTC ROM Allocation**

| Routine Description | Bytes Used |
|---|---|
| Initialization | 50 |
| PLL code calculation | 80 |
| Increment, decrement, PLL I/O | 130 |
| Remote input | 80 |
| Remote input decoder | 20 |
| Keyboard | 100 |
| MM5450 display | 50 |
| 7-segment look-up table | 10 |
| Channel check | 20 |
| Slew control | 40 |
| PLL fine tune | 20 |
| Instruction decoding and main loop | 180 |
| **Total** | **780** |

## Conclusions

A COP420 has been shown to be ideal in performing the functions of a low-end television controller. Manufacturers integrating COPS devices into their television receiver designs would benefit from cost and capability advantages. Due to the fact that ROM and RAM are under utilized in the software described, it would be logical and cost-effective from a product viewpoint to expand the low-end concept and take full advantage of the COP420 by incorporating mid-range features into the controller software. Conversely, a lesser member of the COPS family could perform a subset of the functions presented in more cost-driven applications.

# SIO Input/Output Register Description

National Semiconductor
COP Brief 1
May 1980

## Contents


- Logical Operation
- Software Debug
- Serial Out During Breakpoint
- Serial Out During Trace
- Binary Counter During Breakpoint
- General
- Using SIO as temporary storage


## COP400 Serial SIO Register

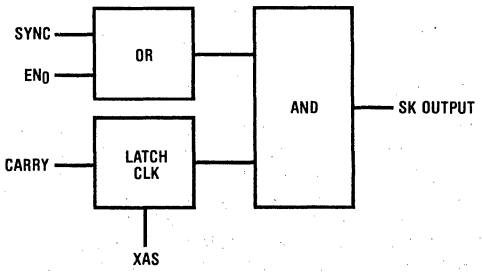
The general operation of the SIO port is treated in the COP400 data sheet. A more detailed look at the internal circuitry, as well as software debug, will be presented in this brief.

## Logical Operation

It is important to examine the logical diagram of the SIO and SK circuitry to fully understand the operation of this I/O port. The output at SK is a function of SYNC, $EN_0$, CARRY, and the XAS instruction.

If CARRY had been set and propogated to the SKL latch by the execution of an XAS instruction, SYNC is enabled to SK and can only be overridden by $EN_0$. Trouble could arise if the user changes the state of $EN_0$ without paying close attention to the state of the latch in the SK circuit.

If the latch was set to a logical high and the SIO register enabled as a binary counter, SK is driven high. From this state, if the SIO register is enabled as a serial shift register, SK will output the SYNC pulse immediately, without any intervening XAS instruction.




**Logical Diagram of SK Circuit**

## Software Debug of Serial Register Functions

In order to understand the method of software debug when dealing with the SIO register, one must first become familiar with the method in which the COPS Product Development System (PDS) BREAKPOINT and TRACE operations are carried out. Once these operations are explained, the difficulties which could arise when interrogating the status of the SIO register should become apparent.

## Serial Out During BREAKPOINT

When the PDS BREAKPOINTs, the COPS user program execution is stopped and execution of a monitor-type program, within the COP device is started. At no time does the COP part "idle". The monitor program loads the development system with the information contained in the COP registers.

Note also that single-step is simply a BREAKPOINT on every instruction.

If the COP chip is BREAKPOINTed while a serial function is in progress, the contents of the SIO register will be destroyed. By the time the monitor program dumps the SIO register to the PDS, the contents of the SIO register will have been written over by clocking in SI. To inspect the SIO register using BREAKPOINT an XAS must be executed prior to BREAKPOINT, therefore the SIO register will be saved in the accumulator.

An even more severe consequence is that the monitor program executes an XAS instruction to get the contents of the SIO register to the PDS. Therefore the SK Latch is dependent on the state of the CARRY prior to the BREAKPOINT. In order to guarrantee the integrity of the SIO register one must carefully choose the position of the BREAKPOINT address.

As can be seen, it is impossible to single-step or BREAKPOINT through a serial operation in the SIO register.

## Serial Out During TRACE

In the TRACE mode, the user's program execution is never stopped. This mode is a real-time description of the program counter and the external event lines, therefore the four external event lines can be used as logic analyzers to monitor the state of any input or output on the COPS device. The external event lines must be tied to the I/O which is to be monitored. The state of these I/O (External Event lines) is displayed along with the TRACE information. The safest way to monitor the real-time state of SO is to use the TRACE function in conjunction with the External Event lines.

## Binary Counter During BREAKPOINT

Since the COPS chip is executing a Monitor Program during BREAKPOINT the SIO register is still active. In the Binary Counter mode SIO register will decrement on every negative transition of the SI line providing the pulse

stays low for at least two instruction cycles. However, if the pulse on SI occurs when the monitor is interrogating the SIO register, an erroneous situation may occur.

### General

During a BREAKPOINT operation data is transmitted to the PDS over the SKIP output on the COP402.

Notice that the D register is not contained in the Auto-Print options. The reason for this is that the contents of D cannot be read via COP software. These may be monitored by the External Event lines in the trace mode.

### Temporary Storage

It is sometimes desirable to temporarily store the value of the accumulator. This can be done by designating a RAM digit and doing an exchange operation. If the user can assure that the SIO register is in the binary counter mode and that SI is at a constant state, the SIO register may be used as a temporary storage location. This is advantagious because the storage and retrieval is accomplished by the single byte XAS instruction and does not require the use of a RAM digit. The use of the SIO register as a binary counter is not available on the COP420C (CMOS version of the COP420), for this reason the SIO register may not be used as temporary storage.

# Easy Logarithms for COP400

National Semiconductor
COP Brief 2
May 1980

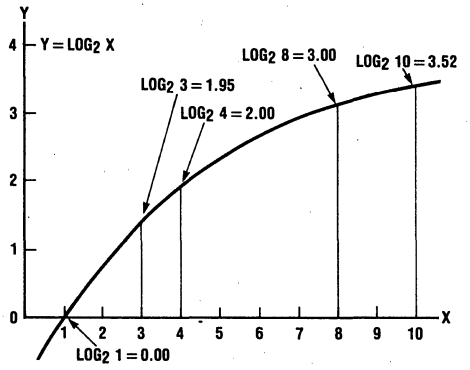
Logarithms have long been a convenient tool for the simplification of multiplication, division, and root extraction. Many assembly language programmers avoid the use of logarithms because of supposed complexity in their application to binary computers. Logarithms conjure up visions of time consuming iterations during the solution of a long series. The problem is far simpler than imagined and its solution yields, for the applications programmer, the classical benefits of logarithms:

1) Multiplication can be performed by a single addition.
2) Division can be performed by a single subtraction.
3) Raising a number to a power involves a single multiply.
4) Extracting a root involves a single divide.

When applied to binary computer operation logarithms yield two further important advantages. First, a broad range of values can be handled without resorting to floating point techniques (other than implied by the characteristic). Second, it is possible to establish the significance of an answer during the body of a calculation, again, without resorting to floating point techniques.

Implementation of base$_{10}$ logarithms in a binary system is cumbersome and unnecessary since logarithmic functions can be implemented in a number system of any base. The techniques presented here deal only with logarithms to the base$_2$.

A logarithm consists of two parts: an integer characteristic and a fractional mantissa.




| | CHARACTERISTIC | MANTISSA |
|---|---|---|
| LOG$_2$ 3 = | 1 | 0.95 |
| LOG$_2$ 4 = | 2 | 0.00 |
| LOG$_2$ 8 = | 3 | 0.00 |
| LOG$_2$ 10 = | 3 | 0.52 |

**Figure 1. The logarithmic function and some example values**

In figure 1 some points on the logarithmic curve are identified and evaluated to the base$_2$. Notice that the characteristic in each case represents the highest even power of 2 contained in the value of X. This is readily seen when binary notation is used.

| $X_{10}$ | $X_2$ $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ | $Log_2$ X Characteristic | $Log_2$ X Where X = Even Power of 2 |
|---|---|---|---|---|---|---|---|
| 3 | 0 | 0 | 0 | 1▲ | 1 | 1 | |
| 4 | 0 | 0 | 1▲ | 0 | 0 | 2 | 010.0000 |
| 8 | 0 | 1▲ | 0 | 0 | 0 | 3 | 011.0000 |
| 10 | 0 | 1▲ | 0 | 1 | 0 | 3 | |

**Figure 2. Identification of the Characteristic**

In Figure 2 each point evaluated in Figure 1 has been repeated using binary notation. An arrow subscript indicates the highest even power of 2 appearing in each value of X. Notice that in X = 3 the highest even power of 2 is $2^1$. Thus the characteristic of the $\log_2 3$ is 1. Where X = 10 the characteristic of the $\log_2 10$ is 3.

To find the $\log_2$ X is very easy where X is an even power of 2. We simply shift the value of X left until a carry bit emerges from the high order position of the register. This procedure is illustrated in Figure 3. This characteristic is found by counting the number of shifts required and subtracting the result from the number of bits in the register. In practice it is easier to begin with the number of bits and count down once prior to each shift.

| Counter For Characteristic | Value of X in Binary | |
|---|---|---|
| 1000 | 0000 1000 | Initial |
| 0111 | 0001 0000 | First Shift |
| 0110 | 0010 0000 | Second Shift |
| 0101 | 0100 0000 | Third Shift |
| 0100 | 1000 0000 | Fourth Shift |
| 0011 | 0000 0000 | Fifth Shift |
| **Characteristic** | **Mantissa** | **Final** |
| 011. | 0000 0000 | $Log_2$ X = 3.00 |

**Figure 3. Conversion to Base$_2$ Logarithm by Base Shift**

Examination of the final value obtained in Figure 3 reveals no bits in the mantissa. The value 3 in the characteristic, however, indicates that a bit did exist in the $2^3$ position of the original number and would have to be restored in order to reconstruct the original value (antilog).

The log of any even power of 2 can be found in this way:

| Decimal | Binary | $Log_2$ |
|---|---|---|
| 128 | 10000000 | 0111.00000000 |
| 64 | 01000000 | 0110.00000000 |
| 32 | 00100000 | 0101.00000000 |
| 4 | 00000100 | 0010.00000000 |
| 2 | 00000010 | 0001.00000000 |
| 1 | 00000001 | 0000.00000000 |

**Figure 4. Base$_2$ Logarithms of Even Powers of 2**

A simple flow chart, and program, can be devised for generating the values found in the table and, as will be apparent, a straight line approximation for values that are not even powers of 2. The method, as already illustrated in Figure 3, involves only shifting a binary number left until the most significant bit moves into the carry position. The characteristic is formed by counting. Since a carry on each successive shift will yield a decreasing power of 2, we must start the characteristic count with the number of bits in the binary value (x) and count down one each shift.

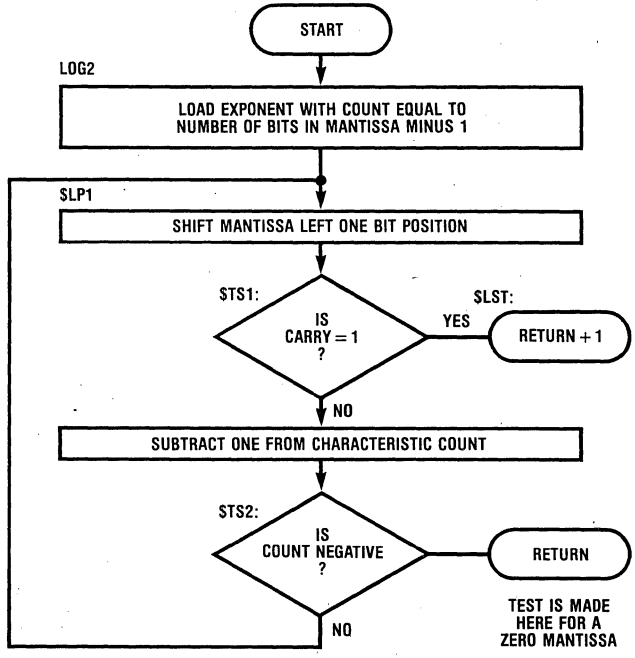



**Figure 5. Log Flowchart**

```
COP CROSS ASSEMBLER     PAGE:  1
LOGS

    1                           ; TITLE LOGS                ; BINARY LOGARITHMS
    2
    3           01A4                        . CHIP 420
    4
    5                           ; ------> CONVERT TO LOGARITHM <------;
    6                           ;
    7                           ;                   RAM ASSIGNMENT
    8                           ;
    9                           ; DIGIT:    15  14  13  12  11  10  09  08  07  06  05  04  03  02  01  00
   10                           ;
   11                           ; REG 0                                     CH  HM  LM                  TEMP
   12                           ;
   13                           ; REG 1                         CH  HM  LM                              TEMP
   14                           ;
   15                           ; REG 2                             TEMP                            CH  HM  LM
   16                           ;
   17                           ; REG 3                                         TEMP                CH  HM  LM
   18                           ;
   19                           ;
   20                                       . LOCAL
   21                           ;
   22                           ; CH, HM, LM REPRESENT ANY THREE SEQUENTIAL MEMORY DIGITS. THEY
   23                           ; MAY BE DEFINED IN ANY REGISTER. THE SYMBOLIC NOTATION CH, HM,
   24                           ; AND LM ARE USED FOR ADDRESSING TO ALLOW USER FLEXIBILITY.
   25                           ; UPON ENTRY TO THE ROUTINE HM AND LM CONTAIN THE HI AND LO
   26                           ; OF SOME VALUE X. THE MEMORY POINTER MUST CONTAIN THE ADDRESS
   27                           ; OF THE CHARACTERISTIC (CH). THE CONTENTS OF THIS LOCATION ARE
   28                           ; IGNORED AND ARE LOST DURING EXECUTION.
   29                           ;
   30                           ; UPON EXIT CH, HM, LM CONTAIN A STRAIGHT LINE APPROXIMATION OF
   31                           ; THE LOG BASE 2 OF X. CH = CHARACTERISTIC  HM = HI ORDER MANTISSA
   32                           ; LM = LO ORDER MANTISSA. AN 8 BIT MEMORY AREA (TEMP) IS USED IN
   33                           ; THE REGISTER OPPOSITE DURING THE CORRECTION OF A STRAIGHT
   34                           LINE APPROXIMATION OF A LOG OR AN ANTILOG.
   35                           ;          A TEST IS MADE FOR X = 0. IF THE VALUE OF X
   36                           ; IS NOT ZERO AN INSTRUCTION IS SKIPPED UPON RETURN
   37                           ; TO THE CALLING ROUTINE.
   38                           ;
   39                           ;                                           — EXAMPLE —
   40                           ;
   41                           ; SUBROUTINE CALL                                   JSR   LOG2
   42                           ; RETURN HERE IF X = 0 ---->                        JP    ZERO
   43                           ; RETURN HERE IF X>0 ---->                           CONTINUE
   44                           ;
   45                           ;
   46                           ;
   47                           ;
   48                           ;
   49                           ;
   50      000     00           LOG2:      CLRA
   51      001     57                      AISC        07          ; SET CHARACTERISTIC.
   52      002     06                      X                       ; TO REG LENGTH -1.
                                                                   ; STORE IN MEMORY.

COP CROSS ASSEMBLER     PAGE:  2
LOGS

   53      003     A4           $LP1:      JSRP        SDB2        ; SET ADDRESS POINTER
   54                                                              ; BACK 2 DIGITS.
   55      004     A9                      JSRP        SHLR        ; RESET CARRY AND SHIFT
   56                                                              ; REG LEFT ONE BIT.
   57      005     20           $TS1:      SKC                     ; IS CARRY = 1 YET?
   58      006     C8                      JP          $NO         ; NO — KEEP GOING.
   59      007     49           $LST:      RETSK                   ; YES — FINISHED!!
   60      008     05           $NO:       LD                      ; NO — LOAD COUNT IN ACC.
   61      009     5F                      AISC        -1          ; SUBTRACT ONE.
   62      00A     48           $TS2:      RET                     ; MANTISSA IS A 0! RETURN
   63      00B     06                      X                       ; STORE CHARACTERISTIC.
   64      00C     C3                      JP          $LP1        ; DO IT AGAIN!
   65
   66
   67
   68
   69
   70                           ; 2 ROUTINES ARE CALLED FROM THE SUBROUTINE PAGE BY THIS
   71                           ; PROGRAM: SDB2, SHLR.
   72
```

**Figure 6.**

The program shown develops the $\log_2$ of any even power of 2 by shifting and testing as previously described. Examine what happens to a value of X that is not an even power of 2. In Figure 7, the number 25 is converted to a base 2 log.

$25_{10} = 00011001_2$

Shift left until carry = 1

| Characteristic | Carry | Mantissa | $\text{Log}_2$ |
|---|---|---|---|
| 0100 | 1 | 10010000 | 0100.10010000 |

**Figure 7. Straight Line Approximation of a $\text{Base}_2$ Log**

The resulting number when viewed as an integer characteristic and fractional mantissa is $4.5625_{10}$. The fraction 0.5625 is a straight line approximation of the logarithmic curve between the correct values for the $\text{base}_2$ logs of $2^4$ and $2^5$. The accuracy of this approximation is sufficient for many applications. The error can be corrected, as will be seen later in this discussion, but for now let's look at the problem of exponents or the conversion to an antilog.

To reconstruct the original value of X, find the antilog, requires only restoration of the most significant bit and then its alignment with the power of 2 position indicated by the characteristic. In the example, approximation ($\log_2 25 = 0100.1001$) restoration of MSB can be accomplished by shifting the mantissa (only) one position to the right. In the process a one is shifted into the MSB position.

| Approximation of $\text{Log}_2$ X | Restoration of MSB |
|---|---|
| Char. Mantissa | Char. Mantissa |
| 0100.10010000 | 0100.11001000 |

The value of the characteristic is 4 so the mantissa must be shifted to the right until MSB is aligned with the $2^4$ position.

| $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |

The completion of this operation restores the value of X (X = 25) and is the procedure used to find an antilog. Figure 8 is a flow chart for finding an antilog using this procedure. The implementation in source code is shown in Figure 9.

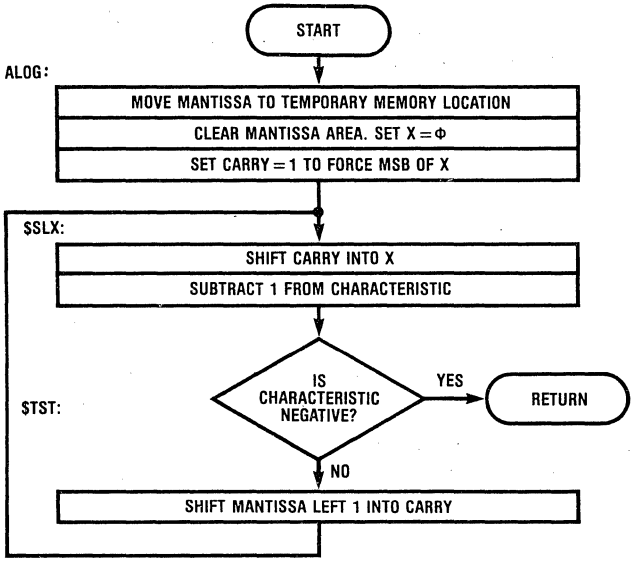



**Figure 8. Flow Chart for Conversion to Antilog**

```
COP CROSS ASSEMBLER PAGE 3
LOGS

                        .FORM           ; ------> CONVERT TO ANTILOG <------ ;


                        ; THE FOLLOWING SUBROUTINE CONVERTS THE STRAIGHT LINE
                        ; THE APPROXIMATION OF A BASE 2 LOGARITHM TO ITS CORRESPONDING
                        ; ANTILOG. UPON EXIT FROM THE ROUTINE THE CONTENTS OF CH
                        ; WILL BE EQUAL TO THE HEXADECIMAL VALUE OF 'ΦF'.

                                        LOCAL


00D  A4         ALOG:   JSRP    SDB2            ; SET ACC TO 0.
00E  00         CLRA                            ; CLEAR MANTISSA AREA.
00F  36                 X       03              ; AND MOVE MANTISSA TO
010  34                 XIS     03              ; TEMPORARY STORAGE.
011  00                 CLRA                    ; LEAVE POINTER AT LO
012  36                 X       03              ; ORDER OF MANTISSA.
013  37                 XDS     03
014  22                 SC                      ; RESTORE MSB OF X.
015  D8                 JP      $SLX
01   A9         $SLM:   JSRP    SHLR            ; SHIFT REMAINDER
                                                ; LEFT INTO CARRY.
017  A3                 JSRP    SDR2            ; MOVE BACK 2 DIGITS.
018  AA         $SLX:   JSRP    SHLC            ; SHIFT X LEFT 1.
019  05                 LD                      ; LOAD CHARACTERISTIC.
01A  5F         $TST:   AISC    -1              ; CHARACTERISTIC -1.
01B  48         $LST:   RET                     ; IF NO CARRY — FINIS.
01C  36                 X       03              ; STORE REMAINDER AND MOVE
                                                ; DOWN ONE REGISTER.
01D  A4                 JSRP    SDB2            ; MOVE BACK 2 DIGITS.
01E  D6                 JP      $SLM            ; DO IT AGAIN.


                        ; 4 ROUTINES ARE CALLED FROM THE SUBROUTINE PAGE BY THIS
                        ; PROGRAM: SDB2, SDR2, SHLR, SHLC.
```

**Figure 9.**

Using the linear approximation technique just described, some error will result when converting any value of X that is not an even power of 2.

Figure 10 contains a table of correct base 2 logarithms for values of X from 1 through 32 along with the error incurred for each when using linear approximation. Notice that no error results for values of X that are even powers of 2. Also notice that the error incurred for multiples of even powers of 2 of any given value of X is always the same:

| Value of X | Error |
|---|---|
| 5 | 0.12 |
| $2 \times 5 = 10$ | 0.12 |
| $4 \times 5 = 20$ | 0.12 |
| 3 | 0.15 |
| $2 \times 3 = 6$ | 0.15 |
| $4 \times 3 = 12$ | 0.15 |
| $8 \times 3 = 24$ | 0.15 |

| X | Hexadecimal Log Base | Linear Approximation of Log Base 2 | Error in Hexadecimal | $E_{M-1} + \frac{EM - EM-1}{2}$ |
|---|---|---|---|---|
| 1 | 0.00 | 0.00 | 0.00 | |
| 2 | 1.00 | 1.00 | 0.00 | |
| 3 | 1.95 | 1.80 | 0.15 | |
| 4 | 2.00 | 2.00 | 0.00 | |
| 5 | 2.52 | 2.40 | 0.12 | |
| 6 | 2.95 | 2.80 | 0.15 | |
| 7 | 2.CE | 2.C0 | 0.0E | |
| 8 | 3.00 | 3.00 | 0.00 | |
| 9 | 3.2B | 3.20 | 0.0B | |
| 10 | 3.52 | 3.40 | 0.12 | |
| 11 | 3.75 | 3.60 | 0.15 | |
| 12 | 3.95 | 3.80 | 0.15 | |
| 13 | 3.B3 | 3.A0 | 0.13 | |
| 14 | 3.CE | 3.C0 | 0.0E | |
| 15 | 3.E8 | 3.E0 | 0.08 | |
| 16 | 4.00 | 4.00 | 0.00 | |
| 17 | 4.16 | 4.10 | 0.06 | 0.03 |
| 18 | 4.2B | 4.20 | 0.0B | 0.09 |
| 19 | 4.3F | 4.30 | 0.0F | 0.0D |
| 20 | 4.52 | 4.40 | 0.12 | 0.11 |
| 21 | 4.67 | 4.50 | 0.17 | 0.15 |
| 22 | 4.75 | 4.60 | 0.15 | 0.16 |
| 23 | 4.87 | 4.70 | 0.17 | 0.16 |
| 24 | 4.95 | 4.80 | 0.15 | 0.16 |
| 25 | 4.A4 | 4.90 | 0.14 | 0.15 |
| 26 | 4.B3 | 4.IA0 | 0.13 | 0.14 |
| 27 | 4.C1 | 4.B0 | 0.11 | 0.12 |
| 28 | 4.CE; | 4.C0 | 0.0E | 0.10 |
| 29 | 4.DB | 4.D0 | 0.0B | 0.0D |
| 30 | 4.E8 | 4.E0 | 0.08 | 0.0A |
| 31 | 4.F4 | 4.F0 | 0.04 | 0.06 |
| 32 | 5.00 | 5.00 | 0.00 | 0.02 |
| 33 | | 5.1- | | |

**Figure 10. Error Incurred by Linear Approximation of Base 2 Logs**

An error that repeats in this way is easily corrected using a look-up table. The greatest absolute error will occur for the least value of X not an even power of 2, x = 3, is about 8%. A 4 point correction table will eliminate this error but will move the greatest uncompensated error to X = 9 where it will be about 4%. This process continues until at 16 correction points the maximum error for the absolute value of the logarithm is less than 1 percent. This can be reduced to 0.3 percent by distributing the error. Interpolated error values are listed in Figure 10 and are repeated in Figure 11 as a binary table.

| High Order 4 Mantissa Bits | Binary Correction Value | Hexadecimal Correction Value |
|---|---|---|
| 0 0 0 0 | 0000 0000 | 0 0 |
| 0 0 0 1 | 0000 1001 | 0 9 |
| 0 0 1 0 | 0000 1101 | 0 3 |
| 0 0 1 1 | 0001 0001 | 1 1 |
| 0 1 0 0 | 0001 0101 | 1 5 |
| 0 1 0 1 | 0001 0110 | 1 6 |
| 0 1 1 0 | 0001 0110 | 1 6 |
| 0 1 1 1 | 0001 0110 | 1 6 |
| 1 0 0 0 | 0001 0101 | 1 5 |
| 1 0 0 1 | 0001 0100 | 1 4 |
| 1 0 1 0 | 0001 0010 | 1 2 |
| 1 0 1 1 | 0001 0000 | 1 0 |
| 1 1 0 0 | 0000 1101 | 0 D |
| 1 1 0 1 | 0000 1010 | 0 A |
| 1 1 1 0 | 0000 0110 | 0 6 |
| 1 1 1 1 | 0000 0010 | 0 2 |

**Figure 11. Correction Table for $L_2$ X Linear Approximations**

Notice in Figure 10 that left justification of the mantissa causes its high order four bits to form a binary sequence that always corresponds to the proper correction value. This works to advantage when combined with the COP400 LQID instruction. LQID implements a table look-up function using the contents of a memory location as the address pointer. Thus we can perform the required table look-up without disturbing the mantissa.

Figure 12 is the flow chart for correction of a logarithm found by linear approximation. Figure 13 is its implementation in COP400 assembly language. Notice that there are two entry points into the program. One is for correction of logs (LADJ:), the other is for correction of a value prior to its conversion to an antilog (AADJ:).

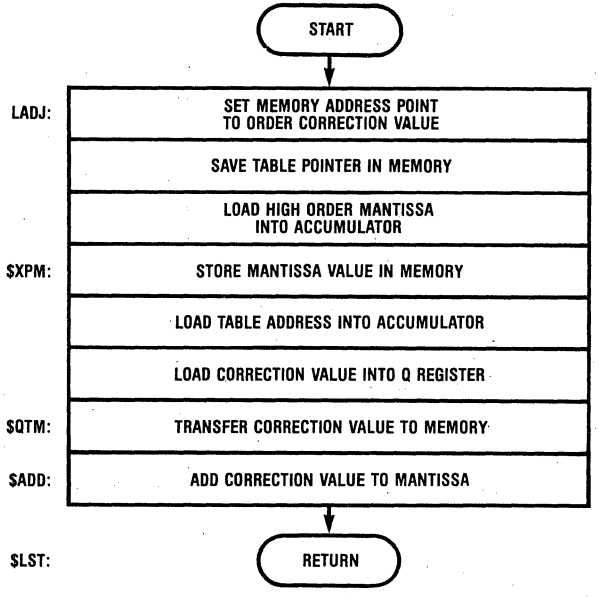



**Figure 12. Flow Chart for Correction of a Value Found by Straight Line Approximation**

```
COP CROSS ASSEMBLER     PAGE:  4
LOGS

  110                           . FORM      ; ·····→ ADJUST VALUE OF LOGARITHM ←····· ;
  111
  112                                       . LOCAL
  113
  114
  115                           ; THE FOLLOWING TABLE IS USED DURING THE CORRECTION OF VALUES
  116                           ; FOUND BY STRAIGHT LINE APPROXIMATION. IT IS PLACED HERE IN
  117                           ; ORDER TO ALIGN ITS BEGINNING ELEMENT WITH A ZERO ADDRESS AS
  118                           REQUIRED BY THE LQID INSTRUCTION.
  119
  120   01F   44                            NOP                         ; REGISTER WITH ZERO ADDRESS.
  121   020   03                TPLS:       . WORD      03,09,0D,011
        021   09
        022   0D
        023   11
  122   024   15                . WORD      015,016,016,016
        025   16
        026   16
        027   16
  123   028   15                            . WORD      015,014,012,010
        029   14
        02A   12
        02B   10
  124   02C   0D                            . WORD      0D,0A,06,02
        02D   0A
        02E   06
        02F   02
  125
  126                           ; THE FOLLOWING SUBROUTINE ADJUSTS THE VALUE OF A BASE 2
  127                           ; LOGARITHM FOUND BY STRAIGHT LINE APPROXIMATION. THE
  128                           ; CORRECTION TERMS ARE TAKEN FROM THE TABLE ABOVE. THE
  129                           ; SUBROUTINE HAS 2 ENTRY POINTS:
  130                           ;
  131                           ;           LADJ: — ADJUSTS A VALUE DURING CONVERSION TO A LOG
  132                           ;
  133                           ;           AADJ: — ADJUSTS A VALUE DURING CONVERSION TO ANTILOG
  134                           ;
  135                           ; THE CARRY FLAG IS SET UPON ENTRY TO DISTINGUISH BETWEEN LOG
  136                           ; (C = 1) AND ANTILOG (C = 0) CONVERSIONS. DURING A LOGARITHM
  137                           ; CONVERSION THE VALUE FOUND IN THE ABOVE TABLE IS ADDED TO
  138                           ; THE MANTISSA. DURING AN ANTILOG CONVERSION THE VALUE FOUND
  139                           ; IN THE ABOVE TABLE IS SUBTRACTED FROM THE MANTISSA.
  140
  141
  142   030   32                AADJ:       RC                          ; C = 0 FOR ANTILOG
  143   031   F3                            JP          $LD             ; CONVERSION.
  144   032   22                LADJ:       SC                          ; C =  FOR LOG2 ADJ.
  145   033   05                $LD         LD                          ; MOVE ADDRESS POINTER BACK
  146   034   07                            XDS                         ; ONE LOCATION.
  147   035   05                            LD                          ; LOAD CONTENTS OF HI MANTISSA
  148   036   37                            XDS         03              ; AND STORE IT IN THE LO ORDER
  149   037   06                            X                           ; OF THE TEMP MEMORY LOCATION.
  150   038   00                            CLRA                        ; SET TABLE POINTER
  151   039   52                            AISC        TBL             ; (ACC) TO TABLE ADDRESS.

COP CROSS ASSEMBLER     PAGE:  5
LOGS

  152   03A   BF                $GTM:       LQID                        ; LOAD CORRECTION VALUE TO Q.
  153   03B   332C                           CQMA                        ; TRANSFER Q REGISTER
  154   03D   04                            XIS                         ; CONTENTS TO MEMORY.
  155   03F   07                            XDS
  156   03F   20                            SKC                         ; ANTILOG?

  157   040   80                            JSRP        COMP            ; YES — COMPLIMENT.
  158   041   98                $ADD:       JSRP        ADRO            ; ADD CORRECTION VALUE
  159                                                                   ; TO MANTISSA.
  160   042   35                            LD          03              ; SET POINTER TO
  161   043   48                $LST:       RET         ; CHARACTERISTIC AND
  162                                                                   ; RETURN.
  163
  164                           ; 2 ROUTINES ARE CALLED FROM THE SUBROUTINE PAGE BY THIS
  165                           ; PROGRAM: COMP, ADRO
  166
  167         0020                          V1 = TPLS&OFF
  168         0002                          TBL = V1/16
  169
  170
  171
```

**Figure 13.**

## Subroutines Used by the Log and Antilog Programs

```
COP CROSS ASSEMBLER     PAGE:  6
LOGS

  172                                       FORM

  173          0080       PAGE 02                    ; ------> SUBROUTINES <------ ;
  174
  175                     ; THE FOLLOWING ROUTINES RESIDE ON THE SUBROUTINE PAGE. THEY
  176                     ; ARE CALLED BY THE LOGS PROGRAM BUT ARE GENERAL PURPOSE IN
  177                     ; NATURE AND FUNCTION AS UTILITY ROUTINES.
  178
  179
  180
  181                               ; ------> COMPLEMENT 8 BITS <------ ;
  182
  183                               . LOCAL
  184
  185                     ; THIS ROUTINE FORMS IN MEMORY THE 2'S COMPLEMENT OF THE TWO
  186                     ; ADJACENT DIGITS IDENTIFIED BY THE ADDRESS POINTER. THE
  187                     ; CONTENTS OF THE ADDRESS POINTER ARE NOT ALTERED.
  188
  189                     ; THERE ARE TWO ENTRY POINTS:
  190                     ;
  191                     ; COP: COMPLEMENT 8 BITS.
  192                     ;
  193                     ; CMPE: EXTEND THE COMPLEMENT TO AN ADDITIONAL 8 BITS
  194                     ;
  195
  196  080     22         COMP:     SC
  197  081     00         CMPE:     CLRA                          ; SET MINUEND = 0
  198  082     06                   X                             ; AND STORE IN MEMORY.
  199  083     10                   CASC
  200  084     44                   NOP                           ;
  201  085     04                   XIS                           ;
  202  086     00                   CLRA                          ; SET MINUEND = 0
  203  087     06                   X                             ; AND STORE IN MEMORY.
  204  083     10                   CASC                          ;
  205  089     44                   NOP
  206  08A     04                   XIS                           ;
  207  08B     44                   NOP                           ; AVOID SKIP IF DIGIT 15.
  208  08C     A4                   JP        SDB2                ; RETURN THRU SDB2
  209                                                             ; TO RESTORE POINTER.
  210
  211
  212
  213                     ; ------> ADD 8 BITS IN ADJACENT REGISTERS <------ ;
  214
  215                               . LOCAL
  216
  217
  218
  219                     ; THIS ROUTINE ADDS TWO BINARY DIGITS (8 BITS) FROM ANY REGISTER
  220                     ; TO THE CORRESPONDING TWO BINARY DIGITS IN EITHER REGISTER
  221                     ; IMMEDIATELY ADJACENT. THERE ARE THREE ENTRY POINTS:
  222                     ;
  223                     ;         LADR: — RESET CARRY AND ADD 2 DIGIT PAIRS
```

```
COP CROSS ASSEMBLER     PAGE:  7
LOGS

                                ;               LADD: — ADD 2 DIGIT PAIRS WITH UNMODIFIED CARRY
                                ;               ADD1: — ADD 2 SINGLE DIGITS WITH UNMODIFIED CARRY
                                ;
                                ;
                                ;
                                ;
0230  08D   32          LADR:   RC                              ; RESET CARRY PRIOR TO ADD.
0231  08E   15          LADD:   :D              01              ; LD ADDEND AND MOVE TO ADJ REG
0232  08F   30                  ASC                             ; ADD AUGEND.
0233  090   44                  NOP                             ; AVOID CARRY!
0234  091   14                  XIS             01              ; STORE SUM AND MOVE TO ADDEND
0235  092   15          ADD1:   LD              01              ; REPEAT PROCESS
0236  093   30                  ASC                             ; FOR
0237  094   44                  NOP                             ; HIGH ORDER
0238  095   14                  XIS             01              ; DIGIT.
0239  096   44                  NOP                             ; AVOID SKIP IF DIGIT 15.
0240  097   48          $LST:   RET                             ; FINISHED — RETURN!!!!
0241
0242
0243
0244
0245                            ; ------> ADD 8 BITS IN OPPOSITE REGISTERS <------ ;
0246
0247                            . LOCAL
0248
0249
0250
0251                    ; THIS ROUTINE ADDS TWO BINARY DIGITS (8BITS) FROM ANY REGISTER
0252                    ; TO THE CORRESPONDING TWO BINARY DIGITS IN EITHER REGISTER
0253                    ; DIRECTLY OPPOSITE. THERE ARE THREE ENTRY POINTS:
0254                    ;
0255                    ;               ADR0: — RESET CARRY AND ADD 2 DIGIT PAIRS
0256                    ;               ADD0: — ADD 2 DIGIT PAIRS WITH UNMODIFIED CARRY
0257                    ;               AD01: — ADD 2 SINGLE DIGITS WITH UNMODIFIED CARRY
0258
0259
0260
0261
0262  098   32          ADR0:   RC                              ; RESET CARRY PRIOR TO ADD.
0263  099   35          ADD0:   LD              03              ; LD ADDEND AND MOVE TO OPP REG
0264  09A   30                  ASC                             ; ADD AUGEND.
0265  09B   44                  NOP                             ; AVOID CARRY!
0266  09C   34                  XIS             03              ; STORE SUM AND MOVE TO ADDEND.
0267  09D   15          AD01:   LD              01              ; REPEAT PROCESS
0268  09E   30                  ASC                             ; FOR
0269  09F   44                  NOP                             ; HIGH ORDER
0270  0A0   34                  XIS             03              ; DIGIT.
0271  0A1   44                  NOP                             ; AVOID SKIP IF DIGIT 15.
0272  0A2   48          $LST:   RET                             ; FINISHED — RETURN!!!!
0273
0274
0275
0276                            ; ------> SET DIGIT ADDRESS BACK TWO <------ ;
0277
```

```
COP CROSS ASSEMBLER     PAGE:  8
LOGS

                                        .LOCAL
                        ; THIS ROUTINE SUBTRACTS 2 FROM THE CONTENTS OF THE
                        ; DIGIT POINTER (B REGISTER). THE CONTENTS OF THE
                        ; ACCUMULATOR ARE LOST IN THE PROCESS. THE USE OF
                        ; SDB2 ALLOWS ADDRESSING WITHIN THE LOGS SUB
                        ; ROUTINE TO BE RELATIVE TO THE CONTENTS OF THE
                        ; ADDRESS POINTER (B REGISTER) UPON ENTRY.
                        ; SDB2 IS COMMONLY USED IN BYTE OPERATIONS TO RESTORE THE
                        ; DIGIT POINTER TO THE LOW ORDER POSITION.
                        ; THERE ARE TWO ENTRY POINTS:
                        ;
                        ; SDR2:      SET DIGIT ADDRESS BACK 2 AND MOVE TO OPPOSITE REGISTER.
                        ;
                        ; SDB2: SET DIGIT ADDRESS BACK 2 RETAINING PRESENT REGISTER.

0A3  35     SDR2:       LD      03          ; MOVE TO OPPOSITE REGISTER.
0A4  4E     SDB2:       CBA                 ; PLACE DIGIT COUNT IN ACC.
0A5  5E                 AISC    -2          ; SUBTRACT 2.
0A6  44                 NOP                 ; SHOULD ALWAYS SKIP.
0A7  50                 CAB                 ; PUT DIGIT COUNT BACK.
0A8  48                 RET                 ; FINISHED — RETURN!!


                        ; ------> SHIFT LEFT <------ ;

                                        .LOCAL
                        ; THIS ROUTINE SHIFTS LEFT THE CONTENTS OF TWO MEMORY
                        ; LOCATIONS ONE BIT. THERE ARE THREE ENTRY POINTS:

                        ;                   SHLR: RESETS THE CARRY BEFORE SHIFTING
                        ;                         IN ORDER TO FILL THE LOW ORDER
                        ;                         BIT POSITION WITH A 0.
                        ;
                        ;                   SHLC: SHIFTS THE STATE OF THE CARRY INTO
                        ;                         THE LOW ORDER BIT POSITION.
                        ;
                        ;                   SHL1: SHIFTS LEFT THE CONTENTS OF ONLY
                        ;                         ONE MEMORY LOCATION. THE STATE
                        ;                         OF THE CARRY IS SHIFTED INTO THE
                        ;                         LOW ORDER POSITION OF MEMORY.


0A9  32     SHLR:       RC                  ; CLEAR CARRY PRIOR TO SHIFT.
0AA  05     SHLC:       LD                  ; LOAD FIRST MEM DIGIT.
0AB  30                 ASC                 ; DOUBLE IT.
0AC  44                 NOP                 ; AVOID SKIP.
0AD  04                 XIS                 ; STORE SHIFTED DIGIT.
0AE  05     SHL1:       LD                  ; LOAD NEXT MEM DIGIT.
0AF  30                 ASC                 ; DOUBLE IT TOO.

COP CROSS ASSEMBLER     PAGE:  9
LOGS

0B0  44                 NOP                 ; AVOID SKIP, IF ANY
0B1  04                 XIS                 ; STORE SHIFTED DIGIT.
0B2  48     $LST:       RET                 ; FINISHED — RETURN!


                                        .END
```

# Use of Macro-Assembled Code

National Semiconductor
COP Brief 3
May 1980

## Introduction

The use of macro assembled code in a COP400 series program can be beneficial to the user if implemented correctly. Care must be taken to insure that ROM space is not being utilized in a wasteful manner. In many cases a block of commonly used code would lend itself to a subroutine rather than repeating a macro. The purpose of this brief is to illustrate the advantages of the macro capability of the COP400 Product Development System (PDS). Due to modifications in the assembler program there is erroneous information concerning macro calls in the *COP400 PDS Manual*. These modifications are discussed in the section labeled GENERAL.

By using macros the programming process becomes much more general in nature. In some circumstances, with a good macro library, a pseudo higher level language can be created. This higher level of instructions inefficiently utilizes ROM space. However, if the ROM space is available, macros can ease the task of programming. A feasable approach to organized programming might be to work from a macro library and in the event of limited ROM space, optimize code by replacing the macros which are repeatedly used, by a single subroutine and calling statements.

Macros also may be used as programming aids which ease the understanding of the instruction set. When utilizing macros to rename single instructions no ROM space is wasted. Macro statements must be declared at the beginning of a source file. However, this does not utilize ROM space unless the macro is called within the source. Various methods of creating multiple and single instructions macros are discussed below.

## Creating Instruction Macros

One very basic use of macros is to rename instructions or groups of instructions to suit individual preferences. In the example shown the user must add the macro to the source file and each time the new mnemonic is encountered the assembler will create the correct code.

```
B1 = 0                          ; EQUATE STATEMENTS
B2 = 0                          ; USED FOR PROGRAMMING
B4 = 2                          ; CLARITY
B8 = 3                          ;

. MACRO     SZ, BIT
SKMBZ       BIT
. ENDM
```

The renamed instruction may now be utilized in the following way:

```
SZ                  B8

        OR

SZ                  3
```

In both cases 'SKMBZ 3' will be assembled.

By utilizing the equate capabilities the user can even further personalize the instruction set. In the above example 'B1' is equated to '0', 'B2' to '1', etc. This translates a bit position '0,1,2,3' to a bit weight of '1,2,4,8' which may be of preference to the programmer. In any case, the ability to manipulate the instruction set is available to the user without direct modification to the assembler program.

Conditional assembly in conjunction with macro capabilities may be utilized to further ease programming. In the following example the 'JSR' and 'JSRP' instructions are replaced with a simple 'CALL' statement. It is important to allocate the proper number of ROM spaces during pass 1 of the assembler so as to assign a ROM location to correspond to each label. It is not until pass 2 of the assembler that information of label addresses is known. Because of this the macro must be able to determine whether the 'CALL' is a one or two byte instruction. This can be accomplished by use of conditional assembly statements. In the example shown, all subroutines located in page 2 must be labeled by an 'A' followed by the subroutine name. Conversely, subroutines not located in page 2 must not begin with the letter 'A'. Note that the character 'A' was chosen arbitrarily and may be modified to any legal character or characters.

```
. MACRO CALL,X,Y                ; MACRO TO RENAME JSR, JSRP
. IFC #1 EQ A                   ; TEST IF LABEL IS PREFACED
                                ; BY AN 'A'
JSRP X^Y                        ; YES, ASSEMBLE SINGLE BYTE
. ELSE
JSR X*Y                         ; NO, ASSEMBLE DOUBLE BYTE
. ENDIF                         ; MUST TERMINATE . IF
. endm                          ; TERMINATE MACRO

CALL        AINC                ; CALL SUB IN PAGE 2
```

This statement will generate:

```
JSRP        AINC
```

AINC must be located in page 2 or an assembler error will occur.

```
CALL SUB                        ; CALL SUB NOT IN PAGE 2
```

This statement will generate:

```
JSR         SUB
```

## Macros of Interest

### Table Look-Up Macro

This macro will place the look-up table in the ROM space designated by the LOC parameter or if the parameter is not specified the table will follow in successive locations after being called.

```
MACRO    TABLE,LOC          ; SEG TABLE LOOKUP
. IFC  #>0                  ; TEST IF PARAMETER IS THERE
. X' LOC                    ; YES, USE IT
. ELSE                      ; NO, ELIMINATE ROM POINTER
. ENDIF                     ; TEMINATE . IF
. WORD   0FD                ; 0
. WORD   061                ; 1
. WORD   0DB                ; 2
. WORD   0F3                ; 3
. WORD   067                ; 4
. WORD   0B7                ; 5
. WORD   03F                ; 6
. WORD   0E1                ; 7
. WORD   0FF                ; 8
. WORD   0E7                ; 9
. WORD   0CF                ; P
. WORD   0EF                ; A
. WORD   07D                ; U
. WORD   09D                ; C
. WORD   08F                ; F
. WORD   000                ; BLANK
. ENDM

TABLE    024                ; SET ROM POINTER AT ROM
                            ; LOCATION 024<hex>
  OR

TABLE                       ; START SEVEN SEG AT PRESENT
                            ; ROM LOCATION
```

The code generated will correspond to the look-up table given in the macro. This table may be modified to suit any particular symbol. Sixteen segment arrays are listed only to take advantage of the LQID instruction. These may be modified to the user's preference.

Additional Macro information is available in the *COP400 Product Development System Manual.*

## General

The *COP PDS Manual* defines parameter delimiters when using macros as commas or blanks. When creating the macro, parameters must be separated by commas whereas blanks are not acceptable. When calling the macro it is acceptable to delimit the parameters by either blanks or commas.

In order to assure correct assembly when using the . IF or . IFC directives it is essential to terminate these directives by a .ENDIF. This point is not emphasized in the manual. However it is important in the assembly process.

The . LIST directive may be used to suppress the macro listing in the source or to expand it. The *COP PDS Manual* covers LIST options in detail.

# L-Bus Considerations

National Semiconductor
COP Brief 4
May 1980

## L-Bus Considerations

Users of the COP400 family of microcontrollers should be aware that certain outputs exhibit peculiarities that preclude their use as clocks for edge sensitive devices such as flip-flops, counters, shift registers, etc. All family members excluding the COP410L and COP411L may generate false states on $L_0$-$L_7$ during the execution of the CAMQ instruction. Figure 1 contains a short program to illustrate this.

```
START:
        CLRA                    ; ENABLE THE Q
        LEI     4               ; REGISTER TO L LINES
        LBI     TEST
        STII    3
        AISC    12
LOOP:
        LBI     TEST            ; LOAD Q WITH X'C3
        CAMQ
        JP      LOOP
```

**Figure 1. Glitch Test Program**

In this program the internal Q register is enabled onto the L lines and a steady bit pattern of logic highs is output on $L_0$, $L_1$, $L_6$, $L_7$, and logic lows on $L_2$-$L_5$ via the two-byte CAMQ instruction. Timing constraints on the device are such that the Q register may be temporarily loaded with the second byte of the CAMQ opcode (X'3C) prior to receiving the valid data pattern. If this occurs, the opcode will ripple onto the L lines and cause negative-going glitches on $L_0$, $L_1$, $L_6$, $L_7$, and positive glitches on $L_2$-$L_5$. Glitch durations are under 2 microseconds, although the exact value may vary due to data patterns, processing parameters, and L line loading. These false states are peculiar only to the CAMQ instruction and the L lines. The user should experience no difficulty interfacing with other COP420 outputs such as $G_0$-$G_3$ and $D_0$-$D_3$ to edge sensitive components.

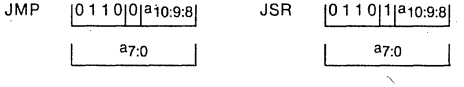
# Software and Opcode Differences in the COP444L Instruction Set

National Semiconductor
COP Brief 5
May 1980

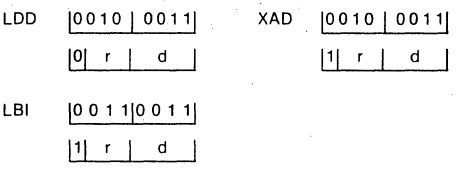
The COP444L is essentially a COP420L with double RAM and ROM. Because of this increased memory space certain instructions have expanded capability in the COP444L. Note that there are no new instructions in the COP444L and that all instructions perform the same operations in the COP444L as they did in the COP420L. The expanded capability is merely to allow appropriate handling of the increased memory space. The affected instructions are:

JMP a (a = address)
JSR a (a = address)
LDD r,d (r,d = RAM address Br,Bd)
XAD r,d (r,d = RAM address Br,Bd)
LBI r,d (r,d = RAM address Br,Bd; only two byte form of the instruction affected)
XABR

The JMP and JSR instructions are modified in that the address a may be anywhere within the 2048 words of ROM space. The opcodes are as follows:




The LDD, XAD, and two byte LBI are modified so that they may address the entire RAM space. The opcodes are as follows:




The XABR instruction change is transparent to the user. The opcode is not changed nor is the function of the instruction. The change is that values of 0 through 7 in A will address registers in the COP444L — i.e. the lower three bits of A become the Br value following the instruction. In the COP420L, the lower two bits of A became the Br value following an XABR instruction.

Note that those instructions which have an exclusive-or argument (LD, X, XIS, XDS) are not affected. The argument is still two bits of the opcode. This means that the exclusive-or aspect of these instructions works within blocks of four registers. It is not possible to toggle Br from a value between 0 and 3 to a value between 4 and 7 by means of these instructions.

There are no other software or opcode differences between the COP444L and the COP420L. Examination of the above changes indicates that the existing opcodes for those instructions have merely been extended. There is no fundamental change.

# RAM Keep-Alive

National Semiconductor
COP Brief 6
May 1980



A COPS™ application is a small scale computer system and the design of a power shut-down is not trivial. During the time that power is available, but out of the designed operating range, the system must be prevented from doing anything to harm protected data. This will typically involve some type of external protection or timing circuit.

There is an option on the COP420, 420L, and 410L parts called "RAM Keep-Alive" that provides a separate power supply to the RAM area of the chip via the CKO pin. The application of power to the RAM while the remainder of the chip has been powered down via $V_{CC}$ will keep the RAM "alive".

However, the integrity of data in the RAM is not only a function of power but is also influenced by transient conditions as power is removed and reapplied. During power-on, the **P**ower **O**n **R**eset (POR) circuit will keep transients from causing changes in the RAM states. The condition of power loss will have some probability of data change if external control is not used.

At some point below the minimum operating voltage certain gates will no longer respond properly while others may still be functional until a much lower voltage. During this transition time any false signal could cause a false write to one or more cells. Another effect could be to turn on multiple address select lines causing data destruction.

Testing the rate of data change is very difficult because it must be done on a statistical basis with many turn/on-turn/off cycles. Two factors have a major bearing on the numbers derived by testing. One is to call any change in a related data block a failure, even though more than one bit in that block may have changed (this latter case may well be due to the "address select mode"). The second factor is that without massive instrumentation it is impossible to examine the data after each power cycle. Indeed, to do so might have caused errors!

By running the power cycle for a period of time and then looking for changes, one could overlook multiple changes thus reducing the error rate. This has been minimized by more frequent checking which indicates that the errors are spread out randomly over time.

With a power supply that drops from 4.5 to 2V in approximately 100 ms, the drop-out rate is 1 in 5k to 6k power cycles. Reducing the voltage fall time will cause an improvement in the number of cycles per drop-out. This will reach a limit condition of a very high number (1 per 1 million?) when the power falls within one instruction cycle (4-10 $\mu$s for the 420, 15-40 $\mu$s for the "L" parts). Attaining very rapid fall time may cause problems due to the lack of decoupling/bypass capacitance. By inserting an electronic switch between the regulator and $V_{CC}$ of the COP chip one might be able to meet this type of fall time. By implication some type of sensing is required to cause the switching.

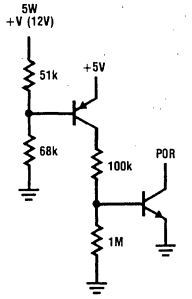
The desirable approach is to force the COP reset input to zero before the voltage falls below 4.5V. This provides a drop out rate of approximately 1 in 50k for the "L" parts and 1 in 100k for the 420. By also stopping the clock of the "L" parts they can achieve a drop-out rate similar to the 420. While not perfect, the number of cycles between data error should be considered with respect to the needs of the application.

The external circuitry to control the chip during the power transition has several implementations each one being a function of the application. The simplest hardware is found in a battery powered (automotive) application. The circuit must sense that the switched 12V is falling (e.g., at some value much below 12V and still greater than 5V). This can be done by using the unswitched 12V as a reference for a divider to a nominal voltage of 8V. As the switched 12V drops below the reference a detector will turn on a clamp transistor to a series switch, the POR, and/or the clock circuit (Figure 1). It should be noted that this draws current during the absence of the switched 12V circuit.

In non-automotive usage a similar circuit can be used where there is a stable reference voltage available to use with the comparator/clamp. Thus a 3.6V rechargable Ni-Cad battery could be used as the reference voltage and $V_{RAM}$ if the appropriate divider is used to level shift to this operating range.

In AC line-powered applications, a similar method could be used with the raw DC being sensed for drop. Another method would be to sense that the line had missed 2-3 cycles either by means of a charge pump or peak detection technique. This will provide the signal to turn on the clamp. One must make this faster than the time to discharge the output capacitance of the power supply, thus assuring that the clamp has performed its function before the supply falls below spec value.

In conclusion, to protect the data stored in RAM during a power-off cycle, the POR should go low before the $V_{CC}$ power drops below spec and come up after $V_{CC}$ is within spec. The first item must be handled with an external circuit like Figure 1 and the latter by an RC per the data sheet.



# MICROBUS™ Programming Considerations

National Semiconductor
COP Brief 7
May 1980

## Introduction

The COP402 MICROBUS™ is a peripheral microprocessor device and its operating characteristics are described in the 402M data sheet and the *Chip User's Manual*. Given in this brief are some clarifications as to the allowable option selection and also as to programming requirements that are not readily obvious.

## COPS IN Input Port Options on the COP402M

Neither the IN Input options that may be selected, nor the input characteristics associated with the INIL and ININ instructions are clearly indicated in the COP402M data sheet (Preliminary, September 1978). In the COP402M configuration $IN_0$ is a general purpose latched input with a load device to $V_{CC}$. All other IN inputs (CS, RD, and WR), are selected as high impedance inputs without pull-up devices.

The COP402M and the COP420M will execute ININ and INIL instructions, yet the exact operation of these instructions is not detailed in the specification. $IN_0$ information will be latched in accordance with the criteria specified in the data sheet (min. 2 inst. cycle time at logic zero), as will the WR, ($IN_3$) input if these criteria are met. If the WR pulse does not meet the 2 instruction cycle criteria, yet does satisfy MICROBUS timing, the status of the IL latch corresponding to the WR input ($IN_3$) cannot be predicted when the status of the IL latches is read in via an INIL instruction.

When executing the ININ instruction, the status of $IN_0$ and the MICROBUS signals will be read in with the exception of the RD ($IN_1$) signal. This signal will always read in as a logical one.

## COPS IN Input Port Options on the COP420M

When selecting a MICROBUS option it is possible to select either load devices to $V_{CC}$ or high impedance inputs on $IN_0$ and all MICROBUS signals. These options may be chosen individually corresponding to $IN_0$, CS, WR, and RD signals. There is also a choice between standard TTL input levels or a High Trip option for the IN and MICROBUS inputs. The only restriction (for all 400 series devices) is that when either a High Trip or TTL trip levels are chosen, they must be selected in blocks corresponding to that input port. For example, all IN lines must have High Trip, rather than just one IN line.

## MICROBUS™ Programming Considerations

The COP402M data sheet describes the handshaking protocall required when implementing the COP420M as a microprocessor peripheral device. When a WR strobe is detected, an internal reset of the $G_0$ latch occurs. This signal indicates that data is ready to be transferred to the Q latches from the microprocessor bus. Due to the relatively short timing requirements on the WR strobe signal it is necessary to latch the write request such that under program control the COP device can service the write request. Upon completion of the data transfer and any task that may have been performed, the user then signals the microprocessor that it is available once again by setting the $G_0$ latch. This portion of the handshaking (setting $G_0$) is the only time that the G Port should be used as an output port. All G Ports in the MICROBUS configuration should be used only as input in order to guarantee that a WR strobe is not missed. When using the G Port as an output Port it is possible that a WR pulse may be ignored as explained in the example below. The G Port may be utilized as an output port in the following way, however, there is a 3 cycle period that if a WR pulse occurred it would be ignored.

```
GPIN: LBI     RAM      ; POINT TO RAM LOCATION
      ING              ; READ THE G PORT
      X                ; STORE IN RAM
      SMB     X        ; CHANGE G PORT INFO TO BE SENT OUT
      SKGBZ   O        ; SEE IF WR STROBE HAS OCCURED
      JP      OUT      ; HAVE NOT BEEN INTERRUPTED (YET)
      JP      SERVICE  ; GO SERVICE WR REQUEST
OUT:  OMG              ; OUTPUT NEW G PORT INFORMATION
```

If a write pulse occured during the JP to OUT or the OMG instructions it would not be recognized because the OMG will set the $G_0$ latch to a logic one, signalling to the microprocessor that the WR strobe has been serviced.

It is possible to output to the G Port after WR and before $G_0$ is set, and not miss a WR request. This means that the data outputted on the G lines wil be updated only after the microprocessor has initiated an interrupt.

## General

The COP402M data sheet specified all IP address lines as TTL compatible, with a fan out of one. Address lines IP4 and IP5 do not meet this criterion, although all other IP lines do. It is sufficient to say that all IP lines are LSTTL compatible with a fan out of one, the restricting factor being IP4 and IP5, ($I_{OL}$ @ 0.4V, $360\mu A = I_{OH}$ @ $3.0V = 50\mu A$.)

# COPS™ Peripheral Chips

National Semiconductor
COP Brief 8
May 1980



## COPS Peripheral Chips

There are several I/O peripheral chips that are compatible with the COPS microcontrollers by communicating through the serial I/O port. Table 1 shows a listing of those circuits. Two different sets of timing employed by them are shown in Figure 1. A brief description of the electrical characteristics of each chip is given below.

### COP450 RAM Interface Chip

The COP450 RAM interface chip is fabricated by a low voltage CMOS process. The chip operates between 2.5V and 5.5V. the clock (SK), data input (DI) and chip enable (CE) may tolerate a 10V signal. When interfacing to a COPS controller with a higher power supply, data output (DO) should not rise above the COP450 supply.

### COP452 Frequency and Counter Chip

The COP452 frequency and counter chip is fabricated by N-channel silicon gate process. The chip operates between 4.5V and 9.5V. It contains a TRI-STATE™ output to be connected to the SI pin of the COPS controller. This output can drive the SI pin of a standard or a low power COPS controller provided that standard TTL input level option is chosen for the SI pin. If the higher input level option is chosen, or a CMOS COPS controller is used, an external resistor may be used to increase the HIGH output level. The LOW level will also increase.

### COP470 V.F. Display Driver

The COP470 V.F. display driver is fabricated by a PMOS process. It operates between 4.5V and 9.5V with a high voltage supply pin for output drivers to drive fluorescent displays. The input levels on this chip are different from other chips. The LOW level is between 0V and $V_{CC} - 4V$, and the HIGH level is between $V_{CC} - 1.5V$ to $V_{CC}$. The input LOW level will be between 0V and 0.5V when $V_{CC}$ is 4.5V. If $V_{CC}$ is above 5V, the input HIGH level will be above the CMOS input HIGH level, e.g., with $V_{CC}$ being 9.5V, the minimum input HIGH level will be 8V, compared to 6.8V for CMOS minimum input HIGH level. The COPS controller data sheet will not accurately show the propagation delay. To obtain a conservative estimate of the propagation delay, assume that delay comes from R-C charging time, with the capacitance and time necessary to charge to $0.7V_{CC}$ given in the data sheet (COPS to CMOS interface), extrapolate the time to the minimum HIGH level for that power supply voltage. This value should be a good conservative estimate.

### COP472 LCD Driver

The COP472 LCD driver is fabricated by a low voltage CMOS process. The driver operates between 3V and 5.5V. The clock (SK), data input (DI), and chip enable (CE) may tolerate a 10V signal. The actual power supply used will depend on the operating voltage of the LCD.

### COP498 Read/Write Memory and Timer Chip

The COP498 read/write memory and timer chip is fabricated by a low voltage CMOS process. The chip operates between 2.5V and 6V. Some I/O, including clock (SK), data input (DI), and chip enable (CE) may tolerate a 10V signal. When interfacing to a COPS controller with a higher power supply, data output (DO) should not rise above the COP 498 power supply.

### DS8906 PLL Chip

DS8906 PLL chip is fabricated by a $I^2L$ process. The chip operates between 4.75V and 5.25V. The inputs may tolerate a 9V signal. The maximum input source current is $10\mu A$ and the maximum input sink current is $25\mu A$.

### MM5450 LED Display Driver

The MM5450 LED display driver is fabricated by an N-channel metal gate process. The chip operates between 4.75V and 11V.

### TTL SSI/MSI/LSI Interface

The 7400 series logic operates between 4.75 and 5.25V only. The standard and CMOS COPS controller outputs can directly drive one input and maintain the TTL valid input levels. If it is also necessary to drive CMOS or PMOS in a 5V system, buffers or an external 4.7k pull-up resistor may be added. This resistor together with a TTL load may increase the maximum output LOW level to 0.5V. If a TTL output needs to drive a CMOS COPS controller input or a standard COPS controller input with a high input option from a TTL buffer, a TTL to MOS buffer or an external pull-up 4.7k resistor may be added.

### LSTTL SSI/MSI/LSI Interface

The 74LS series logic operates between 4.75V and 5.25V only. The standard and CMOS COPS controller outputs can directly drive four inputs and maintain the LSTTL valid input levels. If it is necessary to drive also CMOS or PMOS circuits in a 5V system, buffers or a 4.7k pull-up resistor may be added. This resistor together with four LSTTL loads may increase the maximum output LOW level to 0.5V. If it is necessary to drive a CMOS COPS controller input or the standard COPS controller input with a high input option from an LSTTL output, a TTL to MOS buffer or an external 4.7k pull-up resistor may be added.

The low-power COPS controller outputs can directly drive one LSTTL input and maintain the valid LSTTL input levels. If it is also necessary to drive CMOS or PMOS circuts in a 5V system, buffers or a 22k resistor may be added. This resistor together with the LSTTL load will maintain a maximum output LOW level of 0.3V at the serial out (SO) or clock (SK) outputs. If it is necessary to drive a low power COPS controller input with a high input level option from LSTTL output, a TTL to MOS buffer or an external 22k pull-up resistor may be added.

Table 1. COPS Compatible Peripheral Chips

| Peripheral Chips | Process | $V_{CC}$ (V) | DI/SK | | CE | DI Setup | Set Frequency | |
|---|---|---|---|---|---|---|---|---|
| | | | Max. LOW (V) | Min. HIGH (V) | Polarity | Time ($\mu$s) | Min. (kHz) | Max. (kHz) |
| COP450 | CMOS | 2.5-5.5 | $0.3V_{CC}$ | $0.7V_{CC}$ | + | 1.0 | 24 | 265 |
| COP452 | NMOS | 4.5-9.5 | 0.8 | 2.0 | – | 1.0 | 24 | 265 |
| COP470 | PMOS | 4.5-9.5 | $V_{CC}-4$ | $V_{CC}-1.5$ | – | 1.0 | 0 | 265 |
| COP472 | CMOS | 3.0-5.5 | $0.3V_{CC}$ | $0.7V_{CC}$ | – | 1.0 | 0 | 265 |
| COP498 | CMOS | 2.5-6.0 | $0.3V_{CC}$ | $0.7V_{CC}$ | + | 0.3 | 24 | 265 |
| DS8906 | $I^2L$ | 4.75-5.25 | 0.8 | 2.0 | – | 0.3 | 0 | 625 |
| MM5450 | NMOS | 4.75-11.0 | 0.8 | 2.0 | – | 0.3 | 0 | 500 |

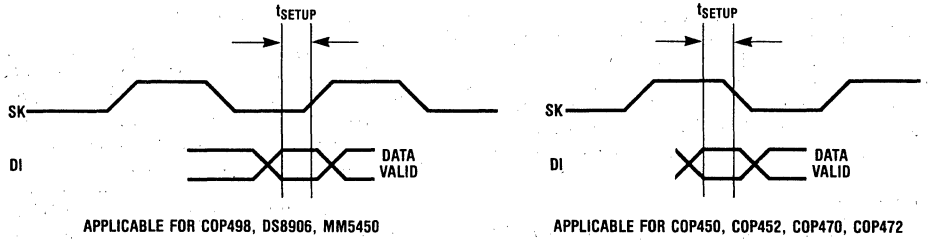



Figure 1. Serial Input Data Timing

# Serial Interface Between COPS™ Microcontrollers and Peripheral Chips

National Semiconductor
COP Brief 9
May 1980



A variety of I/O and data memory expansion chips are available to the COPS™ controllers for different applications. Many of them use the serial port for data transfers, and the COPS controllers allow multiple peripheral chips to be tied in parallel for this purpose (see Figure 1). This paper will discuss the system hardware considerations needed to execute the data transfers. Most COPS controller pins allow various I/O options, and the user should refer to the appropriate data sheet for specific options information. For this discussion, it is assumed that serial input (SI) is a high impedance input for simplicity, and serial output (SO) and clock (SK) are push-pull outputs for lower switching time. All the chips are assumed to have the same power supply. The interface response characteristics may be divided into two parts: static and dynamic.

## I. Static Response

When the output to the serial interface changes state, the input connected to the interface should detect the change. This is done by keeping the output signal level within the specified HIGH or LOW level range of the input. There are two types of transistors used in integrated circuits, namely, MOS and bipolar transistors. They present different equivalent circuits to the output driver and therefore are considered separately.

### 1. MOS (NMOS, CMOS, PMOS)

The MOS inputs look like capacitive loads to these outputs, with a maximum leakage current usually specified. The COPS output driver must be able to sink or source the total maximum leakage current resulting from various inputs connected to it, and keep the signal level within the valid HIGH or LOW value range. Without any leakage, the outputs should reach the same level as that achieved when the output is not loaded.

Different IC devices have different HIGH and LOW input ranges. Most NMOS parts have TTL compatible levels for 5V operation, i.e. 0V to 0.8V for LOW level and 2.0V to $V_{CC}$ for HIGH level. The NMOS COPS controllers also allow a mask-programmed optional range: 0V to 1.2V for LOW level and 3.6V to $V_{CC}$ for HIGH level. Most CMOS parts allow 0V to $0.3V_{CC}$ for LOW level, $0.7V_{CC}$ to $V_{CC}$ for HIGH level. The COP470, a V.F. display controller in PMOS process, has 0V to $V_{CC} - 4V$ for LOW level, and $V_{CC} - 1.5V$ to $V_{CC}$ for HIGH level.

When peripheral chips of different MOSFET types are connected together, the output from the controller must satisfy all the input requirements for each peripheral chip. When peripheral chips with TRI-STATE™ outputs are tied to SI, each of the outputs must satisfy the input level of the COPS controller, while supplying the maximum leakage current to the TRI-STATE outputs. If an input and an output have incompatible levels, external circuits may be necessary for level shifting.

### 2. Bipolar (TTL, LSTTL, $I^2L$)

Standard and CMOS COPS controller outputs are designed to drive one TTL load or four LSTTL loads, whereas the low power COPS controller outputs can drive only one LSTTL load. If more drive is necessary, a buffer will be needed. Standard and low power COPS controller inputs have TTL input levels, therefore multiple TTL/LSTTL TRI-STATE outputs can be connected together directly to SI. The maximum total leakage current at the SI input and all the TRI-STATE outputs determine the maximum number of TRI-STATE outputs that can be tied together. The TTL/LSTTL output levels are not compatible with the CMOS COPS input levels so that extra external components will be necessary for the interface. The simplest solution is to use a pull-up resistor to raise the HIGH output level. A disadvantage is that the LOW output level will be increased.

Bipolar integrated circuits in other processes, e.g., a DS8906 PLL chip manufactured by $I^2L$ process, may have different input levels and different input source and sink requirements. It is necessary to determine whether the COPS output can meet the current requirement and maintain a valid voltage level for the input.

### 3. Mixed (Bipolar and MOS)

Both bipolar and MOS peripheral chips may be used in the same system provided that all the current and voltage requirements are met. Most NMOS and bipolar chips can be mixed together because of similar input voltage levels. CMOS and PMOS chips, on the other hand, cannot be mixed with bipolar chips directly because of the higher HIGH level required. The COPS output HIGH level may be loaded down by the bipolar circuit to an unacceptable HIGH level for the CMOS/PMOS inputs. External circuits will be needed to solve the problem. The simplest solution is a pull-up resistor which improves the source current and raises the output to a higher HIGH level. The resistance should not be too small to increase the LOW level above TTL specification.

## II. Dynamic Response

Provided an ouput can switch between a HIGH level and a LOW level, it must do so in a predetermined amount of time for the data transfer to occur. Since the transfer is synchronous, the timing is relative to the system clock (provided by SK). For example, if a COPS controller outputs a value at the falling edge of the clock and is latched in by the peripheral device at the rising edge, then the following relationship has to be satisfied:

$t_{DELAY} + t_{SETUP} \ll t_{CK}$ (see Figure 1),

where $t_{CK}$ is the time from data output starts to switch to data being latched into the peripheral chip, $t_{SETUP}$ is the setup time for the peripheral device where the data has to be at a valid level, and $t_{DELAY}$ the time for the output to read the valid level. $t_{CK}$ is related to the system

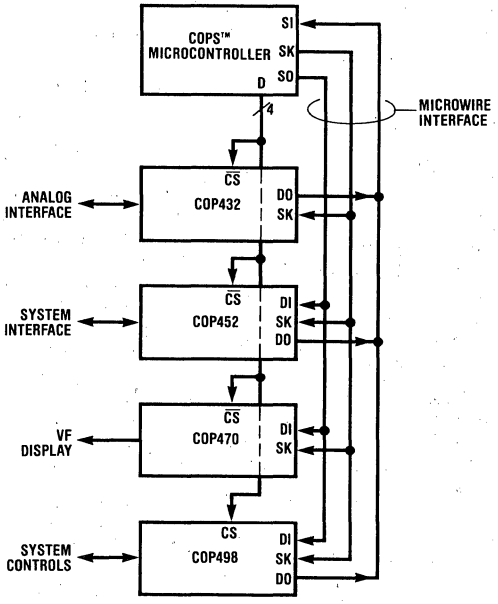
clock provided by the SK pin of the COPS controller and can be increased by increasing the COPS instruction cycle time. Maximum $t_{SETUP}$ is specified in the peripheral chip data sheets. The maximum $t_{SETUP}$ is specified in the peripheral chip data sheets. The maximum $t_{DELAY}$ allowed may then be derived from the above relationship.

Most of the delay time before the output becomes valid comes from charging the capacitive load connected to the output. Each integrated circuit pin has a maximum load of 7 pF. Other sources come from connecting wires and connection from PC boards. The total capacitive load may then be estimated. The propagation delay values given in data sheets assume particular capacitive loads.

If the calculated load is less than the given load, those values should be used. If the calculated load is greater, a conservative estimate is to assume the delay time is proportional to the capacitive load. The COPS data sheet provides two sets of values, one for external loads that includes TTL/LSTTL inputs, the other for pure capacitive loads (MOS inputs).

If the capactive load is too large to satisfy the delay time criterion, then three choices are available. An external buffer may be used to drive the large load. The COPS instruction cycle may be slowed down. An external pull-up resistor may be added to speed up the LOW level to HIGH level transition. The resistor will also increase the output LOW level and increase the HIGH level to LOW level transition time, but the increased time is negligible as long as the output LOW level changes by less than 0.3V. For a 100 pF load, the standard COPS controller may use a 4.7k external resistor, with the output LOW level increased by less than 0.2V. For the same load, the low power COPS controller may use a 22k resistor, with the SO and SK output LOW levels increased by less than 0.1V.

**This is MICROWIRE™**
(Example System)



# COP410L/411L Hardware Subroutine Stack Emulator

National Semiconductor
COP Brief 10
May 1980



The COP410L/411L devices differ from the COP420 devices primarily in the amount of available ROM and RAM, and in the number of subroutine levels available. The COP420 has a 3 level subroutine stack, which allows subroutines to be nested 3 levels deep. If a subroutine is called from the third subroutine level, the first return address is pushed from the top of the stack and lost. The COP410L has a 2 level subroutine stack. When subroutines are nested 3 or more levels deep in a COP410L program, an overflow of the subroutine stack will occur causing the return address from the first subroutine level to be pushed off the top of the stack. The program will not function properly if designed to return from the first subroutine in the normal manner, since that return address will have been lost. The COP410L/411L Hardware Subroutine Stack Emulator is designed as a reliable aid in finding subroutine stack overflow conditions in COP410L programs.

The difficulty in finding a subroutine stack overflow condition lies in emulating the COP410L program. The COP402, which is a ROMless functional equivalent of the COP420, may be used to emulate the COP410L device. However, the COP402 has a 3 level subroutine stack, as does the COP420. Therefore, when emulating a COP410L device, the program may exhibit an overflow of the 2 level stack which will not be detected in the COP402 emulator. Special care must be taken when writing COP410L programs to insure that the 2 level subroutine stack is not violated.

The most obvious method of verifying the 2 level COP410L stack is to systematically count the subroutine levels directly from the program listing. With the listing, the programmer may follow through each subroutine, counting the level of subroutines called from those routines. If subroutines are nested 3 levels deep, the COP410L program will exhibit a stack overflow condition. This could cause improper program execution in the COP410L and should be corrected before submitting the program for production.

An alternate method of verifying the 2 level subroutine stack is by writing a COP420 program that calls the entire COP410L program as a subroutine. This effectively diminishes the COP402 stack by 1 level when emulating the program. At the end of logical execution of the COP410L program a RET statement must be inserted to return to the COP420 routine. If the COP410L program executes correctly and then returns to the COP420 routine at the correct location, the COP410L program may be free of stack overflow violations. The breakpoint feature of the COP Product Development System (PDS) is helpful in detecting the return from the COP410L program. A breakpoint should be set at the COP420 program address immediately following the JSR statement that transfers control to THE COP410L program. The PDS will breakpoint the COP402 at the return address from the COP410L program, providing there are no stack violations in the COP410L routine. All COP410L subroutines must be terminated in a RET or RETSK statement for the PDS to be useful in verifying the 2 level stack. The COP420 return address will be lost if the stack is deliberately overflowed in the COP410L routine. Losing the COP420 return address will cause the PDS to indicate an erroneous stack overflow condition by returning from the COP410L routine to an incorrect location. For this reason, all subroutines must end in a RET.

The previously described procedures for evaluating the 2 level subroutine stack are often unreliable due to different program structures. The most reliable method of locating a subroutine stack overflow is to use the COP410L/411L Hardware Subroutine Stack Emulator. This circuit will count the number of consecutive RET or RETSK instructions that are executed. Counting the return from subroutine instruction rather than the jump to subroutines allows deliberate overflow of the subroutine stack, which may be necessary in some COP410L programs.

## Block Diagram Description

The COP410L/411L Hardware Stack Emulator is designed to be used with the COP400-E02 In-Circuit Emulator Card. The system works in the following manner (see Block Diagram, Figure 1). The COP410L program data is latched onto the Hardware Stack Emulator card via the $\Phi_1$ signal. The program data is decoded by the Data Decode PROM and the appropriate control signals are generated. During the second half of a 2-cycle instruction, the COP402 sends the SKIP line high. This signal is used to disable the data decode PROM control signals by forcing the PROM to decode data from the upper half of memory in which all control signals are inactive (see PROM Data, Table 1). The JSR and JSRP instructions push the stack and the RET and RETSK instructions pop the stack. The LQID instruction is a 2 cycle instruction that first pushes the stack, then during the second instruction cycle, pops the stack. When a LQID is executed at the second subroutine level a stack overflow condition will occur. Each time the stack is pushed, the stack increment logic will clock the count down input of the stack counter once and each time the stack is popped, the stack decrement logic will clock the count down input of the stack counter. The stack counter counts a maximum of 2 pushes of the subroutine stack, after which the increment logic is disabled. This allows deliberate overflowing of the subroutine stack. A subroutine stack overflow is registered when the stack counter is popped 3 consecutive times. At this time, the stack counter underflows and disables both the stack increment and decrement logic, and lights the overflow indicator.

The COP400 devices allow single-byte jumps (JP) within any single page boundaries. Single-byte jumps are also allowed anywhere within the boundaries of the subroutine pages (pages 2 and 3). Single-byte subroutine calls (JSRP) are valid from anywhere in the program to routines on page 2. The op-codes for the JSRP instructions are identical to the codes for the single-byte jumps in pages 2-3. For this reason the JSRP call may not be used

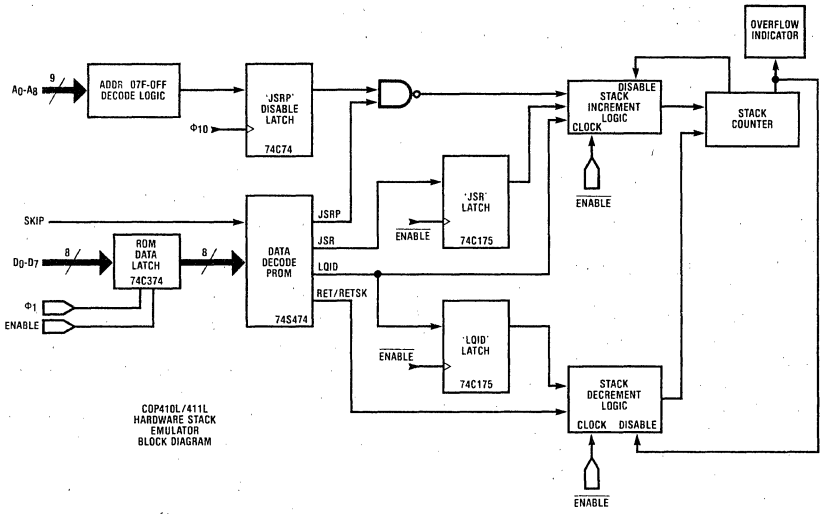



**Figure 1. COP410L/411L Hardware Stack Emulator Block Diagram**

on pages 2 or 3. However, the decode PROM on the Hardware Stack Emulator will generate the JSRP signal whenever a single-byte jump is executed on pages 2 or 3. The Address Decode Logic sets the JSRP disable latch when the program is executing in pages 2 or 3 to inhibit the JSRP control signal generated by executing a single-byte jump on pages 2 or 3. Address 07F, while not actually on page 2, is also decoded since a single-byte jump from this location to anywhere in pages 2 or 3 is also coded as a JSRP and the JSRP signal generated must, therefore, be inhibited. The JSR instruction is a valid subroutine call anywhere within the COP410L program. Consequently, the Address Decode Logic does not inhibit the JSR signal.

**Table 1. PROM Data**

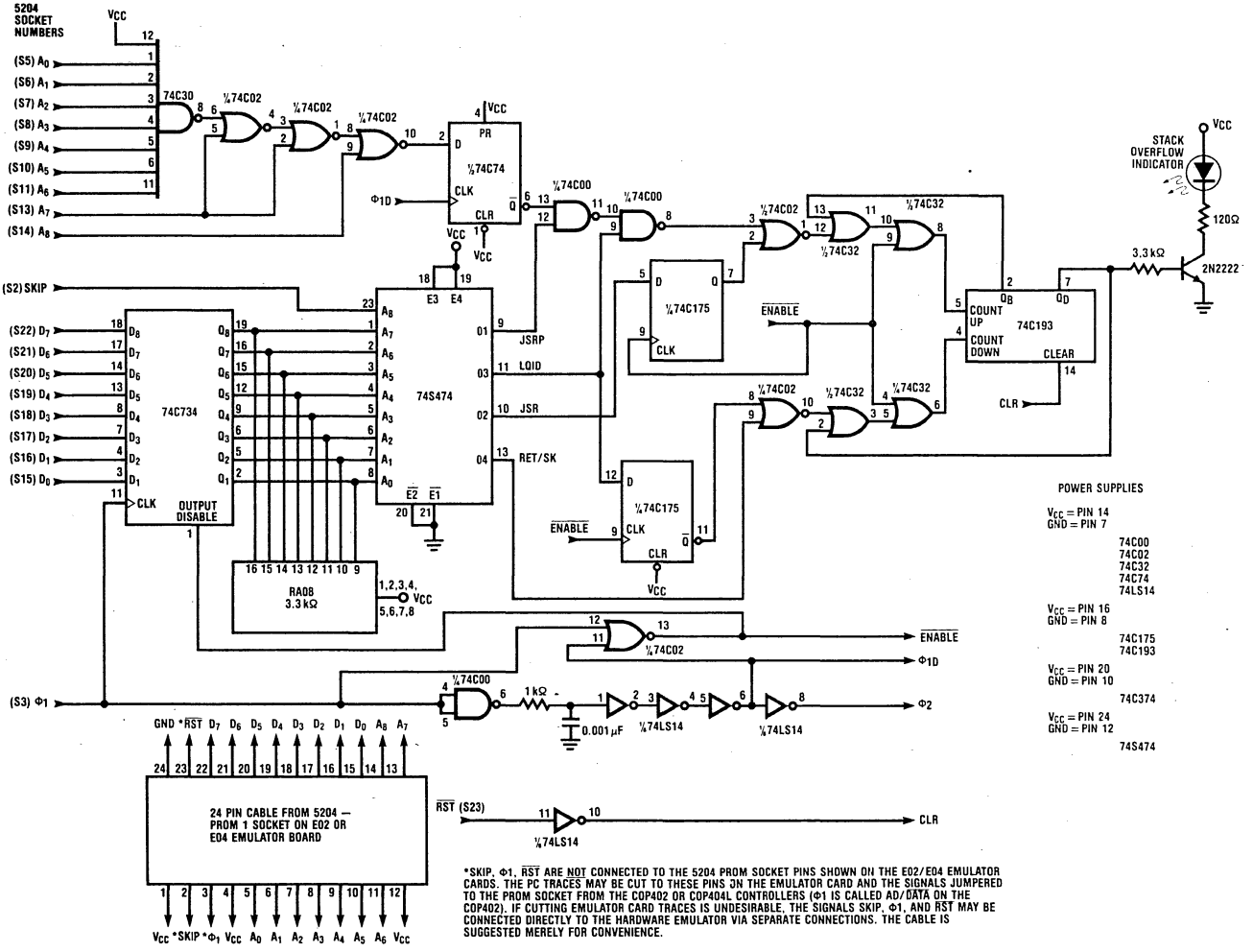
| Address | Data | Address | Data |
|---|---|---|---|
| 000 | 04 | 070 | 04 |
| ↓ | ↓ | ↓ | ↓ |
| 047 | 04 | 07F | 04 |
| 048 | 0C | 080 | 05 |
| 049 | 0C | ↓ | ↓ |
| 04A | 04 | 0BE | 05 |
| ↓ | ↓ | 0BF | 00 |
| 067 | 04 | 0C0 | 04 |
| 068 | 06 | ↓ | ↓ |
| ↓ | ↓ | 1FF | 04 |
| 06F | 06 | | |

## System Timing Description

The COP402 generates the AD/$\overline{\text{DATA}}$ signal which is called $\Phi_1$ in the COP410L/411L Hardware Stack Emulator system (see COP402 Data Sheet). During program execution, address information is outputted by the COP402 during the time that $\Phi_1$ is high and program data is inputted by the COP402 when $\Phi_1$ is low. The $\Phi_1$ signal is the basis for all timing on the Hardware Stack Emulator. $\Phi_{1D}$ is created by delaying $\Phi_1$ by approximately 3$\mu$s. This signal is used to clock the decoded address into the JSRP disable latch. The MM74C74 latches data on a low-to-high clock transition. The $\Phi_{1D}$ signal is used to insure stable, valid data at the D input of the latch prior to the clock pulse. $\overline{\text{ENABLE}}$ is generated by taking the NOR of $\Phi_1$ and $\Phi_{1D}$. During the low period of $\overline{\text{ENABLE}}$, the outputs of the ROM data latch are enabled to drive data to the decode PROM. When the outputs are disabled, the inputs to the decode PROM are held high with pull-ups. At the rising edge of $\overline{\text{ENABLE}}$ valid data is latched into the LQID and JSR latches. The stack counter is also clocked on the rising edge of $\overline{\text{ENABLE}}$. The data that was set up at the stack counter inputs during the previous clock cycle enables or disables the $\overline{\text{ENABLE}}$ clock appropriately.

## Building the Hardware Stack Emulator

All signals required by the Hardware Stack Emulator are available on the COP400-E02 Emulator Card. The signals may be brought to the Hardware Stack Emulator Card with a 24 conductor ribbon cable plugged into the 5204 PROM 1 socket on the E02 Emulator Card. The signals $\Phi_1$, SKIP, and $\overline{\text{RST}}$ are not available at the 5204 PROM 1 socket so they must be either jumpered to the socket or directly to the Hardware Stack Emulator board. If it is desirable to run these signals via the 24 conductor cable, then 3 printed circuit traces must be cut on the E02 Emulator Card. The traces to pins 2, 3, and 23 of the 5204 PROM 1 may be cut since these signals are not required by the Hardware Stack Emulator. SKIP, $\overline{\text{RST}}$, and $\Phi_1$ may then be soldered to the socket and all required signals will then be carried over the 24 conductor cable. Power and ground are also carried to the Hardware Stack Emulator board by the 24 conductor cable. The complete circuit schematic and 5204 PROM 1 socket pinouts are shown in Figure 2.




*SKIP, Φ1, RST ARE NOT CONNECTED TO THE 5204 PROM SOCKET PINS SHOWN ON THE E02/E04 EMULATOR CARDS. THE PC TRACES MAY BE CUT TO THESE PINS ON THE EMULATOR CARD AND THE SIGNALS JUMPERED TO THE PROM SOCKET FROM THE COP402 OR COP404L CONTROLLERS (Φ1 IS CALLED AD/DATA ON THE COP402). IF CUTTING EMULATOR CARD TRACES IS UNDESIRABLE, THE SIGNALS SKIP, Φ1, AND RST MAY BE CONNECTED DIRECTLY TO THE HARDWARE EMULATOR VIA SEPARATE CONNECTIONS. THE CABLE IS SUGGESTED MERELY FOR CONVENIENCE.

**Figure 2. COP410L/411L Hardware Stack Emulator**

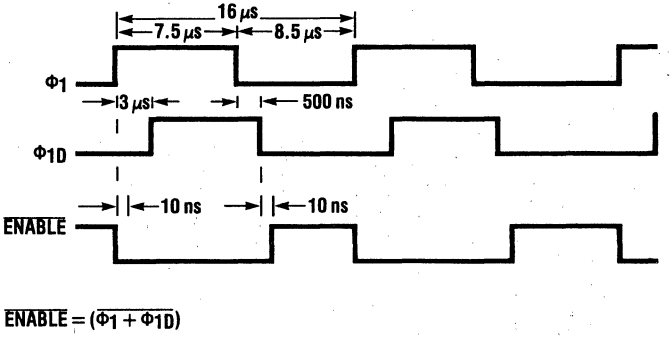



Figure 3. COP410L/411L Hardware Stack Emulator Signal Timing Relationships

# Power Seat with Memory

National Semiconductor
Richard W. Kovener
COP Brief 11
May 1980





### Introduction

As cars continue to be downsized, more extra features are being offered to the car purchaser to individualize the car to his personal taste. This is especially true with electronic equipment. Automobiles are now available with digitally tuned radios, trip computers, digital gauges and other electronic systems. These have been made possible only recently by the increasing level of semiconductor integration and the resulting lower cost for the components that make up each system.

This article describes another application for electronics in an automobile, a power seat with position memory. This seat features powered adjustment in 8 different directions, the ability to store 2 sets of position information in memory, and instant recall and automatic adjustment to either of the 2 positions. The seat can therefore be adjusted to accommodate 2 different drivers or 2 different driving positions for the same driver and automatically adjust to either of these positions on demand.

### System Description

A block diagram of the seat control system is shown in Figure 1. The heart of the system is the COP420L microcontroller. This part is one of National Semiconductor's COP400 Family of 4-bit, 1-chip microcontrollers. Motor control information is output to the TRI-STATE® octal latch and information from the seat sensors is input through the TRI-STATE octal buffer. Manual adjustment of the seat is provided by 8 switches mounted on a console. These manual controls have priority over automatic control via the TRI-STATE control pin on the latch. In addition, the controller software will terminate automatic control if it detects the seat being adjusted in a way different from its programmed positions. This provides for manual override and is necessary as a safety precaution. The system will operate manually even with the controller part removed, which gives a fail-safe operation.

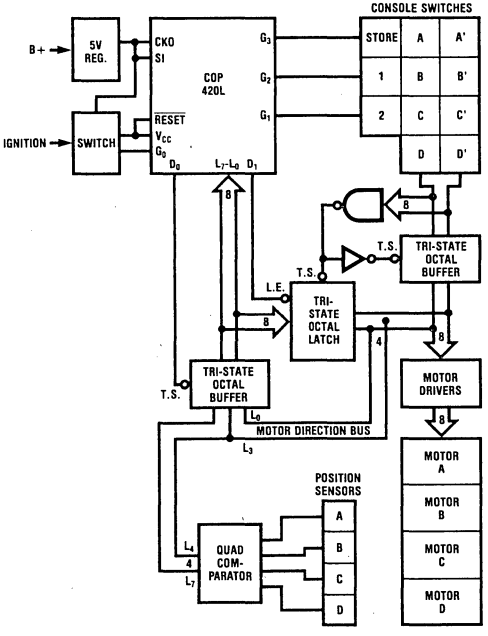



**Figure 1. Block Diagram**

### The Controller

The COP420L is an N-channel MOS device with 1K × 8-bit program memory and a 64 × 4-bit data memory. Its internal architecture is shown in Figure 2, and electrical specifications are shown in Figure 3. In this application, the bidirectional TRI-STATE L lines are used to output motor control information to the motor control latch and also are used to input

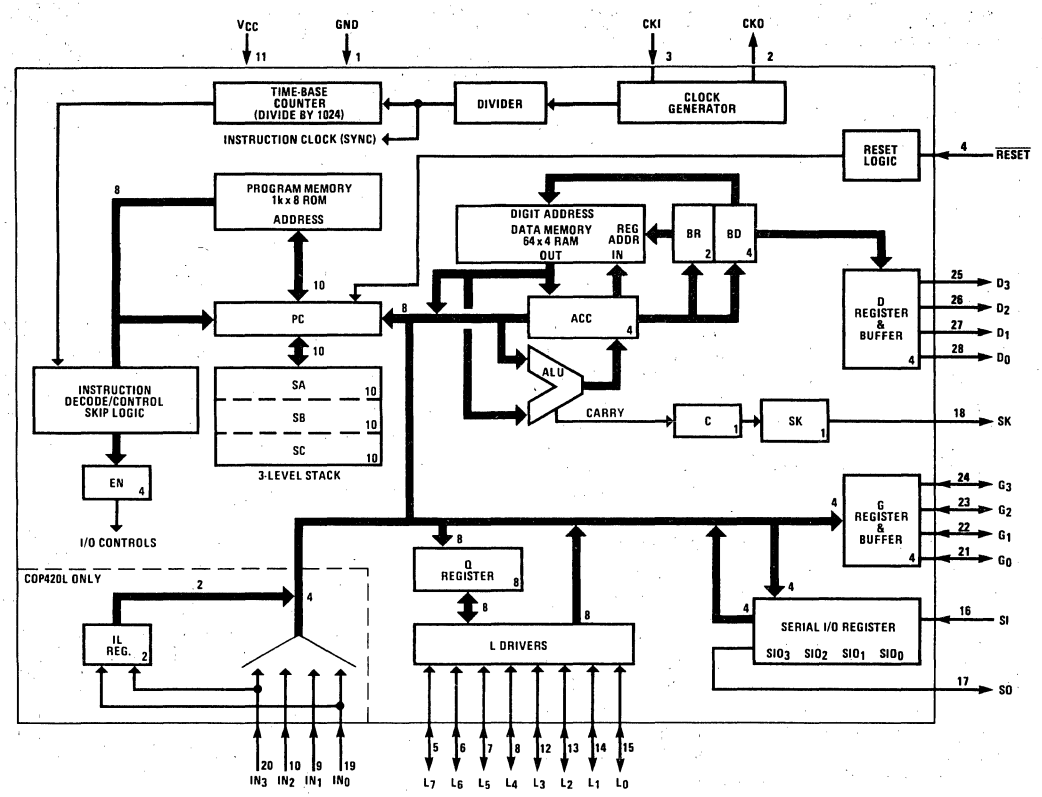



**Figure 2. COP420L Block Diagram**

| | |
|---|---|
| Operating Voltage | 4.5V–9.5V |
| Operating Supply Current | 8mA (max) |
| RAM Supply Requirements | 3mA (max) @ 3.3V |
| Minimum Instruction Cycle Time | 16μs |

**Figure 3.**

seat position sensor information. The selection of the L lines as inputs or outputs is done through software control and a $D_0$ line controls the operation of the TRI-STATE buffer to coordinate the reading of sensor information or outputting motor control information. The $D_1$ line controls the operation of the TRI-STATE latch. The $G_{1-3}$ lines are used to detect closure of the memory control keys. Pressing 1 preceded by pressing SET will store the present seat position in memory location 1 and pressing 2 preceded by SET will store position information in memory location 2. Pressing 1 or 2 without first pressing SET will cause the seat to adjust to the respective previously stored position. The remaining $G_0$ line is used to detect the car's ignition being turned off so the seat can be moved back to allow easy exit from the car.

The IN lines of the COP420L are not used in this design but could be used to interface more memory control keys. There is available space in RAM to store additional seat positions if desired.

The CKO pin is used to provide power to the on-chip RAM in order to retain seat position information when the ignition switch is turned off. Power to the controller and other components is removed in this condition to minimize current drain on the automobile battery.

### System Power Supply

Careful consideration must be given to designing power supply circuitry for automotive electronic systems. Adequate protection must be provided against the electrical transients present in the automotive electrical system. These transients are listed in Figure 4. In addition to these transients, there exists the possibility of 2-battery jumps (+24V) and reversed 2-battery jumps (−24V). All of these must be protected against for reliable operation.

National Semiconductor's LM2930 was specifically designed for supply regulation in automotive electric systems. Its electrical characteristics are listed in

| | |
|---|---|
| Load Dump | 50V $\tau = 200$ms |
| Inductive Load Switching | ±250V $\tau = 1$ms |
| Mutual Coupling | ±450V $\tau = 0.1\mu$s |

**Figure 4. Automotive Transients**

| | |
|---|---|
| Max Operating Input Voltage | 26V |
| Over-Voltage Protection | 40V |
| Output Voltage<br>$6V \leq V_{IN} \leq 26V$, $5mA \leq I_O \leq 200mA$ | 4.5V-5.5V |
| Line Regulation<br>$6V \leq V_{IN} \leq 26V$ | 80mV max |
| Load Regulation<br>$5mA \leq I_O \leq 200mA$ | 50mV max |
| Dropout Voltage<br>$I_O = 200mA$ | 0.6V max |

**Figure 5. LM2930 Specifications**

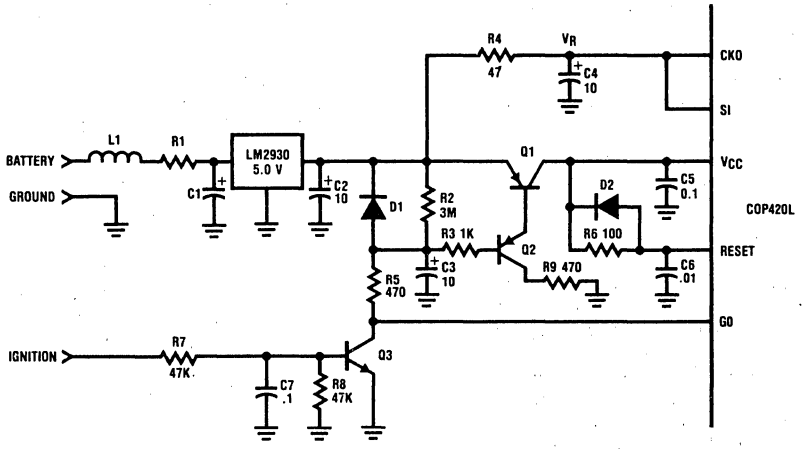



**Figure 6. Power Supply Circuitry**

Figure 5. This part is internally protected against reverse battery installation and 2-battery jumps. Therefore, all that is needed is to protect the part from input voltages over 40V. This is easily done with an R-L-C circuit. Designing for load dump protection will give protection against the larger but faster transients.

In order to minimize battery drain, $V_{CC}$ is turned off to all the circuitry except for the COP's RAM when the ignition is turned off. Refer to Figure 6. When the ignition is on, Q3 provides drive to Q1 and Q2. Q1 also holds $G_0$ low. When the ignition is turned off, the program software detects the low on $G_0$ being released and performs a routine to park the seat. $V_{CC}$ is supplied to the controller and circuitry until C3 charges up through R2 to turn off Q1 and Q2, allowing sufficient time for the seat to reach its parked position. Each time $V_{CC}$ is turned on, the program software checks the contents of the serial register to see if power to the RAM has been lost. If the serial register is all "ones," power has not been lost. If the contents are all "zeros," RAM power has been lost and the RAM and seat are initialized.

This procedure also occurs if the car battery has been disconnected. When it is reconnected, C3 is initially discharged and turns on Q1 and Q2. $V_R$ is delayed by R4 and C4 and therefore the serial register is loaded with "zeros" and the RAM and seat are initialized. C3 then charges up and turns off Q1 and Q2 and the system returns to standby. (Note: The values of the timing components have been established experimentally.)

### System Interface — Output

The 8 different directions of movement of the seat are provided by 4 drive motors. These 8 directions are:

A — Tilt Seat Back Rearward
A′ — Tilt Seat Back Forward
B — Move Seat Backward
B′ — Move Seat Forward
C — Front of the Seat Up
C′ — Front of the Seat Down
D — Rear of the Seat Up
D′ — Rear of the Seat Down

The motors that move the seat typically draw 2 amps each when running, but draw up to 10 amps each when stalled. The motors also require bidirectional

drive to operate them both in forward and reverse. For these reasons, relays were chosen over semiconductors for the interface.

A high voltage open collector buffer is used to energize the desired relay from the motor control bus. Zener diodes are necessary from the collectors to ground to clamp the inductive turn-off transient to a voltage below the $BV_{CEO}$ of the transistor. These diodes also provide protection for the buffers against load dump and the other transients on the battery supply line.

### System Interface — Input

For the controller to be able to store a seat position in memory and then later to adjust the seat to that position, it is necessary for the controller to know the relative seat location at all times. This is accomplished through sensors mounted on the seat mechanism.

In the prototype, two types of sensors were used. Both types of sensors provided digital information to the controller.

A photodetector package was used with a slotted disc on the seat back. The disc was mounted on the gear mechanism, and as it revolved it interrupted the light source in the detector package as the seat back angle was adjusted. A comparator is used to detect these interruptions and provide logic level compatible pulses to the controller. The controller keeps a running count of these pulses to know where the seat back is at all times. Direction information is fed back to the controller from the motor control bus so the controller knows whether to add or subtract the pulses. This is shown in Figure 1.

The other 3 seat movement mechanisms required a different type of sensor due to their construction. These mechanisms are driven through a flexible cable by a motor. A photodetector sensor could not be added without some major modifications. Therefore, the sensor selected was a speed sensor commonly used for automobile cruise control and could be inserted between the motor and the drive cable. This type of sensor generates an AC waveform that corresponds to the revolutions of the motor. The AC signal is conditioned by a comparator to produce logic level pulses. The sensor is constructed with multiple poles so a divider is used after the comparator to provide the correct number of pulses for the full travel of the seat mechanism.

### An Alternative Approach

Another approach to a seat control system is to use analog sensors instead of digital sensors to track seat position. A block diagram of this approach is shown in Figure 7. The position sensors are potentiometers mounted to the seat mechanism. The multiplexer, under software control, selects which sensor is to be measured and the A-to-D converter inputs the position information to the controller in 8-bit binary format.

It is not necessary in this approach to keep a constant account of the seat's position since it can be determined at any time by polling the potentiometer sensors. The software is therefore much simplified and allows the use of a COP410L which has one-half the memory sizes of the COP420L. The signal conditioning circuitry for the digital sensors that was described earlier is also eliminated. These two things plus the lower cost for potentiometer sensors result in an overall system cost advantage.

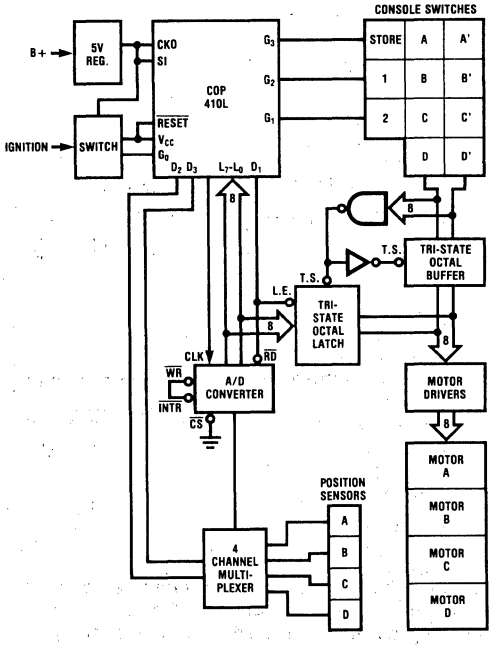



**Figure 7. Block Diagram**

### Conclusion

A control system for a power seat that has the ability to store and recall preferred driving positions can be designed using a low-cost 4-bit, 1-chip microcontroller and adds to the list of electronic systems being offered today for safety, comfort, and convenience of the automobile driver.

### Acknowledgements

My thanks to Recaro, USA, for supplying the seat for the prototype described in this article and to Jim Troutner, National Semiconductor, for writing the software routines.

# An Automotive Diagnostics Display

National Semiconductor
Paul M. Goyke
COP Brief 12
May 1980



### Introduction

The continued downsizing of the automobile has put a premium on instrument panel space. This has provided the opportunity for electronics to merge the various displays now found in the current automobile to one central display to conserve valuable panel space and provide new marketable features. The advances in semiconductor technology have made this concept both technically feasible and cost effective.

### System Description

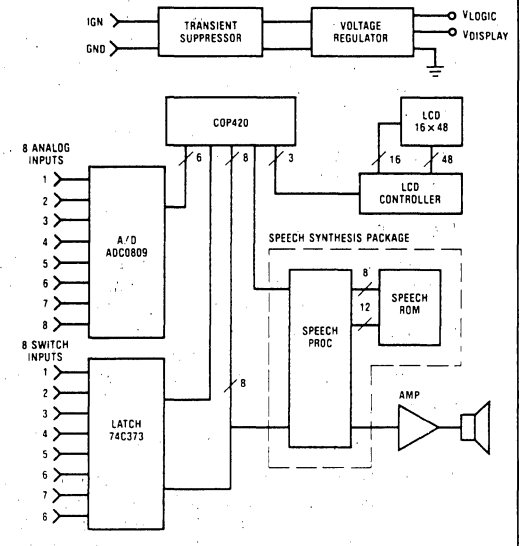
The Diagnostic Display consists of a microcomputer, analog input section, digital input section, liquid crystal display and controller, a speech synthesis package, and a power supply which is outlined on the block diagram. The input section of eight analog channels and eight switch channels was chosen only to demonstrate capability, as the number and mix of analog and digital channels would be tailored to the number of diagnostic messages desired.

From the block diagram, it can be seen that the microcomputer communicates to the liquid crystal display controller via a three-wire bus termed Microwire™ This implies that the display and its controller could be remotely mounted in the instrument cluster, steering wheel, overhead console, etc., while the remainder of the circuitry could be mounted elsewhere under the dashboard.

### Microcomputer

The microcomputer is a National Semiconductor COP 420 which functions as the Diagnostic Display's system controller. The COP 420 is a single-chip processor fabricated using N-channel silicon gate technology. The processor contains 1K x 8 of ROM, 64 x 4 of RAM, clock generator, and 23 input-output lines on board.

In this application, the eight bidirectional L lines are used as a general purpose bus to communicate with the analog-to-digital converter, the switch input latch, and the speech synthesis package. The four G lines are used as chip selects for each of the four peripherals. The four D lines and one IN line are used to control the analog-to-digital converter and to address a particular analog channel. Two additional lines, the SK clock output, and SO serial output line are used to communicate to the liquid crystal display controller.




Diagnostics Display

In normal operation, the microcomputer digitizes and stores all eight analog inputs and stores the states of the eight switch inputs in RAM. If any input is not within programmed limits, it displays the appropriate message and selects the proper verbal phrase. When more than one input is activated simultaneously, the one with the higher priority is selected.

### Analog Input Section

The analog input section consists of National Semiconductor's ADC0809, which is an eight-bit, eight-channel analog-to-digital converter. This CMOS converter is directly compatible with microprocessor control logic.

The purpose of the A/D converter is to interface with new analog sensors such as outside temperature or paralleling existing sensors such as fuel level.

The threshold levels, where the microcomputer displays a given message, is programmable by the application in software. Although eight inputs are shown, any number could be accommodated to suit the system requirements.

Referring to the block diagram, the analog-to-digital converter is controlled by the microcomputer with six control lines. The control lines address the analog channel, start the conversion, signal the microcomputer when conversion is complete, and enable the TRI-STATE™ drivers. All eight analog values are stored in sixteen four-bit memory locations via the eight-bit data bus. Typical conversion time per channel is 100 microseconds with a maximum total unadjusted error of plus or minus one bit. If additional accuracy is needed, a selected part is available with one half bit accuracy.

### Digital Input Section

The digital input section consists of a 74C373 CMOS TRI-STATE™ octal latch. Upon command from the microcomputer, the 74C373 latches the input data and outputs it over the eight-bit data bus. The purpose for the digital input section is to input data from mechanical switches such as door jamb or turn signals.

### Liquid Crystal Display and Controller

The liquid crystal display is a medium area dot matrix multiplexed display. The matrix consists of 16 rows by 48 columns. The display is driven by four CMOS driver circuits, each of which is capable of controlling one quadrant of the display or 8 rows by 24 columns.

The display driver consists of a serial input shift register, an 8 x 24-bit memory, temperature dependent output drivers, and associated clock circuitry. Communication between the driver circuits and the microcomputer is via a three-wire Microwire™ bus in a serial fashion. The data consists of an address of a dot cluster, the data of whether a dot is on or off, and a read/write bit to indicate whether data is being written or read from memory. Once the memory is loaded with the desired pattern, the display is automatically refreshed by the display driver, so no further action is required by the microcomputer. Each driver chip also has an input for temperature compensation of the liquid crystal's threshold voltage. The compensation is in the form of a simple variable voltage from a thermistor or similar transducer.

### Speech Synthesis Package

The speech synthesis package is a system consisting of multiple N-channel devices. It contains a speech processor and speech ROM, and when used with an external filter and amplifier, generates high quality speech.

The speech processor accepts an eight-bit word which is the starting address of the word or phrase to be spoken. Additionally, there is a chip select, write, and interrupt pin to make the part Microbus™ compatible with many microprocessors. An interrupt is generated at the end of any speech sequence, so several sequences or words can be cascaded for additional flexibility.

The speech ROM or ROMs can be as large as 128K bits to be addressed directly by the speech processor. The ROMs can be either static or dynamic clocked types, as the speech processor has a ROM enable pin for use with dynamic ROMs. The ROMs in the package contain the compressed speech data as well as the frequency and amplitude data required for speech output.

### Power Supply

The power supply in an automotive electronic system is perhaps the most critical part for reliable operation. Its function is to transform the noisy vehicle power to the various voltages required by the system. In the Diagnostics Display, the speech processor requires seven volts, the liquid crystal display requires ten volts, while the rest of the circuit operates at five volts.

In addition to supplying the correct voltages, the power supply must protect the circuit from overvoltages and transients. The LM2930 is the first part in a family of voltage regulators designed for automotive applications. This regulator exhibits a low voltage in to voltage out ratio which provides a constant five volts out, for input voltages as low as 5.6 volts. Additionally, this regulator can accept input voltages to 40 volts, which provides protection against two-battery emergency starts. The large maximum input voltage of 40 volts also simplifies the transient protection network, as now the network needs only to protect the regulator from transients greater than 40 volts.

### Conclusion

The purpose of the Diagnostics Display is to show a broad design base and present some novel applications for advanced products such as speech synthesis and multiplexed liquid crystal displays. It also shows a 4-bit COP 420 replacing a more costly 8-bit type processor in this application. This is only one example of the many applications of electronics to automotive instrument panels.

# An Electronic Speedometer and Odometer with Permanent Mileage Accumulation

National Semiconductor
Richard W. Kovener
COP Brief 13
May 1980



### Introduction

As today's automobile becomes more electronic with the addition of engine control systems and digital instrumentation, a need has developed for a method of implementing an electronic odometer that will retain total mileage accumulation information under all conditions, including the loss of vehicle electrical power. This need is made greater by the reduction in available instrument panel space due to downsizing and by a proposed Federal Motor Vehicle Safety Standard requiring tamper-proof odometers.

The requirement of non-volatile mileage storage has been an obstacle for automotive electronic odometer designs. Although an EAROM (Electrically Alterable Read Only Memory) can be used, they are relatively expensive and have a limited number of erase-write cycles. The system described here uses a fusible link bipolar PROM as the mileage storage device and a low-cost, 4-bit microcontroller as the programming device.

### System Description

A block diagram of the electronic speedometer/odometer is shown in Figure 1. The counting of mileage pulses and the PROM programming are done by a COP 420L, a 4-bit, 1-chip microcontroller (see Figure 2). The mileage pulses are input to the controller through its serial data port. These pulses are counted and stored in RAM. These pulses can be from any type of sensor as long as they have TTL compatible levels.

When the number of pulses counted equals one-tenth of a mile traveled the mileage stored in RAM is updated. The number of pulses equivalent to 0.1 mile is of course dependent on the mileage sensor. The algorithm for converting from pulses to miles is a software routine and can be modified accordingly to work with various mileage sensors.

A separate count of pulses is kept in another location in RAM for a trip odometer. This mileage can be output on the odometer display by alternate operation of a pushbutton. Another pushbutton clears the trip odometer register.

The speedometer operation is similar to the odometer routine but the updating is dependent on time instead of mileage. The number of pulses counted during a period of time translates to the vehicle speed. A software algorithm converts the number of pulses to speed using a conversion factor dependent on the mileage sensor and display mode selected.

The bipolar PROM is programmed with mileage information when the running mileage count in RAM reaches a predetermined number. The mileage increment that is permanently stored in the PROM is controlled by the operating software and determines the size of the PROM that is required. This is described in more detail in a later section.

When a mileage bit is to be programmed in the PROM, the address of this bit is latched into the address latch by the controller. The proper data for this bit is then put on the 8-bit bus and the proper programming sequence is initiated.

Since the mileage information in the PROM is non-volatile, all operating power is turned off to the circuit when the vehicle ignition is off except for a standby voltage to maintain the trip mileage and running mileage counts stored in the RAM of the controller.

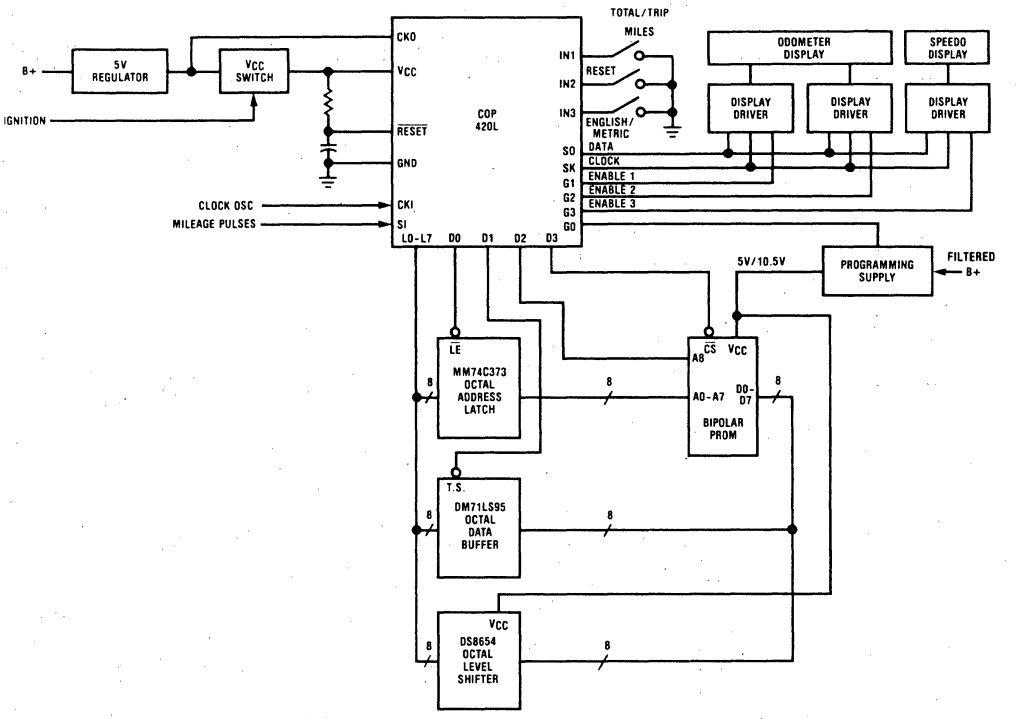



**Figure 1. Electronic Speedometer/Odometer**

### System Software

Using a microcontroller in an odometer design allows great flexibility of operation and features. The flow chart in Figure 3 is for the prototype speedometer/ odometer shown in the block diagram.

When the ignition is turned on, all registers are cleared by the on-chip reset circuitry. After some initial housekeeping, the controller reads a code number from the PROM. This code number is used to provide traceability of the odometer to the vehicle and confirms to the vehicle owner the authenticity of the odometer. The number recorded in the PROM could simply be the vehicle identification number or some other number that has some corresponding vehicle significance. This code number prevents an ingenious individual from replacing the mileage PROM with one of lesser mileage. The number is coded in some manner to prevent easy deciphering.

After this number is displayed for an adequate time, the running mileage in RAM is compared to the total mileage recorded in the PROM. If they are within the predetermined permanent mileage increment the running mileage is accurate and is displayed. If they are not, the RAM has lost data due to a loss of standby power and is restored by transferring the total accumulated mileage recorded in the PROM to the register in RAM. The running mileage is then displayed by the odometer.

The three keys controlling the display mode are read next. Either trip mileage or running mileage is displayed according to the operation of the display key. The trip odometer is cleared when a key depression is detected on the reset button. If a closure is detected on the English/Metric key, a flag is set and all information is displayed in English or Metric units depending on the previous display mode. Next the mileage pulse from the sensor is read from the serial input register. The COP420L has a feature under software control that makes the serial I/O register a binary counter.

In this mode of operation the counter counts high to low level transitions at the SI input. The controller then reads the contents of the register at a rate equal to or greater than the pulse output frequency of the mileage sensor at the maximum vehicle speed. All of the count registers are then incremented.

The mileage registers are examined next. When the pulses counted are equal to 0.1 mile traveled, the trip odometer register and the running mileage register are incremented.

In similar fashion, when the running mileage has accumulated additional mileage equal to the permanent storage increment, the data is programmed into the PROM. The odometer display is updated after the display flags are examined. Either the total

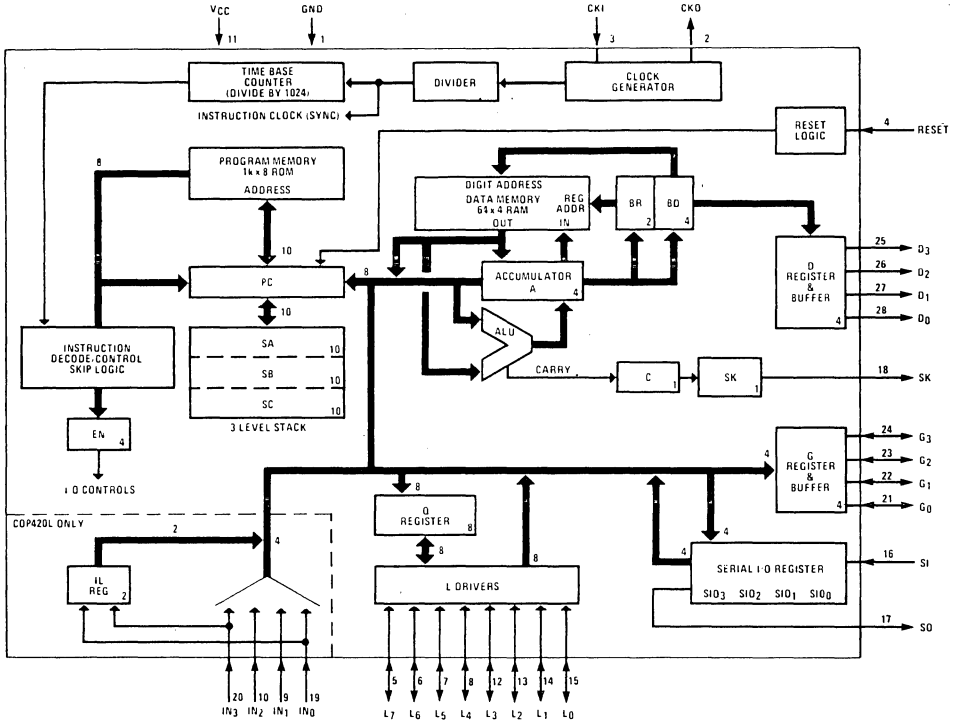



**Figure 2. COP 420L/421L Block Diagram**

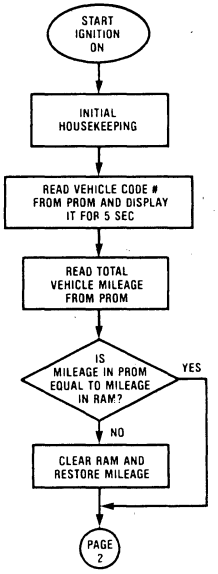



**Figure 3. Electronic Speedometer/Odometer Flow Chart**

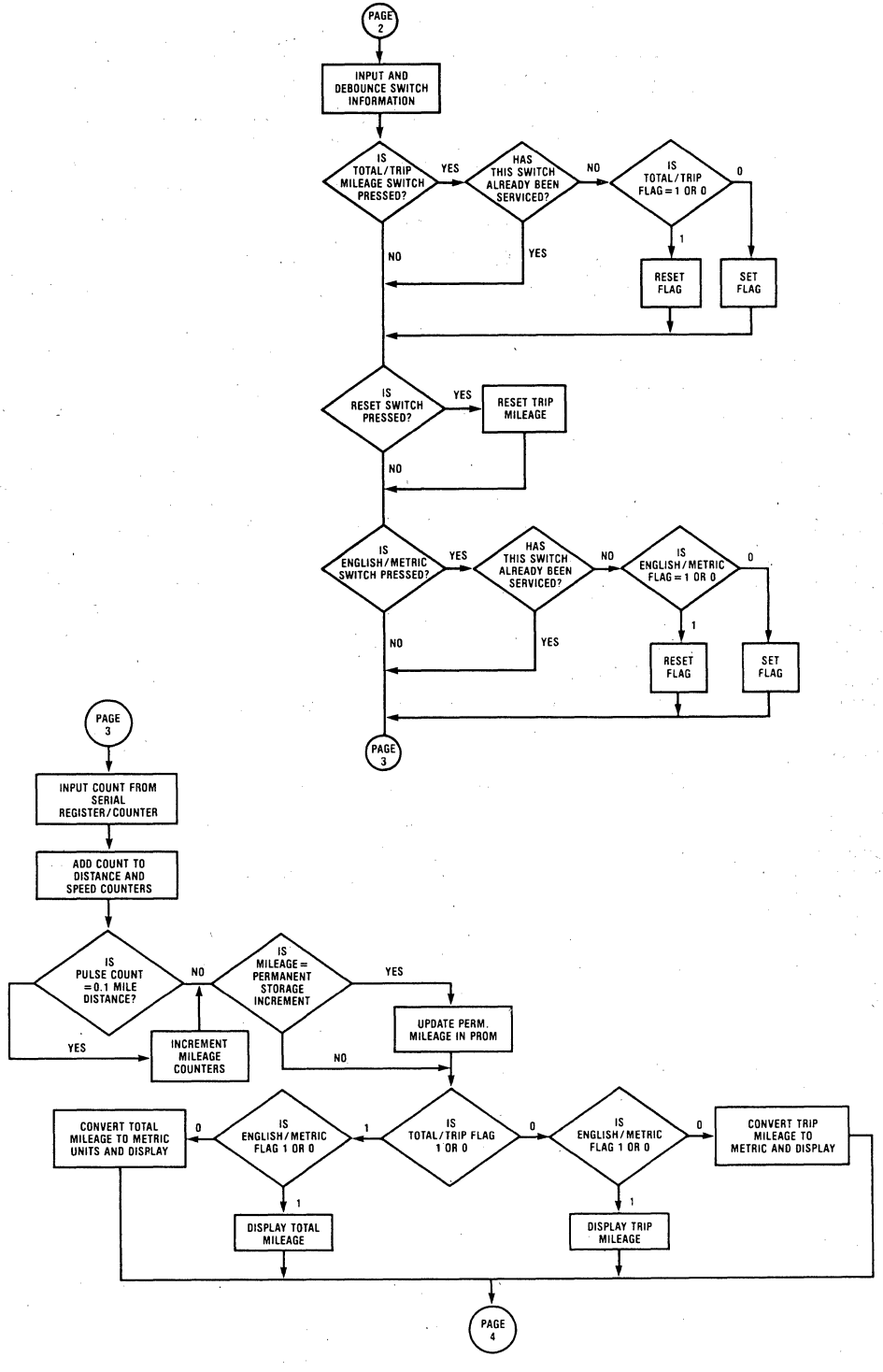



**Figure 3. Electronic Speedometer/Odometer (continued)**

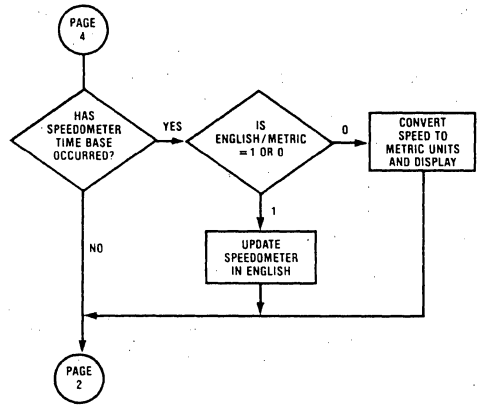



**Figure 3. Electronic Speedometer/Odometer (continued)**

mileage or trip mileage is displayed in English or Metric units according to the corresponding flag condition.

If the time since the last update of the speedometer is equal to the time base for calculation, the speedometer is updated according to the number of pulses counted during this period. Otherwise, the speedometer reading is not changed.

After this step, the programming returns to reading the display mode switches and continues the loop.

### PROM Selection and Programming

The size of the PROM selected for permanent mileage storage depends on the mileage resolution desired. A 512 x 8-bit PROM as shown in the block diagram will allow a bit to be programmed every 25 miles for a storage capability of more than 100,000 miles. If 100-mile resolution is adequate, then a 1024-bit PROM could be used, resulting in a lower system cost.

The proper algorithm for programming fusible link PROMs is dependent on the manufacturer and fuse type. However, all types require a voltage for programming that is different from the operating $V_{CC}$. This voltage can be provided by the circuit shown in Figure 4.

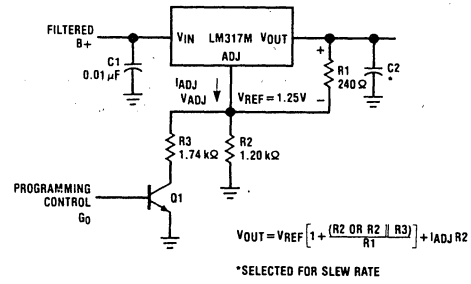



**Figure 4. PROM Programming Voltage Regulator**

The LM317M regulates by maintaining a reference voltage of 1.25V across R1. Therefore, by changing the voltage at the ADJ pin the regulated output voltage can be varied. During normal operating conditions the output voltage is set to 5.0 volts. Q1 is held on by output G0 of the controller and makes $V_{ADJ} = 3.75V$. (Refer to equation in Figure 4.) When the output voltage is to be increased to the required programming voltage, Q1 is turned off and $V_{ADJ}$ increases to 9.25V. The output then increases to 10.5V, the proper programming voltage for National Semiconductor's bipolar Schottky PROMs. The value of C2 is selected to obtain the proper slew rate of the programming voltage transitions.

When a bit is to be programmed, its address is latched into the MM74C373. The PROM is then disabled and the data for the bit is put on the bus. This data word has a "1" in the proper location for the bit to be programmed and "0s" in the other locations. This "1" turns on the driver in the DS8654 for the respective bit. The programming voltage is then applied by making the G0 output of the COP 420L high. This makes $V_{CC}$ and the proper output 10.5 volts. The PROM enable line is then taken low for one instruction cycle time (approx. 16$\mu$s). Then the voltages are restored to normal operating levels and the bit can be verified by enabling the octal buffer after resetting the L lines. If the bit was not programmed, the programming sequence is repeated until the bit is programmed or it is determined that it will not program and is skipped over.

### Speedometer and Odometer Displays

The microcontroller interfaces with the speedometer and odometer displays using National Semiconductor's Microwire™ serial data bus. All display data is sent to the display drivers via the data, clock, and enable lines. This technique allows maximum use of the I/O lines of the microcontroller and also gives great flexibility in choosing the type of display to be used. Table 1 shows a list of National's display drivers that interface by Microwire™.

Table 1.

| Device | Package Size | Type of Driver |
|---|---|---|
| COP 470 | 18-pin | 4-digit × 8-segment MUX VF |
| COP 472 | 20-pin | 3 backplane × 12-segment triplexed LCD |
| *MM54XX | 40-pin | 32-segment direct drive VF |
| MM5450 | 40-pin | 35-segment direct drive LED |
| *MM54XX | 40-pin | 32-segment direct drive LCD |

*Future product.

### Summary

By using a low-cost one-chip microcontroller and bipolar PROM, an automotive electronic odometer can be designed with unique features offering permanent, non-volatile mileage accumulation and protection against tampering.

Section 3

# MOS/ROMs

National Semiconductor



# MM52116 (2316E) 16,384-Bit Read Only Memory

## General Description

The MM52116 is a static MOS 16,384-bit read-only memory organized in an 2048-word-by-8-bit format. It is fabricated using N-channel enhancement and depletion-mode technology which provides complete DTL/TTL compatibility and single power-supply operation.

Three programmable chip selects controlling the TRI-STATE® outputs allow for memory expansion.

Programming of the memory array and chip-select active levels is accomplished by changing two masks during fabrication.

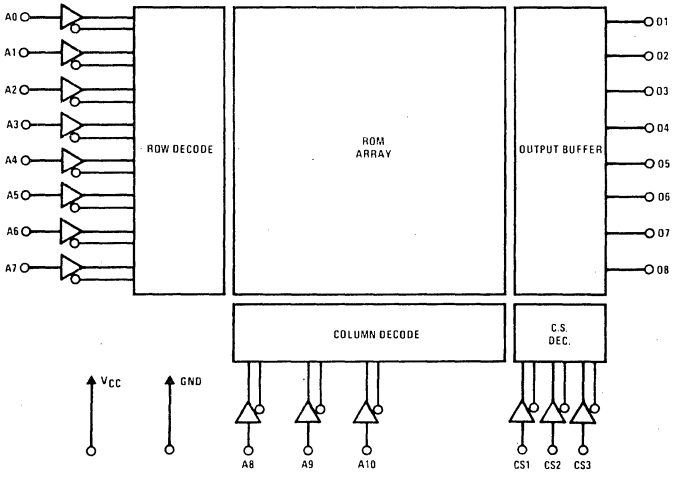
## Features

- Fully decoded
- Single 5V power supply ±10% tolerance
- Inputs and outputs TTL compatible
- Outputs drive 2 TTL loads and 100 pF
- Static operation
- TRI-STATE outputs for bus interface
- Programmable chip selects
- 2048-word-by-8-bit organization
- Maximum access time – 450 ns
- Industry standard pin outs (2316E)
- Compatible to standard EPROMs

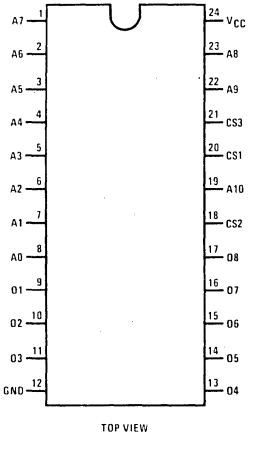
## Applications

- Microprocessor instruction store
- Control logic
- Table look-up

## Block and Connection Diagrams




Dual-In-Line Package




TOP VIEW

**Order Number MM52116D**
**See NS Package D24C**

**Order Number MM52116N**
**See NS Package N24B**

## Absolute Maximum Ratings (Note 1)

| | |
|---|---|
| Voltage at Any Pin | −0.5V to +7.0V |
| Storage Temperature Range | −65°C to +150°C |
| Power Dissipation | 1W |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Operating Conditions

| | |
|---|---|
| Operating Temperature Range | −40°C to +85°C |

## DC Electrical Characteristics

($T_A$ within operating temperature range, $V_{CC}$ = 5V ±10%, unless otherwise specified).

| PARAMETER (Note 2) | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $I_{LI}$ | Input Current | $V_{IN}$ = 0 to $V_{CC}$ | | | 10 | μA |
| $V_{IH}$ | Logical "1" Input Voltage | 0°C | 2.0 | | $V_{CC}$+1.0 | V |
| $V_{IH}$ | Logical "1" Input Voltage | −40°C | 2.2 | | $V_{CC}$+1.0 | V |
| $V_{IL}$ | Logical "0" Input Voltage | | −0.5 | | 0.8 | V |
| $V_{OH}$ | Logical "1" Output Voltage | $I_{OH}$ = −400 μA | 2.4 | | | V |
| $V_{OL}$ | Logical "0" Output Voltage | $I_{OL}$ = 3.2 mA | | | 0.4 | V |
| $I_{LOH}$ | Output Leakage Current | $V_{OUT}$ = $V_{CC}$, Chip Deselected | | | 10 | μA |
| $I_{LOL}$ | Output Leakage Current | $V_{OUT}$ = 0V, Chip Deselected | −10 | | | μA |
| $I_{CC1}$ | Power Supply Current | All Inputs = $V_{CC}$, Data Output Open | | 70 | 100 | mA |

## Capacitance

| PARAMETER (Note 3) | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $C_{IN}$ | Input Capacitance (All Inputs) | $V_{IN}$ = 0V, $T_A$ = 25°C, f = 1 MHz, (Note 2) | | | 7.5 | pF |
| $C_{OUT}$ | Output Capacitance | $V_{OUT}$ = 0V, $T_A$ = 25°C, f = 1 MHz, (Note 2) | | | 15.0 | pF |

## AC Electrical Characteristics

($T_A$ within operating temperature range, $V_{CC}$ = 5V ±10%, unless otherwise specified). See AC test circuit and switching time waveforms.

| PARAMETER | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $t_{AC}$ | Chip Select Access Time | See AC Test Circuit; $t_{AC}$ and $t_A$ Measured to Valid Output Levels with $t_r$ and $t_f$ of Input <20 ns, $t_{OFF}$ Measured to <±20 μA Output Current | | | 120 | ns |
| $t_{OFF}$ | Output Turn OFF Delay | | | | 100 | ns |
| $t_A$ | Address Access Time | | | | 450 | ns |

**Note 1:** "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. Except for "Operating Temperature Range" they are not meant to imply that the devices should be operated at these limits. The table of "Electrical Characteristics" provides conditions for actual device operation.

**Note 2:** Positive true logic notation is used: logical "1" = most positive voltage level, logical "0" = most negative voltage level.

**Note 3:** Capacitance is guaranteed by periodic testing.

**Note 4:** Typical values are for $T_A$ = 25°C and nominal supply voltage.

## Switching Time Waveforms and AC Test Circuit

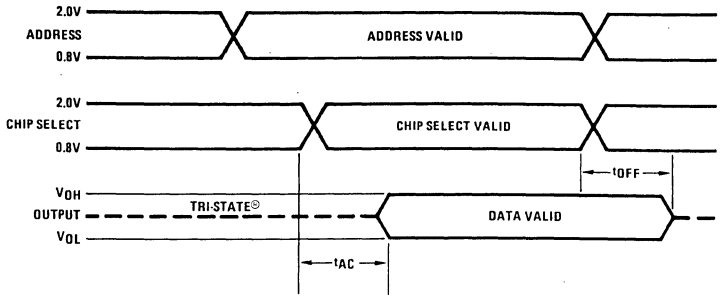



FIGURE 1. Address Precedes Chip Select

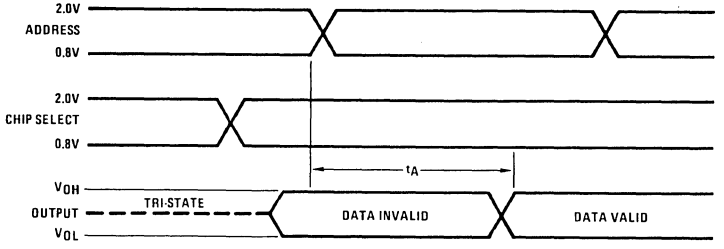
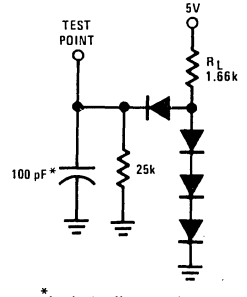



FIGURE 2. Address Follows Chip Select




*Includes jig capacitance

## ROM Programming Information

ROM programs for the MM52116 can be supplied to National in a number of means:

A. 2708 PROM sets

B. 2516 PROM (or equivalent)

C. 2716 PROM (or equivalent)

D. Intellec HEX punched paper tape

E. Binary punched paper tape

Since the MM52116 has programmable chip selects, it is imperative that chip select information be provided along with the ROM program. The information should be supplied as shown:

CS1 is to be programmed logical_____(Hi or Lo)

CS2 is to be programmed logical_____(Hi or Lo)

CS3 is to be programmed logical_____(Hi or Lo)

Given any of the above means of program data is received by National, verification of ROM programs is handled internally via a sophisticated computerized system. The original input device (PROM, tape, etc.) is read, the data is reprocessed to formats required by various production machines, and the final reconstructed data is then compared back to the original input device.

The verification package returned to the customer for approval will consist of a listing of the program and a PROM or tape which matches the data National will use to create the programmed MM52116. In a normal situation, the verification package returned to the customer for approval, because of the system described, may consist of the original PROM or tape submitted by the customer. This program data, now in National's production format, is stored in archives for future customer re-orders.

National Semiconductor



# MM52116FDW, MM52116FDX Character Generators

## General Description

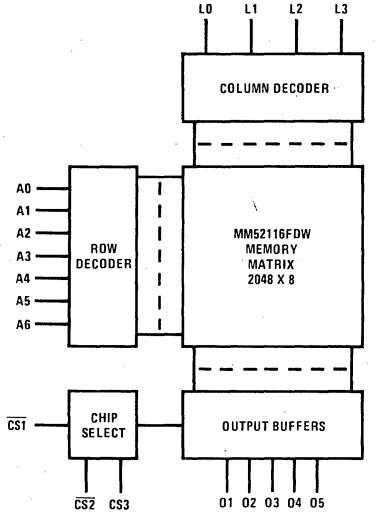
The MM52116FDW, MM52116FDX are 128-character, N-channel, character generators designed primarily for CRT display applications. The MM52116FDW/MM52116 FDX provide 5x7 and 7x9 row scan character fonts, respectively. They provide complete DTL/TTL compatibility with single 5V power supply operation.

## Features

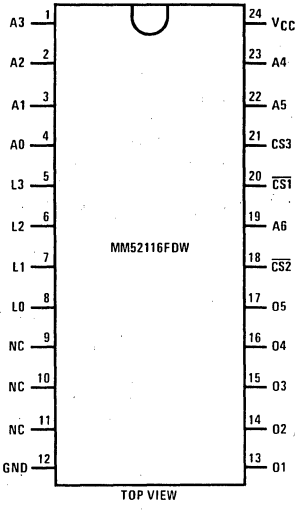
- 128-character row scan
- 5x7 or 7x9 font
- Maximum access time – 450 ns
- TRI-STATE® outputs for bus interface
- Programmable chip selects
- Single 5V power supply
- Inputs and outputs TTL compatible
- MM2316E and MM2716 pin compatible

## Block and Connection Diagrams

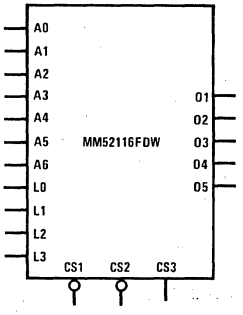
Chip Select Enable Pattern $\overline{CS1}$ = 0, $\overline{CS2}$ = 0, CS3 = 1




Dual-In-Line Package



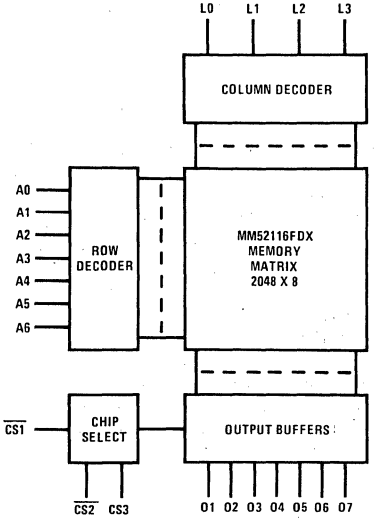

## Logic Symbols



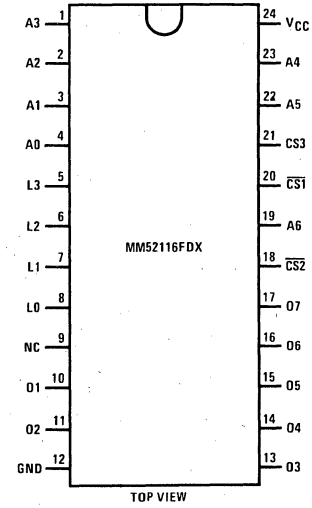

**Order Number MM52116FDW-D or MM52116FDX-D**
**See NS Package D24C**

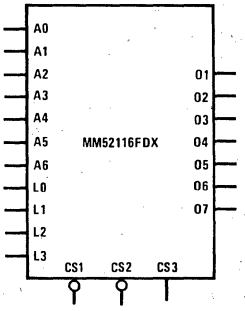
**Order Number MM52116FDW-N or MM52116FDX-N**
**See NS Package N24A**

Chip Select Enable Pattern $\overline{CS1}$ = 1, $\overline{CS2}$ = 0, CS3 = 1




Dual-In-Line Package



## Absolute Maximum Ratings (Note 1)

| | |
|---|---|
| Voltage at Any Pin | −0.5V to +6.5V |
| Operating Temperature Range | −40°C to +85°C |
| Storage Temperature Range | −65°C to +150°C |
| Power Dissipation | 1W |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## DC Electrical Characteristics

($T_A$ within operating temperature range, $V_{CC}$ = 5V ±10%, unless otherwise specified).

| PARAMETER (Note 2) | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $I_{LI}$ | Input Current | $V_{IN}$ = 0 to $V_{CC}$ | | | 10 | μA |
| $V_{IH}$ | Logical "1" Input Voltage | 0°C | 2.0 | | $V_{CC}$+1.0 | V |
| $V_{IH}$ | Logical "1" Input Voltage | −40°C | 2.2 | | $V_{CC}$+1.0 | V |
| $V_{IL}$ | Logical "0" Input Voltage | | −0.5 | | 0.8 | V |
| $V_{OH}$ | Logical "1" Output Voltage | $I_{OH}$ = −400 μA | 2.4 | | | V |
| $V_{OL}$ | Logical "0" Output Voltage | $I_{OL}$ = 3.2 mA | | | 0.4 | V |
| $I_{LOH}$ | Output Leakage Current | $V_{OUT}$ = $V_{CC}$, Chip Deselected | | | 10 | μA |
| $I_{LOL}$ | Output Leakage Current | $V_{OUT}$ = 0V, Chip Deselected | −10 | | | μA |
| $I_{CC1}$ | Power Supply Current | All Inputs = $V_{CC}$, Data Output Open | | 70 | 100 | mA |

## Capacitance

| PARAMETER (Note 3) | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $C_{IN}$ | Input Capacitance (All Inputs) | $V_{IN}$ = 0V, $T_A$ = 25 C, f = 1 MHz, (Note 2) | | | 7.5 | pF |
| $C_{OUT}$ | Output Capacitance | $V_{OUT}$ = 0V, $T_A$ = 25 C, f = 1 MHz, (Note 2) | | | 15.0 | pF |

## AC Electrical Characteristics

($T_A$ within operating temperature range, $V_{CC}$ = 5V ±10%, unless otherwise specified). See AC test circuit and switching time waveforms.

| PARAMETER | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $t_{AC}$ | Chip Select Access Time | See AC Test Circuit; $t_{AC}$ and $t_A$ Measured to Valid Output Levels with $t_r$ and $t_f$ of Input <20 ns; $t_{OFF}$ Measured to <±20 μA Output Current | | | 120 | ns |
| $t_{OFF}$ | Output Turn OFF Delay | | | | 100 | ns |
| $t_A$ | Address Access Time | | | | 450 | ns |

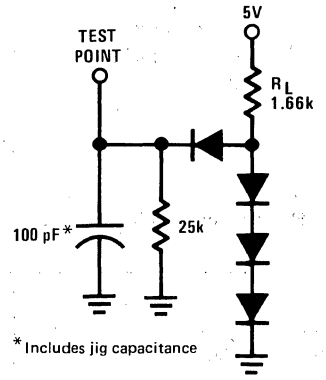
**Note 1:** "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. Except for "Operating Temperature Range" they are not meant to imply that the devices should be operated at these limits. The table of "Electrical Characteristics" provides conditions for actual device operation.

**Note 2:** Positive true logic notation is used: logical "1" = most positive voltage level, logical "0" = most negative voltage level.

**Note 3:** Capacitance is guaranteed by periodic testing.

**Note 4:** Typical values are for $T_A$ = 25°C and nominal supply voltage.

## AC Test Circuit and Switching Time Waveforms



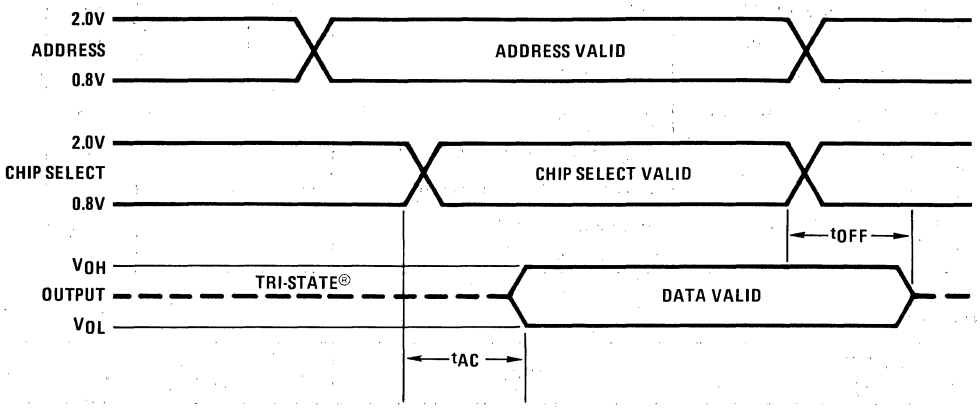




FIGURE 1. Address Precedes Chip Select

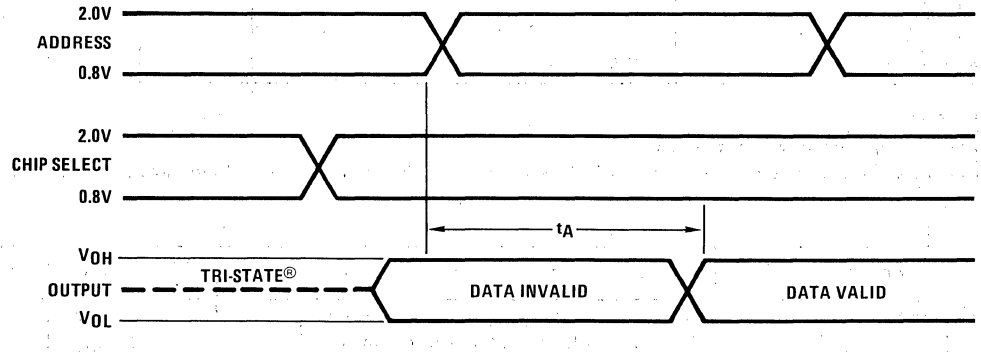


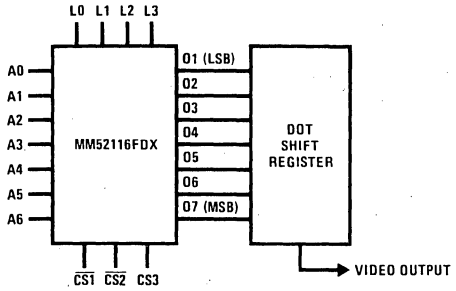

FIGURE 2. Address Follows Chip Select

## Functional Description

The chip is selected by applying the proper logic levels to the 3-chip select pins. A 7-bit binary word must be present at the character address inputs, A0–A6 to select a character. The dot matrix of selected characters is generated by cycling the line count address inputs L0–L3 through the line counts necessary to generate the characters. A dot is generated when an output is a "1" (at $V_{OH}$).

*Figure 3* shows an example of the conditions required at the address and line count pins to generate the dot matrix of the character A. *Figures 5 and 6* show the character fonts of the MM52116FDW and MM52116FDX.

## Functional Description (Continued)




| CHIP SELECT | | | CHARACTER ADDRESS | | | | | | | LINE COUNT | | | | DOT MATRIX | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\overline{CS1}$ | $\overline{CS2}$ | CS3 | A6 | | | A4 | A3 | | | A0 | L3 | L2 | L1 | L0 | O7 | | | | | | | O1 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | | |
| | | | | | | | | | | | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | |
| | | | | | | | | | | | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | |
| | | | | | | | | | | | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | |
| | | | | | | | | | | | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | |
| | | | | | | | | | | | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | |
| | | | | | | | | | | | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | |
| | | | | | | | | | | | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | |
| | | | | | | | | | | | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | |

Note. A "1" = $V_{IH}$ for address, line count and chip select inputs and a "1" = $V_{OH}$ for outputs.

**FIGURE 3. Example of Generating the Character A**

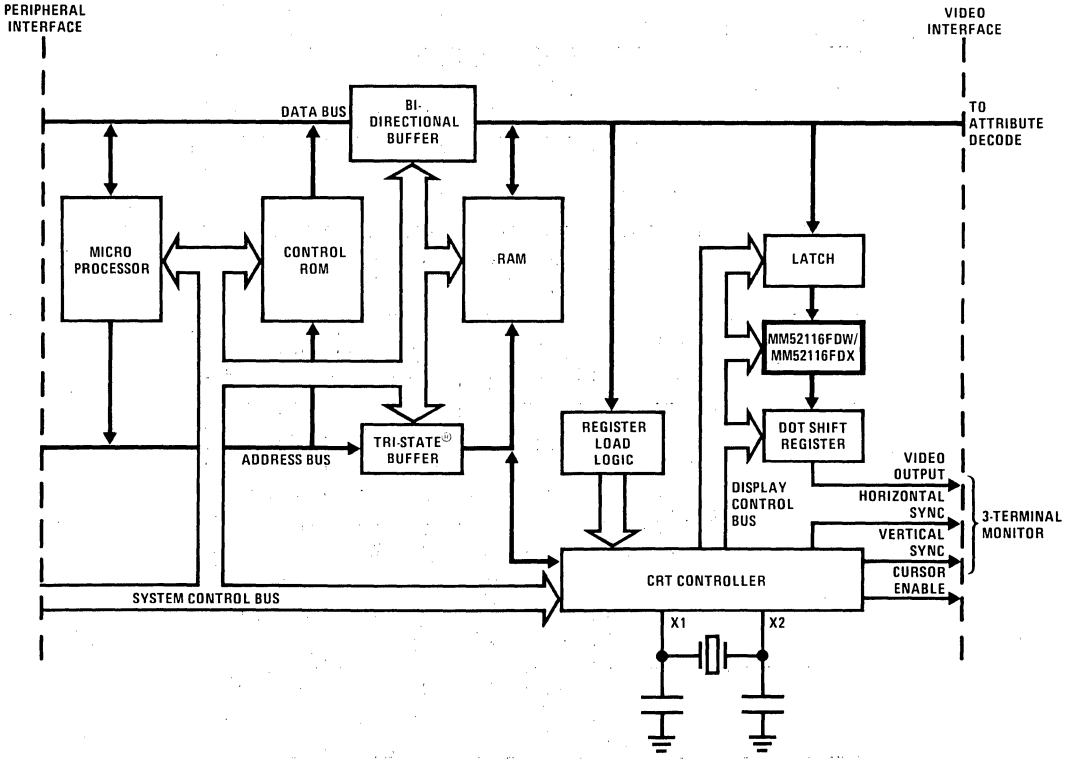



**FIGURE 4. Typical MM52116FDW and MM52116FDX Application**

## Functional Description (Continued)

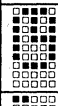
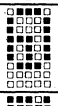
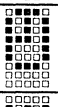
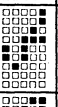
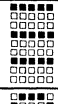
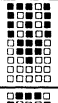
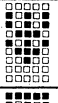
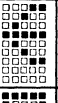
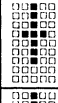
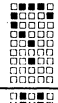
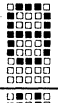
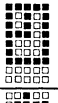
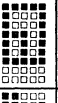
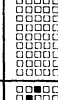
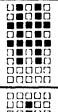
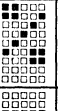
| A6 | A5 A4 A3 \ A2 A1 A0 | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 000 |  |  |  |  |  |  |  |  |
| 0 | 001 |  |  |  |  |  |  |  |  |
| 0 | 010 |  |  |  |  |  |  |  |  |
| 0 | 011 |  |  |  |  |  |  |  |  |
| 0 | 100 |  |  |  |  |  |  |  |  |
| 0 | 101 |  |  |  |  |  |  |  |  |
| 0 | 110 |  |  |  |  |  |  |  |  |
| 0 | 111 |  |  |  |  |  |  |  |  |
| 1 | 000 |  |  |  |  |  |  |  |  |
| 1 | 001 |  |  |  |  |  |  |  |  |
| 1 | 010 |  |  |  |  |  |  |  |  |
| 1 | 011 |  |  |  |  |  |  |  |  |
| 1 | 100 |  |  |  |  |  |  |  |  |
| 1 | 101 |  |  |  |  |  |  |  |  |
| 1 | 110 |  |  |  |  |  |  |  |  |
| 1 | 111 |  |  |  |  |  |  |  |  |

**FIGURE 5. MM52116FDW**

## Functional Description (Continued)

| A6 \ A5 A4 A3 \ A2 A1 A0 | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
|---|---|---|---|---|---|---|---|---|
| 0 | 000 | | | | | | | |
| 0 | 001 | | | | | | | |
| 0 | 010 | | | | | | | |
| 0 | 011 | | | | | | | |
| 0 | 100 | | | | | | | |
| 0 | 101 | | | | | | | |
| 0 | 110 | | | | | | | |
| 0 | 111 | | | | | | | |
| 1 | 000 | | | | | | | |
| 1 | 001 | | | | | | | |
| 1 | 010 | | | | | | | |
| 1 | 011 | | | | | | | |
| 1 | 100 | | | | | | | |
| 1 | 101 | | | | | | | |
| 1 | 110 | | | | | | | |
| 1 | 111 | | | | | | | |

FIGURE 6. MM52116FDX

## Functional Description (Continued)

MM52116FDX ASCII CHARACTER SET IN HEXADECIMAL REPRESENTATION

| Character | 7-Bit Hexadecimal Number | Character | 7-Bit Hexadecimal Number | Character | 7-Bit Hexadecimal Number | Character | 7-Bit Hexadecimal Number |
|---|---|---|---|---|---|---|---|
| NUL | 00 | SP | 20 | @ | 40 | \ | 60 |
| SOH | 01 | ! | 21 | A | 41 | a | 61 |
| STX | 02 | " | 22 | B | 42 | b | 62 |
| ETX | 03 | # | 23 | C | 43 | c | 63 |
| EOT | 04 | $ | 24 | D | 44 | d | 64 |
| ENQ | 05 | % | 25 | E | 45 | e | 65 |
| ACK | 06 | & | 26 | F | 46 | f | 66 |
| BEL | 07 | ' | 27 | G | 47 | g | 67 |
| BS | 08 | ( | 28 | H | 48 | h | 68 |
| HT | 09 | ) | 29 | I | 49 | i | 69 |
| LF | 0A | * | 2A | J | 4A | j | 6A |
| VT | 0B | + | 2B | K | 4B | k | 6B |
| FF | 0C | , | 2C | L | 4C | l | 6C |
| CR | 0D | - | 2D | M | 4D | m | 6D |
| SO | 0E | . | 2E | N | 4E | n | 6E |
| SI | 0F | / | 2F | O | 4F | o | 6F |
| DLE | 10 | 0 | 30 | P | 50 | p | 70 |
| DC1 | 11 | 1 | 31 | Q | 51 | q | 71 |
| DC2 | 12 | 2 | 32 | R | 52 | r | 72 |
| DC3 | 13 | 3 | 33 | S | 53 | s | 73 |
| DC4 | 14 | 4 | 34 | T | 54 | t | 74 |
| NAK | 15 | 5 | 35 | U | 55 | u | 75 |
| SYN | 16 | 6 | 36 | V | 56 | v | 76 |
| ETB | 17 | 7 | 37 | W | 57 | w | 77 |
| CAN | 18 | 8 | 38 | X | 58 | x | 78 |
| EM | 19 | 9 | 39 | Y | 59 | y | 79 |
| SUB | 1A | : | 3A | Z | 5A | z | 7A |
| ESC | 1B | ; | 3B | [ | 5B | | 7B |
| FS | 1C | < | 3C | \ | 5C | | 7C |
| GS | 1D | = | 3D | ] | 5D | ALT | 7D |
| RS | 1E | > | 3E | ↑ | 5E | ESC | 7E |
| US | 1F | ? | 3F | ← | 5F | DEL,RUBOUT | 7F |

National Semiconductor



# MM52132 32,768-Bit (4096 x 8) MAXI-ROM™

## General Description

The MM52132 is a static MOS 32,768-bit read-only memory organized in a 4096-word-by-8-bit format. It is fabricated using N-channel enhancement and depletion-mode technology which provides complete DTL/TTL compatibility and single power-supply operation.

Two programmable chip selects controlling the TRI-STATE® outputs allow for memory expansion.

Programming of the memory array and chip-select active levels is accomplished by changing two masks during fabrication.

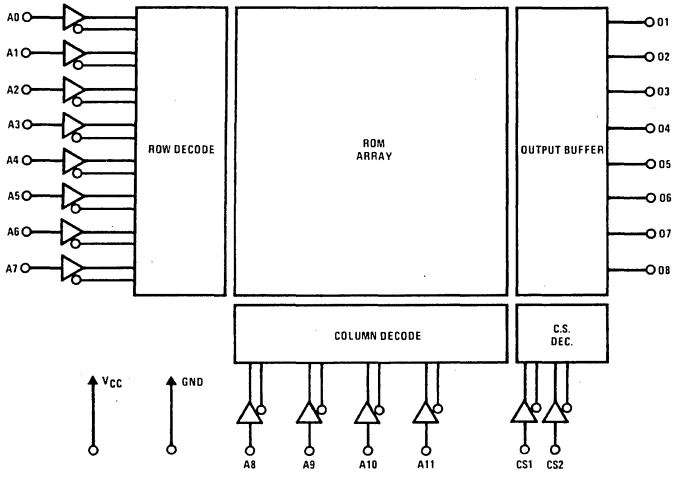
## Features

- Fully decoded
- Single 5V power supply ±10% tolerance
- Inputs and outputs TTL compatible
- Outputs drive 2 TTL loads and 100 pF
- Static operation
- TRI-STATE outputs for bus interface
- Programmable chip selects
- 4096-word-by-8-bit organization
- Maximum access time — 450 ns
- Industry standard pin outs

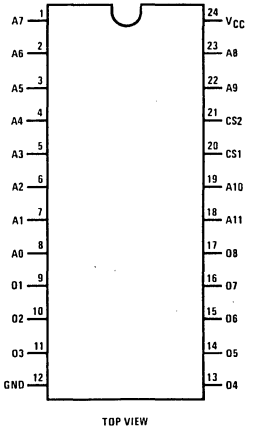
## Applications

- Microprocessor instruction store
- Control logic
- Table look-up

## Block and Connection Diagrams




**Dual-In-Line Package**




**Order Number MM52132D**
**See NS Package D24C**

**Order Number MM52132N**
**See NS Package N24B**

## Absolute Maximum Ratings (Note 1)

| | |
|---|---|
| Voltage at Any Pin | −0.5V to +7.0V |
| Storage Temperature Range | −65°C to +150°C |
| Power Dissipation | 1W |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Operating Conditions

| | |
|---|---|
| Operating Temperature Range | 0°C to +70°C |

## DC Electrical Characteristics

($T_A$ within operating temperature range, $V_{CC}$ = 5V ±10, unless otherwise specified).

| PARAMETER (Note 2) | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $I_{LI}$ | Input Current | $V_{IN}$ = 0 to $V_{CC}$ | | | 10 | μA |
| $V_{IH}$ | Logical "1" Input Voltage | | 2 | | $V_{CC}$+1.0 | V |
| $V_{IL}$ | Logical "0" Input Voltage | | −0.5 | | 0.8 | V |
| $V_{OH}$ | Logical "1" Output Voltage | $I_{OH}$ = −400 μA | 2.4 | | | V |
| $V_{OL}$ | Logical "0" Output Voltage | $I_{OL}$ = 3.2 mA | | | 0.4 | V |
| $I_{LOH}$ | Output Leakage Current | $V_{OUT}$ = $V_{CC}$ Chip Deselected | | | 10 | μA |
| $I_{LOL}$ | Output Leakage Current | $V_{OUT}$ = 0V, Chip Deselected | −10 | | | μA |
| $I_{CC1}$ | Power Supply Current | All Inputs = $V_{CC}$, Data Output Open | | 100 | 130 | mA |

## Capacitance

| PARAMETER (Note 3) | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $C_{IN}$ | Input Capacitance (All Inputs) | $V_{IN}$ = 0V, $T_A$ = 25°C, f = 1 MHz, (Note 2) | | | 7.5 | pF |
| $C_{OUT}$ | Output Capacitance | $V_{OUT}$ = 0V, $T_A$ = 25°C, f = 1 MHz, (Note 2) | | | 15.0 | pF |

## AC Electrical Characteristics

($T_A$ within operating temperature range, $V_{CC}$ = 5V ±10%, unless otherwise specified). See AC test circuit and switching time waveforms.

| PARAMETER | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $t_{AC}$ | Chip Select Access Time | See AC Test Circuit. All Times (Except $t_{OFF}$) | | | 150 | ns |
| $t_{OFF}$ | Output Turn OFF Delay | Measured to 1.5V Level with $t_r$ and $t_f$ of Input < 20 ns, *(Figures 1 and 2)*, $t_{OFF}$ TRI-STATE Output Level Measured to Less than ±20 μA Output Current | | | 150 | ns |
| $t_A$ | Address Access Time | | | | 450 | ns |

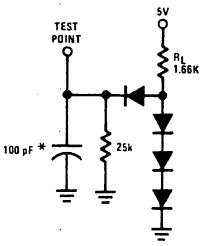
**Note 1:** "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. Except for "Operating Temperature Range" they are not meant to imply that the devices should be operated at these limits. The table of "Electrical Characteristics" provides conditions for actual device operation.

**Note 2:** Positive true logic notation is used: logical "1" = most positive voltage level, logical "0" = most negative voltage level.

**Note 3:** Capacitance is guaranteed by periodic testing.

**Note 4:** Typical values are for $T_A$ = 25°C and nominal supply voltage.

## AC Test Circuit and Switching Time Waveforms




*Includes jig capacitance

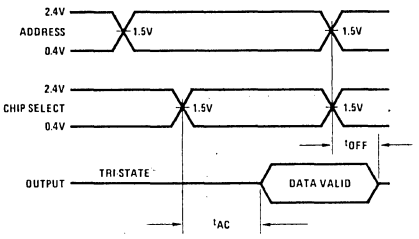



FIGURE 1. Address Precedes Chip Select

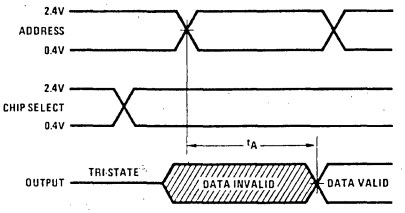



FIGURE 2. Address Follows Chip Select

## ROM Programming Information

ROM programs for the MM52132 can be supplied to National by a number of means.

A. 2708 PROM sets
B. 2516 PROM (or equivalent)
C. 2716 PROM (or equivalent)
D. Intellec HEX punched paper tape
E. Binary punched paper tape

Since the MM52132 has programmable chip selects, it is imperative that chip select information be provided along with the ROM program. The information should be supplied as shown:

CS1 is to be programmed logical _____(Hi or Lo)
CS2 is to be programmed logical _____(Hi or Lo)

Given any of the above means of program data is received by National, verification of ROM programs is handled internally via a sophisticated computerized system. The original input device (PROM, tape, etc.) is read, the data is reprocessed to formats required by various production machines, and the final reconstructed data is then compared back to the original input device.

The verification package returned to the customer for approval will consist of a listing of the program and a PROM or tape which matches the data National will use to create the programmed MM52132. In a normal situation, the verification package returned to the customer for approval, because of the system described, may consist of the original PROM or tape submitted by the customer. This program data, now in National's production format, is stored in archives for future customer re-orders.

National Semiconductor



# MM52164 65,536-Bit (8192 x 8) MAXI-ROM™

## General Description

The MM52164 is a static MOS 65,536-bit read-only memory organized in an 8192-word by 8-bit format. It is fabricated using N-channel enhancement and depletion-mode technology which provides complete DTL/TTL compatibility and single power-supply operation.

One programmable chip select controlling the TRI-STATE® outputs allow for memory expansions.

Programming of the memory array and chip-select active levels is accomplished by changing two masks during fabrication.

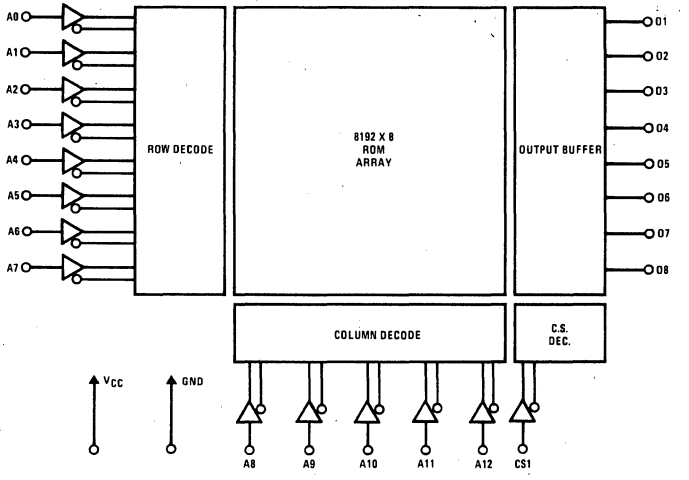
## Features

- Fully decoded
- Single 5V power supply +10%, −5% tolerance
- Inputs and outputs TTL compatible
- Outputs drive 2 TTL loads and 100 pF
- Static operation
- TRI-STATE outputs for bus interface
- Programmable chip select
- 8192-word-by-8-bit organization
- Maximum access time — 450 ns
- Industry standard pin outs

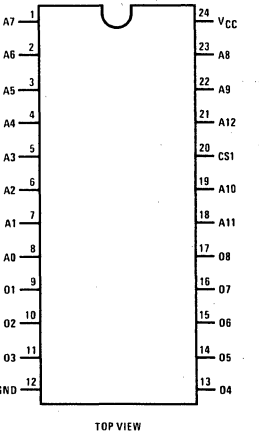
## Applications

- Microprocessor instruction store
- Control logic
- Table look-up

## Block and Connection Diagrams




Dual-In-Line Package




TOP VIEW

**Order Number MM52164D**
**See NS Package D24C**

**Order Number MM52164N**
**See NS Package N24B**

## Absolute Maximum Ratings (Note 1)

| | |
|---|---|
| Voltage at Any Pin | −0.5V to +7.0V |
| Storage Temperature Range | −65°C to +150°C |
| Power Dissipation | 1W |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Operating Conditions

| | |
|---|---|
| Operating Temperature Range | 0°C to +70°C |

## DC Electrical Characteristics

$T_A$ within operating temperature range, $V_{CC}$ = 5V +10%, -5% unless otherwise specified.

| PARAMETER (Note 2) | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $I_{LI}$ | Input Current | $V_{IN}$ = 0 to $V_{CC}$ | | | 10 | μA |
| $V_{IH}$ | Logical "1" Input Voltage | | 2.2 | | $V_{CC}$+1.0 | V |
| $V_{IL}$ | Logical "0" Input Voltage | | −0.5 | | 0.6 | V |
| $V_{OH}$ | Logical "1" Output Voltage | $I_{OH}$ = −400 μA | 2.4 | | | V |
| $V_{OL}$ | Logical "0" Output Voltage | $I_{OL}$ = 3.2 mA | | | 0.4 | V |
| $I_{LOH}$ | Output Leakage Current | $V_{OUT}$ = $V_{CC}$, Chip Deselected | | | 10 | μA |
| $I_{LOL}$ | Output Leakage Current | $V_{OUT}$ = 0V, Chip Deselected | −10 | | | μA |
| $I_{CC1}$ | Power Supply Current | All Inputs = $V_{CC}$ Data Output Open | | 100 | 130 | mA |

## Capacitance

| PARAMETER (Note 3) | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $C_{IN}$ | Input Capacitance (All Inputs) | $V_{IN}$ = 0V, $T_A$ = 25°C, f = 1 MHz, (Note 2) | | | 7.5 | pF |
| $C_{OUT}$ | Output Capacitance | $V_{OUT}$ = 0V, $T_A$ = 25°C, f = 1 MHz, (Note 2) | | | 15.0 | pF |

## AC Electrical Characteristics

$T_A$ within operating temperature range, $V_{CC}$ = 5V +10%, −5% unless otherwise specified. See AC test circuit and switching time waveforms.

| PARAMETER | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $t_{AC}$ | Chip Select Access Time | See AC Test Circuit. All Times (Except $t_{OFF}$) | | | 150 | ns |
| $t_{OFF}$ | Output Turn OFF Delay | Measured to 1.5V Level with $t_r$ and $t_f$ of | | | 150 | ns |
| $t_A$ | Address Access Time | Input < 20 ns, *(Figures 1 and 2)*, $t_{OFF}$ TRI-STATE Output Level Measured to Less than ±20 μA Output Current | | | 450 | ns |

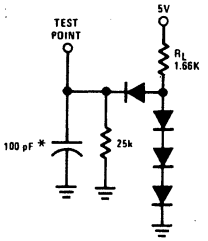
**Note 1:** "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. Except for "Operating Temperature Range" they are not meant to imply that the devices should be operated at these limits. The table of "Electrical Characteristics" provides conditions for actual device operation.

**Note 2:** Positive true logic notation is used: logical "1" = most positive voltage level, logical "0" = most negative voltage level.

**Note 3:** Capacitance is guaranteed by periodic testing.

**Note 4:** Typical values are for $T_A$ = 25°C and nominal supply voltage.

## AC Test Circuit and Switching Time Waveforms




*Includes jig capacitance

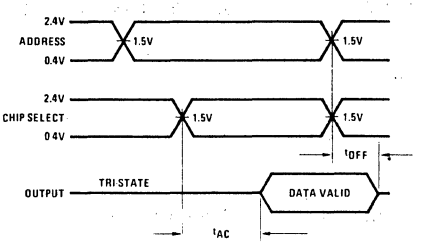



FIGURE 1. Address Precedes Chip Select

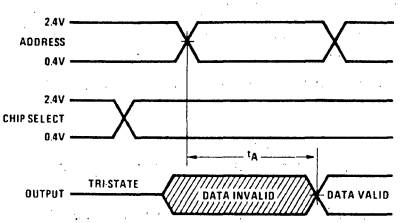



FIGURE 2. Address Follows Chip Select

## ROM Programming Information

ROM programs for the MM52164 can be supplied to National by a number of means.

A. 2708 PROM sets
B. 2516 PROM (or equivalent)
C. 2716 PROM (or equivalent)
D. Intellec HEX punched paper tape
E. Binary punched paper tape

Since the MM52164 has programmable chip selects, it is imperative that chip select information be provided along with the ROM program. The information should be supplied as shown:

CS1 is to be programmed logical _____ (Hi or Lo)

Given any of the above means of program data is received by National, verification of ROM programs is handled internally via a sophisticated computerized system. The original input device (PROM, tape, etc.) is read, the data is reprocessed to formats required by various production machines, and the final reconstructed data is then compared back to the original input device.

The verification package returned to the customer for approval will consist of a listing of the program and a PROM or tape which matches the data National will use to create the programmed MM52164. In a normal situation, the verification package returned to the customer for approval, because of the system described, may consist of the original PROM or tape submitted by the customer. This program data, now in National's production format, is stored in archives for future customer re-orders.

National Semiconductor

MOS/ROMs
PRELIMINARY

# MM52264 MAXI-ROM™ 65,536-Bit Clocked Read Only Memory

## General Description

The MM52264 is a clocked MOS 65,536-bit read-only memory organized in an 8192-word-by-8-bit format. It is fabricated using N-channel enhancement and depletion-mode technology which provides complete DTL/TTL compatibility and single 5V power supply operation.

The MM52264 was designed for those ROM applications requiring fast access time and low power dissipation. Dynamic circuitry has been used extensively to reduce access time. The utilization of a clock input allows the device to be put into a low power standby mode during inactive periods. The device is put into the standby mode by maintaining the clock input $\overline{CE}$ at an input "1" voltage. $\overline{CE}$ must be maintained at a "1" voltage for the minimum specified time (tp) to allow for adequate precharging of the internal dynamic circuitry.

A read operation is initiated and address data are latched by bringing $\overline{CE}$ to an input "0" voltage. The falling-edge of $\overline{CE}$ triggers the generation of a series of internal clock signals which decode addresses into row and column lines and enable output sense amplifiers and buffers. Since the address is latched in address buffers, the input address data can be changed during a read operation after the address-hold-time ($t_{AH}$) specification is met.

Power dissipation increases during a read operation; however, once the output data are latched in the TRI-STATE® output buffers, most of the dynamic circuitry is automatically switched off to conserve power.

The output data remain valid as long as $\overline{CE}$ is maintained at a "0" voltage level. Switching $\overline{CE}$ to a "1" voltage level returns the device to the standby mode and all data outputs to a high-impedance OFF state.

## Features

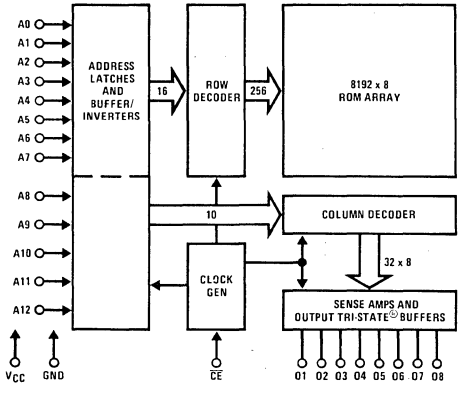
- Fully decoded
- Single 5V power supply ±10% tolerance
- Inputs and outputs TTL compatible
- Outputs drive 2 TTL loads and 100 pF
- Clocked operation
- TRI-STATE outputs for bus interface
- 8192-word-by-8-bit organization
- 300 ns maximum access time
- Industry standard pin outs

## Applications

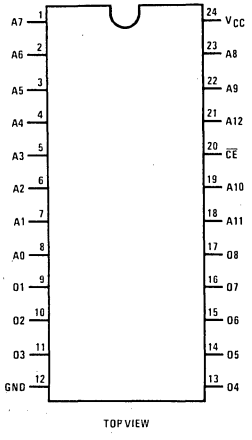
- Microprocessor instruction store
- Control logic
- Table look-up

3

## Block and Connection Diagrams




Dual-In-Line Package




TOP VIEW

## Absolute Maximum Ratings (Note 1)

| | |
|---|---|
| Voltage at Any Pin | −0.5V to +6.5V |
| Storage Temperature Range | −65°C to +150°C |
| Power Dissipation | 1W |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Operating Conditions

| | |
|---|---|
| Operating Temperature Range | 0°C to +70°C |

## DC Electrical Characteristics

($T_A$ within operating temperature range, $V_{CC}$ = 5V ±10%, unless otherwise specified).

| PARAMETER (Note 2) | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $I_{LI}$ | Input Current | $V_{IN}$ = 0 to $V_{CC}$ | | | ±10 | μA |
| $V_{IH}$ | Logical "1" Input Voltage | | 2.0 | | $V_{CC}$+1.0 | V |
| $V_{IL}$ | Logical "0" Input Voltage | | −0.5 | | 0.8 | V |
| $V_{OH}$ | Logical "1" Output Voltage | $I_{OH}$ = −200 μA | 2.4 | | | V |
| $V_{OL}$ | Logical "0" Output Voltage | $I_{OL}$ = 3.2 mA | | | 0.4 | V |
| $I_{LOH}$ | Output Leakage Current | $V_{OUT}$ = 4V, Chip Deselected | | | 10 | μA |
| $I_{LOL}$ | Output Leakage Current | $V_{OUT}$ = 0.45V, Chip Deselected | −10 | | | μA |
| $I_{CC1}$ | Power Supply Standby Current | All Inputs = 5.25V, Data Output Open | | 10 | 15 | mA |
| $I_{CC2}$ | Power Supply Active Current | | | 30 | 50 | mA |

## Capacitance

| PARAMETER (Note 3) | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $C_{IN}$ | Input Capacitance (All Inputs) | $V_{IN}$ = 0V, $T_A$ = 25°C, f = 1 MHz | | | 7.5 | pF |
| $C_{OUT}$ | Output Capacitance | $V_{OUT}$ = 0V, $T_A$ = 25°C, f = 1 MHz | | | 15.0 | pF |

## AC Electrical Characteristics

($T_A$ within operating temperature range, $V_{CC}$ = 5V ±10%, unless otherwise specified). See AC test circuit and switching time waveforms.

| PARAMETER | | CONDITIONS | MIN | TYP (Note 4) | MAX | UNITS |
|---|---|---|---|---|---|---|
| $t_C$ | $\overline{CE}$ Cycle Time | See AC Test Circuit and *Figure 1*. All Times (Except $t_{OFF}$) | 450 | | | ns |
| $t_P$ | $\overline{CE}$ Precharge Time | | 150 | | | ns |
| $t_{CE}$ | $\overline{CE}$ Pulse Width | | 300 | | | ns |
| $t_{AS}$ | Address to $\overline{CE}$ Setup Time | | 0 | | | ns |
| $t_{AH}$ | Address Hold Time from $\overline{CE}$ | | 50 | | | ns |
| $t_{AC}$ | $\overline{CE}$ to Output Access Time | | | | 300 | ns |
| $t_{OFF}$ | Output Turn OFF Delay | AC Test Circuit Load Removed Measured to 1.5V TRI-STATE Level with $t_r$ and $t_f$ of Input < 20 ns | | | 150 | ns |

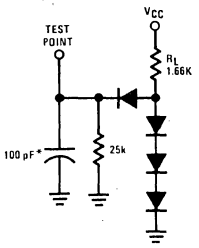
**Note 1:** "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. Except for "Operating Temperature Range" they are not meant to imply that the devices should be operated at these limits. The table of "Electrical Characteristics" provides conditions for actual device operation.

**Note 2:** Positive true logic notation is used: logical "1" = most positive voltage level, logical "0" = most negative voltage level.

**Note 3:** Capacitance is guaranteed by periodic testing.

**Note 4:** Typical values are for $T_A$ = 25°C and nominal supply voltage.

## AC Test Circuit and Switching Time Waveforms




*Includes jig capacitance

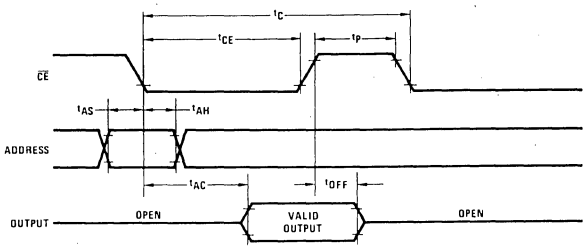



FIGURE 1. AC Electrical Waveforms

## Custom MAXI-ROM Programming

So that National can better serve its customers, we will honor a variety of program format packages. They are, beginning with the most desirable, listed below:

### MM2708 or MM2716 EPROM FORMAT

The MM2708 or MM2716 EPROM is used either singly or in combinations to support larger programs such as:

8 – MM2708 for 1-64k ROM

2 – MM2716 for 1-32k ROM

2 – MM2708 for 1-16k ROM

etc.

**Positive Logic is Preferred:** Positive logic is defined as follows: a logic "1" is the most positive voltage level and a logic "0" is the most negative voltage level. When this definition is applied to the MM2316E and the MM2708, the following definitions result:

A "1" = $V_{IH}$ for addresses and chip selects.

A "1" = $V_{OH}$ for outputs.

A "0" = $V_{IL}$ for addresses and chip selects.

A "0" = $V_{OL}$ for outputs.

This logic definition must be used for all pins on both device types. Any other logic definition (or combination of definitions) will result in delays and additional data processing steps and should be avoided.

**Serialization-Identification (A Must):** Two MM2708 EPROMs are required to store a custom program for a MM2316E ROM. Several custom programs may be included in a single order. The following method of serializing, identifying, and labeling is required to keep everything clearly defined:

a. Each custom program (pattern) is numerically serialized 1, 2, 3, . . .n.

b. The two MM2708s storing a custom program are designated "A" for the first block of 1024 output words and "B" for the second block of 1024 output words. Stated another way: output words corresponding to address 0000 through 1023 are in an MM2708 designated "A;" output words corresponding to addresses 1024 through 2047 are in an MM2708 designated "B."

c. The pair of MM2708s containing a custom program *must be labeled* (stickers,, paint, etc.) with a number corresponding to the program and a letter designating which block of output words it contains. For example, assume three MM2316E custom programs. There would be six MM2708s sent to NSC. They would have labels on them of:

| | | |
|---|---|---|
| 1A | 2A | 3A |
| 1B | 2B | 3B |

### PAPER TAPE FORMAT

The paper tape which should be used is 1" wide paper using 7 or 8-bit ASCII code (such as a Model 33 ASR Teletype produces).

### HEX PAPER TAPE FORMAT

In the Hex Format, a data field contains 8-bit data. *Two* ASCII hexadecimal characters must be used to represent 8-bit data.

Preceding the first data field and following the last data field there must be a leader/trailer length of at least 25 null characters. Comments (except for a colon) may be placed on the tape leader.

The format described below is readily generated by the National Mask Programming System (MPS) or by systems programmed by the user.

**Record Mark Field:** *Frame 0* The ASCII code for a colon (:) is used to signal the start of a record.

**Record Length Field:** *Frames 1 and 2* The number of data bytes in the record is represented by 2 ASCII hexadecimal digits in this field. The high-order digit is in frame 1. The maximum number of data bytes in a record is 255 (FF in hexadecimal). An end-of-file record contains 2 ASCII zeros in this field.

## Custom MAXI-ROM Programming (Continued)

**Load Address Field:** *Frames 3–6* The 4 ASCII hexadecimal digits in frames 3–6 give the address at which the data is loaded. The high-order digit is in frame 3, the lower-order digit in frame 6. The first data byte is stored in the location indicated by the load address; successive bytes are stored in successive memory locations. This field in an end-of-file record contains zeros or the starting address of the program.

**Record Type Field:** *Frames 7 and 8* The 2 ASCII hexadecimal digits in this field specify the record type. The high-order digit is in frame 7. All data records are type 0; end-of-file records are type 1. Other possible values for this field are reserved for future expansion.

**Data Field:** *Frames 9 to 9 + 2 (record length) − 1* A data byte is represented by 2 frames containing the ASCII characters 0–9 or A–F, which represent a hexadecimal value between 0 and FF (0 and 255 decimal). The high-order digit is in the first frame of each pair. If the data is 4-bit, then either the high or low-order digit represents the data and the other digit of the pair may be any ASCII hexadecimal digit. There are no data bytes in an end-of-file record.

**Checksum Field:** *Frames 9 + 2 (record length) to 9 + 2 (record length) + 1* The checksum field contains the ASCII hexadecimal representation of the two's complement of the 8-bit sum of the 8-bit bytes that result from converting each pair of ASCII hexadecimal digits to 1 byte of binary, from the record length field to and including the last byte of the data field. Therefore, the sum of all the ASCII pairs in a record after converting to binary, from the record length field to and including the checksum field, is zero.

**Hex Example:**

```
:10310000311A320E03117E31CD40003A9231B7C2EE
:1031100060310E00117031CD40003A9231B7C2607B
:10312000312A7E31227A310E03117E31CD40003AB0
:103130009231B7C260312A8C317CB5CA50310E044D
:10314000118831CD40003A9231B7C26031C327186
:103150000E01117A31CD40000E09119031CD4000A1
:103160000E0C119231CD40000E09119031CD40006E
:0A3170007E3196310100000092311B
:10317C0092310100963180008C31923100009631F1
:04318E0092319231B7
:02319400923176
:00310001CE
```

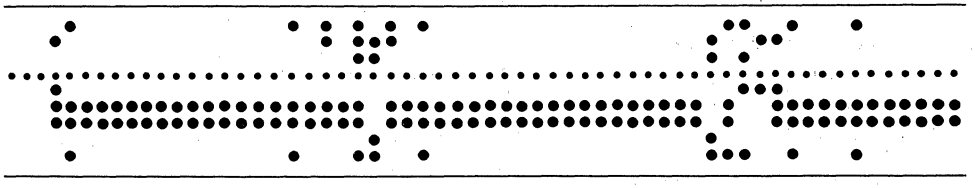
## Hardware Verification

When the custom program is submitted to NSC in a PROM or EPROM, the customer will receive both a verification listing and a duplicate of the original units. The customer can use software (the listing) or the PROMs to verify the program. These PROMs have been programmed and tested with tapes generated by the NSC Mask Programming System (MPS). He will be asked for a Go/No Go response within a week after receipt of the verification package, listing and PROM set.

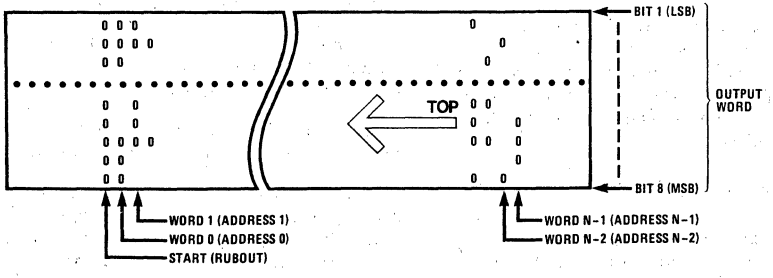
Note that blank 16k bit PROMs for program verification must be supplied by customer, and must be pin compatible to coinciding ROM.

## Tape Format Illustrations

HEX TAPE FORMAT



STANDARD 8-BIT BINARY




Note: Start with a RUBOUT and enter the output words in sequence from word 0 (address 0) to word N − 1 (address N − 1). There must not be *any* other characters on the tape. The order information must indicate whether a punch equals "0" or "1." THIS IS THE LEAST RELIABLE FORMAT. IT HAS NO ERROR CHECKING AND SHOULD BE AVOIDED IF AT ALL POSSIBLE.

Section 4

# Speech Synthesis

National Semiconductor

# DIGITALKER™ Speech Synthesis System



## General Description

The DIGITALKER is a speech synthesis system consisting of multiple N-channel MOS integrated circuits. It contains a speech processor chip (SPC) and speech ROM and when used with external filter, amplifier, and speaker, produces a system which generates high quality speech including the natural inflection and emphasis of the original speech. Male, female, and children's voices can be synthesized.

The SPC communicates with the speech ROM, which contains the compressed speech data as well as the frequency and amplitude data required for speech output. Up to 128k bits of speech data can be directly accessed. This can be expanded with minimal external logic.

With the addition of an external resistor, on-chip debounce is provided for use with a switch interface.

An interrupt is generated at the end of each speech sequence so that several sequences or words can be cascaded to form different speech expressions.

Encoding (digitizing) of custom word or phrase lists must be done by National Semiconductor. Customers submit to the factory high quality recorded magnetic reel to reel tapes containing the words or phrases to be encoded. National Semiconductor will sell kits consisting of the SPC and ROM(s) containing the digitized word or phrases.

## Features

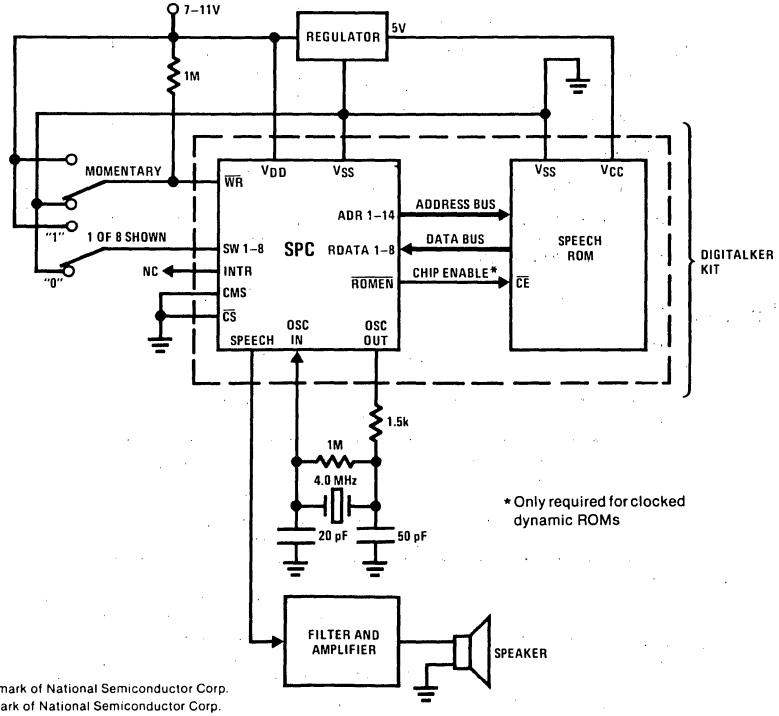
- Designed to be easily interfaced to most popular microprocessors
- 256 possible addressable expressions
- Male, female, and children's voices
- Natural inflection and emphasis of original speech
- Addresses 128k of ROM directly
- Communicates with static or clocked dynamic ROMs
- TTL compatible
- MICROBUS™ compatible
- On-chip switch debounce for interfacing to manual switches independent of a microprocessor
- Easily expandable to greater than 128k ROM
- Interrupt capability for cascading words or phrases
- Crystal controlled or externally driven oscillator
- Ability to store silence durations for timing sequences

## Applications

- Telecommunications
- Appliance
- Automotive
- Teaching aids
- Consumer products
- Clocks
- Language translation
- Annunciators

## Typical Applications

**Minimum Configuration Using Switch Interface**




DIGITALKER is a trademark of National Semiconductor Corp.
MICROBUS is a trademark of National Semiconductor Corp.

## Absolute Maximum Ratings

| | | | |
|---|---|---|---|
| Storage Temperature Range | −65°C to +150°C | Voltage at Any Pin | 12V |
| Operating Temperature Range | 0°C to 70°C | Operating Voltage Range, $V_{DD}$-$V_{SS}$ | 7V to 11V |
| $V_{DD}$-$V_{SS}$ | 12V | Lead Temperature (Soldering, 10 seconds) | 300°C |

## DC Electrical Characteristics $T_A = 0°C$ to 70°C, $V_{DD} = 7V$-11V, $V_{SS} = 0V$, unless otherwise specified.

| Symbol | Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input Low Voltage | | −0.3 | | 0.8 | V |
| $V_{IH}$ | Input High Voltage | | 2.0 | | $V_{DD}$ | V |
| $V_{OL}$ | Output Low Voltage | $I_{OL} = 1.6$ mA | | | 0.4 | V |
| $V_{OH}$ | Output High Voltage | $I_{OH} = -100\ \mu A$ | 2.4 | | 5.0 | V |
| $V_{ILX}$ | Clock Input Low Voltage | | −0.3 | | 0.6 | V |
| $V_{IHX}$ | Clock Input High Voltage | | 4.0 | | $V_{DD}$ | V |
| $I_{DD}$ | Power Supply Current | | | | 50 | mA |
| $I_{IL}$ | Input Leakage | | | | ±10 | μA |
| $I_{ILX}$ | Clock Input Leakage | | | | ±10 | μA |
| $V_S$ | Silence Voltage | | | $0.45 V_{DD}$ | | V |
| $V_{OUT}$ | Peak to Peak Speech Output | $V_{DD} = 11V$ | | 2.0 | | V |

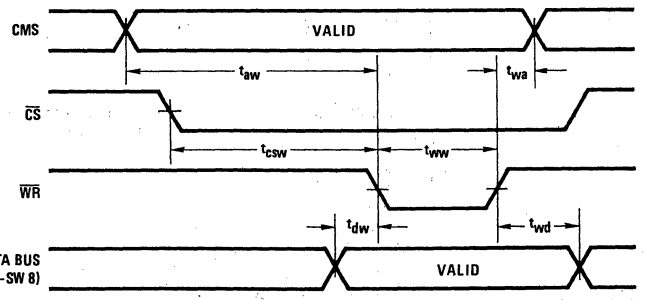
## AC Electrical Characteristics $T_A = 0°C$ to 70°C, $V_{DD} = 7V$-11V, $V_{SS} = 0V$, unless otherwise specified.

| Symbol | Parameter | Min | Max | Units |
|---|---|---|---|---|
| $t_{aw}$ | CMS Valid to Write Strobe | 350 | | ns |
| $t_{csw}$ | Chip Select ON to Write Strobe | 310 | | ns |
| $t_{dw}$ | Data Bus Valid to Write Strobe | 50 | | ns |
| $t_{wa}$ | CMS Hold Time after Write Strobe | 50 | | ns |
| $t_{wd}$ | Data Bus Hold Time after Write Strobe | 100 | | ns |
| $t_{ww}$ | Write Strobe Width (50% Point) | 430 | | ns |
| $t_{red}$ | $\overline{\text{ROMEN}}$ ON to Valid ROM Data | | 2 | μs |
| $t_{wss}$ | Write Strobe to Speech Output Delay | | 410 | μs |
| $f_t$ | External Clock Frequency Tolerance | | ±2 | % |

Note: Rise and fall times (10% to 90%) of MICROBUS signals should be 50 ns maximum.

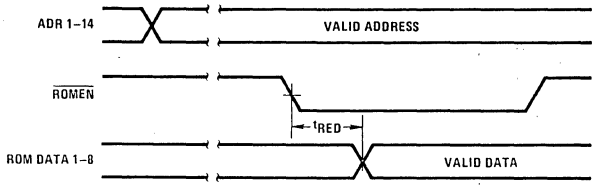
## Timing Waveforms

**Command Sequence**



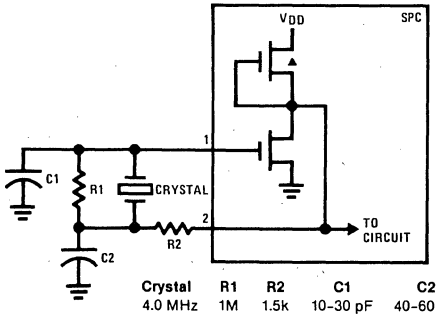
## Timing Waveforms (Continued)

### ROM Data Timing



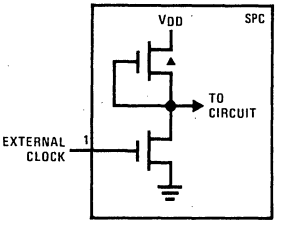

## Crystal Circuit Information

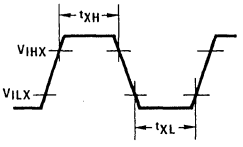
### Typical Crystal Oscillator Network




| Crystal | R1 | R2 | C1 | C2 |
|---|---|---|---|---|
| 4.0 MHz | 1M | 1.5k | 10–30 pF | 40–60 pF |

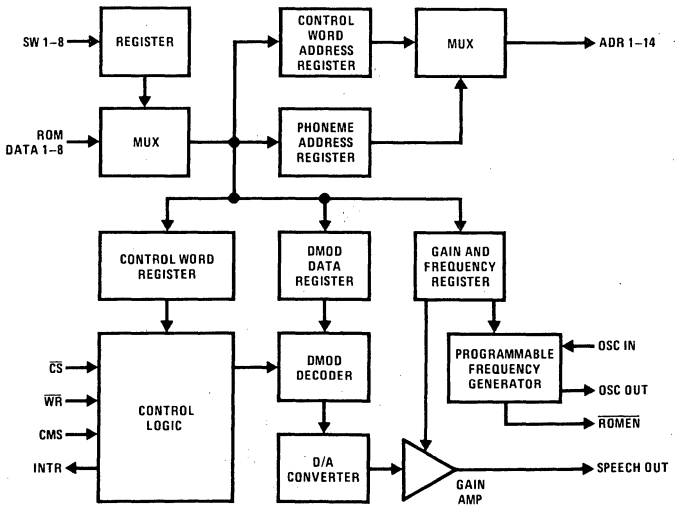
### External Clock Input (4.0 MHz)



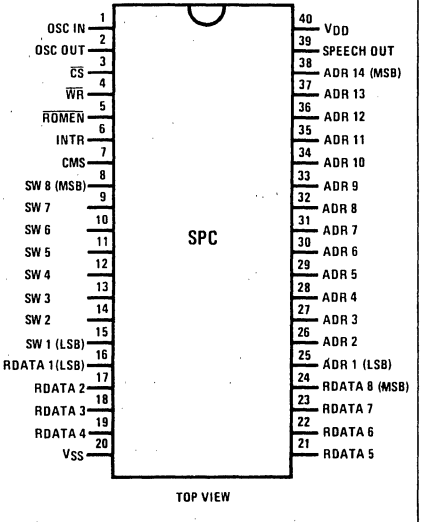




| Timing | Min | Units |
|---|---|---|
| $t_{XH}$ | 100 | ns |
| $t_{XL}$ | 100 | ns |

## Block and Connection Diagrams



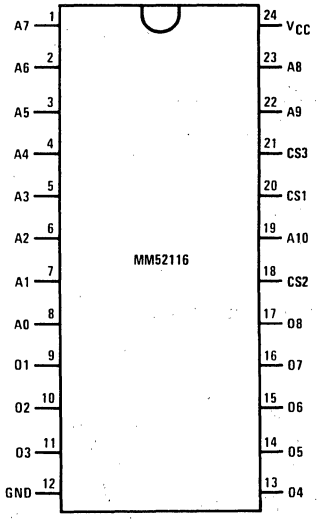

### Dual-In-Line Package




TOP VIEW

## Connection Diagrams (Continued) ($V_{CC}$ = 4.75V–5.25V)

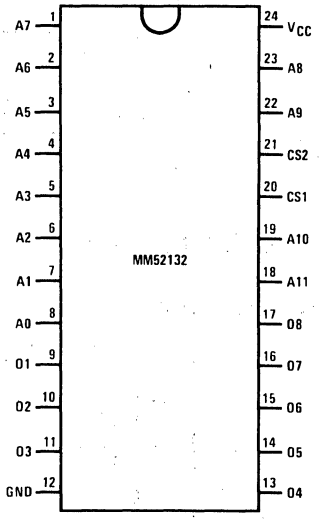
**Dual-In-Line Package**




TOP VIEW

**16k**

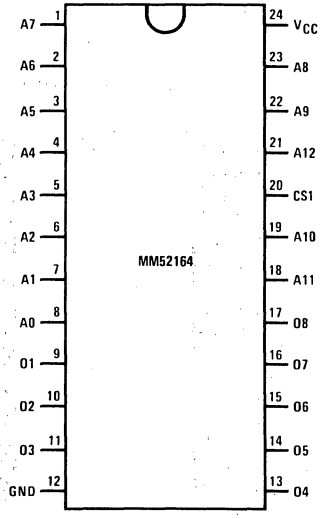
**Dual-In-Line Package**




TOP VIEW

**32k**

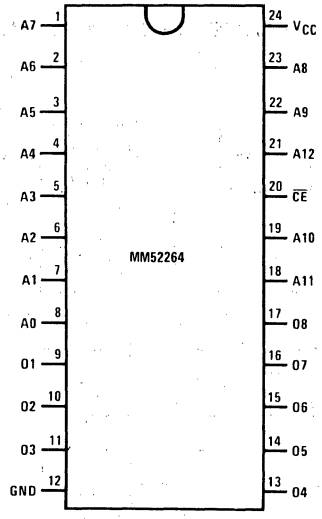
**Dual-In-Line Package**




TOP VIEW

**64k**

**Dual-In-Line Package**




TOP VIEW

**64k "Clocked"**

For specific ROM device information, see MM52116, MM52132, MM52164 or MM52264 data sheets.

## Functional Description

The following describes the function of all SPC input and output pins.

Note: In the following descriptions, a low represents a logic 0 (0.4V nominal), and a high represents a logic 1 (2.4V nominal).

### INPUT SIGNALS

**Chip Select ($\overline{CS}$):** The SPC is selected when $\overline{CS}$ is low. It is only necessary to have $\overline{CS}$ low during a command to the SPC. It is not necessary to hold $\overline{CS}$ low for the duration of the speech data.

**Data Bus (SW 1-8):** This is an 8-bit parallel data bus which contains the starting address of the speech data.

**Command Select (CMS):** This line is used to define the two commands to the SPC.

| CMS | Function |
|---|---|
| 0 | Reset interrupt and start speech sequence |
| 1 | Reset interrupt only |

**Write Strobe ($\overline{WR}$):** This line latches the starting address (SW1-SW8) into a register. On the rising edge of the $\overline{WR}$, the SPC starts execution of the command specified by CMS. The command sequence is shown in the timing waveform section. If a command to start a new speech sequence is issued during a speech sequence, the new speech sequence will be started immediately.

**ROM Data (RDATA 1-8):** This is an 8-bit parallel data bus which contains the speech data from the speech ROM.

### OUTPUT SIGNALS

**Interrupt (INTR):** This signal goes high at the completion of any speech sequence. It is reset by the next valid command. It is also reset at power up.

**ROM Address (ADR1-ADR14):** This is a 14-bit parallel bus that supplies the address of the speech data to the speech ROM.

**ROM Enable ($\overline{ROMEN}$):** This line is for use with clocked dynamic ROMs. When used, the high to low transition must cause the speech ROM to generate a cycle and place the speech data on the RDATA lines. Data must remain on the RDATA lines while $\overline{ROMEN}$ is low. For low power applications, this line can be used to drive a transistor that switches the supply for static speech ROMs. See ROM data timing.

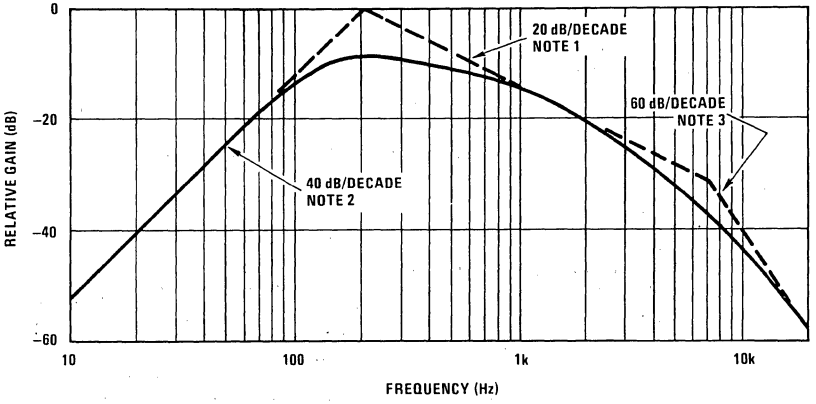
**Speech Output (Speech Out):** This is the analog output that represents the speech data. See frequency response section.

### INPUT/OUTPUT SIGNALS

**Clock Input/Output (OSCIN, OSCOUT):** These two pins connect the main timing reference (crystal) to the SPC.

## Applications Information

**Frequency Response of Combined Amplifier and Speaker**



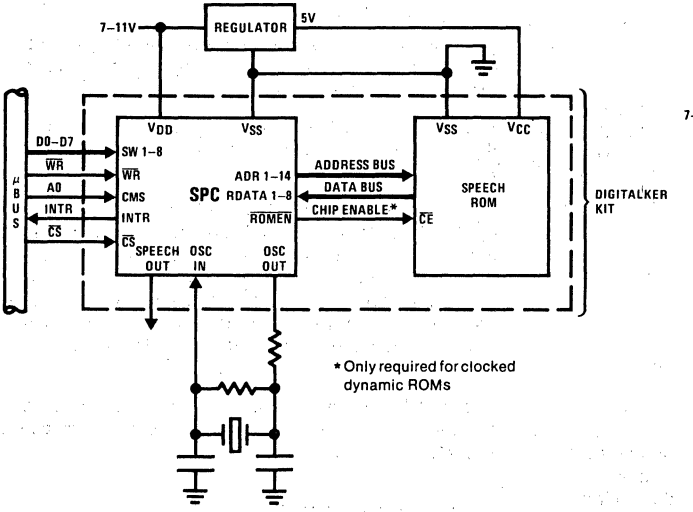

**Note 1:** This curve is the desired response of the entire audio system including speaker. Minimum response is a low pass filter with a cutoff frequency of 200 Hz. For an audio system with a natural cutoff frequency around 200 Hz, this filter can be eliminated. This cutoff frequency may be tuned for the particular voice being synthesized. For a low pitched male voice it may be 100 Hz, while for a high pitched female or child's voice it might be 300 Hz.

**Note 2:** This is optional filtering that can be eliminated by proper selection of the speaker. If this 2 pole response is electronically produced, it should be adjusted as described in Note 1.

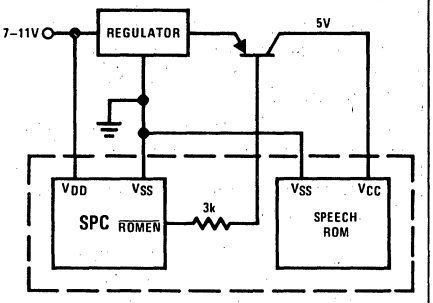
**Note 3:** This is optional filtering that can be eliminated for simpler systems. The acceptable range for this cutoff frequency is 6000 Hz-8000 Hz.

# Typical Applications

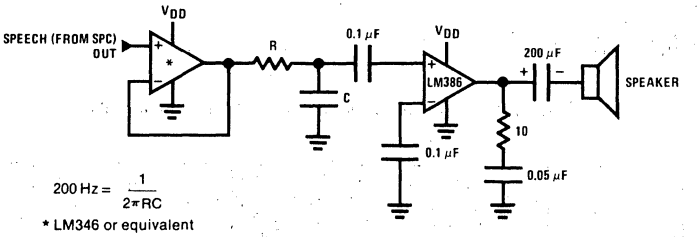
**Complete Applications Schematic for High Quality Voice Reproduction**




**Low Power Configuration Using Static ROM**



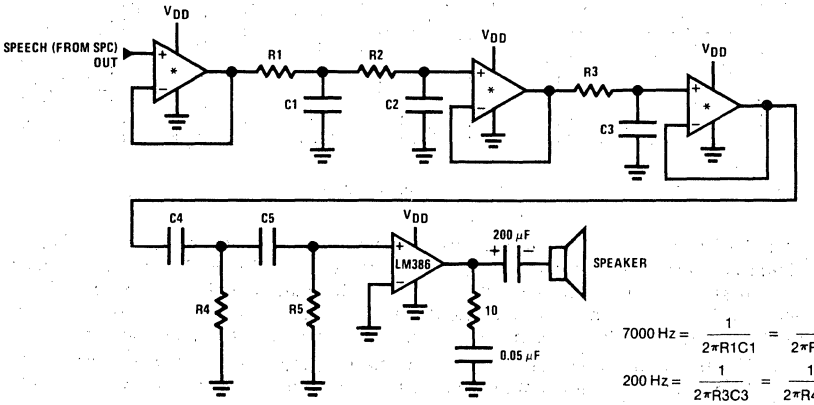

**Minimum Filter Circuit**




$200\ Hz = \frac{1}{2\pi RC}$

* LM346 or equivalent

**Filter Circuit to Produce Maximum Frequency Response**



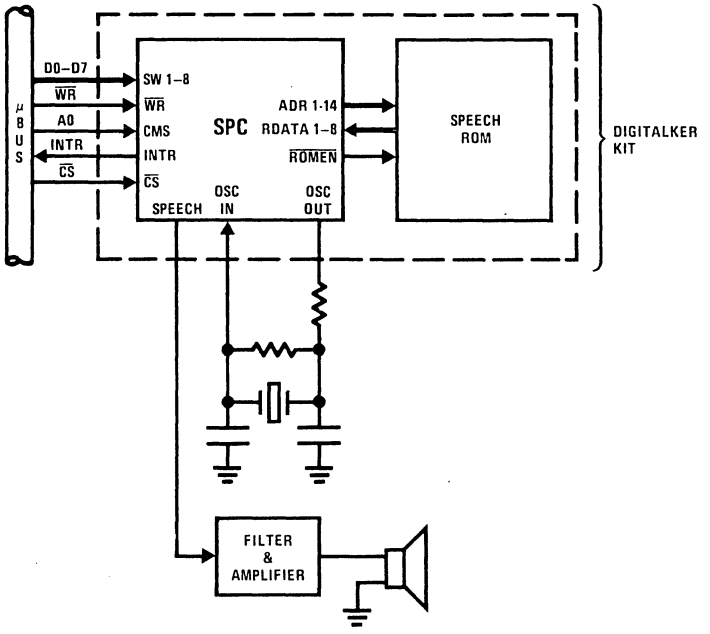

$7000\ Hz = \frac{1}{2\pi R1C1} = \frac{1}{2\pi R2C2}$

$200\ Hz = \frac{1}{2\pi R3C3} = \frac{1}{2\pi R4C4} = \frac{1}{2\pi R5C5}$

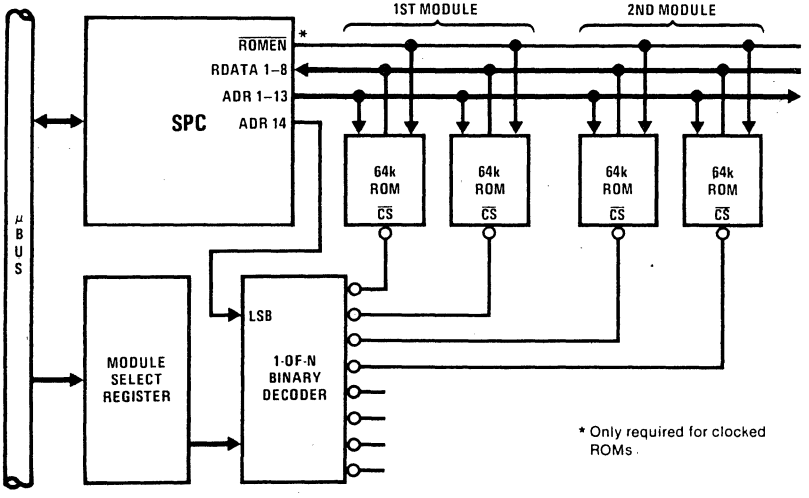
* LM346 or equivalent

## Typical Applications (Continued)

**DIGITALKER System Utilizing MICROBUS™ Interface**




**Speech ROM Expansion for Requirements Greater Than 128k**



Section 5

# Standard MOS/LSI

# Digital Clock Product Selection Guide

| Product Number | Applications | Display Type | Digits | Time Mode | Input Frequency | Output Form | Indicators | | | | | Features | | | | | | | | | Notes |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | Functional | | | | | Electrical | | | | |
| | | | | | | | AM | PM | Colon | Power Fail | Brightness | Alarm | Sleep | Snooze | Fast/Slow Set | Hrs./Mins. Set | Timekeeping | Segment | Digit | Pins | |
| MM5309 | Clock | LED | 4/6 | 12/24 Hrs. | 50/60 Hz | MUX 7-Segment | | | | | | | | | • | | 11-19V | 2 mA | 5 mA | 28 | Reset, Output Enable |
| MM5311 | Clock | LED | 4/6 | 12/24 Hrs. | 50/60 Hz | MUX 7-Segment | | | | | | | | | • | | 11-19V | 2 mA | 5 mA | 28 | Hold, Output Enable |
| MM5312 | Clock | LED | 4 | 12/24 Hrs. | 50/60 Hz | MUX 7-Segment | | | | | | | | | • | | 11-19V | 2 mA | 5 mA | 24 | 1 PPS, Output Enable |
| MM5313 | Clock | LED | 4/6 | 12/24 Hrs. | 50/60 Hz | MUX 7-Segment | | | | | | | | | • | | 11-19V | 2 mA | 5 mA | 28 | Hold, 1 PPS |
| MM5314 | Clock | LED | 4/6 | 12/24 Hrs. | 50/60 Hz | MUX 7-Segment | | | | | | | | | • | | 11-19V | 2 mA | 5 mA | 24 | Hold |
| MM5315 | Clock | LED | 4/6 | 12/24 Hrs. | 50/60 Hz | MUX 7-Segment | | | | | | | | | • | | 11-19V | 2 mA | 5 mA | 28 | Reset Hold |
| MM5316 | Clock Radio | VF | 4 | 12/24 Hrs. | 50/60 Hz | Direct 7-Segment | • | • | • | • | • | • | • | • | • | | 8-29V | 0.5 mA | | 40 | Four Display Modes (Time, Second, Alarm, and Sleep) |
| MM5387AA (MM53108) | Clock Radio | LED | 4 | 12/24 Hrs. | 50/60 Hz | Direct 7-Segment | • | • | • | • | • | • | • | • | • | | 8-26V | 8 mA | | 40 | MM53108 is Mirror Image |
| MM53110AA | Auto Clock | VF | 4 | 12 Hrs. | 2 MHz | MUX 7-Segment | | | • | | • | | | | | • | 5-28V | 2 mA | 8 mA | 22 | Elapse Time (19 Hrs., 59 Mins.) |
| MM53110AB | Auto Clock | VF | 4 | 12 Hrs. | 2 MHz | MUX 7-Segment | | | • | | • | | | | | • | 5-28V | 2 mA | 8 mA | 22 | Elapse Time (19 Mins., 59 Secs.) |
| MM53113 | Clock Radio | VF | 4 | 12/24 Hrs. | 50/60 Hz | Direct 7-Segment | • | • | • | • | • | • | • | • | • | | 8-29V | 0.5 mA | | 40 | Pin-Out Same as MM5316 |
| MM53124 | Auto Clock | VF | 4 | 12/24 Hrs. | 4 MHz | Direct 7-Segment | | | • | | • | | | | | • | 5.25V | 0.4 mA | | 40 | Elapse Time (24 Hrs.) Min./Secs. to Hrs./Mins. |
| MM53224 | Auto Clock | VF | 4 | 12/24 Hrs. | 4 MHz | Direct 7-Segment | | | • | | • | | | | | • | 5.25V | 0.4 mA | | 40 | Elapse Time (24 Hrs.) Min./Secs. to Hrs./Mins. |
| MM5402 (MM5405) | Clock Radio | LED | 4 | 12/24 Hrs. | 50/60 Hz | Direct 7-Segment | • | • | • | • | • | • | • | • | • | | 7-11V | 10 mA | | 40 | MM5405 is Mirror Image |
| MM5406 | Temperature Clock Radio | LED | 4 | 12/24 Hrs. | 50/60 Hz | Duplex 7-Segment | | • | • | • | • | • | • | • | • | • | 9-11V | 20 mA | | 40 | Duplex Display Up/Down Setting |
| MM5407 | Digital Thermometer | LED | 4 | | | | | | | | | | | | | | 9-11V | | | 14 | Interface to MM5406 for Time/Temperature Clock |
| MM5455 | Clock Radio | LED | 4 | 12/24 Hrs. | 50/60 Hz | Duplex 7-Segment | | | | • | | • | • | • | • | | 7-11V | 20 mA | | 24 | Duplex Display, Alarm Tone |
| MM5456 | Clock Radio | LED | 4 | 12 Hrs. | 50/60 Hz | Duplex 7-Segment | | | | • | | • | • | • | • | | 7-11V | 20 mA | | 22 | Duplex Display, Alarm Tone |
| MM5457 | Clock Radio | LED | 4 | 12 Hrs. | 50 Hz | Duplex 7-Segment | | | | • | | • | • | • | • | | 7-11V | 20 mA | | 22 | Duplex Display, Alarm Tone |
| MM58143 (MM58183) | Clock Radio | LCD | 4 | 12/24 Hrs. | 32.8 kHz | Direct 7-Segment | | • | • | | | • | • | • | • | | 1.5V | 5 μA | | 40 | Audio/Static Alarm Tone, Available in die form MM58183 is Mirror Image |
| MM58144 (MM58184) | Travel Alarm Clock | LCD | 4 | 12/24 Hrs. | 32.8 kHz | Direct 7-Segment | | • | • | | | • | | • | • | | 1.5V | 5 μA | | 40 | Voltage Multiplier Audio/Static Alarm, MM58184 is Mirror Image |
| MM7317B | Calendar Clock Radio | LED | 4 | 12/24 Hrs. | 50/60 Hz | Direct 7-Segment | • | • | • | • | • | • | • | • | • | | 8-26V | 8 mA | | 40 | Calendar (Month-Date) |
| MM7318B | Calendar Clock Radio | LED | 4 | 12/24 Hrs. | 50/60 Hz | Direct 7-Segment | • | • | • | • | • | • | • | • | • | | 8-26V | 8 mA | | 40 | Calendar (Month-Date) |

# Television/Radio Product Selection Guide

| Product Number | Description | Process | Package Pins | Package Type | Notes |
|---|---|---|---|---|---|
| MM5321 | TV Camera Sync Generator | PMOS | 16 | N/D | Horizontal/Vertical Control, Field Indexing and Color Burst Sync |
| MM5322 | Color Bar Generator | PMOS | 16 | N/D | 16 Patterns, 3.58 MHz Crystal Control |
| MM53100 | Programmable TV Timer | CMOS | 24 | N | 4/6 Digit, 24 Hr., 50/60 Hz Programmable TV "ON" Time, +18V Reference Voltage |
| MM53105 | Programmable TV Timer | CMOS | 24 | N | 4/6 Digit, 24 Hr., 50/60 Hz Progammable TV "ON" Time, 0V Reference Voltage |
| MM53118AA | TV Digital Tuner | NMOS | 28 | N | 117-Channel, 3 Band + Cable TV, Up/Down or Keyboard Entry, Complete PLL Frequency Synthesizer |
| MM5430 | AM/FM Radio Frequency Display | NMOS | 40 | N | AM/FM Frequency Display, 4.19 MHz Crystal or 50/60 Line Operation, Programmable IF Offset, Direct Interface to LED Display |
| MM5431 | AM/FM Radio Frequency Display | NMOS | 40 | N | Mirror Image Pin-Out of the MM5430 |
| MM5439 | Microprocessor-Compatible PLL | NMOS | 40 | N | 6 Potentiometer Outputs, 6 General Purpose Outputs, 7 General Purpose Input/Outputs |
| MM55108 | PLL Frequency Synthesizer | CMOS | 18 | N | Programmable $2^9$-1 Division, 10.24 MHz Crystal provides 5 kHz Reference Frequency |
| MM55110 | PLL Frequency Synthesizer | CMOS | 24 | N | Programmable $2^9$-1 or $2^{10}$-1 Division, 10.24 MHz Crystal provides 5 kHz or 10 kHz Reference Frequency |
| MM55121 | PLL Frequency Synthesizer | CMOS | 16 | N | Programmable $2^{13}$-1 Division, 10.24 MHz Crystal provides 5 kHz Reference Frequency, 320 kHz/300 Hz/60 Hz Buffered Outputs |
| MM55122 | PLL Frequency Synthesizer | CMOS | 18 | N | Programmable $2^9$-1 Division, 10.24 MHz Crystal provides 10 kHz Reference, Serial Data, 3 D to A Outputs |
| MM55123 | PLL Frequency Synthesizer | CMOS | 16 | N | Programmable $2^{13}$-1 Division, 10.24 MHz Crystal provides 1 kHz Reference Frequency, 320 kHz/300 Hz/60 Hz Buffered Outputs |
| MM55124 | PLL Frequency Synthesizer | CMOS | 16 | N | Programmable $2^8$-1 Division, 5 kHz or 10 kHz Reference Frequency |
| MM55126 | PLL Frequency Synthesizer | CMOS | 18 | N | Programmable $2^9$-1 Division, 5 kHz or 10 kHz Reference Frequency |
| MM5837 | Digital Noise Source | PMOS | 8 | N | White Noise Source |
| MM5840 | TV Channel Number/Time Display | CMOS | 28 | N | 16-Channel, 5/8 Digit, 12/24 Hr. Time with Interface to MM53100 or MM53105 |
| MM58106 | Digital Clock/TV Channel Display | CMOS | 28 | N | 5/8 Digit, Channel (2-83) or Program (1-16) Display, 12/24 Hr., 50/60 Hz |
| MM58142 | TV Digital Tuner | NMOS | 24 | N | 82-Channel PLL Frequency Synthesizer, Up/Down or Keyboard Entry, Inteface to the MM58146 |
| MM58146 | TV Channel/Time Display | NMOS | 22 | N | 4-Digit, Channel (2-83) Display, 12 Hr. Time, Interface to the MM58142 |
| MM58313 | TV Varactor Tuner Display | CMOS | 20 | N | PAL/NTSC Option, Channel Number Display, Screen Tuning Bar Graph Scale. |

# Games/Calculators Product Selection Guide

| Product Number | Description | Process | Package Pins | Package Type | Notes |
|---|---|---|---|---|---|
| MM5780 | Educational Arithmetic Game | PMOS | 24 | N | 8-Digit, Four-Function (+, −, ×, ÷) Algebraic, "Right"/"Wrong" Indicators |
| MM57455 | Advanced Educational Arithmetic Game | NMOS | 28 | N | Four Function (+, −, ×, ÷) Algebraic, Table/Complex/Amateur-Pro Modes Interface to Special Format Display (NSA1481) |
| MM57459 | LED Calculator with Memory | NMOS | 24 | N | 8-Digit, 5-Function (+, −, ×, ÷, %), Four Function Memory (M+, M−, MR, MC), Auto Constant, Direct LED Display Interface |

# Telecommunication Device Product Selection Guide

| Product Number | Description | Process | Package Pins | Package Type | Notes |
|---|---|---|---|---|---|
| MM5393 | Push Button Pulse Dialer | CMOS | 18 | J | 60/40 Break/Make Pulse Ratio, RC Oscillator, Radial of Last Number, 600 Hz Pacifier Tone |
| MM5394 | Push Button Pulse Dialer | CMOS | 16 | J | Same as MM5393 without Pacifier Tone |
| MM5395 | DTMF (Touch Tone® ) Generator | CMOS | 18 | N | Eight (8) Audio Output Frequencies from a 3.58 MHz Crystal, Operation from a 2-of-8 Keypad |
| MM53125 | DTMF (Touch Tone® ) Generator | CMOS | 18 | N | Eight (8) Audio Output Frequencies from a 3.58 MHz Crystal, Operation from a Single-Contact Keypad |
| MM53130 | DTMF (Touch Tone® ) Generator | CMOS | 18 | N | Eight (8) Audio Output Frequencies from a 3.58 MHz Crystal, BCD/Binary or 2-of-8 Interface Options, Tone Disable, Single/ Dual Tone Mode Select |
| MM53143 | Push Button Pulse Dialer | CMOS | 18 | J | 2:1 Break/Make Pulse Ratio, RC Oscillator, Redial of Last Number, 600 Hz Pacifier Tone |
| MM53144 | Push Button Pulse Dialer | CMOS | 18 | J | 2:1 Break/Make Pulse Ratio, RC Oscillator, Redial of Last Number |
| MM53190 | Push Button Pulse Dialer | CMOS | 20 | N | Selectable Outpulsing Rate, Interdigit Pause and Break/Make Ratio, Redial, Pacifier Tone, 2-of-7 or Single Contact Keyboard |

# Display Driver Product Selection Guide

| Product Number | Description | Process | Package Pins | Package Type | Notes |
|---|---|---|---|---|---|
| MM5445 | V.F. Display Driver | PMOS | 40 | N | 33-Segment Direct Drive, Serial Data Input, Brightness Control, Data Enable |
| MM5446 | V.F. Display Driver | PMOS | 40 | N | 34-Segment Direct Drive, Serial Data Input, Data Enable |
| MM5447 | V.F. Display Driver | PMOS | 40 | N | 34-Segment Direct Drive, Serial Data Input, Brightness Control |
| MM5448 | V.F. Display Driver | PMOS | 40 | N | 35-Segment Direct Drive, Serial Data Input |
| MM5450 | LED Display Driver | NMOS | 40 | D,N | 34-Segment Direct Drive, Serial Data Input |
| MM5451 | LED Display Driver | NMOS | 40 | D,N | 35-Segment Direct Drive, Serial Data Input |
| MM5452 | LCD Display Driver | CMOS | 40 | N | 32-Segment Direct Drive, Serial Data Input, Data Enable |
| MM5453 | LCD Display Driver | CMOS | 40 | N | 33-Segment Direct Drive, Serial Data Input |
| MM5480 | LED Display Driver | NMOS | 28 | D,N | 23-Segment Direct Drive, Serial Data Input, Brightness Control |
| MM5481 | LED Display Driver | NMOS | 20 | D,N | 14-Segment Direct Drive, Serial Data Input, Brightness Control |
| MM58201 | LCD Matrix Display Driver | CMOS | 40 | N | Multiplexed Drive, 8 Backplanes, 24 Segments, 192-Bit RAM, Cascadable, R/C Oscillator |

# Oscillator/Divider Product Selection Guide

| Product Number | Description | Process | Package Pins | Package Type | Notes |
|---|---|---|---|---|---|
| MM5368 | Oscillator/Divider | CMOS | 8 | N | 32 kHz to 50 Hz or 60 Hz with 1 and 10 Hz Buffered Outputs |
| MM5369AA | Oscillator/Divider | CMOS | 8 | N | 3.58 MHz to 60 Hz |
| MM5369EST | Oscillator/Divider | CMOS | 8 | N | 3.58 MHz to 100 Hz |
| MM5369EYR | Oscillator/Divider | CMOS | 8 | N | 3.58 MHz to 50 Hz |
| MM53107AA | Oscillator/Divider | CMOS | 8 | N | 2.097152 MHz to 60 Hz |
| MM53107FDU | Oscillator/Divider | CMOS | 8 | N | 2.097152 MHz to 100 Hz |

# Electronic Data Processing Product Selection Guide

| Product Number | Description | Process | Package Pins | Package Type | Notes |
|---|---|---|---|---|---|
| MM5034 | Shift Register | NMOS | 22 | D,N | Octal 80-Bit Shift Register with Tri-State Outputs and Recirculate |
| MM5035 | Shift Register | NMOS | 20 | D,N | Octal 80-Bit Shift Register with Recirculate |
| MM5303 | UART | PMOS | 40 | D,N | Universal Asynchronous Receiver/Transmitter |
| MM5307 | Baud Rate Generator | PMOS | 14 | D,N | See Data Sheet for Specific Baud Rate Output Frequencies |
| MM5330 | A to D Converter | PMOS | 16 | D,N | 4½-Digit Panel Meter Block, also called ADB4500PCN |
| MM53200 | Digital Code Transmitter/Receiver | NMOS | 18 | N | Single-Chip contains both Encoder and Decoder, 4 valid words for a Receive |
| MM54240 | Asynchronous Transmitter/Receiver | NMOS | 24 | N | Single-Wire, 128-Location Bi-Directional Data Transmission, Master/Slave Configuration, 8-Bit Data |
| MM57109 | Number-Oriented Processor | PMOS | 28 | N | Microprocessor-Compatible Scientific Calculator, Controller, Memory Device |
| MM57436 | Up/Down Counter | NMOS | 24 | N | Decimal or Binary Count, Up/Down, 4- or 8-Digit (16- or 32-Bit) Counter Length |
| MM57499 | Serial Keyboard Interface | NMOS | 28 | N | 128/144 Keys, Serial Transmit/Receive |
| MM5863 | A to D Converter | PMOS | 28 | D,N | 12-Bit Binary A/D Block, also called ADB1200PCN |
| MM5865 | Universal Timer | PMOS | 40 | N | 7 Programmable Functions |
| MM58167 | Microprocessor Real Time Clock | CMOS | 24 | N | Addressable Timekeeping from $1/1000$ sec. to hrs., Day, Date, and Month, with corresponding Latches for Alarm-Type Functions, Power Down Mode, 2 Interrrupt Outputs, 32 kHz Crystal Controlled Oscillator |
| MM58174 | Microprocessor Real Time Clock | CMOS | 16 | N | Addressable Timekeeping from $1/10$ sec., to hrs., Day, Date, and Month, Selectable Interrupt Output, 32 kHz Crystal-Controlled Oscillator |

# Digital Clocks

National Semiconductor

# MM5309, MM5311, MM5312, MM5313, MM5314, MM5315 Digital Clocks

## General Description

These digital clocks are monolithic MOS integrated circuits utilizing P-channel low-threshold, enhancement mode and ion implanted. depletion mode devices. The devices provide all the logic required to build several types of clocks. Two display modes (4 or 6-digits) facilitate end-product designs of varied sophistication. The circuits interface to LED and gas discharge displays with minimal additional components, and require only a single power supply. The timekeeping function operates from either a 50 or 60 Hz input, and the display format may be either 12 hours (with leading-zero blanking) or 24 hours. Outputs consist of multiplexed display drives ($\overline{BCD}$ and 7-segment) and digit enables. The devices operate over a power supply range of 11V to 19V and do not require a regulated supply. These clocks are packaged in dual-in-line packages.

## Features

- 50 or 60 Hz operation
- 12 or 24-hour display format
- Leading-zero blanking (12-hour format)
- 7-segment outputs
- Single power supply
- Fast and slow set controls
- Internal multiplex oscillator
- For features of individual clocks, see Table I

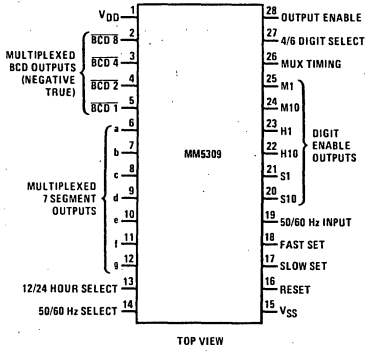
## Applications

- Desk clocks
- Automobile clocks
- Industrial clocks
- Interval Timers

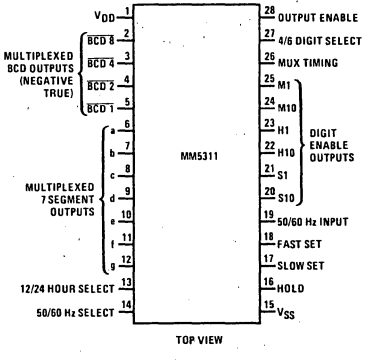
TABLE I

| FEATURES | MM5309 | MM5311 | MM5312 | MM5313 | MM5314 | MM5315 |
|---|---|---|---|---|---|---|
| $\overline{BCD}$ Outputs | X | X | X | X | | X |
| 4/6-Digit Display Mode | X | X | | X | X | X |
| Hold Count Control | | X | | X | X | X |
| 1 Hz Output | | | X | X | | |
| Output Enable Control | X | X | | | X | |
| Reset | X | | | | | X |

## Connection Diagrams (Dual-In-Line Packages)




**Order Number MM5309M**
**See Package 23**




**Order Number MM5311N**
**See Package 23**

## Absolute Maximum Ratings

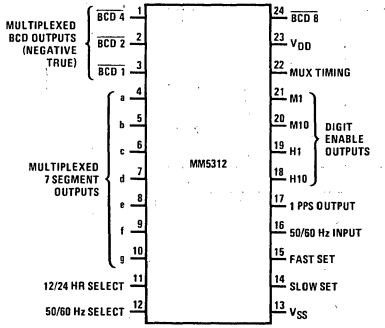
| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ + 0.3 to $V_{SS}$ −20V |
| Operating Temperature | −25°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

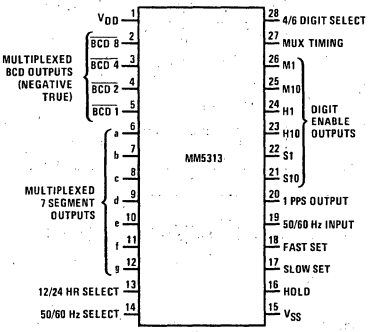
$T_A$ within operating range, $V_{SS}$ = 11V to 19V, $V_{DD}$ = 0V, unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage | $V_{SS}$ ($V_{DD}$ = 0V) | 11 | | 19 | V |
| Power Supply Current | $V_{SS}$ = 14V, (No Output Loads) | | | 10 | mA |
| 50/60 Hz Input Frequency | | dc | 50 or 60 | 60k | Hz |
| 50/60 Hz Input Voltage | | | | | |
| Logical High Level | | $V_{SS}$−1 | $V_{SS}$ | $V_{SS}$ | V |
| Logical Low Level | | $V_{DD}$ | $V_{DD}$ | $V_{SS}$−10 | V |
| Multiplex Frequency | Determined by External R & C | 0.100 | 1.0 | 60 | kHz |
| All Logic Inputs | Driven by External Timebase | dc | | 60 | kHz |
| Logical High Level | Internal Depletion Device to $V_{SS}$ | $V_{SS}$−1 | $V_{SS}$ | $V_{SS}$ | V |
| Logical Low Level | | $V_{DD}$ | $V_{DD}$ | $V_{SS}$−10 | V |
| $\overline{BCD}$ and 7-Segment Outputs | | | | | |
| Logical High Level | Loaded 2 kΩ to $V_{DD}$ | 2.0 | | 20 | mA source |
| Logical Low Level | | | | 0.01 | mA source |
| Digital Enable Outputs | | | | | |
| Logical High Level | | | | 0.3 | mA source |
| Logical Low Level | Loaded 100 Ω to $V_{SS}$ | 5.0 | | 25 | mA sink |

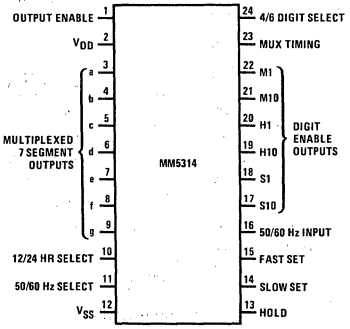
## Connection Diagrams (Cont'd) Dual-In-Line Packages (Top Views)



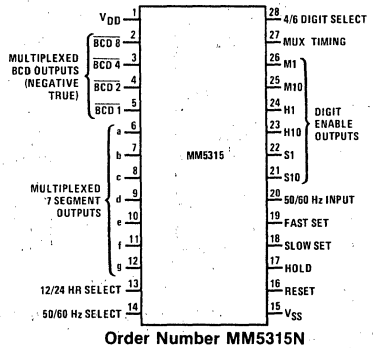

**Order Number MM5312N**
**See Package 22**




**Order Number MM5313N**
**See Package 23**




**Order Number MM5314N**
**See Package 22**




**Order Number MM5315N**
**See Package 23**

## Functional Description

A block diagram of the MM5309 digital clock is shown in *Figure 1*. MM5311, MM5312, MM5313, MM5314 and MM5315 clocks are bonding options of MM5309 clock. Table I shows the pin-outs for these clocks.

**50 or 60 Hz Input:** This input is applied to a Schmitt Trigger shaping circuit which provides approximately 5V of hysteresis and allows using a filtered sinewave input. A simple RC filter such as shown in *Figure 10* should be used to remove possible line voltage transients that could either cause the clock to gain time or damage the device. The shaper output drives a counter chain which performs the timekeeping function.

**50 or 60 Hz Select Input:** This input programs the prescale counter to divide by either 50 or 60 to obtain a 1 Hz timebase. The counter is programmed for 60 Hz operation by connecting this input to $V_{DD}$. An internal depletion device is common to this pin; simply leaving this input unconnected programs the clock for 50 Hz operation. As shown in *Figure 1*, the prescale counter provides both 1 Hz and 10 Hz signals, which can be brought out as bonding options.

**Time Setting Inputs:** Both fast and slow setting inputs, as well as a hold input, are provided. Internal depletion devices provide the normal timekeeping function. Switching any of these inputs (one at a time) to $V_{DD}$ results in the desired time setting function.

The three gates in the counter chain *(Figure 1)* are used for setting time. During normal operation, gate A connects the shaper output to a prescale counter (÷50 or ÷60); gates B and C cascade the remaining counters. Gate A is used to inhibit the input to the counters for the duration of slow, fast or hold time-setting input activity. Gate B is used to connect the shaper output directly to a seconds counter (÷60), the condition for slow advance. Likewise, gate C connects the shaper output directly to a minutes counter (÷60) for fast advance.

Fast set then, advances hours information at one hour per second and slow set advances minutes information at one minute per second.

**12 or 24-Hour Select Input:** This input is used to program the hours counter to divide by either 12 or 24, thereby providing the desired display format. The 12-hour display format is selected by connecting this input to $V_{DD}$; leaving the input unconnected (internal depletion device) selects the 24-hour format.

**Output Multiplexer Operation:** The seconds, minutes, and hours counters continuously reflect the time of day. Outputs from each counter (indicative of both units and tens of seconds, minutes, and hours) are time-division multiplexed to provide digit-sequential access to the time data. Thus, instead of requiring 42 leads to interconnect a 6-digit clock and its display (7 segments per digit), only 13 output leads are required. The multiplexer is addressed by a multiplex divider decoder, which is driven by a multiplex oscillator. The oscillator and external timing components set the frequency of the multiplexing function and, as controlled by the 4 or 6-digit select input, the divider determines whether data will be output for 4 or 6 digits. A zero-blanking circuit suppresses the zero that would otherwise sometimes appear in the tens-of-hours display; blanking is effective only in the 12-hour format. The multiplexer addresses also become the display digit-enable outputs. The multiplexer outputs are applied to a decoder which is used to address a programmable (code converting) ROM. This ROM generates the final output codes, i.e., $\overline{BCD}$ and 7-segment. The sequential output order is from digit 6 (unit seconds) through digit 1 (tens of hours).

**Multiplex Timing Input:** The multiplex oscillator is shown in *Figure 2*. Adding an external resistor and capacitor to this circuit via the multiplex timing input (as shown in *Figure 4a*) produces a relaxation oscillator. The waveform at this input is a quasi-sawtooth that is squared by the shaping action of the Schmitt Trigger in *Figure 2*. *Figure 3* provides guidelines for selecting the external components relative to desired multiplex frequency.

*Figure 4* also illustrates two methods of synchronizing the multiplex oscillator to an external timebase. The external RC timing components may be omitted and this input may be driven by an external timebase; the required logic levels are the same as 50 or 60 Hz input.

**Reset:** Applying $V_{DD}$ to this input resets the counters to 0:00:00.00 in 12-hour format and 00:00:00.00 in 24-hour formats leaving the input unconnected (internal depletion pull-up) selects normal operation. Proper reset will be ensured when $V_{DD}$ to $V_{SS}$ slew rate is no faster than one volt per microsecond. This can be accomplished with a capacitor from the reset input to $V_{SS}$.

**4 or 6-Digit Select Input:** Like the other control inputs, this input is provided with an internal depletion pull-up device. With no input connection the clock outputs data for a 4-digit display. Applying $V_{DD}$ to this input provides a 6-digit display.

**Output Enable Input:** With this pin unconnected the $\overline{BCD}$ and 7-segment outputs are enabled (via an internal depletion pull-up). Switching $V_{DD}$ to this input inhibits these outputs. (Not applicable to MM5312, MM5313, and MM5315 clocks.)

**Output Circuits:** *Figure 5a* illustrates the circuit used for the $\overline{BCD}$ and 7-segment outputs. *Figure 5b* shows the digit enable output circuit. *Figure 6* illustrates interfacing these outputs to standard and low power TTL. *Figures 7 and 8* illustrate methods of interfacing these outputs to common anode and common cathode LED displays, respectively. A method of interfacing these clocks to gas discharge display tubes is shown in *Figure 9*. When driving gas discharge displays which enclose more than one digit in a common gas envelope, it is necessary to inhibit the segment drive voltage(s) during inter-digit transitions. *Figure 9* also illustrates a method of generating a voltage for application to the output enable input to accomplish the required inter-digit blanking.

## Functional Description (Cont'd)

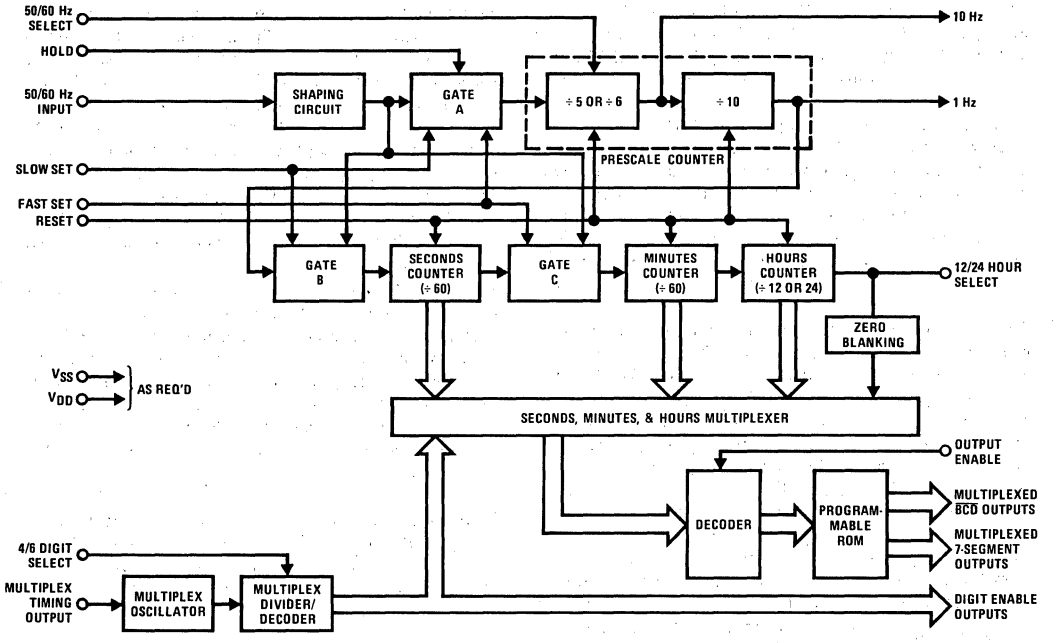



FIGURE 1. MM5309 Digital Clock Block Diagram

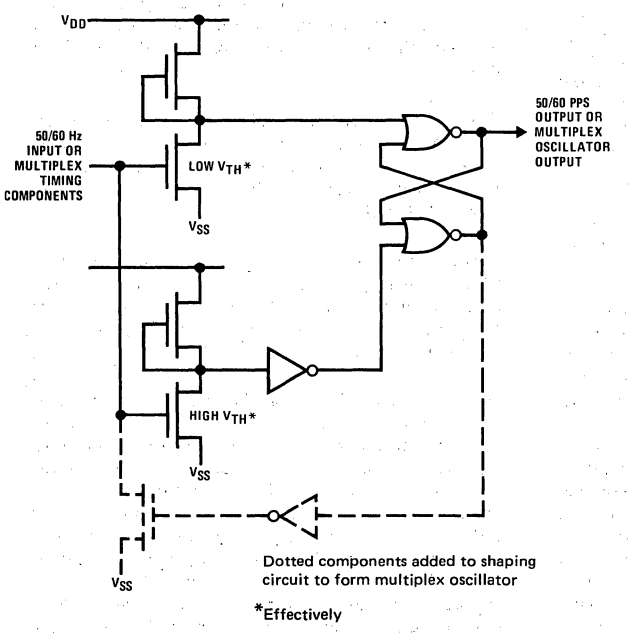



FIGURE 2. 50/60 Hz Shaping Circuit/Multiplex Oscillator

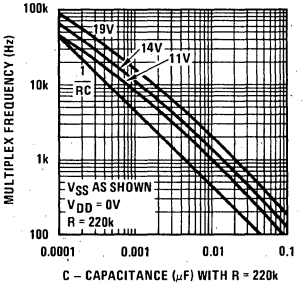



FIGURE 3. Multiplex Timing Component Selection Guide

## Functional Description (Cont'd)

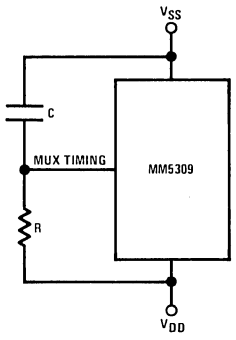



FIGURE 4a. Relaxation Oscillator

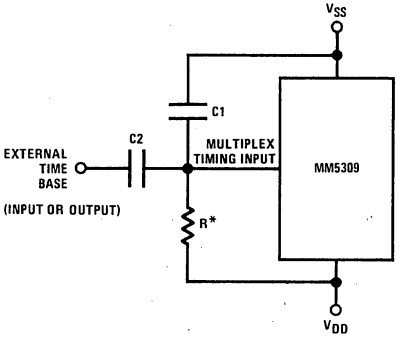



FIGURE 4b. External Time Base

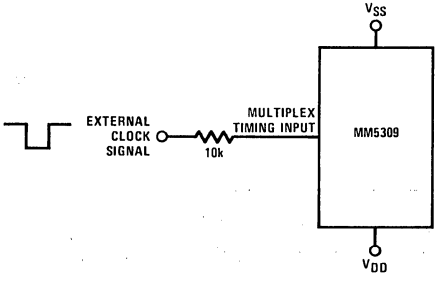



FIGURE 4c. External Clock

Note. Free running frequency should be set to run slightly lower than system frequency over temperature. External time base may be input or output.
* R=100k.

FIGURE 4. Synchronizing or Triggering Multiplex Oscillators

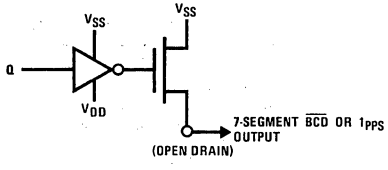



FIGURE 5a

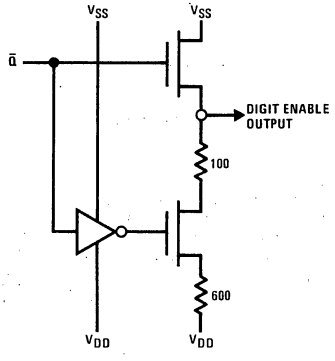
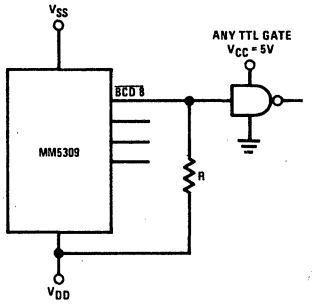



FIGURE 5b

FIGURE 5. Output Circuits

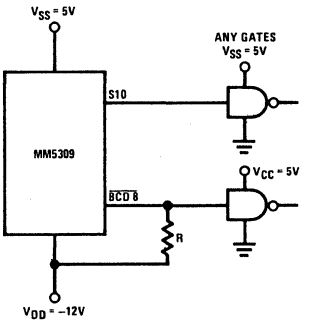
## Functional Description (Cont'd)

**MOS to Low Power TTL Interface**



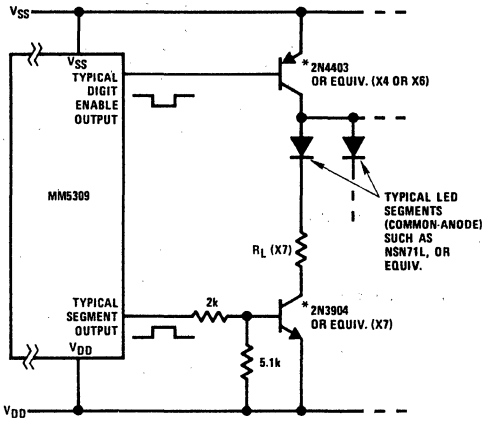

For $V_{SS}$ = 5, $V_{DD}$ = 12, R = 10k
For $V_{SS}$ = 10 to 17V, $V_{DD}$ = Gnd, R = 3k

**MOS to TTL Interface**




For $V_{SS}$ = 5, $V_{DD}$ = −12, R = 7.5k
Note. Digit select will drive TTL directly when 5, −12 supplies are used.

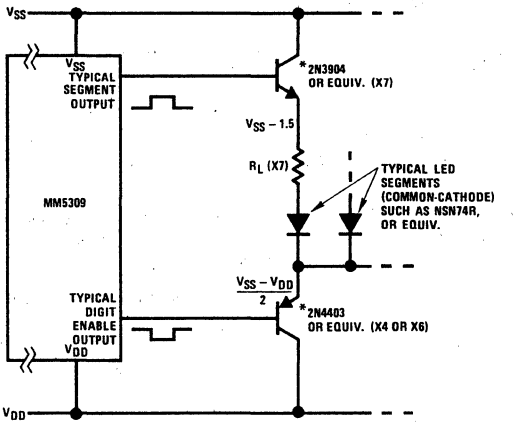
**FIGURE 6. Interfacing TTL**




$$R_L = \frac{V_{SS} - V_{DD} - V_F - 0.6V}{N(I_F)}$$

Where $R_L$ as in kΩ
And $V_F$ = forward drop of LED
0.6V ≈ voltage drop of transistors
N = number of digits in display
$I_F$ = required average LED current

**FIGURE 7. Interfacing Common Anode LED Displays**




$$R_L = \frac{(V_{SS} - V_{DD})/2 - V_F - 1.5V}{N(I_F)}$$

Where $R_L$ is in kΩ
And $V_F$ = forward drop of LED
0.9V = voltage drop of transistors
N = number of digits in display
$I_F$ = required average LED current

**FIGURE 8. Interfacing Common Cathode LED Displays**

*Transistors may be replaced by DM75491, DM75492, DM8861, DM8863 or equivalent segment/digit drivers.

## Functional Description (Cont'd)

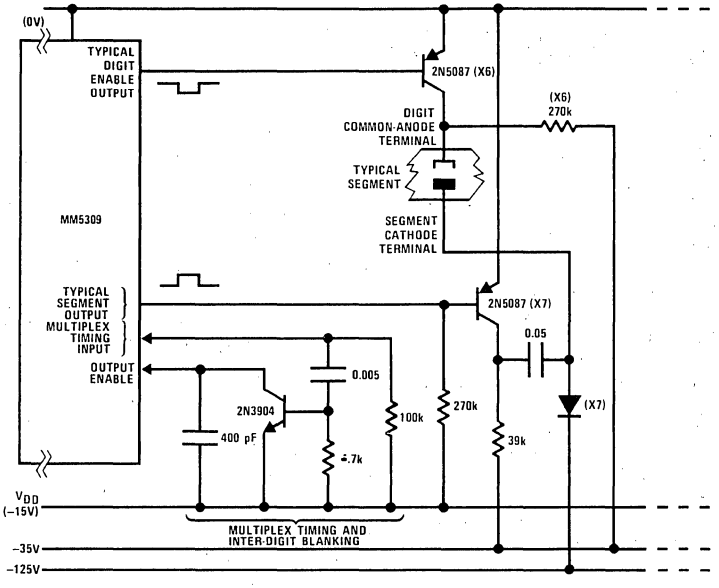



FIGURE 9. Interface Panaplex II* Neon Display Tube

*TM of Burroughs Corp.

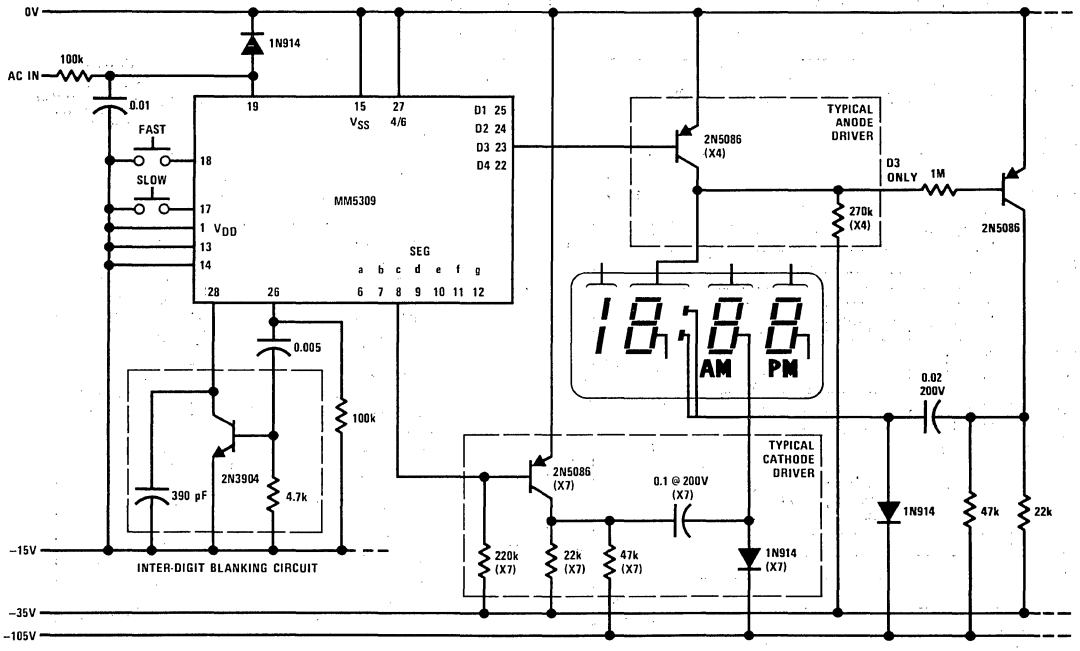



FIGURE 10. MM5309 Driving Gas Discharge Display, Typical Applications

National Semiconductor



# MM5316 Digital Alarm Clock

## General Description

The MM5316 digital alarm clock is a monolithic MOS integrated circuit utilizing P-channel low-threshold, enhancement mode and ion-implanted depletion mode devices. It provides all the logic required to build several types of clocks and timers. Four display modes (time, seconds, alarm and sleep) are provided to optimize circuit utility. The circuit interfaces directly with 7-segment fluorescent tubes, and requires only a single power supply. The timekeeping function operates from either a 50 or 60 Hz input, and the display format may be either 12 hours (with leading-zero blanking and AM/PM indication) or 24 hours. Outputs consist of display drives, sleep (e.g., timed radio turn off), and alarm enable. Power failure indication is provided to inform the user that incorrect time is being displayed. Setting the time cancels this indication. The device operates over a power supply range of 8–29V and does not require a regulated supply. The MM5316 is packaged in a 40-lead dual-in-line package.

## Features

- 50 or 60 Hz operation
- Single power supply
- Low power dissipation (36 mW at 9V)
- 12 or 24-hour display format
- AM/PM outputs } 12-hour format
- Leading-zero blanking }
- 24-hour alarm setting
- All counters are resettable
- Fast and slow set controls
- Power failure indication
- Blanking/brightness control capability
- Elimination of illegal time display at turn on
- Direct interface to fluorescent tubes
- 9-minute snooze alarm
- Presettable 59-minute sleep timer

## Applications

- Alarm clocks
- Desk clocks
- Clock radios
- Automobile clocks
- Stopwatches
- Industrial clocks
- Portable clocks
- Photography timers
- Industrial timers
- Appliance timers
- Sequential controllers

## Block and Connection Diagrams

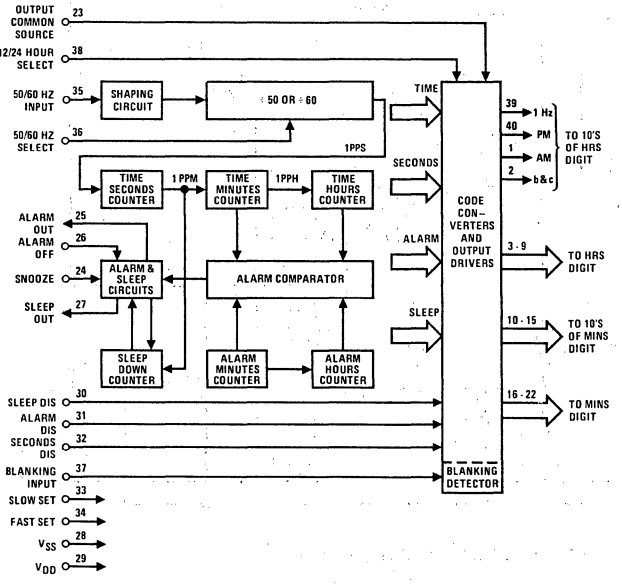
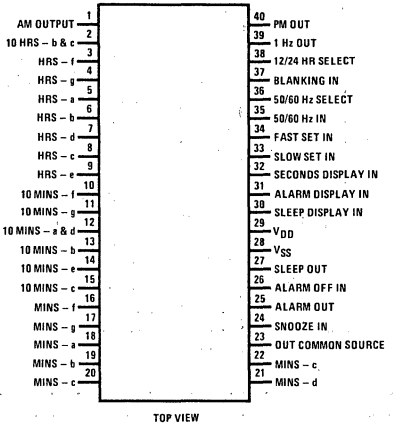



FIGURE 1.

**Dual-In-Line Package**




**Order Number MM5316N**
**See Package 24**

FIGURE 2.

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ + 0.3 to $V_{SS}$ − 30V |
| Operating Temperature | −25°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, $V_{SS}$ = 21V to +29V, $V_{DD}$ = 0V, unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage | $V_{SS}$ ($V_{DD}$ = 0V) | 21 | | 29 | V |
| Power Supply Current | No Output Loads | | | | |
| | $V_{SS}$ = 8V | | | 4 | mA |
| | $V_{SS}$ = 29V | | | 5 | mA |
| Counter Operation Voltage | | 8 | | 29 | V |
| 50/60 Hz Input Frequency Voltage | | dc | 50 or 60 | 10k | Hz |
| Logical High Level | | $V_{SS}$−1 | $V_{SS}$ | $V_{SS}$ | V |
| Logical Low Level | | $V_{DD}$ | $V_{DD}$ | $V_{DD}$+1 | V |
| Blanking Input Voltage | | | | | |
| Logical High Level | | $V_{SS}$−1.5 | $V_{SS}$ | $V_{SS}$ | V |
| Logical Low Level | | $V_{DD}$ | $V_{DD}$ | $V_{SS}$−4 | V |
| All Other Input Voltages | | | | | |
| Logical High Level | | $V_{SS}$−1 | $V_{SS}$ | $V_{SS}$ | V |
| Logical Low Level | Internal Depletion Device to $V_{DD}$ | $V_{DD}$ | $V_{DD}$ | $V_{DD}$+2 | V |
| Power Failure Detect Voltage | ($V_{SS}$ Voltage) | 10 | | 20 | V |
| Output Currents, 1 Hz Display | $V_{SS}$ = 21V to 29V, Output Common = $V_{SS}$ | | | | |
| Logical High Level | $V_{OH}$ = $V_{SS}$ − 2V | 1500 | | | μA |
| Logical Low Level, Leakage | $V_{OL}$ = $V_{DD}$ | | | 1 | μA |
| 10's of Hours (b & c), 10's of Minutes (a & d) | | | | | |
| Logical High Level | $V_{OH}$ = $V_{SS}$ − 2V | 1000 | | | μA |
| Logical Low Level, Leakage | $V_{OL}$ = $V_{DD}$ | | | 1 | μA |
| All Other Display, Alarm and Sleep Outputs | | | | | |
| Logical High Level | $V_{OH}$ = $V_{SS}$ − 2V | 500 | | | μA |
| Logical Low Level, Leakage | $V_{OL}$ = $V_{DD}$ | | | 1 | μA |

## Functional Description

A block diagram of the MM5316 digital alarm clock is shown in *Figure 1*. The various display modes provided by this clock are listed in Table I. The functions of the setting controls are listed in Table II. *Figure 2* is a connection diagram. The following discussions are based on *Figure 1*.

**50 or 60 Hz Input (pin 35):** A shaping circuit *(Figure 3)* is provided to square the 50 or 60 Hz input. This circuit allows use of a filtered sinewave input. The circuit is a Schmitt Trigger that is designed to provide about 6V of hysteresis. A simple RC filter, such as shown in *Figure 6*, should be used to remove possible line-voltage transients that could either cause the clock to gain time or damage the device. The shaper output drives a counter chain which performs the timekeeping function.

**50 or 60 Hz Select Input (pin 36):** A programmable prescale counter divides the input line frequency by either 50 or 60 to obtain a 1 Hz time base. This counter is programmed to divide by 60 simply by leaving pin 36 unconnected; pull-down to $V_{DD}$ is provided by an internal depletion device. Operation at 50 Hz is programmed by connecting pin 36 to $V_{SS}$.

**Display Mode Select Inputs (pins 30–32):** In the absence of any of these three inputs, the display drivers present time-of-day information to the appropriate display digits. Internal pull-down depletion devices allow use of simple SPST switches to select the display mode. If more than one mode is selected, the priorities are as noted in Table I. Alternate display modes are selected by applying $V_{SS}$ to the appropriate pin. As shown in *Figure 1* the code converters receive time, seconds, alarm and sleep information from appropriate points in the clock circuitry. The display mode select inputs control the gating of the desired data to the code converter inputs and ultimately (via output drivers) to the display digits.

**Time Setting Inputs (pins 33 and 34):** Both fast and slow setting inputs are provided. These inputs are applied either singly or in combination to obtain the control functions listed in Table II. Again, internal pull-down depletion devices are provided; application of $V_{SS}$ to these pins effects the control functions. Note that the control functions proper are dependent on the selected display mode. For example, a hold-time control function is obtained by selecting seconds display and actuating the slow set input. As another example, the clock time may be reset to 12:00:00 AM, in the 12-hour format (00:00:00 in the 24-hour format), by selecting seconds display and actuating both slow and fast set inputs.

**Blanking Control Input (pin 37):** Connecting this Schmitt Trigger input to $V_{DD}$ places all display drivers in a non-conducting, high-impedance state, thereby inhibiting the display, (see *Figures 3 and 4*). Conversely, $V_{SS}$ applied to this input enables the display.

**Output Common Source Connection (pin 23):** All display output drivers are open-drain devices with all sources common to pin 23 *(Figure 4)*. When using fluorescent tube displays, $V_{SS}$ or a display brightness control voltage is permanently connected to this pin. Since the brightness of a fluorescent tube display is dependent on the anode (segment) voltage, applying a variable voltage to pin 23 results in a display brightness control. This control is shown in *Figure 6*.

**12 or 24-Hour Select Input (pin 38):** By leaving this pin unconnected, the outputs for the most-significant display digit (10's of hours) are programmed to provide a 12-hour display format. An internal depletion pull down device is again provided. Connecting this pin to $V_{SS}$ programs the 24-hour display format. Segment connections for 10's of hours in 24-hour mode are shown in *Figure 5b*.

**Power Fail Indication:** If the power to the integrated circuit drops indicating a momentary ac power failure and possible loss of clock, the power fail latch is set. The power failure indication consists of a flashing of the AM or PM indicator at a 1 Hz rate. A fast or slow set input resets an internal power failure latch and returns the display to normal. In the 24-hour format, the power failure indication consists of flashing segments "c" and "f" for times less than 10 hours, and of a flashing segment "c" for times equal to or greater than 10 hours but less than 20 hours; and a flashing segment "g" for times equal to or greater than 20 hours.

**Alarm Operation and Output (pin 25):** The alarm comparator *(Figure 1)* senses coincidence between the alarm counters (the alarm setting) and the time counters (real time). The comparator output is used to set a latch in the alarm and sleep circuits. The latch output enables the alarm output driver *(Figure 4)*, the MM5316 output that is used to control the external alarm sound generator. The alarm latch remains set for 59 minutes, during which the alarm will therefore sound if the latch output is not temporarily inhibited by another latch set by the snooze alarm input (pin 24) or reset by the alarm "OFF" input (pin 26). If power fail occurs and power comes back up, the alarm output will be in high impedance state.

**Snooze Alarm Input (pin 24):** Momentarily connecting pin 24 to $V_{SS}$ inhibits the alarm output for between 8 and 9 minutes, after which the alarm will again be sounded. This input is pulled-down to $V_{DD}$ by an internal depletion device. The snooze alarm feature may be repeatedly used during the 59 minutes in which the alarm latch remains set.

**Alarm "OFF" Input (pin 26):** Momentarily connecting pin 26 to $V_{SS}$ resets the alarm latch and thereby silences the alarm. This input is also returned to $V_{DD}$ by an internal depletion device. The momentary alarm "OFF" input also readies the alarm latch for the next comparator output, and the alarm will automatically sound again in 24 hours (or at a new alarm setting). If it is desired to silence the alarm for a day or more, the alarm "OFF" input should remain at $V_{SS}$.

**Sleep Timer and Output (pin 27):** The sleep output at pin 27 can be used to turn off a radio after a

## Functional Description (Cont'd)

desired time interval of up to 59 minutes. The time interval is chosen by selecting the sleep display mode (Table I) and setting the desired time interval (Table II). This automatically results in a current-source output via pin 27, which can be used to turn on a radio (or other appliance). When the sleep counter, which counts downwards, reaches 00 minutes, a latch is reset and the sleep output current drive is removed, thereby turning off the radio. The turn off may also be manually controlled (at any time in the countdown) by a momentary $V_{SS}$ connection to the snooze input (pin 24). The output circuitry is the same as the other outputs *(Figure 4)*.

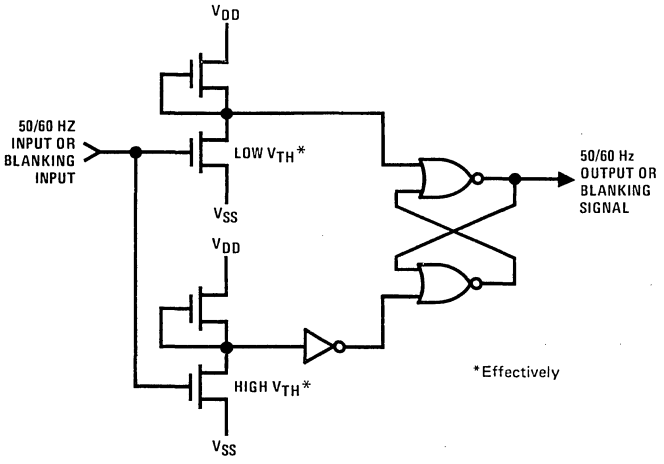
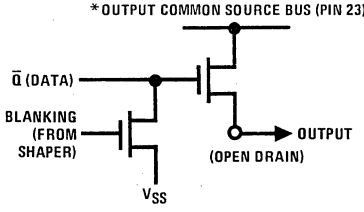
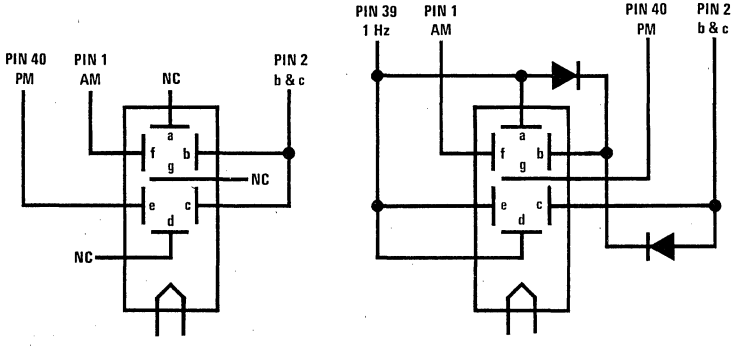



FIGURE 3. 50/60 Hz or Blanking Input Shaping Circuit




*Alarm and sleep output sources are connected to $V_{SS}$: blanking is not applied to these outputs.

FIGURE 4. Output Circuit




(a) 12-Hour Display Format (b) 24-Hour Display Format

FIGURE 5. Wiring Ten's-of-Hours Digit

## Functional Description (Cont'd)

TABLE I. MM5316 Display Modes

| *SELECTED DISPLAY MODE | DIGIT NO. 1 | DIGIT NO. 2 | DIGIT NO. 3 | DIGIT NO. 4 |
|---|---|---|---|---|
| Time Display | 10's of Hours & AM/PM | Hours | 10's of Minutes | Minutes |
| Seconds Display | Blanked | Minutes | 10's of Seconds | Seconds |
| Alarm Display | 10's of Hours & AM/PM | Hours | 10's of Minutes | Minutes |
| Sleep Display | Blanked | Blanked | 10's of Minutes | Minutes |

*If more than one display mode input is applied, the display priorities are in the order of Sleep (overrides all others), Alarm, Seconds, Time (no other mode selected).

TABLE II. MM5316 Setting Control Functions

| SELECTED DISPLAY MODE | CONTROL INPUT | CONTROL FUNCTION |
|---|---|---|
| *Time | Slow | Minutes Advance at 2 Hz Rate |
| | Fast | Minutes Advance at 60 Hz Rate |
| | Both | Minutes Advance at 60 Hz Rate |
| Alarm | Slow | Alarm Minutes Advance at 2 Hz Rate |
| | Fast | Alarm Minutes Advance at 60 Hz Rate |
| | Both | Alarm Resets to 12:00 AM (12-hour format) |
| | Both | Alarm Resets to 00:00 (24-hour format) |
| Seconds | Slow | Input to Entire Time Counter is Inhibited (Hold) |
| | Fast | Seconds and 10's of Seconds Reset to Zero Without a Carry to Minutes |
| | Both | Time Resets to 12:00:00 AM (12-hour format) |
| | Both | Time Resets to 00:00:00 (24-hour format) |
| Sleep | Slow | Substracts Count at 2 Hz |
| | Fast | Substracts Count at 60 Hz |
| | Both | Substracts Count at 60 Hz |

*When setting time sleep minutes will decrement at rate of time counter, until the sleep counter reaches 00 minutes (sleep counter will not recycle).

## Typical Application

*Figure 6* is a schematic diagram of a general purpose alarm clock using the MM5316 and a fluorescent tube display.

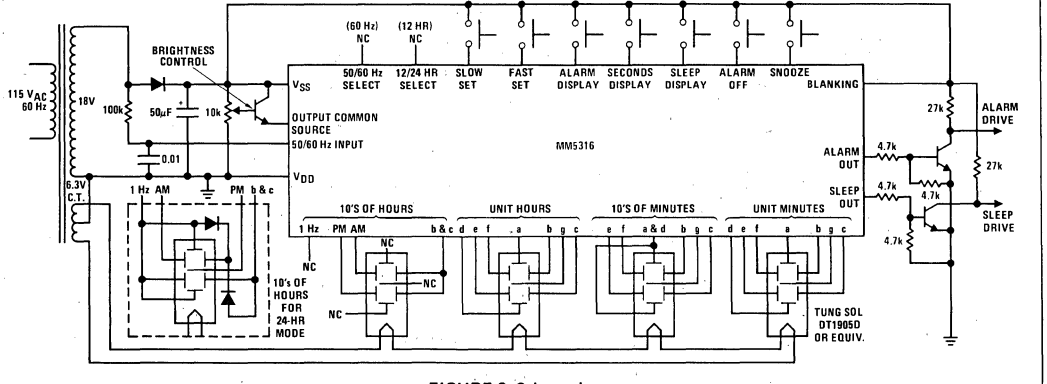



FIGURE 6. Schematic

# MM5387AA, MM53108 Digital Alarm Clocks

## General Description

The MM5387AA, MM53108 digital alarm clocks are monolithic MOS integrated circuits utilizing P-channel low-threshold, enhancement mode and ion-implanted depletion mode devices. They provide all the logic required to build several types of clocks and timers with up to four display modes (time, seconds, alarm and sleep) to maximize circuit utility, but are specifically intended for clock-radio applications. Both devices will directly-drive 7-segment LED displays in either a 12 hour format (3½ digits) with lead-zero blanking, AM/PM indication and flashing colon, or 24 hour format (4 digits) through hard-wire pin selection; the timekeeping function operates from either a 50 or 60 Hz input, also through pin selection. Outputs consist of display drivers, sleep (e.g., timed radio turn-off), and alarm enable. A power-fail indication mode is provided to inform the user of incorrect time display by flashing all "ON" digits at a 1 Hz rate, and is cancelled by simply resetting time. The device operates over a supply range of 24–26V which does not require regulation.

The MM53108 is electrically identical to the MM5387AA, but with mirror-image pin-out to facilitate PC board layout when designing a "module" where the LED display and MOS chip are mounted on the same side; the MM5387AA is more suited for "L" shaped module designs (vertical LED display, horizontal component board). Both devices are supplied in a 40-lead dual-in-line package.

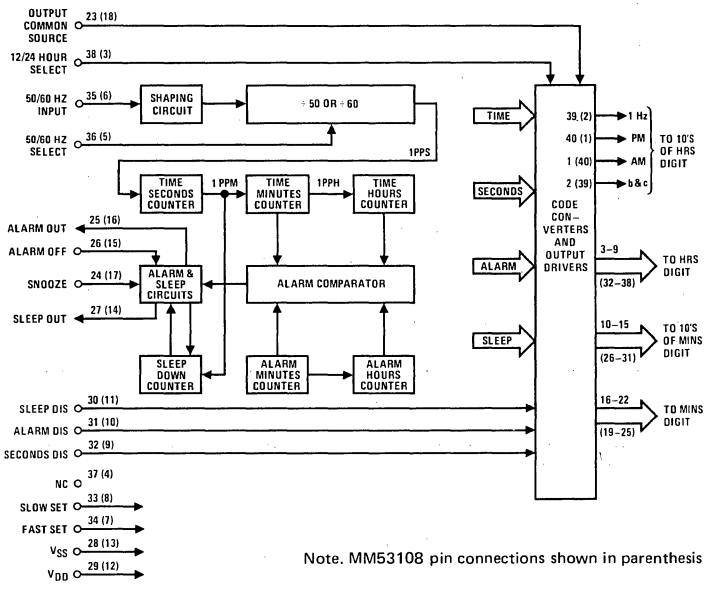
## Features

- 50 or 60 Hz operation
- Single power supply
- 12 or 24 hour display format
- AM/PM outputs } 12 hour format
- Leading-zero blanking }
- 24-hour alarm setting
- All counters are resettable
- Fast and slow set controls
- Power failure indication
- Elimination of illegal time display at turn "ON"
- Direct interface to LED displays
- 9-minute snooze alarm
- Presettable 59-minute sleep timer
- Available in standard (MM5387AA) or mirror image (MM53108) pin-out

## Applications

- Alarm clocks
- Desk clocks
- Clock radios
- Automobile clocks
- Stopwatches
- Industrial clocks
- Portable clocks
- Photography timers
- Industrial timers
- Appliance timers
- Sequential controllers

## Block Diagram




Note. MM53108 pin connections shown in parenthesis

FIGURE 1

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Except Segment Outputs | $V_{SS}$ + 0.3 to $V_{SS}$ − 30V |
| Voltage at Segment Outputs | $V_{SS}$ + 0.3 to $V_{SS}$ − 15V |
| Operating Temperature | −25°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, $V_{SS}$ = 24V–26V, $V_{DD}$ = 0V, unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage | Output Driving Display | 24 | | 26 | V |
| | Functional Clock | 8 | | 26 | V |
| Power Supply Current | No Output Loads | | | | |
| | $V_{SS}$ = 8V | | | 4 | mA |
| | $V_{SS}$ = 26V | | | 5 | mA |
| 50/60 Hz Input | | | | | |
| Frequency Voltage | $V_{SS}$ = 8V to 26V | dc | 50 or 60 | 10k | Hz |
| Logical High Level | | $V_{SS}$−1 | $V_{SS}$ | $V_{SS}$ | V |
| Logical Low Level | | $V_{DD}$ | $V_{DD}$ | $V_{DD}$+2 | V |
| Input Leakage | | | | 100 | μA |
| All Other Input Voltages | | | | | |
| Logical High Level | | $V_{SS}$−1 | $V_{SS}$ | $V_{SS}$ | V |
| Logical Low Level | Internal Depletion Load to $V_{DD}$ | $V_{DD}$ | $V_{DD}$ | $V_{SS}$−6 | V |
| Power Failure Detect Voltage | ($V_{SS}$ Voltage), (Note 2) | 1 | | 7.5 | V |
| Count Operating Voltage | | 8 | | 26 | V |
| Hold Count Voltage | | (Note 2) | | 26 | V |
| Output Current Levels | $V_{SS}$ = 24V to 26V, Output Common = $V_{SS}$ | | | | |
| 10's of Hours (b & c), 10's of Minutes (a & d) | | | | | |
| Logical High Level, Source | $V_{OH}$ = $V_{SS}$ − 4V | 16 | | | mA |
| Logical Low Level, Leakage | $V_{OL}$ = $V_{SS}$ − 14V | | | 10 | μA |
| 1 Hz Display | | | | | |
| Logical High Level, Source | $V_{OH}$ = $V_{SS}$ − 4 | 24 | | | mA |
| Logical Low Level, Leakage | $V_{OL}$ = $V_{SS}$ − 14 | | | 10 | μA |
| All Other Displays | | | | | |
| Logical High Level, Source | $V_{OH}$ = $V_{SS}$ − 4V | 8 | | (Note 1) | mA |
| Logical Low Level, Leakage | $V_{OL}$ = $V_{SS}$ − 14V | | | 10 | μA |
| Alarm and Sleep Outputs | $V_{SS}$ = 24V | | | | |
| Logical High , Source | $V_{OH}$ = $V_{SS}$ − 2 | 500 | | | μA |
| Logical Low, Sink | $V_{OL}$ = $V_{SS}$ − 2 | 1 | | | μA |

**Note 1:** Segment output current must be limited to 11 mA maximum by user; power dissipation must be limited to 900 mW at 70°C and 1.2W at 25°C.

**Note 2:** The power-fail detect voltage is 0.5V or more above the hold count voltage. The power-fail latch trips into power-fail mode at least 0.5V above the voltage at which data stored in the time latch is lost.

## Functional Description

A block diagram of the MM5387AA, MM53108 digital clock radio circuit is shown in *Figure 1*. The various display setting modes are listed in Table I, and Table II shows the setting control functions. The following description is based on *Figure 1* and refers to both devices as they are electrically identical.

**50 or 60 Hz Input:** A shaping circuit *(Figure 3)* is provided to square the 50 or 60 Hz input. This circuit allows use of a filtered sinewave input. The circuit is a Schmitt trigger that is designed to provide about 6V of hysteresis. A simple RC filter such as shown in *Figure 7*, should be used to remove possible line-voltage transients that could either cause the clock to gain time or damage the device. The shaper output drives a counter chain which performs the timekeeping function.

**50 or 60 Hz Select Input:** A programmable prescale counter divides the input line frequency by either 50 or 60 to obtain a 1 Hz time base. This counter is programmed to divide by 60 simply by leaving 50/60 Hz select unconnected; pull-down to $V_{DD}$ is provided by an internal depletion load. Operation at 50 Hz is programmed by connecting 50/60 Hz select to $V_{SS}$.

**Display Mode Select Inputs:** In the absence of any of these three inputs, the display drivers present time-of-day information to the appropriate display digits. Internal depletion pull-down devices allow use of simple SPST switches to select the display mode. If more than one mode is selected, the priorities are as noted in Table I. Alternate display modes are selected by applying $V_{SS}$ to the appropriate pin. As shown in *Figure 1* the code converters receive time, seconds, alarm and sleep information from appropriate points in the clock circuitry. The display mode select inputs control the gating of the desired data to the code converter inputs and ultimately (via output drivers) to the display digits.

**Time Setting Inputs:** Both fast and slow setting inputs are provided. These inputs are applied either singly or in combination to obtain the control functions listed in Table II. Again, internal depletion pull-down devices are provided; application of $V_{SS}$ to these pins affects the control functions. Note that the control functions proper are dependent on the selected display mode. For example, a hold-time control function is obtained by selecting seconds display and actuating the slow set input. As another example, the clock time may be reset to 12:00:00 AM, by selecting seconds display and actuating both slow and fast set inputs.

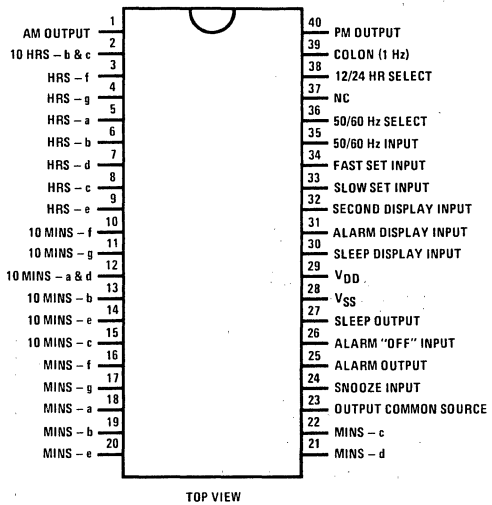
**Output Common Source Connection:** All display output drivers are open-drain devices with all sources common *(Figure 4a)*. The common source pin should be connected to $V_{SS}$.

**12 or 24 Hour Select Input:** By leaving this pin unconnected, the outputs for the most-significant display digit (10's of hours) are programmed to provide a 12-hour display format. An internal depletion pull-down device is again provided. Connecting this pin to $V_{SS}$ programs the 24-hour display format. Segment connections for 10's of Hours in 24-hour mode are shown in *Figure 6*.

**Power Fail Indication:** If the power to the integrated circuit drops, indicating a momentary ac power failure and possible loss of clock, all "ON" segments will flash at 1 Hz rate. A fast or slow set input resets an internal power failure latch and returns the display to normal.

## Connection Diagrams

Dual-In-Line Package

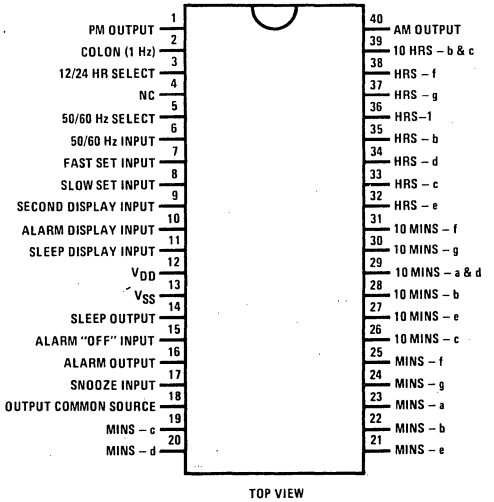



TOP VIEW

Order Number MM5387AA/N
See Package 24

FIGURE 2(a). MM5387AA

Dual-In-Line Package




TOP VIEW

Order Number MM53108N
See Package 24

FIGURE 2(b). MM53108 (Mirror Image Pin-Out)

## Functional Description (Cont'd)

**Alarm Operation and Output:** The alarm comparator *(Figure 1)* senses coincidence between the alarm counters (the alarm setting) and the time counters (real time). The comparator output is used to set a latch in the alarm and sleep circuits. The latch output enables the alarm output driver *(Figure 4b)* which is used to control the external alarm sound generator. The alarm latch remains set for 59 minutes, during which the alarm will therefore sound if the latch output is not temporarily inhibited by another latch set by the snooze alarm input or reset by the alarm "OFF" input.

**Snooze Alarm Input:** Momentarily connecting snooze to $V_{SS}$ inhibits the alarm output for between 8 and 9 minutes, after which the alarm will again be sounded. This input is pulled-down to $V_{DD}$ by an internal depletion device. The snooze alarm feature may be repeatedly used during the 59 minutes in which the alarm latch remains set.

**Alarm "OFF" Input:** Momentarily connecting alarm "OFF" to $V_{SS}$ resets the alarm latch and thereby silences the alarm. This input is also returned to $V_{DD}$ by an internal depletion device. The momentary alarm "OFF" input also readies the alarm latch for the next comparator output, and the alarm will automatically sound again in 24 hours (or at a new alarm setting). If it is desired to silence the alarm for a day or more, the alarm "OFF" input should remain at $V_{SS}$.

**Sleep Timer and Output:** The sleep output can be used to turn "OFF" a radio after a desired time interval of up to 59 minutes. The time interval is chosen by selecting the sleep display mode, (Table I) and setting the desired time interval (Table II). This automatically results in a current-source output which can be used to turn "ON" a radio (or other appliance). When the sleep counter, which counts downwards, reaches 00 minutes, a latch is reset and the sleep output current drive is removed, thereby turning "OFF" the radio. This turn "OFF" may also be manually controlled (at any time in the countdown) by a momentary $V_{SS}$ connection to the Snooze input. The output circuitry is the same as the other outputs *(Figure 4b)*.

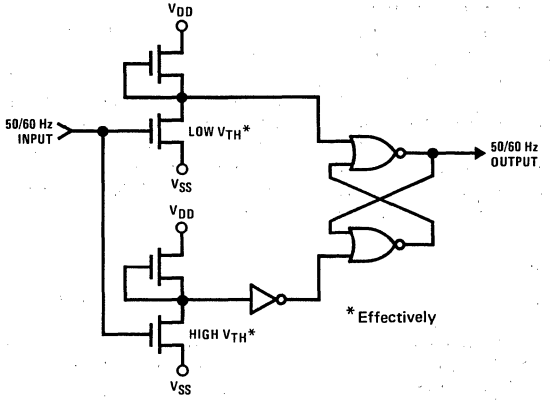



FIGURE 3. 50/60 Hz Input Shaping Circuit

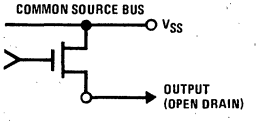



FIGURE 4(a). Segment Outputs

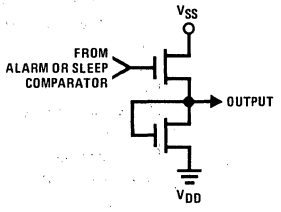



FIGURE 4(b). Alarm and Sleep Outputs

## Functional Description (Cont'd)

TABLE I. MM5387AA, MM53108 Display Modes

| *SELECTED DISPLAY MODE | DIGIT NO. 1 | DIGIT NO. 2 | DIGIT NO. 3 | DIGIT NO. 4 |
|---|---|---|---|---|
| Time Display | 10's of Hours & AM/PM | Hours | 10's of Minutes | Minutes |
| Seconds Display | Blanked | Minutes | 10's of Seconds | Seconds |
| Alarm Display | 10's of Hours & AM/PM | Hours | 10's of Minutes | Minutes |
| Sleep Display | Blanked | Blanked | 10's of Minutes | Minutes |

*If more than one display mode input is applied, the display priorities are in the order of Sleep (overrides all others), Alarm, Seconds, Time (no other mode selected).

TABLE II. MM5387AA, MM53108 Setting Control Functions

| SELECTED DISPLAY MODE | CONTROL INPUT | CONTROL FUNCTION |
|---|---|---|
| *Time | Slow | Minutes Advance at 2 Hz Rate |
| | Fast | Minutes Advance at 60 Hz Rate |
| | Both | Minutes Advance at 60 Hz Rate |
| Alarm | Slow | Alarm Minutes Advance at 2 Hz Rate |
| | Fast | Alarm Minutes Advance at 60 Hz Rate |
| | Both | Alarm Resets to 12:00 AM (Midnight) (12-Hour Format) |
| | Both | Alarm Resets to 00:00 (24-Hour Format) |
| Seconds | Slow | Input to Entire Time Counter is Inhibited (Hold) |
| | Fast | Seconds and 10's of Seconds Reset to Zero Without a Carry to Minutes |
| | Both | Time Resets to 12:00:00 AM (Midnight) (12-Hour Format) |
| | Both | Time Resets to 00:00:00 (24-Hour Format) |
| Sleep | Slow | Subtracts Count at 2 Hz |
| | Fast | Subtracts Count at 60 Hz |
| | Both | Subtracts Count at 60 Hz |

*When setting time sleep minutes will decrement at rate of time counter, until the sleep counter reaches 00 minutes (sleep counter will not recycle).

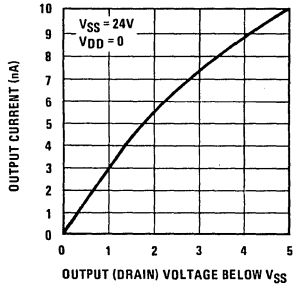
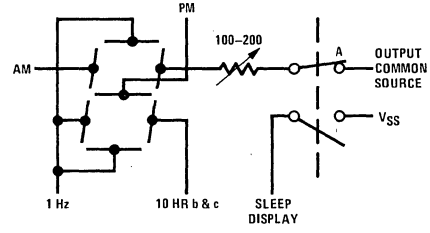



FIGURE 5. Typical Output Current Characteristics of MM5387AA, MM53108




Switch A must be ganged with Sleep display as shown.

FIGURE 6. 24-Hour Operation: 10's of Hours Digit Connections

## Typical Application

*Figure 7* is a schematic diagram of a general purpose alarm clock circuit (12-hour mode) using the MM5387AA or MM53108 and a 3 1/2-digit LED display.

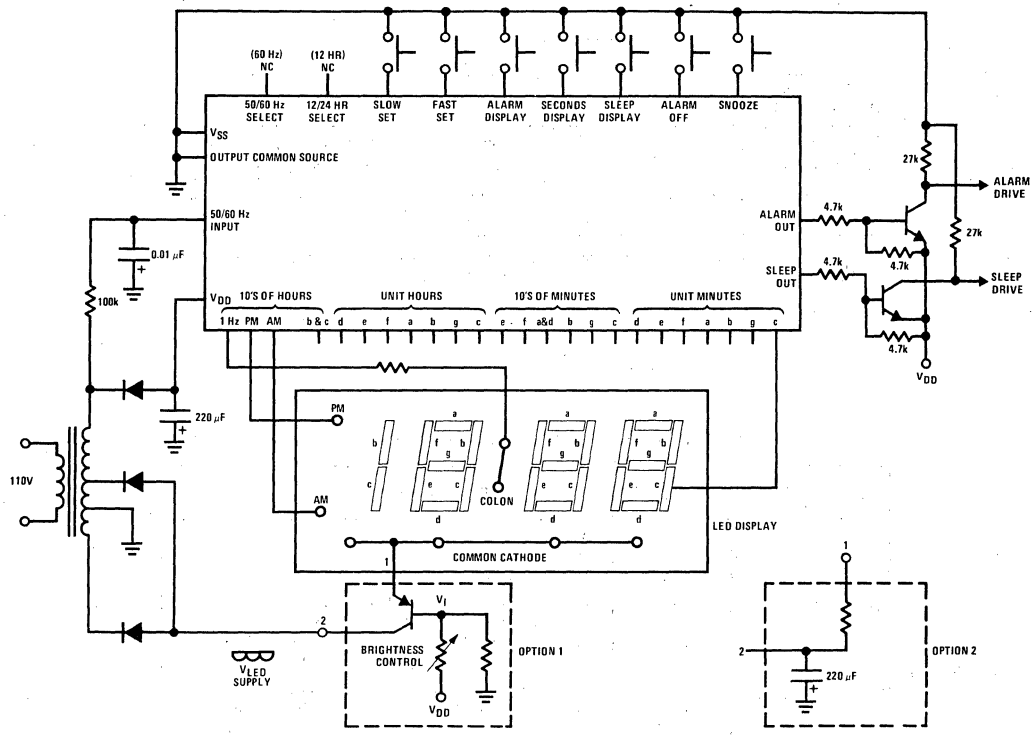



FIGURE 7

National Semiconductor



# MM53110 Series Auto Clock and Elapsed Timer

## General Description

The MM53110 series clock is a monolithic MOS integrated circuit utilizing P-channel low-threshold, enhancement mode and ion-implanted depletion mode devices. The circuit interfaces simply with vacuum-fluorescent 4-digit displays. The display format is 12 hours for time display and 20 hours for elapsed time display with leading-zero blanking and colon indication. The timekeeping function operates from a 2 MHz crystal-controlled source.

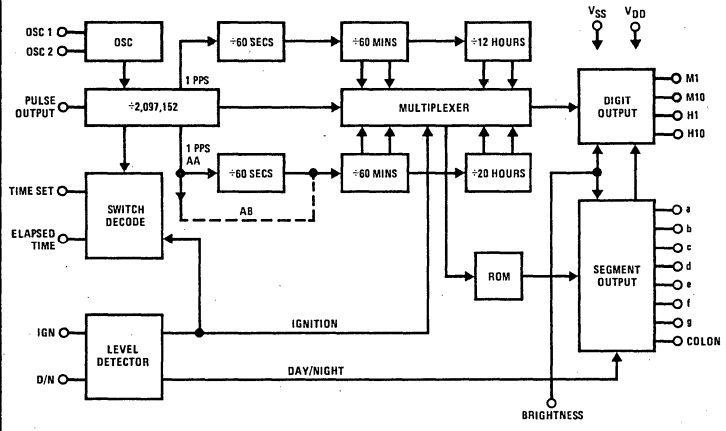
## Features

- Elapsed time display
- Crystal-controlled oscillator (2.097152 MHz)
- 12-hour display format
- Colon output (1 1/2 seconds ON and 1/2 second OFF)
- Leading-zero blanking
- Hours and minutes set controls
- Elapsed time reset and display control
- Brightness control capability
- DC/DC converter pulse output
- Day/night control
- Elimination of illegal time display at turn-on
- Simple interface to vacuum-fluorescent displays
- Low standby power dissipation

## Applications

- Elapsed timer
- Automobile clocks
- Desk clocks
- Portable clocks
- High accuracy clocks

## Block Diagram



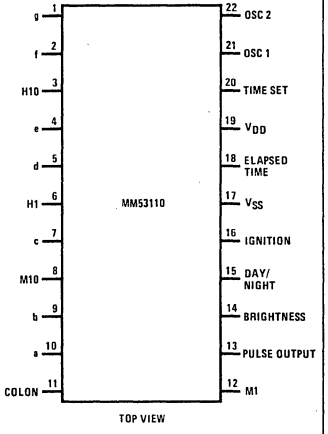

**Available Options Table**

| OPTION NAME | ELAPSED TIME FORMAT |
|---|---|
| AA | Hours-Minutes |
| AB | Minutes-Seconds |

**FIGURE 1. MM53110 Auto Clock Diagram**

## Connection Diagram

**Dual-In-Line Package**




TOP VIEW

**Order Number MM53110AA/N or MM53110AB/N**
**See Package 21**

5

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ + 0.3V to $V_{SS}$ − 29V |
| Voltage at Any Display Output Pin | $V_{SS}$ + 0.3V to $V_{SS}$ − 29V |
| Operating Temperature | −40°C to +85°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, $V_{SS}$ = 9V to 28V, $V_{DD}$ = 0V, unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage ($V_{SS}$) | Outputs and Osc. Operational | 9 | 25 | 28 | V |
| Power Supply Voltage ($V_{SS}$) | No Loss of Time Memory | 5 | 12 | 28 | V |
| Power Supply Current ($I_{SS}$) | $V_{SS}$ = 12V, No Output Loads, (Ignition OFF) | 1 | 5 | 8 | mA |
| Power Supply Current ($I_{SS}$) | $V_{SS}$ = 25V, No Output Loads, (Ignition ON) | 1 | 9 | 15 | mA |
| Power Supply Rise Time ($t_R$) | | 100 | | | μs/V |
| Input Frequency | Osc. 1 | DC | 2.097152 | 2.1 | MHz |
| Multiplex Rate | $f_{IN}$ = 2.097152 MHz | | 2048 | | Hz |
| Brightness Input Voltage | (Note 1) | | | | |
| 0–100% Intensity | Day/Night Input = High | 0 | 14 | $V_{SS}$ | V |
| 100% Intensity | Day/Night Input = Low | | $V_{SS}$ | | V |
| Set Input Current | $V_{IN}$ = $V_{SS}$ or $V_{DD}$, $V_{SS}$ = 25V | | | ±4.0 | mA |
| Set Input Voltage | | | | | |
| Logical High Level | Set Minutes | $V_{SS}$−1.0 | $V_{SS}$ | $V_{SS}$ | V |
| Logical Low Level | Set Hours | $V_{DD}$ | $V_{DD}$ | $V_{DD}$+1.0 | V |
| Input Floating | Display Time | | Floating | | |
| Elapsed Time Input Voltage | | | | | |
| Logical High Level | Display Elapsed Time | $V_{SS}$−1.0 | $V_{SS}$ | $V_{SS}$ | V |
| Logical Low Level | Reset Elapsed Time | $V_{DD}$ | $V_{DD}$ | $V_{DD}$+1.0 | V |
| Input Floating | Display Time | | Floating | | |
| Ignition Input Voltage | Internal 1M Resistor to $V_{DD}$ | | | | |
| Logical High Level (ON) | $V_{SS}$ = 12V to 25V | 6.5 | 14 | $V_{SS}$ | V |
| Logical Low Level (OFF) | $V_{SS}$ = 25V | 0 | 0 | 1.8 | V |
| Day/Night Input Voltage | Internal 1M Resistor to $V_{DD}$ | | | | |
| Logical High Level (Night) | $V_{SS}$ = 25V | 6.5 | 14 | $V_{SS}$ | V |
| Logical Low Level (Day) | $V_{SS}$ = 25V | 0 | 0 | 1.8 | V |
| Ignition Input Current | $V_{SS}$ = 25V, $V_{IN}$ = 12V | 4 | | 40 | μA |
| Day/Night Input Current | $V_{SS}$ = 25V, $V_{IN}$ = 12V | 4 | | 40 | μA |
| Pulse Output | $V_{OH}$ = $V_{SS}$ − 2V | | | | |
| Logical High Level | $V_{SS}$ = 9V | 1 | | | mA |
| Logical High Level | $V_{SS}$ = 28V | 4 | | | mA |
| Logical Low Level | $V_{OL}$ = $V_{DD}$ + 1V<br>Internal Diffused Resistor to $V_{DD}$ | 6 | | | μA |
| Digit Outputs | | | | | |
| Logical High Level (ON) | $V_{OH}$ = $V_{SS}$ − 1V | 8.0 | | | mA |
| Logical Low Level (OFF) | $V_{OL}$ = $V_{DD}$ + 2V | 40 | | | μA |
| Segment, Colon Outputs | | | | | |
| Logical High Level (ON) | $V_{OH}$ = $V_{SS}$ − 1V | 2.0 | | | mA |
| Logical Low Level (OFF) | $V_{OL}$ = $V_{DD}$ + 2V | 40 | | | μA |

**Note 1:** In the day position, the brightness input is internally forced to $V_{SS}$ in order to supply maximum voltage to the display.

## Functional Description

A block diagram of the MM53110 auto clock is shown in *Figure 1*. Connection diagrams for this device are shown on the front page. Unless otherwise indicated, the following discussions are based on *Figure 1*.

**Crystal Oscillator (Pins 21 and 22):** A quartz crystal, resonant at 2.097152 MHz, 2 capacitors and 1 resistor together with the internal MOS circuits form a crystal-controlled oscillator as shown in *Figure 2*. Varying 1 of the capacitors allows precise frequency setting. For test purposes, Osc. 1 is the input and Osc. 2 is the output of an inverting amplifier.

**Time Setting (Pin 20):** Time setting is accomplished via the set input pin. If this input is a logic high, the minutes counter will advance at a 2 Hz rate with no carry to the hours counter and will also cause seconds counter to reset. If the set input is a logic low, the hours counter will advance at a 2 Hz rate, minutes and seconds counter will continue in real time. If the set input is floating, the MM53110 will display normal time.

**Output Multiplex Operation:** Outputs from the appropriate internal counter are time division multiplexed to provide digit-sequential access to the data. Thus, instead of requiring 28 leads to interconnect a 4-digit clock and its display (7 segments per digit), only 11 outputs are required. Note that the MM53110 actually provides 12 outputs (4 digit-grid drive outputs plus 8 segment-anode drive outputs). The additional "segment" drive is provided to accommodate displays which feature a colon. The colon output is switched at 1/2 Hz rate with 1 1/2 seconds ON and 1/2 second OFF to provide a blinking colon as a short-time indication that the clock is operating. A zero-blanking circuit suppresses the zero that would otherwise sometimes appear in the tens-of-hours display. Each digit is sequentially enabled for a period of ~0.5 ms.

When time division multiplexing vacuum-fluorescent displays, it is necessary to inhibit the segment drive voltage(s) for a short time during inter-digit transitions. The MM53110 auto clock utilizes an interlaced output sequence and inter-digit blanking circuitry to prevent display ghosting problems. The digit sequence is: (1) digit no. 1 (unit minutes), (2) digit 3 (unit hours), (3) digit no. 2 (ten's of minutes), (4) digit no. 4 (ten's of hours). Both segment data and digit enables are blanked. *Figure 3* is a timing diagram which illustrates output timing for the MM53110.

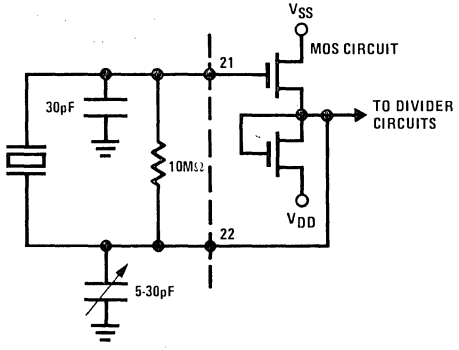



FIGURE 2. Crystal Oscillator

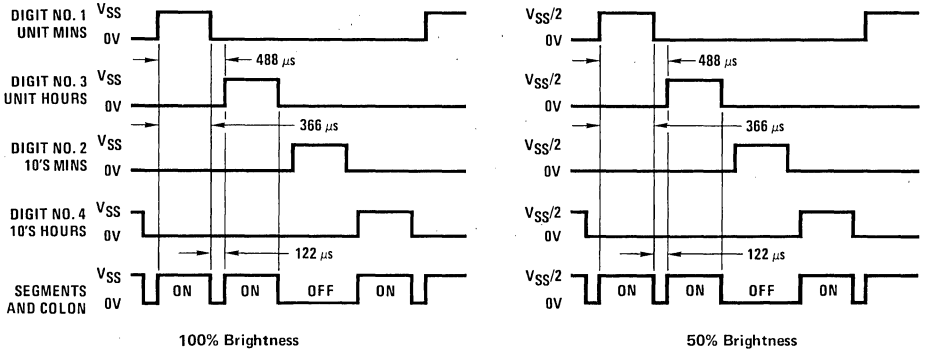



FIGURE 3. MM53110 Output Timing Diagram

## Functional Description (Continued)

**Brightness Control (Pin 14):** Since display brightness is a function of digit and segment voltage, a capability of directly driving the output logical high voltage results in a brightness control. See *Figure 6* for output voltage vs brightness control voltage. If the day-night input is switched to a logical low, the brightness input will be forced to $V_{SS}$, thus supplying $V_{SS}$ to the outputs for 100% brightness. If the day-night input is switched to a logical high, the display brightness will vary from 0% to 100% depending on the magnitude of the voltage applied to the brightness input. This is illustrated in *Figure 3*.

**Output Circuits (Pins 1–12):** All display output drivers, both digit and segment outputs, are push-pull drivers. The pull-ups are enhancement devices with sources common to the brightness input and the pull-downs are depletion devices with drains common to $V_{DD}$ *(Figure 4)*. Thus, all outputs are capable of sourcing and sinking the required display currents. *Figure 5* illustrates one method of interfacing these outputs to a vacuum-fluorescent display. *Figure 5* also shows the entire circuitry required for a complete automobile clock application.

**Pulse Output (Pin 13):** This output drives the external DC/DC converter circuitry with a frequency of 8192 Hz and a 25% duty cycle (31 $\mu$s at $V_{SS}$ and 91 $\mu$s at $V_{DD}$). The pulse output will start in a low state under any combination of battery and ignition voltages. If the ignition input is switched OFF ($V_{DD}$), the pulse output will go to $V_{DD}$, thus disabling the DC/DC converter.

*Figure 5* illustrates the interface of this output to the voltage doubler circuit.

**Ignition Input (Pin 16):** This input is provided with an internal 1 MΩ pull-down to $V_{DD}$ and switch debounce circuitry. If the ignition input is switched to a logical low, or left floating, all outputs will go to $V_{DD}$ and the time set and elapsed time inputs will be disabled. Since the pulse output goes to $V_{DD}$, the voltage doubler will become disabled and the battery voltage will be applied to $V_{SS}$. If the ignition input is switched to a logical high, all outputs and inputs will be enabled. The voltage doubler will also be enabled and will allow $V_{SS}$ to increase to the zener reference voltage.

**Day-Night Input (Pin 15):** This input is provided with debounce circuitry and a 1 MΩ pull-down to $V_{DD}$. If the day-night input is switched to a logical low, or left floating, the MM53110 will force the brightness input to $V_{SS}$, thus providing the maximum voltage to the display. If the day-night input is switched to a logical high, the digit and segment output high levels can be varied from 0V to $V_{SS}$ by varying the brightness input voltage from 0V to $V_{SS}$ respectively. The brightness input current will be the total of the digit and segment average currents.

**Elapsed Time Input (Pin 18):** If this input is switched to a logical low, the elapsed time counter will reset and display 0:00. If the input is switched to a logical high, the MM53110 will display elapsed time. If left floating, the MM53110 will display real time. During elapsed time display, the MM53110AA will display hours-minutes up to 19:59; the next count will be 0.00. The MM53110AB will display minutes-seconds up to 19:59, the next count will be 0:00. The elapsed time input also contains debounce circuitry.

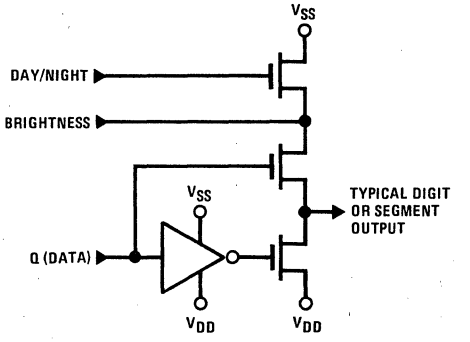



**FIGURE 4. Output Circuit**

## Functional Description (Continued)

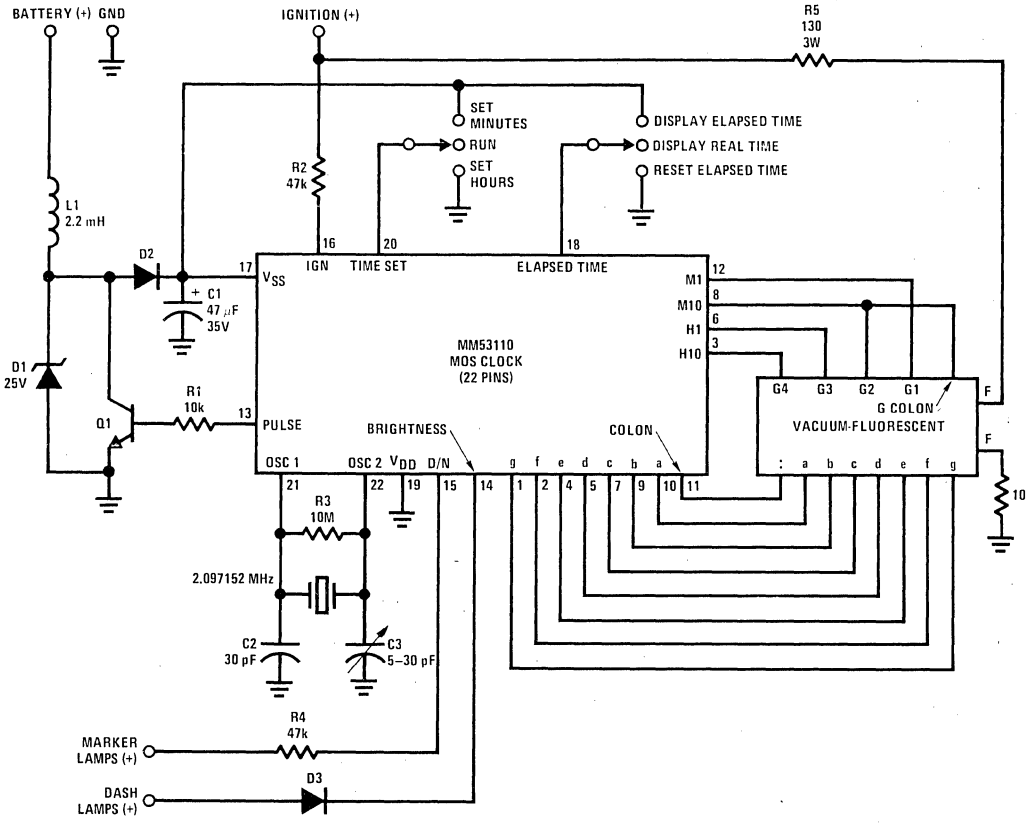



FIGURE 5. Typical Application for MM53110

National Semiconductor



# MM53113 Digital Alarm Clock

## General Description

The MM53113 digital alarm clock is a monolithic MOS integrated circuit utilizing P-channel low-threshold, enhancement mode and ion-implanted depletion mode devices. It provides all the logic required to build several types of clocks and timers. Four display modes (time, seconds, alarm and sleep) are provided to optimize circuit utility. The circuit interfaces directly with 7-segment fluorescent tubes, and requires only a single power supply. The timekeeping function operates from either a 50 or 60 Hz input, and the display format may be either 12 hours (with leading-zero blanking and AM/PM indication) or 24 hours. Outputs consist of display drives, sleep (e.g., timed radio turn off), and alarm enable. Power failure indication is provided to inform the user that incorrect time is being displayed. Setting the time cancels this indication. The device operates over a power supply range of 8–29V and does not require a regulated supply. The MM53113 is packaged in a 40-lead dual-in-line package.

## Features

- 50 or 60 Hz operation
- Single power supply
- Low power dissipation (36 mW at 9V)
- 12 or 24-hour display format
- AM/PM outputs } 12-hour format
- Leading-zero blanking }
- 24-hour alarm setting
- All counters are resettable
- Fast and slow set controls
- Power failure indication
- Blanking capability
- Elimination of illegal time display at turn on
- Direct interface to fluorescent tubes
- 9-minute snooze alarm
- Presettable 59-minute sleep timer

## Applications

- Alarm clocks
- Desk clocks
- Clock radios
- Automobile clocks
- Stopwatches
- Industrial clocks
- Portable clocks
- Photography timers
- Industrial timers
- Appliance timers
- Sequential controllers

## Block and Connection Diagrams

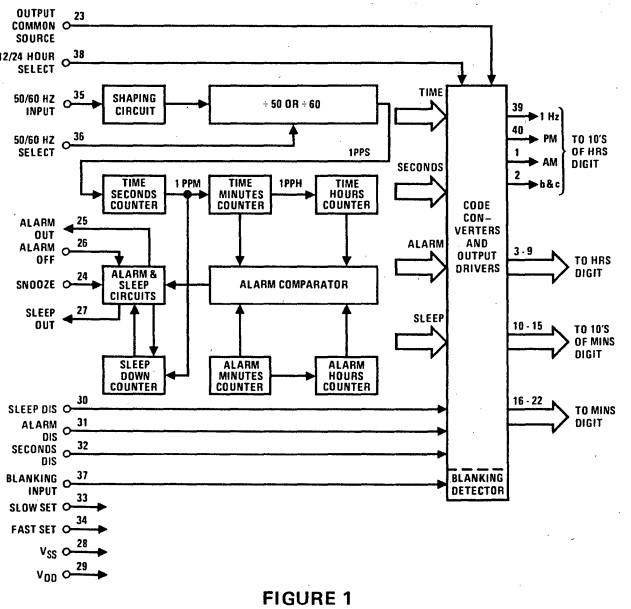
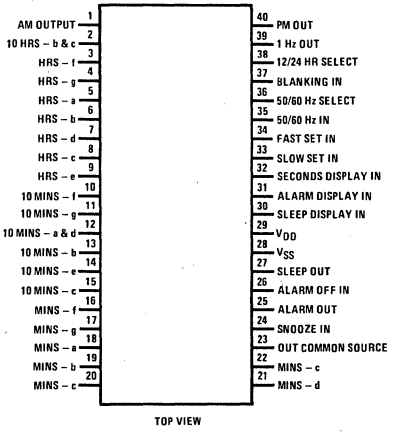



FIGURE 1

**Dual-In-Line Package**




TOP VIEW

**Order Number MM53113N**
**See Package 24**

FIGURE 2

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ + 0.3 to $V_{SS}$ − 30V |
| Operating Temperature | −25°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, $V_{SS}$ = 21V to +29V, $V_{DD}$ = 0V, unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage | $V_{SS}$ ($V_{DD}$ = 0V) | 21 | | 29 | V |
| Power Supply Current | No Output Loads | | | | |
| | $V_{SS}$ = 8V | | | 4 | mA |
| | $V_{SS}$ = 29V | | | 5 | mA |
| Counter Operation Voltage | (Note 2) | 8 | | 29 | V |
| 50/60 Hz Input Frequency Voltage | | dc | 50 or 60 | 10k | Hz |
| Logical High Level | | $V_{SS}$−1 | $V_{SS}$ | $V_{SS}$ | V |
| Logical Low Level | | $V_{DD}$ | $V_{DD}$ | $V_{DD}$+1 | V |
| Blanking Input Voltage | | | | | |
| Logical High Level | | $V_{SS}$−1 | $V_{SS}$ | $V_{SS}$ | V |
| Logical Low Level | | $V_{DD}$ | $V_{DD}$ | $V_{SS}$−4 | V |
| All Other Input Voltages | | | | | |
| Logical High Level | | $V_{SS}$−1 | $V_{SS}$ | $V_{SS}$ | V |
| Logical Low Level | Internal Depletion Device to $V_{DD}$ | $V_{DD}$ | $V_{DD}$ | $V_{DD}$+2 | V |
| Power Failure Detect Voltage | ($V_{SS}$ Voltage), (Note 1) | 1 | | 8 | V |
| Output Currents, | $V_{SS}$ = 21V to 29V, | | | | |
| 1 Hz Display | Output Common = $V_{SS}$ | | | | |
| Logical High Level | $V_{OH}$ = $V_{SS}$ − 2V | 1500 | | | μA |
| Logical Low Level, Leakage | $V_{OL}$ = $V_{DD}$ | | | 1 | μA |
| 10's of Hours (b & c), 10's of Minutes (a & d) | | | | | |
| Logical High Level | $V_{OH}$ = $V_{SS}$ − 2V | 1000 | | | μA |
| Logical Low Level, Leakage | $V_{OL}$ = $V_{DD}$ | | | 1 | μA |
| All Other Display, Alarm and Sleep Outputs | | | | | |
| Logical High Level | $V_{OH}$ = $V_{SS}$ − 2V | 500 | | | μA |
| Logical Low Level, Leakage | $V_{OL}$ = $V_{DD}$ | | | 1 | μA |

**Note 1:** The power fail detect voltage is 0.25V or more above the hold count voltage. The power-fail latch trips into power-fail mode at least 0.25V above the voltage at which data stored in the time latch is lost.

**Note 2:** Output drive capability is not guaranteed over the range of 8–21V.

## Functional Description

A block diagram of the MM53113 digital alarm clock is shown in *Figure 1*. The various display modes provided by this clock are listed in Table I. The functions of the setting controls are listed in Table II. *Figure 2* is a connection diagram. The following discussions are based on *Figure 1*.

**50 or 60 Hz Input (pin 35):** A shaping circuit *(Figure 3)* is provided to square the 50 or 60 Hz input. This circuit allows use of a filtered sinewave input. The circuit is a Schmitt Trigger that is designed to provide about 6V of hysteresis. A simple RC filter, such as shown in *Figure 6*, should be used to remove possible line-voltage transients that could either cause the clock to gain time or damage the device. The shaper output drives a counter chain which performs the timekeeping function.

**50 or 60 Hz Select Input (pin 36):** A programmable prescale counter divides the input line frequency by either 50 or 60 to obtain a 1 Hz time base. This counter is programmed to divide by 60 simply by leaving pin 36 unconnected; pull-down to $V_{DD}$ is provided by an internal depletion device. Operation at 50 Hz is programmed by connecting pin 36 to $V_{SS}$.

**Display Mode Select Inputs (pins 30–32):** In the absence of any of these three inputs, the display drivers present time-of-day information to the appropriate display digits. Internal pull-down depletion devices allow use of simple SPST switches to select the display mode. If more than one mode is selected, the priorities are as noted in Table I. Alternate display modes are selected by applying $V_{SS}$ to the appropriate pin. As shown in *Figure 1* the code converters receive time, seconds, alarm and sleep information from appropriate points in the clock circuitry. The display mode select inputs control the gating of the desired data to the code converter inputs and ultimately (via output drivers) to the display digits.

**Time Setting Inputs (pins 33 and 34):** Both fast and slow setting inputs are provided. These inputs are applied either singly or in combination to obtain the control functions listed in Table II. Again, internal pull-down depletion devices are provided; application of $V_{SS}$ to these pins effects the control functions. Note that the control functions proper are dependent on the selected display mode. For example, a hold-time control function is obtained by selecting seconds display and actuating the slow set input. As another example, the clock time may be reset to 12:00:00 AM, in the 12-hour format (00:00:00 in the 24-hour format), by selecting seconds display and actuating both slow and fast set inputs.

**Blanking Control Input (pin 37):** Connecting this Schmitt Trigger input to $V_{DD}$ places all display drivers in a non-conducting, high-impedance state, thereby inhibiting the display, (see *Figures 3 and 4*). Conversely, $V_{SS}$ applied to this input enables the display.

**Output Common Source Connection (pin 23):** All display output drivers are open-drain devices with all sources common to pin 23 *(Figure 4)*. When using fluorescent tube displays, $V_{SS}$ or a display brightness control voltage is permanently connected to this pin. Since the brightness of a fluorescent tube display is dependent on the anode (segment) voltage, applying a variable voltage to pin 23 results in a display brightness control. This control is shown in *Figure 6*.

**12 or 24-Hour Select Input (pin 38):** By leaving this pin unconnected, the outputs for the most-significant display digit (10's of hours) are programmed to provide a 12-hour display format. An internal depletion pull down device is again provided. Connecting this pin to $V_{SS}$ programs the 24-hour display format. Segment connections for 10's of hours in 24-hour mode are shown in *Figure 5b*.

**Power Fail Indication:** If the power to the integrated circuit drops indicating a momentary ac power failure and possible loss of clock, the power fail latch is set. The power failure indication consists of a flashing of the AM or PM indicator at a 1 Hz rate. A fast or slow set input resets an internal power failure latch and returns the display to normal. In the 24-hour format, the power failure indication consists of flashing segments "c" and "f" for times less than 10 hours, and of a flashing segment "c" for times equal to or greater than 10 hours but less than 20 hours; and a flashing segment "g" for times equal to or greater than 20 hours.

**Alarm Operation and Output (pin 25):** The alarm comparator *(Figure 1)* senses coincidence between the alarm counters (the alarm setting) and the time counters (real time). The comparator output is used to set a latch in the alarm and sleep circuits. The latch output enables the alarm output drive *(Figure 4)*, the MM53113 output that is used to control the external alarm sound generator. The alarm latch remains set for 59 minutes, during which the alarm will therefore sound if the latch output is not temporarily inhibited by another latch set by the snooze alarm input (pin 24) or reset by the alarm "OFF" input (pin 26). If power fail occurs and power comes back up, the alarm output will be in high impedance state.

**Snooze Alarm Input (pin 24):** Momentarily connecting pin 24 to $V_{SS}$ inhibits the alarm output for between 8 and 9 minutes, after which the alarm will again be sounded. This input is pulled-down to $V_{DD}$ by an internal depletion device. The snooze alarm feature may be repeatedly used during the 59 minutes in which the alarm latch remains set.

**Alarm "OFF" Input (pin 26):** Momentarily connecting pin 26 to $V_{SS}$ resets the alarm latch and thereby silences the alarm. This input is also returned to $V_{DD}$ by an internal depletion device. The momentary alarm "OFF" input also readies the alarm latch for the next comparator output, and the alarm will automatically sound again in 24 hours (or at a new alarm setting). If it is desired to silence the alarm for a day or more, the alarm "OFF" input should remain at $V_{SS}$.

**Sleep Timer and Output (pin 27):** The sleep output at pin 27 can be used to turn off a radio after a

## Functional Description (Continued)

desired time interval of up to 59 minutes. The time interval is chosen by selecting the sleep display mode (Table I) and setting the desired time interval (Table II). This automatically results in a current-source output via pin 27, which can be used to turn on a radio (or other appliance). When the sleep counter, which counts downwards, reaches 00 minutes, a latch is reset and the sleep output current drive is removed, thereby turning off the radio. The turn off may also be manually controlled (at any time in the countdown) by a momentary $V_{SS}$ connection to the snooze input (pin 24). The output circuitry is the same as the other outputs *(Figure 4)*.

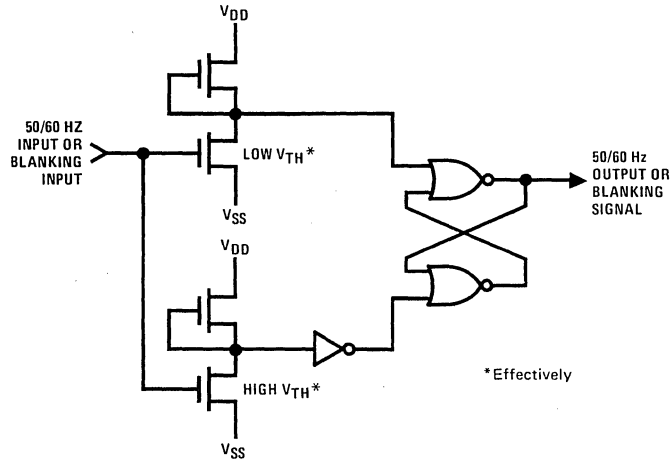
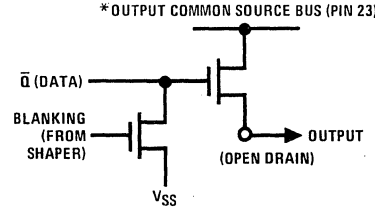
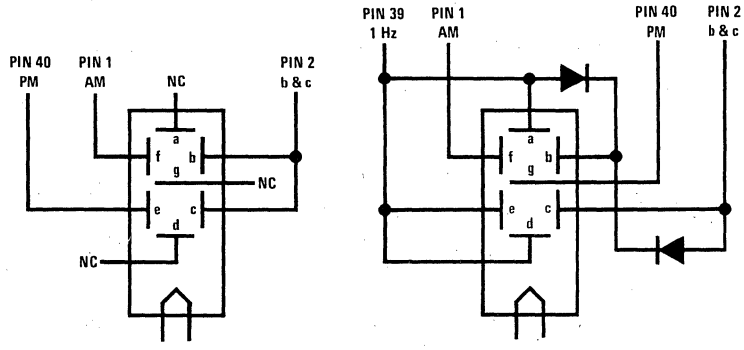



FIGURE 3. 50/60 Hz or Blanking Input Shaping Circuit




*Alarm and sleep output sources are connected to $V_{SS}$: blanking is not applied to these outputs.

FIGURE 4. Output Circuit




(a) 12-Hour Display Format (b) 24-Hour Display Format

FIGURE 5. Wiring Ten's-of-Hours Digit

## Functional Description (Continued)

TABLE I. MM53113 DISPLAY MODES

| *SELECTED DISPLAY MODE | DIGIT NO. 1 | DIGIT NO. 2 | DIGIT NO. 3 | DIGIT NO. 4 |
|---|---|---|---|---|
| Time Display | 10's of Hours & AM/PM | Hours | 10's of Minutes | Minutes |
| Seconds Display | Blanked | Minutes | 10's of Seconds | Seconds |
| Alarm Display | 10's of Hours & AM/PM | Hours | 10's of Minutes | Minutes |
| Sleep Display | Blanked | Blanked | 10's of Minutes | Minutes |

*If more than one display mode input is applied, the display priorities are in the order of Sleep (overrides all others), Alarm, Seconds, Time (no other mode selected).

TABLE II. MM53113 SETTING CONTROL FUNCTIONS

| SELECTED DISPLAY MODE | CONTROL INPUT | CONTROL FUNCTION |
|---|---|---|
| *Time | Slow | Minutes Advance at 2 Hz Rate |
| | Fast | Minutes Advance at 60 Hz Rate |
| | Both | Minutes Advance at 60 Hz Rate |
| Alarm | Slow | Alarm Minutes Advance at 2 Hz Rate |
| | Fast | Alarm Minutes Advance at 60 Hz Rate |
| | Both | Alarm Resets to 12:00 AM (12-hour format) |
| | Both | Alarm Resets to 00:00 (24-hour format) |
| Seconds | Slow | Input to Entire Time Counter is Inhibited (Hold) |
| | Fast | Seconds and 10's of Seconds Reset to Zero Without a Carry to Minutes |
| | Both | Time Resets to 12:00:00 AM (12-hour format) |
| | Both | Time Resets to 00:00:00 (24-hour format) |
| Sleep | Slow | Substracts Count at 2 Hz |
| | Fast | Substracts Count at 60 Hz |
| | Both | Substracts Count at 60 Hz |

*When setting time sleep minutes will decrement at rate of time counter, until the sleep counter reaches 00 minutes (sleep counter will not recycle).

## Typical Application

*Figure 6* is a schematic diagram of a general purpose alarm clock using the MM53113 and a fluorescent tube display.

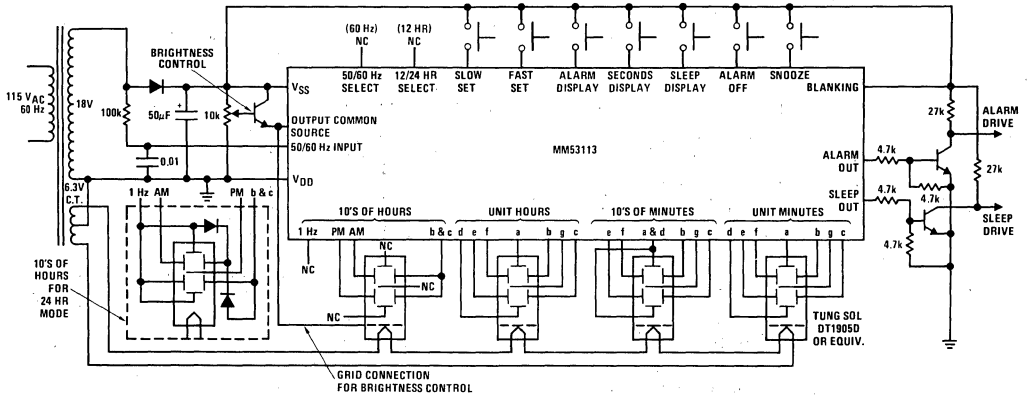



FIGURE 6

National Semiconductor

Digital Clocks

# MM53124 Automobile Clock and Elapsed Timer

## General Description

The MM53124 is a monolithic MOS integrated circuit utilizing P-channel low-threshold, enhancement mode and ion-implanted depletion mode devices. The circuit is designed for continuous direct-drive vacuum fluorescent displays. The display format is 12 or 24-hours for real time display and 24-hours for elapsed time display with leading zero blanking and colon indication. The timekeeping function operates from a 4 MHz crystal-controlled source. The circuit is available in a dual-in-line plastic, 40-pin package.

## Features

- 12-hour or 24-hour real time display format
- 24-hour elapsed time display
- Elapsed time rolls over from minutes-seconds display to hours-minutes display after accumulated time reaches 1 hour
- Elapsed time hold mode
- Time hold mode and seconds reset for easy synchronizing
- Crystal-controlled oscillator (4.194304 MHz)
- Push to read feature if ignition OFF
- Brightness control capability
- Day/night control
- DC to DC converter pulse output
- Direct interface to vacuum fluorescent displays
- Low standby power dissipation

## Applications

- Automobile clocks
- Desk clocks
- Elapsed timer

## Block and Connection Diagrams

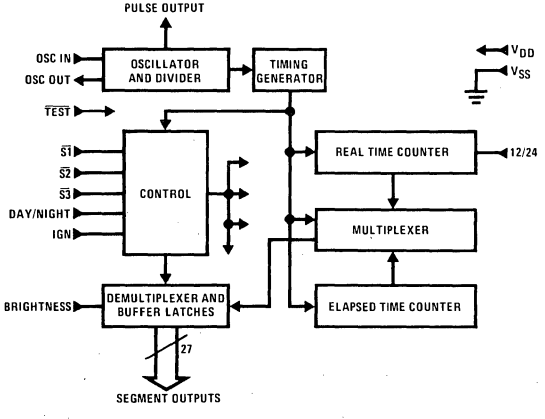
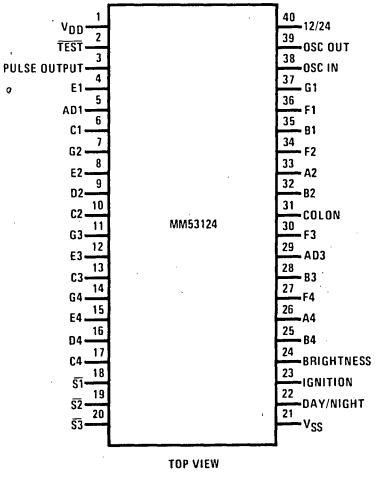



FIGURE 1

Dual-In-Line Package




TOP VIEW

Order Number MM53124N
See Package 24

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ + 0.3V to $V_{SS}$ − 26V |
| Operating Temperature | −40°C to +85°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, $V_{DD}$ = 0V unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| $V_{SS}$ Supply Voltage | No Loss of Memory | 5 | | | V |
| $V_{SS}$ Supply Voltage | Operational | 6 | | 25 | V |
| $I_{SS}$ Supply Current | No Output Load (@ 25°C) | | | 7.5 | mA |
| $\overline{S1}$, $\overline{S2}$, $\overline{S3}$ and 12/24 Hour Select Inputs | Input Pull-Down Current Source @ $V_{IN}$ = $V_{SS}$, ($I_{SINK}$) @ 25°C | 10 | | 100 | μA |
| | $V_{IL}$ Input Logical "0" | 0 | | 1 | V |
| | $V_{IH}$ Input Logical "1" | $V_{SS}$ − 1 | | $V_{SS}$ + 0.3 | V |
| Ignition Input and Day/Night Input | Input Pull-Down Current Source @ $V_{IN}$ = $V_{SS}$, ($I_{SINK}$) @ 25°C | 10 | | 100 | μA |
| | $V_{IL}$ Input Logical "0" | 0 | | 1.2 | V |
| | $V_{IH}$ Input Logical "1" | 5 | | $V_{SS}$ + 0.3 | V |
| AD1, AD3 Segments (Brightness = $V_{SS}$) | $V_{OL}$ = 0.5 x $V_{SS}$ ($I_{SINK}$) | 10 | | | μA |
| | $V_{OH}$ = $V_{SS}$ −1.5, $V_{SS} \geq$ 9V, ($I_{SOURCE}$) | 0.8 | | | mA |
| All Other Segments (Brightness = $V_{SS}$) | $V_{OL}$ = 0.5 x $V_{SS}$, ($I_{SINK}$) | 10 | | | μA |
| | $V_{OH}$ = $V_{SS}$ −1.5, $V_{SS} \geq$ 9V, ($I_{SOURCE}$) | 0.4 | | | mA |
| Pulse Output | Output Impedance, Logical "0" | 30 | | 120 | kΩ |
| | $V_{OH}$ = $V_{SS}$ − 2, Logical "1" | 1 | | | mA |
| OSC IN, OSC OUT | Oscillator Frequency | DC | 4.194304 | 4.2 | MHz |

## Functional Description

**Crystal Oscillator (Pins 38 and 39):** A quartz crystal, resonant at 4.194304 MHz, 2 capacitors and 1 resistor together with the internal MOS circuits form a crystal-controlled oscillator as shown in *Figure 3*. Varying the capacitor on OSC OUT (pin 39) allows precise frequency setting.

**Pulse Output (Pin 3):** In addition to being the crystal oscillator calibration reference, this output also drives the external DC/DC converter circuitry, if required. The frequency is 8192 Hz and the duty cycle is 75% at $V_{SS}$, 25% at $V_{DD}$. If the ignition input is switched OFF ($V_{DD}$), the pulse output will go to $V_{DD}$, thus disabling the DC/DC converter.

**Ignition Input (Pin 23):** This input is provided with an internal pull-down resistor to $V_{DD}$ and switch debounce circuitry. If the ignition input is switched low (towards $V_{DD}$) or left floating, all outputs will go to $V_{DD}$ and all inputs except for S2 are disabled. If S2 is connected to $V_{SS}$ at this time, the segment drivers will display time for as long as the S2 contact is made. When the pulse output goes to $V_{DD}$ as a result of ignition being turned OFF, the voltage doubler will become disabled and the battery voltage will be applied to $V_{SS}$.

## Functional Description (Continued)

**Segment Outputs (Pins 4-17, 25-37):** All segment output drivers are push-pull structures. The pull-ups are enhancement devices with sources common to the Brightness input. The pull-downs are depletion devices with drains common to $V_{DD}$. *Figure 2* shows a typical segment output circuit with 2 of such push-pull devices.

**Day/Night Input (Pin 22):** This input is provided with an internal pull-down resistor to $V_{DD}$ and switch debounce circuitry. If the Day/Night input is switched low (towards $V_{DD}$) or left floating, the MM53124 will force the Brightness input to $V_{SS}$ thus providing the maximum voltage to the display. If the Day/Night input is switched high (towards $V_{SS}$), the segment output drivers will track the input voltage at the Brightness input.

**Brightness Control (Pin 24):** When the Day/Night input is switched high (towards $V_{SS}$), the MM53124 is in the night mode and the segment outputs will track the voltage at the Brightness pin.

**12/24 Hour Input (Pin 40):** This input is provided with an internal pull-down resistor to $V_{DD}$. When this input is switched low (to $V_{DD}$) or left floating, real time will be displayed in the 24-hour mode. When switched high (to $V_{SS}$), real time will be displayed in the 12-hour mode.

**S1, S2, S3 Switches (Pins 18, 19, 20):** The MM53124 has 4 functional modes and these switches control the mode it is in. All 3 inputs are provided with internal pull-down resistors to $V_{DD}$ and input debounce circuitries. In addition to the 4 functional modes, the ignition input can also inhibit S1 and S3 and puts S2 in a push-to-read mode. When ignition is initially turned from OFF to ON, the clock will be in the display time mode 1. If minutes are incremented, seconds counter is reset and held until clock is returned to display time mode 1.

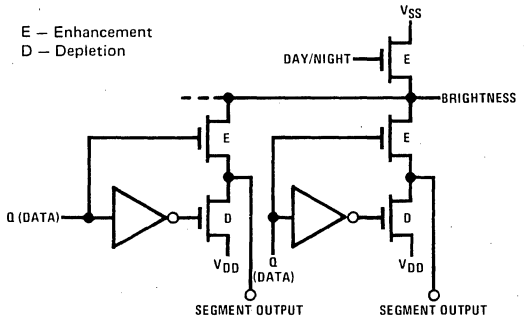



**FIGURE 2. Segment Output Circuit**

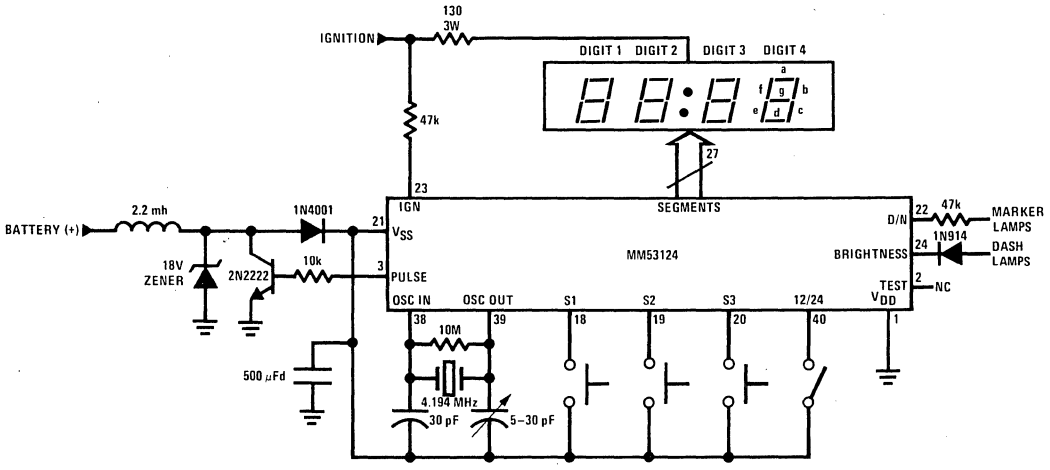



**FIGURE 3**

## Functional Description (Continued)

Tables I and II illustrate the different modes. Switch closure is connection to $V_{SS}$.

In the elapsed time mode, the display shows minutes-seconds until 1 hour has been accumulated. The display then rolls over to hours-minutes. The colon output after rollover will be continuously flashing at a 1/2 Hz rate. The duty cycle of the flashing colon is 75% ON and 25% OFF. The colon does not flash in the real time display mode.

**Test Sequence:** A test pin is supplied for testing purposes. This pin is normally low and raising it high toward $V_{SS}$ will put the IC in the test mode. In this mode, 128 Hz is output on the Pulse Output pin and internal real time and elapsed time counters are sped up by a factor of 8192. If the test pin and the ignition input pin are held at $V_{SS}$, a lamp test condition exists and the real time counter is reset to 12:00.

**TABLE I. IGNITION IS ON**

| NO. | MODE | SWITCH CLOSURE | SWITCH FUNCTION |
|---|---|---|---|
| 1 | Display Time | S1 | Changes to Set Hours Mode 2 |
| | | S2 | Changes to Elapsed Time Mode 4 |
| | | S3 | Does Nothing |
| 2 | Set Hour | S1 | Changes to Set Minutes Mode 3 |
| | | S2 | Changes to Elapsed Time Mode 4 |
| | | S3 | Increments Hours at 2 Hz Rate |
| 3 | Set Minutes | S1 | Changes to Display Time Mode 1 |
| | | S2 | Changes to Elapsed Time Mode 4 |
| | | S3 | Increments Minutes at 2 Hz Rate |
| 4 | Elapsed Time | S1 | Starts Elapsed Timer |
| | | S2 | Changes to Display Time Mode 1 |
| | | S3 | Stops and Holds Elapsed Timer |
| Any Mode 1–4 | | S1 and S2 Together | Changes to Elapsed Time Mode 4 and Resets Elapsed Timer |

**TABLE II. IGNITION IS OFF**

| SWITCH CLOSURE | SWITCH FUNCTION |
|---|---|
| S1 | No Action |
| S2 | Displays Real Time, Hours-Minutes for as Long as S2 is Closed |
| S3 | No Action |

National Semiconductor

Digital Clocks

# MM53224 Automobile Clock and Elapsed Timer

## General Description

The MM53224 is a monolithic MOS integrated circuit utilizing P-channel low-threshold, enhancement mode and ion-implanted depletion mode devices. The circuit is designed for continuous direct-drive vacuum fluorescent displays. The display format is 12 or 24-hours for real time display and 24-hours for elapsed time display with leading zero blanking and colon indication. The timekeeping function operates from a 4 MHz crystal-controlled source. The circuit is available in a dual-in-line plastic, 40-pin package.

## Features

- 12-hour or 24-hour real time display format
- 24-hour elapsed time display
- Elapsed time rolls over from minutes-seconds display to hours-minutes display after accumulated time reaches 1 hour
- Elapsed time hold mode
- Time hold mode and seconds reset for easy synchronizing
- Crystal-controlled oscillator (4.194304 MHz)
- Push to read feature if ignition OFF
- Brightness control capability
- Day/night control
- DC to DC converter pulse output
- Direct interface to vacuum fluorescent displays
- Low standby power dissipation

## Applications

- Automobile clocks
- Desk clocks
- Elapsed timer

## Block and Connection Diagrams

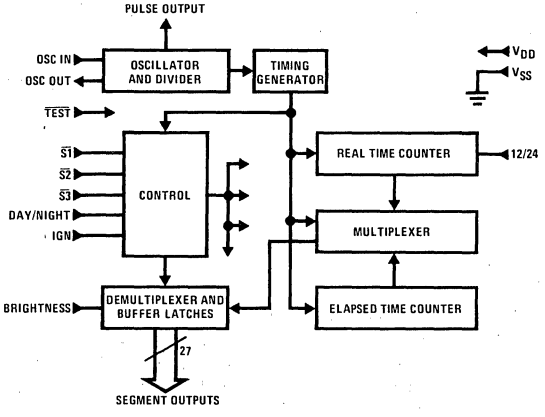
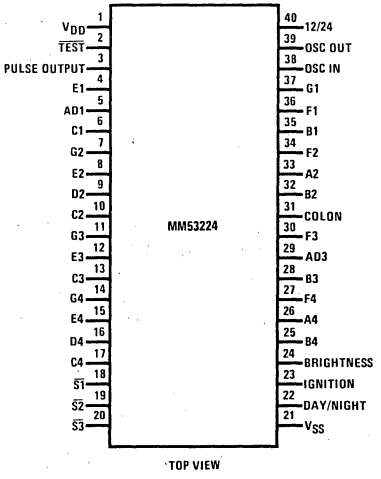



FIGURE 1

Dual-In-Line Package




TOP VIEW

Order Number MM53224N
See Package 24

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ + 0.3V to $V_{SS}$ − 26V |
| Operating Temperature | −40°C to +85°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, $V_{DD}$ = 0V unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| $V_{SS}$ Supply Voltage | No Loss of Memory | 5 | | | V |
| $V_{SS}$ Supply Voltage | Operational | 6 | | 25 | V |
| $I_{SS}$ Supply Current | No Output Load (@ 25°C) | | | 7.5 | mA |
| $\overline{S1}$, $\overline{S2}$, $\overline{S3}$ and 12/24 Hour | Input Pull-Down Current Source | | | | |
| Select Inputs | @ $V_{IN}$ = $V_{SS}$, ($I_{SINK}$) @ 25°C | 10 | | 100 | μA |
| | $V_{IL}$ Input Logical "0" | 0 | | 1 | V |
| | $V_{IH}$ Input Logical "1" | $V_{SS}$ − 1 | | $V_{SS}$ + 0.3 | V |
| Ignition Input and Day/Night | Input Pull-Down Current Source | | | | |
| Input | @ $V_{IN}$ = $V_{SS}$, ($I_{SINK}$) @ 25°C | 10 | | 100 | μA |
| | $V_{IL}$ Input Logical "0" | 0 | | 1.2 | V |
| | $V_{IH}$ Input Logical "1" | 5 | | $V_{SS}$ + 0.3 | V |
| AD1, AD3 Segments | $V_{OL}$ = 0.5 x $V_{SS}$ ($I_{SINK}$) | 10 | | | μA |
| (Brightness = $V_{SS}$) | $V_{OH}$ = $V_{SS}$ −1.5, $V_{SS} \geq$ 9V, ($I_{SOURCE}$) | 0.8 | | | mA |
| All Other Segments | $V_{OL}$ = 0.5 x $V_{SS}$, ($I_{SINK}$) | 10 | | | μA |
| (Brightness = $V_{SS}$) | $V_{OH}$ = $V_{SS}$ −1.5, $V_{SS} \geq$ 9V, ($I_{SOURCE}$) | 0.4 | | | mA |
| Pulse Output | Output Impedance, Logical "0" | 30 | | 120 | kΩ |
| | $V_{OH}$ = $V_{SS}$ − 2, Logical "1" | 1 | | | mA |
| OSC IN, OSC OUT | Oscillator Frequency | DC | 4.194304 | 4.2 | MHz |

## Functional Description

**Crystal Oscillator (Pins 38 and 39):** A quartz crystal, resonant at 4.194304 MHz, 2 capacitors and 1 resistor together with the internal MOS circuits form a crystal-controlled oscillator as shown in *Figure 3.* Varying the capacitor on OSC OUT (pin 39) allows precise frequency setting.

**Pulse Output (Pin 3):** In addition to being the crystal oscillator calibration reference, this output also drives the external DC/DC converter circuitry, if required. The frequency is 32,768Hz and the duty cycle is 50% at $V_{SS}$, 50% at $V_{DD}$. If the ignition input is switched OFF ($V_{DD}$), the pulse output will go to $V_{DD}$, thus disabling the DC/DC converter.

**Ignition Input (Pin 23):** This input is provided with an internal pull-down resistor to $V_{DD}$ and switch debounce circuitry. If the ignition input is switched low (towards $V_{DD}$) or left floating, all outputs will go to $V_{DD}$ and all inputs except for S2 are disabled. If S2 is connected to $V_{SS}$ at this time, the segment drivers will display time for as long as the S2 contact is made. When the pulse output goes to $V_{DD}$ as a result of ignition being turned OFF, the voltage doubler will become disabled and the battery voltage will be applied to $V_{SS}$.

## Functional Description (Continued)

**Segment Outputs (Pins 4-17, 25-37):** All segment output drivers are push-pull structures. The pull-ups are enhancement devices with sources common to the Brightness input. The pull-downs are depletion devices with drains common to $V_{DD}$. *Figure 2* shows a typical segment output circuit with 2 of such push-pull devices.

**Day/Night Input (Pin 22):** This input is provided with an internal pull-down resistor to $V_{DD}$ and switch debounce circuitry. If the Day/Night input is switched low (towards $V_{DD}$) or left floating, the MM53224 will force the Brightness input to $V_{SS}$ thus providing the maximum voltage to the display. If the Day/Night input is switched high (towards $V_{SS}$), the segment output drivers will track the input voltage at the Brightness input.

**Brightness Control (Pin 24):** When the Day/Night input is switched high (towards $V_{SS}$), the MM53224 is in the night mode and the segment outputs will track the voltage at the Brightness pin.

**12/24 Hour Input (Pin 40):** This input is provided with an internal pull-down resistor to $V_{DD}$. When this input is switched low (to $V_{DD}$) or left floating, real time will be displayed in the 24-hour mode. When switched high (to $V_{SS}$), real time will be displayed in the 12-hour mode.

**S1, S2, S3 Switches (Pins 18, 19, 20):** The MM53224 has 4 functional modes and these switches control the mode it is in. All 3 inputs are provided with internal pull-down resistors to $V_{DD}$ and input debounce circuitries. In addition to the 4 functional modes, the ignition input can also inhibit S1 and S3 and puts S2 in a push-to-read mode. When ignition is initially turned from OFF to ON, the clock will be in the display time mode 1. If minutes are incremented, seconds counter is reset and held until clock is returned to display time mode 1.

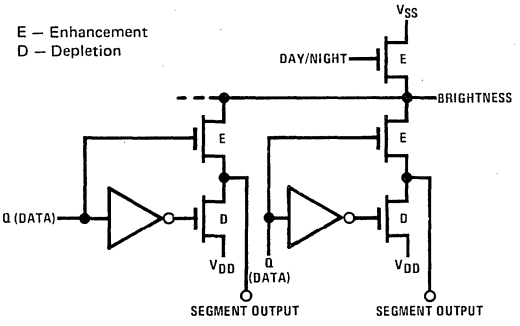



**FIGURE 2. Segment Output Circuit**

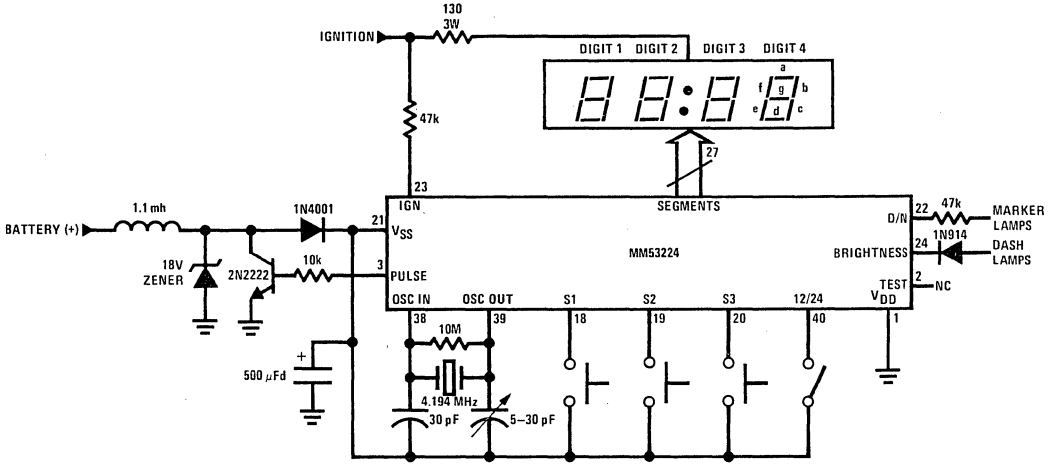



**FIGURE 3**

## Functional Description (Continued)

Tables I and II illustrate the different modes. Switch closure is connection to $V_{SS}$.

In the elapsed time mode, the display shows minutes-seconds until 1 hour has been accumulated. The display then rolls over to hours-minutes. The colon output after rollover will be continuously flashing at a 1/2 Hz rate. The duty cycle of the flashing colon is 75% ON and 25% OFF. The colon does not flash in the real time display mode.

**Test Sequence:** A test pin is supplied for testing purposes. This pin is normally low and raising it high toward $V_{SS}$ will put the IC in the test mode. In this mode, 512Hz is output on the Pulse Output pin and internal real time and elapsed time counters are sped up by a factor of 8192. If the test pin and the ignition input pin are held at $V_{SS}$, a lamp test condition exists and the real time counter is reset to 12:00.

**TABLE I. IGNITION IS ON**

| NO. | MODE | SWITCH CLOSURE | SWITCH FUNCTION |
|---|---|---|---|
| 1 | Display Time | S1 | Changes to Set Hours Mode 2 |
| | | S2 | Changes to Elapsed Time Mode 4 |
| | | S3 | Does Nothing |
| 2 | Set Hour | S1 | Changes to Set Minutes Mode 3 |
| | | S2 | Changes to Elapsed Time Mode 4 |
| | | S3 | Increments Hours at 2 Hz Rate |
| 3 | Set Minutes | S1 | Changes to Display Time Mode 1 |
| | | S2 | Changes to Elapsed Time Mode 4 |
| | | S3 | Increments Minutes at 2 Hz Rate |
| 4 | Elapsed Time | S1 | Starts Elapsed Timer |
| | | S2 | Changes to Display Time Mode 1 |
| | | S3 | Stops and Holds Elapsed Timer |
| Any Mode 1–4 | | S1 and S2 Together | Changes to Elapsed Time Mode 4 and Resets Elapsed Timer |

**TABLE II. IGNITION IS OFF**

| SWITCH CLOSURE | SWITCH FUNCTION |
|---|---|
| S1 | No Action |
| S2 | Displays Real Time, Hours-Minutes for as Long as S2 is Closed |
| S3 | No Action |

National Semiconductor



# MM5402, MM5405 Digital Alarm Clocks

## General Description

The MM5402, MM5405 digital alarm clocks are monolithic MOS integrated circuits utilizing N-channel low-threshold, enhancement mode and ion-implanted depletion mode devices. They provide all the logic required to build several types of clocks and timers with up to four display modes (time, seconds, alarm and sleep) to maximize circuit utility, but are specifically intended for clock-radio applications. Both devices will directly-drive 7-segment LED displays in either a 12-hour format (3 1/2 digits) with lead-zero blanking, AM/PM indication and flashing colon, or 24-hour format (4 digits) through hard-wire pin selection; the timekeeping function operates from either a 50 or 60 Hz input, also through pin selection. Outputs consist of display drivers, sleep (e.g., timed radio turn-off), and alarm enable. A power-fail indication mode is provided to inform the user of incorrect time display by flashing all "ON" digits at a 1 Hz rate, and is cancelled by simply resetting time. The device operates over a supply range of 7V–11V which does not require regulation.

The MM5405 is electrically identical to the MM5402, but with mirror-image pin-out to facilitate PC board layout when designing a "module" where the LED display and MOS chip are mounted on the same side; the MM5402 is more suited for "L" shaped module designs (vertical LED display, horizontal component board). Both devices are supplied in a 40-lead dual-in-line package.

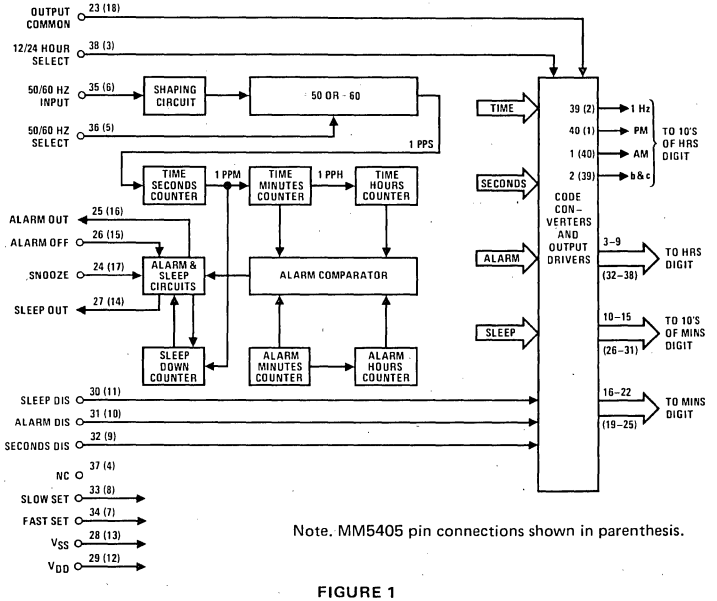
## Features

- 50 or 60 Hz operation
- Single power supply
- 12 or 24 hour display format
- AM/PM outputs } 12 hour format
- Leading-zero blanking }
- 24-hour alarm setting
- All counters are resettable
- Fast and slow set controls
- Power failure indication
- Elimination of illegal time display at turn "ON"
- Direct interface to LED displays
- 9-minute snooze alarm
- Presettable 59-minute sleep timer
- Available in standard (MM5402) or mirror-image (MM5405) pin-out

## Applications

- Alarm clocks
- Desk clocks
- Clock radios
- Automobile clocks
- Stopwatches
- Industrial clocks
- Portable clocks
- Photography timers
- Industrial timers
- Appliance timers
- Sequential controllers

## Block Diagram




Note. MM5405 pin connections shown in parenthesis.

FIGURE 1

## Absolute Maximum Ratings

| | | | |
|---|---|---|---|
| Voltage at Any Pin | $V_{SS}$ to $V_{SS}$ +12V | Lead Temperature (Soldering, 10 seconds) | 300°C |
| Operating Temperature | −25°C to +70°C | Segment Output Current | (Note 1) |
| Storage Temperature | −65°C to +150°C | | |

## Electrical Characteristics

$T_A$ within operating range, $V_{DD}$ = 7V to 11V, $V_{SS}$ = 0V, unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage | Output Driving Display | 9 | | 11 | V |
| | Functional Clock | 7 | | 11 | V |
| Power Supply Current | No Output Loads | | | | |
| | $V_{DD}$ = 7V | | | 4 | mA |
| | $V_{DD}$ = 11V | | | 5 | mA |
| 50/60 Hz Input | | | | | |
| Frequency | $V_{DD}$ = 7V to 11V | dc | 50 or 60 | 10k | Hz |
| Logical Low Level | | $V_{SS}$ | $V_{SS}$ | $V_{SS}$+0.5 | V |
| Logical High Level | | $V_{DD}$−3 | $V_{DD}$ | $V_{DD}$ | V |
| Input Leakage | | | | 100 | μA |
| All Other Input Voltages | | | | | |
| Logical Low Level | | $V_{SS}$ | $V_{SS}$ | $V_{SS}$+0.5 | V |
| Logical High Level | Internal Depletion Load to $V_{DD}$ | $V_{DD}$−3 | $V_{DD}$ | $V_{DD}$ | V |
| Power Failure Detect Voltage | ($V_{DD}$ Voltage), (Note 2) | 1 | | 5 | V |
| Count Operating Voltage | | 7 | | 11 | V |
| Hold Count Voltage | | (Note 2) | | 11 | V |
| Alarm and Sleep Outputs | $V_{DD}$ = 11V | | | | |
| Logical High, Source | $V_{OH}$ = $V_{SS}$ + 2 | 1 | | | μA |
| Logical Low, Sink | $V_{OL}$ = $V_{SS}$ + 2 | 5 | | | mA |
| Output Current Levels | $V_{DD}$ = 9V to 11V | | | | |
| Common Anode | *(Figure 5a)* | | | | |
| 10's of Hours (b & c), 10's of Minutes (a & d) | Output Common = $V_{SS}$ | | | | |
| Logical High Level, Leakage | $V_{OH}$ = $V_{DD}$ | | | 10 | μA |
| Logical Low Level, Sink | $V_{OL}$ = $V_{SS}$ + 2V | 24 | | | mA |
| 1 Hz Display | | | | | |
| Logical High Level, Leakage | $V_{OH}$ = $V_{DD}$ | | | 10 | μA |
| Logical Low Level, Sink | $V_{OL}$ = $V_{SS}$ + 2V | 36 | | | mA |
| All Other Segment Displays | | | | | |
| Logical High Level, Leakage | $V_{OH}$ = $V_{DD}$ | | | 10 | μA |
| Logical Low Level, Sink | $V_{OL}$ = $V_{SS}$ + 2V | 12 | | | mA |
| Output Current Levels | $V_{DD}$ = 9V to 11V | | | (Note 1) | |
| Common Cathode | *(Figure 5b)* | | | | |
| 10's of Hours (b & c), 10's of Minutes (a & d) | Output Common = $V_{SS}$ + 4 | | | | |
| Logical High Level, Source | $V_{OH}$ = $V_{SS}$ + 1.5V | 20 | | | mA |
| Logical Low Level, Leakage | $V_{OL}$ = $V_{SS}$ | | | 10 | μA |
| 1 Hz Display | | | | | |
| Logical High Level, Source | $V_{OH}$ = $V_{SS}$ + 1.5V | 30 | | | mA |
| Logical Low Level, Leakage | $V_{OL}$ = $V_{SS}$ | | | 10 | μA |
| All Other Segment Displays | | | | | |
| Logical High Level, Source | $V_{OH}$ = $V_{SS}$ + 1.5V | 10 | | | mA |
| Logical Low Level, Leakage | $V_{OL}$ = $V_{SS}$ | | | 10 | μA |

**Note 1:** Segment output current must be limited to 15 mA maximum by user; power dissipation must be limited to 900 mW at 70°C and 1.2W at 25°C.

**Note 2:** The power-fail detect voltage is 0.25V or more above the hold count voltage. The power-fail latch trips into power-fail mode at least 0.25V above the voltage at which data stored in the time latch is lost.

**Note 3:** Power supply voltage should not exceed a maximum voltage of 12V under any circumstances, such as during plug in, power up, display "ON"/"OFF", or power supply ripple. Doing so runs the risk of permanently damaging the device.

## Functional Description

A block diagram of the MM5402, MM5405 digital clock radio circuit is shown in *Figure 1*. The various display setting modes are listed in Table I, and Table II shows the setting control functions. The following description is based on *Figure 1* and refers to both devices as they are electrically identical.

**50 or 60 Hz Input:** A shaping circuit *(Figure 3)* is provided to square the 50 or 60 Hz input. This circuit allows use of a filtered sinewave input. The circuit is a Schmitt trigger that is designed to provide about 0.8V hysteresis. A simple RC filter such as shown in *Figure 7*, should be used to remove possible line-voltage transients that could either cause the clock to gain time or damage the device. The shaper output drives a counter chain which performs the timekeeping function.

**50 or 60 Hz Select Input:** A programmable prescale counter divides the input line frequency by either 50 or 60 to obtain a 1 Hz time base. This counter is programmed to divide by 60 simply by leaving 50/60 Hz select unconnected; pull-up to $V_{DD}$ is provided by an internal depletion load. Operation at 50 Hz is programmed by connecting 50/60 Hz select to $V_{SS}$.

**Display Mode Select Inputs:** In the absence of any of these three inputs, the display drivers present time-of-day information to the appropriate display digits. Internal depletion pull-up devices allow use of simple SPST switches to select the display mode. If more than one mode is selected, the priorities are as noted in Table I. Alternate display modes are selected by applying $V_{SS}$ to the appropriate pin. As shown in *Figure 1* the code converters receive time, seconds, alarm and sleep information from appropriate points in the clock circuitry. The display mode select inputs control the gating of the desired data to the code converter inputs and ultimately (via output drivers) to the display digits.

**Time Setting Inputs:** Both fast and slow setting inputs are provided. These inputs are applied either singly or in combination to obtain the control functions listed in Table II. Again, internal depletion pull-up devices are provided; application of $V_{SS}$ to these pins affects the control functions. Note that the control functions proper are dependent on the selected display mode. For example, a hold-time control function is obtained by selecting seconds display and actuating the slow set input. As another example, the clock time may be reset to 12:00:00 AM, by selecting seconds display and actuating both slow and fast set inputs.

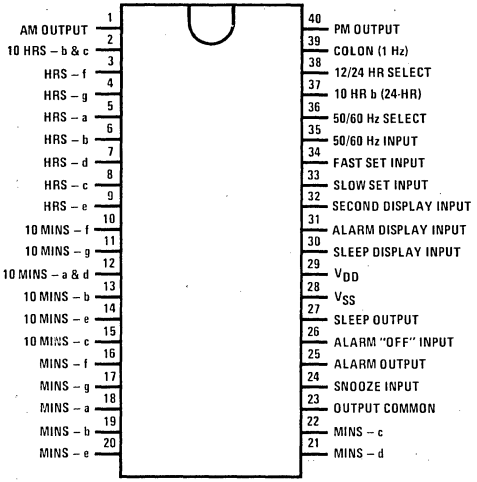
**Output Common:** All display output drivers are open drain devices with all the sources connected to output common pin. This pin can be used as a common source or a common drain. When used as a common source, this pin is connected to $V_{SS}$ and when used as a common drain, this pin is connected to $V_{DD}$. This allows the use of either common anode or common cathode LED's for displays. *Figure 5* shows these connections.

**12 or 24 Hour Select Input:** By leaving this pin unconnected, the outputs for the most-significant display digit (10's of hours) are programmed to provide a 12-hour display format. An internal depletion pull-up device is again provided. Connecting this pin to $V_{SS}$ programs the 24-hour display format. Segment connections for 10's of hours in 24-hour mode are shown in *Figure 6*.

**Power Fail Indication:** If the power to the integrated circuit drops, indicating a momentary ac power failure and possible loss of clock, all "ON" segments will flash at 1 Hz rate. A fast or slow set input resets an internal power failure latch and returns the display to normal.

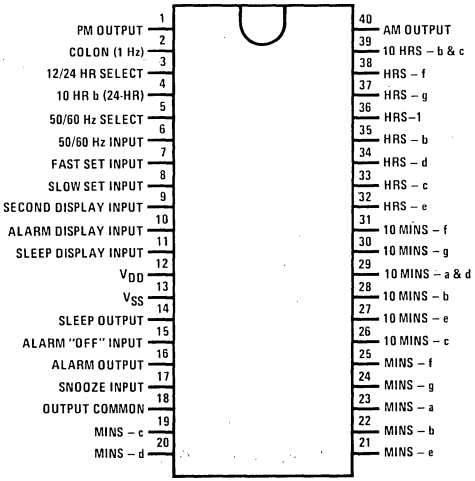
## Connection Diagrams (Top Views)

Dual-In-Line Package




Order Number MM5402N
See Package 24

Dual-In-Line Package




Order Number MM5405N
See Package 24

## Functional Description (Continued)

**Alarm Operation and Output:** The alarm comparator *(Figure 1)* senses coincidence between the alarm counters (the alarm setting) and the time counters (real time). The comparator output is used to set a latch in the alarm and sleep circuits. The latch output enables the alarm output driver *(Figure 4b)* which is used to control the external alarm sound generator. The alarm latch remains set for 59 minutes, during which the alarm will therefore sound if the latch output is not temporarily inhibited by another latch set by the snooze alarm input or reset by the alarm "OFF" input.

**Snooze Alarm Input:** Momentarily connecting snooze to $V_{SS}$ inhibits the alarm output for between 8 and 9 minutes, after which the alarm will again be sounded. This input is pulled-up to $V_{DD}$ by an internal depletion device. The snooze alarm feature may be repeatedly used during the 59 minutes in which the alarm latch remains set.

**Alarm "OFF" Input:** Momentarily connecting alarm "OFF" to $V_{SS}$ resets the alarm latch and thereby silences the alarm. This input is also returned to $V_{DD}$ by an internal depletion device. The momentary alarm "OFF" input also readies the alarm latch for the next comparator output, and the alarm will automatically sound again in 24 hours (or at a new alarm setting). If it is desired to silence the alarm for a day or more, the alarm "OFF" input should remain at $V_{SS}$.

**Sleep Timer and Output:** The sleep output can be used to turn "OFF" a radio after a desired time interval of up to 59 minutes. The time interval is chosen by selecting the sleep display mode, (Table I) and setting the desired time interval (Table II). This automatically results in a current sink output which can be used to turn "ON" a radio (or other appliance). When the sleep counter, which counts downwards, reaches 00 minutes, a latch is reset and the sleep output current drive is removed, thereby turning "OFF" the radio. This turn "OFF" may also be manually controlled (at any time in the countdown) by a momentary $V_{SS}$ connection to the Snooze input. The output circuitry is the same as the other outputs *(Figure 4b)*.

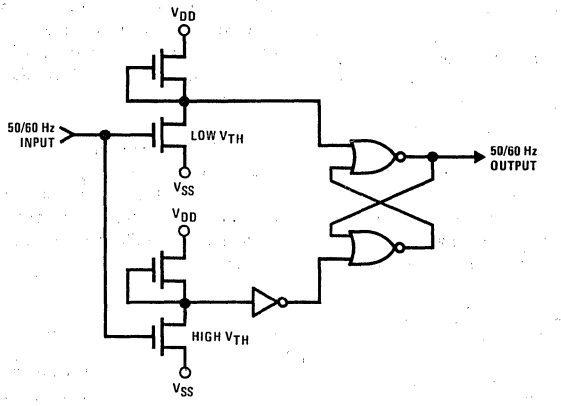



FIGURE 3. 50/60 Hz Input Shaping Circuit

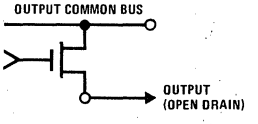



FIGURE 4(a). Segment Outputs

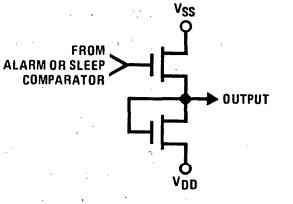



FIGURE 4(b). Alarm and Sleep Outputs

## Functional Description (Continued)

**TABLE I. MM5402, MM5405 Display Modes**

| *SELECTED DISPLAY MODE | DIGIT NO. 1 | DIGIT NO. 2 | DIGIT NO. 3 | DIGIT NO. 4 |
|---|---|---|---|---|
| Time Display | 10's of Hours & AM/PM | Hours | 10's of Minutes | Minutes |
| Seconds Display | Blanked | Minutes | 10's of Seconds | Seconds |
| Alarm Display | 10's of Hours & AM/PM | Hours | 10's of Minutes | Minutes |
| Sleep Display | Blanked | Blanked | 10's of Minutes | Minutes |

* If more than one display mode input is applied, the display priorities are in the order of Sleep (overrides all others), Alarm, Seconds, Time (no other mode selected).

**TABLE II. MM5402, MM5405 Setting Control Functions**

| SELECTED DISPLAY MODE | CONTROL INPUT | CONTROL FUNCTION |
|---|---|---|
| *Time | Slow | Minutes Advance at 2 Hz Rate |
| | Fast | Minutes Advance at 60 Hz Rate |
| | Both | Minutes Advance at 60 Hz Rate |
| Alarm | Slow | Alarm Minutes Advance at 2 Hz Rate |
| | Fast | Alarm Minutes Advance at 60 Hz Rate |
| | Both | Alarm Resets to 12:00 AM (Midnight) (12-Hour Format) |
| | Both | Alarm Resets to 00:00 (24-Hour Format) |
| Seconds | Slow | Input to Entire Time Counter is Inhibited (Hold) |
| | Fast | Seconds and 10's of Seconds Reset to Zero Without a Carry to Minutes |
| | Both | Time Resets to 12:00:00 AM (Midnight) (12-Hour Format) |
| | Both | Time Resets to 00:00:00 (24-Hour Format) |
| Sleep | Slow | Subtracts Count at 2 Hz |
| | Fast | Subtracts Count at 60 Hz |
| | Both | Subtracts Count at 60 Hz |

*When setting time sleep minutes will decrement at rate of time counter, until the sleep counter reaches 00 minutes (sleep counter will not recycle).

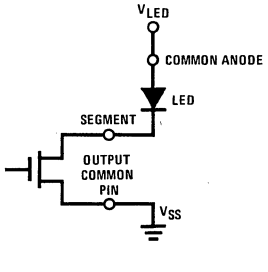



**FIGURE 5(a). Common Anode Application**

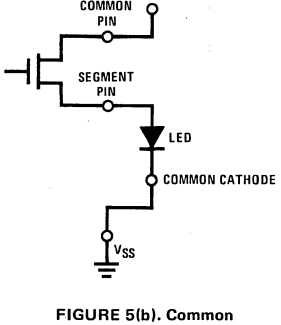



**FIGURE 5(b). Common Cathode Application**

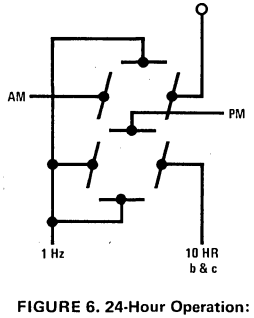



**FIGURE 6. 24-Hour Operation: 10's of Hours Digit Connections**

National Semiconductor



# MM5406 Deluxe Display and Clock Radio

## General Description

The MM5406 display and clock radio is a monolithic MOS integrated circuit utilizing N-channel low-threshold, enhancement mode and ion-implanted depletion mode devices. The MM5406 circuit interfaces easily with 4 digit LED displays. The MM5406 will also display alpha-numeric data originating from 16 possible Mini-DIP add-on circuits. One of these circuits is the MM5407 digital thermometer circuit for a time/temperature clock. Data communications will be via one common I/O bus. Time information is transmitted from this chip to the peripheral circuits.

## Features

- Capability for 16 Mini-DIP add-on circuits
- Single line communications data bus
- Priority display system
- Alpha-numeric display
- 50 or 60 Hz operation
- 12 or 24-hour
- Up/down, fast-slow and hour-minute set control
- Power-ON reset to 12:00:00 (time), 12:00 (alarm), 59 (sleep)
- Alarm and PM indication
- Duplexed display
- Power failure indication
- Brightness control
- Presettable 59-minute sleep timer
- Alarm display
- Seconds display
- 9-minute snooze alarm
- Blinking or constant colon
- Alarm tone output
- Stand-by oscillator for power failure

## Applications

- Sequential controller
- Alarm clock
- Desk clock
- Indoor-outdoor thermometer and clock
- Elapsed timer
- Industrial and military clock
- Humidity display
- Portable clock

## Block Diagram

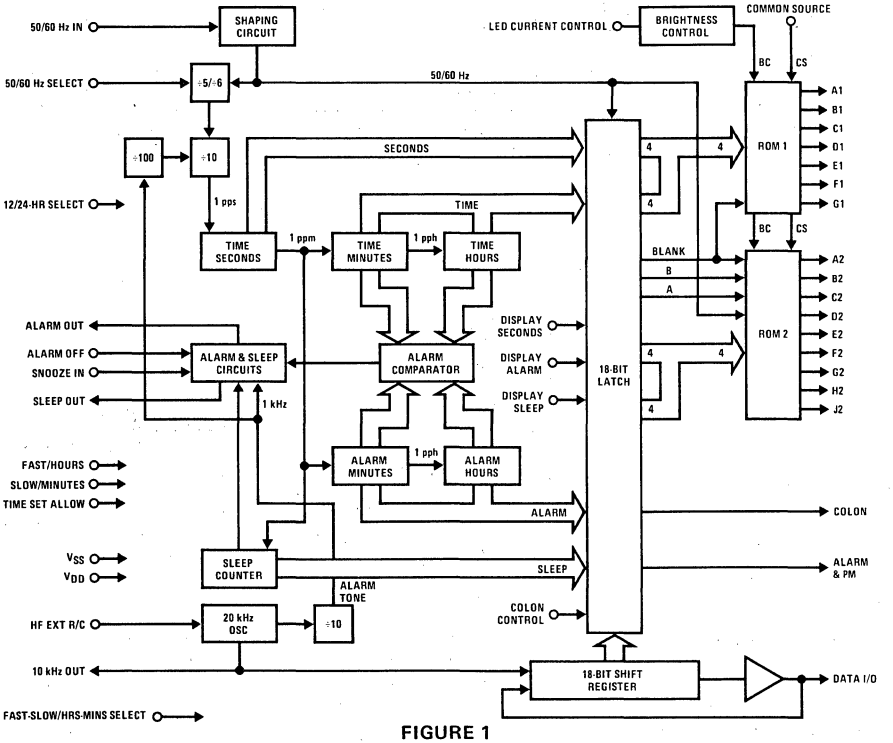
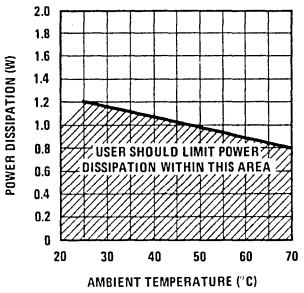



FIGURE 1

## Absolute Maximum Ratings

| | |
|---|---|
| Maximum Voltage at Any Pin | $V_{SS}$ to $V_{SS}$ + 12V |
| Operating Temperature | −25°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

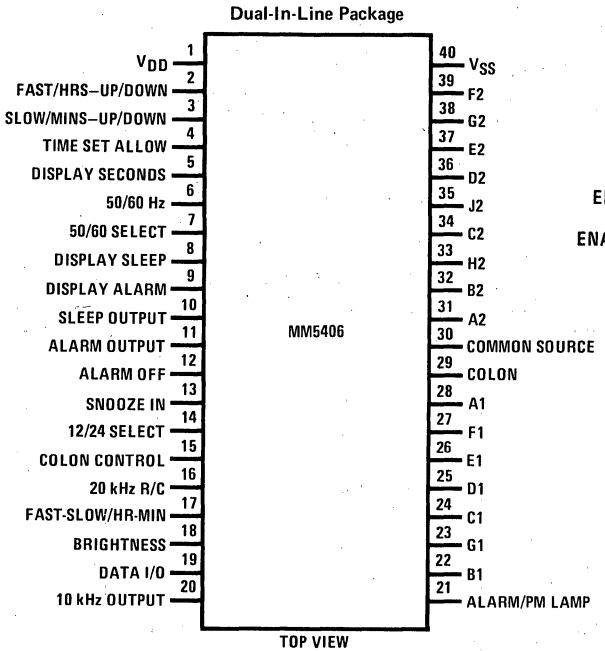



## Electrical Characteristics ($T_A$ within operating range) Functional voltage, $V_{SS}$ = 0, $V_{DD}$ = 9V to 11V

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage | | 9.0 | | 11 | V |
| Power Supply Current | Exclude LED Current | | | | |
| | $V_{DD}$ = 9V | | | 10 | mA |
| | $V_{DD}$ = 11V | | | 13 | mA |
| 50/60 Hz Input | Logic "0" | −0.3 (Note 2) | | 0.5 | V |
| | Logic "1" | 2.0 | | $V_{DD}$ | V |
| Fast/Hours–Up/Down Input | Logic "0" (Source, 1 mA) | 0 | | 0.5 | V |
| Slow/Mins–Up/Down Input | Logic "1" (Sink, 1 mA) | $V_{DD}$−0.5 | | $V_{DD}$ | V |
| Time Set Allow Input | | | | | |
| Display Seconds Input | | | | | |
| 50/60 Hz Select Input | | | | | |
| Display Sleep Input | | | | | |
| Display Alarm Input | Logic "0" (Source, 20 μA) | 0 | | 0.5 | V |
| Alarm OFF Input | Logic "1" (Leakage, 10 μA) | 3.0 | | $V_{DD}$ | V |
| Snooze Input | | | | | |
| 12/24-Hour Select Input | | | | | |
| Colon Control Input | | | | | |
| Fast-Slow/Hr-Min Input | | | | | |
| Sleep Out | Logic "0" (Sink, 5 mA) | | | 2.0 | V |
| | Logic "1" (Source, 10 μA) | $V_{DD}$−1.0 | | $V_{DD}$ | V |
| Alarm Out | Logic "0" (Sink, 5 mA) | | | 2.0 | V |
| | Logic "1" (Source, 10 μA) | $V_{DD}$−1.0 | | $V_{DD}$ | V |
| Data I/O | Logic "0" (Sink, 100 μA) | | | 1.5 | V |
| | Logic "1" (Source, 100 μA) | $V_{DD}$−4.0 | | $V_{DD}$ | V |
| 10 kHz Output | Logic "0" (Sink, 3 mA) | | | 1.5 | V |
| | Logic "1" (Source, 200 μA) | $V_{DD}$−4.0 | | $V_{DD}$ | V |
| LED Current Control | Input Current Source | 0 | | 1.5 | mA |
| All Outputs For LED Drive (Note 1) | Output Current at 1.5V | | | 30 | mA |

**Note 1:** Segment output current must be limited to 30 mA by the user; power dissipation must be limited to 800 mW at 70°C and 1.2W at 25°C.
**Note 2:** Applies to voltages directly at pin. See application in *Figure 4* for recommended configuration.

## Functional Description




Order Number MM5406N
See Package 24

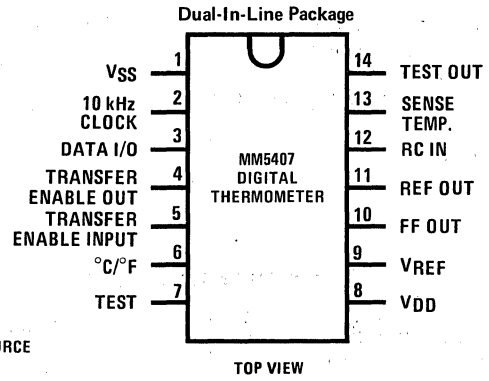



FIGURE 2

A block diagram of the MM5406 is shown in *Figure 1*. The various display modes and their priorities are listed in Table I. The functions of the setting controls in combination with the selected display mode are listed in Table II. *Figure 2* is a connection diagram and one add-on circuit. The following discussions are based on *Figure 1*.

**50 or 60 Hz Input (Pin 6):** A shaping circuit is provided to square the 50 or 60 Hz input. A simple RC filter, such as shown in *Figure 4*, should be used to remove possible line voltage transients that could cause the clock to gain time or cause damage to the clock. The shaper circuit drives a counter chain which performs the timekeeping function.

**50 or 60 Hz Select Input (Pin 7):** A programmable prescale counter divides the input line frequency by either 50 or 60 to obtain a 1 pps time base. This counter is programmed to divide by 60 by simply leaving pin (7) unconnected. A pull-up to $V_{DD}$ is provided by an internal depletion load. 50 Hz operation is programmed by connecting pin (7) to $V_{SS}$.

**Alarm, Sleep, and Seconds Display Inputs (Pins 9, 8, and 5):** In the absence of any of these inputs (i. e. pin open) the display drivers present time-of-day information to the appropriate display digits. All 3 inputs have internal pull-up depletion loads to $V_{DD}$. Connection of any combination of the 3 inputs to $V_{SS}$ results in 1-of-5 display modes whose priorities and functions are listed in Table I. Note that display sleep (pin 8) and display alarm (pin 9) have equal priorities and when both pins are connected to $V_{SS}$, all of the display drivers are turned on, providing a lamp test mode.

**Time Setting Inputs (Pins 2, 3 and 17):** Both fast-slow and hours-minutes (up/down) setting inputs are provided. These inputs are applied either singly or in combination to obtain the control functions listed in Table II. Fast/hours (pin 2) and slow/minutes (pin 3) are both 3-level inputs with internal debounce circuitry. The 3 states are: $V_{SS}$, $V_{DD}$, and open. The fast-slow/hours-minutes select input (pin 17) is a 2-level input. When pin 17 is connected to $V_{SS}$, the fast-slow set mode will be selected. When pin 17 is connected to $V_{DD}$, the hours-minutes set mode will be selected.

When either set input (pin 2 or pin 3) is connected to $V_{DD}$, the appropriate counters will count up. When connected to $V_{SS}$, the counters will count down. Note that the control functions are dependent on the selected display mode. For one example, a time reset function to 12:00:00 may be made by selecting either time or seconds display mode and connecting pins 2 and 3 to $V_{DD}$ or $V_{SS}$ (provided the time set allow input is applied). However, if sleep is the selected display mode, the sleep counter is displayed, and reset to 59 minutes.

**10 kHz Output (Pin 20):** The 10 kHz output of the mother chip (MM5406) provides the data clock for each of the daughter chips. This clock is used for transmitting and receiving serial data over the data I/O bus line. All daughter chips will accept data from the data I/O output (pin 19) during the high state ($V_{DD}$), and will output data on the low state ($V_{SS}$). The MM5406 will accept data during the low state and will output data on the high state. See *Figure 5* for the data I/O timing.

## Functional Description (Continued)

**Data I/O (Pin 19):** The data I/O pin is used for transmitting and receiving serial data via the common data bus line. During the transmit mode, the MM5406 sends out 16 bits of real-time data (H10, H1, M10, and M1 from time counter) followed by a select code containing 4 bits for selecting one of the 16 daughter chips. After the select code is sent out, 4 status bits are sent. These bits represent the status of the fast and slow set inputs, the 10 Hz internal signal, and the one pulse-per-day signal. If the daughter chip for that select code is not enabled, or if that daughter chip is not used, the MM5406 will then send out the next higher select code. Whenever a daughter chip is enabled, it will deselect the select code and send 20 bits of data to the MM5406. This 20-bit data word contains 4 digits of 4 bits each, 2 special bits used for Alpha information, and 2 bits for enabling the PM DOT and COLON. If the daughter circuit is constantly enabled, the MM5406 will send the same select code on each cycle until the daughter circuit is disabled. If the MM5406 data I/O pin is left open (daughter circuits disabled or not used), the circuit will then display

**TABLE I. MM5406 DISPLAY MODES**

| SELECTED DISPLAY MODES (Note 1) | DIGIT NO. 1 | DIGIT NO. 2 | DIGIT NO. 3 | DIGIT NO. 4 |
|---|---|---|---|---|
| Time Display | Time 10's Hrs, AM/PM | Time Hours | Time 10's Mins | Time Mins, Alarm Set Indicator |
| Seconds Display | Time 10's Mins | Time Mins | Time 10's Secs | Time Seconds |
| Alarm Display | Alarm 10's Hrs, AM/PM | Alarm Hours | Alarm 10's Mins | Alarm Mins, Alarm Set Indicator |
| Sleep Display | Blanked | Blanked | Sleep 10's Mins | Sleep Minutes |
| Alarm and Sleep | Lamp Test | Lamp Test | Lamp Test | Lamp Test |

**Note 1:** If more than one display mode input is applied, the display priorities are in the order of alarm *or* sleep, seconds, then time. Alarm and sleep have equal priority over seconds; however, when both alarm and sleep are applied, all outputs are ON, providing a lamp test. This display mode has priority above all others.

**TABLE II. MM5406 CONTROL SETTING FUNCTIONS**

| SELECTED DISPLAY MODE | CONTROL INPUT | PIN 17 | CONTROL FUNCTION |
|---|---|---|---|
| Time and Seconds Display | Time Set Allow and Slow Set | $V_{SS}$ | Minutes advance at 2 Hz and seconds counter resets to zero. |
| | Time Set Allow and Fast Set | $V_{SS}$ | Minutes advance at 60 Hz and seconds counter resets to zero. |
| | Time Set Allow and Minutes Set | $V_{DD}$ | Minutes advance at 2 Hz and seconds counter resets to zero. Hours counter in hold mode. |
| | Time Set Allow and Hours Set | $V_{DD}$ | Hours advance at 2 Hz. Minutes and seconds in normal count mode. |
| | Time Set Allow and Fast and Slow | Don't Care | Hours, minutes, and seconds are reset to zero (12:00). |
| Alarm Display | Slow Set | $V_{SS}$ | Alarm minutes counter advances at 2 Hz. |
| | Fast Set | $V_{SS}$ | Alarm minutes counter advances at 60 Hz. |
| | Minutes Set | $V_{DD}$ | Alarm minutes counter advances at 2 Hz. Alarm hours in hold mode. |
| | Hours Set | $V_{DD}$ | Alarm hours counter advances at 2 Hz. Alarm minutes in hold mode. |
| | Fast and Slow Set | Don't Care | Alarm minutes and hours counters are reset to zero (12:00). |
| Sleep Display* | Slow | | Sleep counter is decremented at a 2 Hz rate. |
| | Fast | | Sleep counter is decremented at a 10 Hz rate. |
| | Fast and Slow | | Sleep counter is reset to 59 minutes. |
| Sleep and Alarm Display | All outputs are driven to provide a lamp test. | | |

*Only when contents of sleep counter are zero *and* sleep is the selected display mode, sleep counter is set to 59 minutes.

Note. Alarm and time counters will also count down at the same rates specified above.

## Functional Description (Continued)

its own time, alarm, and sleep counters. If sleep, seconds, or alarm is selected, the MM5406 will disable the data I/O and display the selected counter. The MM5406 has an internal pull-up to $V_{DD}$, therefore, all daughter chips will have open drain outputs.

**Time Set Allow Input (Pin 4):** This input is used to enable fast-slow or hours-minutes setting of time when the selected display mode is time or seconds. An internal pull-up depletion load is provided on the input. To set time, one must connect pin 4 to $V_{SS}$ in combination with pin 2 and/or pin 3 (provided time or seconds is the selected display mode). Note when the selected display mode is alarm or sleep, the time set allow input does not inhibit alarm or sleep setting. Time set allow must be applied before pins 2 and 3, and must be released after releasing pins 2 and 3.

**12/24-Hour Select Input (Pin 14):** This input is used to select between 12 and 24-hour output display formats. An internal depletion pull-up is provided on the input. If left open, the 12-hour format is selected. If connected to $V_{SS}$, the 24-hour format is selected.

**Colon Control Input (Pin 15):** This input is used to select between a flashing or non-flashing colon. If left unconnected, the colon will flash at a 1 Hz rate, due to an internal depletion pull-up on the input. Connection to $V_{SS}$ will produce a non-flashing (always ON) colon.

**Alarm Output and Alarm OFF Input (Pins 11 and 12):** The alarm comparator *(Figure 1)* senses coincidence between the alarm counters (the alarm setting) and the time counters (real time). The comparator output is used to set a latch in the alarm circuit, whose output enables the alarm tone to appear at the alarm output (pin 11), which easily interfaces to a loudspeaker. The alarm latch will remain set for 59 minutes, during which time the alarm will sound, provided the latch is not temporarily inhibited by another latch which is set by the snooze input (pin 13) or reset by the alarm OFF input (pin 12). The alarm tone is generated by an on-chip oscillator which also provides the timing for the serial data input (pin 19). The alarm tone will be at 1 kHz. This square wave is then gated by a 2 Hz square wave signal before being enabled at the alarm output. Momentarily connecting pin 12 to $V_{SS}$ resets the alarm latch and thereby silences the alarm. This input is also returned to $V_{DD}$ by an internal depletion load. The momentary alarm OFF input also readies the alarm latch for the next comparator output, and the alarm will automatically sound again in 24 hours (or at a new alarm setting). If it is desired to silence the alarm for a day or more, the alarm OFF input should remain at $V_{SS}$. An alarm indicator output (pin 21) is provided to indicate the status of the alarm OFF input.

**Snooze Input (Pin 13):** Momentarily connecting pin 13 to $V_{SS}$ disables the sleep output (pin 10). If the alarm has sounded just prior to this, the alarm output is disabled for 8 to 9 minutes (depending upon the contents of the real time seconds counter) after which the alarm will again be sounded. The snooze feature may be repeatedly used during the 59 minutes in which the alarm latch remains set.

**Sleep Timer Output (Pin 10):** The sleep output at pin 10 can be used to turn off a radio after a desired time interval of up to 59 minutes. The time interval is chosen by selecting the sleep display mode (Table I) and setting the desired time interval (Table II). This automatically results in a current sink output via pin 10, which can be used to turn on a radio (or other appliance). When the sleep counter, which counts downward, reaches 00 minutes, the sleep latch is reset and the sleep output drive is removed, thereby turning off the radio.

This turn-off may also be manually controlled by a momentary connection of the snooze input (pin 13) to $V_{SS}$. Also, when the contents of the sleep counter reach 00 minutes, displaying the contents of the sleep counter automatically sets the sleep counter to 59 minutes and enables the sleep output, causing a radio to turn on. If a sleep time other than 59 minutes is desired, a manual set must be performed, but most users desire the longest sleep time possible before bedtime. This feature minimizes pre-bedtime setting operation. Note that 00 minutes on the sleep counter will never be displayed, as displaying sleep automatically sets it to 59 minutes.

**Standby Oscillator Input (Pin 16):** By supplying a capacitor and resistor from this input to $V_{SS}$ and $V_{DD}$ respectively, the stand-by oscillator can be used for timekeeping purposes when the normal 60 Hz line power fails *(Figure 4)*. A 9 V battery is used to power the MOS circuit and although the display is blanked, the correct time is held by the time counters and counting continues. When line power resumes, the display returns to normal brightness, displaying the correct time, without flashing.

The stand-by oscillator is set to 20 kHz nominally and has 3 functions. One is to supply the 10 kHz data rate for the data communications I/O bus (pin 19). Another is to generate the 1 kHz tone for the alarm. The third function is to provide a stand-by clock to the timekeeping circuits in the event of a power failure.

**Brightness Input (Pin 18):** The LED output currents may be varied by the simple connection of this input to $V_{DD}$ through a variable resistor $R_b$. This simple 1-pin operation thereby controls the brightness of the LED display. The output current typically equals 20 times the reference current set through $R_b$. Internal resistance is included to limit the maximum current *(Figure 4)*.

**LED Display Outputs and Common Source (Pins 21-39):** All LED display outputs are open drain devices with all sources connected directly to common source pin 30. Each display output has a separate driver which may sink a maximum of 30 mA directly at 1.5V across the output device. Fourteen segments A1–G1 and A2–G2 are used to drive the numeric-duplex display. Segments H2 and J2 are also provided in order to display alpha-numeric data in the hours-tens and hours-units position. See Table III for the alpha characters available.

## Functional Description (Continued)

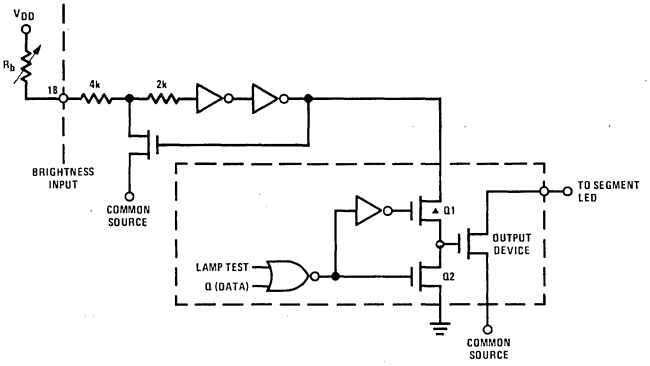



FIGURE 3 Brightness Control Circuitry

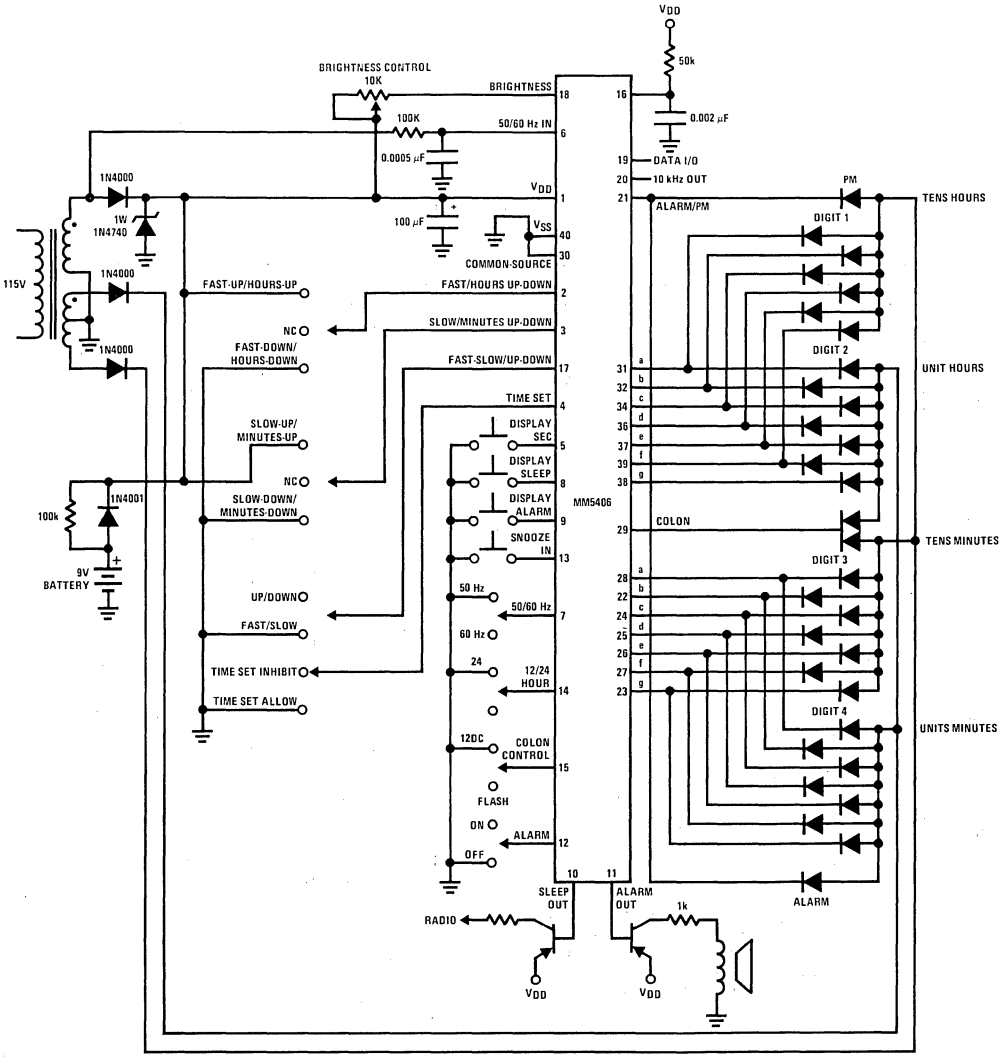



FIGURE 4 Typical Application for MM5406

## Functional Description (Continued)

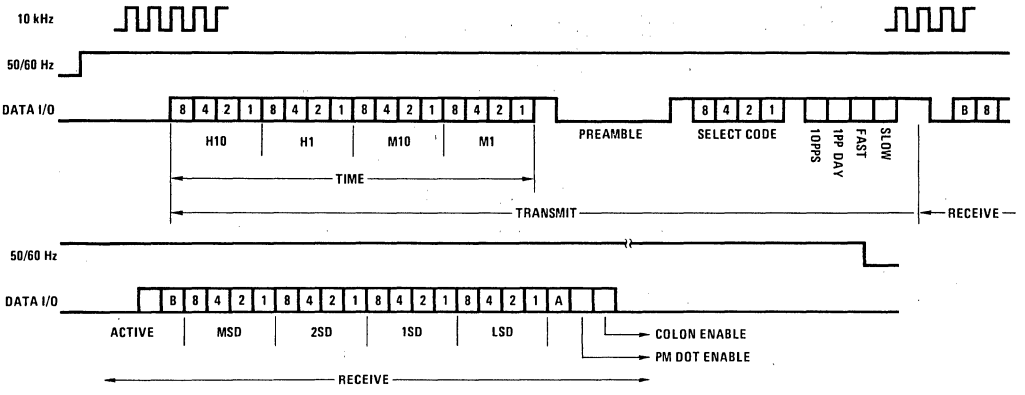



FIGURE 5. MM5406 Timing

### TABLE III. MM5406 DISPLAY

| H10 (MSD) | | | | | | H1 (2SD) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| B | 8 | 4 | 2 | 1 | CHARACTER | A | 8 | 4 | 2 | 1 | CHARACTER |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 | 2 | 1 | 1 | 1 | 0 | 1 | 2 |
| 1 | 1 | 1 | 0 | 0 | 3 | 1 | 1 | 1 | 0 | 0 | 3 |
| 1 | 1 | 0 | 1 | 1 | 4 | 1 | 1 | 0 | 1 | 1 | 4 |
| 1 | 1 | 0 | 1 | 0 | 5 | 1 | 1 | 0 | 1 | 0 | 5 |
| 1 | 1 | 0 | 0 | 1 | 6 | 1 | 1 | 0 | 0 | 1 | 6 |
| 1 | 1 | 0 | 0 | 0 | 7 | 1 | 1 | 0 | 0 | 0 | 7 |
| 1 | 0 | 1 | 1 | 1 | 8 | 1 | 0 | 1 | 1 | 1 | 8 |
| 1 | 0 | 1 | 1 | 0 | 9 | 1 | 0 | 1 | 1 | 0 | 9 |
| 1 | 0 | 1 | 0 | 1 | (Blank) | 1 | 0 | 1 | 0 | 1 | (Blank) |
| 1 | 0 | 1 | 0 | 0 | - * | 1 | 0 | 1 | 0 | 0 | - * |
| 0 | 1 | 1 | 1 | 1 | A | 0 | 1 | 1 | 1 | 1 | A |
| 0 | 1 | 1 | 1 | 0 | b | 0 | 1 | 1 | 1 | 0 | b |
| 0 | 1 | 1 | 0 | 1 | C | 0 | 1 | 1 | 0 | 1 | C |
| 0 | 1 | 1 | 0 | 0 | d | 0 | 1 | 1 | 0 | 0 | d |
| 0 | 1 | 0 | 1 | 1 | E | 0 | 1 | 0 | 1 | 1 | E |
| 0 | 1 | 0 | 1 | 0 | F | 0 | 1 | 0 | 1 | 0 | F |
| 0 | 1 | 0 | 0 | 1 | G | 0 | 1 | 0 | 0 | 1 | G |
| 0 | 1 | 0 | 0 | 0 | H | 0 | 1 | 0 | 0 | 0 | H |
| 0 | 0 | 1 | 1 | 1 | L | 0 | 0 | 1 | 1 | 1 | L |
| 0 | 0 | 1 | 1 | 0 | P | 0 | 0 | 1 | 1 | 0 | P |
| 0 | 0 | 1 | 0 | 1 | S | 0 | 0 | 1 | 0 | 1 | S |
| 0 | 0 | 1 | 0 | 0 | U | 0 | 0 | 1 | 0 | 0 | U |

Note. M1 and M10 will display 0–9, blank and minus.
The following are examples of the alpha display:
JA, FE, Mr, AP, MA, JU, JL, AU, SE, OC, NO, dE
SU, MO, TU, WE, TH, Fr, SA
AM, PM, FM, CH (channel no.), FT (Feet)
MG (miles per gallon), MH (miles per hour), GA (gallons)
MI (miles)
Mn (minutes), Hr (hours), SC (seconds)
*Minus

National Semiconductor

# MM5407 Digital Thermometer

## General Description

The MM5407 Digital Thermometer is a monolithic MOS integrated circuit utilizing N-channel metal gate low threshold, enhancement mode and ion-implanted depletion mode devices. The MM5407 interfaces directly with the temperature transducer LM134 and MM5406 digital clock. LM134 senses the temperature and outputs voltage proportional to temperature in degrees Kelvin at 10 mV/°K. MM5407 senses this analog voltage and converts it into BCD Centrigrade or Fahrenheit data. MM5407 transfers temperature data serially to the MM5406 clock which in turn converts this BCD data into 7-segment display code. MM5406 interfaces directly with LED display. MM5407 has capability of generating serial data either for Centigrade or Fahrenheit temperatures. The temperature ranges are from −40°C to +88°C and −40°F to +194°F. MM5407 operates over the supply range of 9-11V and is packaged in 14-pin plastic package. Serial data output can also be generated by bringing the transfer enable input low. In this mode the MM5407 can easily interface to a microprocessor.

## Features

- Centigrade or Fahrenheit data
- Direct interface to MM5406 clock chip
- Simple interface to LM134 temperature transducer
- Simple interface to microprocessor
- Serial data output
- Convenient 14-lead DIP package

## Applications

- Indoor and outdoor digital thermometer
- Temperature sensor for microprocessor or minicomputer
- Digital thermostat

## Block and Connection Diagrams

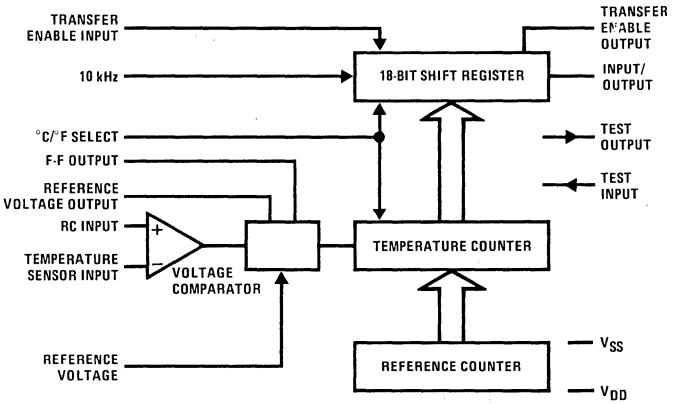
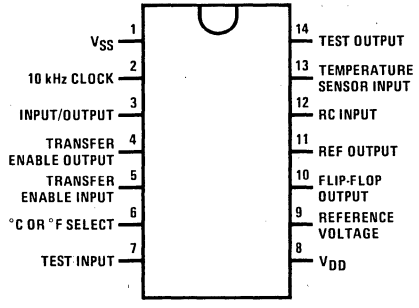



FIGURE 1

Dual-In-Line Package




TOP VIEW

Order Number MM5407N
See Package 18

## Absolute Maximum Ratings

| | |
|---|---|
| Maximum Voltage at Any Pin | $V_{SS}$ to $V_{SS}$ + 12V |
| Operating Temperature | −25°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics ($T_A$ within operating range) Functional Voltage, $V_{SS}$ = 0, $V_{DD}$ = 9 to 11V

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage | | 9 | | 11 | V |
| Power Supply Current | $V_{DD}$ = 9V | | | 15 | mA |
| | $V_{DD}$ = 11V | | | 18 | mA |
| Temperature Sensor Input Voltage | 10 mV/°C | 2.33 | | 3.61 | V |
| RC Input, Input,Voltage | 10 mV/°C | 2.33 | | 3.61 | V |
| $V_{REF}$ Input, Input Voltage | | 3.627 | 3.63 | 3.633 | V |
| $V_{REF}$ Input Resistance to $V_{SS}$ | | 1.2 | | 3.5 | kΩ |
| Temperature Display Resolution | | | 1 | | °C or °F |
| Temperature Display Range | | −40 | | 88 | °C |
| | | −40 | | 194 | °F |
| Temperature Accuracy (ΔT) Over the Range of −40°F to 194°F | $V_{DD}$ = 9 to 11V | | | ±2 | °F |
| 10 kHz Clock Input | Logic "0" | $V_{SS}$ | | $V_{SS}$+0.6 | V |
| | Logic "1" | $V_{SS}$+2.4 | | $V_{DD}$ | V |
| Input Output I/O as Input | Logic "0" | $V_{SS}$ | | $V_{SS}$+0.3 | V |
| | Logic "1" | $V_{DD}$−2 | | $V_{DD}$ | V |
| Input Output I/O as Output | Logic "0", ($I_{SINK}$ = 2 mA) | $V_{SS}$ | | $V_{SS}$+0.4 | V |
| | Logic "1", ($I_{LEAKAGE}$ = 10 μA) | | | $V_{DD}$ | V |
| Transfer Enable Input | Logic "0" | $V_{SS}$ | | $V_{SS}$+0.6 | V |
| | Logic "1" | $V_{SS}$+2.4 | | $V_{DD}$ | V |
| | Pulsewidth | 1/2 | | 18 | 10 kHz clks |
| Transfer Enable Output | Logic "0", ($I_{SINK}$ = 2 mA) | $V_{SS}$ | | $V_{SS}$+0.4 | V |
| | Logic "1", ($I_{SOURCE}$ = 10 μA) | | | $V_{DD}$ | V |

## Functional Description

A block diagram of the MM5407 is shown in *Figure 1*. Individual pin function is described in the following description. *Figure 2A* shows the typical application of digital thermometer and clock using MM5407, MM5406 and temperature transducer LM134. *Figure 3A* shows an application of the MM5407 as temperature sensor for a microprocessor.

The MM5407 utilizes an analog-to-digital converter circuit which senses analog voltage and converts it into BCD digital data. The analog-to-digital converter uses a positive and negative reference voltage and a comparator to determine the value of an analog input voltage. A positive reference voltage is applied to an RC input and this voltage is compared with the tempera-

## Functional Description (Continued)

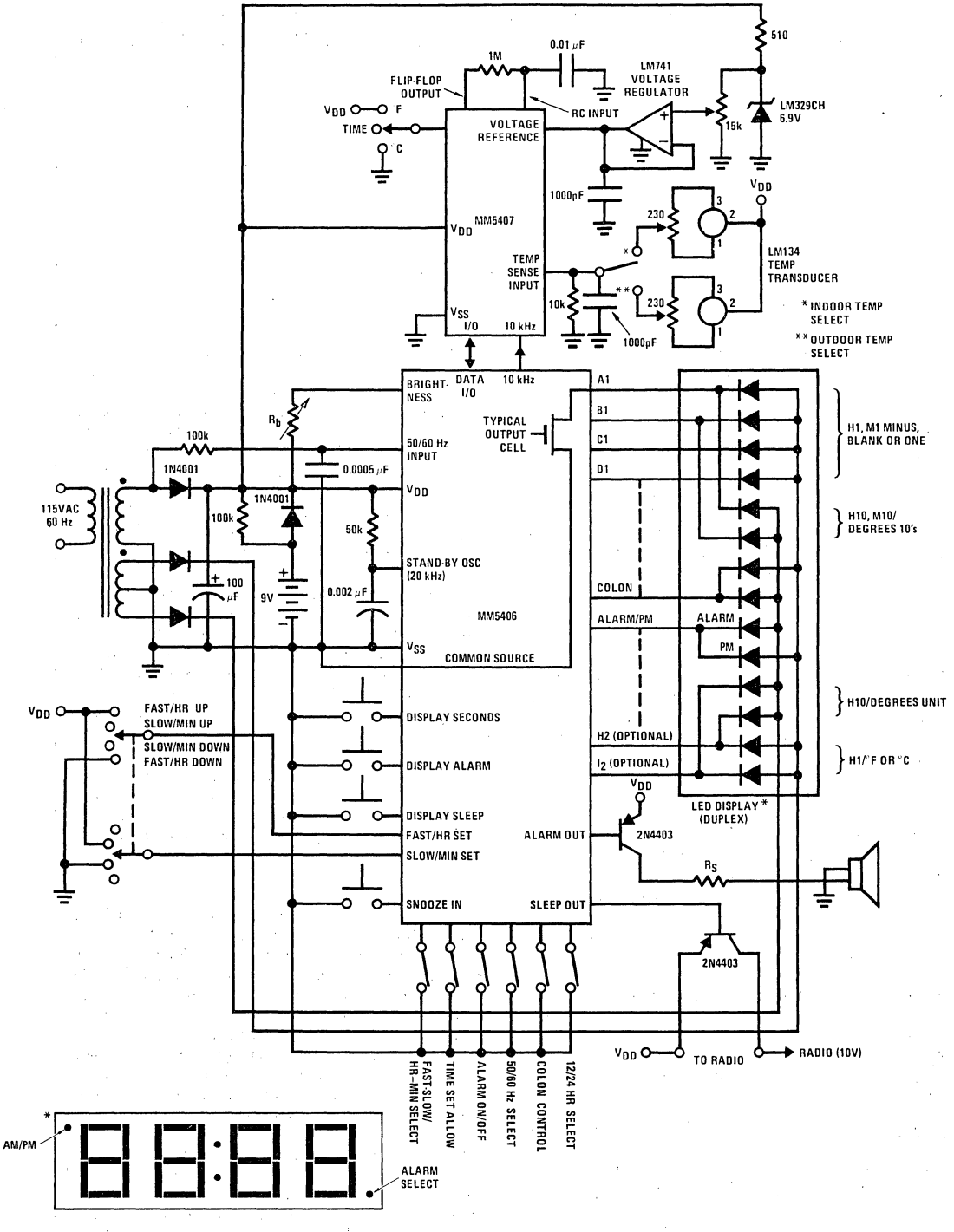
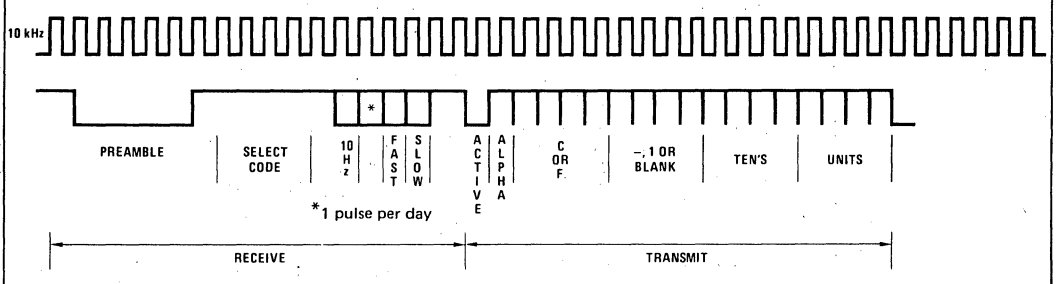



FIGURE 2A. Typical Application of MM5407 Temperature IC and MM5406 Clock IC

## Functional Description (Continued)




Select code for MM5407 is "1111"
10 Hz is used for generating internal time clock

**FIGURE 2B. MM5407 Input/Output Timing Diagram for MM5406 Configuration**

ture sensor input voltage. When the RC input voltage is greater than the temperature sensor voltage, the reference voltage polarity is reversed and the RC input discharges toward the negative reference voltage. The RC input voltage is then compared again with the temperature sensor voltage. When it is less than the temperature sensor voltage the circuit again switches, applying the positive reference voltage to the RC input. The circuit eventually settles down to a duty cycle which when measured by digital counters gives the absolute value of the input voltage. Data from the digital counter is stored in latches. The selection of °C or °F transfers the data serially on the I/O bus to MM5406. The MM5406 converts this data into 7-segment data and displays it on a duplex driven LED display.

**Temperature Sensor Input (Pin 13):** This input senses analog voltage generated by a temperature transducer (e.g., LM134). This input is one of the 2 comparator inputs. The input senses voltage between the range 2.33V (–40°C) to 3.61V (88°C). The input has 10 mV/°C resolution.

**RC Input and Flip-Flop Output (Pins 12, 10):** RC input is the second input to the analog comparator. A 1 MΩ resistor is connected between pin 12 and 10 and a 0.01 μF capacitor is connected between pin 10 and $V_{SS}$, pin 1. Voltage at the RC input (pin 12) is dependent on flip-flop output which in turn depends on the analog input at pin 13. The external RC does not require tight tolerances.

**Reference Voltage (Pin 9):** This input requires precision reference voltage of 3.630 ±0.003V. On-chip voltage divider produces second reference. The accuracy of these 2 references will determine the accuracy of displayed temperature. The on-chip voltage divider can draw current up to 3 mA.

**Input/Output (Pin 3):** The data I/O pin is used for transmitting and receiving serial data via the common data bus line. During the transmit mode, the MM5406 sends out 16 bits of real-time data followed by a select code containing 4 bits for selecting one of the 16 "daughter" chips such as MM5407. After the select code is sent out, 4 status bits are sent. If the daughter chip for that select code is not enabled, or if that daughter chip is not used, the MM5406 will then send out the next higher select code. Whenever a daughter chip such as the MM5407 is enabled, it will deselect the select code and send 20 bits of data to the MM5406. This 20-bit data word contains 4 digits of 4 bits each, 2 special bits used for Alpha information, and 2 bits for enabling the PM DOT and COLON. If the MM5407 is constantly enabled, the MM5406 will send the same select code on each cycle until the MM5407 is disabled. All data on the data I/O pin will be active low ($V_{SS}$). The MM5406 has an internal pull-up to $V_{DD}$, therefore the MM5407 is designed to have an open drain output.

**°C or °F Select (Pin 6):** This input is a TRI-STATE® input. If this pin is connected to $V_{DD}$ or $V_{SS}$, the I/O, pin 3, will transfer the serial temperature data through an LED display. Connecting this pin to $V_{DD}$ selects °F and connecting it to ground ($V_{SS}$) selects °C. In floating mode, the I/O pin will stop sending the data to MM5406, which in turn will take over the display select mode and display its own information.

**10 kHz Clock:** External clock of 10 kHz is used for transmitting and receiving serial data over the data I/O bus line. The MM5406 provides 10 kHz clock for MM5407. MM5407 will accept data from data I/O (pin 3) during ($V_{SS}$). The transmit and receive timing is shown in *Figure 2B*.

**Transfer Enable Input (Pin 5) and Transfer Enable Output (Pin 4):** MM5407 can be used with a microprocessor for sensing temperature. Transfer enable input pin 5 strobes the °C or °F input selection at pin 6 and enables output pin 4. Output enable signal can be used to strobe serial data into shift register. After output enable is on, I/O pin transmits serial data to the external shift register. The 18-bit serial data contains 4 digits of temperature information. Pins 4 and 5 have internal pull-up to $V_{DD}$ capable of sinking a maximum of 10 μA. Use external resistor for higher current capability.

**Test Input:** This pin and the Test output are not used during normal operation of the device. When the Test input is brought to $V_{SS}$ the on-chip oscillator is connected to Test output, so its frequency range can be easily measured. In addition the temperature counters are accessed to fascilitate testing. There is an internal pull-up to the Test input so during normal operation it should be left open.

## Functional Description (Continued)

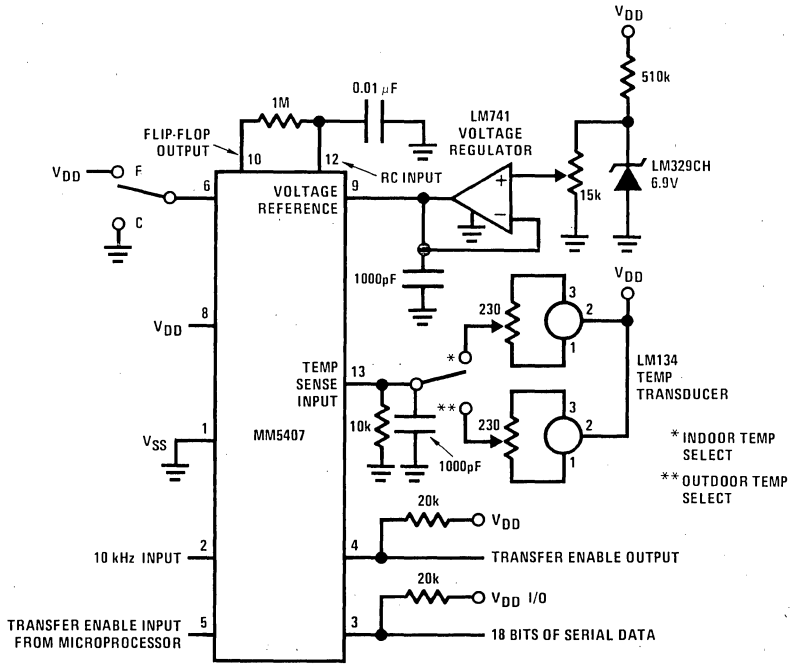
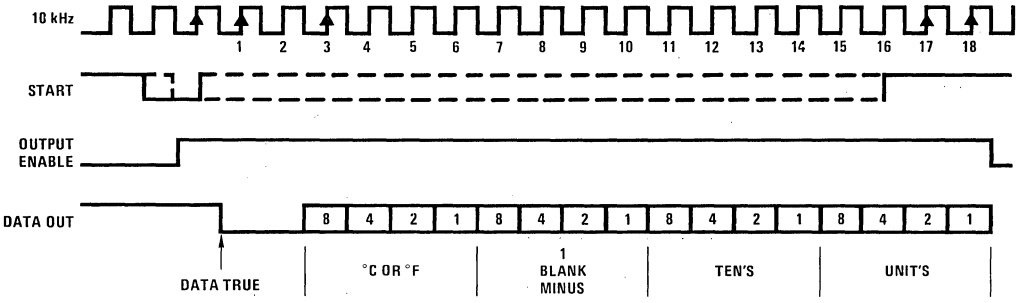



FIGURE 3A. Typical Application of MM5407 Used as a Temperature Sensor for Microprocessor




| DATA OUT | 8 | 4 | 2 | 1 |
|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 0 |
| 2 | 1 | 1 | 0 | 1 |
| 3 | 1 | 1 | 0 | 0 |
| 4 | 1 | 0 | 1 | 1 |
| 5 | 1 | 0 | 1 | 0 |
| 6 | 1 | 0 | 0 | 1 |
| 7 | 1 | 0 | 0 | 0 |
| 8 | 0 | 1 | 1 | 1 |
| 9 | 0 | 1 | 1 | 0 |
| C | 1 | 1 | 1 | 0 |
| F | 1 | 1 | 0 | 1 |
| (−) | 0 | 1 | 0 | 0 |
| Blank | 0 | 1 | 0 | 1 |

FIGURE 3B. Timing for the Above Application

National Semiconductor



# MM5455 Digital Alarm Clock

## General Description

The MM5455 digital alarm clock radio chip is a monolithic MOS integrated circuit utilizing N-channel, low threshold, enhancement mode and ion-implanted depletion mode devices.

The MM5455 contains all the logic necessary for a digital clock with sleep and alarm control and is intended for clock-radio applications.

Real time and alarm time are displayed in hours-minutes and sleep time is displayed in minutes when setting the sleep counter.

An alarm output is provided that "beeps" a 700 Hz tone gated by 2 Hz rate when the alarm set time and the real time matches. A sleep output that provides a DC level is used to control the radio. It is activated with the alarm output or programmed via the sleep counter to turn OFF from 0 to 59 minutes after the sleep counter is set.

A snooze feature is provided for a 9-minute recurrence of the alarm after it has sounded. Setting is done via the standard fast and slow set buttons when in the time set, alarm set or sleep set modes. These control inputs are TRI-STATE® inputs to reduce pin count.

The 50/60 Hz clock selects what segment data is on the outputs, i.e. a duplex LED display interface.

The MM5455 is bonded in a 24-pin package and is capable of 24-hour/50 Hz, 12-hour/60 Hz and 12-hour/50 Hz operations.

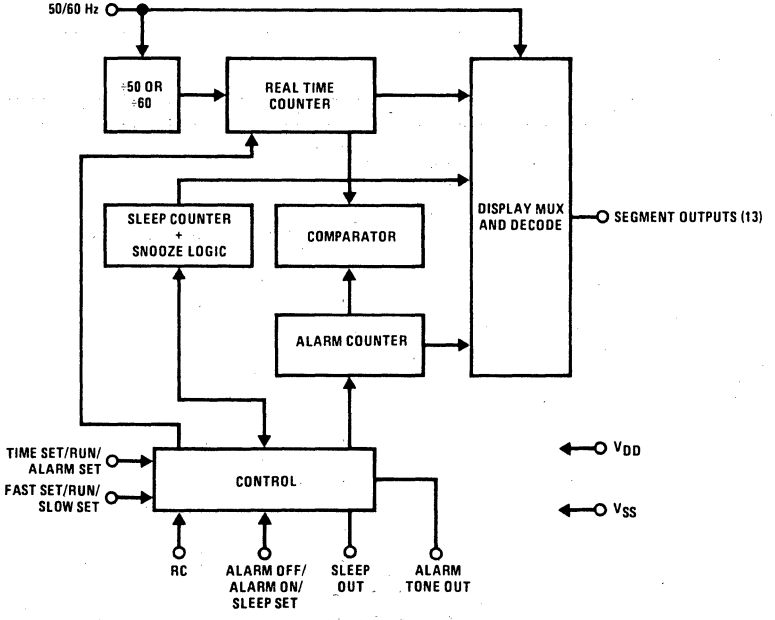
## Features

- Duplex LED display drive
- Fast/slow set capability
- 24-hour alarm
- "Snooze" function (9 minutes)
- On-chip alarm oscillator
- Alarm tone output gated at a 2 Hz rate
- Power fail indication—entire display flashes at a 1 Hz rate
- Automatic power-on reset
- PM display indicator
- Presettable 59 minute sleep timer

## Applications

- Alarm clocks
- Desk clocks
- Clock radios
- Automobile clocks
- Stopwatches
- Industrial clocks
- Portable clocks
- Timers

## Block Diagram



## Absolute Maximum Ratings (Exceeding the following ratings may result in permanent damage to the device)

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ to $V_{SS}$ +12V |
| Operating Temperature | −25°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics Power supply voltage should not exceed a maximum of 12V under any circumstances.

$T_A$ within operating range, $V_{DD}$ = 7V to 11V

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage | Output Driving Display | 9 | | 11 | V |
| | Functional -- Count Operating | 7 | | 11 | V |
| Power Supply Current | No Output Loads | | | | |
| | $V_{DD}$ = 7V | | | 4 | mA |
| | $V_{DD}$ = 11V | | | 5 | mA |
| 50/60 Hz Input Frequency | | DC | 50/60 | 10k | Hz |
| Logical "0", Low Level | | $V_{SS}$ | $V_{SS}$ | $V_{SS}$+0.5 | V |
| Logical "1", High Level | | 0.7 $V_{DD}$ | $V_{DD}$ | $V_{DD}$ | V |
| Input Leakage | $V_{IN}$ = $V_{DD}$ or $V_{SS}$ | | | 10 | μA |
| All Other Input Voltages | | | | | |
| Logical "0", Low Level | | $V_{SS}$ | $V_{SS}$ | $V_{SS}$+0.5 | V |
| Logical "1", High Level | | $V_{DD}$−1 | $V_{DD}$ | $V_{DD}$ | V |
| Power Failure Detect Voltage | ($V_{DD}$ Voltage) | 1 | | 5 | V |
| Alarm Output, Sleep Output | $V_{DD}$ = 11V | | | | |
| Sourcing, Logical "1" High | $V_{OH}$ = $V_{DD}$ − 1V | 1 | | | μA |
| Sinking, Logical "0" Low | $V_{OL}$ = $V_{SS}$ + 6V | 5 | | | mA |
| Alarm Output Frequency | Fixed R and C | 560 | 700 | 840 | Hz |
| Segment Outputs (Except HTADEG) | $V_{DD}$ = 10V | | | | |
| Output Sink, Logical "0" Low | $V_{OL}$ = $V_{SS}$ + 2V | 20 | | | mA |
| Leakage, Logical "1" High | $V_{OH}$ = $V_{DD}$ | | | 10 | μA |
| Segment Output (HTADEG) | $V_{DD}$ = 10V | | | | |
| Output Sink, Logical "0" Low | $V_{OL}$ = $V_{SS}$ + 2V | 40 | | | mA |
| Leakage, Logical "1" High | $V_{OH}$ = $V_{DD}$ | | | 10 | μA |

## Functional Description

**50 or 60 Hz Input:** A shaping circuit is provided internally to square the 50 or 60 Hz input. This circuit allows use of a half sinewave input. A resistor in series must be used to limit current into the MOS device.

**24-Hour/50 Hz, 12-Hour/60 Hz, 12-Hour/50 Hz:** This input is a tri-level input. When connected to $V_{SS}$, the display will be 24 hours with 50 Hz input. If connected to $V_{DD}$, the display will be 12 hours with 50 Hz input. If left floating, the display will be 12 hours with 60 Hz input.

**Sleep Set/Alarm Enable/Alarm OFF:** Whenever this input is connected to $V_{SS}$, the sleep counters will display and be set to 59 minutes, if previously at 00 minutes. Fast set or slow set may be used to preset time other than 59 minutes. When the alarm sounds, the snooze can be activated by momentarily connecting fast set pin to $V_{DD}$. This will turn off the alarm for. 9–10 minutes, after which the alarm will again be sounded. The snooze alarm feature may be repeatedly used for up to 59 minutes or until alarm OFF is activated by momentarily connecting this pin to $V_{DD}$. When alarm OFF is activated, the alarm latch will reset and silence the alarm. The alarm will automatically sound again in 24 hours (or at a new alarm setting). If it is desired to silence the alarm for a day or more, the alarm OFF input should remain at $V_{DD}$. The alarm can be enabled again by allowing the input to float.

**External RC:** The resistor and capacitor on this pin set the frequency of the alarm tone. The frequency is nominally set to 1400 Hz in order to produce a 700 Hz (50% duty cycle) tone at the alarm output.

## Functional Description (Continued)

**Time Set/Run/Alarm Set:** This input must be activated in order to set the time counter and alarm counter. When connected to $V_{DD}$, the time counter will be displayed and may be set with the fast-slow input. Upon release of fast and slow set, the time counter will be in a hold mode. If time set/alarm set is left floating, the time counter will be displayed, and the fast-slow input will be disabled (normal run mode). When this input is connected to $V_{SS}$, the alarm counter will be displayed and may be set with the fast-slow input.

**Fast Set/Run/Slow Set:** Whenever this input is connected to $V_{DD}$ and time set or alarm set is activated, the appropriate counter will advance at a 60 Hz rate. If connected to $V_{SS}$,the minutes counter will advance at 2 Hz with carry into the hours counter. If left floating, the clock will keep normal time. When connected to $V_{DD}$ and sleep set is activated, the sleep counter will count downward at a 10 Hz rate. If connected to $V_{SS}$, the sleep counter will count downward at the slower rate; 2 Hz.

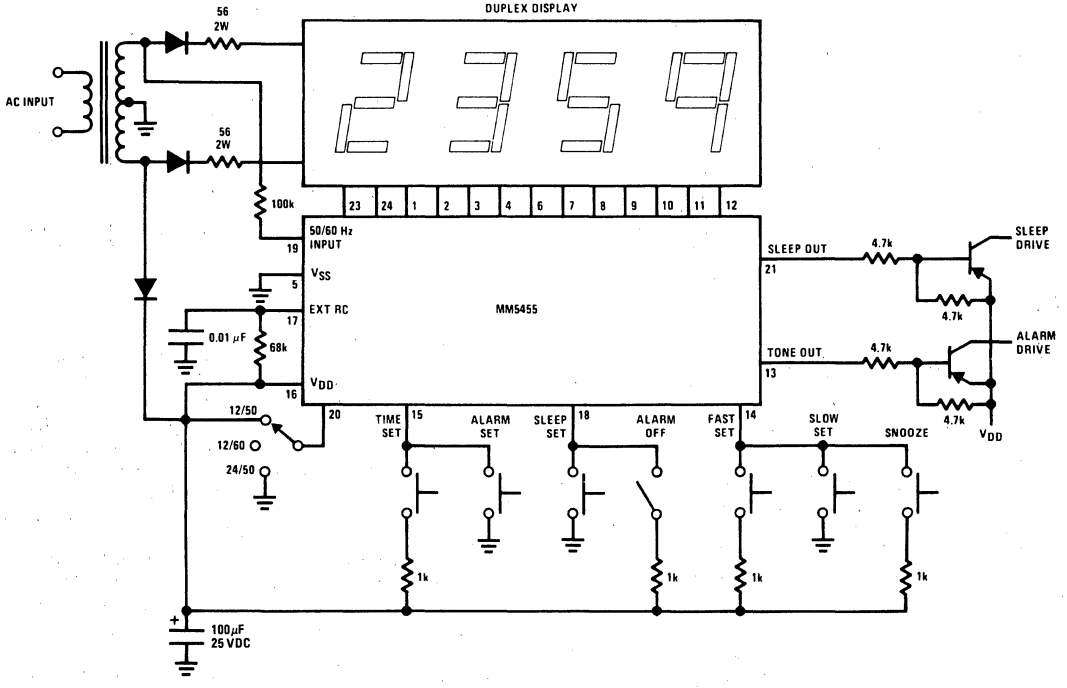
**Alarm Output:** This output remains at $V_{DD}$ when inactive. When the alarm latch becomes set, the alarm 700 Hz tone will be gated by a 50% duty cycle 2 Hz signal (i.e., the alarm tone will be ON for 250 ms and OFF for 250 ms). This output contains an internal pull-up to $V_{DD}$ in order to turn OFF the external PNP transistor driver when the alarm is inactive.

**Sleep Output:** This output remains at $V_{DD}$ when inactive. When the alarm latch becomes set or the sleep counter is at other than 00 minutes, the sleep output will be ON. Snooze or alarm OFF function will disable this output simultaneously with alarm output if activated by alarm latch. Snooze function will disable this output at any time if activated by sleep counter being set. This output contains an internal pull-up to $V_{DD}$ in order to turn OFF the external PNP transistor switch when the output is inactive.

**Segment Outputs:** All segment outputs are open-drain devices with all sources common to $V_{SS}$.

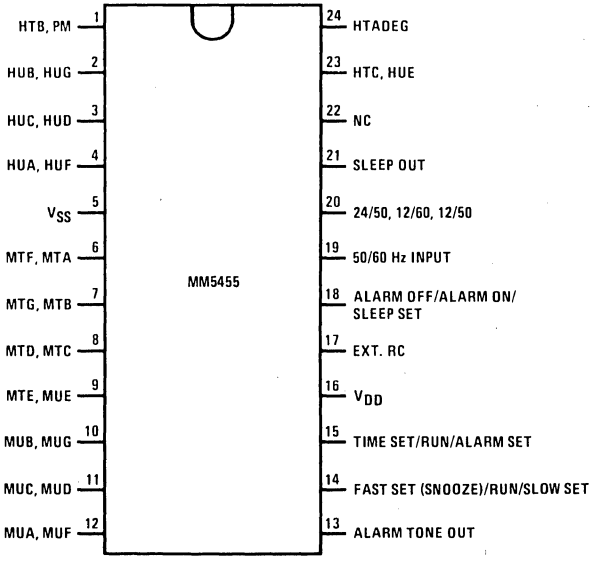
**Power ON and Power Failure:** After power ON or power fail, the display will flash at 1 Hz rate and time, alarm, and seconds counters will be reset to 0:00:00 (12:00:00). *The alarm output will be held OFF also.* The power fail can be reset by enabling time set allow.

## Typical Application



## Connection Diagram

Dual-In-Line Package
12 or 24-Hour/50 or 60 Hz Clock




TOP VIEW

Order Number MM5455N
See Package 22

National Semiconductor



# MM5456, MM5457 Digital Alarm Clocks

## General Description

The MM5456, MM5457 digital alarm clock radio chips are monolithic MOS integrated circuits utilizing N-channel, low threshold, enhancement mode and ion-implanted depletion mode devices.

Each circuit contains all the logic necessary for a digital clock with sleep and alarm control and is intended for clock-radio applications.

Real time and alarm time are displayed in hours-minutes and sleep time is displayed in minutes when setting the sleep counter.

An alarm output is provided that "beeps" a 50% duty cycle, 700 Hz signal gated at 2 Hz rate when the alarm set time and the real time matches. A sleep output that provides a DC level is used to control the radio. It is activated with the alarm output or programmed via the sleep counter to turn OFF from 0 to 59 minutes after the sleep counter is set.

A snooze feature is provided for a 9-minute recurrence of the alarm after it has sounded.

Setting is done via the standard fast and slow set buttons when in the time set, alarm set or sleep set modes. These control inputs are TRI-STATE® inputs to reduce pin count.

The 50/60 Hz clock selects what segment data is on the outputs, i.e., a duplex LED display interface.

The MM5456, MM5457 are bonded in a 22-pin package. The MM5457 has a 24-hour/50 Hz option and the MM5456 has the 12-hour/50 Hz or 12-hour/60 Hz options.

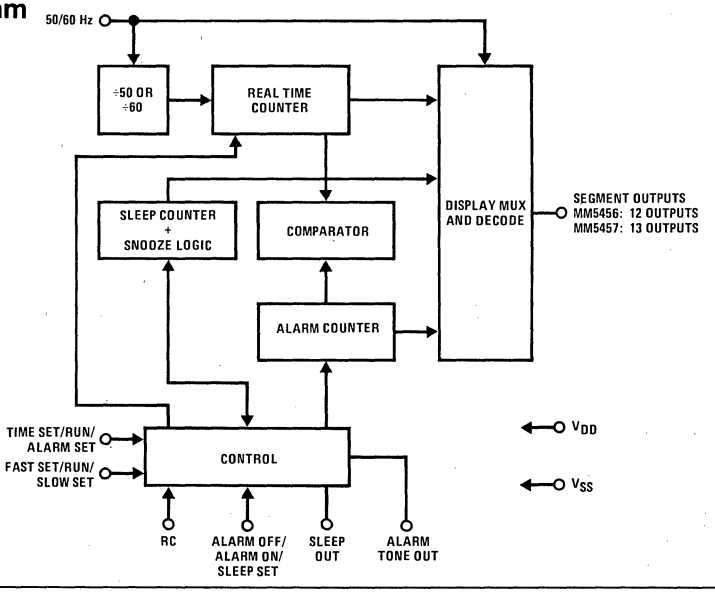
## Features

- Duplex LED display drive
- Fast/slow set capability
- 24-hour alarm
- "Snooze" function (9 minutes)
- On-chip alarm oscillator
- Alarm tone output gated at a 2 Hz rate
- Power fail indication—entire display flashes at a 1 Hz rate
- Automatic power-on reset
- PM display indicator
- Presettable 59 minute sleep timer

## Applications

- Alarm clocks
- Desk clocks
- Clock radios
- Automobile clocks
- Stopwatches
- Industrial clocks
- Portable clocks
- Timers

## Block Diagram



## Absolute Maximum Ratings (Exceeding the following ratings may result in permanent damage to the device)

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ to $V_{SS}$ +12V |
| Operating Temperature | −25°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics Power supply voltage should not exceed a maximum of 12V under any circumstances.

$T_A$ within operating range, $V_{DD}$ = 7V to 11V

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage | Output Driving Display | 9 | | 11 | V |
| | Functional – Count Operating | 7 | | 11 | V |
| Power Supply Current | No Output Loads | | | | |
| | $V_{DD}$ = 7V | | | 4 | mA |
| | $V_{DD}$ = 11V | | | 5 | mA |
| 50/60 Hz Input Frequency | | DC | 50/60 | 10k | Hz |
| Logical "0", Low Level | | $V_{SS}$ | $V_{SS}$ | $V_{SS}$+0.5 | V |
| Logical "1", High Level | | 0.7 $V_{DD}$ | $V_{DD}$ | $V_{DD}$ | V |
| Input Leakage | $V_{IN}$ = $V_{DD}$ or $V_{SS}$ | | | 10 | μA |
| All Other Input Voltages | | | | | |
| Logical "0", Low Level | | $V_{SS}$ | $V_{SS}$ | $V_{SS}$+0.5 | V |
| Logical "1", High Level | | $V_{DD}$−1 | $V_{DD}$ | $V_{DD}$ | V |
| Power Failure Detect Voltage | ($V_{DD}$ Voltage) | 1 | | 5 | V |
| Alarm Output, Sleep Output | $V_{DD}$ = 11V | | | | |
| Sourcing, Logical "1" High | $V_{OH}$ = $V_{DD}$ – 1V | 1 | | | μA |
| Sinking, Logical "0" Low | $V_{OL}$ = $V_{SS}$ + 6V | 5 | | | mA |
| Alarm Output Frequency | R = 68k, C = 0.1μF | | 700 | | Hz |
| Segment Outputs (Except HTADEG) | $V_{DD}$ = 10V | | | | |
| Output Sink, Logical "0" Low | $V_{OL}$ = $V_{SS}$ + 2V | 20 | | | mA |
| Leakage, Logical "1" High | $V_{OH}$ = $V_{DD}$ | | | 10 | μA |
| Segment Output (HTADEG) | $V_{DD}$ = 10V | | | | |
| Output Sink, Logical "0" Low | $V_{OL}$ = $V_{SS}$ + 2V | 40 | | | mA |
| Leakage, Logical "1" High | $V_{OH}$ = $V_{DD}$ | | | 10 | μA |

## Functional Description

**50 or 60 Hz Input:** A shaping circuit is provided internally to square the 50 or 60 Hz input. This circuit allows use of a half sinewave input. A resistor in series must be used to limit current into the MOS device.

**24-Hour/50 Hz, 12-Hour/60 Hz, 12-Hour/50 Hz (MM5456):** When this input is connected to $V_{DD}$, the display will be 12 hours with 50 Hz input. If left floating, the display will be 12 hours with 60 Hz input. (The MM5457 is internally bonded to select the 24-hour/50 Hz mode. The MM5456 selects the 12-hour/50 Hz or the 12-hour/60 Hz modes.)

**Sleep Set/Alarm Enable/Alarm OFF:** Whenever this input is connected to $V_{SS}$, the sleep counters will display and be set to 59 minutes, if previously at 00 minutes. Fast set or slow set may be used to preset time other than 59 minutes. When the alarm sounds, the snooze can be activated by momentarily connecting fast set pin to $V_{DD}$. This will turn off the alarm for 9–10 minutes, after which the alarm will again be sounded. The snooze alarm feature may be repeatedly used for up to 59 minutes or until alarm OFF is activated by momentarily connecting this pin to $V_{DD}$. When alarm OFF is activated, the alarm latch will reset and silence the alarm. The alarm will automatically sound again in 24 hours (or at a new alarm setting). If it is desired to silence the alarm for a day or more, the alarm OFF input should remain at $V_{DD}$. The alarm can be enabled again by allowing the input to float.

**External RC:** The resistor and capacitor on this pin set the frequency of the alarm tone. The frequency is

## Functional Description (Continued)

nominally set to 1400 Hz in order to produce a 700 Hz (50% duty cycle) tone at the alarm output.

**Time Set/Run/Alarm Set:** This input must be activated in order to set the time counter and alarm counter. When connected to $V_{DD}$, the time counter will be displayed and may be set with the fast-slow input. Upon release of fast and slow set, the time counter will be in a hold mode. If time set/alarm set is left floating, the time counter will be displayed, and the fast-slow input will be disabled (normal run mode). When this input is connected to $V_{SS}$, the alarm counter will be displayed and may be set with the fast-slow input.

**Fast Set/Run/Slow Set:** Whenever this input is connected to $V_{DD}$ and time set or alarm set is activated, the appropriate counter will advance at a 60 Hz rate. If connected to $V_{SS}$,the minutes counter will advance at 2 Hz with carry into the hours counter. If left floating, the clock will keep normal time. When connected to $V_{DD}$ and sleep set is activated, the sleep counter will count downward at a 10 Hz rate. If connected to $V_{SS}$, the sleep counter will count downward at the slower rate; 2 Hz.

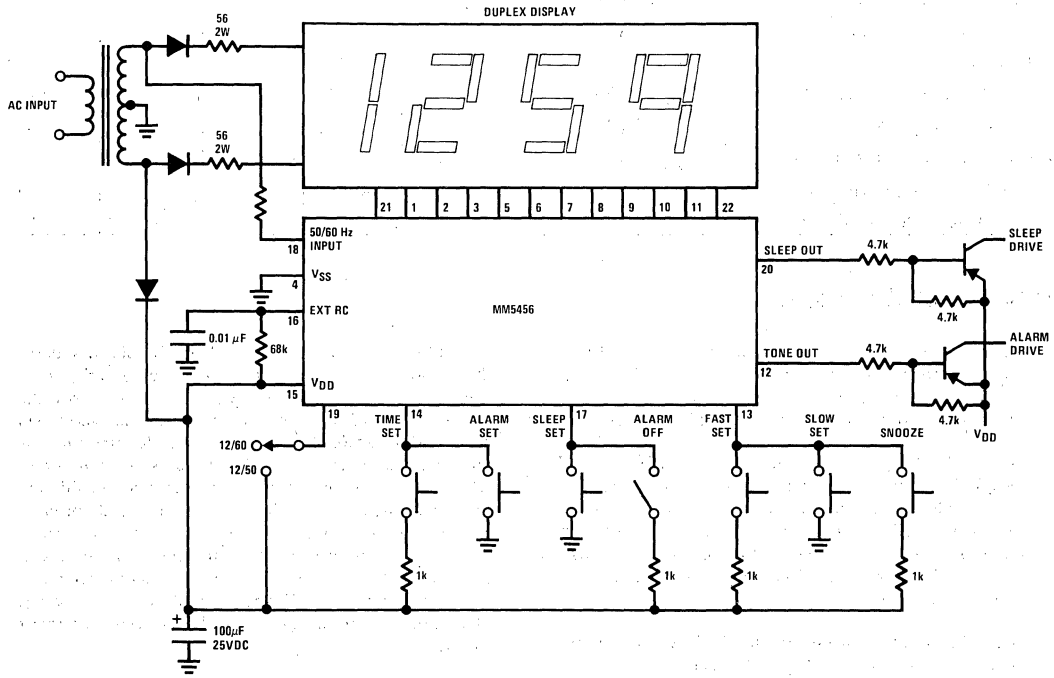
**Alarm Output:** This output remains at $V_{DD}$ when inactive. When the alarm latch becomes set, the alarm 700 Hz tone will be gated by a 20% duty cycle 2 Hz signal (i.e., the alarm tone will be ON for 100 ms and OFF for 400 ms). This output contains an internal pull-up to $V_{DD}$ in order to turn OFF the external PNP transistor driver when the alarm is inactive.

**Sleep Output:** This output remains at $V_{DD}$ when inactive. When the alarm latch becomes set or the sleep counter is at other than 00 minutes, the sleep output will be ON. Snooze or alarm OFF function will disable this output simultaneously with alarm output if activated by alarm latch. Snooze function will disable this output at any time if activated by sleep counter being set. This output contains an internal pull-up to $V_{DD}$ in order to turn OFF the external PNP transistor switch when the output is inactive.

**Segment Outputs:** All segment outputs are open-drain devices with all sources common to $V_{SS}$.

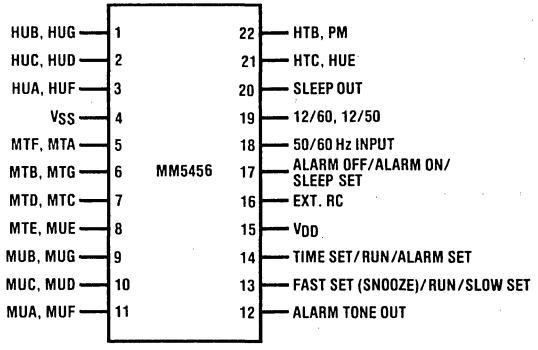
**Power ON and Power Failure:** After power ON or power fail, the display will flash at 1 Hz rate and time, alarm, and seconds counters will be reset to 0:00:00 (12:00:00). *The alarm output will be held OFF also.* The power fail can be reset by enabling time set allow.

## Typical Application



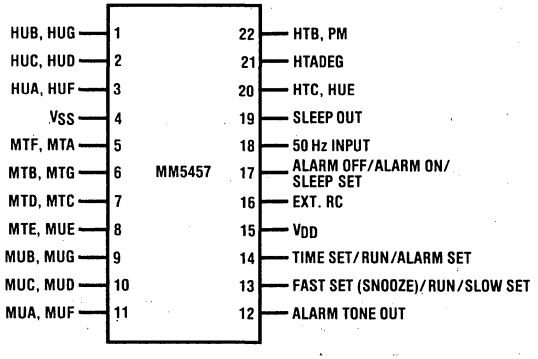
## Connection Diagrams (Dual-In-Line Packages, Top Views)

### 12-Hour/50 or 60 Hz Clock




Order Number MM5456N
See Package 22

### 24-Hour/50 Hz Clock




Order Number MM5457N
See Package 22

5

National Semiconductor

Digital Clocks

# MM58143, MM58144, MM58183, MM58184 LCD Alarm Clock Circuits

## General Description

The MM58143, MM58144 and their mirror versions MM58183, MM58184 are low threshold voltage, ion implanted, metal-gate CMOS integrated circuits that provide all the functions necessary to implement several liquid crystal alarm clocks. Both circuits use a 32,768 Hz quartz oscillator as the time base. Necessary RC components for the oscillator are included on-chip to minimize system components cost. 28 phase-controlled 32 Hz outputs are available to direct drive a 4-digit display with PM, alarm, and sleep flags. The circuits include a 24-hour alarm function, a 59-minute sleep timer, and a 7-minute snooze timer. Hours-minutes in either 12 or 24-hour format are normally displayed. Alternate display modes include seconds, alarm and sleep time. The alarm output is a pulsating tone with a maximum duration of one minute to minimize power consumption.

Two packaged versions are available for clock-radio and travel alarm applications. The MM58143 or MM58183 clock-radio circuit allows multiple battery supplies of 1.5V for timing and logic controls and −1.5V or −3.0V for display drive. The MM58144 or MM58184 travel alarm circuit eliminates the sleep timer functions, but includes an on-chip voltage multiplier for single 1.5V battery operation. The voltage multiplier requires 3 external capacitors and can provide greater than−2.0V drive at 1 $\mu$A load in order to drive 0.5" displays.

The MM58143, MM58144, MM58183 and MM58184 are available in standard 40-lead epoxy package. In addition, the basic 49 pad MM58143 die is available unpackaged suitable for PCB module assembly systems.

## Features

- Direct drive of 0.3" to 1.0" LCD displays
- 4-digit plus PM, sleep, and alarm flags
- Selectable 12 or 24-hour time display
- 24-hour alarm
- Presettable 59-minute sleep timer
- 7-minute snooze timer
- Selectable flashing or non-flashing colon (die only)
- Fast and slow set controls
- Low power dissipation
- On-chip oscillator RC components
- Efficient on-chip voltage multiplier

## Block Diagram

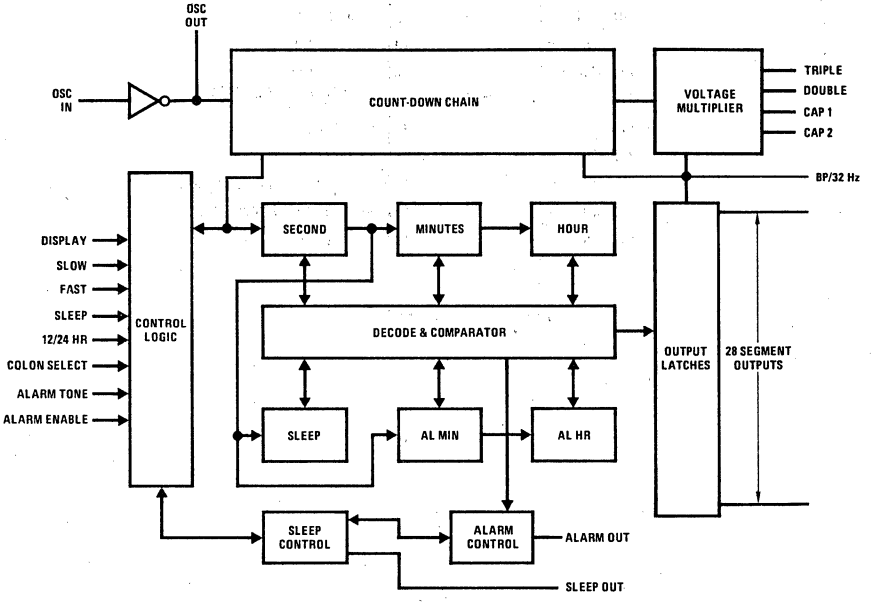



FIGURE 1. Block Diagram of MM58143, MM58144, MM58183, MM58184

## Absolute Maximum Ratings

| | | | |
|---|---|---|---|
| Voltage at Double, Osc. Out, and All Inputs | $V_{DD}$ + 0.3V to $V_{SS}$ − 0.3V | Storage Temperature Range | −25°C to +85°C |
| Voltage at Triple Output | 2 $V_{DD}$ to $V_{SS}$ − 0.3V | $V_{DD} - V_{EE}$ | 8V |
| Voltage at All Other Outputs | $V_{DD}$ + 0.3V to $V_{EE}$ − 0.3V | $V_{DD} - V_{SS}$ | 3V |
| Operating Temperature Range | −5°C to +70°C | Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, $V_{DD} - V_{SS}$ = 1.5V, $V_{DD} - V_{EE}$ = 4.5V unless otherwise noted.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Oscillator Start Voltage | $T_A$ = 25°C, (Note 1) | 1.40 | | | V |
| Oscillator Sustaining Voltage | $T_A$ = −5°C, (Note 1) | 1.30 | | | V |
| Input Voltage Levels | | | | | |
| BP/32 Hz Input | | | | | |
| Logical "1" | | $V_{DD}$−0.25 | | $V_{DD}$ | V |
| Logical "0" | | $V_{EE}$ | | $V_{EE}$+0.25 | V |
| All Others | | | | | |
| Logical "1" | | $V_{DD}$−0.25 | | $V_{DD}$ | V |
| Logical "0" | Internal Pull-Down to $V_{SS}$ | | Open | | |
| Input Current Levels | | | | | |
| Fast, Slow, Display, Sleep | $V_{IN}$ = $V_{DD}$ | 5 | | 60 | μA |
| 12/24 Hr, Alarm Enable, Alarm Tone, Colon Select | $V_{IN}$ = $V_{DD}$ | | | 0.5 | μA |
| Output Current Levels | | | | | |
| Segment Drivers | | | | | |
| Logical "1" | $V_{OUT}$ = $V_{DD}$ − 0.25V, $V_{DD} - V_{EE}$ = 3V | 4 | | | μA |
| Logical "0" | $V_{OUT}$ = $V_{EE}$ − 0.25V, $V_{DD} - V_{EE}$ = 3V | 4 | | | μA |
| BP/32 Hz Output | | | | | |
| Logical "1" | $V_{OUT}$ = $V_{DD}$ − 0.25V, $V_{DD} - V_{EE}$ = 3V | 40 | | | μA |
| Logical "0" | $V_{OUT}$ = $V_{EE}$ − 0.25V, $V_{DD} - V_{EE}$ = 3V | 40 | | | μA |
| FB Resistor R1 | $T_A$ = 25°C | 5 | | 40 | MΩ |
| Input Capacitance | | | | | |
| Osc. Out | f = 1 MHz, $V_{IN}$ = 0V | | 20 | | pF |
| All Others | All Other Pads GND | | | 5 | pF |
| Supply Current ($I_{DD}$) | | | | | |
| Doubler Operation | $T_A$ = 25°C, $I_{EE}$ = 1 μA, f = 32,768 Hz, | | | 15 | μA |
| Tripler Operation | $V_{DD}$ = 1.5V | | | 20 | μA |
| Supply Voltage ($V_{EE}$) | $T_A$ = 25°C, C = 0.047 μF, | | | | |
| Doubler Operation | $I_{EE}$ = 1 μA, f = 32,768 Hz, | −1.0 | | | V |
| Tripler Operation | $V_{DD} - V_{SS}$ = 1.5V | −2.0 | | | V |
| Output Drive Current (Source) | | | | | |
| Alarm Out, Sleep Out | $V_{DD} - V_{EE}$ = 2.5V | | | | |
| | $V_{OUT}$ = $V_{DD}$ − 0.8V | 1 | | | mA |

**Note 1:** In oscillator network shown in *Figure 6*.

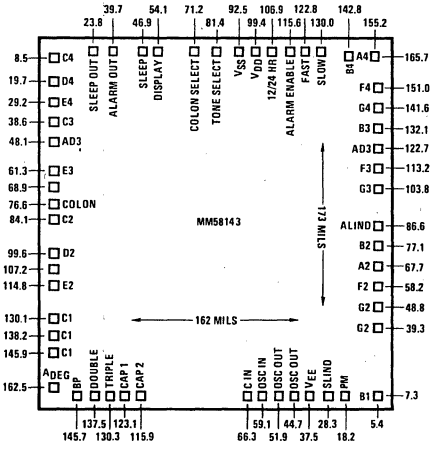
## Functional Description

A block diagram of the MM58143, MM58144, MM58183 and MM58184 LCD clock is shown in *Figure 1*, chip pad layouts in *Figure 2*, connection diagrams in *Figure 3*, and typical application in *Figure 4*.

**Time Base:** The precision time base of the clock is provided by connecting a quartz crystal network to the on-chip CMOS inverter/amplifier as shown in *Figure 6*. For proper operation, the network should be tuned to 32,768 Hz. Resistor R1 is used to bias the on-chip inverter for Class A amplifier operation. Resistor R2 is used to a) reduce the voltage sensitivity of the network; b) limit the power dissipation in the quartz crystal, and c) provide added phase shift for good start-up and low voltage operation. C1 and C2 in series provide the parallel load capacitance required for precise tuning of the quartz crystal. The network shown in *Figure 6* provides greater than 100 ppm tuning range when used with standard X-Y flexure quartz crystals trimmed for $C_L$ = 13 pF. Tuning to better than 2 ppm is easily obtainable. The 32 Hz output can be used to monitor the oscillator frequency during the initial trimming without disturbing the network itself.

## Functional Description (Continued)

**Display Control:** The clock can display real time (real hour and minutes), alarm time (alarm hour and minutes), seconds, and sleep time (sleep minutes) under the control of the "Display", "Sleep", "Fast" and "Slow" switch inputs. The hour and the alarm hour are displayed in digit positions 1 and 2; the minutes, alarm minutes, seconds and sleep minutes are displayed in digits 3 and 4. Colon will be ON or flashing depending on the logical state of the "Colon Select" input for all modes, and OFF when alarm time is displayed. Hour and Alarm Hour can be in either 12-hour format with PM indicator or 24-hour format under the control of "12/24 hour" input. Leading zero values of Hour and Alarm Hour are blanked. Depressing the Display switch once while the clock is in real time display will cause the alarm time to be displayed and will return to real time display after 5 seconds. Depressing Slow and/or Fast while the clock is in real-time display mode will cause the seconds to be displayed for as long as the said switch is held and will return to real time display when the switch is released.

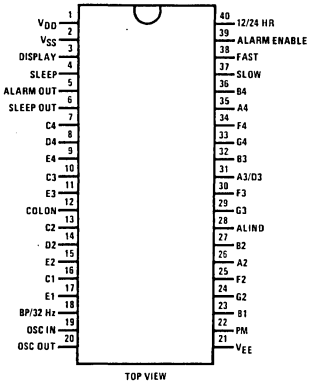



Do not use pads without function assigned

**FIGURE 2. Chip Pad Layout**

## Functional Description (Continued)

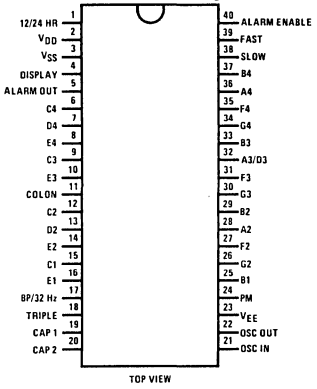
Dual-In-Line Package




**Order Number MM58143N**
**See Package 24**

**FIGURE 3a. MM58143 Clock-Radio Connection Diagram**

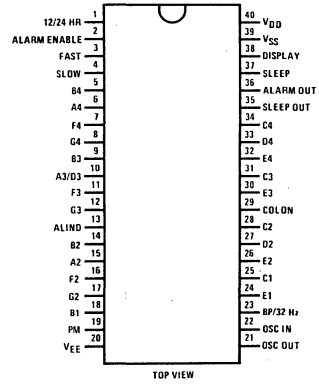
Dual-In-Line Package




**Order Number MM58144N**
**See Package 24**

**FIGURE 3b. MM58144 Travel Alarm Clock Connection Diagram**

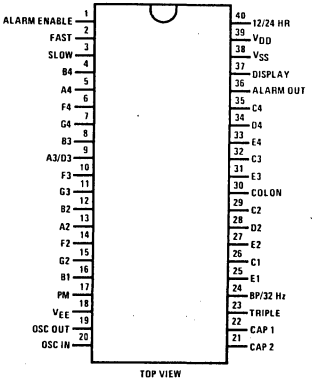
Dual-In-Line Package




**Order Number MM58183N**
**See Package 24**

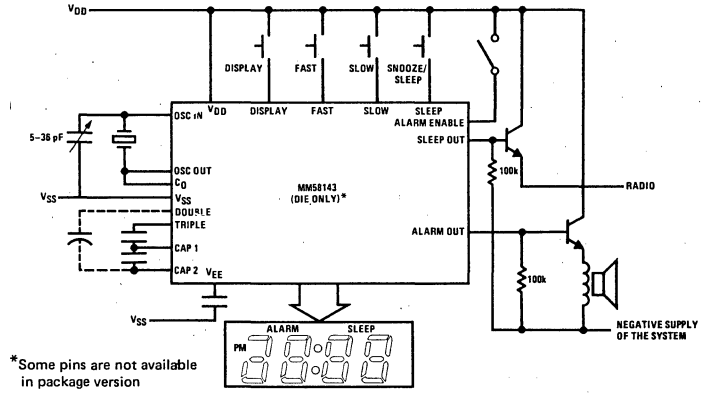
**FIGURE 3c. MM58183 Clock-Radio Connection Diagram**

Dual-In-Line Package




**Order Number MM58184N**
**See Package 24**

**FIGURE 3d. MM58184 Travel Alarm Clock Connection Diagram**




*Some pins are not available in package version

**FIGURE 4. Typical Application**



5

## Functional Description (Continued)

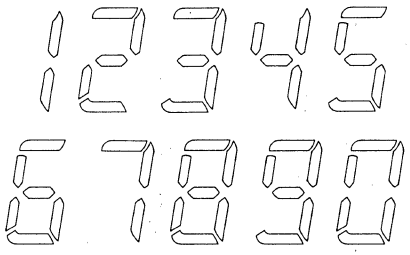


FIGURE 5. Character Font

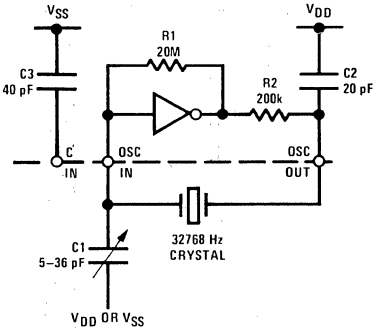



FIGURE 6. Oscillator Network

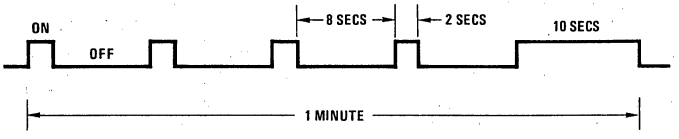



FIGURE 7. Alarm Output Waveform

**Alarm Setting:** Depressing the Display switch once while the clock is in real time display will cause the alarm hour and minutes to be displayed. Depressing the Fast switch or both Fast and Slow switches will advance the alarm time at 32 minutes per second until the switches are released. Depressing the Slow switch alone will advance the alarm time at 2 Hz rate until the switch is released. The clock will return to real time display in 5 seconds after the alarm time setting.

**Time Setting:** Depressing the Display switch once while the clock is in alarm time display will cause the real time to be displayed with hour and minutes flashing at 1 Hz rate indicating time setting is enabled. Depressing the Fast switch or both Fast and Slow switch will advance the real time at 32 minutes per second until the switches are released. Depressing the Slow switch alone will advance the real time at 2 Hz rate until the switch is released. The clock will be in a hold mode whenever the real time has been set, in which case, the seconds counter is reset and held at zero; the colon will change to the opposite mode, i.e., flashing colon with real time display will change to fixed colon or vice versa. During the time setting (Fast or Slow is depressed) the display stops flashing, but the time is advancing at the setting rate. Depressing Display switch once while the clock is in time setting mode or hold mode will return the clock to real time display and free the hold mode. This makes it possible to easily synchronize seconds exactly. Real and alarm time display and setting controls are further explained by the control state diagram shown in *Figure 8*.

**Sleep Timer and Output:** The sleep output can be used to turn off a radio (or other appliance) after a desired interval of up to 59 minutes. The time interval is chosen by selecting the sleep display mode and setting the desired time interval. This automatically turns on a current-source sleep output. When the sleep counter which counts downwards, reaches 00 minutes, a latch is reset and the sleep output current drive is removed. The sleep output can be turned off by depressing "Slow" or "Fast" switches while the clock is in real time display. The sleep output will also be turned on for 59 minutes whenever the alarm output turns on.

**Alarm Output:** The current-source alarm output will turn on at the preset alarm time if "Alarm Enable" is at a logical "1" state. The alarm signal can be either a tone or a DC level output depending on the logical state of the "Alarm Tone" input. The tone output signal is 1024 Hz gated with 2 Hz. The alarm tone will be ON and OFF periodically for 1 minute as shown in *Figure 7* to conserve power. If the "Slow", "Fast" or "Display" switch is pushed during this minute, the whole cycle as indicated will repeat seven (7) minutes thereafter (Snooze). The sleep out will also be OFF for 7 minutes, then ON for another 59 minutes.

## Functional Description (Continued)

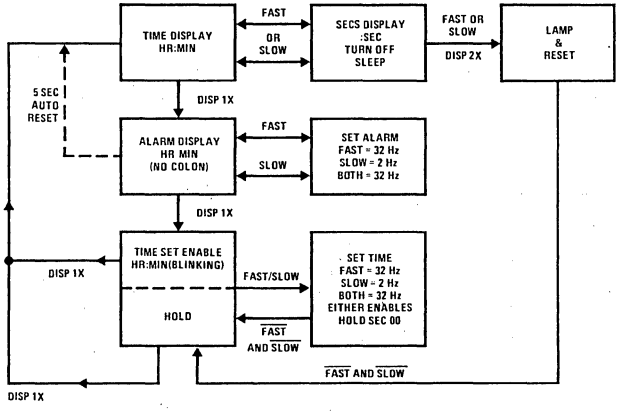



FIGURE 8. Time and Alarm Display Setting Control State Diagram

**Sleep Timer Setting:** Depressing the Sleep switch while the clock is in real time display will cause the content of the sleep counter to be displayed (if sleep counter is at 00, it will reset to 59 immediately) until the switch is released. Holding the switch closed for longer than 2 seconds will decrement the sleep counter by ten at a 2 Hz rate (i.e., 59, 49, etc.) until the switch is released. However, if Fast or Slow is also depressed at the same time, the unit portion of sleep counter will decrement at a 2 Hz rate until all switches are released. The clock will return to real time display after all switches have been released for 2 seconds.

All snooze and sleep control features are further described by the control state diagram of *Figure 9.*

**Indicator Outputs:** a) alarm indicator (ALIND) will be flashing when alarm time is displayed. It will be ON and fixed when alarm enable is at logical "1" state. This output is not available for MM58144 and MM58184 packaged units, b) sleep indicator (SLIND) will be flashing when sleep minutes is displayed. It will be ON and fixed when sleep output is ON. This output is not available for all packaged versions, c) PM indicator will be on for PM time in 12-hour mode for both real time and alarm time.

**Power-ON Reset:** When power is first applied, the internal power-ON reset signal will reset and hold the real time to 1:00 AM and alarm time to 1:01 AM. The real time will be displayed with fix colon. Depressing the Display switch will free the clock into normal running mode.

**Test Features:** a) master reset and lamp test. Depressing the Display switch twice while either the Slow or Fast switch is held in will cause all segments and indicators to turn on and also reset real time to 1:00 AM and alarm time to 1:01 AM. The clock will return to Hold mode when Fast and Slow are released and resume running after Display is depressed once. This feature will enable fast check of all segment interconnections and set the clock to a known state. b) backplane/32 Hz (BP/32 Hz) output can be used also as an input to speed up functional testing.

### SUMMARY OF CONTROL INPUTS

Each of the following inputs has a pull-down resistor to $V_{SS}$.

| | $V_{SS}$ OR OPEN | $V_{DD}$ |
|---|---|---|
| 12/24 Hour | 12-Hour Mode | 24-Hour Mode |
| Alarm Tone | Tone | DC |
| Alarm Enable | Alarm OFF | Alarm Enable |
| Colon Select | Flashing Colon | Fix Colon |

**Contact Bounce:** Debounce circuitry is provided on the Display, Slow, Fast and Sleep inputs to remove any logic uncertainty upon either closure or release of switches provided switch bounce settles within 125 ms (8 Hz debounce frequency).

**Segment Outputs:** The segment outputs are designed to drive field-effect liquid crystal displays. Each display segment has its own output which furnishes the proper 32 Hz drive signal. By definition, the segment is OFF when its drive signal is in phase with the display backplane signal (BP/32 Hz). The segment is ON when its drive signal is 180° out of phase with the display backplane signal. Typical output waveforms are shown in *Figure 11.*

## Functional Description (Continued)

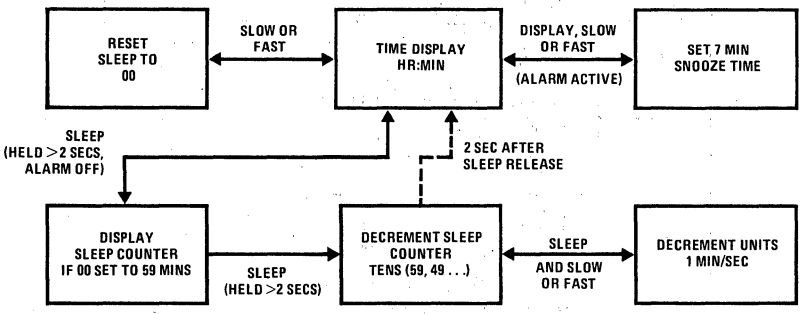



FIGURE 9. Snooze and Sleep Control State Diagram

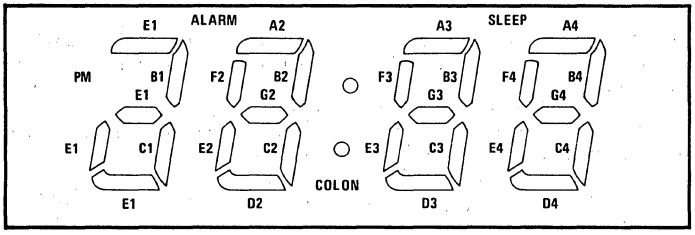



FIGURE 10. Display Format

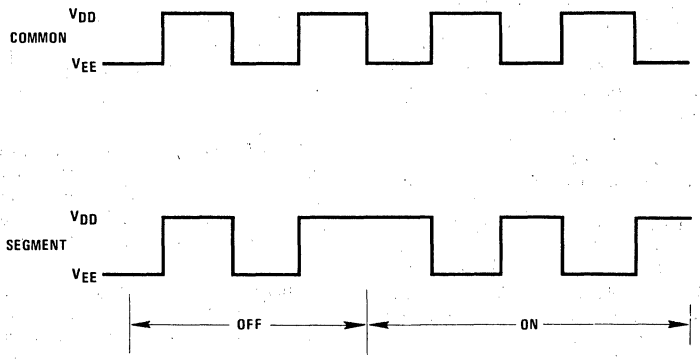



FIGURE 11. Common and Segment Output Signals

National Semiconductor

**Digital Clocks**
**Preliminary**

# MM7317B, MM7318B Alarm Clock Calendar

## General Description

The MM7317B, MM7318B digital alarm clocks are monolithic MOS integrated circuits utilizing P-channel low-threshold enhancement mode and ion implanted depletion mode devices. Featuring a 4-year calendar, each chip includes all the logic required to build several types of clocks and timers with up to 5 display modes: HOUR and MINUTE, MINUTE and SECONDS, SLEEP DELAY, ALARM SETTING and CALENDAR DISPLAY. Setting is done via the standard Fast/Slow Set buttons when the chip is in the various display modes. They are specifically intended for clock-radio applications as radio noise interference commonly associated with multiplexed display drives is eliminated by direct drive output buffers. 0.3" to 1.0" LED or fluorescent 7-segment displays can be used without buffering. Through hardware selection, users can have 12 HR with AM/PM indicator or 24 HR display format, 50 or 60 Hz input for the timekeeping function. Outputs also include the sleep (e.g. timed radio turn-off), 1 Hz activity indication and alarm output. A snooze feature is provided for a 9-minute repeat of the alarm after it has sounded.

The chips operate over a wide supply range and low standby operating voltage facilitates use of a battery back-up in case of line voltage failure, whose occurrence is indicated by a special display mode. The MM7318B incorporates, in addition, on-chip 50/60 Hz line frequency back-up oscillator, 900 Hz alarm tone generator, a 9-minute delayed alarm mode and loss of line frequency indication.

## Features

- 50/60 Hz operation
- 12/24 hour display mode
- AM/PM outputs
- 24 hour alarm set
- 9-minute snooze alarm
- Fast/Slow set
- Direct LED, VF drive
- Single power supply
- Power failure indication
- 4 year calendar
- Presettable 59-minute, sleep timer
- Battery back-up
- No illegal time display at Power-ON
- Sink/source current to radio
- Display blanking (MM7317B)
- Separate alarm and radio (sleep) outputs for simple off-chip mode switching
- 1 Hz activity indicator
- On-chip line frequency back-up oscillator (MM7318B)
- On-chip alarm tone output (MM7318B)
- Line frequency failure indication (MM7318B)
- 3 alarm modes (MM7318B): Alarm ON, Radio ON, Alarm ON 9-minute after Radio ON.

## Applications

- Alarm clock-calendars
- Desk clock-calendars
- Clock radios
- Automobile clocks
- Stop watches
- Industrial clocks
- Sequential controller
- Portable clocks
- Industrial timers
- Appliance timers
- Photography timers

## Block Diagram

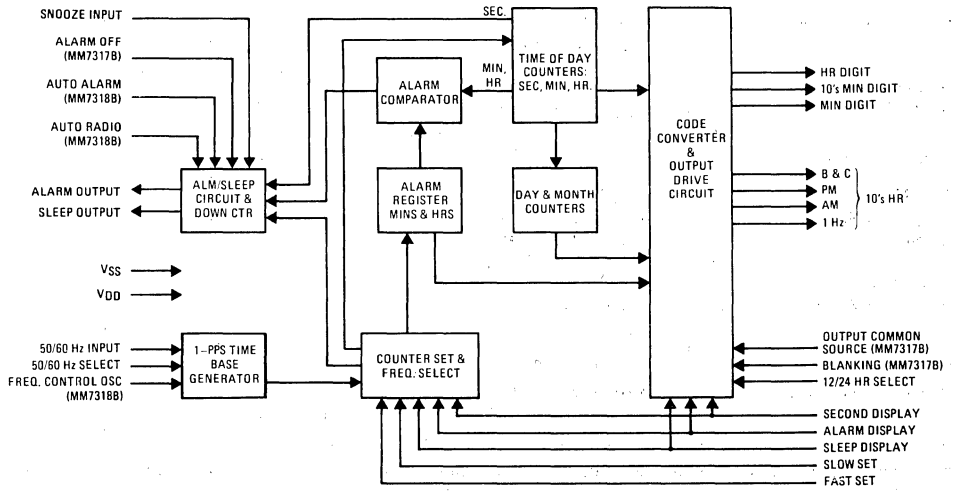



**Figure 1.**

## Absolute Maximum Ratings (Exceeding the following ratings may result in permanent damage to the device).

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ +0.3V to $V_{SS}$ −30V |
| Operating Temperature (ambient) | 0° to +70°C |
| Storage Temperature | −55°C to 150°C |
| Lead Temperature (Soldering, 10seconds) | 300°C |

## Electrical Characteristics

(MM7318B values in brackets when they are different from those of MM7317B)
$T_A$ within operating range, $V_{DD}$ = 0V, $V_{SS}$ = 8 to 30V ($V_{SS}$ = 7.5 to 28V)
Unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage | Operating | 8 (7.5) | | 30 (28) | V |
| | Functional Clock | 7 (7.5) | | 30 (28) | V |
| Power Supply Current | No Output Loads | | | | |
| | $V_{SS}$ = 7V (7.5V) | | | 4 (5) | mA |
| | $V_{SS}$ = 30V (28V) | | | 5 (6) | mA |
| 50/60 Hz Input Frequency | | DC | 50/60 | 15K | Hz |
| 50/60 Hz and Blanking Input Voltages | | | | | |
| Logical High Level | | $V_{SS}$ − 1 ($V_{SS}$ − 1.5) | | $V_{SS}$ | V |
| Logical Low Level | | $V_{DD}$ | | $V_{SS}$ − 6 ($V_{SS}$ − 5) | V |
| Frequency Control Input | | | | | |
| Input Current | $V_{IN}$ = $V_{SS}$ − 1:8V, $T_A$ = 25°C | (0.1) | | (1.1) | mA |
| Frequency OSC | C = 0.005 μF, R Adjusted | (DC) | | (5K) | Hz |
| OSC Frequency Variation | For 1800 Hz at $V_{SS}$ = 9V | (− 14) | | (+ 14) | % |
| All Other Input Voltages | 7.5V ⩽ $V_{SS}$ ⩽ 28V | | | | |
| Logical High Level | | $V_{SS}$ − 1.5 | | $V_{SS}$ | V |
| Logical Low Level | | $V_{DD}$ | | $V_{SS}$ − 5 | V |
| Alarm Output Frequency | | | (900) | | Hz |
| Alarm/Sleep Output Current | $V_{SS}$ ⩾ 8V | | | | |
| Logical High Level | $V_{OH}$ = $V_{SS}$ − 2V | 3.5 | | | mA |
| Logical Low Level | $V_{SS}$ = $V_{DD}$ + 0.6V | 1.0 | | | μA |
| Output Current Levels | $V_{SS}$ ⩾ 24V Output Common = $V_{SS}$ | | | Note 1 | |
| 10's Hours (B/C) | | | | | |
| Logical High Level | $V_{OH}$ = $V_{SS}$ − 3.1V 12 Hour Mode | 20 | | | mA |
| | 24 Hour Mode | 10 | | | mA |
| Logical Low Level | $V_{OL}$ = $V_{SS}$ − 28V | −10 | | | μA |
| 10's of Minutes (A/D) | $V_{SS}$ ⩾ 24V | | | | |
| Logical High Level | $V_{OH}$ = $V_{SS}$ − 3.1V | 20 | | | mA |
| Logical Low Level | $V_{OL}$ = $V_{SS}$ − 28V | − 10 | | | μA |
| 1 Hz display | $V_{SS}$ ⩾ 24V | | | | |
| Logical High Level | $V_{OH}$ = $V_{SS}$ − 3.1V | 25 | | | mA |
| Logical High Level | $V_{OL}$ = $V_{SS}$ − 28V | − 10 | | | μA |
| All Other Displays | $V_{SS}$ ⩾ 24V | | | | |
| Logical High Level | $V_{OH}$ = $V_{SS}$ − 3.1V | 10 | | | mA |
| Logical Low Level | $V_{OL}$ = $V_{SS}$ − 28V | − 10 | | | μA |

**Note 1:** Output current must be limited so that total device power with loads does not exceed 750 mW at 70°C and 1.4W at 25°C.

## Functional Description

A block diagram of the MM7317B, MM7318B digital clock calendar circuit is shown in *Figure 1*. The various display setting modes are listed in Table 1 and Table 2 shows the setting control functions. The following decription is based on *Figure 1* and refers to both devices unless otherwise specified.

**50 or 60 Hz Input:** A simple RC filter should be used to remove possible line voltage transients that could either cause the clock to gain time or damage the device. A schmidt trigger circuit of 2V hysteresis is provided to allow the use of sinewave input and its output drives a counter chain that performs the timekeeping function.

**50 or 60 Hz Select Input:** A programmable prescale counter divides the input frequency to obtain a 1 Hz time base. 50 Hz operation is selected by connecting this input to $V_{SS}$. Leaving this input unconnected selects 60 Hz operation. Pull down to $V_{DD}$ is provided by an internal depletion load (typical 100KΩ).

**Display Mode Select Inputs:** (Refer to Table I) Alternate display modes are selected by applying $V_{SS}$ to the appropriate pins. An internal pull-down resistor (typical 100KΩ) allows use of simple SPST switches to select the display mode. In the absence of any of these inputs, the display drivers present time of day information. If more than one mode is selected, the priorities are as noted in Table I. Constant calendar display is possible by tying both alarm display and seconds display inputs to $V_{SS}$. This mode has an automatic interrupt from calendar to real time when the alarm output is enabled.

**Time Setting Inputs:** (Refer to Table II) Both Fast and Slow Setting inputs are provided. These inputs are applied either singly or in combination to obtain the control function listed in Table II. Again, internal pull-down resistors (typical 100KΩ) are provided; application of $V_{SS}$ to these pins effects the control functions.

**12 or 24 HR Select Input:** By leaving this pin unconnected, the outputs for the most significant display digits (10's of hours) are programmed to provide a 12 hour display format. An internal pull-down resistor (typical 100KΩ) is provided. Connecting this pin to $V_{SS}$ programs the 24 hour display format. The output connections are different for each format as illustrated in *Figure 2*.

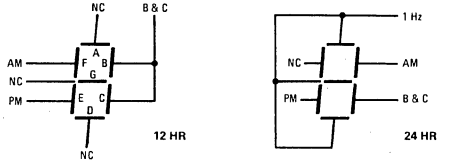



**Figure 2. 10's of Hours Digit Wiring Options**

**Snooze/Calendar Mode Input:** Momentarily connecting this pin to $V_{SS}$ inhibits the alarm output for between 8 and 9 minutes, after which the alarm will again be sounded. This input is pulled down to $V_{DD}$ by an internal resistor (typical 100KΩ). The snooze feature may be repeatedly used during the time (45 minutes for MM7317B, and 59 minutes for MM7318B) in which the alarm latch remains set. Connecting this pin to $V_{SS}$ also enables the calendar display mode as shown in Table I.

**Alarm Modes (MM7318B):** There are 3 alarm modes which are auto alarm, auto radio and delayed alarm. Internal pull-down resistors (typical 100KΩ) allow use of simple SPST switches to select alarm modes. Selection is effected by connecting the corresponding input pin to $V_{SS}$. In the auto radio mode, the radio output is enabled when real time is coincident with alarm time. In the auto alarm mode, the alarm output is enabled when real time is coincident with alarm time. If both auto radio and auto alarm modes are selected, a third mode results. This is the delayed alarm mode in which the radio output is enabled when real time is coincident to the alarm time. 8 to 9 minutes later the alarm output is enabled and snooze input only affects the alarm output. To reset the alarm mode, connect the selection pins to $V_{DD}$.

**Frequency Control Input (MM7318B):** A resistor in parallel with a capacitor is typically connected from this pin to $V_{SS}$. The frequency of oscillation is divided by 2 to give the alarm tone. The frequency is further divided by 18 for 50 Hz operation or by 15 for 60 Hz operation. The frequency coming out of the divider chain is used as a backup frequency when there is a loss of input line frequency. For accurate backup operation, the oscillator frequency must be set to 1800 Hz by adjustment of the externally connected resistor/capacitor at this pin.

**Blanking Control Input (MM7317B):** Connecting this schmidt trigger input to $V_{DD}$ places all display drivers in a non-conducting, high-impedance state, thereby inhibiting the display. Conversely, $V_{SS}$ applied to this input enables the display. This pin has a dual application. Pulse percentage modulation can offer display brightness control. The input can also be used as a display chip select.

**Alarm Off Input (MM7317B):** Momentarily connecting this pin to $V_{SS}$ resets the alarm latch and thereby silences the alarm. This input is returned to $V_{DD}$ by an internal resistor (typcial 100KΩ). The momentary alarm off input also enables the alarm latch for the next comparator output and the alarm will automatically sound again in 24 hours (or at a new alarm setting). If it is desired to silence the alarm for a day or more, this pin should remain at $V_{SS}$.

**Sleep Output:** The sleep output may be used to turn off a radio or other device after a desired time interval of up to 59 minutes. The time interval is chosen by selecting the sleep display mode and setting the desired time interval (Table II). This automatically results in a current source sleep output which may be used to turn on a radio or other device. When the sleep counter which counts downwards, reaches 0 minutes, the sleep output current drive is removed, thereby turning off the device. This turning off may also be manually controlled (at any time in the countdown) by a momentary $V_{SS}$ connection to the snooze input.

## Functional Descriptions (Continued)

**Table I. DISPLAY MODES**

| *Selected Display Mode | Digit # 1 | Digit # 2 | Digit # 3 | Digit # 4 |
|---|---|---|---|---|
| Time Display | 10's of Hours and AM/PM | Hours | 10's of Minutes | Minutes |
| Seconds Display | Blanked | Minutes | 10's of Seconds | Seconds |
| Alarm Display | 10's of Hours and AM/PM | Hours | 10's of Minutes | Minutes |
| Sleep Display | Blanked | Blanked | 10's of Minutes | Minutes |
| Calendar Display 12 Hour Mode | 10's of Months or Blanked | Months | 10's of Days or Blanked | Days |
| 24 Hour Mode | 10's of Days or Blanked | Days | 10's of Months or Blanked | Months |

*If more than one display mode input is applied, the display priorities are in order of Sleep (overrides all others), Calendar, Alarm, Seconds, and Time.

**Table II. SETTING CONTROL FUNCTIONS**

| Selected Display Mode | Enable Via | Control Input | Control Function |
|---|---|---|---|
| *Time | | Slow | Minutes advance at 2 Hz rate. |
| | | Fast | Minutes advance at 60 Hz rate. |
| | | Both | Minutes advance at 60 Hz rate. |
| Alarm | Alarm Display | Slow | Alarm minutes advance at 2 Hz rate. |
| | | Fast | Alarm minutes advance at 60 Hz rate. |
| | | Both | Alarm resets to 12:00 AM (12-hour format). |
| | | Both | Alarm resets to 00:00 (24-hour format). |
| Seconds | Seconds Display | Slow | Input to entire time counter is inhibited (hold). |
| | | Fast | Seconds and 10's of seconds reset to zero without a carry to minutes. |
| | | Both | Time resets to 12:00 AM (12-hour format). |
| | | Both | Time resets to 00:00:00 (MM7318B), – 0:00:00 (MM7317B) (24 hour format). |
| Sleep | Sleep Display | Slow | Subtracts count at 2 Hz. |
| | | Fast | Subtracts count at 60 Hz. |
| | | Both | Subtracts count at 60 Hz. |
| Date | Snooze | Slow | Day advances at 2 Hz rate. Month advances after proper number of days. |
| | Alarm (MM7318B) Sleep (MM7317B) and Seconds Display | Fast | Day advances at 60 Hz rate. Month advances after proper number of days. |

*When setting time, sleep minutes will decrement at rate of time counter until the sleep counter reaches 00 minutes (Sleep counter will not recycle).

## Functional Descriptions (Continued)

**Display Outputs:** The 1 Hz display output is normally used to drive the colon of the associated display. Refer to Table III. for output drive capability.

Table III. OUTPUT BUFFER DRIVE CAPABILITY

| Display | 10's Hrs | 10's Mins | 1 Hz | AM | PM |
|---|---|---|---|---|---|
| 12 Hour | 2 | 2 | 3 | 1 | 1 |
| 24 Hour | 1 | 2 | 3 | 1 | 1 |

All other buffers have 1 segment drive capability.

**Power Fail Indication**

**(MM7317B):** This indication consists of all on segments flashing simultaneously at a 1 Hz rate. A Fast or Slow Set input resets an internal power failure latch and returns the display to normal. During a power fail condition, the alarm setting may become invalid. In this case, the alarm output is inhibited.

**(MM7318B):** There are two different power fail indications Frequency Fail and Voltage Fail. Frequency failure will occur when the 50/60 Hz generated by the internal oscillator frequency counts 4 cycles without the line frequency making a positive transition. While line frequency is missing, display will be blanked, and time and alarm functions will continue to operate from the internal oscillator frequency evenly divided down to 50/60 Hz. Auto alarm will override auto radio and delayed alarm. Frequency failure is self-clearing, except for the frequency fail indication, when line frequency is restored. Frequency failure is indicated by the entire display flashing at a 1 Hz rate.

Voltage failure will occur when $V_{DD}$ drops below a minimum operating voltage into the detect range. During low voltage failure, the display is blanked and the time functions are reset and held in that state. Sleep counter is reset to 59 and seconds to 00. Alarm time and time are reset to 12:00 AM (12 Hour Mode) or 00:00 (24 Hour Mode). Date is reset to 12 in month units and 1 in day units.

Also alarm and sleep outputs are inhibited by the voltage failure. When voltage is returned, voltage failure is indicated by 000 on the display, and all counters hold at the above reset values.

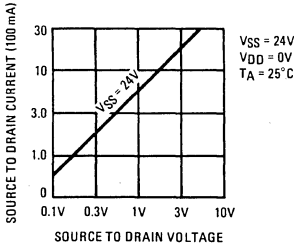
A fast or slow input will reset both voltage and frequency failure latches and returns the chip to normal operation.

**Alarm Operation and Output:** The alarm comparator senses coincidence between the alarm counters (the alarm setting) and the time counters (real time). The comparator output is used to set latch(es) in the alarm and sleep circuits.

**(MM7317B):** One latch is set whose output enables the alarm output driver that is used to control an external alarm sound generator. During the 45 minutes that the latch remains set, the alarm will sound if the latch output is not temporarily inhibited by another latch set by the snooze input or reset by the alarm off input. During a power fail detect condition, the alarm circuitry may function as normal. However the power voltage may drop enough so that the alarm circuitry inhibits its output. In this case, the alarm has to be reset by going to alarm set mode via the fast or slow set, resetting the alarm time.

**(MM7318B):** Two latches are set to enable the alarm output or sleep output followed by delayed alarm output, depending on which mode was selected. The sleep output provides a DC transition from $V_{DD}$ to $V_{SS}$ at the alarm time, which can be used to turn on a radio or other device. The alarm output is a square wave at the frequency of ½ of the on-chip oscillator frequency. If the frequency oscillator input pin is returned to $V_{DD}$, there will be no backup oscillator and the alarm output will be a DC transition from $V_{DD}$ to $V_{SS}$. The alarm latch remains set for 59 minutes. During this time the alarm will sound if the latch output is not temporarily inhibited by another latch set by the snooze input or reset by auto alarm and auto radio inputs. If snooze is enabled while an alarm mode output is enabled, the contents of the alarm counter may be altered, multiple use of the snooze function may cause the total alarm enable time to be less than 59 minutes.

**Output Common Source Connection (MM7317B):** All display output drivers are open drain devices with all sources common to this pin. This feature allows use of either common cathode or common anode LED displays. When using fluorescent tube displays, $V_{SS}$ or a display brightness control voltage may be connected to this pin. The common source connection also facilitates generating AC drive voltages when liquid crystal displays are used.



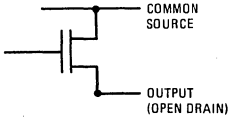




**Figure 3. Typical Single Segment Output Buffer**

## Typical Applications

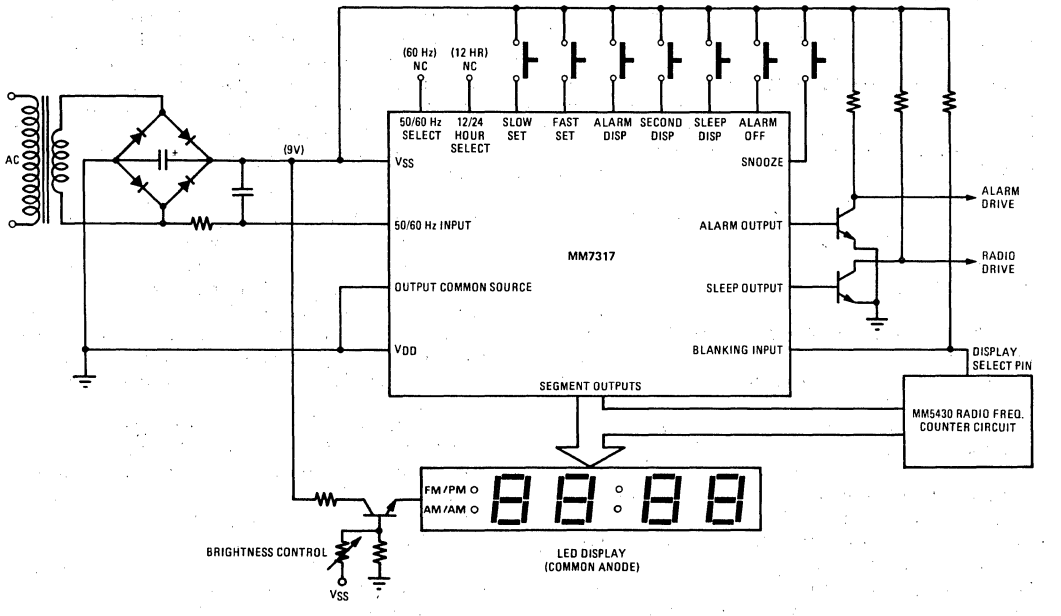



Figure 4. Typical Application (MM7317BN): Clock/Calendar with Digital Frequency Readout for Radio

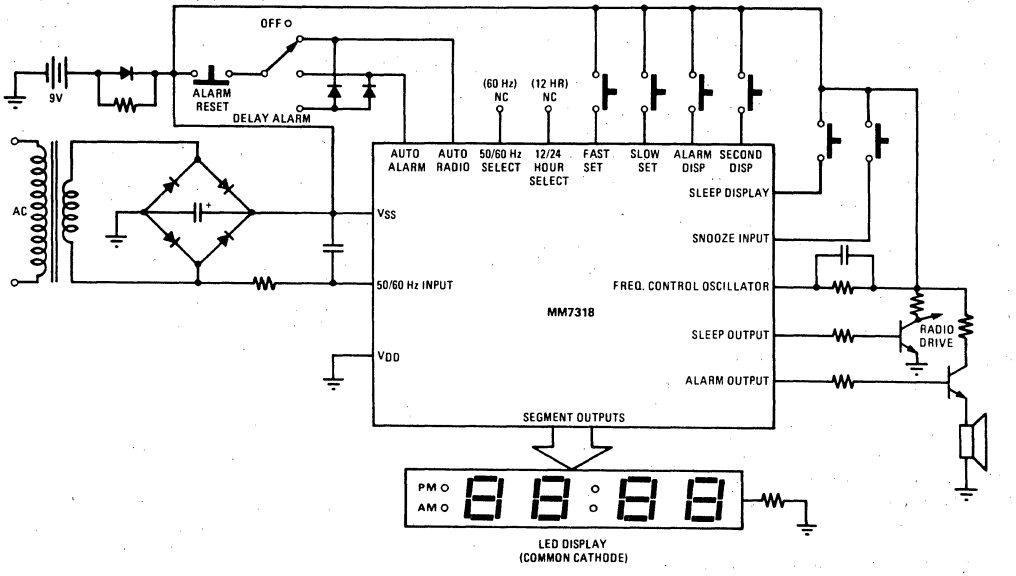
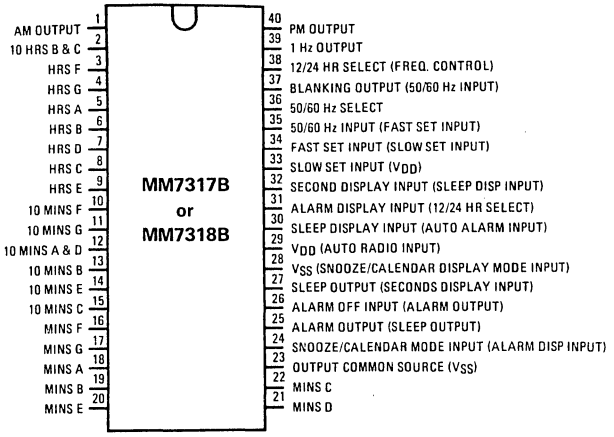



Figure 5. Typical Application (MM7318BN): Clock/Calendar with Battery Back-up

## Connection Diagram Dual-In-Line Package

**TOP VIEW**




**( ) indicates pin connection for MM7318B**

**Order Number MM7317BN**
**or MM7318BN**
**See Package 24**

# Television/Radio

# MM5321 TV camera sync generator

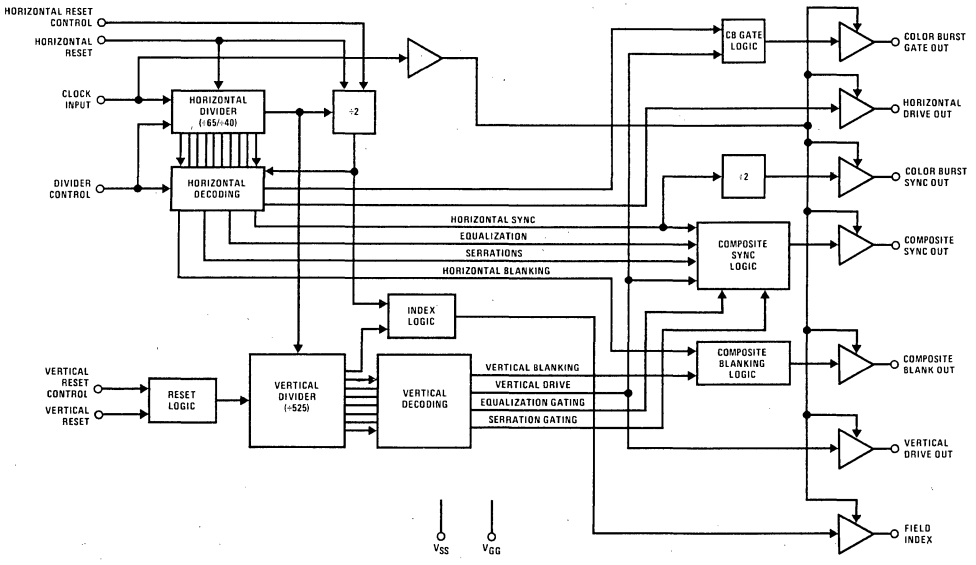
## general description

The MM5321 TV camera sync generator is a MOS, P-channel enhancement mode, LSI chip designed to supply the basic sync functions for either color or monochrome 525 line/60 Hz interlaced camera and video recorder applications. Required power supplies are +5V and −12V, or any other combination resulting in $V_{SS}$ − 17V. All inputs and outputs are TTL compatible without the use of external components.

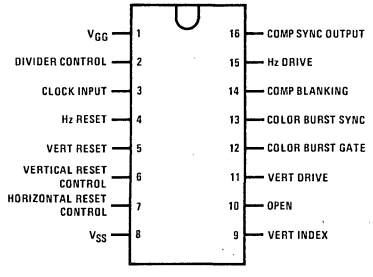
## features

- Multi-function gen lock input provides flexible control of multiple camera installations
- 16-lead dual-in-line package
- Conventional +5V, −12V power supplies
- Uses 2.04545 MHz or 1.260 MHz input reference
- Field indexing provided for VTR applications
- Color burst gate and sync allow stable color operation

## logic and connection diagrams




**Dual-In-Line Package**




TOP VIEW

**Order Number MM5321N**
**See Package 19**

## absolute maximum ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ + 0.3 to $V_{SS}$ − 22 |
| Operating Temperature | 0°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## dc electrical characteristics

$T_A$ within operating temperature range $V_{SS}$ = 5V ±5%, $V_{GG}$ = −12V ±5%, unless otherwise stated.

| | PARAMETER | CONDITIONS | MIN | MAX | UNITS |
|---|---|---|---|---|---|
| | Input Levels | | | | |
| $V_{IH}$ | Logical High Level | | $V_{SS}$−1.5 | $V_{SS}$+0.3 | V |
| $V_{IL}$ | Logical Low Level | | $V_{SS}$−18 | $V_{SS}$−4.2 | V |
| | Input Leakage | $V_{IN}$ = −10V, $T_A$ = 25°C, All Other Pins GND | | 0.5 | μA |
| | Input Capacitance | $V_{IN}$ = 0V, f = 1 MHz, All Other Pins GND, (Note 1) | | 6 | pF |
| | Clock Input Leakage | $V_{IN}$ = −10V, $T_A$ = 25°C, All Other Pins GND | | 0.5 | μA |
| | Clock Input Capacitance | $V_{IN}$ = 0V, f = 1 MHz, All Other Pins GND, (Note 1) | | 6 | pF |
| | Output Levels | | | | |
| $V_{OH}$ | Logical High Level | $I_{SOURCE}$ = −0.5 mA | 2.4 | $V_{SS}$ | V |
| $V_{OL}$ | Logical Low Level | $I_{SINK}$ = 1.6 mA | | 0.4 | V |
| | | MOS Load | $V_{SS}$−12.5 | $V_{SS}$−9 | V |
| $I_{GG}$ | Power Supply Current | $T_A$ = 25°C, $V_{GG}$ = −12V, $\phi_{PW}$ = 235 ns, $V_{SS}$ = 5V, Input Clock Frequency = 2.04545 MHz | | 36 | mA |

## ac electrical characteristics

$T_A$ within operating temperature range $V_{SS}$ = 5V ±5%, $V_{GG}$ = −12V ±5%, unless otherwise stated.

| | PARAMETER | CONDITIONS | MIN | MAX | UNITS |
|---|---|---|---|---|---|
| $\phi_{PW}$ | Input Clock Pulse Width | Input Clock Frequency = 2.04545 MHz, $\phi t_r$, $\phi t_f$ = 20 ns | 190 | 280 | ns |
| | | Input Clock Frequency = 1.26 MHz, $\phi t_r$ = $\phi t_f$ = 20 ns | 300 | 570 | ns |
| | Horizontal Reset Pulse Width | Within 400 ns after the Falling Edge of Master Clock, *(Figure 5)* Rise and Fall Time = 20 ns | 500 | 800 | ns |
| $t_{pd}$ | Output Propagation Delay | | | | |
| $V_{OH}$ | Logical High Level | Capacitance at the Output = 15 pF | | 750 | ns |
| $V_{OL}$ | Logical Low Level | *(Figure 5)* | | 750 | ns |

**Note 1:** Capacitance is guaranteed by periodic testing.

## functional description

### EXTERNAL CONTROL LEVELS

Horizontal Reset occurs for Logic "0." This resets the horizontal counter to a state shown in *Figures 2 and 3.*

Vertical Reset occurs for Logic "0." This resets the vertical counter to a state determined by reset control input as shown below:

| VERTICAL RESET CONTROL INPUT | PERMITS THE VERTICAL COUNTER TO RESET TO THE: |
|---|---|
| $V_{IH}$, ($V_{SS}$) | 0th count |
| $V_{IL}$, ($V_{GG}$) | 11th count |

| HORIZONTAL RESET CONTROL INPUT | RESETS THE HORIZONTAL DIVIDER TO: |
|---|---|
| $V_{IH}$ | Beginning of line |
| $V_{IL}$ | Center of line |

Logic "0" = $V_{IL}$

Logic "1" = $V_{IH}$

Divide select input = $V_{IL}$, ($V_{GG}$) for master clock frequency of 1.26 MHz.

Divide select input = $V_{IH}$, ($V_{SS}$) for master clock frequency of 2.04545 MHz.

### INPUTS

The user may select either of two input clock frequencies by properly programming the Divider Control pin. In one case the input frequency is 2.04545 MHz, which is 14.31818 MHz divided by seven. The other is eighty times the horizontal frequency, or 1.26 MHz. The divider control will be programmed by connecting it to $V_{IH}$ ($V_{SS}$) and $V_{IL}$, ($V_{GG}$) respectively.

There are separate Vertical and Horizontal Reset inputs which allow directly resetting the appropriate divider(s) by a control pulse generated by external means. Both horizontal and vertical dividers may be reset simultaneously by connecting the Vertical and Horizontal Reset pins together and driving them with the same reset signal. Actual resetting of the vertical divider is to either of two states, depending upon the state of the Vertical Reset Control input; to zero, or to the fifth vertical serration pulse (eleven 0.5H time intervals from leading edge of Vertical Blanking). Refer to the reset table. The horizontal divider will always be reset to a position which is 8 input clock pulses from the leading edge of the serration gate in the horizontal timing scheme *(Figures 2 and 3)*. The generator is reset to the odd field (field one). The Field Index output pulse occurs once each odd field at the leading edge of Vertical Blanking. It can be used to reset, or "gen-lock," similar sync generator chips by connecting it to their Vertical and Horizontal Reset inputs. The Horizontal Reset Control selects Horizontal Reset to the start or center of a line. For "gen-lock" both Horizontal and Vertical Reset pulses should not exceed 800ns.

### OUTPUTS

The generator supplies the following standard output functions: Horizontal Drive Out, Vertical Drive Out, Composite Blanking Out, Composite Sync Out and the Color Burst Gate.

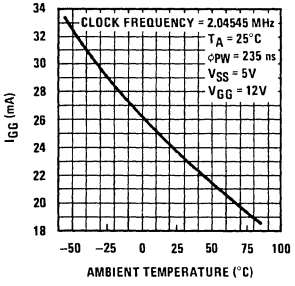
In addition, Field Index and Color Burst Sync outputs are provided. The Field Index identifies the odd field, or field one, by occurring for two clock periods at the leading edge of Vertical Blanking in that field. Thus, its rate is 30 Hz. As described above, it can also be used to "gen-lock" other sync generator chips.

The Color Burst Sync output signal occurs at half the horizontal rate with the same timing as the Color Burst Gate output. It may be used to sync the color burst as it will have the same delay characteristics as the other outputs (including, of course, the Color Burst Gate) — the color burst sync is present during the vertical interval.

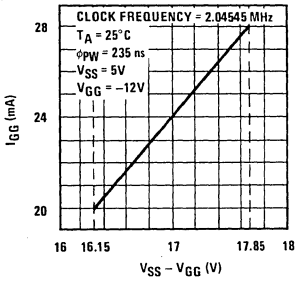
Differences in phasing between outputs are minimized by the use of identical push-pull output buffers clocked by the internal clock.

## typical performance characteristics

Typical $I_{GG}$ vs Temperature




Typical $I_{GG}$ vs Power Supply Voltage ($V_{SS} - V_{GG}$)



## switching time waveforms

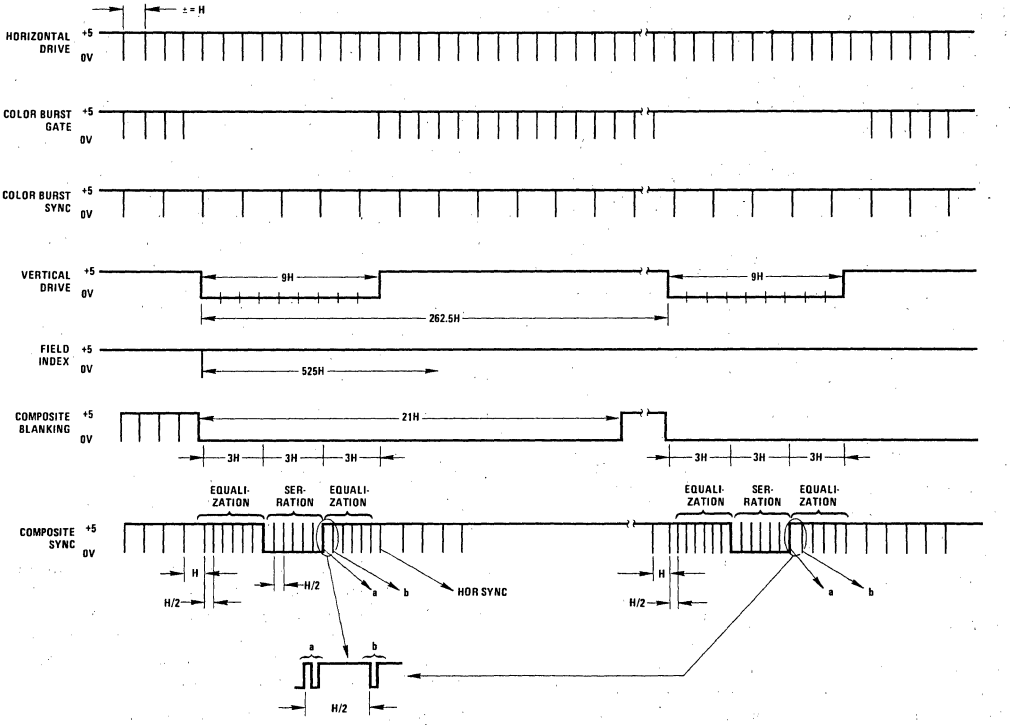



FIGURE 1.

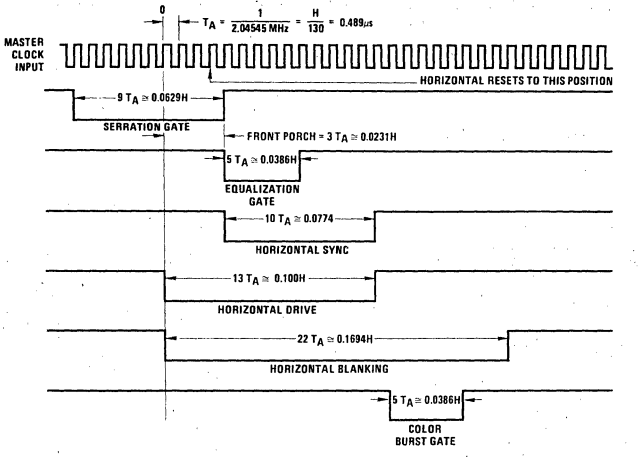



FIGURE 2. Horizontal Timing Master Clock = 2.04545 MHz

## switching time waveforms (Continued)

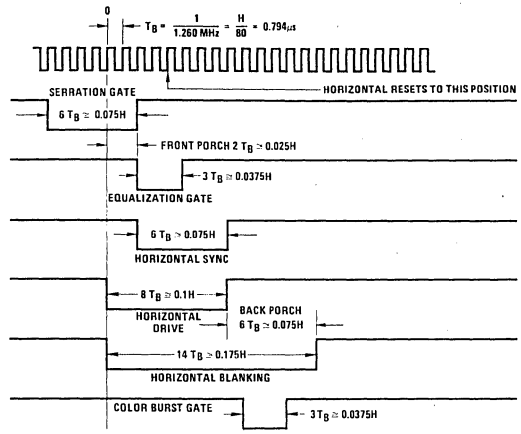



**FIGURE 3. Horizontal Timing Master Clock = 1.26 MHz**

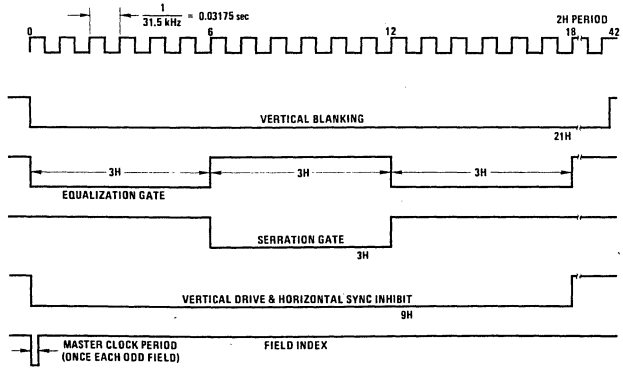



**FIGURE 4. Vertical Timing**

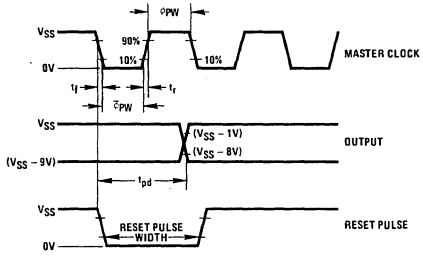



**FIGURE 5.**

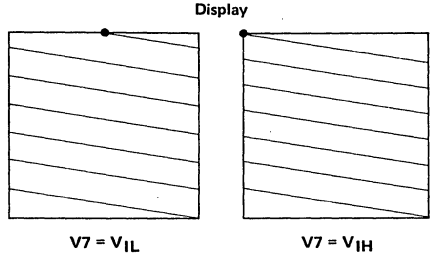
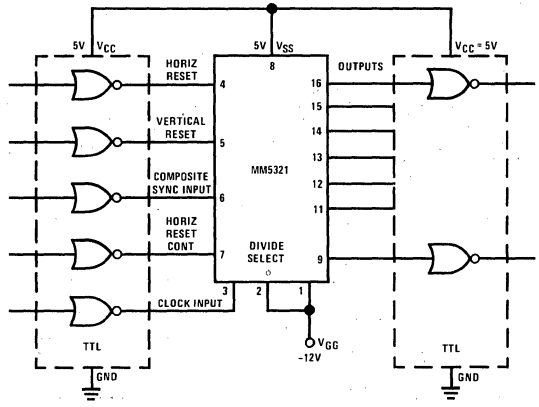



**FIGURE 6. Horizontal Reset Characteristics**

## typical application




**TTL Interface**

## MM5322 color bar generator chip

### general description

The MM5322 Color Bar Generator Chip is a complete dot-bar and color hue generation system in a single monolithic P-channel MOS integrated circuit. The chip divides an internal oscillator (crystal controlled) frequency to provide the various timing, synchronization, and video information required in the alignment of color television receivers. A composite video output is provided for complete black and white dot-bar operation. It consists of all synchronization, blanking, and video information required for a fairly standard set of dot, bar, and cross hatch screen patterns. In addition a separate output for precise gating of 3.56 MHz color bursts is provided. For servicing ease an oscilloscope trigger is provided on either the horizontal blanking or vertical synchronization time slots.

### features

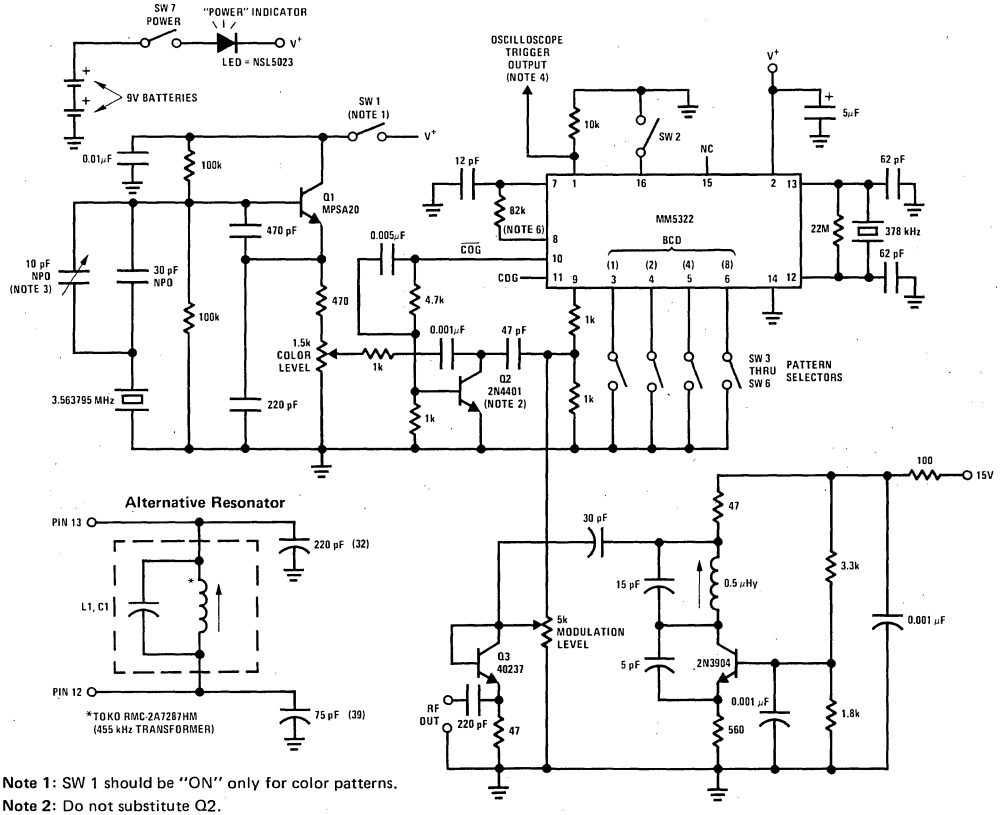
- Battery operation
- Oscilloscope trigger
- Composite video output signal
- Crystal controlled oscillator
- Multiple screen patterns
- Variable dot size

### applications

- Battery or bench powered test instruments
- Manufacturing test sets
- Built in test capability

### typical application

Typical Color Bar Generator Circuit




**Note 1:** SW 1 should be "ON" only for color patterns.
**Note 2:** Do not substitute Q2.
**Note 3:** Variable cap may be used to trim color crystal to exact frequency.
**Note 4:** SW 2 and 10k resistor on pins 16 and 1 are needed only if scope trigger pulse is desired.
**Note 5:** SW 2 selects "H" or "V" trigger output pulses.
**Note 6:** A 27k resistor in series with a 100k trimpot may be used in place of 82k resistor for variable vertical line width.
**Note 7:** Modulation level adjusted for best patterns as viewed on TV screen.

## absolute maximum ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$+0.3V to $V_{SS}$−25V |
| Operating Temperatures | −25°C to +75°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperatures (Soldering, 10 seconds) | 300°C |

## electrical characteristics $T_A$ within operating range, $V_{SS}$ = +12 to +19V, $V_{GG}$ = 0V

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage ($V_{SS}$) | | 12 | | 19 | V |
| Clock Input Frequency OSC 1 and 2 | Crystal or External Drive (Note 1) | | 378 | | kHz |
| Clock Input Levels | For External Drive (Note 1) | | | | |
| Logical High | | $V_{SS}$−2 | | $V_{SS}$+0.3 | V |
| Logical Low | | $V_{GG}$ | | $V_{GG}$+2 | V |
| Control Inputs BCD and Trigger | Internal Resistor To $V_{SS}$, 1M Ω Min. | | | | |
| Logical High | (Note 2) | $V_{SS}$−2 | | $V_{SS}$+0.3 | V |
| Logical Low | | $V_{GG}$ | | $V_{GG}$+2 | V |
| Control Output Currents Cog and $\overline{Cog}$ | | | | | |
| Logical High | $V_{SS}$ − 2.0V | 2.5 | | | mA |
| Logical Low | $V_{GG}$ − $V_{GG}$/2 (Note 3) | 0.25 | | | mA |
| Trigger and Z | | | | | |
| Logical High | With 10k to $V_{GG}$, $V_{GG}$ + 5.0V (Note 4) | 0.5 | | | mA |
| Logical High | With 1k to $V_{GG}$, $V_{GG}$ + 1 (Note 4) | 1.0 | | | mA |
| Video Output | | | | | |
| Analog Highs | With 2k to $V_{GG}$ (Note 5) | | 2.0 to 4.0 | | mA |
| Power Supply Current | $T_A$ = 25°C, Freq = 378 kHz, $V_{GG}$ = 0V, $V_{SS}$ = +19V | | | 30 | mA |

**Note 1:** The oscillator may be operated with external components to oscillate at 378 kHz or it may be driven by an external pulse source using OSC 2 (Pin 13) as an input.

**Note 2:** These inputs are driven by switches.

**Note 3:** The color gate outputs are push-pull buffers.

**Note 4:** The trigger output and Z output are open drain outputs and require a resistor to $V_{GG}$ for operation. Two possible resistor values are shown with their associated voltage and current levels.

**Note 5:** The video output requires a resistor to $V_{GG}$ for operation. This resistor must be trimmed externally to achieve the desired output levels. The minimum voltage swing is 4.0 volts with a 10% change with temperature and from unit to unit. The percentage magnitude change with supply voltage can approach one.

## composite video output

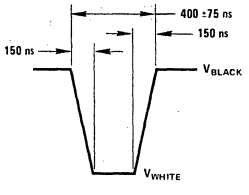



**FIGURE 1. White Dot Video Information Pulse Width**

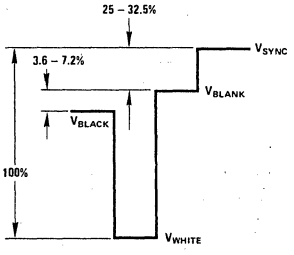



**FIGURE 2. Composite Video Voltage Percentages**

## composite video output (con't)

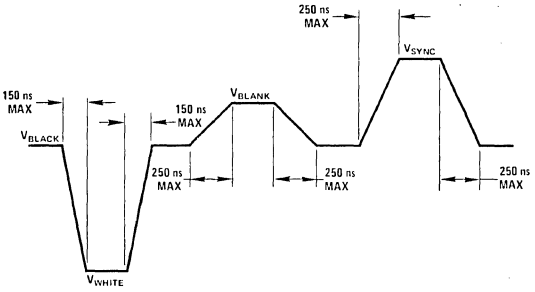
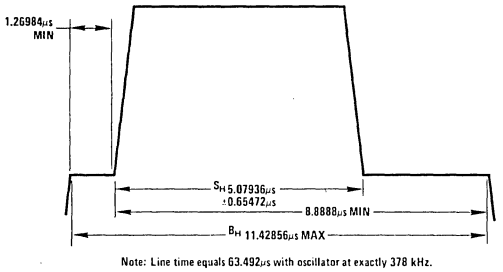
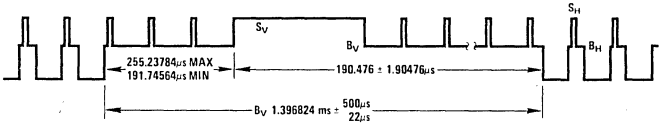



FIGURE 3. Composite Video Rise and Fall Times




Note: Line time equals 63.492μs with oscillator at exactly 378 kHz.

FIGURE 4. Composite Video Pulse Timing, Horizontal Sync




Note: Frame frequency equals 60.114665 Hz.

FIGURE 5. Composite Video Pulse Timing, Vertical Sync

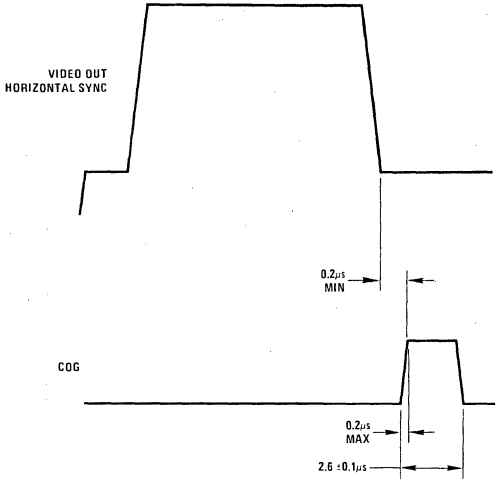



FIGURE 6. Color Gate Signal Timing

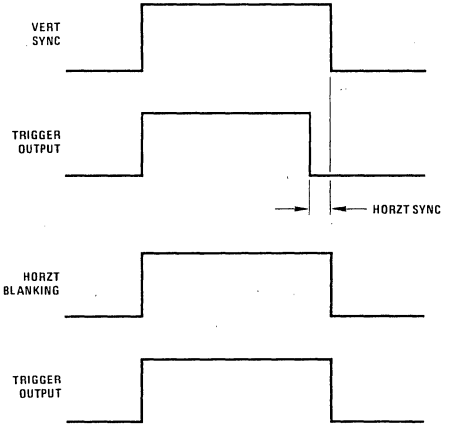
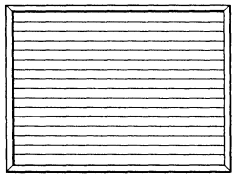
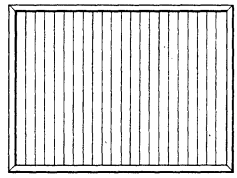



FIGURE 7. Trigger Output Timing Relationship

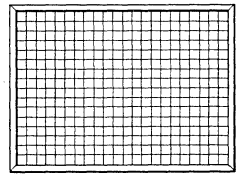
## video output patterns



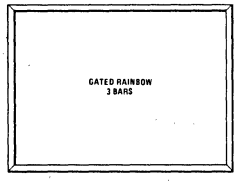
15 Horizontal Lines
0000



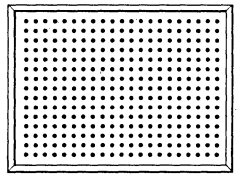
21 Vertical Lines
0001



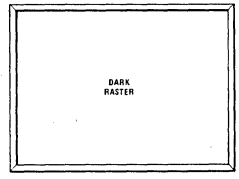
15 x 21 Cross Hatch
0010




Gated Rainbow
0011

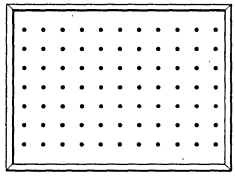


Dots 15 x 21
0100

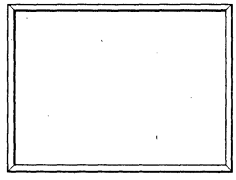



Note: Pattern switch codes are BCD 1248 positive logic.

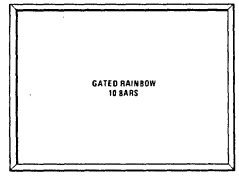
Purity
0101



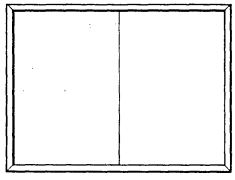
Dots 7 x 11
0110



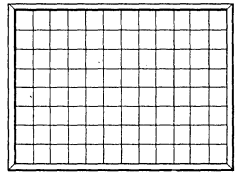
Single Dot
0111



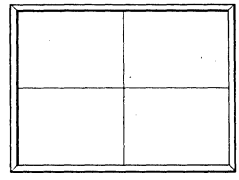

Gated Rainbow
1000



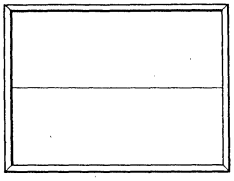
Single Vertical Line
1001



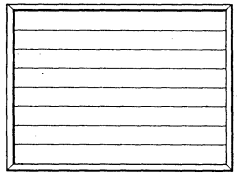
7 x 11 Cross Hatch
1010



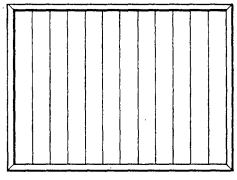
Single Crosshair
1011



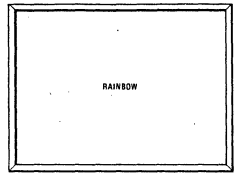
Single Horizontal Line
1100



7 Horizontal Lines
1101



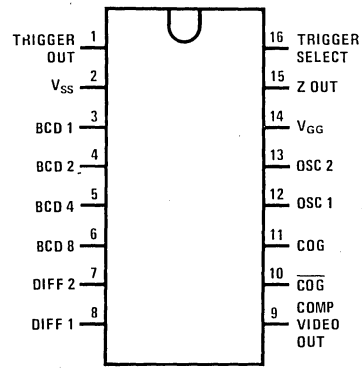
11 Vertical Lines
1110




Ungated Rainbow
1111

## connection diagram

Dual-In-Line Package




TOP VIEW

Note. $Z_{OUT}$ is an internal counter test point.

**Order Number MM5322N**
**See Package 19**

National Semiconductor

Television/Radio

# MM53100, MM53105 Programmable TV Timers

## general description

The MM53100 and MM53105 programmable TV timers are monolithic CMOS integrated circuits utilizing P and N-channel low threshold enhancement devices. These circuits contain all the logic to give a 4 or 6-digit, 24-hour display from a 50 or 60 Hz input, and control the "ON" time of the TV. The duration of the viewing period is 5, 10, 20 or 30 mins, selected by 2 input pins. Manual "ON" and "OFF" inputs are also provided. The MM53100 and MM53105 have ultra-low power dissipation in the stand-by mode and are ideally suited to crystal controlled battery-operated systems. The MM53100 is designed for an optimum interface in TVs with a positive common reference voltage (e.g., +18V). The MM53105 is designed for an optimum interface for TVs with a 0V reference voltage. Both are packaged in a 24-lead dual-in-line epoxy package.

## features

- 50 or 60 Hz operation
- 24-hour display format
- Programmable TV on time
- Selectable view time
- Ultra-low power dissipation
- All counters resettable
- Low voltage operation
- Elimination of illegal time display at turn-on
- Daily repeat or non-repeating operating
- Fool-proof safety features
- Compatible with MM5840 display circuit

## applications

- TV time display
- Remote TV "ON"/"OFF" switch
- Computer clock
- Time data–logging systems

## block diagram

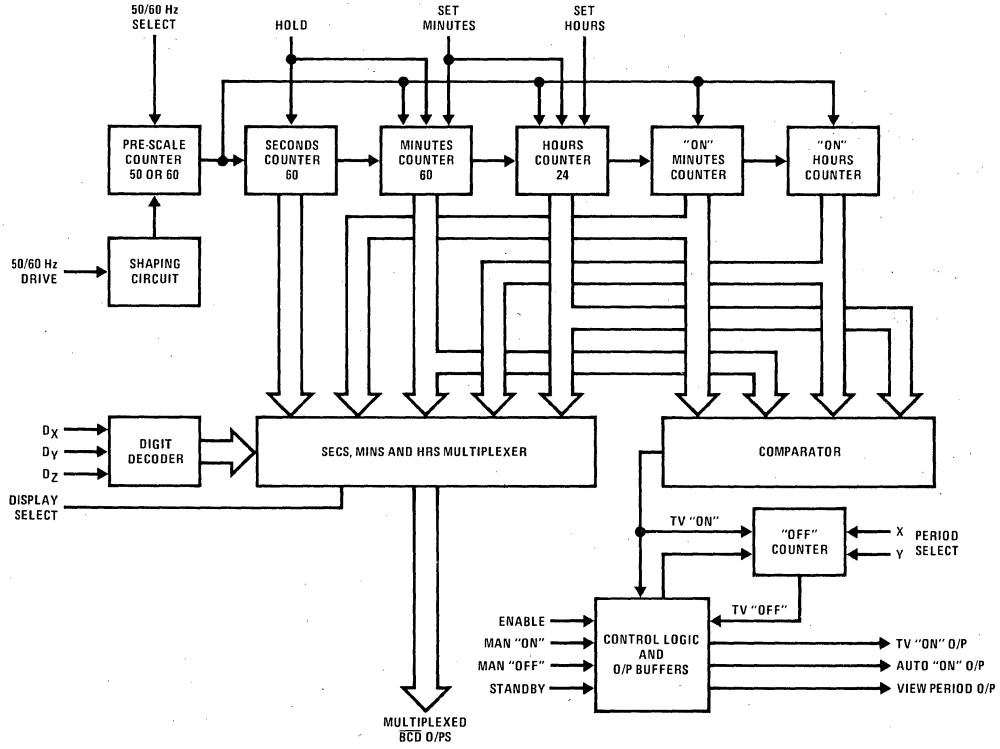



FIGURE 1. MM53100, MM53105 Block Diagram

## absolute maximum ratings (MM53100) ($V_{DD}$ common voltage reference)

| | |
|---|---|
| Supply Voltage ($V_{DD} - V_{SS}$) | 6V |
| Voltage at 50/60 Hz Select and Period Select Inputs | $V_{SS} - 0.3V$ to $V_{DD} + 0.3V$ |
| Current Into or Out of Any Other Input | 100 $\mu$A max |

## electrical characteristics (MM53100) $T_A = 25°C$, $V_{DD} = 4.5V$, $V_{SS} = 0V$ unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Supply Voltage | | 2.8 | | 5.0 | V |
| Supply Current | $V_{DD} = 4.5V$ | | 10 | 25 | $\mu$A |
| Input Logic Levels | | | | | |
| 50/60 Hz Input, Digit Select Inputs, Display Select, "ON", "OFF", Time Setting Control, Standby Control | | | | | |
| Logic "1" | | $V_{DD}-0.5$ | | $V_{DD}$ | V |
| Logic "0" | (Note 1) | | | $V_{SS}+0.5$ | V |
| 50/60 Hz Select, Period Select (X, Y) | | | | | |
| Logic "1" | | $V_{DD}-0.5$ | | $V_{DD}$ | V |
| Logic "0" | | $V_{SS}$ | | $V_{SS}+0.5$ | V |
| Display Select Input Delay | | 0.5 | | 2.0 | $\mu$s |
| Output Logic Levels | | | | | |
| BCD Outputs | External Resistor, 15 k$\Omega$ to $V_{DD} - 12V$, $C_L = 15$ pF | | | | |
| Logic "1" | | $V_{DD}-0.8$ | | | V |
| Logic "0" | | | | $V_{DD}-11.2$ | V |

**Note 1:** If input voltages go more negative than $V_{SS}$, the input current must be limited to a maximum of 100 $\mu$A by the use of external series resistors. No resistors are required on the $D_X$, $D_Y$, $D_Z$ inputs when interfacing with the MM5840.

## absolute maximum ratings (MM53105) ($V_{SS}$ common voltage reference)

| | |
|---|---|
| Supply Voltage ($V_{DD} - V_{SS}$) | 6V |
| Voltage at 50/60 Hz Select and Period Select Inputs | $V_{SS} + 6V$ |
| Voltage at Any Other Pin | $V_{SS} + 13V$ |

## electrical characteristics (MM53105) $T_A = 25°C$, $V_{DD} = 4.5V$, $V_{SS} = 0V$ unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Supply Voltage | | 2.8 | | 5.0 | V |
| Supply Current | $V_{DD} = 4.5V$ | | 10 | 25 | $\mu$A |
| Input Logic Levels | | | | | |
| 50/60 Hz Input, Digit Select Inputs, "ON", "OFF", Display Select, Time Setting Controls, Standby Control | | | | | |
| Logic "1" | | $V_{DD}-0.5$ | | 13 | V |
| Logic "0" | | $V_{SS}$ | | $V_{SS}+0.5$ | V |
| 50/60 Hz Select, Period Select (X, Y) | | | | | |
| Logic "1" | | $V_{DD}-0.5$ | | $V_{DD}$ | V |
| Logic "0" | | $V_{SS}$ | | $V_{SS}+0.5$ | V |
| Display Select Input Delay | | 0.5 | | 2.0 | $\mu$s |

## electrical characteristics (Continued) (MM53105) $T_A = 25°C$, $V_{DD} = 4.5V$, $V_{SS} = 0V$ unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Output Logic Levels | | | | | |
| BCD Outputs | External Resistor 15 kΩ to 12V, $C_L$ = 15 pF | | | | |
| Logic "1" | | 11.2 | | | V |
| Logic "0" | | | | 0.8 | V |
| TV "ON" Output, Auto "ON" Output, View Period Output | | | | | |
| Logic "1" | Loaded 2.7 kΩ to $V_{SS}$ | 0.5 | | | mA |
| Logic "0" | Loaded 2.7 kΩ to $V_{DD}$ | 1.0 | | | mA |

**Note 1:** Input voltages to go more positive than $V_{DD}$.

## functional description

A block diagram of the MM53100, MM53105 TV timers is shown in *Figure 1*. A connection diagram is shown in *Figure 2*. Unless otherwise indicated, the following discussions are based on *Figure 1. Figures 5a and 5b* illustrate the system configuration for a crystal controlled TV display system using both circuits.

**Dual-In-Line Package, Top View**

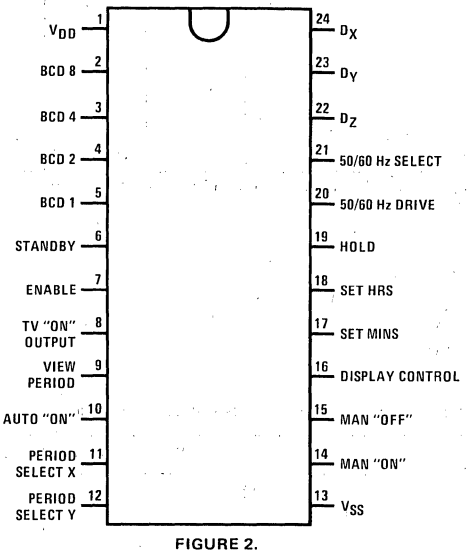



**FIGURE 2.**

**Order Number MM53100N or MM53105N**
**See Package 22**

**50 or 60 Hz Drive:** This input is applied to a Schmitt trigger shaping circuit which allows use of a filtered sinewave input. A simple RC filter should be used to remove possible line voltage transients that could either cause the clock to gain time or damage the device. The input should swing between $V_{SS}$ and $V_{DD}$. The shaper output drives a counter chain which performs the time-keeping function.

Alternatively, in a crystal controlled battery operated system, an oscillator and prescaler such as the MM53107 could be used as a time base.

**50 or 60 Hz Select Input:** This input programs the prescale counter to divide by either 50 or 60 to obtain a 1 pps time base. The counter is programmed for 60 Hz operation by connecting this input to $V_{DD}$. An internal 1 MΩ pull-down resistor is common to this pin; simply leaving this input unconnected programs the clock for 50 Hz operation.

**Time Setting Inputs:** Inputs to set hours and set minutes as well as hold input, are provided. Internal 1 MΩ pull-down resistors provide the normal timekeeping function. Switching any 1 of these inputs (1 at a time) to "1" results in the desired time setting function. Set Hours advances hours information at 1 hour/second and Set Minutes advances minutes information at 1 minute/second, without roll over into the hours counter. Set Minutes also resets the seconds counter to 0. The hold input stops the clock to the minutes counter and resets the seconds counter. Activating Set Minutes and Set Hours simultaneously resets the displayed counters to all 0's.

**Display:** This input controls the display and time-setting operation. It has an internal 1 MΩ pull-down resistor to $V_{SS}$. When taken to Logic "0" or in open circuit condition, the real time is displayed and the Set Hours and Set Minutes inputs operate the real time counters. When taken to logic "1", the "ON" time is displayed and the time-setting inputs operate on the "ON" counters.

**Digital Select Inputs ($D_X$, $D_Y$, $D_Z$):** These 3 inputs are used to determine which digit will be displayed. Table IA shows the code for each digit. Seconds will be displayed as "00" when the "ON" time is being displayed.

**Enable:** This input has an internal resistor to $V_{SS}$. When taken to logic "1", this input disables the programmed "ON" time for the TV output.

**Period Select Inputs (X, Y):** These inputs have pull-down resistors to $V_{SS}$. They determine the view period, i.e., 5, 10, 20 or 30 mins. Table IB shows the Period Select Code.

## functional description (Continued)

**Standby Control Input:** This input has an internal resistor to $V_{SS}$. Its function is to sense when the line generated 12V supply is turned off and to then disable the outputs. In the TV, this input should be connected to the 12V supply.

**Manual "ON" Input:** This input has an internal resistor to $V_{SS}$. When taken to logic "1", this input turns the TV output to the "0" state. It is designed to have typically 0.75 second debounce time to prevent mal-operation.

**Manual "OFF" Input:** This input has an internal resistor to $V_{SS}$. When taken to logic "1", this input turns the TV output to the "1" state. It is designed to have typically 0.75 second debounce time to prevent mal-operation.

**TV "ON" Output:** *Figure 3* illustrates the CMOS inverter output circuit used.

In the manual mode of operation, the manual "ON" input sets this output to "0", the manual "OFF" input resets this output to "1". The manual "ON" input inhibits the auto "ON" output.

In the programmable mode, this output goes to "0" when the programmed "ON" time coincides with the real time (unless enable = 1). The output will then stay at "0" for the selected period of 5, 10, 20 or 30 minutes before returning to "1" state. During this period, a signal on the manual "ON" input will prevent the automatic switch-off.

Manual "OFF" input will always reset the output to a logic "1" state.

**Auto "ON" TV Output:** An additional output is provided to indicate that the TV is "ON" in the automatic mode of operation. This output goes to a logic "0" for the duration of the auto "ON" time. Manual "ON" switches this output back to a logic "1".

**View Period Indicator:** This output normally is a logic "1". When the TV switches on at the programmed time, this output transmits a 1 Hz waveform for the duration of the selected view period. Hence, it can be used to indicate that the TV is switched on for a limited period only by means of a flashing on-screen and/or off-screen display. The output will permanently return to "1" at the end of the viewing period or when a valid manual "ON" or "OFF" input signal is received during the view period.

**BCD Outputs:** *Figure 4* illustrates the open drain output circuits used, a) MM53100, b) MM53105.

With the use of the external respective pull-up and pull-down resistors, these outputs are designed to be compatible with the MM5840 and MM5841 TV display circuits.

Note. Case (a) for common $V_{DD}$, case (b) for common $V_{SS}$ when used with the MM5840.

**TABLE IA. Digit Select Code**

| DIGIT SELECT LINES | DIGIT DISPLAYED | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | S1 | S10 | * | M1 | M10 | * | H1 | H10 |
| $D_X$ | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| $D_Y$ | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| $D_Z$ | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

**TABLE IB. Period Select Code**

| PERIOD SELECT INPUTS | | VIEW PERIOD PROGRAMMED |
|---|---|---|
| X | Y | |
| 0 | 0 | 5 mins |
| 0 | 1 | 10 mins |
| 1 | 0 | 20 mins |
| 1 | 1 | 30 mins |

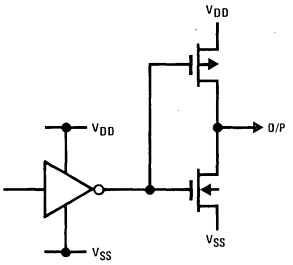



FIGURE 3. CMOS Output (TV "ON", Auto "ON", Indicator)

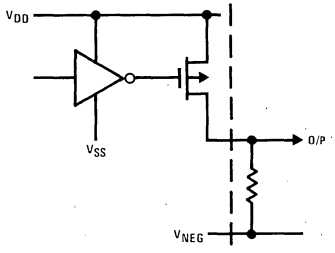



FIGURE 4a. BCD Outputs, MM53100

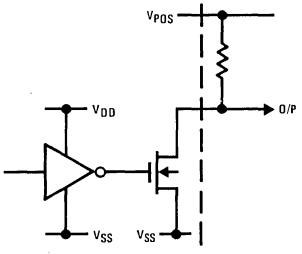



FIGURE 4b. BCD Outputs, MM53105

## functional description (Continued)

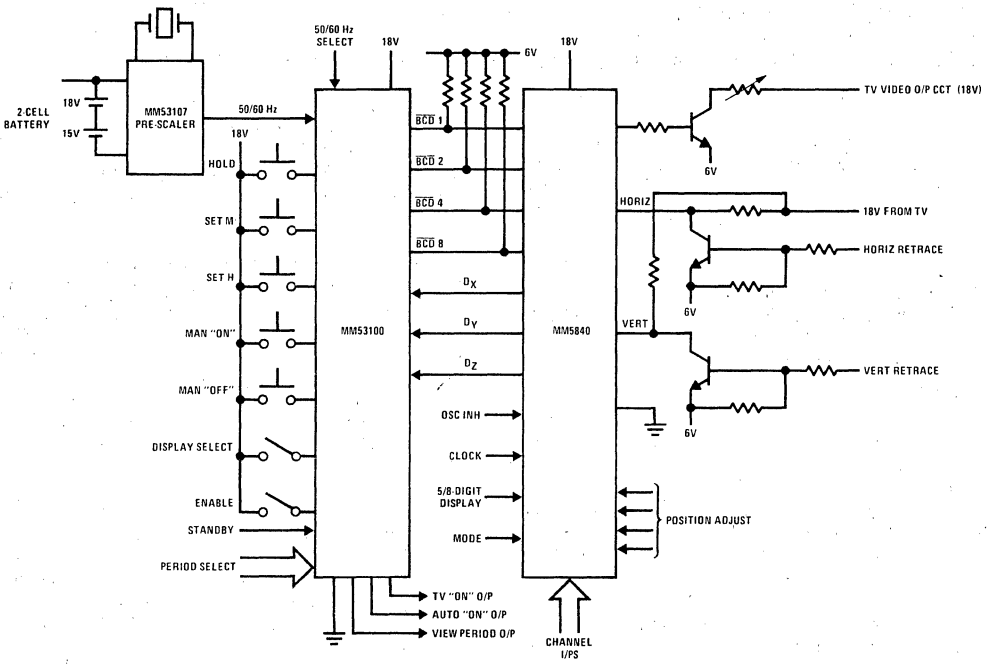



**FIGURE 5a. Typical System Diagram, MM53100**

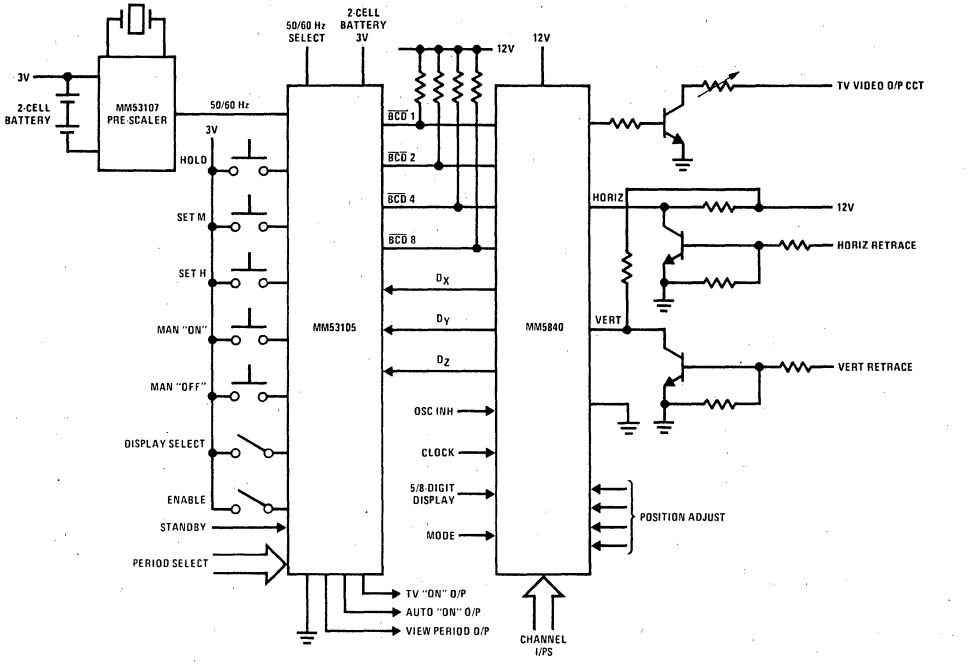



**FIGURE 5b. Typical System Diagram, MM53105**

National Semiconductor

Television/Radio

# MM53118AA TV Digital Tuning

## General Description

The MM53118AA is a monolithic MOS integrated circuit utilizing N-Channel silicon gate low threshold, enhancement mode and ion-implanted depletion mode devices. This IC contains all input control, output control, and PLL circuits required for a closed-loop digital tuning system.

The MM53118AA contains complete input decoding logic, up/down counters, a channel decoding ROM with 117 channel capacity, a complete PLL circuit including the input oscillator, reference counter, phase comparator, error detector, and output mode control circuits, and a set of output buffers that drive external BCD display drivers.

## Features

- Easy interface to television tuner
- Wide choice of channel indicator displays
- Choice of simple up/down or full keyboard channel entry
- Interface to MM58146 digital clock/channel display circuit
- Optional non-volatile memory for active channel storage
- Compatible with electronic remote control circuits
- Standard broadcast and cable TV frequencies
- PLL/AFT operation
- Manual fine tune capability
- Muting capability
- Leading zero blanking

## Functional Description

*Figure 1* is a block diagram of the MM53118AA. *Figure 2* is a system block diagram showing the major elements required for a closed loop tuning system with up/down channel selection. *Figure 3* is an expansion of the basic system to include direct access channel selection and interface to a full 16 command remote control system.

## Block Diagram

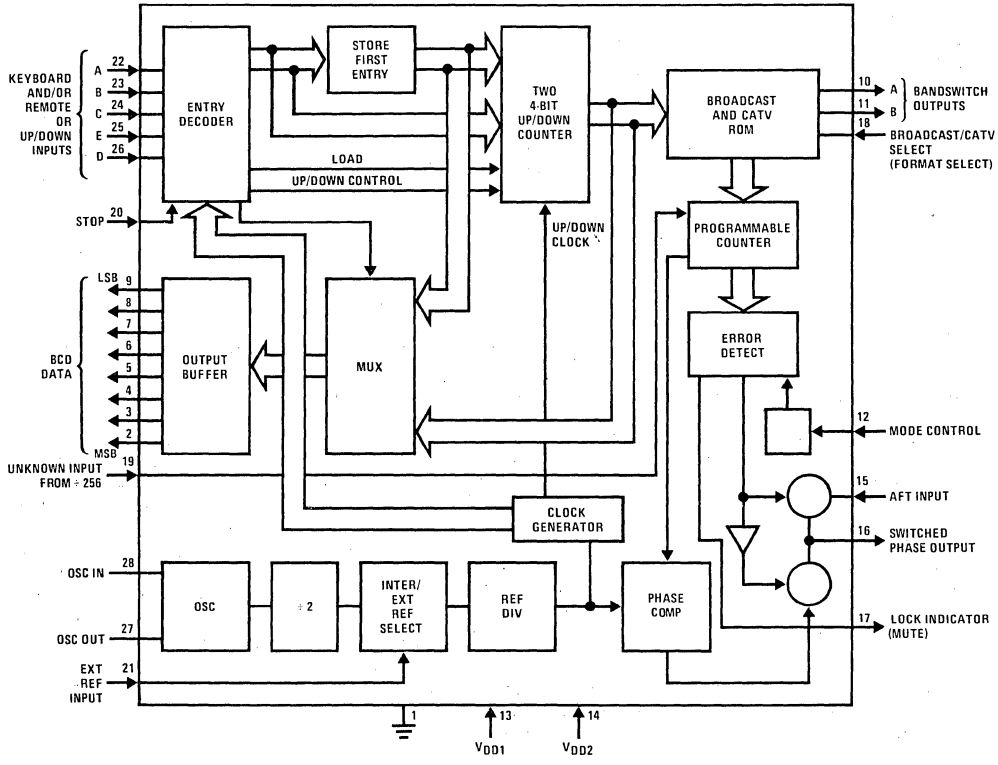



FIGURE 1

## Absolute Maximum Ratings

| | |
|---|---|
| Maximum Voltage at Any Pin | $V_{SS}$ − 0.3V to $V_{SS}$ + 10V |
| Operating Temperature | 0°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics ($T_A$ within operating range) functional voltages $V_{SS}$ = 0, $V_{DD1}$ = 5, $V_{DD2}$ = 8

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supplies | | | | | |
| $V_{DD1}$ | | 4.75 | 5 | 5.25 | V |
| $V_{DD2}$ | | 7.6 | 8 | 8.4 | V |
| $V_{DD2}$ | Memory Only | 7.6 | | 8.4 | V |
| Power Supply Current | $V_{DD1}$ = 5.25 | | | 20 | mA |
| | $V_{DD2}$ = 8.4 | | | 2 | mA |
| Unknown Signal Input | AC Coupled | 0.6 | | 2.6 | Vp-p |
| Reference Signal Input | AC Coupled | 0.6 | | 2.6 | Vp-p |
| Input Voltage Levels | 4.75 < $V_{DD1}$ < 5.25 | | | | |
| Keyboard and Stop | | | | | |
| High Level | | $V_{DD1}$ − 1 | | $V_{DD1}$ | V |
| Low Level | | $V_{SS}$ − 0.3 | | $V_{SS}$ + 0.9 | V |
| Mode Control and Broadcast/CATV | | | | | |
| High Level | | $V_{DD1}$ − 1 | | $V_{DD1}$ | V |
| Low Level | | $V_{SS}$ − 0.3 | | $V_{SS}$ + 0.3 | V |
| Input Current Levels | Internal Pull-ups | | | | |
| Keyboard (Except Pin 26) and Stop | | −15 | | −35 | μA |
| Mode Control, Pin 26, and Broadcast/CATV | | | | | |
| Source | | | | −60 | μA |
| Sink | | | | 60 | μA |
| BCD Data Outputs or Bandswitch Output | 4.75 < $V_{DD}$ < 5.25 | | | | |
| High Level Output Voltage | $I_{OUT}$ = 100 μA | 2.6 | | | V |
| Low Level Output Voltage | $I_{OUT}$ = −1.6 mA | | | 0.4 | V |
| Phase Detector Output | | | | | |
| High Level Output Voltage | $I_{OUT}$ = 0.2 mA | 4 | | | V |
| Low Level Output Voltage | $I_{OUT}$ = −0.2 mA | | | 0.5 | V |
| AFT IN | | −0.3 | | 4.5 | V |
| Crystal Frequency | 4 ±0.005% | | | | MHz |
| External Reference Input | | 1.5 | 2.0 | 2.5 | MHz |

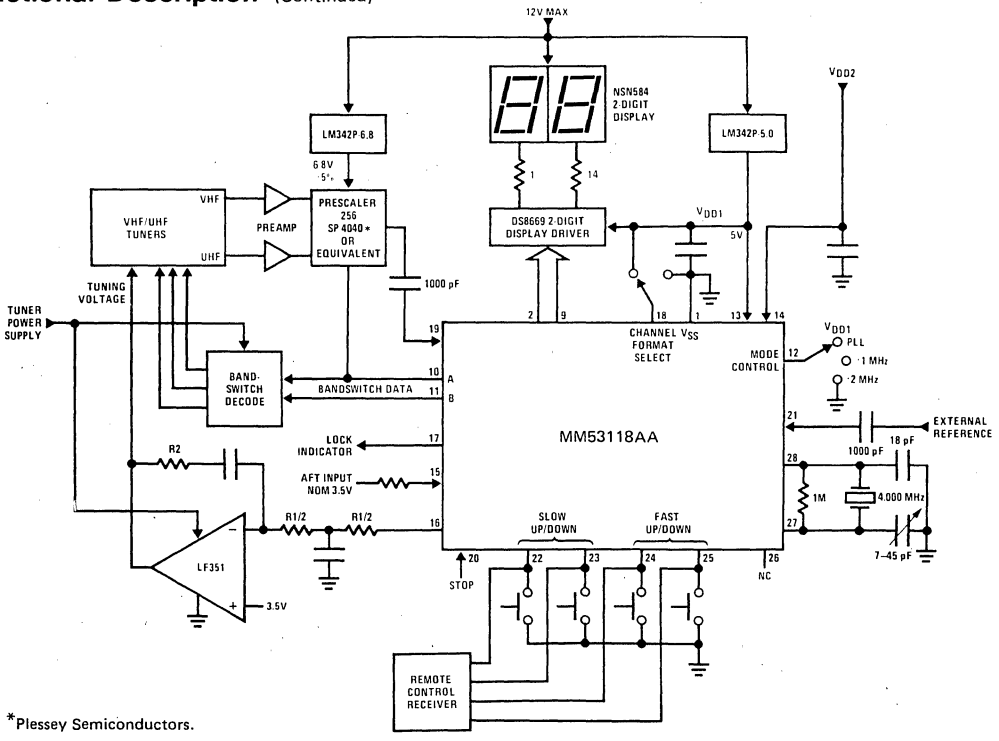
## Functional Description (Continued)

The MM53118AA was designed to be used in several configurations depending on the features required. The basic tuning system is shown in *Figure 2*.

The timing is derived from a 4.000 MHz crystal oscillator which is divided down to provide the keyboard debounce timing, the clock for 4 different up/down speeds and the reference signal to the phase comparator which is 3.90625 kHz.

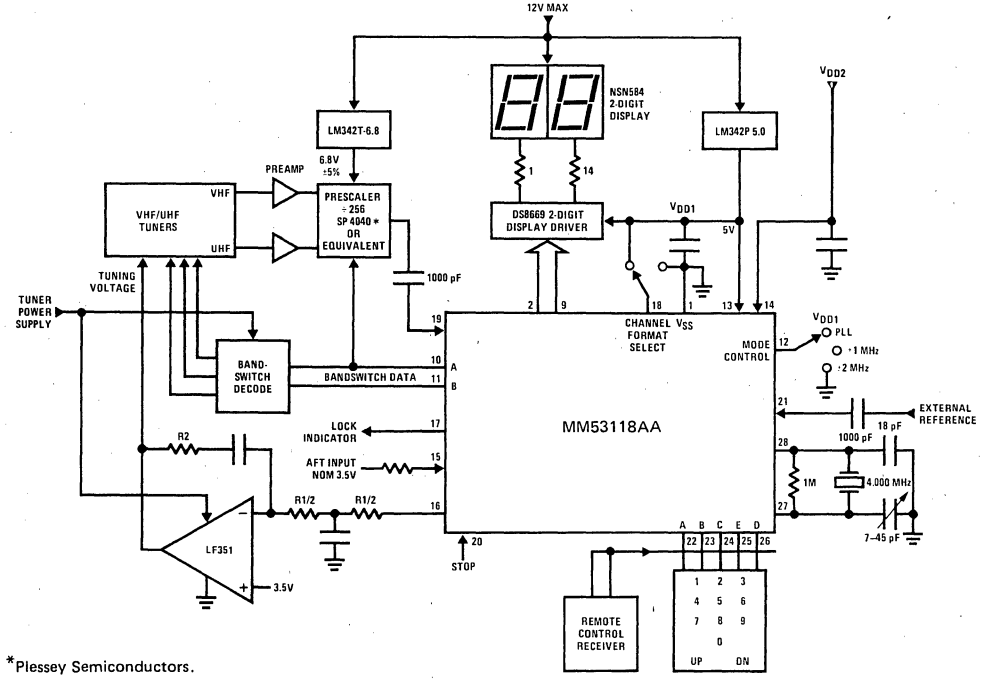
Channel entry by either up/down or direct access is done via the 5 input lines A, B, C, E and D (pins 22 thru 26). Input D acts as a control. If it is at the logic "1" state ($V_{DD}$ − 1), the chip will ignore any other inputs. When at logic "0" state, there are 16 possible commands via the other 4 input lines. The functions performed are summarized in Table I. If the D input is left floating, then inputs A and B become up/down controls at 1.5 channels per second and C and D are up/down controls at 10 channels per second. This is the simplest form of channel selection. The fast speed is used to get close and the slow one to select the desired channel. Also the slow speed allows scanning thru the channels to see what programs are available. The up/down method of channel selection makes possible the use of a simple remote control, since it requires only 4 commands for channel selection.

## Functional Description (Continued)




*Plessey Semiconductors.

FIGURE 2. Basic Tuning System With Simple Remote Control Option




*Plessey Semiconductors.

FIGURE 3. Tuning System With Full Keyboard And Remote Control

## Functional Description (Continued)

TABLE I. KEYBOARD INTERFACE TO MM53118AA

| FUNCTION | MM53118AA INPUTS | | | | |
|---|---|---|---|---|---|
| | A | B | C | E | D |
| None | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 0 | 0 |
| 2 | 1 | 1 | 0 | 1 | 0 |
| 3 | 1 | 1 | 0 | 0 | 0 |
| 4 | 1 | 0 | 1 | 1 | 0 |
| 5 | 1 | 0 | 1 | 0 | 0 |
| 6 | 1 | 0 | 0 | 1 | 0 |
| 7 | 1 | 0 | 0 | 0 | 0 |
| 8 | 0 | 1 | 1 | 1 | 0 |
| 9 | 0 | 1 | 1 | 0 | 0 |
| 10 Memory Up | 0 | 1 | 0 | 1 | 0 |
| 11 Memory Down | 0 | 1 | 0 | 0 | 0 |
| 12 Slow Up | 0 | 0 | 1 | 1 | 0 |
| 13 Slow Down | 0 | 0 | 1 | 0 | 0 |
| 14 Search Up | 0 | 0 | 0 | 1 | 0 |
| 15 Search Down | 0 | 0 | 0 | 0 | 0 |

Note: The ITT SAA1025 can be used for remote input to the MM53118AA

When direct access is used for channel selection, a diode matrix is required to provide the necessary input code. *Figure 4* shows the connection for entering numbers 0 thru 9 plus slow up and slow down. Inputs A, B, C and E have internal pull-up current sources, so only one resistor is required at input D. *Figure 6* shows the use of the MM74C922 keyboard encoder as an alternative to the diode matrix. It shows the connection for all 16 functions. In practice only the necessary number will be used.

To select a channel via the keyboard or direct access remote control, 2 entries are necessary. The first enters the MSB (tens) and the second enters the LSB (units). For channels 2 thru 9, the 0 has to be entered first. For all channels the sequence is as follows: When the first entry is made, the number will appear on the MSB and a dash (–) will appear on the LSB. The second entry will complete the sequence and then the new channel number will be synthesized. If the second entry is not completed the display will convert to the original number approximately 4.5 seconds after the first entry.

There are 2 debounce times available. One is approximately 92 ms and the second is approximately 184 ms. As shown in Table I, there are 16 possible direct access functions. Ten are for direct access channel select, two for memory up/down at 250 channels per second, two for slow up/down at 1.5 channels per second and 2 for search up/down at 5 channels per second. The memory and search operation are explained later on.

The outputs of the 2 BCD counters are brought out to drive display drivers directly, which allows for wide choice of display types, plus eliminating the possible radiation caused by multiplexing. In the 2 system diagrams (*Figure 2* and *Figure 3*) the DS8669 is shown as a 2-digit LED driver. Outputs A and B (pins 10 and 11) provide bandswitch information as shown in Table II. This information has to be decoded and level shifted as required for particular tuners.

The lock indicator output can be used to mute the sound as a new channel is selected. In direct access selection the output goes to logic "1" after the second entry and stays at logic "1" for approximately 92 ms. This is sufficient time for the loop to acquire lock since the reference to the phase comparator is approximately 4 kHz. In any of the up/down speeds the lock indicator output stays at logic "1" for as long as any of the keys is depressed and goes to logic "0" approximately 92 ms after the key is released. The operation described is for the PLL mode. If the MM53118AA is used in the AFT mode the operation is slightly different and is described later.

As shown in *Figure 2* the output of the divide-by-256 prescaler is AC coupled to the MM53118AA. This is because the input is internally biased to accept sine wave input of ECL p-p levels. This will minimize radiation caused by fast rise and fall times at the output of the divide-by-256 at TTL levels. If a prescaler with TTL outputs is used it can either drive the MM53118AA directly or a simple RC filter can be used to slow rise and fall times. The same input levels can be used for the external reference input.

A digital phase comparator is used which provides a pulse output proportional to the time difference between the edges of the 2 inputs of the phase comparator. Under ideal conditions, when the loop is in phase lock, the output of the phase comparator will be open circuit and the tuning voltage will be stored at the feedback capacitor. However because of leakage and input bias current for the operational amplifier there will always be small pulses either to ground or to $V_{DD}$. Those pulses after integration will appear as noise on the tuning voltage and change the frequency of the oscillator. The amount of noise that can be tolerated is a function of the gain constant of the oscillator and also of the effect it has on the picture quality. The operational amplifier shown has a FET input whose input bias is in the pA range. If this type of input is necessary, it will be determined by the overall system requirements.

An alternative solution is shown in *Figure 5*, where the FET input op-amp has been replaced by a Darlington amplifier.

## Functional Description (Continued)

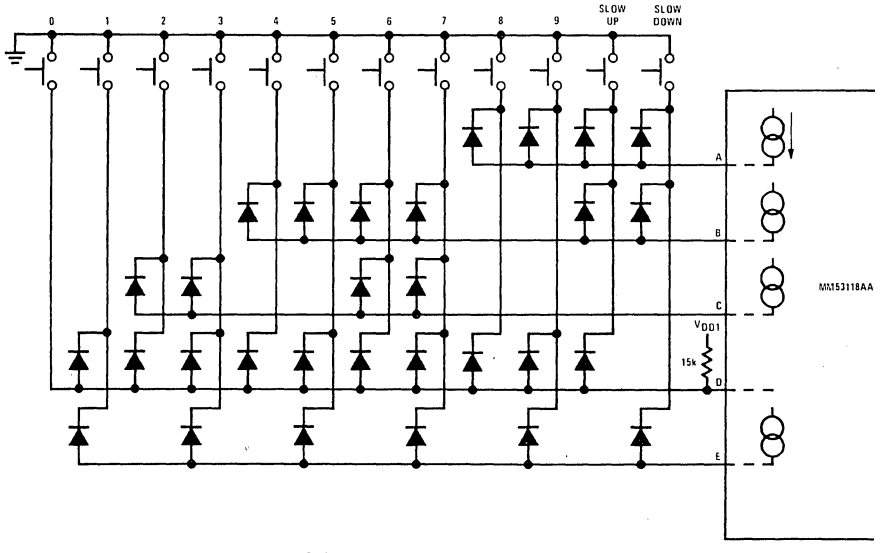
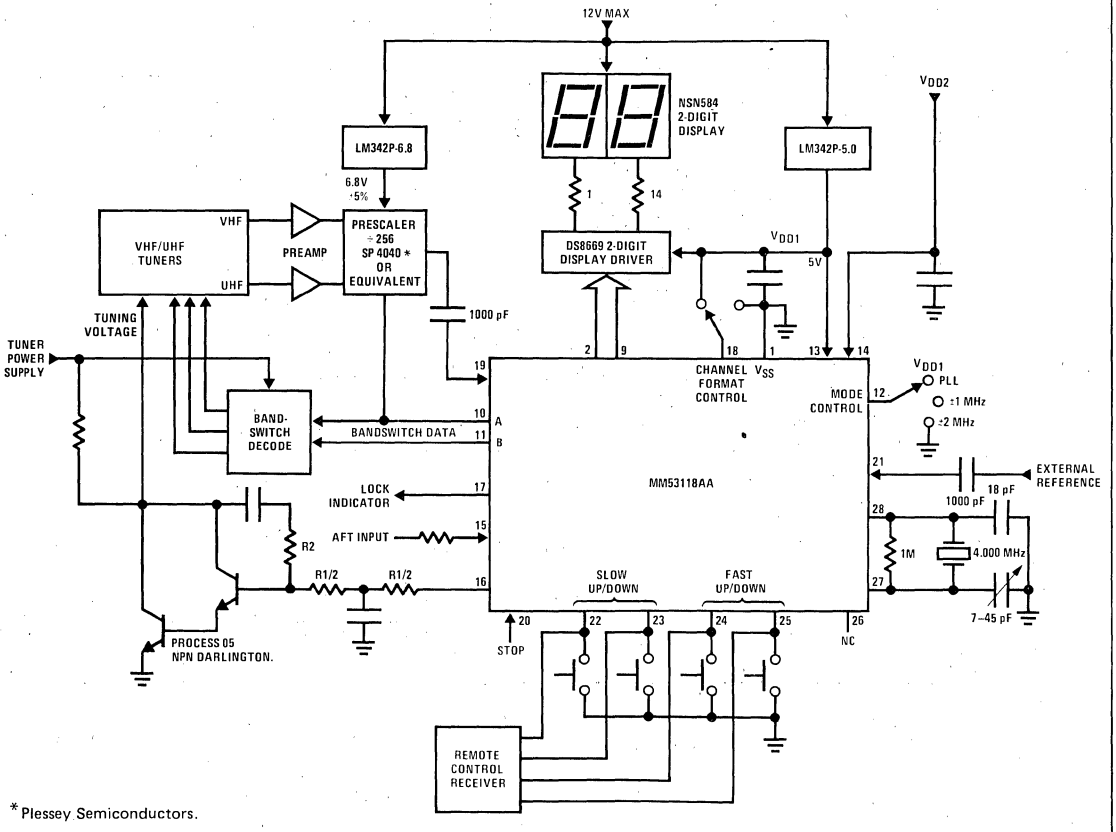



FIGURE 4. Keyboard Interface

TABLE II. BANDSWITCH OUTPUT CODES

| CABLE TV MODE | | | | BROADCAST MODE | | | | |
|---|---|---|---|---|---|---|---|---|
| CH. NO. | L.O. FREQUENCY | A | B | CH. NO. | L.O. FREQUENCY | A | B | BAND |
| 2 | 101 | 1 | 1 | 2 | 101 | 1 | 1 | I |
| 3 | 107 | 1 | 1 | 3 | 107 | 1 | 1 | I |
| 4 | 113 | 1 | 1 | 4 | 113 | 1 | 1 | I |
| 5 | 123 | 1 | 1 | 5 | 123 | 1 | 1 | I |
| 6 | 129 | 1 | 1 | 6 | 129 | 1 | 1 | I |
| 7 | 221 | 0 | 1 | 7 | 221 | 0 | 1 | II |
| 8 | 227 | 0 | 1 | 8 | 227 | 0 | 1 | II |
| 9 | 233 | 0 | 1 | 9 | 233 | 0 | 1 | II |
| 10 | 239 | 0 | 1 | 10 | 239 | 0 | 1 | II |
| 11 | 245 | 0 | 1 | 11 | 245 | 0 | 1 | II |
| 12 | 251 | 0 | 1 | 12 | 251 | 0 | 1 | II |
| 13 | 257 | 0 | 1 | 13 | 257 | 0 | 1 | II |
| 14 | 167 | 0 | 1 | 14 | 517 | 0 | 0 | III |
| 15 | 173 | 0 | 1 | • | | | | |
| 16 | 173 | 0 | 1 | • | | | | |
| 17 | 185 | 0 | 1 | 83 | 931 | 0 | 0 | III |
| 18 | 191 | 0 | 1 | | | | | |
| 19 | 197 | 0 | 1 | | | | | |
| 20 | 203 | 0 | 1 | | | | | |
| 21 | 209 | 0 | 1 | | | | | |
| 22 | 215 | 0 | 1 | | | | | |
| 23 | 263 | 1 | 0 | | | | | |
| 24 | 269 | 1 | 0 | | | | | |
| 25 | 275 | 1 | 0 | | | | | |
| 26 | 281 | 1 | 0 | | | | | |
| 27 | 287 | 1 | 0 | | | | | |
| 28 | 293 | 1 | 0 | | | | | |
| 29 | 299 | 1 | 0 | | | | | |
| 30 | 305 | 1 | 0 | | | | | |
| 31 | 311 | 1 | 0 | | | | | |
| 32 | 317 | 1 | 0 | | | | | |
| 33 | 323 | 1 | 0 | | | | | |
| 34 | 329 | 1 | 0 | | | | | |
| 35 | 335 | 1 | 0 | | | | | |
| 36 | 341 | 1 | 0 | | | | | |

## Functional Description (Continued)




*Plessey Semiconductors.

FIGURE 5. Basic Tuning System With Simple Remote Control Option

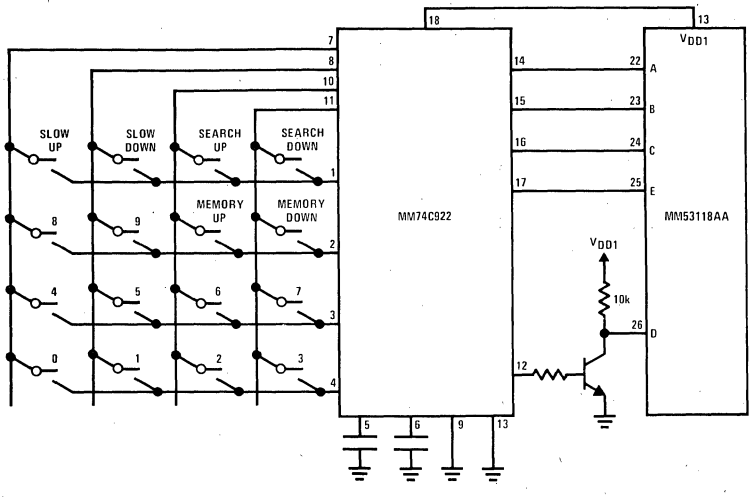



FIGURE 6. Keyboard Encoder (MM74C922) Interface

## Functional Description (Continued)

### CABLE TV OPERATION

Cable TV channels are transmitted in 3 different bands. Band I is the normal VHF channel and contains channels 2–13. Band II accommodates 9 channels in the frequency range between VHF channels 6 and 7. Band III accommodates 14 channels in the frequency range above VHF channel 13. The specific frequencies for each channel have not been standardized by the FCC, but are adapted through common practice by cable TV broadcasters. Furthermore, there has been no standardization of channel number display for identification.

The MM53118AA has 1 input which identifies operation in either the cable TV or standard broadcast mode. If the input is at logical "1" the chip synthesizes broadcast channels 2–83 and with the input at logical "0", it synthesizes cable TV channels 2–36 and displays them as such. (If CATV channel 14 is selected, it displays 14, and the synthesized frequency is the one corresponding to cable channel 14 or 167 MHz.)

If up/down clocking is used instead of direct access, the rollover is at 36 and 2. However, if channels 37–83 are entered by direct access, the MM53118AA synthesizes the normal broadcast channels and the rollover is at 83 until the cycle begins again at 2 or 36. Channels 84-99 are illegal. If an attempt is made to enter one of these channels, the MM53118AA will return to the previously selected channel.

The MM53118AA has the capability of synthesizing 82 standard broadcast and 35 CATV channels for a total of 117. Table II shows the channel number, local oscillator frequency and bandswitch data for all channels. There are separate ROM locations for all CATV and standard broadcast channels so that if a different local oscillator frequency is desired it can be accomplished by a simple ROM change. This might be necessary if CATV frequencies change.

### FINE TUNING

The reference frequency to the phase comparator is derived by dividing down the output of the crystal controlled 4 MHz oscillator. However, if an external 2 MHz signal is applied to the external reference input (pin 21), this signal automatically disconnects the output of the 4 MHz crystal oscillator to the divider chain. By varying the external frequency slightly, the frequency of the tuner oscillator varies proportionally. For example, if the frequency varies by 2 kHz, the oscillator at channel 2 will vary by 100 kHz and 254 kHz at channel 13.

An LC oscillator can be used for the external frequency which is activated only if fine tuning is required. If the fine tuning is not used, the input should be tied to $V_{DD1}$.

### LAST CHANNEL MEMORY

There are 2 power supply voltages required for the operation of the MM53118AA. $V_{DD1}$ is nominally at 5.0 volts, and it is the main power supply. $V_{DD2}$ is nominally at 8.0 volts and it is required for several reasons. It provides the drive for the BCD, bandswitch, and lock indicator output buffers so that they are TTL compatible. It also provides the drive for the phase comparator output buffer when the input to the operational amplifier is offset by 3.5 volts. By allowing $V_{DD2}$ to remain ON while $V_{DD1}$ is OFF, it is possible to have last channel memory. In other words, the set will power on the last channel entered when $V_{DD1}$ is turned ON again.

### AFT OPERATION

There are instances where the frequency of the local oscillator will be different than the ones shown in Table II. This might be due to either master antenna distribution systems, CATV or any other. To accommodate those frequency offsets the MM53118AA can operate in frequency lock instead of phase lock, and requires that the output of the AFT discriminator be connected to pin 15 at the proper bias level.

When the mode control (pin 12) is tied to $V_{DD1}$, the MM53118AA operates as a phase lock loop even if the AFT is connected. If the mode control is floating the sequence of operation is as follows: when a new channel is selected, either via direct access or up/down the correct channel frequency is at first synthesized. After approximately 92 ms the output of the phase detector is switched OFF and the AFT is applied to the input of the operational amplifier. This is accomplished via the 2 gates that are controlled from the error detector shown in *Figure 1*. The loop is now in frequency lock and the AFT is controlling it with the operational amplifier providing all the gain instead of the AFT discriminator. Also the AFT voltage under correct tuning, should be offset from $V_{SS}$. *Figure 2* and *Figure 3* show an offset of 3.5V. Under this condition the output of the AFT discriminator could swing by ±1V or 2V p-p. The series resistor shown at the AFT input is probably necessary to reduce the loop gain, which, if too high, could cause the loop to oscillate. Another reason for the 3.5V is that the LF351 needs that minimum offset from the $V_{SS}$ rail to operate properly. By using a different type of operational amplifier this offset voltage could vary.

Under the frequency lock operation, the tuner frequency is continuously monitored by checking the state of the programmable counters. With the mode control floating, as long as this frequency is within ±1M from the nominal the loop is in frequency lock and under AFT control. If it exceeds the ±1 MHz range the loop will revert to PLL, synthesize the correct frequency and then switch back to AFT. This switching between PLL and AFT will be continuous as long as the offset is greater than ±1 MHz. The closer to ±1 MHz the slower the switching.

It is possible to extend the range to ±2 MHz by connecting the mode control to $V_{SS}$. However this should be done with great care, because the range of the normal AFT is restricted to something less than 1 MHz in one direction and actually, because of the presence of sound carrier, will reverse polarity and tune the oscillator

## Functional Description (Continued)

toward the wrong direction. The ±1 MHz range was chosen because this is the normal operating range of a conventional AFT. The ±2 MHz was chosen because offsets of that magnitude are possible in certain master antenna distribution systems.

In the frequency lock mode the lock indicator output is at logic "0" when the oscillator is within the error range and at logic "1" when outside that range.

The AFT operation is presently restricted to VHF and CATV only. When a UHF channel is tuned the MM53118AA is automatically in the PLL mode even if the AFT is connected.

### FAVORITE CHANNEL OPERATION

*Figure 7* is a block diagram of the components required for favorite channel operation. The memory can be any non-volatile type including CMOS with battery back-up. The outputs of the 2 BCD's counters that drive the display driver also address the RAM. A logic "1" is stored at the channels that are watched and logic "0" at all others. This is a one time operation. When the memory up or memory down code shown in Table I is entered at the 5 inputs, the MM53118AA will advance to the next higher or lower channel. If the output of the RAM is at logic "1", the counters will stop from advancing any further. If the output is at logic "0" the counters will advance until a logic "1" is found. The counter speed is at 250 channels per second, which makes the next channel selection appear instantaneously. To advance to the next channel stored, the inputs A thru E have to return to the normal idle state and after the proper debounce time the memory up/down code entered again.

As an alternative to the non-volatile RAM an active channel search can be used. The search up/down code will be used whose speed is at 5 channels per second. The 200 ms time between channels should be sufficient to make a decision if a channel is present or not. There is an internal pull-up, so that if this feature is not used no connection is necessary to this input.

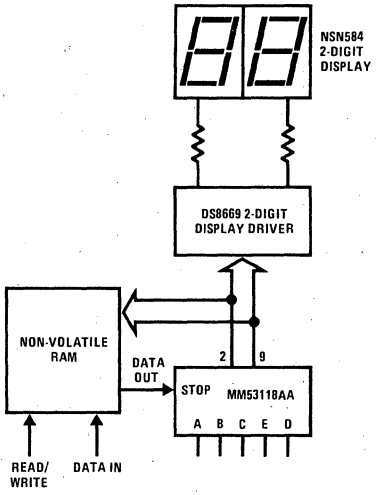



FIGURE 7

National Semiconductor



# MM5430, MM5431 AM/FM Radio Frequency Display

## General Description

The MM5430, MM5431 are monolithic N-channel MOS integrated circuits containing all the necessary logic, timing, control switching and segment drive circuits to display AM or FM radio tuning frequency on a 3 1/2-digit LED display.

The radio station frequency is determined by measuring the radio local oscillator and subtracting the IF frequency. A 10 Hz signal, derived from the 50 or 60 Hz input or from the 4.194304 crystal oscillator and divider circuit, is used to gate the internal digit counter, whose outputs are latched, then decoded into 7-segment format. Three FM IF frequencies of 10.6 MHz to 10.8 MHz (0.1 MHz steps) may be selected by appropriate connection of package pin to supply lines. Three AM IF frequencies of 262.5 kHz, 455 kHz, or 460 kHz may also be selected by connection of package pin*. Two outputs display mode of operation—AM or FM. The display LSD is configured to only show odd integers in the USA FM mode (106.9, 107.1, 107.3, etc.) and held at zero in the USA AM mode (850, 860, 870, etc.). USA/Europe pin select allows even and odd integers in FM (106.8, 106.9, 107.0) and 1 kHz resolution in AM (850, 851, 852). The MM5430, MM5431 interfaces directly with 7-segment LED displays and can drive up to 15 mA/segment for use with low-efficiency green displays. Brightness control is easily obtained by using a photo-resistor or potentiometer for manual operation.

The 4.194304 MHz crystal oscillator and countdown to 60 Hz run from a separate power pin for car clock operation.

*See Table II for selection details.

A 60 Hz output signal is available when the 4 MHz crystal oscillator operation is selected.

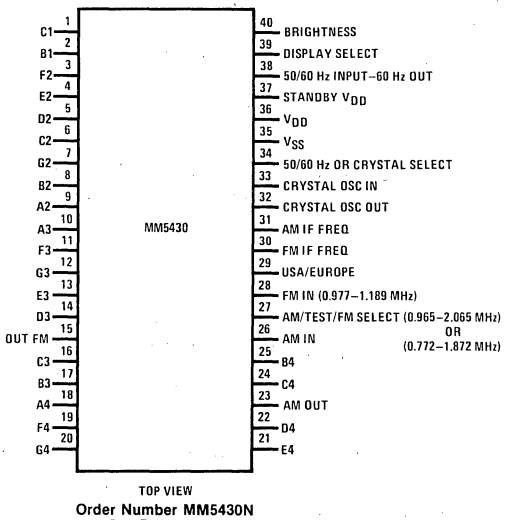
The MM5431 is electrically identical to the MM5430, but with mirror-image pin out. Both devices are supplied in a 40-lead dual-in-line package.

## Features

- AM and FM frequency display
- Programmable IF offset—FM and AM
- Single power supply, 7—11V
- 50/60 Hz line operation or 4.194304 MHz crystal operation
- 60 Hz output from crystal oscillator
- Separate power pin for crystal oscillator and countdown for automobile clock use
- AM and FM indicator outputs
- USA/Europe display resolution selection
- Non-multiplexed direct interface to LED displays
- Single pin brightness control capability
- RFI elimination slow up circuitry at the outputs
- Display enable input (frequency display for 4 seconds after switch release)
- Self select to display enable mode when input frequency changes
- Tuning hysteresis to eliminate display instability

## Connection Diagram

Dual-In-Line Package




TOP VIEW

**Order Number MM5430N**
**See Package 24**

**FIGURE 1**

## Absolute Maximum Ratings (Note 1)

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ to $V_{SS}$ +12V |
| Operating Temperature | −40°C to +85°C |
| Storage Temperature | −65°C to +150°C |
| Segment Drive Current *(Figure 4)* | ≤ 15 mA |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

| PIN NO. | PIN DESCRIPTION | CONDITIONS | | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|---|---|
| 37 | Standby Voltage | Crystal Oscillator In Use | | 5 | | 11 | V |
| | Standby Current | Crystal Oscillator In Use | | | | 10.0 | mA |
| 36 | $V_{DD}$ | | | 7 | | 11 | V |
| | $I_{DD}$ | | | | 13 | 20 | mA |
| **INPUTS** | | | | | | | |
| 26 | AM Osc. In | Sinewave | | 1 | | 2 | Vp-p |
| | | Frequency for Display | | 0.5 | | 2.2 | MHz |
| 28 | FM Osc. In | Squarewave | Logic "0" | $V_{SS}$ | | $V_{SS}$+0.5 | V |
| | | | Logic "1" | 4.0 | | $V_{DD}$ | V |
| | | Frequency for Display | | 0.5 | | 1.5 | MHz |
| 33 | Osc. In | Sinewave | | 0.5 | | | Vp-p |
| | | Frequency | | | | 4.2 | MHz |
| 38 | 50/60 Hz Input | Pin 34 is Not | Logic "0" | $V_{SS}$ | | $V_{SS}$+0.5 | V |
| | | Selecting Crystal | Logic "1" | $V_{DD}$−1 | | $V_{DD}$ | V |
| | | Frequency | | DC | 50/60 | 100k | Hz |
| 29 | USA/Europe | | Logic "0" | $V_{SS}$ | | $V_{SS}$+0.5 | V |
| | | | Logic "1" | $V_{DD}$−1 | | $V_{DD}$ | V |
| 27 | AM/Test/FM | | | | | | |
| 31 | AM IF | $V_{IN}$ = $V_{SS}$ Sink | Logic "0" | | | 200 | μA |
| 30 | FM IF | $V_{IN}$ = $V_{DD}$ Source | Logic "1" | | | 2 | mA |
| 34 | 50/60/Crystal | | | | | | |
| 40 | Brightness | $V_{IN}$ = $V_{DD}$ @ Max. Brightness | | | | 5 | mA |
| 39 | Display Select | @ 1 μA | Logic "0" | | | $V_{SS}$ | V |
| | | | Logic "1" | $V_{SS}$+1.0 | | $V_{DD}$ | V |
| **OUTPUTS** | | | | | | | |
| 1-25 | Segment Drives | @ Max. Brightness, $V_{OUT}$ = 2V | | 15 | | | mA |
| | | Outputs Off Leakage, $V_{OUT}$ = $V_{DD}$ | | | | 10 | μA |
| 32 | Osc. Out | $I_{SINK}$ @ 4 mA | Logic "0" | | | 1.0 | V |
| | | $I_{SOURCE}$ @ 0.8 mA | Logic "1" | 1.0 | | | V |
| 38 | 60 Hz Output | @ $V_{OUT}$ = $V_{SS}$+0.5V | | 0.05 | | | mA |
| | | @ $V_{OUT}$ = $V_{DD}$−3.0V | | 20 | | | μA |
| 39 | Display Select | Output Device ON @ 1V | | 1.7 | | | mA |

**Note 1:** Exceeding the Absolute Maximum Ratings may permanently damage the device.

## MM5430 Block Diagram

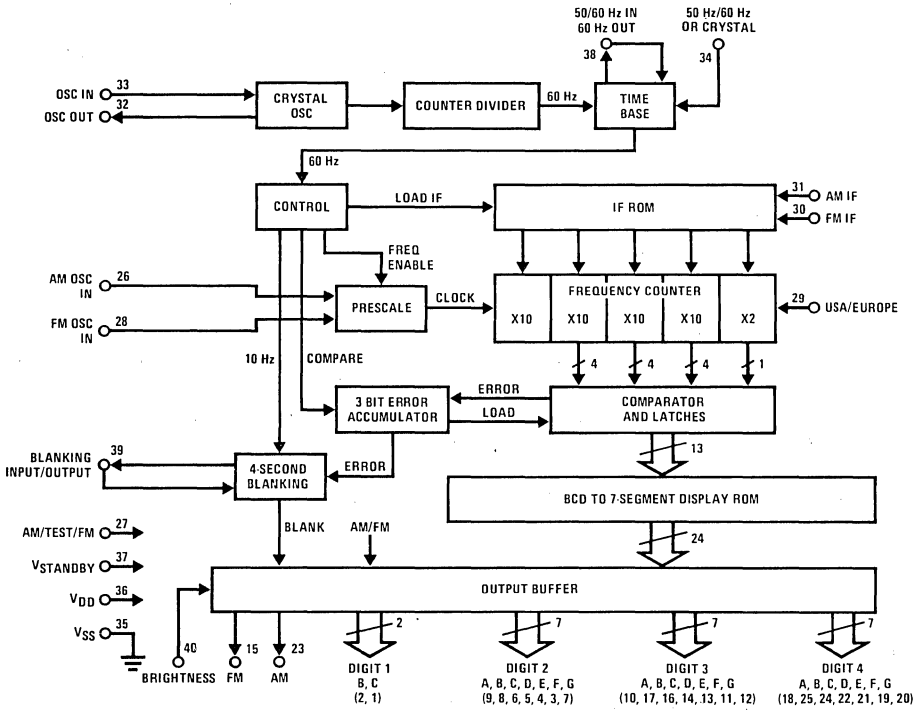



FIGURE 2

## Pin Functional Description

**1-25. Segment outputs.** 25 open-drain outputs are provided with direct drive capability—15 mA max. per segment—to the AM and FM indicators and to all the segments of a 3-1/2-digit display.

**26. AM oscillator input.** The AM band oscillator can be connected to the AM oscillator input with a capacitor in series. A pre-amplifier circuit is provided internally to facilitate interfacing with the AM local oscillator. This input accepts 1 to 2 Vp-p signals up to 2.2 MHz.

**27. AM/FM band select.** When this input is pulled low ($V_{SS}$), the AM band is selected and when it is pulled high ($V_{DD}$), the FM band is selected. If left floating, the device will go into a special test mode, Table 2.

**28. FM oscillator input.** The FM oscillator input accepts a ÷ 100 prescaled frequency from the local oscillator.

**29. USA/Europe.** This input selects the USA/Europe modes of operation. Table 1 shows the channel spacing selections. $V_{SS}$ selects Europe.

**30. FM IF.** Three FM IF frequencies are selectable through this input. Table 2 shows the selections.

**31. AM IF.** Three AM IF frequencies are selectable through this input. Table 2 shows the selections.

**32-33. Oscillator output, oscillator input.** These 2 pins form direct connections to a 4.194 MHz crystal. Pin 32 forms the low impedance output and pin 33 forms the high impedance input.

**34. 50/60 Hz or crystal select.** Time base can be selected from external 50 Hz or 60 Hz sources or the divided down 60 Hz from the crystal oscillator. Table 2 shows the selections.

**37. Standby $V_{DD}$.** The crystal oscillator and the circuit that divides down and outputs 60 Hz operates off the standby $V_{DD}$ supply. When only the 60 Hz output is required, the rest of the circuit therefore does not draw power.

**38. 50/60 Hz input—60 Hz output.** Depending on the state of pin 34, external 50/60 Hz or internal 60 Hz can be selected for input or output on this pin.

**39. Display select.** This pin exhibits internal low impedance to $V_{SS}$ as long as the display segment drives are active. If it is externally connected to $V_{SS}$, display segment drives are constantly active. *Figure 3* shows the display scheme.

**40. Brightness.** Variable voltages applied will generate different current drives on the segment outputs. Maximum drive occurs when this pin is connected to $V_{DD}$ directly.

## Pin Functional Description (Continued)

TABLE I

| MODE | PIN 29 | CHANNEL SPACING | |
|---|---|---|---|
| | | AM BAND | FM BAND |
| Europe | "0" | 1 kHz | 100 kHz |
| USA | "1" | 10 kHz | 200 kHz |

TABLE II

| | BAND SELECT | EUROPE FM IF | USA FM IF | AM IF | TIME BASE |
|---|---|---|---|---|---|
| PIN | 27 | 30 | 30 | 31 | 34 |
| "0" | AM | 10.8 | 10.75 | 262.5 | External 60 Hz |
| Floating | Test | 10.7 | 10.65 | 455 | Crystal 60 Hz |
| "1" | FM | 10.6 | 10.55 | 460 | External 50 Hz |

## Display Selection

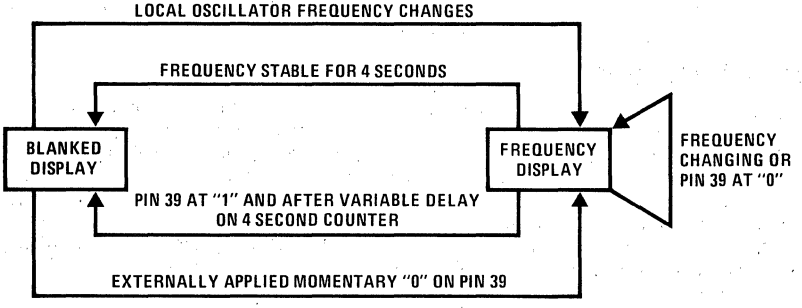



FIGURE 3

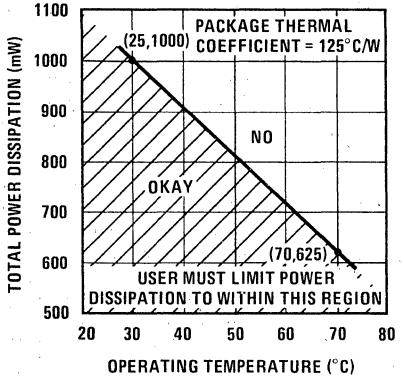



FIGURE 4

## Typical Applications

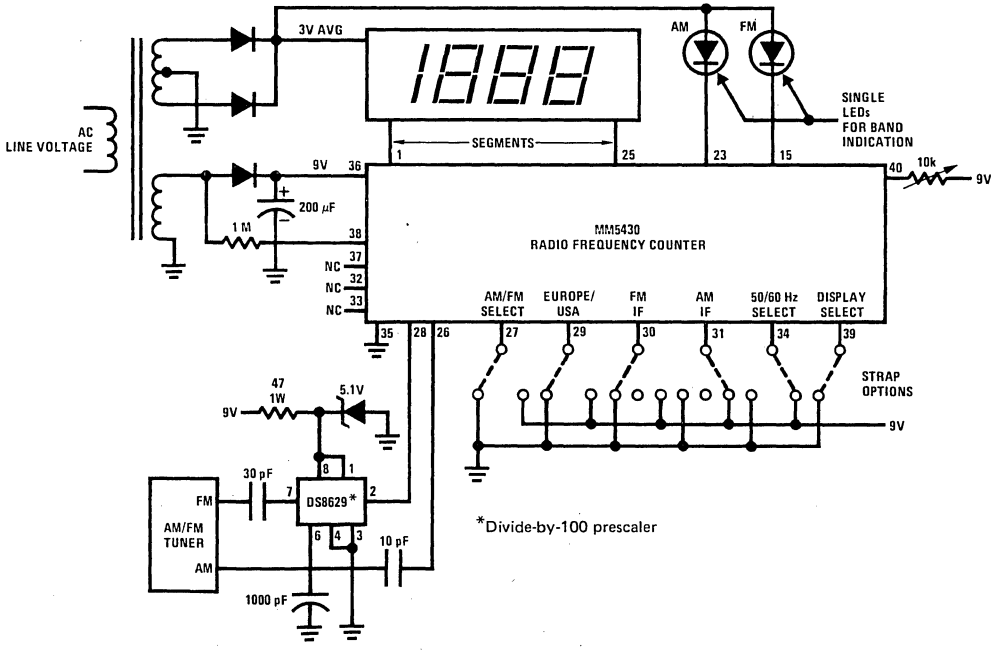



FIGURE 5a. AM/FM Frequency Counter with 50/60 Hz Input from Line Source

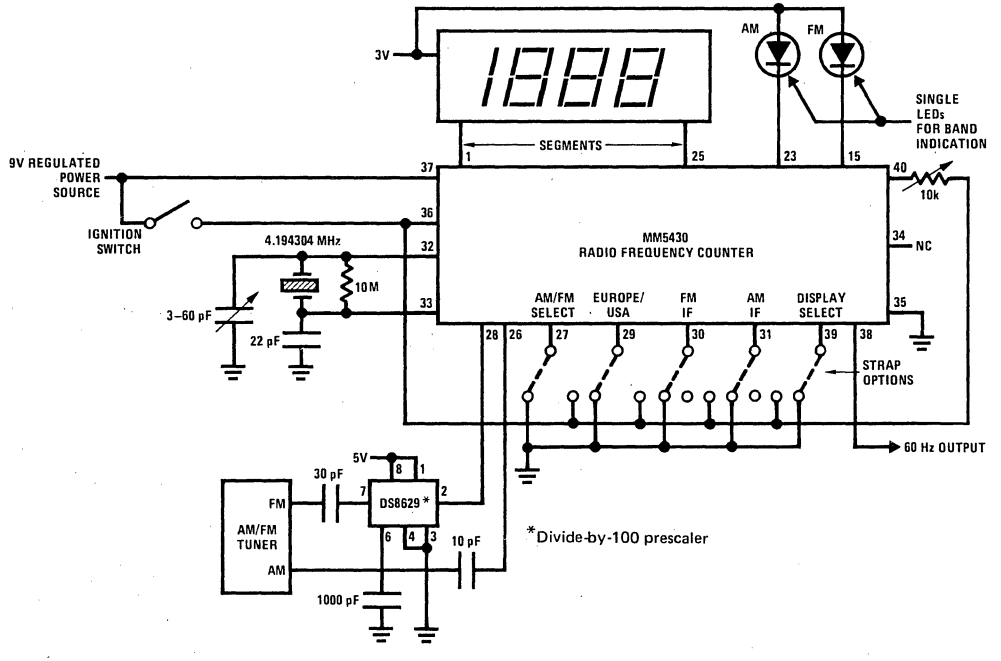



FIGURE 5b. AM/FM Frequency Counter in Automotive Environment with Crystal Oscillator Frequency Source

National Semiconductor

Television/Radio

# MM5439 Microprocessor Compatible Phase Lock Loop

## General Description

The MM5439 is a low threshold N-channel silicon gate bus orientated circuit that can be used in frequency synthesis applications in TV or radio receivers. The PLL operates on the pulse swallower technique and controls the frequency of a VCO by means of the up-down outputs. Six potentiometers are provided which can be used to control such functions in the TV as volume, color, brightness, etc. Extended I/O capability can be achieved by the use of the 6 general purpose latches and the 7 I/O ports which can be written to and read by the microprocessor.

## Features

- MICROBUS™ compatible
- Fast locking PLL
- Critically damped second order response
- Uses pulse swallower technique
- 10-bit main counter, 4-bit swallow counter
- Up to 4 MHz reference frequency input
- Up to 4 MHz tuning frequency input
- 62.5 kHz resolution for use in TVs
- 6 microprocessor controlled potentiometers
- 6 microprocessor controlled latches
- 7 microprocessor controlled I/O ports
- Easily interfaced to microprocessor
- TTL compatible
- Uses single 5V power supply

## Block Diagram

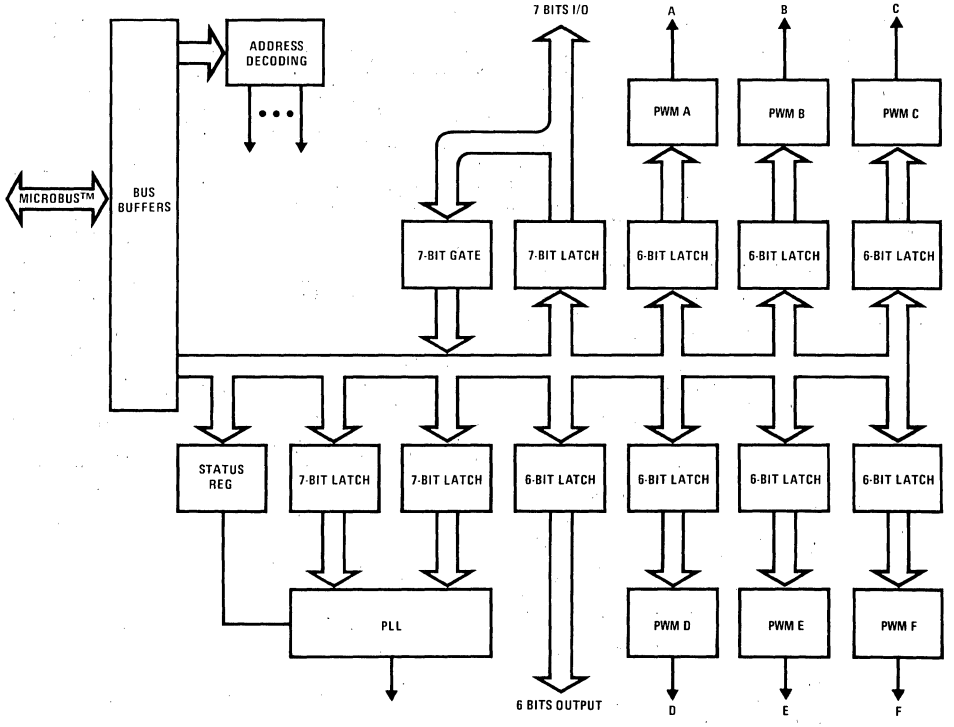



FIGURE 1

MICROBUS™ is a trademark of National Semiconductor Corp.

## Absolute Maximum Ratings

| | |
|---|---|
| Max Voltage at Pins 4 to 13, 18 to 27, 38, 39 | −0.5V to +13.5V |
| Max Voltage at All Other Pins | −0.5V to +7.0V |
| Operating Temperature Range | 0°C to 70°C |
| Storage Temperature Range | −65°C to +150°C |

## Electrical Characteristics $T_A = 0°C$ to $70°C$, $V_{SS} = 0V$, $V_{DD} = 5.0V$ +5% unless otherwise specified.

| Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| $V_{DD}$ Supply Voltage | | 4.75 | | 5.25 | V |
| $I_{DD}$ Supply Current | | | | 50 | mA |
| Input Logic Levels for Signals AD0-AD3, DB0-DB6, NWDS, NRDS, CLOCK, Chip Select | $V_{DD} = 5.0V$ | | | | |
| Logic 1 | | 2.0 | | | V |
| Logic 0 | | | | 0.8 | V |
| General Purpose I/O Hysteresis Inputs (Note 1) | | | | | |
| Logic 0-Logic 1 | | 3.2 | | | V |
| Logic 1-Logic 0 | | | | 1.7 | V |
| All Inputs | | | | | |
| Input Capacitance | | | | 10 | pF |
| Input Leakage | | | | 10 | μA |
| Output Logic Levels | | | | | |
| General Purpose Outputs and D-A A thru F (Note 2) | Open Drain | | | | |
| Logic 0 | $V_{DS} = 1V$ | 5 | | | mA |
| Logic 1 (Off) | $V_{DS} = 12V$ | | | 10 | μA |
| DB0-DB6, Dual Modulus Control | | | | | |
| Logic 0 | $I_L = -1.6$ mA | | | 0.4 | V |
| Logic 1 | $I_L = 0.6$ mA | 2.4 | | | V |
| DB0-DB6 | | | | | |
| Hi-Z | $V_{SS} < V_O < V_{DD}$ | | | 10 | μA |
| Up-Down Outputs | | | | | |
| Low | $I_L = -1.0$ mA | | | 100 | mV |
| High | $V_{DS} = 12V$ | | | 10 | μA |
| General Purpose I/O | | | | | |
| Logic 0 | $V_{DS} = 1V$ | 1 | | | mA |
| Logic 1 | $V_{DS} = 12V$ | | | 10 | μA |

**Note 1:** Maximum allowable input voltage on general purpose I/O is 13.2V

**Note 2:** Maximum allowable voltage on D-A and general purpose outputs is 13.2V

## Functional Description (Continued)

**TABLE I. REGISTER ADDRESSING**

| Function | Address | | | | | |
|---|---|---|---|---|---|---|
| | A3 | A2 | A1 | A0 | HEX | |
| Load D-A A | 0 | 0 | 0 | 0 | 0 | Write |
| Load D-A B | 0 | 0 | 0 | 1 | 1 | Write |
| Load D-A C | 0 | 0 | 1 | 0 | 2 | Write |
| Load D-A D | 0 | 0 | 1 | 1 | 3 | Write |
| Load D-A E | 0 | 1 | 0 | 0 | 4 | Write |
| Load D-A F | 0 | 1 | 0 | 1 | 5 | Write |
| PLL Bits 7-13 (MSB) | 0 | 1 | 1 | 0 | 6 | Write |
| PLL Bits 0-6 (LSB) | 0 | 1 | 1 | 1 | 7 | Write |
| 6-Bit Output | 1 | 0 | 0 | 0 | 8 | Write |
| 7-Bit Input/Output | 1 | 0 | 0 | 1 | 9 | Write/Read |
| A PLL Status | 1 | 0 | 1 | 0 | A | Write/Read |
| Test | 1 | 1 | 1 | 1 | F | Write |

### Tuning PLL

Two 7-bit write only registers latch tuning frequency data from the bus. The first is addressed by binary 8 appearing on the address lines AD0-AD3 and holds the 7 most significant bits of data; while the second holds the 7 LSBs and is accessed by an address 7.

When the two registers have been loaded with the new tuning frequency data, the load logic is primed by writing a 1 in DB6 of the status register (AD = HEX A). The data will then be transferred to the 14-bit register on the next all zero detect pulse from the 10-bit programmable counter. Data bit DB6 of the status register is reset at the same time as this data transfer and can be used as a ready/busy indication.

The ten most significant bits of the tuning data are used to parallel load the 10-bit programmable counter such that the frequency of the pulses to the phase comparator is the input frequency from the prescaler divided by this 10-bit number.

At the same time as the 10 most significant bits of the tuning data are loaded into the programmable counter, the 4 least significant bits are loaded into the 4-bit programmable counter. This counter is clocked by the input from the prescaler and counts down to zero, holding the $15/\overline{16}$ control output high. As soon as a count of zero is reached, the counter stops counting and the $15/\overline{16}$ control is set low.

If N1 represents the number loaded into the ten most significant bits, then the total division ratio of the programmable counter plus a ÷ 15/16 prescaler is given by

$$16\,N1-N2$$

where N2 is the number loaded in the 4 LS bits.

This technique of using a swallow counter presupposes that the 10-bit programmable counter is never loaded with a number less than the modulus of the 4-bit swallow counter, e.g.

$N1 > 16$.

Ignoring this precondition can result in an indeterminate output frequency of the PLL or an irrecoverable latch-up condition.

The 10-bit counter takes the reference signal and divides this by 1024 to provide the reference input to the phase comparator. In the case of a 4.000 MHz system reference, a frequency of 976.5625 Hz is produced.

The phase error detector operates in two modes, depending on whether the phase difference between the outputs of the programmable and reference counters is greater or less than 0.72°.

For phase errors greater than 0.72°, the counter providing the earlier output is held in its initial condition, by means of preset or reset pulses respectively, until the second counter provides an output. Thus both counters start counting simultaneously keeping their phase difference to a minimum. (See *Figure 5*).

When the phase error is less than 0.72° (2 $\mu$s), reset pulses to the counters are no longer provided and the circuit functions as a classical phase lock loop providing up or down pulses on the phase comparator output proportional to the phase difference between the outputs of the two counters. The up and down outputs are active low open drain outputs intended to drive an operational amplifier connected as an integrator. An applications circuit is shown in *Figure 4*.

The incorporation of counter synchronization logic for large phase errors reduces considerably the time to attain lock after step changes of frequency.

Lock detection is performed by setting a flip-flop after 32 cycles of the reference counter output during which no preset/reset signals have been generated by the phase error detection circuit. This flip-flop may be read by the bus at DB5 with address HEX A on A0-A3, and is set at logic 1 to indicate lock.

### STATUS REGISTER

Data bit 4 of the status register controls the up and down outputs. Writing a logic 1 to DB4 allows either output to go active low under the control of the phase comparator. Writing a logic 0 to DB4 causes the up-down outputs to remain permanently high impedance. Table II defines the designations of the status register bits.

**TABLE II. STATUS REGISTER BIT DESIGNATIONS (ADDRESS HEX A)**

| Bit | | Read | Write |
|---|---|---|---|
| 6 | 0 | Not ready for new data | Do nothing |
| | 1 | Ready for new data | Transfer new data to 14-bit register |
| 5 | 0 | Circuit not in lock | No function |
| | 1 | PLL in lock | No function |
| 4 | 0 | Always 0 | Disable up-down outputs |
| | 1 | Always 0 | Enable up-down outputs |
| 0-3 | | Always 0 | No function |

## Pin Description

**Digital-Analog Converter Outputs — Pins 4,5,12,13,38 and 39:** 6-bit pulse width modulated outputs with open drain output transistors.

**General Purpose Outputs — Pins 6-11:** 6 latched outputs with open drain output transistors.

**General Purpose Input/Output — Pins 18, 19 and 21-25:** 7-bit input/output logic. For inputs hysteresis amplifiers are used, for outputs open drain transistors are used.

**Chip Select, NRDS, NWDS — Pins 1, 2 and 3:** These pins select the circuit and enable read or write operations. All these signals are active low.

**AD0-AD3 — Pins 14-17:** These are address bits that select the internal registers for read/write operations.

**DB0-DB6 — Pins 30-36:** These are bidirectional data bus pins for transferring data to and from the microprocessor.

**Clock — Pin 37:** The basic timing for the pulse width modulators is obtained from this signal — the maximum frequency is 4.5 MHz.

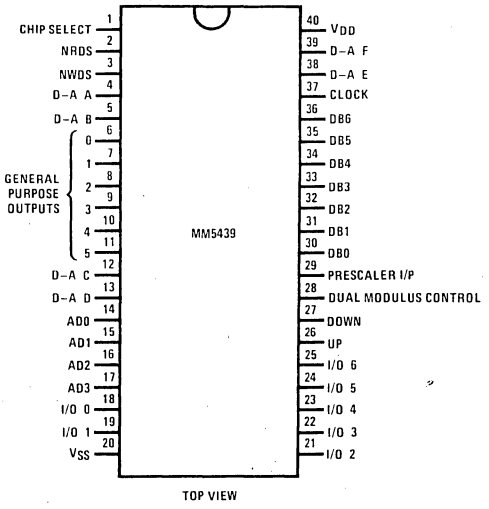
**Prescaler Input — Pin 29:** TTL compatible input from Dual Modulus Counter chain driven from TV tuner local oscillator.

**Dual Modulus Control — Pin 28:** TTL compatible output. Controls modulus of prescaler counter chain.

**Up-Down — Pins 26-27:** Output of PLL phase detector. Active low outputs interfacing to tuner via integrator.

## Connection Diagram

**Dual-In-Line Package**




**Order Number MM5439N**
**See Package 24**

## Functional Description

### CIRCUIT DESCRIPTION

The block diagram in *Figure 1* describes the requirements of the circuit. The package is a 40-pin DIP.

### HF Divider

This two stage divider takes the 4.00 MHz clock and produces two non-overlapping clocks which control the D to A converters and PLL.

### Potentiometers A-F

The data representing a particular potentiometer output is stored in a 6-bit latch. This latch is parallel loaded with the data on the I/O bus by the load command. Table I shows the address decoding required for each set of latches. The outputs of the latch are compared with the outputs of a 6-bit reference counter. When the two are equal, the output flip-flop is reset. The flip-flop is set during each zero crossing state of the reference counter. Thus a variable duty cycle appears at the output, variable up to 63 steps. Output frequency is approximately 17 kHz. These potentiometers could be used to control brightness, volume, color saturation, contrast, tone and fine tune. Loading 0 to these circuits will result in the least positive output voltage when integrated. See *Figure A1* for a typical application. See Appendix A for description.

### General Purpose Latches

These 6 latches store data from the microprocessor. The large number of outputs removes the need for decoders external to the chip. All envisaged band/standard switching functions can be carried out with these bits.

### General Purpose I/O

These 7 pins are used for either input or output under processor control. *Figure B1* illustrates the principle. The latched output controls an open drain output transistor. Writing a logic 1 in this bit turns off the output transistor thus permitting the use of this bit as an input. Data bus bits DB0-DB6 are used for the I/O. See Appendix B for description.

### Chip Select

The chip is selected by an active low signal from a separate peripheral select decoder.

### Register Addressing

Table I shows the coding on address lines (AD0 to AD3) which selects the register to be parallel loaded.

## Functional Description (Continued)

### I/O TIMING

#### Read Mode

*Figure 2* gives detailed timing in accordance with the MICROBUS™ Specification for Class 1 Microprocessors for the transfer of data from peripheral to microprocessor (Table III).

All times are measured from (or to) valid logic 0 level = 0.8V or valid logic 1 level = 2.0V.

#### Write Mode

*Figure 3* gives detailed timing in accordance with the MICROBUS Specification for Class 1 Microprocessors for the transfer of data from microprocessor to peripheral (Table IV).

**TABLE III. TIMING: DATA FROM PERIPHERAL TO MICROPROCESSOR**

| Symbol | Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $t_{ACS0}$ | Address Bus Valid to Chip Select ON (CS = 0) | | | 40 | Note 1 | ns |
| $t_{CSR}$ | Chip Select ON to Read Strobe | Note 2 | 70 | | | ns |
| $t_{AR}$ | Address Bus Valid to Read Strobe | | 250 | | | ns |
| $t_{RD}$ | Read Cycle Access Time from Read Strobe to Data Bus Valid | $C_L$ = 100 pF | | | 375 | ns |
| $t_{RH}$ | Data Hold Time from Trailing Edge of Read Strobe | | 0 | | 250 | ns |
| $t_{RA}$ | Address Bus Hold Time from Trailing Edge of Read Strobe | | 50 | 500 | | ns |
| $t_{ACS1}$ | Address Change to Chip Select OFF | | | 40 | Note 1 | ns |
| $t_{AD}$ | Address Bus Valid to Data Valid | $C_L$ = 100 pF | 560 | | | ns |
| $t_{HZ}$ | Time from Trailing Edge of Read Strobe until Interface Device Bus Drivers are in TRI-STATE® Mode | | 0 | | 250 | ns |

**TABLE IV. TIMING: DATA FROM MICROPROCESSOR TO PERIPHERAL**

| Symbol | Parameter | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| $t_{ASC0}$ | Address Bus Valid to Chip Select ON (CS = 0) | | 40 | Note 1 | ns |
| $t_{CSW}$ | Chip Select ON to Write Strobe | 310 | 450 | | ns |
| $t_{AW}$ | Address Bus Valid to Write Strobe | 350 | | | ns |
| $t_{WW}$ | Write Strobe Width | 430 | | | ns |
| $t_{DW}$ | Data Bus Valid before Write Strobe | 200 | | | ns |
| $t_{WD}$ | Data Bus Hold Time Following Write Strobe | 100 | | | ns |
| $t_{WA}$ | Address Bus Hold Time Following Write Strobe | 50 | | | ns |
| $t_{ACS1}$ | Address Change to Chip Select OFF (CS = 1) | | 40 | Note 1 | ns |

**Note 1:** The maximum value of this parameter is dependent on the implementation of the chip select circuit.

**Note 2:** $t_{CSR} = t_{AR} - t_{ACS0}$

TRI-STATE® is a registered trademark of National Semiconductor Corp.

## Functional Description (Continued)

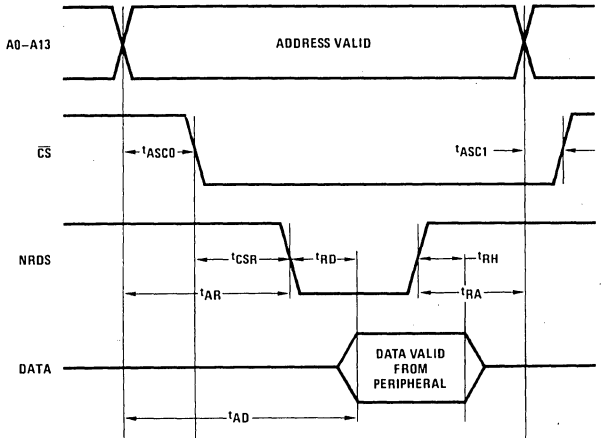



FIGURE 2. Read Cycle

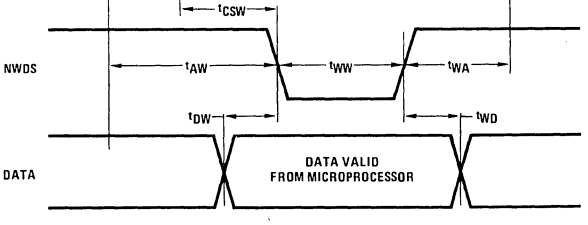



FIGURE 3. Write Cycle

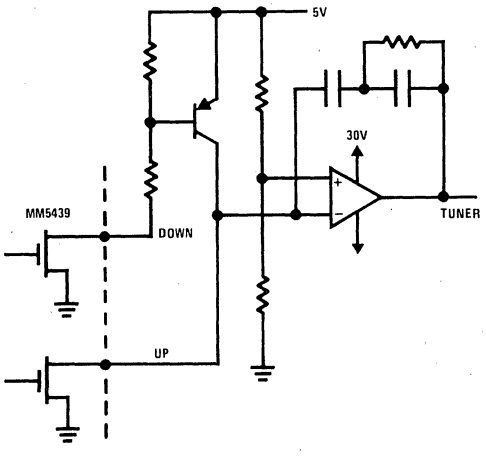
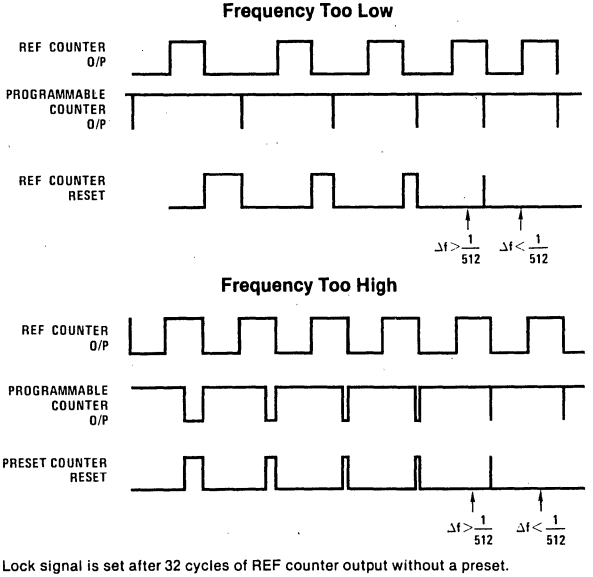



FIGURE 4. Typical PLL Output Circuit



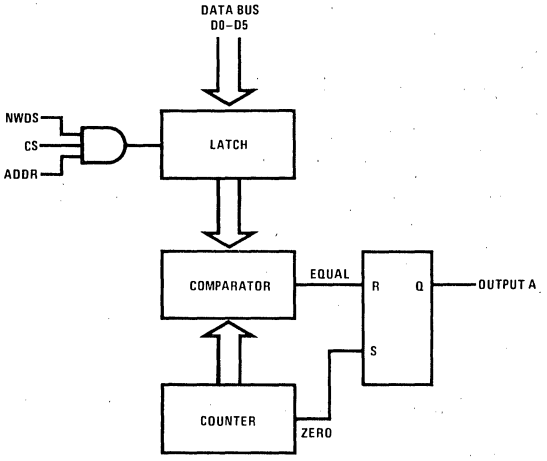

Lock signal is set after 32 cycles of REF counter output without a preset.

FIGURE 5

## APPENDIX A

### Potentiometer Voltage Generation

This integrated circuit contains 6 potentiometers of the type shown in *Figure A1*. These are used to provide waveforms of varying duty cycle which can be integrated and, as an analog voltage, used to control volume, brightness, color, etc. in a TV set. The circuit operation is shown in *Figure A1*.




Pulse width modulated output for analog functions
6-bit resolution/63 discrete steps

**FIGURE A1**

The flip-flop is set by a circuit which detects the all 0's state of the 6-bit counter. A 6-bit binary word is stored by the microprocessor in the latch and its value is compared with the state of the counter.

When the comparator detects coincidence a signal resets the flip-flop, thus the control word determines the mark space ratio of the signal produced at output A. The PRF of this waveform is 1/64 MHz with a duty cycle variable in 63 steps. This is shown in *Figure A2*.

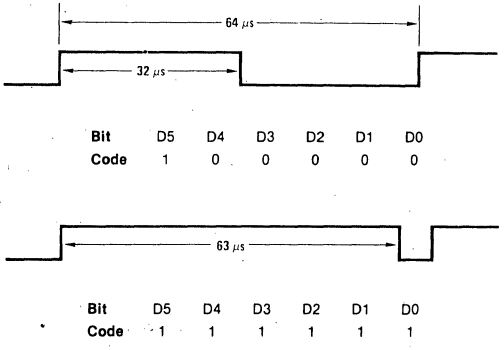



**FIGURE A2. Output Voltages for Two Different Digital Values**

## APPENDIX B

### I/O Function

The seven pins dedicated to bidirectional operation are identical, the logic diagram is given in *Figure B1*.

Output is quite straightforward — data present on the data bus is latched by a write strobe, with the correct address, and presented at the output. However it is clear, if the output transistor is on, then the pin cannot be used as an input. Thus a pin used as an input must have been previously set, as an output, to a logic 1.

The signal read at the pin is a wired-OR function. As the 7 I/O bits are addressed at the same time by the microprocessor, it is important to note (in machine code programming of the circuit) that by writing all bits used for inputs, they are set to 1. Thus subsequent read operations do not need to be preceded by a write.

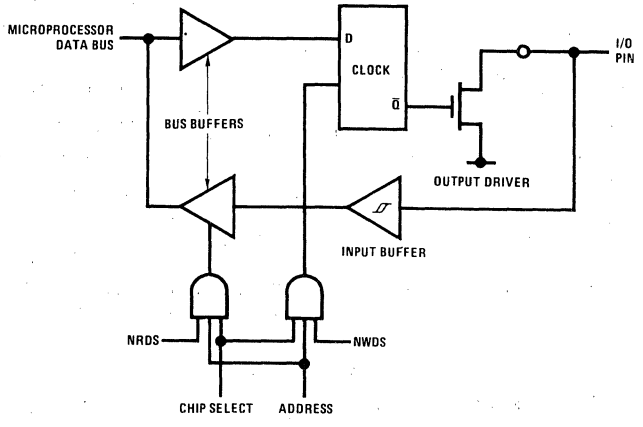



**FIGURE B1. Logic of Input/Output Circuitry**

National Semiconductor



# MM55108, MM55110 PLL Frequency Synthesizer with Receive/Transmit Mode

## General Description

The MM55108 and MM55110 PLL frequency synthesizers are monolithic metal gate CMOS integrated circuits which contain phase locked loop circuits useful for frequency synthesis applications. The devices operate from a single power supply and contain an oscillator with feedback resistor, divider chain, a binary input programmable divider with control logic for the transmit mode (÷ by (N + 91)), and the necessary phase detector logic. The devices may be used in double IF or single IF systems.

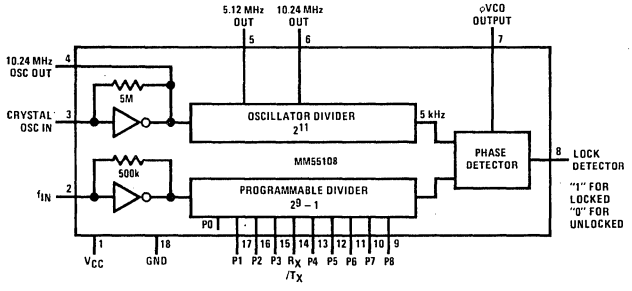
Both the MM55108 and the MM55110 use a 10.24 MHz quartz crystal to determine the reference frequency. The MM55108 has a $2^{11}$ divider chain which generates a 5 kHz reference frequency. The MM55110 has a selectable $2^{10}$ or $2^{11}$ divider chain which gives either a 10 kHz or 5 kHz reference frequency. The selection of reference frequency is made by use of the FS pin. In addition, the MM55110 contains an amplifier for filter applications and an additional input to the programmable divider which allows $2^{10} - 1$ division of the input frequency ($f_{IN}$) for FM applications. Due to the internal amplifier stage at input frequency input ($f_{IN}$), the MM55108 and MM55110 may take a 0.5 Vp-p signal at $f_{IN}$ as the input frequency for the programmable divider. Inputs to the programmable divider are standard binary signals. Selection of a channel is accomplished by mechanical switches or by external electronic programming of the programmable divider. The $\phi$VCO output provides a high level voltage (sources current) when the $\phi$VCO frequency is lower than the lock frequency, and $\phi$VCO provides a low level voltage (sinks current) when the $\phi$VCO frequency is higher than the lock frequency. The $\phi$VCO output goes to a high impedance state (TRI-STATE®) while in lock mode, and the lock detector output LD also goes to a high state under lock condition.

## Features

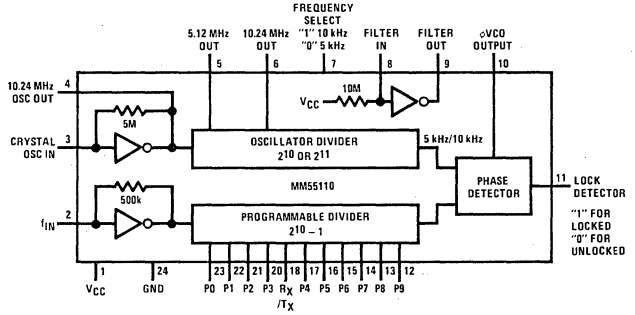
- Single crystal operation
- Single power supply
- Low power CMOS technology
- Binary input channel select code
- $2^{10}$ or $2^{11}$ divider chain from oscillator input (MM55110), $2^{11}$ divider chain (MM55108)
- Buffered 5.12 MHz and buffered 10.24 MHz outputs
- On-chip oscillator with bias resistor
- Pull-down resistors on programmable divider inputs
- Receive/transmit input for ÷ by (N+91) while in transmit mode
- Amplifier for filter applications (MM55110)
- Programmable $2^9 - 1$ division of $f_{IN}$
- Additional programmable input for $2^{10} - 1$ division of $f_{IN}$ (MM55110)
- Amplifier stage on $f_{IN}$ input to accept 0.5 Vp-p signal

## Block Diagrams

MM55108




MM55110



## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{CC}$ + 0.3V to Gnd − 0.3V |
| Operating Temperature Range | −30°C to +75°C |
| Storage Temperature | −40°C to +125°C |
| Operating $V_{CC}$ | 12V |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating temperature range, GND = 0V, unless otherwise specified

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Supply Voltage ($V_{CC}$) | | 4.5 | | 10.0 | V |
| Supply Current ($I_{CC}$) | Freq. at Osc. In = 10.24 MHz at $f_{IN}$ = 2.5 MHz, All Other I/O Pins Open | | | | |
| | $V_{CC}$ = 5V | | 10 | 15 | mA |
| | $V_{CC}$ = 10V | | 30 | 45 | mA |
| Logical "1" Input Voltage ($V_{IN(1)}$) | | | | | |
| P0–P9, FS, $R_X/T_X$ | | $V_{CC}$−0.5 | | | V |
| Logical "0" Input Voltage ($V_{IN(0)}$) | | | | | |
| P0–P9, FS, $R_X/T_X$ | | | | 0.5 | V |
| Logical "1" Output Voltage ($V_{OUT(1)}$) | | | | | |
| Osc. Out, 10.24 MHz Out, 5.12 MHz Out, LD, $\phi$VCO, Filter Out | $I_{OUT}$ = −0.5 mA | $V_{CC}$−0.5 | | | V |
| Logical "0" Output Voltage ($V_{OUT(0)}$) | | | | | |
| Osc. Out, 10.24 MHz Out, 5.12 MHz Out, LD, $\phi$VCO, Filter Out | $I_{OUT}$ = 0.5 mA | | | 0.5 | V |
| Logical "1" Input Current ($I_{IN(1)}$) | | | | | |
| Filter In (Pull-Up) | $V_{CC}$ = 4.5V, $V_{IN}$ = 4V | −300 | | −50 | nA |
| | $V_{CC}$ = 10V, $V_{IN}$ = 9.5V | −500 | | −100 | nA |
| $R_X/T_X$ (Pull-Up) | $V_{CC}$ = 4.5V, $V_{IN}$ = 4V | −500 | | −40 | μA |
| | $V_{CC}$ = 10V, $V_{IN}$ = 9.5V | −600 | | −50 | μA |
| FS, P0–P9 (Pull-Down) | $V_{CC}$ = 4.5V, $V_{IN}$ = 4V | 4 | | 40 | μA |
| | $V_{CC}$ = 10V, $V_{IN}$ = 9.5V | 20 | | 200 | μA |
| Logical "0" Input Current ($I_{IN(0)}$) | | | | | |
| Filter In (Pull-Up) | $V_{CC}$ = 4.5V, $V_{IN}$ = 0.5V | −600 | | −100 | nA |
| | $V_{CC}$ = 10V, $V_{IN}$ = 0.5V | −3.0 | | −0.5 | μA |
| $R_X/T_X$ (Pull-Up) | $V_{CC}$ = 4.5V, $V_{IN}$ = 0.5V | −800 | | −100 | μA |
| | $V_{CC}$ = 10V, $V_{IN}$ = 0.5V | −5.0 | | −0.8 | mA |
| FS, P0–P9 (Pull-Down) | $V_{CC}$ = 4.5V, $V_{IN}$ = 0.5V | 1 | | 10 | μA |
| | $V_{CC}$ = 10V, $V_{IN}$ = 0.5V | 2 | | 30 | μA |
| Maximum Toggle Frequency at $f_{IN}$ | $V_{CC}$ = 5V | 3 | | | MHz |
| | $V_{CC}$ = 7.5V | 5 | | | MHz |
| Input Signal at $f_{IN}$ | Small Signal (AC Coupled) or | 0.5 | | | Vp-p |
| | $V_{IN(1)}$ | $V_{CC}$−0.5 | | | V |
| | $V_{IN(0)}$ | | | 0.5 | V |
| Duty Cycle at $f_{IN}$ | | 30 | | 70 | % |
| Maximum Osc. Frequency at Osc. In | $V_{CC}$ = 5V, 10.24 MHz Crystal | 10.24 | | | MHz |
| TRI-STATE® Leakage at $\phi$VCO | $V_{OUT}$ = $V_{CC}$ or Gnd | | | \|±1\| | μA |

## Typical Applications

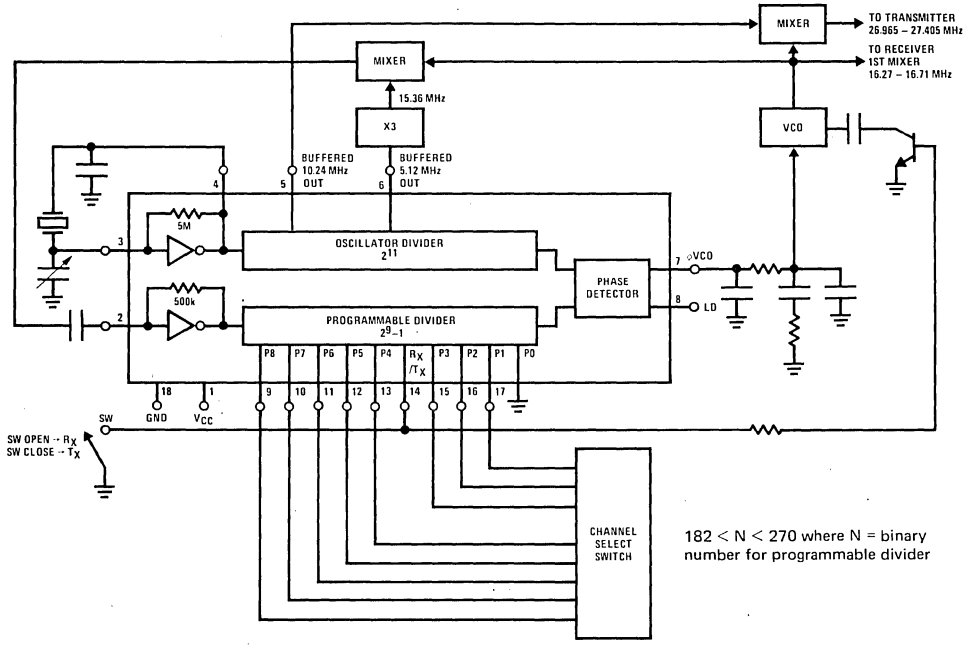



**FIGURE 1. MM55108 Single Crystal 40-Channel Low Side Injection**

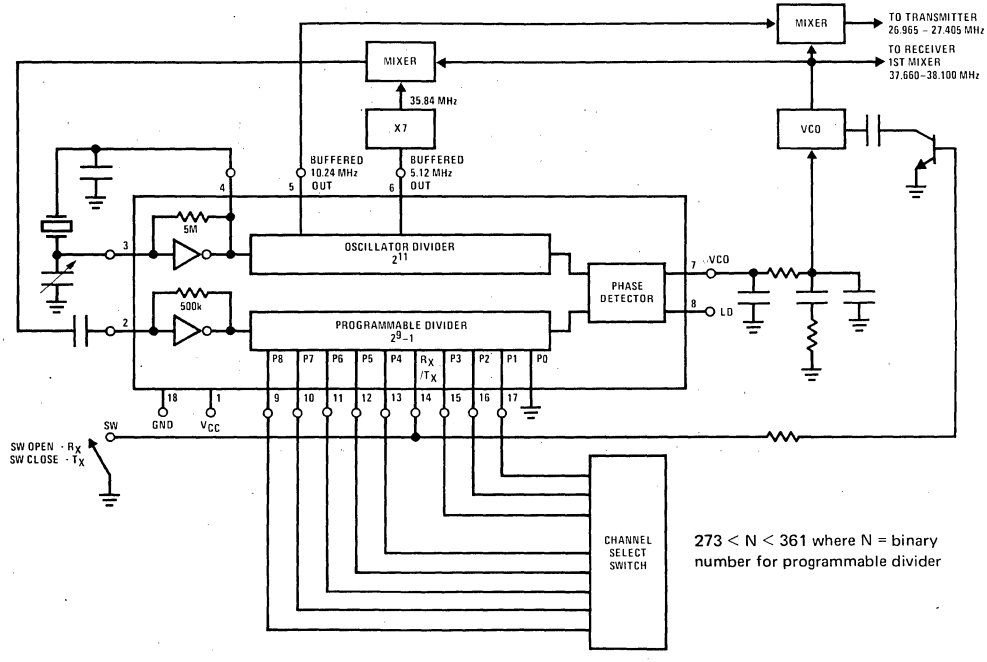



**FIGURE 2. MM55108 Single Crystal 40-Channel High Side Injection**

## Truth Tables

**TABLE I. Binary Inputs to Programmable Divider for MM55108**

| | $R_X/T_X$ "1" OR "OPEN" N | $R_X/T_X$ "0" OR "CLOSED" N | INPUTS $2^8$ P8 | $2^7$ P7 | $2^6$ P6 | $2^5$ P5 | $2^4$ P4 | $2^3$ P3 | $2^2$ P2 | $2^1$ P1 |
|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 92 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 2 | 93 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| | 4 | 95 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| | . | . | . | . | . | . | . | . | . | . |
| | . | . | . | . | . | . | . | . | . | . |
| Channel 1 → | 182 | 273 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| | . | . | . | . | . | . | . | . | . | . |
| | . | . | . | . | . | . | . | . | . | . |
| Channel 40 → | 270 | 361 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| | . | . | . | . | . | . | . | . | . | . |
| | . | . | . | . | . | . | . | . | . | . |
| | 510 | 601 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

1 = logical "1"
0 = logical "0"

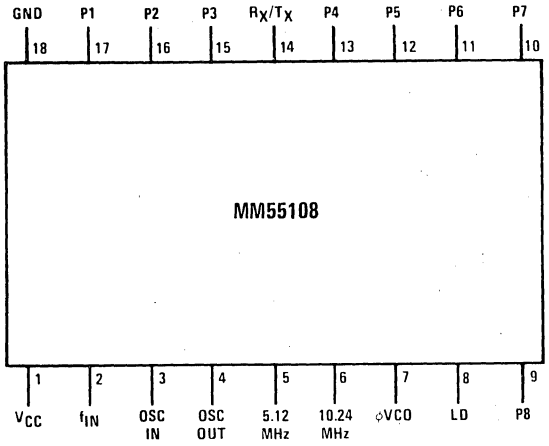
**TABLE II. Binary Inputs to Programmable Divider for MM55110**

| | $R_X/T_X$ "1" OR "OPEN" N | $R_X/T_X$ "0" OR "CLOSED" N | INPUTS $2^9$ P9 | $2^8$ P8 | $2^7$ P7 | $2^6$ P6 | $2^5$ P5 | $2^4$ P4 | $2^3$ P3 | $2^2$ P2 | $2^1$ P1 | $2^0$ P0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 92 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | X |
| | 2 | 93 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| | 3 | 94 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| | . | . | . | . | . | . | . | . | . | . | . | . |
| | . | . | . | . | . | . | . | . | . | . | . | . |
| Channel 1 → | 182 | 273 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| | . | . | . | . | . | . | . | . | . | . | . | . |
| | . | . | . | . | . | . | . | . | . | . | . | . |
| Channel 40 → | 270 | 361 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| | . | . | . | . | . | . | . | . | . | . | . | . |
| | . | . | . | . | . | . | . | . | . | . | . | . |
| | 1023 | 1114 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

X = don't care
1 = logical "1"
0 = logical "0"

## Connection Diagrams

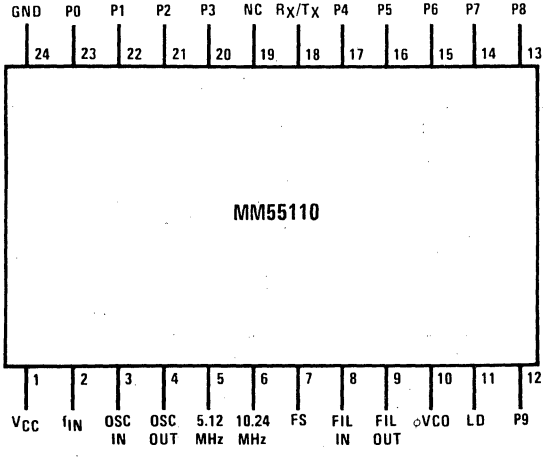
Dual-In-Line Package




TOP VIEW

Order Number MM55108N
See Package 20

Dual-In-Line Package




TOP VIEW

Order Number MM55110N
See Package 22

## Pin Descriptions

| | |
|---|---|
| P0–P9 | Programmable Divider Inputs |
| $f_{IN}$ | Frequency Input From VCO (Mixed down) |
| OSC IN | Oscillator Amplifier Input |
| OSC OUT | Oscillator Amplifier Output |
| LD | Lock Detector |
| $\phi$VCO | Output of Phase Detector for Control of VCO |
| FS | Frequency Division Select<br>"1" for $2^{10}$ Division<br>"0" for $2^{11}$ Division |
| 5.12 MHz OUT | Buffered 5.12 MHz Output (Oscillator Frequency ÷ By 2) |
| 10.24 MHz OUT | Buffered 10.24 MHz Output (Oscillator Frequency) |
| FILTER IN | Filter Amplifier Input |
| FILTER OUT | Filter Amplifier Output |
| $R_X/T_X$ | Receive/Transmit Input<br>"0" for Transmit Mode (÷ by (N+91)) |

National Semiconductor

Television/Radio

# MM55121 Serial Data/PLL Frequency Synthesizer

## General Description

The MM55121 device is a monolithic metal gate CMOS integrated circuit which contains a phase locked loop circuit useful for frequency synthesizer applications in the AM, FM, CB and SW frequency bands. It operates from a single power supply and contains an oscillator with a feedback resistor, a $2^{11}$ divider chain, a binary input programmable divider, and phase detector circuitry. Selection of a channel is accomplished by external programming of the programmable divider with a 13-bit serial code derived from a 16-bit data string fed to the SERIAL DATA IN. The serial data format consists of a leading logical "1" synchronization bit, two control bits that are latched and made available at OUT 1 and OUT 2 and a 13-bit binary input channel selection code.

The phase detector output, $\phi$VCO, provides a high level voltage (sources current) when the VCO frequency is lower than the lock frequency, and it provides a low level voltage (sinks current) when the VCO frequency is higher than the lock frequency. The $\phi$VCO output goes to a high impedance (TRI-STATE®) condition and the lock detector output LD goes to a high state under lock conditions.

## Features

- Single crystal operation
- Single power supply
- Low power CMOS technology
- On-chip oscillator with feedback resistor
- Buffered outputs: 10.24 MHz, 320 kHz, 300 Hz and 60 Hz
- Serial data input format consisting of a leading "1" synchronization bit, 2 control bits available at separate pins, and a 13-bit channel selection code
- $f_{IN}$ input amplifier stage to accept an AC coupled 0.8 Vp-p signal

## Block Diagram

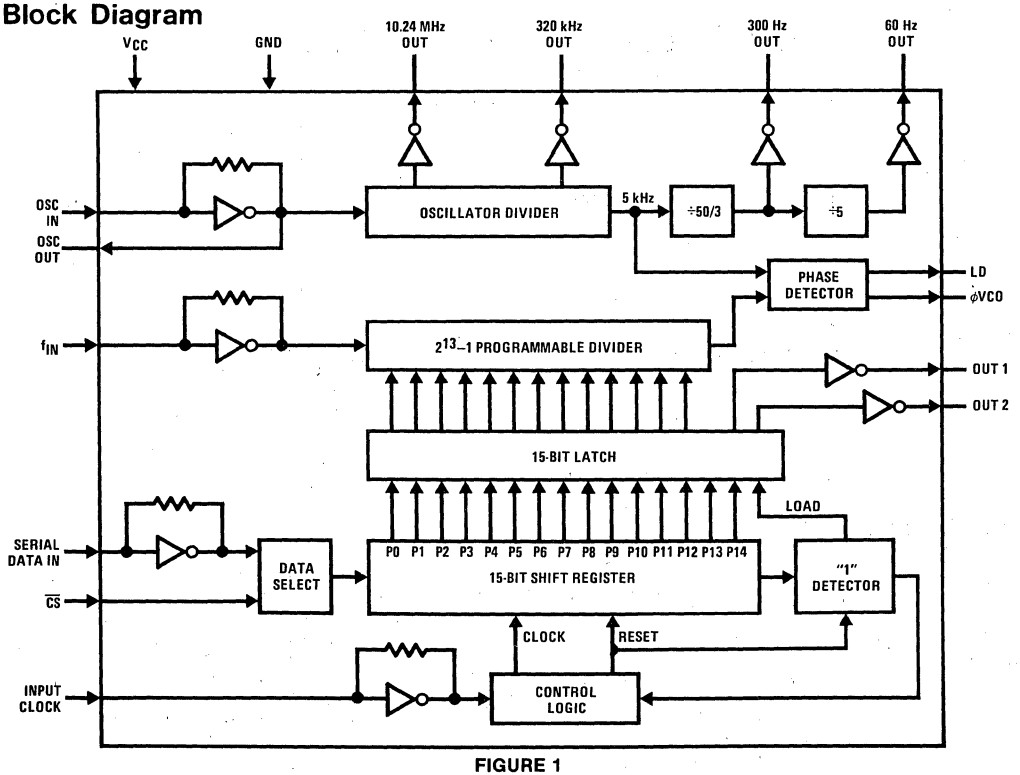



FIGURE 1

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{CC}$ + 0.3V to Gnd − 0.3V |
| Operating Temperature Range | − 30 °C to + 75 °C |
| Storage Temperature Range | − 40 °C to + 125 °C |
| Operating Supply Voltage, $V_{CC}$ | 12V |
| Lead Temperature (Soldering, 10 seconds) | 300 °C |

## Electrical Characteristics

Ambient temperature within the operating range and Gnd = 0V, unless otherwise specified.

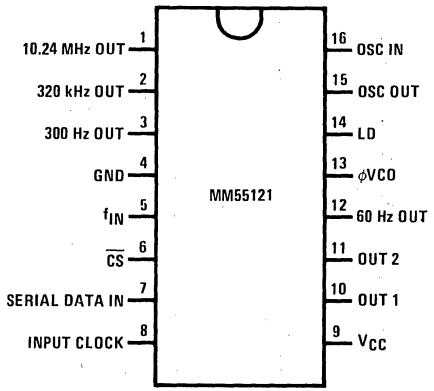
| Parameter | | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Supply Voltage ( $V_{CC}$) | | | 6.5 | | 10.0 | V |
| Supply Current ($I_{CC}$) | | Osc Frequency = 10.24 MHz<br>Prog. Divider Input ($f_{IN}$) Frequency = 4.5 MHz<br>$\overline{CS}$, Serial Data In, and Input Clock at Gnd; All Other Pins Open | | | | |
| | | $V_{CC}$ = 6.5V | | 20 | 30 | mA |
| | | $V_{CC}$ = 10.0V | | 30 | 45 | mA |
| LOGIC: | Logical "1" Input Voltage ($V_{IN(1)}$)<br>INPUT CLOCK<br>SERIAL DATA IN<br>$\overline{CS}$ | | 0.8 $V_{CC}$ | | | V |
| | Logical "0" Input Voltage ($V_{IN(0)}$)<br>INPUT CLOCK<br>SERIAL DATA IN<br>$\overline{CS}$ | | | | 0.2 $V_{CC}$ | V |
| | Logical "1" Output Voltage ($V_{OUT(1)}$)<br>OSC OUT<br>10.24 MHz OUT<br>LD<br>$\phi$VCO | $I_{OUT}$ = − 0.7 mA | 0.8 $V_{CC}$ | | | V |
| | Logical "0" Output Voltage ($V_{OUT(0)}$)<br>OSC OUT<br>10.24 MHz OUT<br>LD<br>$\phi$VCO | $I_{OUT}$ = 0.7 mA | | | 0.2 $V_{CC}$ | V |
| | Logical "1" Output Voltage ($V_{OUT(1)}$)<br>320 kHz OUT<br>300 Hz OUT<br>60 Hz OUT | $I_{OUT}$ = − 0.5 mA | 0.8 $V_{CC}$ | | | V |
| | Logical "0" Output Voltage ($V_{OUT(0)}$)<br>320 kHz OUT<br>300 Hz OUT<br>60 Hz OUT | $I_{OUT}$ = 0.5 mA | | | 0.2 $V_{CC}$ | V |

## Electrical Characteristics (Continued) Ambient temperature within the operating range and Gnd = 0V, unless otherwise specified.

| Parameter | | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $f_{IN}$: | Toggle Frequency | | 100k | | 4.5M | Hz |
| | Input Signal | Small Signal (AC-Coupled) or | 0.8 | | | Vp-p |
| | | $V_{IN(1)}$ | 0.8 $V_{CC}$ | | | V |
| | | $V_{IN(0)}$ | | | 0.2 $V_{CC}$ | V |
| | Duty Cycle at $f_{IN}$ | | 40 | | 60 | % |
| $f_{IN}$, SERIAL DATA IN, INPUT CLOCK: | Input Capacitance | | | 5 | 10 | pF |
| | Input Resistance Including Feed-back Resistor | | 100 | | | kΩ |
| OSC | Oscillator Frequency | | 10.24 | | | MHz |
| IN: | Oscillator Input Resistance Including Feedback Resistor | | 1 | | | MΩ |
| $\overline{CS}$: | Input Capacitance | | | 5 | 10 | pF |
| | Input Resistance | | 1 | | | MΩ |
| INPUT CLOCK: | Toggle Frequency | | DC | | 500 | kHz |
| | Clock Rise Time and Fall Time | | 0.5 | | | μs |
| ϕVCO: | TRI-STATE Leakage at $V_{OUT} = V_{CC}$ or Gnd | | −0.1 | | +0.1 | μA |
| DELAYS: | Select to Data Time Delay, $t_{sd}$ | | | | 0 | ns |
| | Data to Clock Time Delay, $t_{dc}$ | | | | 100 | ns |

## Connection Diagram

**Dual-In-Line Package**




TOP VIEW

Order Number MM55121N
See Package 19

**FIGURE 2**

**Pin Descriptions**

| | |
|---|---|
| $f_{IN}$ | Frequency Input from VCO (mixed down) |
| OSC IN | Oscillator Amplifier Input |
| OSC OUT | Oscillator Amplifier Output |
| LD | Lock Detector Output: "1" for locked condition; "0" for unlocked condition |
| ϕVCO | Output of Phase Detector for VCO Control |
| 10.24 MHz OUT | Buffered 10.24 MHz Output (Oscillator Frequency) |
| 320 kHz OUT | Buffered 320 kHz External Clock Output |
| 300 Hz OUT | Buffered 300 Hz Output |
| 60 Hz OUT | Buffered 60 Hz Output |
| SERIAL DATA IN | Serial Data Input |
| INPUT CLOCK | Internal Shift Register Clock |
| $\overline{CS}$ | Chip Select Input |
| OUT 1-2 | Control Outputs |

## Functional Description

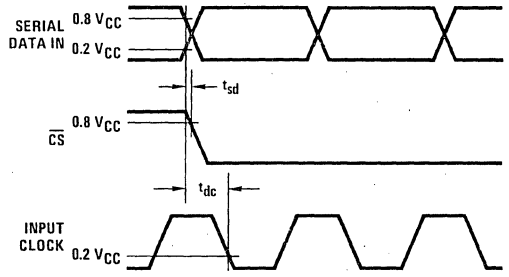
A block diagram of the MM55121 serial data/PLL frequency synthesizer is shown in *Figure 1*. The connection diagram is shown in *Figure 2*.

The 5 kHz reference frequency of the phase detector is generated by division of a 10.24 MHz oscillator frequency in the $2^{11}$ divider chain. This is compared with the output of the programmable divider until a match in phase is reached to attain a lock condition. The inputs to the programmable divider are the VCO frequency (mixed down) input at $f_{IN}$ and the 13-bit binary channel selection code which is the divisor of the $f_{IN}$ input to generate a corresponding 5 kHz signal for the channel frequency in question.

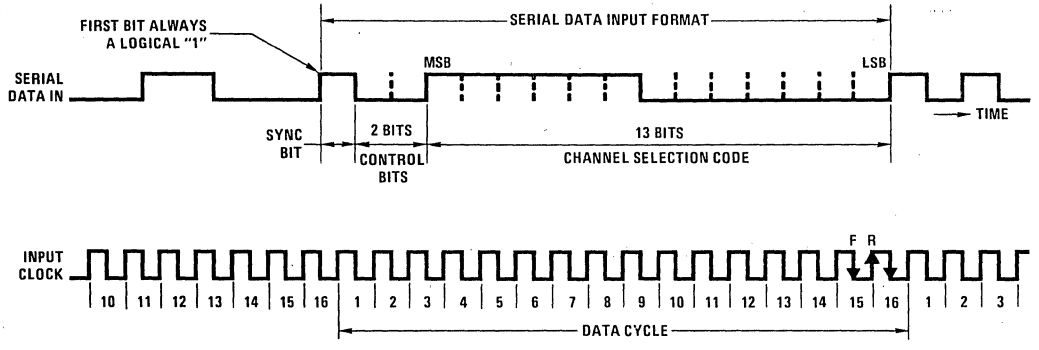
Format of serial data generated in a controller and fed into the SERIAL DATA IN pin is shown in *Figure 3*. The logic "1" synchronization bit, two control bits (that are latched and made available at separate pins) and the 13-bit channel selection code are derived from this data string. The chip select input, CS, enables serial data transfer when it is at ground potential.

Synchronization is maintained between the controller and the serial PLL by use of a logical handshake and by clearing all the information out of the internal serial-to-parallel shift register when the logical "1" signal is detected. The operation is described as follows (*Figure 3*).

Sixteen-bit data is shifted into the 15-bit shift register and "1" detector. The first bit is always a logical "1." After this is detected on the 15th clock pulse, data in the shift register will be transferred (at the rising edge of the 16th clock pulse) into the 15-bit latch. The falling edge of the 16th clock pulse disables transfer and resets all the shift register bits.




$t_{sd}$ = Delay time between active chip select command and initial data input.
$t_{dc}$ = Delay time between initial data input and active clock.




**SYNCHRONIZATION STEPS**

| Clock Edge | Operation |
|---|---|
| 15F | Detect Sync Bit |
| 16R | Transfer Data into Latch |
| 16F | Disable Transfer, Reset S/R and "1" Detector |
| 1R | Clock In "1" (Sync Bit) |
| 2R | Clock In Data |

**FIGURE 3**

## Typical Application

*Figure 4* shows an interconnect diagram for an MM55121 PLL/frequency synthesizer with a COP420L acting as a keyboard/display controller. The prescaler DS8626 enables the system to operate at a higher frequency range. The 60 Hz OUT provides a clock for timekeeping function while the 300 Hz OUT is usually used as an alarm tone.

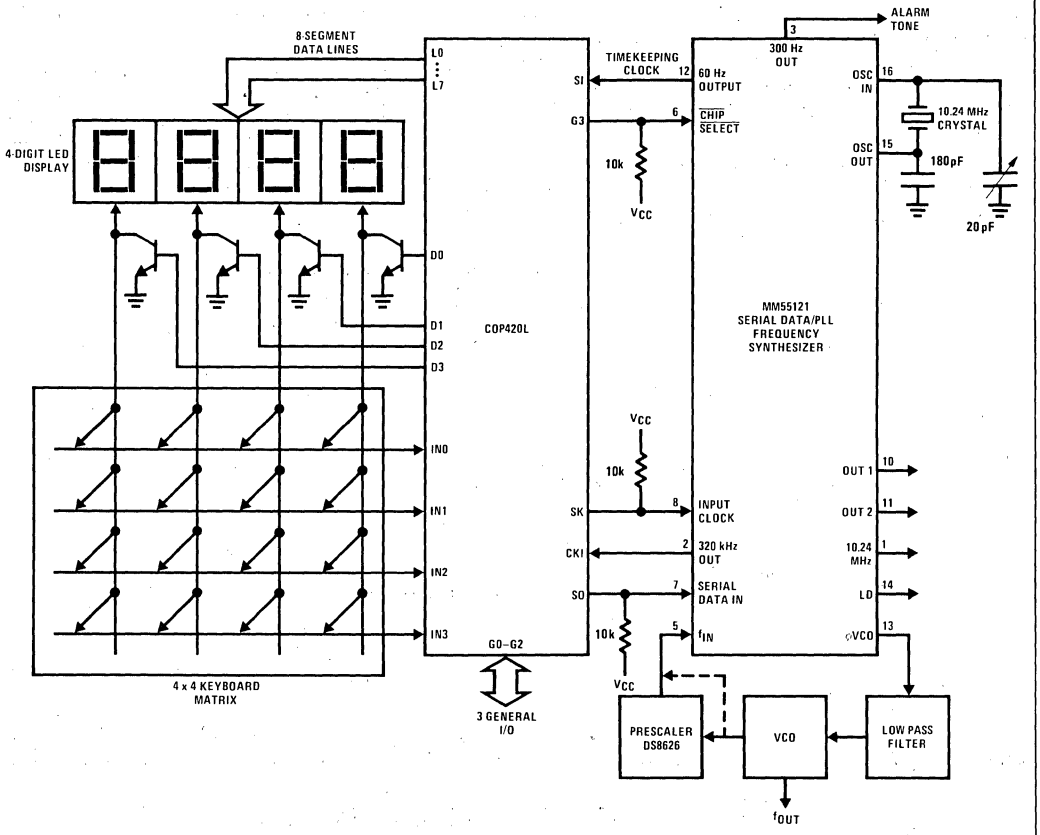



**FIGURE 4. MM55121, COP420L Frequency Synthesizer/Controller**

National Semiconductor

Television/Radio

# MM55122 Serial Data/PLL Frequency Synthesizer

## General Description

The MM55122 is a monolithic metal gate CMOS integrated circuit which contains a phase-locked loop circuit useful for frequency synthesizer application in CB transceivers. The device operates from a single power supply and contains an oscillator, a $2^{10}$ divider chain, a binary input programmable divider, and phase detector circuitry. Selection of a channel is accomplished by external programming of the programmable divider with a 9-bit serial code derived from a 26-bit data string fed to the data I/O pin. The serial data format consists of a leading logical "1" synchronization bit, three 4-bit data to generate analog outputs (such as squelch, volume, or A.V.C.), 4 control bits that are latched and made available at pins A–D, and a 9-bit binary input channel select code.

The MM55122 may be used in single or double IF systems. It uses a 10.24 MHz quartz crystal to determine the reference frequency. It has an output pin which provides a 5.12 MHz signal, which may be tripled for use as a reference oscillator frequency in 2-crystal systems.

The phase detector output, $\phi$VCO, provides a high level voltage (sources current) when the VCO frequency is lower than the lock frequency, and it provides a low level voltage (sinks current) when the VCO frequency is higher than lock frequency. The $\phi$VCO output goes to a high impedance (TRI-STATE®) condition and the lock detector output, LD, goes to a high state under lock conditions.

## Features

- Single crystal operation
- Single power supply
- Low power CMOS technology
- On-chip oscillator with feedback resistor
- Buffered 5.12 MHz output
- Serial data input format consisting of a leading "1" synchronization bit, three 4-bit data to generate analog outputs, 4 control bits available at separate pins, and a 9-bit binary input channel selection code
- $f_{IN}$ input amplifier stage to accept 1 Vp-p signal
- Programmable $2^9 - 1$ division of $f_{IN}$
- Relative transmit/receive signal strength comparison

## Block Diagram

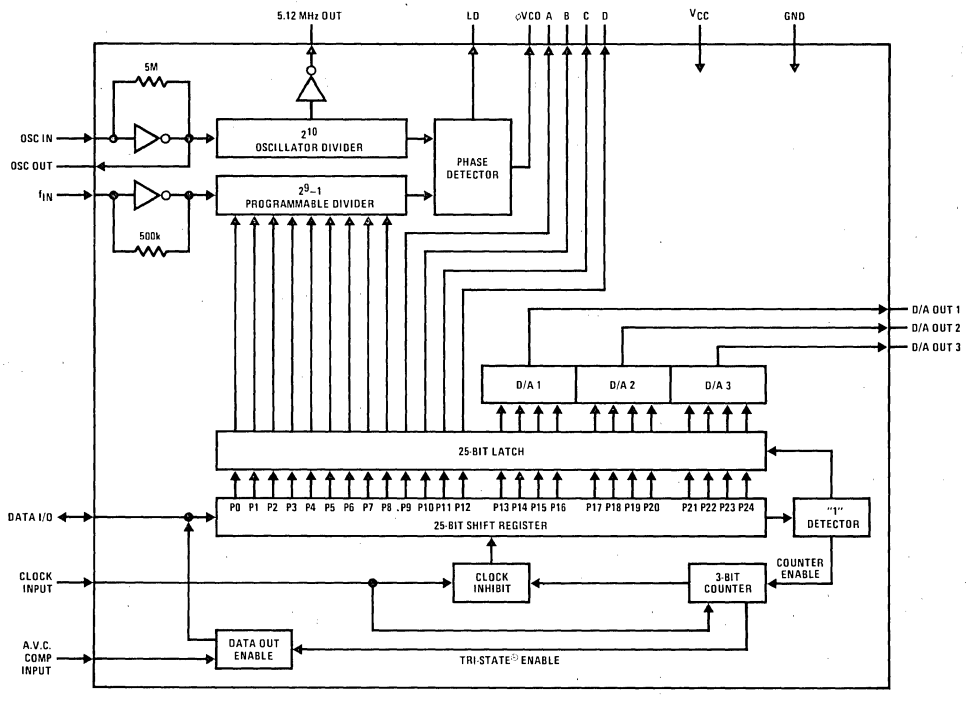



FIGURE 1

## Absolute Maximum Ratings

| | | | |
|---|---|---|---|
| Voltage at Any Pin | $V_{CC}$ + 0.3V to Gnd − 0.3V | Operating $V_{CC}$ | 10V |
| Operating Temperature Range | −30°C to +70°C | Lead Temperature (Soldering, 10 seconds) | 300°C |
| Storage Temperature Range | −40°C to +125°C | | |

## Electrical Characteristics

$T_A$ within operating temperature range, GND = 0V, unless otherwise specified

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Supply Voltage ($V_{CC}$) | | 7 | 8.5 | 10 | V |
| Supply Current ($I_{CC}$) | Frequency at Osc In = 10.24 MHz and $f_{IN}$ = 3 MHz, All Other Pins Open | | | | |
| | $V_{CC}$ = 7V | | 20 | 30 | mA |
| | $V_{CC}$ = 10V | | 30 | 45 | mA |
| Logical "1" Input Voltage ($V_{IN(1)}$) Data I/O, Clock Input, A.V.C. Comp. Input | | $V_{CC}$−0.5 | | | V |
| Logical "0" Input Voltage ($V_{IN(0)}$) Data I/O, Clock Input, A.V.C. Comp. Input | | | | 0.5 | V |
| Logical "1" Output Voltage ($V_{OUT(1)}$) LD, $\phi$VCO, Osc Out, 5.12 MHz Out | $I_{OUT}$ = −1 mA | $V_{CC}$−1.0 | | | V |
| Logical "0" Output Voltage ($V_{OUT(0)}$) LD, $\phi$VCO, Osc Out, 5.12 MHz Out | $I_{OUT}$ = 1 mA | | | 1.0 | V |
| Logical "1" Output Voltage ($V_{OUT(1)}$) A, B, C, D | $I_{OUT}$ = −0.5 mA | $V_{CC}$−0.5 | | | V |
| Logical "0" Output Voltage ($V_{OUT(0)}$) A, B, C, D | $I_{OUT}$ = 0.5 mA | | | 0.5 | V |
| Analog Output Resistance D/A Out 1, 2, 3 | | 21 | 30 | 39 | kΩ |
| Analog Output Voltage D/A Out 1, 2, 3 | Full-Scale | $V_{CC}-\frac{V_{CC}}{30}$ | | $V_{CC}$ | V |
| | LSB ON | $\frac{V_{CC}}{15}-\frac{V_{CC}}{30}$ | | $\frac{V_{CC}}{15}+\frac{V_{CC}}{30}$ | V |
| | Zero-Scale | 0 | | $\frac{V_{CC}}{30}$ | V |
| Logical "1" Input Current ($I_{IN(1)}$) | | | | | |
| Input Clock (Pull-Down) | $V_{CC}$ = 7V, $V_{IN}$ = 6.5V | 5 | | 60 | μA |
| | $V_{CC}$ = 10V, $V_{IN}$ = 9.5V | 20 | | 200 | μA |
| Data I/O (Pull-Down) | $V_{CC}$ = 7V, $V_{IN}$ = 6.5V | 5 | | 60 | μA |
| | $V_{CC}$ = 10V, $V_{IN}$ = 9.5V | 20 | | 200 | μA |
| Maximum Toggle Frequency at $f_{IN}$ | | 4.0 | | | MHz |
| Input Signal at $f_{IN}$ | Small Signal (AC Coupled) or | 1.0 | | | Vp-p |
| | $V_{IN(1)}$ | $V_{CC}$−0.5 | | | V |
| | $V_{IN(0)}$ | | | 0.5 | V |

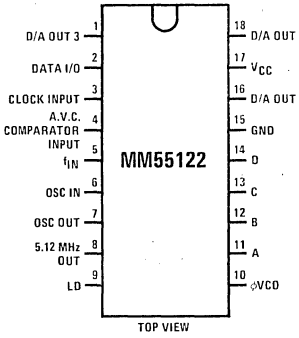
## Electrical Characteristics (Continued)

$T_A$ within operating temperature range, Gnd = 0V, unless otherwise specified

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Duty Cycle at $f_{IN}$ | Frequency = 4 MHz | 30 | | 70 | % |
| $f_{IN}$ Amplifier Feedback Resistor | | | 500 | | kΩ |
| Oscillator Frequency at Osc In | 10.24 MHz Crystal | 10.24 | | | MHz |
| Osc In Feedback Resistor | | | 5 | | MΩ |
| Maximum Toggle Frequency at Input Clock | | 500 | | | kHz |
| TRI-STATE Leakage at $\phi$VCO | $V_{OUT}$ = $V_{CC}$ or Gnd | | | ±1 | μA |

## Connection Diagram

**Dual-In-Line Package**




TOP VIEW

**Order Number MM55122N**
**See Package 20**

**FIGURE 2**

### PIN DESCRIPTION

| | |
|---|---|
| $f_{IN}$ | Frequency input from VCO (mixed down) |
| OSC IN | Oscillator amplifier input |
| OSC OUT | Oscillator amplifier output |
| LD | Lock detector output: "1" for locked condition, "0" for unlocked condition |
| $\phi$VCO | Phase detector output for VCO control |
| 5.12 MHz OUT | Buffered 5.12 MHz output (oscillator frequency ÷2) |
| Data I/O | Serial data input/output |
| A.V.C. Comparator Input | Automatic volume control comparator input |
| A, B, C, D | Latched outputs derived from serial data input |
| D/A OUT 1, 2, 3 | Analog outputs formed in increments of $V_{CC}$/15 per LSB. |
| Clock Input | Internal shift register clock |

## Functional Description

A block diagram of the MM55122 Serial Data/PLL Frequency Synthesizer is shown in *Figure 1*, while a connection diagram is shown in *Figure 2*.

The 10 kHz reference frequency of the phase detector is generated by division of a 10.24 MHz oscillator frequency in the $2^{10}$ divider chain. This is compared with the output of the programmable divider until a match in phase is reached to attain a lock condition. The inputs to the programmable divider are the VCO frequency (mixed down) input at $f_{IN}$ and the 9-bit binary channel selection code which is the divisor of the $f_{IN}$ input to generate a corresponding 10 kHz signal for the channel frequency in question.

Format of serial data generated in a controller and fed into the Data I/O pin is shown in *Figure 3*. From this data string, the logic "1" synchronization bit, three 4-bit data that are used to generate analog outputs, 4 control bits that are latched and made available at separate pins and 9-bit channel selection code are derived.

Synchronization is maintained between the controller-oriented processor and the MM55122 by use of a logical "1" handshake and by clearing all of the information out of the shift register when the logical "1" signal is detected. The chip operation is described as follows *(Figure 3)*:

Data is shifted into the 25-bit shift register by the clock input.

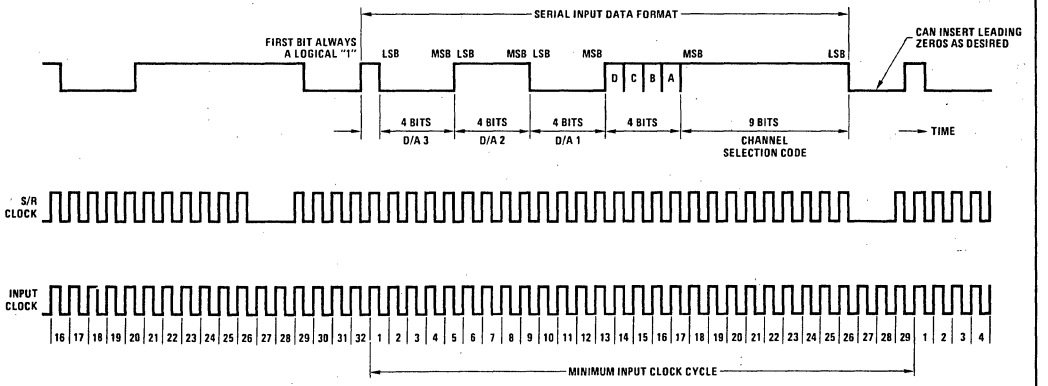
The first data bit will always be a logical "1". After this "1" is detected by the "1" detector on the 25th clock pulse, the data in the S/R will be transferred (at the rising edge of the 26th clock pulse) to the 25-bit latch. The falling edge of the 26th clock pulse disables the transfer, enables the TRI-STATE Data I/O for output mode, resets all of the S/R bits, and inhibits the clock internally to the shift register.

The rising edge of the 27th clock pulse disables the reset and will be used in the controller oriented processor to strobe the "1" data which will be present on the Data I/O pin.

The falling edge of the 27th clock pulse removes the "1" from the Data I/O pin and enables the A.V.C. comparator data. This data will be strobed into the controller by the rising edge of the 28th clock pulse. The falling edge of the 28th clock pulse disables the A.V.C. data output. The rising edge of the 29th clock pulse enables the Data I/O as an input and the next clock pulse clocks data into the S/R.

## Functional Description (Continued)

The A.V.C. comparator input is an additional feature that enables the user to display relative transmit and receive signal strengths. Actual signal strengths are dynamically compared with a weighted output from the controller and made available at one of the D/A outputs until a match is reached. Appropriate numbers can then be displayed to indicate incoming signal strength when in receive mode and relative power output when in transmit mode.




**SYNCHRONIZATION STEPS:**

| CLOCK EDGE | OPERATION | CLOCK EDGE | OPERATION |
|---|---|---|---|
| 26R* | Transfer data into latch | 28R | Strobe Data I/O A.V.C. level into controller |
| 26F | Reset S/R, enable TRI-STATE Data I/O for output mode, inhibit clock | 28F | Remove A.V.C. data level from Data I/O line |
| 27R | Strobe Data I/O "1" level into controller | 29R | Disable clock inhibit |
| 27F | Enable A.V.C. comparator on Data I/O | 1R | Clock in "1" (sync bit) |
| | | 2R | Clock in data |

*R = rising edge
F = falling edge

FIGURE 3

## Typical Applications

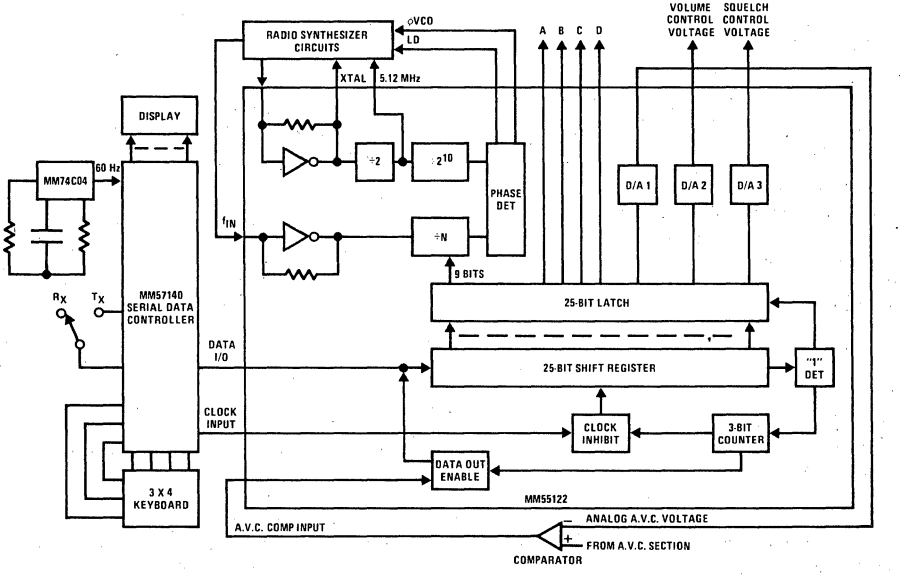
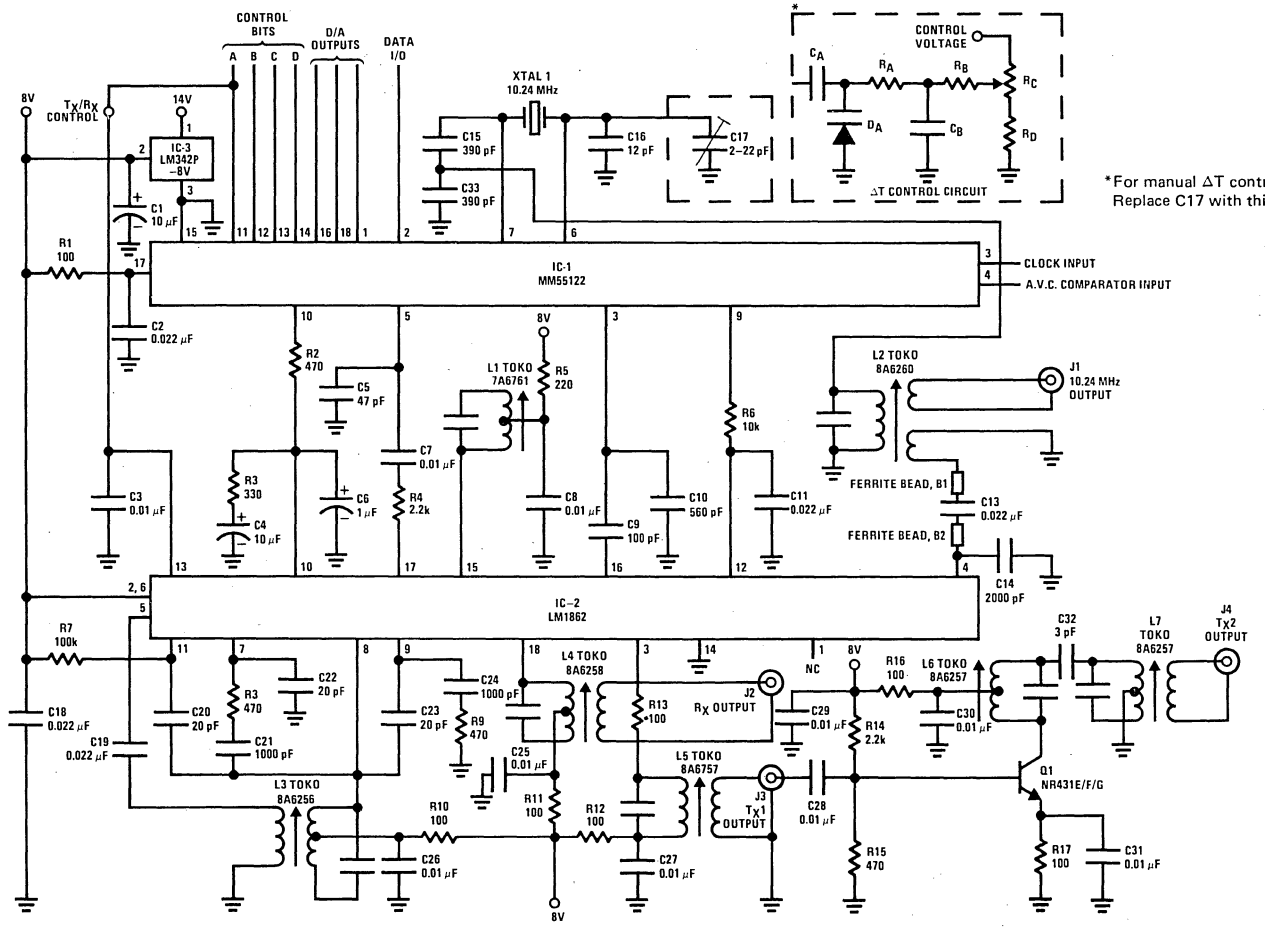



FIGURE 4. Remote Microphone Serial Data System




All resistors shown in ohms, ±5% tolerance. Capacitors in μF and pF, 80%–20% tolerance for μF, ±10% for pF.
Tuning capacitors parallel all transformers are built-in type with the coils. Details referred to Toko published specifications.

**FIGURE 5. LM1862, MM55122 Single Crystal 40-Channel PLL Schematic Diagram**

National Semiconductor

# MM55123 Serial Data/PLL Frequency Synthesizer

## General Description

The MM55123 device is a monolithic metal gate CMOS integrated circuit which contains a phase locked loop circuit useful for frequency synthesizer applications in the AM, FM, CB and SW frequency bands. It operates from a single power supply and contains an oscillator with a feedback resistor, a $2^{11}$ divider chain, a binary input programmable divider, and phase detector circuitry. Selection of a channel is accomplished by external programming of the programmable divider with a 13-bit serial code derived from a 16-bit data string fed to the SERIAL DATA IN. The serial data format consists of a leading logical "1" synchronization bit, two control bits that are latched and made available at OUT 1 and OUT 2 and a 13-bit binary input channel selection code.

The phase detector output, $\phi$VCO, provides a high level voltage (sources current) when the VCO frequency is lower than the lock frequency, and it provides a low level voltage (sinks current) when the VCO frequency is higher than the lock frequency. The $\phi$VCO output goes to a high impedance (TRI-STATE®) condition and the lock detector output LD goes to a high state under lock conditions.

## Features

- Single crystal operation
- Single power supply
- Low power CMOS technology
- On-chip oscillator with feedback resistor
- Buffered outputs: 10.24 MHz, 320 kHz, 300 Hz and 60 Hz
- Serial data input format consisting of a leading "1" synchronization bit, 2 control bits available at separate pins, and a 13-bit channel selection code
- $f_{IN}$ input amplifier stage to accept an AC coupled 0.8 Vp-p signal

## Block Diagram

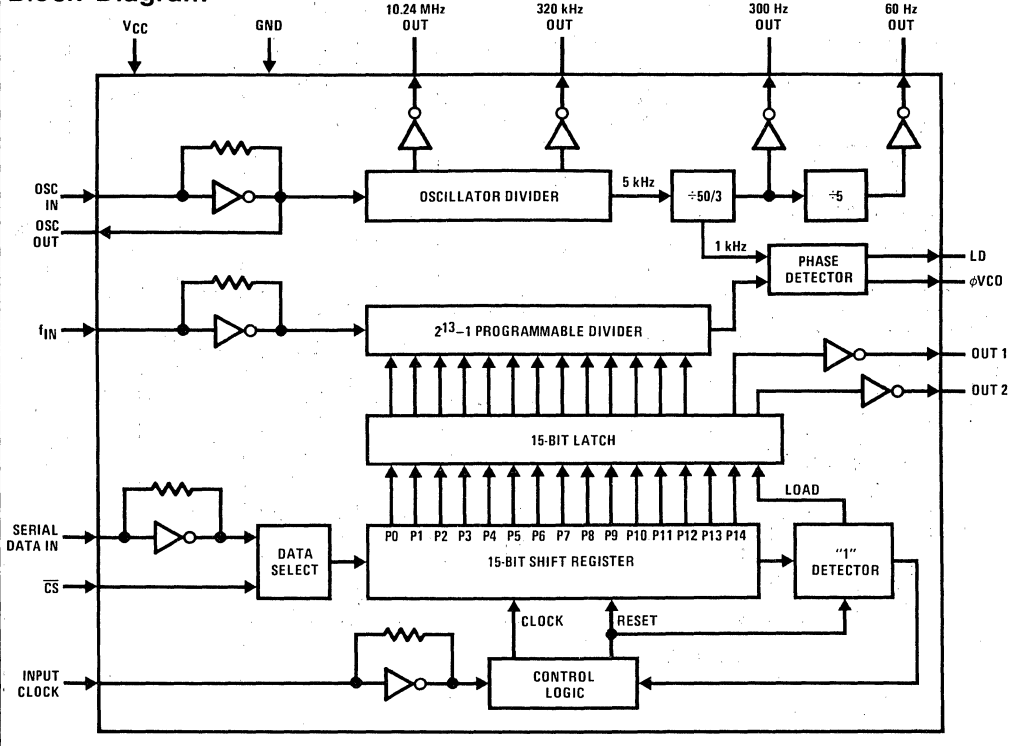



FIGURE 1

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{CC}$ + 0.3V to Gnd − 0.3V |
| Operating Temperature Range | −30°C to +75°C |
| Storage Temperature Range | −40°C to +125°C |
| Operating Supply Voltage, $V_{CC}$ | 12V |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

Ambient temperature within the operating range and Gnd = 0V, unless otherwise specified.

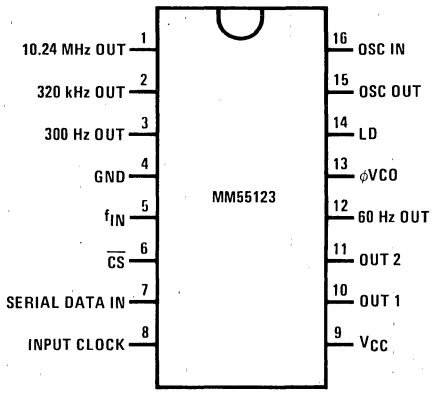
| Parameter | | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Supply Voltage ( $V_{CC}$) | | | 6.5 | | 10.0 | V |
| Supply Current ($I_{CC}$) | | Osc Frequency = 10.24 MHz<br>Prog. Divider Input ($f_{IN}$) Frequency = 4.5 MHz<br>$\overline{CS}$, Serial Data In, and Input Clock at Gnd; All Other Pins Open | | | | |
| | | $V_{CC}$ = 6.5V | | 20 | 30 | mA |
| | | $V_{CC}$ = 10.0V | | 30 | 45 | mA |
| LOGIC: | Logical "1" Input Voltage ($V_{IN(1)}$)<br>INPUT CLOCK<br>SERIAL DATA IN<br>$\overline{CS}$ | | 0.8 $V_{CC}$ | | | V |
| | Logical "0" Input Voltage ($V_{IN(0)}$)<br>INPUT CLOCK<br>SERIAL DATA IN<br>$\overline{CS}$ | | | | 0.2 $V_{CC}$ | V |
| | Logical "1" Output Voltage ($V_{OUT(1)}$)<br>OSC OUT<br>10.24 MHz OUT<br>LD<br>$\phi$VCO | $I_{OUT}$ = −0.7 mA | 0.8 $V_{CC}$ | | | V |
| | Logical "0" Output Voltage ($V_{OUT(0)}$)<br>OSC OUT<br>10.24 MHz OUT<br>LD<br>$\phi$VCO | $I_{OUT}$ = 0.7 mA | | | 0.2 $V_{CC}$ | V |
| | Logical "1" Output Voltage ($V_{OUT(1)}$)<br>320 kHz OUT<br>300 Hz OUT<br>60 Hz OUT | $I_{OUT}$ = −0.5 mA | 0.8 $V_{CC}$ | | | V |
| | Logical "0" Output Voltage ($V_{OUT(0)}$)<br>320 kHz OUT<br>300 Hz OUT<br>60 Hz OUT | $I_{OUT}$ = 0.5 mA | | | 0.2 $V_{CC}$ | V |

## Electrical Characteristics (Continued) Ambient temperature within the operating range and Gnd = 0V, unless otherwise specified.

| Parameter | | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $f_{IN}$: | Toggle Frequency | | 100k | | 4.5M | Hz |
| | Input Signal | Small Signal (AC-Coupled) or | 0.8 | | | Vp-p |
| | | $V_{IN(1)}$ | 0.8 $V_{CC}$ | | | V |
| | | $V_{IN(0)}$ | | | 0.2 $V_{CC}$ | V |
| | Duty Cycle at $f_{IN}$ | | 30 | | 70 | % |
| $f_{IN}$, SERIAL DATA IN, INPUT CLOCK: | Input Capacitance | | | 5 | 10 | pF |
| | Input Resistance Including Feedback Resistor | | 100 | | | kΩ |
| OSC IN: | Oscillator Frequency | | 10.24 | | | MHz |
| | Oscillator Input Resistance Including Feedback Resistor | | 1 | | | MΩ |
| $\overline{CS}$: | Input Capacitance | | | 5 | 10 | pF |
| | Input Resistance | | 1 | | | MΩ |
| INPUT CLOCK: | Toggle Frequency | | DC | | 500 | kHz |
| | Clock Rise Time and Fall Time | | 0.5 | | | μs |
| ϕVCO: | TRI-STATE Leakage at $V_{OUT} = V_{CC}$ or Gnd | | −0.1 | | +0.1 | μA |
| DELAYS: | Select to Data Time Delay, $t_{sd}$ | | | | 0 | ns |
| | Data to Clock Time Delay, $t_{dc}$ | | | | 100 | ns |

## Connection Diagram

**Dual-In-Line Package**




**TOP VIEW**
**Order Number MM55123N**
**See Package 19**

**FIGURE 2**

**Pin Descriptions**

| | |
|---|---|
| $f_{IN}$ | Frequency Input from VCO (mixed down) |
| OSC IN | Oscillator Amplifier Input |
| OSC OUT | Oscillator Amplifier Output |
| LD | Lock Detector Output: "1" for locked condition; "0" for unlocked condition |
| ϕVCO | Output of Phase Detector for VCO Control |
| 10.24 MHz OUT | Buffered 10.24 MHz Output (Oscillator Frequency) |
| 320 kHz OUT | Buffered 320 kHz External Clock Output |
| 300 Hz OUT | Buffered 300 Hz Output |
| 60 Hz OUT | Buffered 60 Hz Output |
| SERIAL DATA IN | Serial Data Input |
| INPUT CLOCK | Internal Shift Register Clock |
| $\overline{CS}$ | Chip Select Input |
| OUT 1-2 | Control Outputs |

## Functional Description

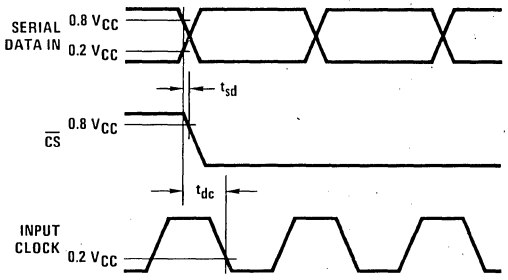
A block diagram of the MM55123 serial data/PLL frequency synthesizer is shown in *Figure 1*. The connection diagram is shown in *Figure 2*.

The 1 kHz reference frequency of the phase detector is generated by division of a 10.24 MHz oscillator frequency in the $2^{11}$ divider chain. This is compared with the output of the programmable divider until a match in phase is reached to attain a lock condition. The inputs to the programmable divider are the VCO frequency (mixed down) input at $f_{IN}$ and the 13-bit binary channel selection code which is the divisor of the $f_{IN}$ input to generate a corresponding 1 kHz signal for the channel frequency in question.

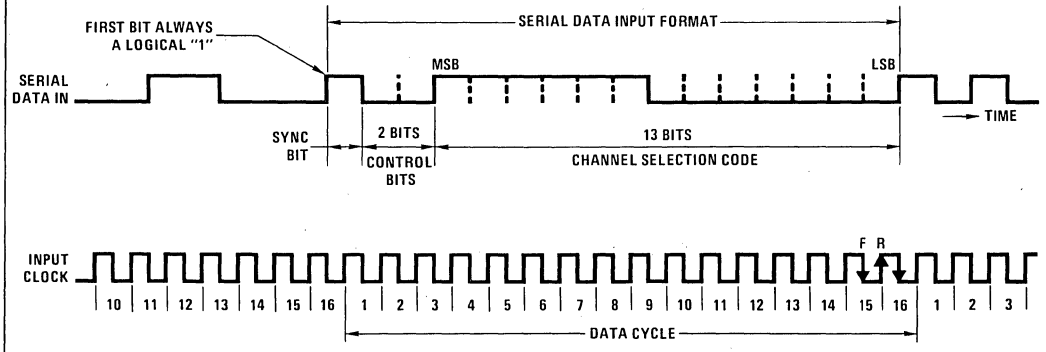
Format of serial data generated in a controller and fed into the SERIAL DATA IN pin is shown in *Figure 3*. The logic "1" synchronization bit, two control bits (that are latched and made available at separate pins) and the 13-bit channel selection code are derived from this data string. The chip select input, CS, enables serial data transfer when it is at ground potential.

Synchronization is maintained between the controller and the serial PLL by use of a logical handshake and by clearing all the information out of the internal serial-to-parallel shift register when the logical "1" signal is detected. The operation is described as follows (*Figure 3*).

Sixteen-bit data is shifted into the 15-bit shift register and "1" detector. The first bit is always a logical "1." After this is detected on the 15th clock pulse, data in the shift register will be transferred (at the rising edge of the 16th clock pulse) into the 15-bit latch. The falling edge of the 16th clock pulse disables transfer and resets all the shift register bits.




$t_{sd}$ = Delay time between active chip select command and initial data input.
$t_{dc}$ = Delay time between initial data input and active clock.




**SYNCHRONIZATION STEPS**

| Clock Edge | Operation |
|---|---|
| 15F | Detect Sync Bit |
| 16R | Transfer Data into Latch |
| 16F | Disable Transfer, Reset S/R and "1" Detector |
| 1R | Clock In "1" (Sync Bit) |
| 2R | Clock In Data |

**FIGURE 3**

## Typical Application

*Figure 4* shows an interconnect diagram for an MM55123 PLL/frequency synthesizer with a COP420L acting as a keyboard/display controller. The prescaler DS8626 enables the system to operate at a higher frequency range. The 60 Hz OUT provides a clock for timekeeping function while the 300 Hz OUT is usually used as an alarm tone.

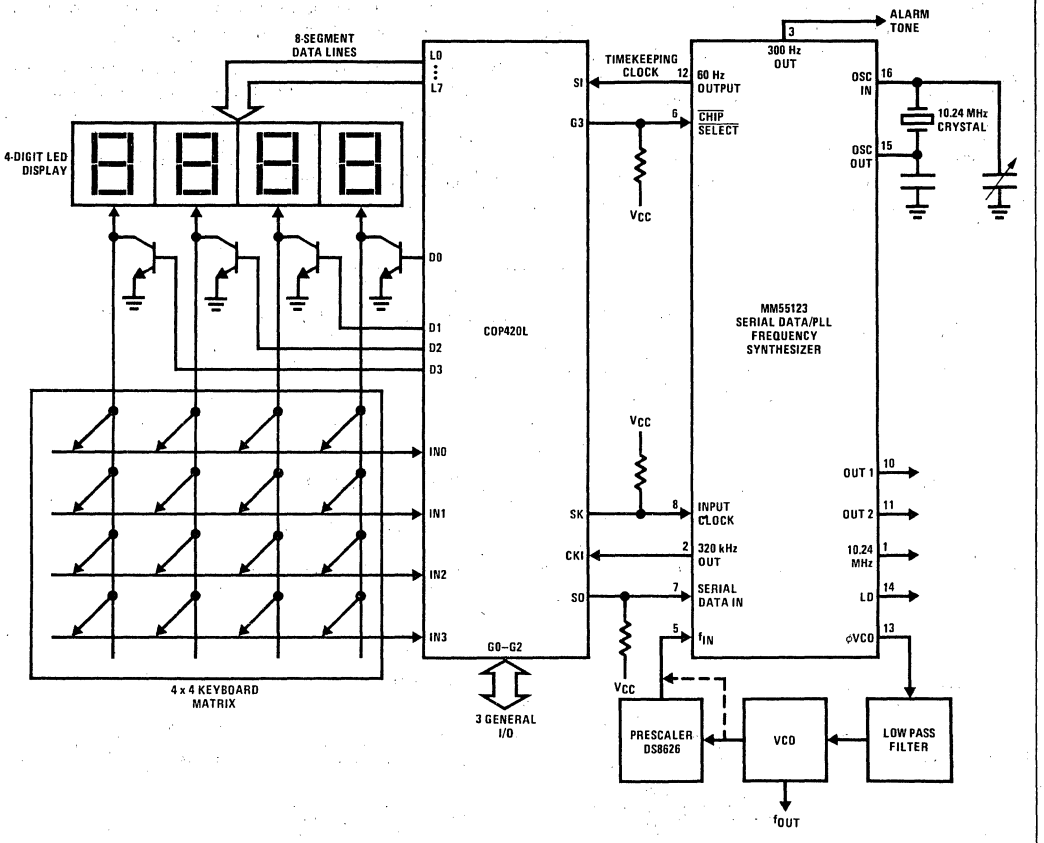



**FIGURE 4. MM55123, COP420L Frequency Synthesizer/Controller**

National Semiconductor

Television/Radio

# MM55124, MM55126 PLL frequency synthesizer

## general description

The MM55124 and MM55126 are phase locked loop circuits for frequency synthesizer applications fabricated with low threshold, metal gate CMOS technology. The MM55124 and MM55126 operate from a single power supply over a voltage range of 4.5–10V. An input amplifier has been included on the $F_{IN}$ input, thereby allowing operation of the programmable divider with as little as a 0.5V peak-to-peak input signal. A feedback resistor is also included on both circuits to bias the reference oscillator.

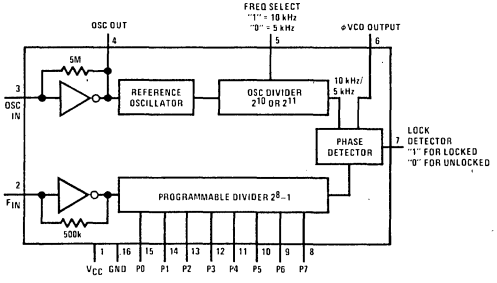
The MM55124 and MM55126 may be used in double IF or single IF systems. The devices use a 10.24 MHz or 5.12 MHz quartz crystal to determine the reference frequency. The MM55126 has a buffered 5.12 MHz output signal which may be tripled to generate a reference frequency in 1 or 2 crystal systems. Also, the MM55126 provides an additional input to the programmable divider which allows $2^9-1$ division of the input frequency ($F_{IN}$). The inputs to the programmable divider are standard binary signals. Selection of a channel is accomplished by mechanical switches or by external electronic programming of the programmable divider.

The $\phi$VCO output provides a high level voltage (sources current) when the VCO frequency is lower than the lock frequency, and a low level voltage (sinks current) when the VCO frequency is higher than the lock frequency. The $\phi$VCO output goes to a high impedance state (TRI-STATE®) and the lock detector output LD goes to a high state while the device is in lock mode.

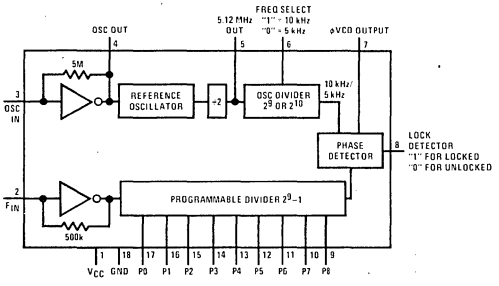
## features

- Input amplifier on $F_{IN}$
- Single power supply
- 4.5–10V voltage operation
- Low power CMOS technology
- Binary input channel select code
- 5 kHz or 10 kHz output from oscillator divide
- Buffered 5.12 MHz output (MM55126 only)
- On-chip oscillator with bias resistor
- Pull-down resistors on programmable divider inputs

## block diagrams




MM55124




MM55126

## pin descriptions

| | |
|---|---|
| P0–P8 | Programmable divider inputs |
| $F_{IN}$ | Frequency input from VCO (mixed down) |
| OSC IN | Oscillator amplifier input terminal |
| OSC OUT | Oscillator amplifier output terminal |
| LD | Lock detector |
| $\phi$VCO | Output of phase detector for control of the VCO |
| FS | Frequency division select 10 kHz or 5 kHz – "1" is 10 kHz; "0" is 5 kHz |
| 5.12 MHz OUT | OSC Frequency divided by 2 buffered output |

## truth table

**Truth table for binary inputs to programmable divider.**

| N | P8 | P7 | P6 | P5 | P4 | P3 | P2 | P1 | P0 |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | X |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| | . | | | | | | | | |
| | . | | | | | | | | |
| 511 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

$F_{OUT} = F_{IN}/N$

1 = High voltage level, $V_{OH}$

0 = Low voltage level, $V_{OL}$

X = Don't care

5

## absolute maximum ratings

| | | | |
|---|---|---|---|
| Voltage at Any Pin | $V_{CC}$ + 0.3V to Gnd − 0.3V | $V_{CC}$ Max | 12V |
| Operating Temperature Range | −30°C to +75°C | Lead Temperature (Soldering, 10 seconds) | 300°C |
| Storage Temperature Range | −40°C to +125°C | | |

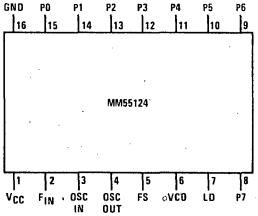
## electrical characteristics

$T_A$ within operating temperature range, Gnd = 0V unless otherwise specified

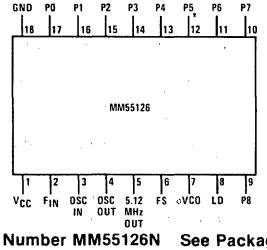
| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Supply Voltage ($V_{CC}$) | | 4.5 | | 10.0 | V |
| Supply Current ($I_{CC}$) | Freq at Osc. In = 10.24 MHz, $F_{IN}$ = 2.5 MHz, All Other I/O Pins Open | | | | |
| | $V_{CC}$ = 4.5V | | 8 | 12 | mA |
| | $V_{CC}$ = 10V | | 30 | 45 | mA |
| Logical "1" Input Voltage ($V_{IN(1)}$) | | | | | |
| P0–P8, FS | | $V_{CC}$−0.5 | | | V |
| Logical "0" Input Voltage ($V_{IN(0)}$) | | | | | |
| P0–P8, FS | | | | 0.5 | V |
| Logical "1" Output Voltage | | | | | |
| 5.12 MHz Out, LD | $I_O$ = −0.5 mA | $V_{CC}$−0.5 | | | V |
| $\phi$VCO | $I_O$ = −0.4 mA | $V_{CC}$−0.5 | | | V |
| Osc. Out | $I_O$ = −0.25 mA | $V_{CC}$−0.5 | | | V |
| Logical "0" Output Voltage | | | | | |
| 5.12 MHz Out, LD | $I_O$ = 0.5 mA | | | 0.5 | V |
| $\phi$VCO | $I_O$ = 0.4 mA | | | 0.5 | V |
| Osc. Out | $I_O$ = 0.25 mA | | | 0.5 | V |
| Logical "1" Input Current | | | | | |
| FS (Pull-Up) | $V_{CC}$ = 4.5V, $V_{IN}$ = 4V | −5 | | −0.5 | μA |
| | $V_{CC}$ = 10V, $V_{IN}$ = 9.5V | −15 | | −1 | μA |
| P0–P8 (Pull-Down) | $V_{CC}$ = 4.5V, $V_{IN}$ = 4V | 4 | | 40 | μA |
| | $V_{CC}$ = 10V, $V_{IN}$ = 9.5V | 20 | | 200 | μA |
| Logical "0" Input Current | | | | | |
| FS (Pull-Up) | $V_{CC}$ = 4.5V, $V_{IN}$ = 0.5V | −20 | | −2 | μA |
| | $V_{CC}$ = 10V, $V_{IN}$ = 0.5V | −100 | | −10 | μA |
| P0–P8 (Pull-Down) | $V_{CC}$ = 4.5V, $V_{IN}$ = 0.5V | 1 | | 10 | μA |
| | $V_{CC}$ = 10V, $V_{IN}$ = 0.5V | 2 | | 30 | μA |
| Input Leakage | | | | | |
| FS (Pull-Up) | $V_{IN}$ = $V_{CC}$ | | | 500 | nA |
| P0–P8 (Pull-Down) | $V_{IN}$ = Gnd | | | \|−500\| | nA |
| Maximum Toggle Frequency at $F_{IN}$ | | 3.0 | | | MHz |
| Input Signal at $F_{IN}$ (Max 3.0 MHz) | For AC Signal or | 0.5 | | | Vp-p |
| | $V_{IN(1)}$ | $V_{CC}$−0.5 | | | V |
| | $V_{IN(0)}$ | | | 0.5 | V |
| $F_{IN}$ Duty Cycle | $F_{IN}$ = 3 MHz, 0.5 Vp-p (AC Coupled) | 30 | | 70 | % |
| Maximum Osc. Frequency At Osc. In | 10.24 MHz Quartz Crystal $V_{CC}$ = 5V | 10.24 | | | MHz |
| TRI-STATE® Leakage at $\phi$VCO | $V_{OUT}$ = $V_{CC}$ or Gnd | | | \|±1\| | μA |

## connection diagrams

(Dual-In-Line Packages, Top View)




**Order Number MM55124N**
**See Package 19**




**Order Number MM55126N** **See Package 20**

## typical applications

### INTRODUCTION TO FREQUENCY SYNTHESIS

The components of a frequency synthesizer are shown in *Figure 1*. The voltage controlled oscillator produces the desired output frequencies spaced $f_V$ Hz apart according to the relation:

$$f_V = f_r N$$

The reference frequency, $f_r$, must be equal to or less than the (channel) spacing between the frequencies being synthesized.

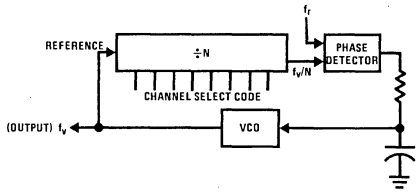



**FIGURE 1. Basic Frequency Synthesizer**

Although simple in concept, the circuit of *Figure 1* has certain difficulties. In CB, we are synthesizing the following frequencies:

| | |
|---|---|
| Ch. 1 | 26.965 MHz |
| Ch. 2 | 26.975 MHz |
| . | . |
| . | . |
| . | . |
| Ch. 40 | 27.405 MHz |

Although the channel spacing is 10 kHz, a reference frequency of 5 kHz would be necessary due to the odd 5 kHz in the assigned channel. This in itself poses no problem; however, present technology limits the counting speed of programmable dividers to something less than 5 MHz, ruling out the approach shown in *Figure 1*.

Two solutions to this problem are shown in *Figure 2*.

Frequency prescaling shown in *Figure 2a* reduces the VCO frequency by M (a fixed number) to a frequency that can be divided by the programmable counter. The reference frequency $f_r$ must also be reduced by M. In the case of CB, if M = 10, $f_V$ = 26.965 MHz, the input to the programmable divider will be 2.6965 MHz, and the 5 kHz reference frequency will be reduced to 500 Hz. This poses problems in speed of response of the phase locked loop.

The second technique mixes the output frequency of the VCO with a stable fixed frequency to obtain a related reference frequency.

$$f_V = Nf_r + f_o$$

This technique has the advantage of allowing a 10 kHz reference frequency in the loop instead of 5 kHz.

Further complexity arises when one considers that the synthesizer must also generate a local oscillator signal as well as a transmitter input signal for the radio *(Figure 3)*. A system which provides these frequencies, as well as the proper offset to allow the programmable divider to operate within its limits is shown in the typical applications diagrams *(Figure 4)*. The only departure from the ideal situation shown in *Figure 3* is that the first IF frequency of 10.7 MHz must be changed to 10.695 MHz (a change of 5 kHz).

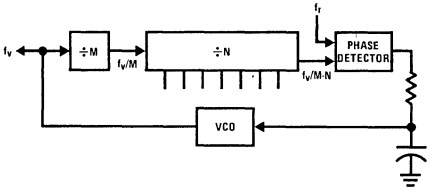



**FIGURE 2a. Frequency Prescaling**

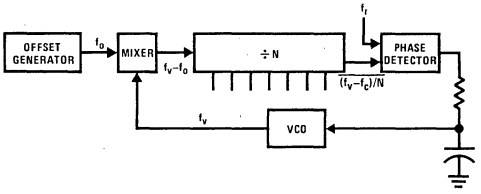



**FIGURE 2b. Frequency Offset**

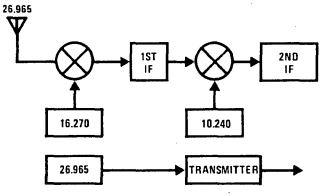



**FIGURE 3. Signals Needed to Transmit and Receive Ch. 1**

## typical applications (Continued)

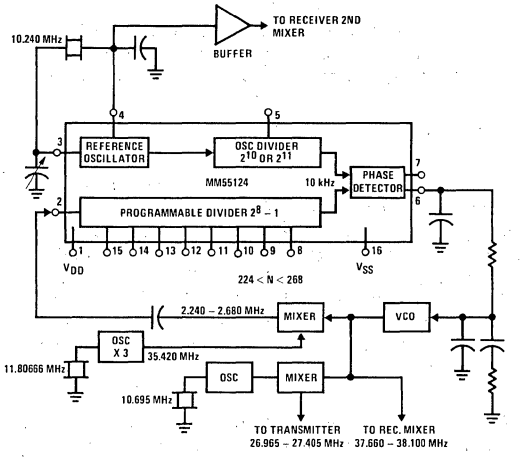



FIGURE 4a. MM55124 3-Crystal Application

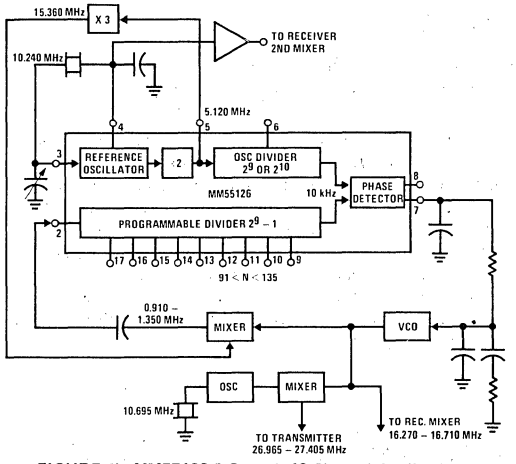



FIGURE 4b. MM55126 2-Crystal, 40-Channel Application

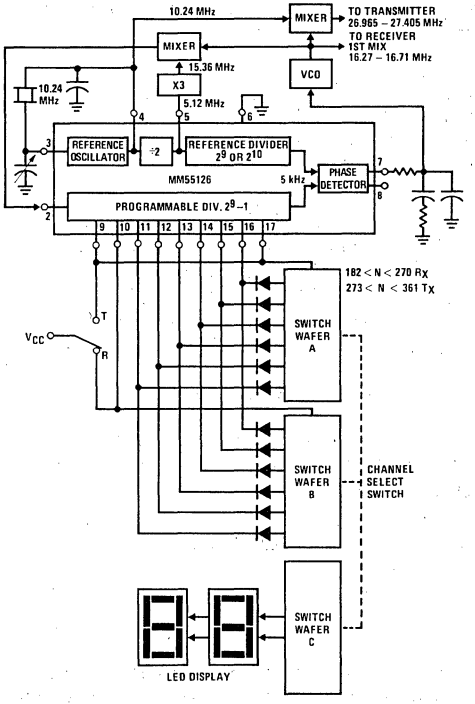



FIGURE 4c. MM55126 Single Crystal, Low Side Injection 40-Channel Application

National Semiconductor

# MM5837 Digital Noise Source

## General Description

The MM5837 digital noise source is an MOS pseudo-random sequence generator, designed to produce a broadband white noise signal for audio applications. Unlike traditional semiconductor junction noise sources, the MM5837 provides very uniform noise quality and output amplitude. The shift register starts at a random non-zero state when power is applied. The circuit is packaged in an 8-lead plastic DIP.

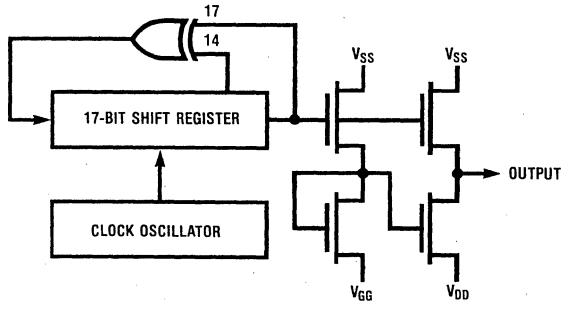
## Features

- Uniform noise quality
- Uniform noise amplitude
- Eliminates noise preamps
- Self-contained oscillator
- Single component insertion

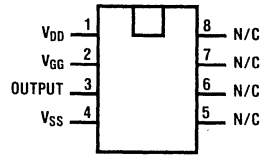
## Applications

- Electronic musical rhythm instrument sound generators
- Music synthesizer white and pink noise generators
- Room acoustics testing/equalization

## Logic and Connection Diagrams




Dual-In-Line Package




TOP VIEW

Order Number MM5837N
See Package 17

## Absolute Maximum Ratings

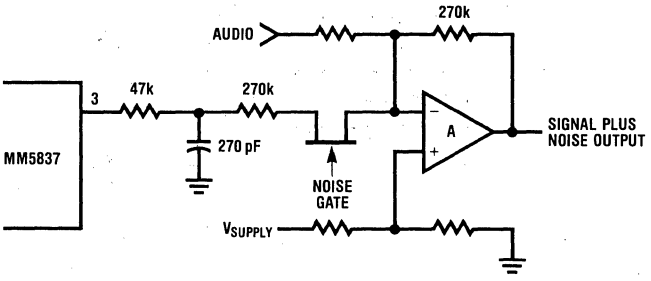
| | |
|---|---|
| Optional Gate Supply Voltage, $V_{GG}$ | $V_{SS} - 33V$ to $V_{SS} + 0.3V$ |
| Logic Supply Voltage, $V_{DD}$ | $V_{SS} - 25V$ to $V_{SS} + 0.3V$ |
| Storage Temperature, $T_S$ | $-55°C$ to $+100°C$ |
| Operating Temperature, $T_A$ | $0°C$ to $+70°C$ |
| Lead Temperature (Soldering, 10 seconds) | $300°C$ |

## Electrical Characteristics

$T_A$ within operating range, $V_{SS} = 0V$, $V_{DD} = -14V \pm 1.0V$, $V_{GG} = -27V, \pm 2V$ unless otherwise noted

| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| Output (Loaded 20 kΩ to $V_{SS}$ and 20 kΩ to $V_{DD}$) | $T_A = 25°C$ | | | | |
| Logical "1" Level | | $V_{SS} - 1.5$ | | $V_{SS}$ | V |
| Logical "0" Level | | $V_{DD}$ | | $V_{DD} + 1.5$ | V |
| Logical "0" Level | $V_{GG} = -14V \pm 1V$ | $V_{DD}$ | | $V_{DD} + 3.5$ | V |
| Supply Currents | | | | | |
| $I_{DD}$ | No Output Load | 3 | | 8 | mA |
| $I_{GG}$ | | | | 7 | mA |
| Half Power Point | | 24 | | 56 | kHz |
| Cycle Time | | 1.1 | | 2.4 | Sec. |

## Typical Application



# MM5840 TV channel number (16 channels) and time display circuit

## general description

The MM5840 TV Channel Number and Time Display Chip is a monolithic metal gate CMOS integrated circuit which generates a display of channel numbers (up to 16 channels) and time readouts on the television screen.

By external connection, it has the option of displaying the channel number only while switching channels with a period controlled by the external RC time constant of a timeout monostable.

This chip includes all the logic required to provide two modes of operation, namely channel number, or channel number and time display.

In addition, it can have a five (hour tens, hour units, colon, minute tens, and minute units) or eight digit (hour tens, hour units, colon, minute tens, minute units, colon, second tens, and second units) display, depending on the digit select input logic level.

By employing the video gating input together with the video output, a symmetrical blanked rectangular frame around the display may be generated on the TV screen.

This chip serves as a display generator with BCD channel inputs, as provided from the clock chips MM53100 or MM53105. The position of the display on the TV screen can be controlled by adjusting external RC time constants.

## functional description

The channel number and time readout circuit operates with a 2 to 4.5 MHz input clock. Counters are incorporated in the chip, operated by the input clock to keep track of the time slots of the display.

The position of the display is controlled by adjusting the external RC time constants of the horizontal and vertical monostable multivibrators.

A 7-segment decoder is used to decode either channel inputs or time which is stored temporarily in the channel number buffers or 4-bit latches, respectively, depending on the time slot of the display. Each digit of time is stored in a 4-bit latch while it is being decoded and displayed, and the next digit enters the latch while the horisontal sweep is between digits. Digits 0 to 9 are decoded and can be displayed.

A time slot decoder is employed to decode the appropriate time slot and the digit to be displayed. It generates a video output signal that modulates the sweep of the television tube for the display on the screen.

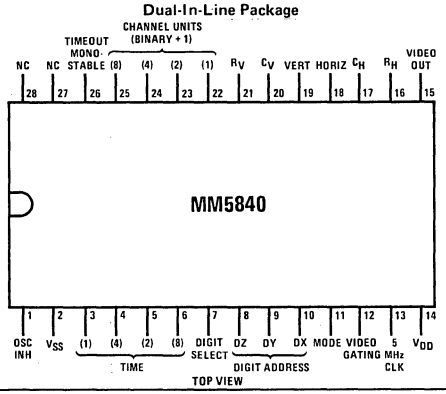
## features

- 12 or 24-hour operation (controlled by clock chip)
- 5 or 8-digit display
- Channel number leading zero blanking
- Single power supply
- Channel number only or channel number and time display
- Video gating output for generating a symmetrical blanked rectangular frame around the display
- Oscillator inhibit output
- Channel number display only while switching channels
- 4-bit binary plus one code for channel numbers

## functions

- 8-digit mode is selected by a logic "1" at digit select input
- Channel number and time mode is selected by a logic "1" at mode input
- Permanent channel number display is selected by a logic "0" at timeout monostable input

## connection diagram




**Order Number MM5840N**
**See Package 23**

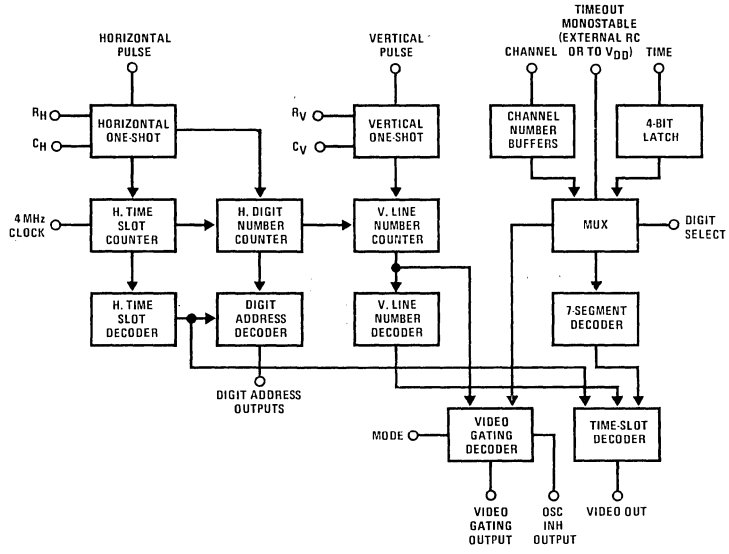
## absolute maximum ratings

| | |
|---|---|
| Supply Voltage ($V_{DD} - V_{SS}$) | −0.3V to +15V |
| Voltage at Any Pin | $V_{SS} - 0.3V$ to $V_{DD} + 0.3V$ |
| Operating Temperature | 0°C to +70°C |
| Storage Temperature | −55°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## electrical characteristics $V_{DD}$ = 12V, $V_{SS}$ = 0V, unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage | | | | | |
| $V_{DD}$ | $V_{SS}$ = 0 | 11 | 12 | 14 | V |
| Power Supply Current | | | | 800 | μA |
| Input Voltage Levels | | | | | |
| Time, Oscillator, Digit | | | | | |
| Select, and Mode Inputs | | | | | |
| Logical Low | | $V_{SS}-0.3$ | $V_{SS}$ | $V_{SS}+0.9$ | V |
| Logical High | | $V_{DD}-0.5$ | $V_{DD}$ | $V_{DD}+0.3$ | V |
| Channel Inputs | | | | | |
| Logical Low | | $V_{SS}-0.3$ | $V_{DD}-5$ | $V_{DD}-4.5$ | V |
| Logical High | | $V_{DD}-0.5$ | $V_{DD}$ | $V_{DD}+0.3$ | V |
| Horizontal and Vertical Inputs | | | | | |
| Logical Low | | $V_{SS}-0.3$ | $V_{DD}-5$ | $V_{DD}-4.5$ | V |
| Logical High | | $V_{DD}-0.5$ | $V_{DD}$ | $V_{DD}+0.3$ | V |
| Input Frequency | | 2 | | 4.5 | MHz |
| Oscillator | Interfacing with MM53100, MM53105 | | | | |
| Horizontal | Pulse Width = 14 μs | | 15.75 | | kHz |
| Vertical | Pulse Width = 1 ms | | 60 | | Hz |
| Output Voltage Levels | | | | | |
| Video Gating, Osc. Inhibit | | | | | |
| Digit Address and Video Outputs | | | | | |
| Logical Low | | $V_{SS}-0.3$ | $V_{SS}$ | $V_{SS}+0.9$ | V |
| Logical High | | $V_{DD}-0.5$ | $V_{DD}$ | $V_{DD}+0.3$ | V |
| One-Shot Output Pulse Duration | | | | | |
| Horizontal | | 15 | | 50 | μs |
| Vertical | | 1.5 | | 13 | ms |
| Output Drive | | | | | |
| Video Output | | | | | |
| Logical Low | $V_{SS}$ + 1V | \|−1\| | | | mA |
| Logical High | $V_{DD}$ − 1V | 1 | | | mA |
| Video Gating and Osc. Inhibit Outputs | | | | | |
| Logical Low | Output Forced Up to $V_{DD}$ − 4.5V | \|−2\| | | | mA |
| Logical High | $V_{DD}$ − 1V | 0.2 | | | mA |
| External RC | | | | | |
| $C_{VERTICAL}$ | | | 0.1 | | μF |
| $C_{HORIZONTAL}$ | | | 0.001 | | μF |
| $R_{VERTICAL}$ | | | 50 | | kΩ |
| $R_{HORIZONTAL}$ | | | 100 | | kΩ |
| $C_{TIMEOUT}$ | | | 5 | | μF |
| $R_{TIMEOUT}$ | | | | 1 | MΩ |
| Propagation Delay | | | | | |
| Video Gating and Osc. Inhibit Outputs | From Input Clock to Oscillator Inhibit or Video Gating Outputs | | | 2 | clock pulses |
| Input Leakage | | | | 1 | μA |
| Input Capacitance | | | | 5 | pF |

## block diagram



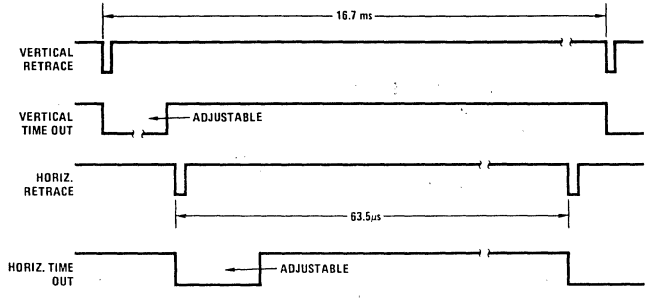

## truth table

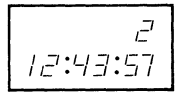
**Digit Address (DX, DY, DZ) Codes**

| CODES | DURING RESET | DIGITS 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|---|
| DX | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| DY | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| DZ | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |

## timing diagram

**With Video Gating, Output Gated with Video Output**

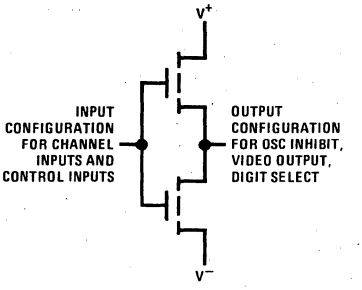


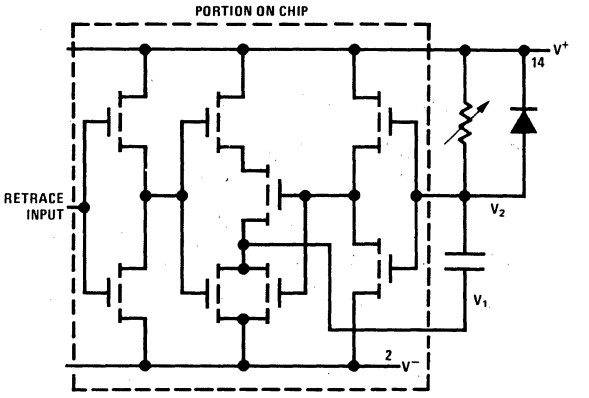



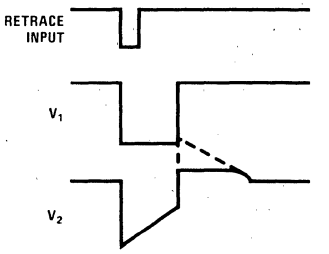

Note: Display Slightly Slanted

## typical applications

**Horizontal and Vertical One-Shot Circuit**

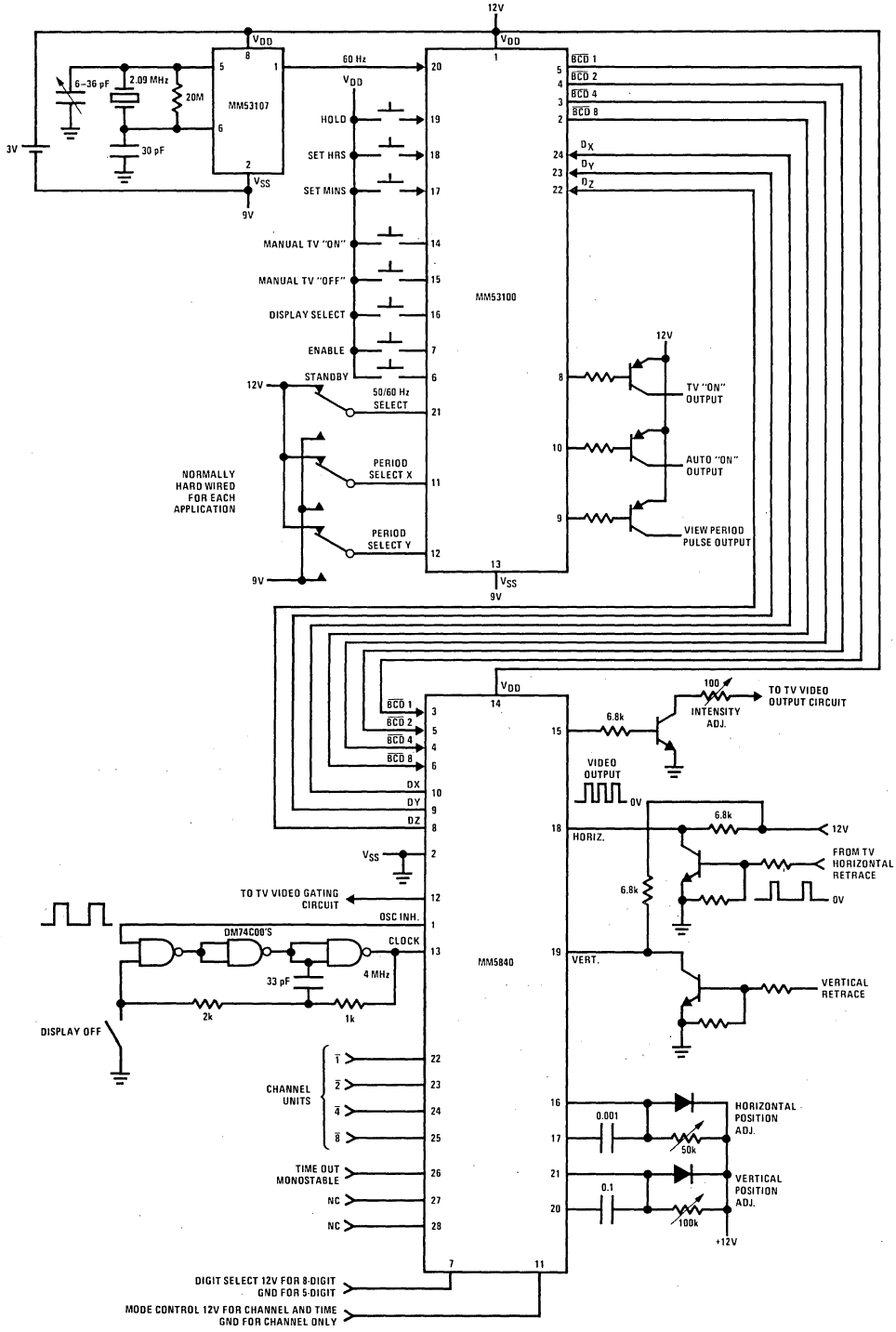


## typical applications (Continued)

**TV Channel and Time Display Interfacing MM53100**




Note. For interfacing with MM53105, refer to MM53105 specifications.

National Semiconductor

# MM58106 Digital Clock and TV Display Circuit

## General Description

The MM58106 is a monolithic CMOS integrated circuit which generates a display of channel number and time on the television screen. The circuit can either display United States channel number (2–83) or European program number (1–16). Time display can be 4 or 6-digit, in either 12 or 24-hour mode. Timekeeping is controlled from a 50 Hz or 60 Hz input. The position of the display on the TV screen is controlled by adjusting the external RC time constants.

The circuit is packaged in a 28-lead dual-in-line epoxy package.

## Features

- Single chip clock and channel/program number display
- 12 or 24-hour operation
- 4 or 6-digit time display
- Channel or program number display
- 50/60 Hz operation
- Channel and time display on channel change
- Continuous time display
- Continuous time and channel display

## Block and Connection Diagrams

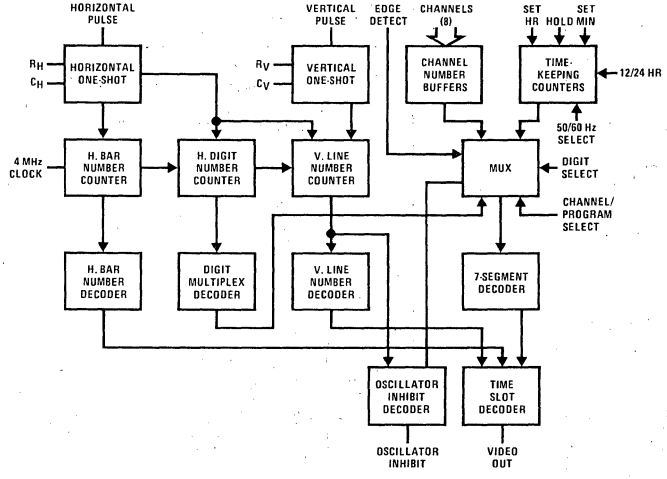
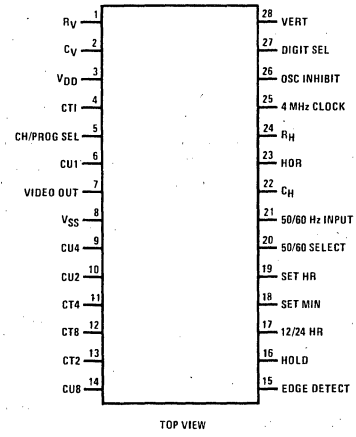



FIGURE 1

Dual-In-Line Package




**Order Number MM58106N**
**See Package 23**

FIGURE 2

## Absolute Maximum Ratings

| | |
|---|---|
| Supply Voltage ($V_{DD} - V_{SS}$) | 5.5V |
| Voltage at Any Pin | $V_{SS}$ − 0.3V to +5.5V |
| Operating Temperature | 0°C to +70°C |
| Storage Temperature | −55°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics $V_{DD}$ = 5V, $V_{SS}$ = 0V, unless otherwise specified

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage, $V_{DD}$ | $V_{SS}$ = 0 | 4.75 | 5 | 5.25 | V |
| Power Supply Current | | | | 800 | μA |
| Input Voltage Levels | | | | | |
| Channel Inputs | | | | | |
| Logical Low | | $V_{SS}$−0.3 | $V_{DD}$−5 | $V_{DD}$−4.5 | V |
| Logical High | | $V_{DD}$−0.3 | $V_{DD}$ | $V_{DD}$+0.3 | V |
| Horizontal and Vertical Inputs | | | | | |
| Logical Low | | $V_{SS}$−0.3 | $V_{DD}$−5 | $V_{DD}$−4.5 | V |
| Logical High | | $V_{DD}$−0.3 | $V_{DD}$ | $V_{DD}$+0.3 | V |
| Set Mins, Set Hours, Hold, 12/24-Hour Select, 50/60 Hz Select, Channel/Program Select | Internal Pull-Up Resistor to $V_{DD}$ (600k Min) | | | | |
| Logical Low | | $V_{SS}$−0.3 | $V_{SS}$ | $V_{SS}$+0.3 | V |
| Logical High | | | Open | | |
| All Others | | | | | |
| Logical Low | | $V_{SS}$−0.3 | $V_{SS}$ | $V_{SS}$+0.3 | V |
| Logical High | | $V_{DD}$−0.3 | $V_{DD}$ | $V_{DD}$+0.3 | V |
| Input Frequency | | | | | |
| 4 MHz Clock | | 1 | 4 | 4.5 | MHz |
| Horizontal | Pulse Width = 14 μs | | 15.75 | | kHz |
| Vertical | Pulse Width = 1 ms | | 60 | | Hz |
| Output Voltage Levels | | | | | |
| Oscillator Inhibit and Video Output | | | | | |
| Logical Low | | $V_{SS}$−0.3 | $V_{SS}$ | $V_{SS}$+0.9 | V |
| Logical High | | $V_{DD}$−0.5 | $V_{DD}$ | $V_{DD}$+0.3 | V |
| One-Shot Output Pulse Duration | | | | | |
| Horizontal | | | 50 | | μs |
| Vertical | | | 13 | | ms |
| Output Drive | | | | | |
| Video Output | | | | | |
| Logical Low | $V_{SS}$ + 1V | (−1) | | | mA |
| Logical High | $V_{DD}$ − 1V | 1 | | | mA |
| Oscillator Inhibit Output | | | | | |
| Logical Low | Output Forced Up to $V_{DD}$ − 4.5V | (−2) | | | mA |
| Logical High | $V_{DD}$ − 1V | 0.2 | | | mA |
| External RC | | | | | |
| $C_{VERTICAL}$ | | | 0.1 | | μF |
| $C_{HORIZONTAL}$ | | | 0.001 | | μF |
| $R_{VERTICAL}$ | | | 100 | | kΩ pot |
| $R_{HORIZONTAL}$ | | | 100 | | kΩ pot |
| Propagation Delay Oscillator Inhibit Output | From Input Clock to Oscillator Inhibit Output | | | 2 | clock pulses |
| Input Leakage | | | | 1 | μA |
| Input Capacitance | | | | 5 | pF |
| Edge Detect Pulse Duration | C = 2 μF, R = 1 MΩ | | 2 | | sec |

## Functional Description

A block diagram of the MM58106 TV timer is shown in *Figure 1*. A connection diagram is shown in *Figure 2*. Unless otherwise indicated, the following discussions are based on *Figure 1*.

**50 or 60 Hz Input:** This input has a shaping circuit which allows using a filtered sinewave input. A simple RC filter such as shown in *Figure 3* should be used to remove possible line voltage transients that could either cause the clock to gain time or damage the device. The input should swing between $V_{SS}$ and $V_{DD}$. The shaper output drives a counter chain which performs the timekeeping function.

Alternatively, in a crystal controlled battery operated system, an oscillator and prescaler circuit such as the MM5369 or MM5368 could be used as a timebase. With a backup battery, the clock will keep time even if the TV power is OFF.

**50 or 60 Hz Select Input:** This input programs the prescale counter to divide by either 50 or 60 to obtain an internal 1 pps timebase. The counter is programmed for 60 Hz operation by connecting this input to $V_{SS}$. This input has an internal pull-up resistor. Therefore, leaving this input unconnected programs the clock for 50 Hz operation.

**Time Setting Inputs:** Inputs to set hours and set minutes as well as a hold input, are provided. Internal pull-up resistors provide the normal timekeeping function. Switching any one of these inputs (one at a time) to $V_{SS}$ results in the desired time setting function. Set Hours advances hours information at one hour per second, and Set Minutes advances minutes information at one minute per second, without roll over into the hours counter. The hold input stops the clock to the minutes counter and resets the seconds counter to 00.

**Display Control:** The channel number and time display circuits operate from the 4 MHz input clock frequency. This display character size is inversely proportional to the input clock frequency.

The horizontal and vertical position of the display is controlled by adjusting the external RC time constants ($R_H$, $C_H$, $R_V$, $C_V$).

These monostables are triggered by the horizontal and vertical retrace signals as shown in the timing diagram in *Figure 4 and Figure 5*.

A 7-segment decoder is used to decode either channel inputs or time. Also a time slot decoder is employed to decode the appropriate time slot and the digit to be displayed. It generates a video output signal that can modulate the sweep of the television tube for the on-screen display.

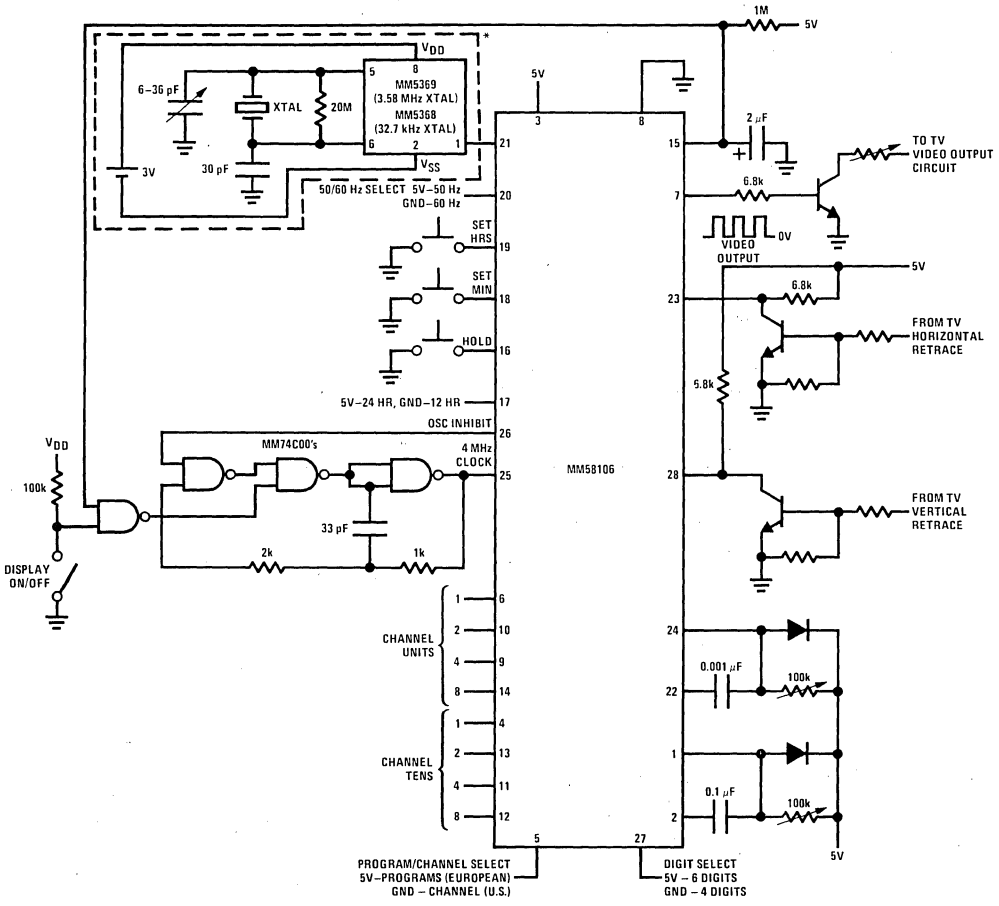
**Channel/Program Number Select:** This control pin has a pull-up resistor to $V_{DD}$. With the input open, the chip will accept a binary plus 1 code on the CU1 to CU8 inputs and display the program number from 1 to 16 (European application). With this input at $V_{SS}$, inputs CU1 to CU8 and CT1 to CT8 will accept BCD inputs for channel units and channel tens, respectively, and display channels 2–83, (U.S. applications), as shown in Table I. Actually, the circuit can display channel number from 0 to 99. Inputs CU1 to CU8 and CT1 to CT8 are negative logic inputs: "1" being $V_{SS}$ level and "0" being $V_{DD}$ level.

**Edge Detect:** It is an open-drain output and is a voltage sense input. The output will be ON and sink current to $V_{SS}$ when there is a channel or program change. When the device senses the voltage across the external capacitor *(Figure 3)* less than the lower trip point, the output will be turned OFF (channel display is ON) and the capacitor is then charged towards $V_{DD}$ until the voltage reaches the higher trip point (channel display is OFF). The external RC on this pin will control the ON time of the channel or program number display. When this output is tied to $V_{SS}$, the channel or program number will display continuously with time display. With Edge Detect tied to $V_{DD}$, CU1 to CU8 and CT1 to CT8 tied to either $V_{DD}$ or $V_{SS}$, the device can be used to display time information without channel/program number displayed.

**TABLE I. CHANNEL/PROGRAM FUNCTIONAL TRUTH TABLE**

| CU1/CT1 | CU2/CT2 | CU4/CT4 | CU8/CT8 | CHANNEL/PROGRAM SELECT | | |
|---|---|---|---|---|---|---|
| | | | | $V_{DD}$ OR OPEN | $V_{SS}$ | |
| | | | | | TEN | UNIT |
| 0 | 0 | 0 | 0 | 1 | BLANK | 0 |
| 1 | 0 | 0 | 0 | 2 | 1 | 1 |
| 0 | 1 | 0 | 0 | 3 | 2 | 2 |
| 1 | 1 | 0 | 0 | 4 | 3 | 3 |
| 0 | 0 | 1 | 0 | 5 | 4 | 4 |
| 1 | 0 | 1 | 0 | 6 | 5 | 5 |
| 0 | 1 | 1 | 0 | 7 | 6 | 6 |
| 1 | 1 | 1 | 0 | 8 | 7 | 7 |
| 0 | 0 | 0 | 1 | 9 | 8 | 8 |
| 1 | 0 | 0 | 1 | 10 | 9 | 9 |
| 0 | 1 | 0 | 1 | 11 | BLANK | BLANK |
| 1 | 1 | 0 | 1 | 12 | BLANK | BLANK |
| 0 | 0 | 1 | 1 | 13 | BLANK | BLANK |
| 1 | 0 | 1 | 1 | 14 | BLANK | BLANK |
| 0 | 1 | 1 | 1 | 15 | BLANK | BLANK |
| 1 | 1 | 1 | 1 | 16 | BLANK | BLANK |

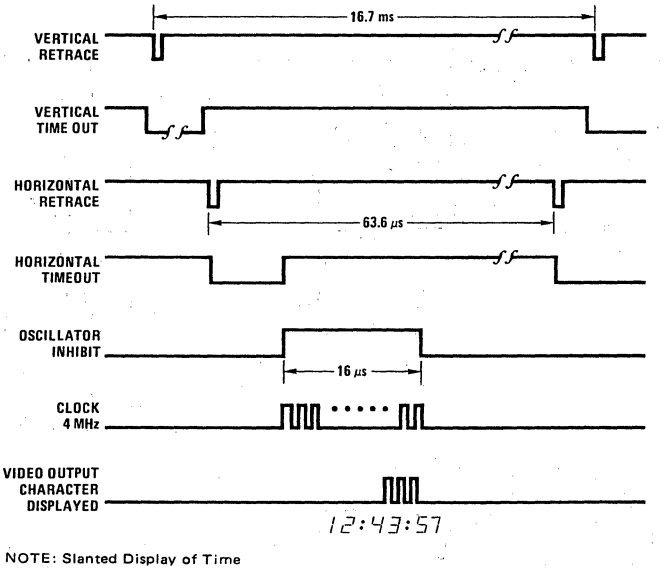
## Functional Description (Continued)




* Optional Input Circuit Utilizing 50/60 Hz AC

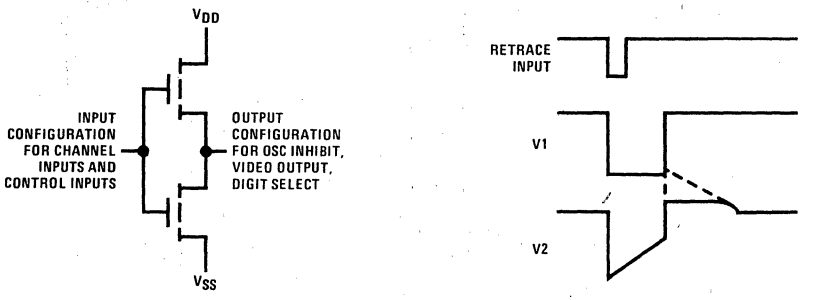
FIGURE 3

## Functional Description (Continued)




NOTE: Slanted Display of Time

FIGURE 4. Timing Diagram



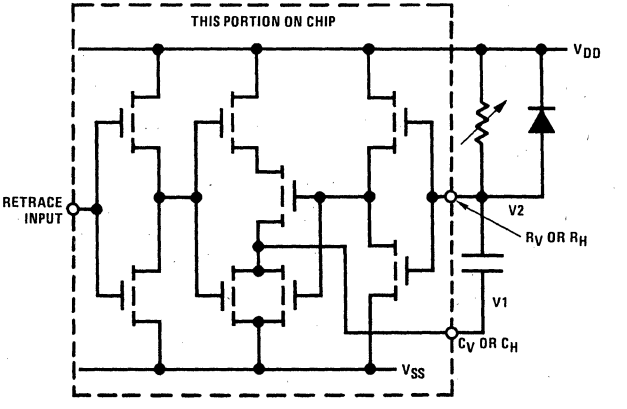




FIGURE 5. Horizontal and Vertical One-Shot Circuit

National Semiconductor

Television/Radio

# MM58142 TV Synthesizer

## General Description

The MM58142 is designed to interface with a varactor VHF/ UHF television tuner, a high speed variable modulus prescaler, and an active low pass filter to form a TV frequency synthesizer phase-locked loop capable of tuning the local oscillator to the required frequency for each of the 82 available channels.

The MM58142 also provides interfacing, control, and timing circuits for remote or local customer control, plus optional channel display means.

## Features

- Up/down channel stepping
- Up/down channel skipping when used with an EPROM skip memory
- Last channel recovery after momentary power shut down
- Direct keyboard address of channel number
- Compatibility with CMOS display driver IC's
- Compatibility with an on-screen display drive MM48146
- AFT operation for offset channels

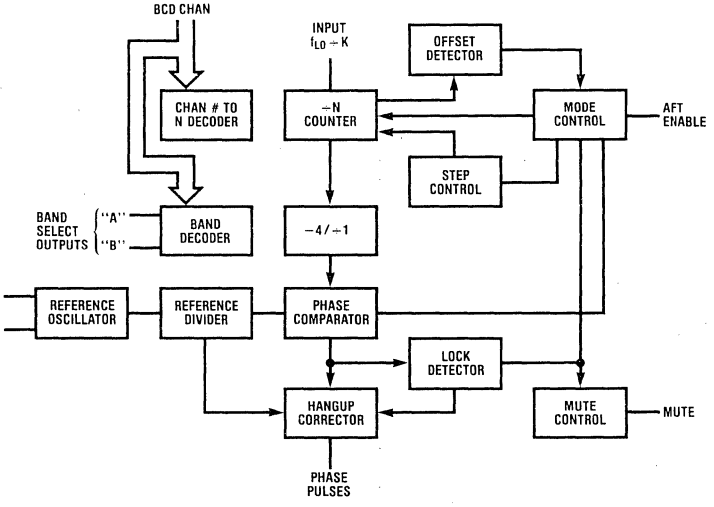
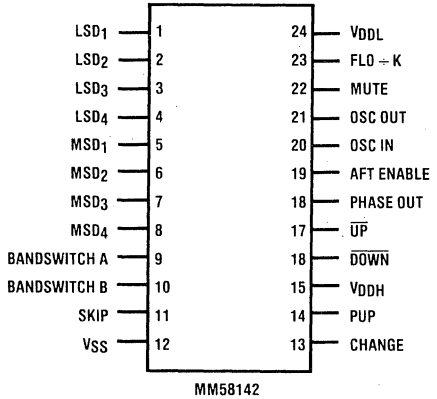



**Figure 1. MM48142 TV Synthesizer with AFT Drift Simplified Block Diagram**




MM58142

**Pin Connection Diagram**

**Order Number MM58142N**

**See Package 22**

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ to $V_{SS}$ + 12V |
| Operating Temperature | 0°C to 70°C |
| Storage Temperature | −55°C to +150°C |
| Power Dissipation | 500 mW at 25°C ambient |

## Electrical Characteristics

$T_A$ within operating range, $V_{DDL}$ = 4.5V to 6.0V, $V_{SS}$ = 0V

| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| $V_{DDL}$ | | 4.5 | | 6.0 | V |
| $V_{DDH}$ | | 10 | | 12 | V |
| $I_{DDL}$ | $V_{DDL}$ = 6.0V | | 50 | 70 | mA |
| $I_{DDH}$ | $V_{DDH}$ = 12.0V | | 3 | 6 | mA |
| Power Supply Rise Time | | 1.0 | | | ms |
| LSD BCD Input/Output | | | | | |
| Input Logic "1" | | 2.4 | | $V_{DDH}$ | V |
| Input Logic "0" | | −0.3 | | 0.6 | V |
| Output Logic "1" | At 0.5 mA | $V_{DDL}$ − 0.5 | | | V |
| Output Logic "0" | At 0.4 mA | | | 0.4 | V |
| Tristate | | | | 10.0 | μA |
| MSD Outputs | | | | | |
| Logic "1" | At 0.5 mA | $V_{DDL}$ − 0.5 | | | V |
| Logic "0" | At 0.4 mA | | | 0.4 | V |
| PUP Buss | | | | | |
| Input Logic "1" | | 2.4 | | $V_{DDH}$ | V |
| Input Logic "0" | | −0.3 | | 0.6 | V |
| Output Logic "0" | At 0.2 mA | | | 0.4 | V |
| Pull-Up to $V_{DDL}$ | At 2.4V | 30 | | 150 | μA |
| Skip Input | | | | | |
| Logic "1" | | 2.4 | | $V_{DDH}$ | V |
| Logic "0" | | −0.3 | | 0.6 | V |
| Input Leakage | | −1.0 | | 1.0 | μA |
| AFT Enable I/O | | | | | |
| Input Logic "1" | | 2.4 | | $V_{DDH}$ | V |
| Input Logic "0" | | −0.3 | | 0.6 | V |
| Output Logic "0" | At 0.4 mA | | | 0.4 | V |
| Pull-Up to $V_{DDL}$ | At 2.4V | 30 | | 150 | μA |
| Change Output | | | | | |
| Logic "1" | At 0.5 mA | $V_{DDL}$ − 0.5 | | | V |
| Logic "0" | At 0.4 mA | | | 0.4 | V |
| Mute Output | | | | | |
| ON Saturation Voltage | At 1 mA | | | 1 | V |
| OFF (Leakage) Current | At 6.6V | | | 1 | μA |
| FLO ÷ K | AC Coupled Internally | 0.7 | | 1.5 | $V_{RMS}$ |
| Input DC Level | | 1.0 | | $V_{DDL}$ | V |
| Input Signal Rise/Fall | | 40 | | 100 | ns |
| Input Leakage | | −1.0 | | 1.0 | μA |
| Input Capacitance | | | | 5.0 | pF |
| Band Select Out | | | | | |
| Logic "1" | At 100 μA | $V_{DDH}$ − 2.0 | | | V |
| Logic "0" | At 150 μA | | | 0.4 | V |
| Phase Pulses Output | | | | | |
| Logic "1" | At 1 mA | $V_{DDL}$ − 0.5 | | | V |
| Logic "0" | At 1 mA | | | 0.5 | V |
| Tristate | | | | 200 | nA |
| $\overline{UP}$, $\overline{DOWN}$ Inputs | | | | | |
| Logic "1" | | 2.4 | | 12 | V |
| Logic "0" | | −0.3 | | 0.6 | V |
| Debounce/Release Time | | 16 | | 25 | ms |

**Note:** Nominal oscillator frequency: 4.0 mHz

## Functional Description

The MM58142 TV Synthesizer is to be operated in conjunction with a VHF/UHF band varactor-controlled television tuner, a high-speed frequency pre-scaler, and an active low-pass filter.

The above elements in combination comprise a TV frequency synthesizer phase-locked loop, capable of precisely tuning the tuner local oscillator to the required frequency for each of the 82 TV channels. Additionally, an interface to conventional AFT and control logic in the MM58142 provide precise tuning of nonstandard-frequency VHF band channels whose picture carrier frequencies are within ±2 MHz of nominal, and UHF band channels whose picture carrier frequencies are within ±1 MHz of nominal. The method of operation, synthesis only or a combination of synthesis and AFT, is controlled by an external mode select switch.

The synthesizer LSI also provides interfacing, control, and timing circuits for remote or local customer control and optional channel number display.

The control and display options include:

1. Up or Down Channel stepping
2. Up or Down Channel skipping (non-volatile skip memory required)
3. Last channel recovery after power shutdown (non-volatile last channel memory required)
4. Direct keyboard address of channel number
5. Compatibilty with an on-screen display generator IC, the MM58146

## General Operation

### $\overline{\text{UP}}/\overline{\text{DOWN}}$ (Input)

The up/down input commands, after being debounced, produce an immediate increase or decrease in channel number to the next legal channel number. Then after a pause of approximately 650 ms, the channel number changes at a rate of approximately 10 counts per second. Once the switch is closed, momentary opening (less than 16 ms) or contact noise due to switch aging or to pressure variation on the switch, is ignored.

With a "Skip Memory" present, the "$\overline{\text{UP}}/\overline{\text{DOWN}}$" command produces a "skip" to the next programmed channel as quickly as is practical and then pauses for 655 ms, i.e., stepping between programmed channels at approximately 1.5 changes per second.

### DISPLAY (Output/Input)

The output consists of two sets of four lines, one set of which determines the tens digit (MSD); the other set determines the units digit (LSD). The output is a standard positive logic BCD code with output drivers capable of providing sufficient isolation to prevent any signal pick-up on the output lines from changing the output state of the binary counters.

### MASTER/SLAVE (Input)

The four lines used for the units BCD (LSD) output are bi-directional, under control of the M/S input, and capable of loading direct entries either from a keyboard or from a remote system. When the "M/S" input is debounced and a "slave" state is sensed, a sufficient time (> 100 $\mu$s) is allowed for the bi-directional LSD bus to switch to the "slave" or input mode before the input data on the bus is entered. The input data is entered into the "Buffer" or the "Units Counter". After completion of the entry, independent of the state of the "M/S" line, the LSD bi-directional bus is switched to the output state. No additional entries are possible until the "M/S line reverts back to the "master" state and a new "slave" state is sensed. The "slave" input is defined as the simultaneous closure to ground (within one debounce time) of both the "$\overline{\text{UP}}$" and "$\overline{\text{DOWN}}$" inputs.

### SKIP (Input)

A logic "1" on the "Skip" input, during the time "Change" is low, causes the channel number to increment or decrement to the next legal channel depending upon the state of $\overline{\text{UP}}/\overline{\text{DOWN}}$, in conjunction with a "Skip Memory", it is thereby possible to limit the set of channels accessible via the "$\overline{\text{UP}}/\overline{\text{DOWN}}$" command to those programmed into the "Skip Memory". The "Skip" input can also be used with a "Last Channel Memory" to load, by serial interrogration, the "Last Channel Memory" contents. In addition, the "Skip" input can be used to start the long timer ($\tau_L$). During the time "Change" is high, a high on the "Skip" input starts the long timer ($\tau_L$) without executing a "Skip" operation.

In order to prevent false channel changes from occurring due to kine-arc or system noise, the "Skip" input is inhibited approximately 98 ms after the last "$\overline{\text{UP}}/\overline{\text{DOWN}}$" operation of the interface-control logic.

### CHANGE (Output)

A logic "1" on the "Change" output line indicates that the 8-line BCD output is in the process of changing and shall not be considered valid. The change signal will go high at least 2 $\mu$s prior to any change on the output bus and will stay high for at least 40 $\mu$s after the output bus has been changed.

During the "Half Entry" mode, "Change" remains high until either the "Half Entry" mode has been aborted or a second digit entry has been made. "Change" will stay high for at least 40 $\mu$s after either the second digit entry or abort operation has been completed and the output bus data is valid. After a momentary power drop out ("Standby" mode), "Change" will remain high for approximately 5.25 seconds, or until after the completion of the next entry.

### "PUP" (Bus)

The "Power is Up" (PUP) signal is internally generated on the chip and has the capability of being bussed with the "PUP" lines of other integrated circuits used in the tuning system. A high on the "PUP" line indicates that the logic power is within the normal operating range and that the chip logic is capable of normal speed operations.

### Direct Keyboard Entry

In order to tune the TV receiver, two consecutive keyboard entries are required within a 5.25 second period.

The first entry is loaded into the "Buffer" and puts the system into a "Half Entry" state. In the "Half Entry" state, the output bus MSD is connected to the "Buffer" and the LSD is forced to a code which will cause the display to show a "dash". During the time the system is in the "Half Entry" state, the "Change" output line is held high since the bus data is no longer valid. The "Change" line will remain high until after either the second keyboard entry is made or the "Half Entry" state is aborted due to a time out of UP/DOWN entry. Only after the second keyboard entry is made is the synthesizer logic instructed to tune to a new channel.

If the second entry is an UP/DOWN command rather than a digit, the "Half Entry" state will be aborted and the display will revert back to the channel number corresponding to the channel being watched for 655 ms and then if the UP/DOWN input is still valid, will continue with the normal UP/DOWN operation.

If one of the direct keyboard entries resulting from the simultaneous closure of two or more keys is non-BCD or Hex (A-F), the system will reject that entry and revert the display back to the channel number corresponding to the channel being watched. If the system is in the "Half Entry" state at the time of the Hex entry, the logic will inhibit the normal data transfer clock to the "Tens" and "Units" counter. This will result in the reversion back to the old channel number. If the system is not in the "Half Entry" state at the time of the Hex entry, a normal load operation is executed with the exception that the input clock to the 5.25 second timer (TIM5) is switched from the normal 488 Hz clock to a high speed 31 kHz clock. This will result in the execution of a "Time Out Abort" in 82 ms. At the end of the 82 ms the display will revert back to the original channel number.

### Standby Power Operation

After initial turn-on, any momentary power failure will result in the logic establishing a power "Standby" state with the "Binary" UP/DOWN counter inputs disconnected or isolated from the 5V logic and powered by the separate 10V supply. Normal power-up will occur if the 10V supply ever drops below the level. With the supply below the "P10" level it must be assumed that the counter contents are no longer valid and therefore upon the return of power the normal "PUP" or initialization process will be executed. If during the momentary power failure the 10V supply remains above the "P10" level, but the 5V supply drops below the "P5" level, the system will immediately generate a "Standby" state for the 5V logic and "Disconnect" state for the Binary UP/DOWN counter. "Disconnect" is generated whenever the 5V power drops below the P5 level.

"Standby" will go high at start of "Disconnect" and will remain high for approximately 262 ms after "Disconnect" goes low. During "Standby" all I/O functions in process will be aborted and no new I/O function will be initiated. During "Disconnect" the Binary UP/DOWN counter's inputs will be inhibited and no counter state changes will be allowed. Independent of the Binary Counter's input state, the transitions of "Disconnect" will not alter the counter state. In addition, the output "Change" will go high during "Standby" and will remain high for approximately 5.25 seconds, or until after the completion of the next entry. During the time "Disconnect" is high the on-chip load on the 10V supply is reduced to a minimum.

| | Min. | Max. | Units |
|---|---|---|---|
| P10 high | 4.0 | 9.8 | Volts |
| P10 low | 1.8 | 5.8 | Volts |
| P5 | 1.2 | 4.0 | Volts |

P10 high $\geq$ P10 low + 1.0V

## Standard Synthesizer Blocks

Figure 1 is a simplified block diagram of the tuner control portion of the synthesizer IC. The signal input to the synthesizer is $f_{LO} \div K$, i.e., the appropriate VHF or UHF local oscillator signal having been divided in frequency by a factor K by the high-speed prescaler. The blocks in the IC which comprise a standard synthesizer are:

| | |
|---|---|
| ÷N Counter: | A programmable counter which divides the input signal by a factor N where N is an integer number equal to the local oscillator frequency, in MHz, of the requested channel. |
| Channel # to N Decoder | Its input is the 8-bit, BCD channel number, supplied by the input portion of the IC, and its output is the factor N. |
| ÷4/÷1 Counter | A bi-modulus counter with 4 as the modulus for VHF band channels and 1 the modulus for UHF band channels. |
| Reference Oscillator | Connected through IC pins to a 4 MHz crystal feedback network |
| Reference Divider | An 11-stage binary ripple counter, used to provide timing signals, and, in particular, 10 stages are used to provide the 3.90625 kHz phase-lock reference. |
| Phase Comparator | An edge-triggered phase comparator. In the synthesis mode, the output is tri-state. When the signals are phase-locked, with 0° phase error, the output is off. The polarity of the output pulses is determined by the polarity of phase error, and the pulse width determined by the magnitude of the phase error. In the AFT mode, the phase comparator output is not used. |
| Band Decoder | Provides two band select logic outputs used by the operational amplifier band switch IC to control tuner power for low VHF, high VHF, and UHF bands. |
| Hangup Corrector | Automatically senses and corrects for the situation(s) in which the varactor tuning voltage is at one of the power supply stops and the synthesizer phase detector output pulse polarity is such as to hold the tuning voltage at the power supply stop. |

Mute Control — Provides a short to ground for the sound signal when a channel change is requested. The short is removed once the tuning voltage has stabilized as indicated by the LOCK detector. Muting remains effective when an illegal channel is being tuned.

The remaining blocks are associated with the second tuner control method of operation, in which both synthesis and AFT are used to provide a precise tuning capability for nonstandard VHF TV signal frequencies within ±2 MHz of nominal and UHF frequencies within ±1 MHz of nominal:

Lock Detector — When in the synthesis mode, provides an indication that the frequency difference between the reference and the counted-down local oscillator signals is arbitrarily small. This indication is a prerequisite for switching to the AFT mode and for disabling the muting function.

Offset Detector — When in the AFT mode, provides an indication whenever the local oscillator has deviated by more than 1.25 MHz from the frequency synthesized just prior to entering the AFT mode.

Step Control — Controls which of three frequencies are to be synthesized for each VHF channel. The frequencies, in the order used, are 0, or nominal, +1 MHz, and −1 MHz. For UHF channels only the 0, or nominal frequency is synthesized.

Mode Control — Provides synchronized switching from synthesis to AFT mode and vice versa. It also generates various sampling, switching, and reset signals for the system.

The **Reference Oscillator** is an inverter with both input and output connected through IC pins to a crystal feedback network. The oscillator frequency is to be 4.000000 MHz ± 0.00537.

The **Reference Divider** contains a 13 stage binary ripple counter. The input to the divider chain is the signal 4 MHz from the Reference Oscillator. The output of the 13th stage, the signal 500 Hz, is fed to the Step Timing block in the input portion of the synthesizer IC. 500 Hz is the input clock to a timing chain which produces a number of time constants.

## Band Select Outputs

**Logic Truth Table**

| A | B | Output |
|---|---|---|
| 0 | 0 | No Band |
| 0 | 1 | Low VHF |
| 1 | 0 | High VHF |
| 1 | 1 | UHF |

National Semiconductor



# MM58146 TV Clock and Channel Display

## General Description

The MM58146 TV Clock and Channel Display Circuit is a monolithic NMOS integrated circuit which generates a display of time and channel number on a television screen.

The chip contains a time of day clock and all the logic required to display time and a 2 digit channel number from an external source, such as the MM53118 or MM58142 TV Digital Tuning Chips.

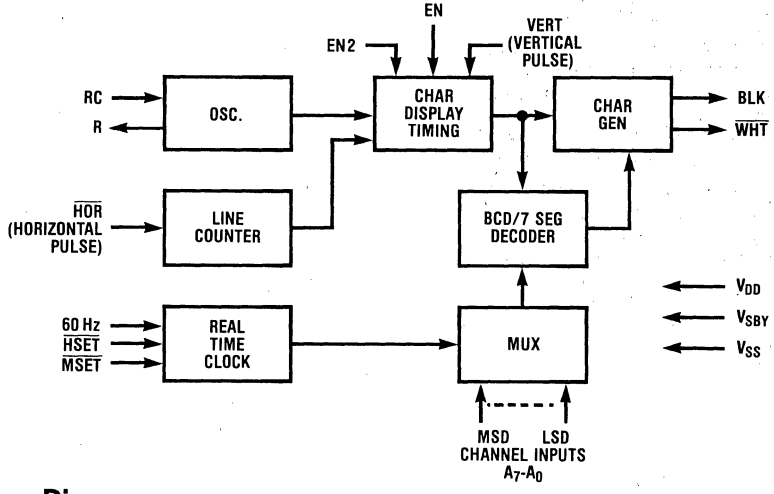
The horizontal, vertical, and 60 Hz inputs and display outputs are designed to directly interface to the video system of many TV sets.

The time and channel number are displayed on the same line near the bottom of the screen.

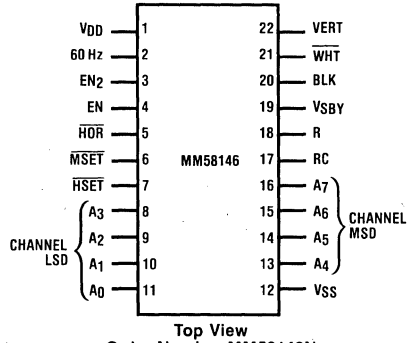
## Features

- 12 hour operation
- Leading zero blanking on hours display
- Black border around white character
- Interfaces to video system directly on many TVs
- Clock and channel display or channel only display
- Two digit display need not be channel number. Channel number is entered as BCD.

## Block Diagram




## Connection Diagram




Top View
Order Number MM58146N
See Package 21

## Absolute Maximum Ratings

| | |
|---|---|
| Supply Voltage | 12 Volts |
| Voltage at Any Pin | 12 Volts |
| Operating Temperature | 0 °C to 70 °C |
| Storage Temperature | −55 °C to +150 °C |
| Lead Temperature (Soldering, 10 seconds) | 300 °C |

## Electrical Characteristics $V_{DD} = 4.5$ to 6.0V, $V_{SBY} = 10$ to 12V, $V_{SS} = 0V$, $T_A = 0$ to 70 °C

| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| $V_{DD}$ | $V_{SS} = 0V$ | 4.5 | | 6.0 | V |
| $V_{SBY}$ | $V_{SS} = 0V$ | 10 | | 12 | V |
| Power Supply Current, $I_{DD}$ | $V_{DD} = 6.0V$ | | | 45 | mA |
| Power Supply Current, $I_{SBY}$ | $V_{DD} = 4.5$ to 6.0V | | | 7.0 | mA |
| | $V_{DD} = 0V$ | | | 1.4 | mA |
| Input Logic Levels | | | | | |
| BCD Inputs, EN, $\overline{HSET}$, $\overline{MSET}$ | | | | | |
| Logic Low | | −0.3 | | 0.6 | V |
| Logic High | | 2.4 | | 12 | V |
| Internal Pullup ($\overline{HSET}$ and $\overline{MSET}$ only) | | 30 | | 150 | μA |
| 60 Hz Input | | | | | |
| Logic Low | Without 100k Series | −0.3 | | 0.6 | V |
| Logic High | Input Resistor | 4.0 | | 12 | V |
| Logic Low | With 100k Series | −5.5 | | 0.6 | V |
| Logic High | Input Resistor | 4.0 | | 12 | V |
| Horizontal, Vertical Input | | | | | |
| Logic Low | Without 100k Series | −0.3 | | 0.4 | V |
| Logic High | Input Resistor | 4.0 | | 12 | V |
| Logic Low | With 100k Series | −70 | | 0.4 | V |
| Logic High | Input Resistor | 4.0 | | 30 | V |
| Output Logic Levels | | | | | |
| BLK Output | | | | | |
| Logic Low | I = 0.5 mA, Sink | | | 1.0 | V |
| Logic High | I = 2.5 mA, Source | 6 | | 11 | V |
| Rise and Fall Time | $C_L = 30$ pF | | | 70 | ns |
| $\overline{WHT}$ Output | | | | | |
| Logic Low | I = 1.5 mA, Sink | | | 0.4 | V |
| Logic High | I = 0.5 mA, Source | 6 | | 12 | V |
| Rise and Fall time | $C_L = 30$ pF | | | 70 | ns |
| Oscillator Frequency | | 4.5 | 5.0 | 5.7 | MHz |
| External RC (Note 1) | | | | | |
| R | | | 2.5 | | k |
| C | f = 5.25 MHz | | 33 | | pF |
| C | f = 5.7 MHz | | 20 | | pF |
| Power Supply Rise Time | | 1.0 | | | ms |

**Note 1:** The external resistor and capacitor must be located as close as possible to pins 17 and 18.

## Functional Description

The display is generated by means of an on-chip RC oscillator that operates nominally at 5 MHz. This 5 MHz oscillator is divided down to form a 2.5 MHz reference clock. Each character frame of the display occupies a rectangle 20 lines high and 8 reference clock periods wide. The characters are composed either from the segments of a figure "8" or from a centered "1". The main character output is the WHT output. The BLK output is used to provide a border around the main character which provides contrast on a normally white picture. The top line of the characters begins on the 192th horizontal line after the vertical retrace pulse. The time information is displayed on the left side of the screen and the channel number is displayed on the right side of the screen.

With power up of $V_{SBY}$ the real time clock is set to "00" and the display is enabled. It will display a "- -:- -" for 22 to 30 seconds. It is cleared when one of the time set inputs is brought low as described under $\overline{HSET}$ and $\overline{MSET}$ inputs.

If the $V_{DD}$ and $V_{SBY}$ supplies are on and $\overline{HOR}$ becomes a logic "1" for at least 7.5 seconds, the display will then be enabled for 22 to 30 seconds when $\overline{HOR}$ again becomes a logic "0".

If $V_{DD}$ goes down and up while $V_{SBY}$ remains up, the seconds count is set to "30" and the minutes and hours remain unchanged. The clock remains unchanged because it receives power from $V_{SBY}$. Also, under these power up conditions, a lock is established on display enable such that EN must go low and then high again to cause the display to appear.

The following describes the functions of all the input and output pins. In the following description, a low represents a logic "0" and a high represents a logic "1".

### Input Signals

**Vertical Pulse (VERT):** This signal resets the line counter and synchronizes the display to the vertical TV display.

**Horizontal Pulse ($\overline{HOR}$):** This signal increments the line counter and synchronizes the display to the horizontal TV display.

**60 Hz:** This input provides the 60 Hz time base for the real time clock. This input is designed to accomodate a sine wave input.

**Enable (EN):** This input, when high, will initiate the display. The display will stay on as long as EN is a logic "1". The display will remain on from 3 to 4 seconds after EN goes to a logic "0".

**Hours Set** and **Minutes Set ($\overline{HSET}$** and **$\overline{MSET}$):** These two inputs are used to set the clock to the desired time. When $\overline{HSET}$ is low, the hours counter will advance at a 2 Hz rate. When $\overline{MSET}$ is low, the minutes counter will advance at a 2 Hz rate. Setting one counter does not affect the other counter. When either $\overline{HSET}$ or $\overline{MSET}$ is low, the seconds counter is forcibly reset to "00". Also, when $\overline{MSET}$ and $\overline{HSET}$ are both low, the time remains constant. This is used to synchronize the clock to the master time source. Either or both input being low will cause the display to be active for the duration of the low signal, plus 3 to 4 seconds after the input goes high. If $\overline{MSET}$ and $\overline{HSET}$ are both low when power ($V_{DD}$) is applied and remain low after power ($V_{DD}$) is applied, the time of day information will be blanked from the display and only the channel number will be displayed.

A typical clock setting sequence is as follows:

1. Force $\overline{HSET}$ low until the hours display equals the desired value. Then raise $\overline{HSET}$ to a high.
2. Force $\overline{MSET}$ low until the minutes display equals the desired value.
3. To synchronize to another time source, force both $\overline{MSET}$ and $\overline{HSET}$ low simultaneously. This will hold the time information constant. When the clock and time source are equal, raise both $\overline{MSET}$ and $\overline{HSET}$, and the clock will start keeping time.

**Enable 2 (EN2):** This is a mode control pin that must be connected to $V_{SS}$.

**Channel Number Inputs ($A_7$-$A_0$):** These inputs are the external channel inputs. The most significant digits are $A_7$-$A_4$, with $A_7$ being the most significant bit (MSB). The least significant digits are $A_3$-$A_0$, with $A_3$ being the most significant bit. See the Truth Table for display font coding.

### Output Signals

**White Output ($\overline{WHT}$):** This is the main character output.

**Black Output (BLK):** This output is used to provide a border around the main character output ($\overline{WHT}$).

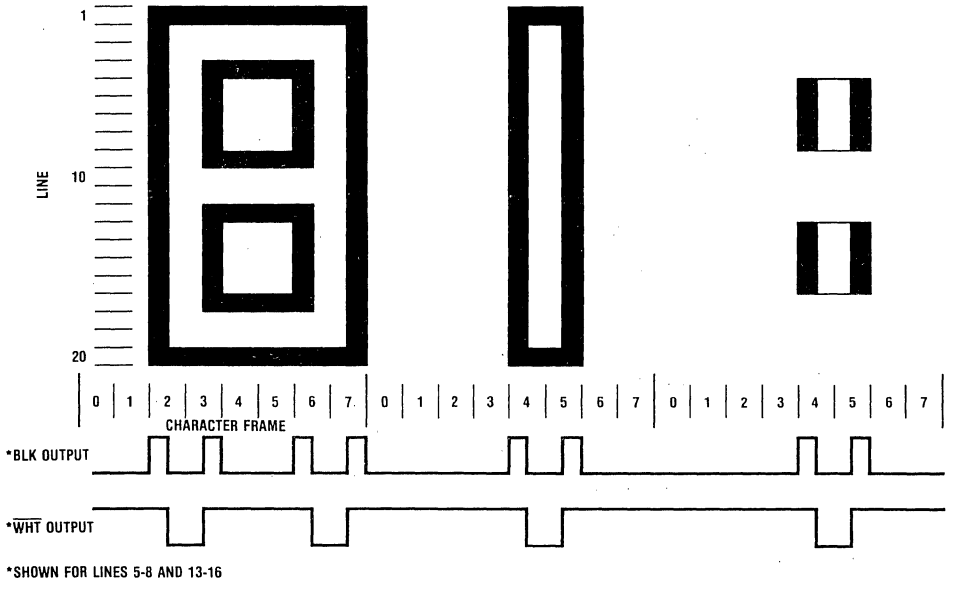
### Input/Output Signals

**Oscillator Input and Output (RC, R):** These pins connect the external resistor and capacitor to the oscillator to generate the main timing for the chip. See the Application Section for connection details.

### Display Font Coding

| Inputs $A_3$ $A_2$ $A_1$ $A_0$ or $A_7$ $A_6$ $A_5$ $A_4$ | $\overline{WHT}$ Output | |
|---|---|---|
| 0 0 0 0 | 0 | |
| 0 0 0 1 | 1 | |
| 0 0 1 0 | 2 | |
| 0 0 1 1 | 3 | |
| 0 1 0 0 | 4 | |
| 0 1 0 1 | 5 | |
| 0 1 1 0 | 6 | |
| 0 1 1 1 | 7 | |
| 1 0 0 0 | 8 | |
| 1 0 0 1 | 9 | |
| 1 0 1 0 | : | (Colon) |
| 1 0 1 1 | . | (Lower half of Colon) |
| 1 1 0 0 | P | |
| 1 1 0 1 | ▫ | (Upper half of 8) |
| 1 1 1 0 | – | (Dash) |
| 1 1 1 1 | | Blank |

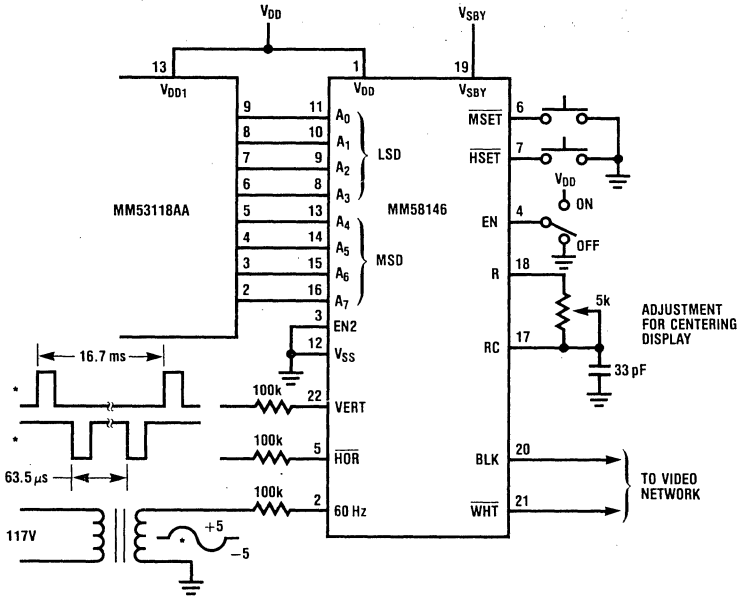
### Display Format and Timing




*SHOWN FOR LINES 5-8 AND 13-16

Display Format and Timing

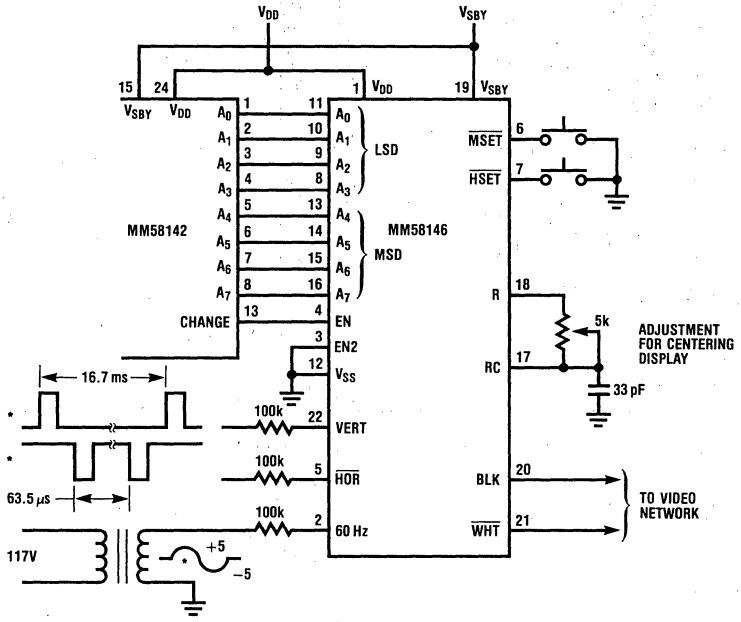
## Applications




*SEE ELECTRICAL SPECIFICATION FOR LIMITS ON THESE INPUTS

TV Channel and Time Display Interfacing with MM53118AA

## Applications (cont'd)




*SEE ELECTRICAL SPECIFICATION FOR LIMITS ON THESE INPUTS

**TV Channel and Time Display Interfacing with MM58142**

National Semiconductor

**Television/Radio**
**PRELIMINARY**

# MM58313 Varactor Tuner Display Circuit

## General Description

The TV varactor display circuit is a monolithic metal gate CMOS integrated circuit designed to provide an on-screen tuning bar graph scale for varactor tuned TV sets.

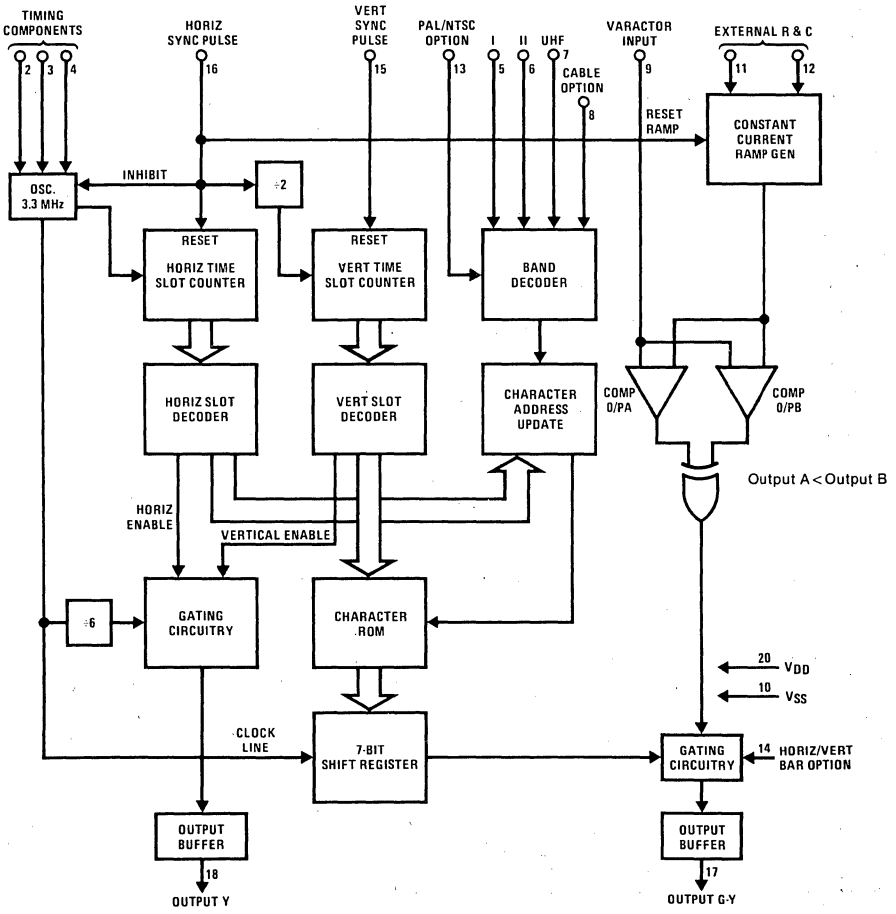
The tuning voltage is indicated by a vertical bar 36 lines deep and approximately 6 mm wide which traverses the screen in a linear relationship to the tuning voltage. An option input is provided so that the tuning voltage can be indicated by a horizontal line of variable length.

Three bands are provided, with an option to allow for cable transmission. Characters relating to the appropriate band selected are displayed on screen (Table I), the characters being formed by a 6 × 7 dot matrix.

## Features

- Electronic tuning scale for 3 bands
- Linear tuning indication
- Digital channel number display
- Pin option for cable TV systems (PAL only)
- Pin option for PAL and NTSC systems
- Pin option for finger or horizontal bar tuning voltage indication
- 12V operation compatible with digital tuning systems
- CMOS technology
- On-chip oscillator, frequency governed by external timing components

## Block Diagram



## Absolute Maximum Ratings

| | |
|---|---|
| Supply Voltage ($V_{DD}$-$V_{SS}$) | −0.3V to +18V |
| Voltage at Any Pin | $V_{SS}$ −0.3V to $V_{DD}$ +0.3V |
| Operating Temperature | 0°C to 70°C |
| Storage Temperature | −55°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | +300°C |

## Electrical Characteristics

$V_{DD}$ = 12V, $V_{SS}$ = 0V, $T_A$ = 0°C to 70°C, Clock Frequency 3.3 MHz, unless otherwise noted.

| Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| Power Supply Voltage $V_{DD}$ | $V_{SS}$ = 0 | 11.5 | 12 | 18 | V |
| Power Supply Current | | | 6 | 14 | mA |
| Input Voltage Levels | | | | | |
| Logic 0 | | $V_{DD}$ − 0.3 | $V_{SS}$ | $V_{SS}$ + 3.5 | V |
| Logic 1 | | $V_{DD}$ − 4.0 | $V_{DD}$ | $V_{DD}$ + 0.3 | V |
| Analog Input | | 0 | | 11 | V |
| Input Frequency | | | | | |
| Horizontal | | | 15.625 | | kHz |
| Vertical | | | 50 | | Hz (PAL) |
| Horizontal | | | 15.75 | | kHz |
| Vertical | | | 60 | | Hz (NTSC) |
| Output Voltage Levels | | | | | |
| Logic 0 | | $V_{SS}$ | $V_{SS}$ | $V_{SS}$ + 50 mV | V |
| Logic 1 | $I_{OUT}$ = \|10 μA\| | $V_{DD}$ − 50 mV | $V_{DD}$ | $V_{DD}$ | V |
| Output Drive | | | | | |
| Logic 0 | $V_{SS}$ + 2V | −1 | | | mA |
| Logic 1 | $V_{DD}$ − 2V | 1 | | | mA |
| External RC | | | | | |
| C Timing | | | 47 | | pF |
| R1 Timing | | | 1.2 | | kΩ |
| R2 Timing | | | 2.0 | | kΩ |
| C Ramp | | | 33 | | pF |
| R Ramp | | | 475 | | kΩ |
| Input Leakage (Except Pins 2,3,13) | $V_{IN}$ = 12V | | 1 | | μA |
| Input Capacitance | | | 5 | | pF |
| Input Current (Pin 13) | $V_{IN}$ = 0V | | | 80 | μA |

**TABLE I. BAND SELECT CHARACTER DISPLAY**

**PAL System**

| Band I | Band III | UHF | Transmission Option | Display LHS | Display RHS |
|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 2 | 4 |
| 1 | 0 | 1 | 0 | 5 | 12 |
| 1 | 1 | 0 | 0 | 21 | 69 |
| 0 | 1 | 1 | 1 | 2 | S1 |
| 1 | 0 | 1 | 1 | S2 | 20 |
| 1 | 1 | 0 | 1 | 21 | 69 |

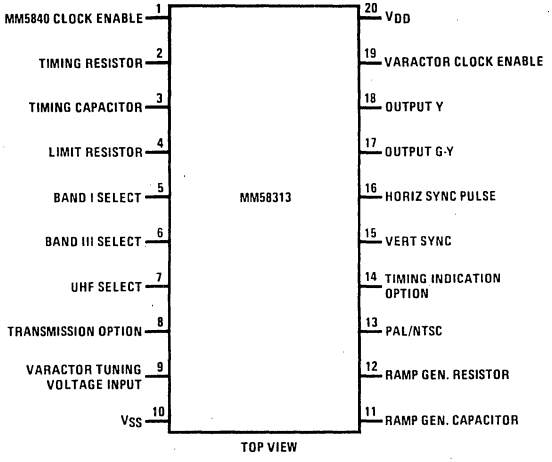
**NTSC Option**

| Band I | Band III | UHF | Transmission Option | Display LHS | Display RHS |
|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 2 | 4 |
| 1 | 0 | 1 | 0 | 5 | 13 |
| 1 | 1 | 0 | 0 | 14 | 83 |

Note: For NTSC system only, channel display is equivalent for both broadcast and cable systems.

## Connection Diagram

Dual-In-Line Package




**Order Number MM58313N**
**See Package 20A**

## Description of Pin Functions

| Pin No. | Name | Function |
|---|---|---|
| 20 | $V_{DD}$ | Positive supply. |
| 10 | $V_{SS}$ | Ground. |
| 16 | Horizontal Sync Input | Positive sync pulse from TV set. |
| 15 | Vertical Sync Input | Positive sync pulse from TV set. |
| 5 | Band I Select Input | Connect to $V_{SS}$ to select required band, hence displaying corresponding band numbers. |
| 6 | Band II Select Input | |
| 7 | Band UHF Select Input | |
| 9 | Varactor Tuning Voltage Input | Tuning voltage from varactor diodes. Indicator position is proportional to this voltage (external resistor divider required). |
| 8 | Transmission Option Input | Connect to $V_{SS}$ for normal broadcast reception or to $V_{DD}$ for cable transmission. |
| 14 | Tuning Indication Option Input | Connect to $V_{SS}$ for 'finger' display or $V_{DD}$ for horizontal bar indication. |
| 3 | Timing Capacitor | Frequency determining components. |
| 2 | Timing Resistor | |
| 4 | Limit Resistor | |
| 12 | Ramp Generator Resistor | Components governing the internally generated ramp voltage. Connect to $V_{SS}$. |
| 11 | Ramp Generator Capacitor | |
| 13 | * PAL/NTSC Option | Connect to $V_{SS}$ for PAL. Connect to $V_{DD}$ for NTSC. |
| 18 | Video Output Y | Active high output used to blank video. |
| 17 | Video Output G-Y | Active high output used to drive color gun. |
| 19 | Varactor Clock Enable | Enables 3.3 MHz oscillator from vertical sync pulse to vertical line 58 ($+V_E$ enable). |
| 1 | MM5840 Clock Enable | Enables 3.3 MHz oscillator when MM5840 O/P is required ($+V_E$ enable). |

* For this pin there is an internal pull up resistor to $V_{DD}$.

# Functional Description

**Operation** (Block Diagram, *Figures 6 and 7* for Timing Diagrams).

Both graticule and character displays are positioned digitally in the horizontal and vertical directions.

**Graticule** *(Figure 1)*

The top of the graticule display is 24 lines down after the vertical flyback pulse. The position is determined by counting the horizontal flyback pulses.

The graticule is 14 lines high and is therefore displayed from line 24 to line 38.

In the horizontal direction the display is positioned by counting pulses from the internal 3.3 MHz osc. A divide by 6 counter is used to derive the 550 kHz required to generate the graticule display.

**Characters** *(Figure 3)*

Provision is made for displaying two digit numbers on both the left hand side and right hand side of the screen. The initial and final characters being coincident with the graticule extremes *(Figure 2)*.

The characters are built up from 42 dots on a $6 \times 7$ matrix. The dot rate is defined by the 3.3 MHz clock, each dot being 0.303 $\mu$s wide and 2 lines high. Spacing between the two digits is 0.606 $\mu$s.

**Tuning Indicator** *(Figure 4 and Figure 5)*

The tuning indicator is derived by comparing the varactor voltage with the ramp signal derived from a capacitor fed with constant current.

A comparator organized as a window detector determines the position and width of the tuning indicator.

The first trip point occurs when the ramp voltage equals that of the varactor input, and the second a set voltage after.

A constant width bar is then generated due to the linear ramp.

Note that an option input permits the tuning indication to appear as a horizontal line. This is achieved by enabling one comparator output only.

**Oscillator***

Three pin oscillator frequency being determined by external timing components. Output capable of driving MM5840, MM5841 series of display chips is available from one of the pins.

**Horizontal Counter**

9-stage dynamic shift register counter utilizing exclusive NOR feedback. This counts the 3.3 MHz pulses and is reset during horizontal sync pulse time.

**Vertical Counter**

9-stage static shift register counter with exclusive NOR feedback. This counts the horizontal sync pulses and is reset during vertical sync pulse time.

**Horizontal Time Slot Decoder**

ROM outputs from which are generated the required decode times. This is used to generate the start/stop commands for the graticule display and to update the character address circuitry with respect to the bank selected.

**Vertical Time Slot Decoder**

ROM used to generate outputs at required vertical decode time.

**Divide-by-Two**

Used to divide the horizontal sync pulses by two; i.e., each vertical decode implies that twice the number of line scans has elapsed.

**Divide-by-Six**

3-stage Johnson counter used to obtain the 550 kHz required to produce the graticule display.

**Band Decoder**

4 to 5-line decoder one output of which is selected according to the band selected and the state of the option input.

**Character Address Circuitry**

Circuitry controlling the character addressed. This is updated at required times so that the required characters will be displayed during the correct time slots with respect to the band selected.

**Character ROM**

ROM organized to produce characters of $6 \times 7$ format.

**Shift Register**

7-bit shift register, 6 bits of character information are loaded at the required horizontal times during each of the 7 vertical decode times for which the characters are displayed. Character information is then clocked out serially at 3.3 MHz.

**Constant Current Ramp Generator**

A current mirror circuit used to charge an external capacitor to produce linear voltage ramp.

**Comparators**

Analog comparator organized so that trip point 1 < trip point 2 where trip point 1 = varactor input voltage between 0V and 9V. A logic 1 output is obtained when trip point 1 < V ramp < trip point 2.

---

* 1. Oscillator is enabled by varactor enable pin (high level) during the time from vertical sync pulse to vertical decode 29.

AND/OR

2. Oscillator is enabled by output from MM5840 pin.

## Functional Description (Continued)

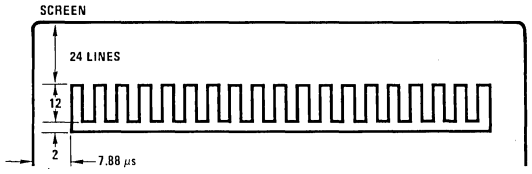
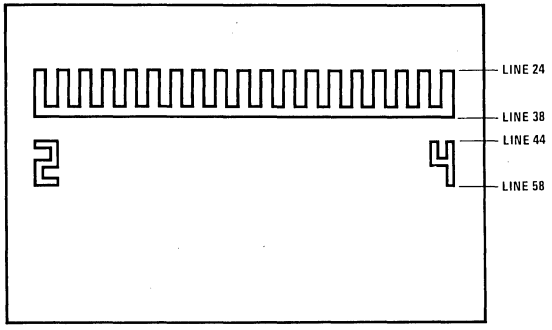



FIGURE 1. Graticule Format and Screen Position




For cable display characters are 2. S2

FIGURE 2. Band I Display

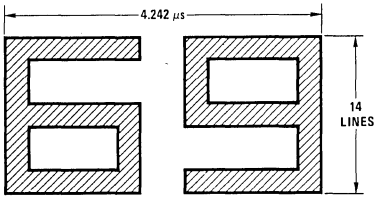



FIGURE 3. Typical Character Format - 6 x 7 Matrix

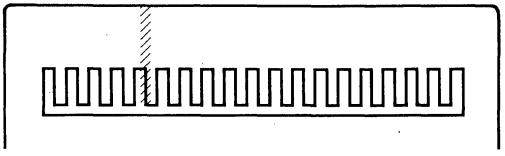


FIGURE 4. Vertical Tuning Indication

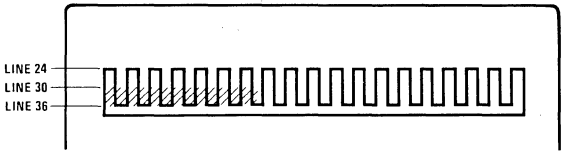



FIGURE 5. Optional Horizontal Tuning Indication

## Functional Description (Continued)

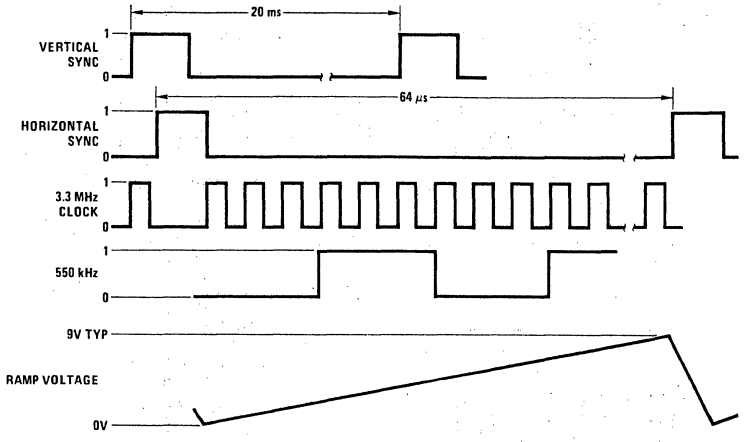



**FIGURE 6. Input and Internally Generated Waveforms**

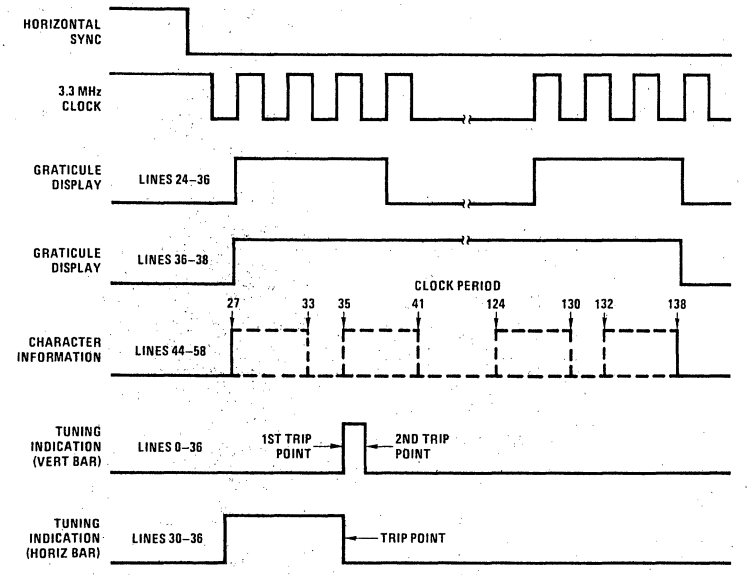



**FIGURE 7. Output Waveforms**

# Games/Calculators

National Semiconductor



# MM5780 Educational Arithmetic Game

## General Description

The MM5780 single-chip educational game was developed using a metal gate, P-channel, enhancement and depletion mode MOS process. It was designed with low end-product cost as the primary objective and is directed toward the educational toy market. Besides the MM5780 as shown in *Figure 1*, requires only a keyboard, "Right" and "Wrong" LED display, a 9V battery and an on/off switch. Keyboard encoding and key debounce circuitry, all clock and timing generation and the capability to drive the two LEDs are all included on-chip and require *no* external discrete components.

The MM5780 educational game was designed to be an arithmetic aid to school age children. Problems are entered into the machine in algebraic form exactly as they are written across a printed page. The student provides the answer or missing factor and when finished, depresses the Test key. "Right" and "Wrong" outputs provide an indication of the results of the test. If wrong, the student trys the problem again. If correct, he can move on to the next problem. Most problems using +, −, x and ÷ can be learned using this machine. The game does not have provisions for remainders in division or negative number entries. A negative result can be entered before the Test key is depressed.

The MM5780 is a low power device which operates directly from a 9V battery. Battery life is estimated to be 10 to 30 hours depending on battery quality and operating schedule.

When the battery voltage falls below an operational level, an internal circuit will disable both indicator outputs; i.e., neither indicator will be on after depression of Test.

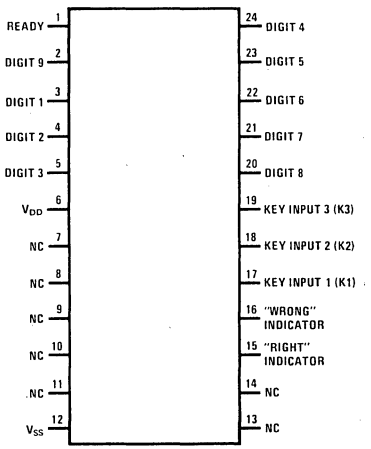
The Ready output signal is used to indicate when the game is performing an operation. It is useful in testing of the device or if interfacing with other logic. Another feature that is important in testing is the capability of reducing the key debounce time from seven word times to four word times by forcing the Digit 7 output high during Digit 9 time.

## Features

- Full 8-digit entry capacity
- Four functions (+, −, x, ÷)
- Convenient algebraic key entry notation
- Floating point input and output
- Chain operations
- Direct 9V battery compatibility; low power
- Direct interface to LED indicators
- No external components required other than keyboard and LED display for complete educational game.
- Overflow and divide-by-zero error indication
- Low battery voltage sensing

## Connection Diagram

Dual-In-Line Package




TOP VIEW

**Order Number MM5780N**
**See Package 22**

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Relative to $V_{SS}$. (All other pins connected to $V_{SS}$.) | $V_{SS}$ + 0.3V to $V_{SS}$ − 12.0 |
| Ambient Operating Temperature | 0°C to +70°C |
| Ambient Storage Temperature | −55°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Operating Voltage Range (Note 1)

$6.5V \leq V_{SS} - V_{DD} \leq 9.5V$

($V_{SS}$ is always defined as the most positive supply voltage.)

## DC Electrical Characteristics

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Operating Supply Current ($I_{DD}$) | $V_{DD} = V_{SS}-9.5V$, $T_A = 25°C$ | | 8.0 | 14.0 | mA |
| Keyboard Scan Input Levels (K1, K2 and K3) | | | | | |
| Logical High Level ($V_{IH}$) | $V_{SS}-6.5V \leq V_{DD} \leq V_{SS}-9.5V$ | $V_{SS}-2.5$ | | | V |
| Logical Low Level ($V_{IL}$) | $V_{DD} = V_{SS}-6.5V$ | | | $V_{SS}-5.0$ | V |
| | $V_{DD} = V_{SS}-9.5V$ | | | $V_{SS}-6.0$ | V |
| Digit Output Levels (Note 1) | | | | | |
| Logical High Level ($V_{OH}$) | $V_{SS}-6.5V \leq V_{DD} \leq V_{SS}-9.5V$ | $V_{SS}-1.5$ | | | V |
| Logical Low Level ($V_{OL}$) | $V_{DD} = V_{SS}-6.5V$ | | | $V_{SS}-6.0$ | V |
| | $V_{DD} = V_{SS}-9.5V$ | | | $V_{SS}-7.0$ | V |
| Indicator Output Current | | | | | |
| Source Current | $T_A = 25°C$ | | | | |
| | $V_{OUT} = V_{SS}-4.5$, $V_{DD} = V_{SS}-6.5V$ | −10.0 | −15.0 | | mA |
| | $V_{OUT} = V_{SS}-4.8$, $V_{DD} = V_{SS}-9.5V$ | | −25.0 | −32.0 | mA |
| Ready Output Levels | | | | | |
| Logical High Level ($V_{OH}$) | $I_{OUT}$ = −0.4 mA | $V_{SS}-1.0$ | | | V |
| Logical Low Level ($V_{OL}$) | $I_{OUT}$ = 10μA | | | $V_{DD}+1.0$ | V |

## AC Electrical Characteristics (Figure 2)

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Word Time | | 0.6 | 1.5 | 5.2 | ms |
| Digit Time | | 70 | 170 | 580 | μs |
| Keyboard Input (K1, K2, K3) High to Low Transition Time After Key Release | $C_{LOAD}$ = 100 pF | | 4 | | μs |
| Ready Propagation Time | $C_{LOAD}$ = 100 pF | | | | |
| Low to High Level ($t_{PDH}$) | | 60 | 140 | 480 | μs |
| High to Low Level ($t_{PDL}$) | | | 0.5 | 1.5 | ms |
| Key Bounce-out Stability Time (The time a keyboard input must be continuously higher than the minimum logical high level to be accepted as a key closure, or continuously lower than the maximum logical low level to be accepted as a key release.) | | 4.2 | 10.5 | 35.0 | ms |
| Calculation Time for 99999999 ÷ 1 = 99999999 | | 90 | 220 | 765 | ms |

**Note 1:** The internal low battery voltage sensing circuit will disable both indicator outputs when $V_{SS}-V_{DD}$ falls below a safe operating voltage. That voltage may be less than or greater than 6.5V depending on process variables; the MM5780 will have been tested to operate correctly for any voltage less than 9.5V at which an indicator output is enabled.

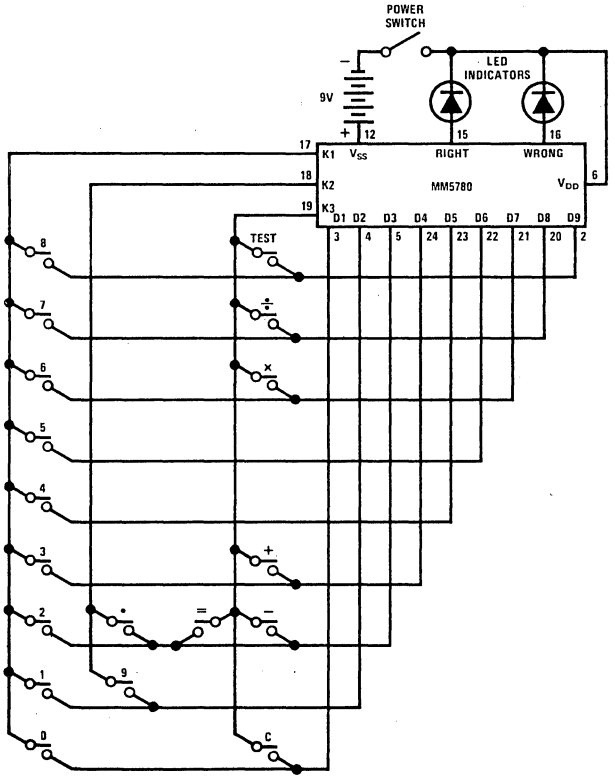



**FIGURE 1. Complete Game**

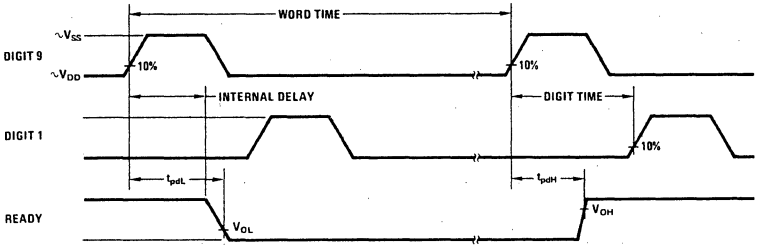



**FIGURE 2. Output Timing**

## KEY INPUT BOUNCE AND NOISE REJECTION

The MM5780 game chip is designed to interface with low cost keyboards, which are often the least desirable from a noise and false entry standpoint.

A key closure is sensed by the game chip when one of the Key Input Lines, K1, K2 or K3 are forced more positive than the Logical High Level specified in the Electrical Specifications. At the instant of closure, an internal "Key Bounce-out Stability Time" counter is started. Any significant voltage perturbation occurring on the switched key input during timeout will reset the timer. Hence, a key is not accepted as a valid entry until noise or ringing has stopped and the stability time counter has timed out. Noise that persists will inhibit key entry indefinitely. Key release is timed in the same manner.

One of the popular types of low cost keyboards available, the elastomeric conductor type, has a key pressure versus contact resistance characteristic that can generate continuous noise during "teasing" or low pressure key depressions. The MM5780 defines a series contact resistance up to 50 kΩ as a valid key closure, providing an optimum interface to that type of keyboard as well as more conventional types.

### Error Conditions

In the event of an overflow or divide-by-zero the "Wrong" light will come on and remain on until a Clear key is depressed. Normally the indicator lights are activated only after depression of the TEST key.

## KEY OPERATIONS

### Clear Key

The Clear key clears all registers to zero and places the machine in an idle state.

### Number Entries

First entry clears the entry register and enters the number into the least significant digit (LSD) of the entry register and extinguishes the indicator lights. Second through eighth entry shifts the entry register left one digit and enters the number into the LSD. The ninth and subsequent entries, are ignored and no error condition is generated. Because only seven positions are allowed to follow the decimal point, the eighth and subsequent entries after a decimal point entry are ignored.

### Decimal Point

Depression results in a decimal point entry into the entry register.

### Add, Subtract, Multiply or Divide Keys

First depression after a number entry will terminate the entry, perform the previously recorded operation, if any, and record the function key depressed as the next operation to be performed after another number entry. Subsequent depressions of any function key, without an interceding number or decimal point entry will supersede the previous function as the next to be performed. If a function key is depressed after an equal key, the result of the operation will be re-entered and the function key depressed will become the next operation to be performed after a number entry is followed by another function key (including equal).

### Equal

First depression after a number entry will terminate the entry, perform the previously recorded operation and record the fact that an equal key has been depressed. Depression after the add, subtract or divide keys, without an interceding number or decimal point entry, will be ignored. After a multiply key, the number in the entry register will be squared.

### Resultant Entries

Results are entered as number entries after an equal key and before the Test key. Results are assumed positive and a plus key should *not* be entered prior to the resultant. Negative results must be preceded by a minus key.

### Test

The Test key is used to terminate computations and to initiate a test of the student's answer versus the game's answer. If the answers match, the "Right" indicator is enabled, otherwise the "Wrong" indicator is enabled. If the results are incorrect the problem must be worked again from the beginning.

**TABLE I. Ready Signal Description**

| GAME FUNCTION | READY SIGNAL |
|---|---|
| Idle | *READY* is quiescently at a Logical High Level ($\sim V_{SS}$). |
| Key Entry and Functional Operation | When a key is depressed, the bounce-out stability timer is initiated. *READY* remains high until the bounce-out time is completed and the key is entered, at which time it changes to a Logical Low Level ($\sim V_{DD}$). |
| Key Release and Return to Idle | *READY remains low until key release is debounced and the game returns to the idle state. The low to high transition signals the return to idle.* |

**TABLE II. Indicator Truth Table**

| GAME CONDITION | INDICATOR OUTPUT | |
|---|---|---|
| | PIN 15 | PIN 16 |
| Test was last key depressed with correct answer entered. | HIGH | LOW |
| Test was last key depressed with incorrect answer entered or the problem has resulted in an error or overflow condition. | LOW | HIGH |
| Any key other than Test was last depressed and calculator is not in an error or overflow condition. | LOW | LOW |
| Clear was last key depressed. | LOW | LOW |
| The battery supply voltage has fallen below a valid operating voltage for the MM5780. Independent of keys depressed. | LOW | LOW |

## sample problems

**I. Simple Addition:** 4 + 5 = ?

| Key | Display | Comments |
|---|---|---|
| C | | |
| C | NONE | Clear necessary on power-up |
| 4 | NONE | |
| + | NONE | |
| 5 | NONE | |
| = | NONE | |
| 8 | NONE | Answer supplied |
| TEST | WRONG | Wrong answer |
| 4 | NONE | Indicator goes out |
| + | NONE | |
| 5 | NONE | |
| = | NONE | |
| 9 | NONE | |
| TEST | RIGHT | |

**II. Missing Factor Addition:** 6 + ? = 11

| Key | Display | Comments |
|---|---|---|
| 6 | NONE | Indicator goes out |
| + | NONE | |
| 5 | NONE | Missing factor supplied |
| = | NONE | |
| 11 | NONE | |
| TEST | RIGHT | |

**III. Subtraction:** 4 − 7 = ?

| Key | Display | Comments |
|---|---|---|
| 4 | NONE | Indicator goes out |
| − | NONE | |
| 7 | NONE | |
| = | NONE | |
| − | NONE | |
| 3 | NONE | Negative answer supplied |
| TEST | RIGHT | |

**IV. Multiplication:** 7 x 3 = ?

| Key | Display | Comments |
|---|---|---|
| 7 | NONE | Indicator goes out |
| x | NONE | |
| 3 | NONE | |
| = | NONE | |
| 21 | NONE | Answer supplied |
| TEST | RIGHT | |

## sample problems (con't)

V. **Missing Factor Multiplication:** 6 x ? = 12

| Key | Display | Comments |
|---|---|---|
| 6 | NONE | Indicator goes out |
| x | NONE | |
| 3 | NONE | Missing factor supplied |
| = | NONE | |
| 12 | NONE | |
| TEST | WRONG | Incorrect |
| 6 | NONE | Indicator goes out |
| x | NONE | |
| 2 | NONE | Missing factor supplied |
| = | NONE | |
| 12 | NONE | |
| TEST | RIGHT | |

VI. **Division:** 15 ÷ 3 = ?

| Key | Display | Comments |
|---|---|---|
| 15 | NONE | Indicator goes out |
| ÷ | NONE | |
| 3 | NONE | |
| = | NONE | |
| 5 | NONE | Answer supplied |
| TEST | RIGHT | |

VII. **Complex Chain:** (6 + 2 − 10) x 3 = ?

| Key | Display | Comments |
|---|---|---|
| 6 | NONE | Indicator goes out |
| + | NONE | |
| 2 | NONE | |
| − | NONE | |
| 10 | NONE | |
| x | NONE | |
| 3 | NONE | |
| = | NONE | |
| − | NONE | |
| 6 | NONE | Negative answer supplied |
| TEST | RIGHT | |

National Semiconductor

Games Calculators
PRELIMINARY

# MM57455 Advanced Educational Arithmetic Game

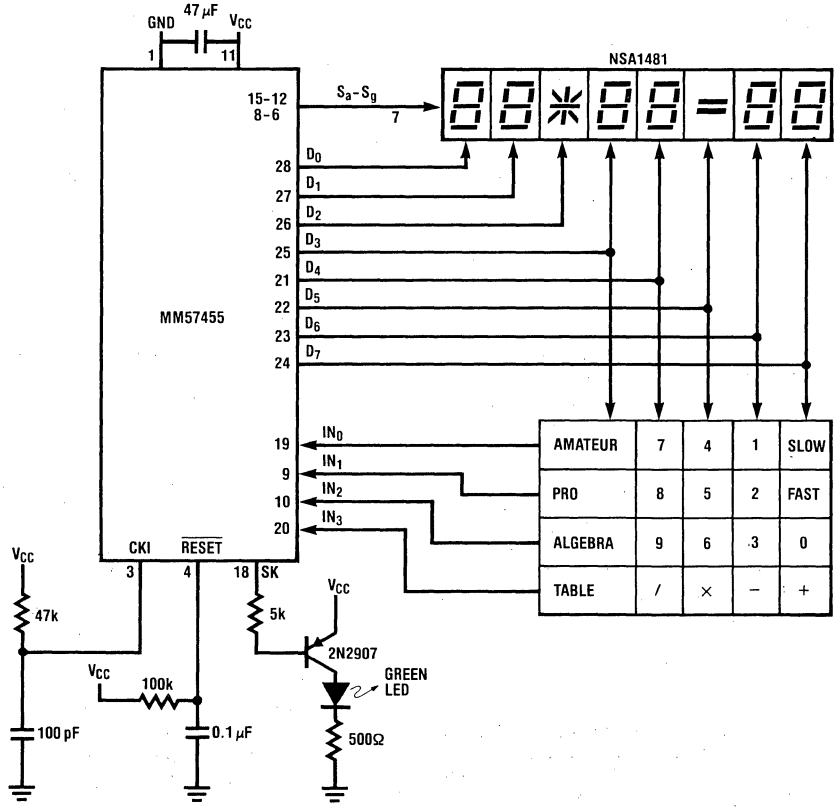
## General Description

*Figure 1* contains an electrical diagram of a complete teaching game system.

## Features

- Produces add, subtract, multiply, and divide problems which teach basic arithmetic
- 6,562 different problems are produced
- Problems are generated randomly and automatically
- Automatic entry, no "ENTER" key is needed
- Green LED lights when the correct answer is entered
- If the wrong answer is entered, "E" appears in the display and the user gets a second try
- If the user answers incorrectly on both tries, the correct answer is flashed in the display
- Internal timer gives the user about 10 seconds to answer. If he doesn't answer, the problem is counted wrong
- Ten problems in each problem set
- Number of problems correct appears in the display at the end of a problem set, with the green LED flashing
- "TABLE" button causes non-random problems to be generated
- "COMPLEX" button causes algebra-type problems to be generated
- "AMATEUR/PRO" buttons select easy/hard addition and subtraction problems
- "NORMAL/FAST" buttons select 10 or 3 seconds to answer a problem
- Automatically begins game on power "ON"
- Low system cost (*Figure 1*)

## Electrical Diagram



## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Relative to GND1 | −0.3V to +10V |
| Ambient Operating Temperature | 0°C to +70°C |
| Ambient Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 Seconds) | 300°C |
| Power Dissipation | 0.75 Watt at 25°C<br>0.4 Watt at 70°C |

*"Absolute Maximum Ratings" indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.*

## DC Electrical Characteristics $0°C \leq T_A \leq 70°C$, $4.5V \leq V_{CC} \leq 9.5V$, unless otherwise specified

| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| Operating Voltage ($V_{CC}$) | | 4.5 | | 9.5 | V |
| Operating Supply Current | $V_{CC} = 5V$, $T_A = +25°C$<br>(all inputs and outputs open) | | | 8 | mA |
| Input Voltage Levels | | | | | |
| OSC IN, $\overline{RESET}$ | | | | | |
| Logic High ($V_{IH}$) | | $0.7V_{CC}$ | | | V |
| Logic Low ($V_{IL}$] | | | | 0.6 | V |
| $\overline{RESET}$ Hysteresis | | 1.0 | | | V |
| All Other Inputs | | | | | |
| Logic High ($V_{IH}$) | $V_{CC} = 9.5V$ | 3.0 | | | V |
| Logic High ($V_{IH}$) | $V_{CC} = 5V \pm 10\%$ | 2.0 | | | V |
| Logic Low ($V_{IL}$) | | | | 0.8 | V |
| Output Current Levels | | | | | |
| Ouptut Sink Current | | | | | |
| $D_0$-$D_7$ ($I_{OL}$) | $V_{CC} = 9.5V$, $V_{OL} = 1.0V$ | 30 | | 150 | mA |
| | $V_{CC} = 4.5V$, $V_{OL} = 1.0V$ | 15 | | 70 | mA |
| $S_a$-$S_g$ ($I_{OL}$) | $V_{CC} = 9.5V$, $V_{OL} = 1.0V$ | 2.0 | | 9.0 | mA |
| | $V_{CC} = 4.5V$, $V_{OL} = 1.0V$ | 1.0 | | 4.5 | mA |
| Output Source Current | | | | | |
| $S_a$-$S_g$ ($I_{OH}$) | $V_{CC} = 9.5V$, $V_{OH} = 2.0V$ | −3.0 | | −30 | mA |
| | $V_{CC} = 6.0V$, $V_{OH} = 2.0V$ | −3.0 | | −20 | mA |

## Functional Description

### Display Configuration

The special LED display used with the MM57455 displays any of the 4 symbols "+", "−", "×", "/" in the third digit position. An "=" is displayed in the sixth digit position. The remaining 6 digits are normal 7-segment numeral displays.

### Power "ON"

Upon powering "ON" the MM57455, it begins displaying the sysmbols "+", "−", "×", "/", "+",. . . one after another, each lasting about ½ second. This indicates that it is at the beginning of a "problem set" and ready to accept a function key input.

### Key Operations

**Function Keys, "+", "−", "×", "/"**
One of these keys is depressed to begin a problem set. After pressing one of these keys, a randomly generated problem appears in the display. The problem is either "+", "−", "×", "/", depending on the key that was pressed.

**Number Keys, "0-9"**

These keys are used to enter answers to problems. After a problem appears in the display, the user has 2 tries to answer it correctly.

### Green LED

If the user keys in the correct answer to a problem, the green LED lights up immediately for 1½ seconds. Then a new problem appears.

### Incorrect Answer Indicator

If the user keys in a wrong answer to a problem, his answer disappears in the display and an "E" appears.

### Second Try

If the user answers incorrectly, he gets a second try. When the "E" appears (indicating that the answer is wrong), he types in his second try. Again, the green LED lights if correct, and an "E" appears if wrong.

# Functional Description (cont'd)

### Internal Timer

The MM57455 has an internal timer which allows the user 10 seconds to answer a problem. If he doesn't answer in 10 seconds, an "E" appears in the display, indicating a wrong answer. The user then gets a second try and again must answer within 10 seconds.

### Flashing of a Correct Answer

In the user answers wrong on both tries, the correct answer flashes in the display. Then the next problem appears.

### Ten Problems per Problem Set

New problems appear one after another until 10 problems have been done.

### Score at End of Problem Set

After 10 problems are done, the number of problems the user got right appears in the display, and the green LED flashes. Only first try answers are counted correct. After 16 flashes, the MM57455 again displays "+", "−", "×", "/", "+",... and is ready for another function key entry.

### "TABLE" Key

If the "TABLE" key is depressed just before pressing a function key at the start of a problem set, table problems will appear, with a random table digit.

**Example:** press "TABLE" ×
and these problems may appear:

6 × 1 =
6 × 2 =
6 × 3 =
.
.
.
6 × 10 =

A non-random table digit can be selected by depressing the desired number (1-10) just before pressing a function button at the start of a problem set.

**Example:** press 9 ×
and these problems will appear:

9 × 1 =
9 × 2 =
9 × 3 =
.
.
.
9 × 0 =

### "ALGEBRA" Key

If the "ALGEBRA" key is depressed just before pressing a function key at the start of a problem set, algebra-type problems will be displayed (the answer is present and one of the factors is blank, as: (15 + = 21). The user must enter the missing factor. (Note. Both "ALGEBRA" and "TABLE" buttons may be pressed before pressing a function key. This will cause algebra-type table problems to be displayed.) The order of depression is unimportant; i.e., "ALGEBRA" or "TABLE" may be pressed first.

### "AMATEUR/PRO" Keys

These keys select easy ("AMATEUR") or hard ("PRO") addition and subtraction problems. Easy means sum < 30 and difference < 20. Hard means sum < 100 and difference < 100.

When power is turned "ON", the machine is in easy ("AMATEUR") mode.

### "NORMAL/FAST" Keys

These keys are used to select 10 second ("NORMAL") or 3 second ("FAST") answer time.

When power is turned "ON", the machine is in the 10 second ("NORMAL") mode.

National Semiconductor



# MM57459 8-Digit LED Direct-Drive Memory Calculator

## General Description

The single-chip MM57459 calculator was developed using an N-channel enhancement and depletion mode MOS/LSI technology with a primary object of low end-product cost. A complete calculator as shown in *Figure 1* requires only the MM57459 calculator chip, and X-Y matrix keyboard, an NSA1188 LED display and a 9V battery.

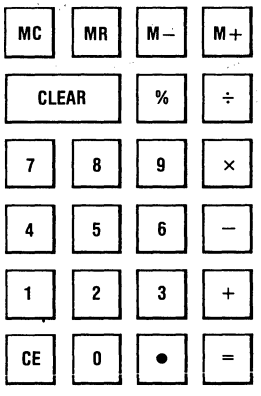
Keyboard decoding and key debounce circuitry, all clocks, and timing generators, power-on clear, and 7-segment output display decoding are included on-chip, and require no external components. Segments and digits can usually be driven directly from the MM57459, as the segments source up to 30 mA max. peak current and the digit drivers sink 30 mA min.

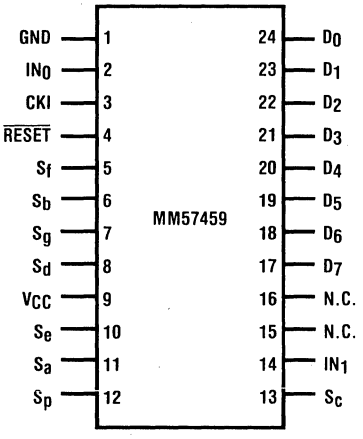
Leading zero suppression and a floating negative sign allow convenient reading of the display and conserve power. Up to 8 digits for positive numbers and 7 for negative numbers can be displayed, with the negative sign displayed in the left-most position.

## Features

- 8 Digits with four key memory (M+, M−, MR, MC)
- Low voltage operation (single power supply)
- Direct interface with digits and segments of LED display
- Percent function with add-on/discount
- Automatic constant on all five functions
- Floating minus sign
- Leading zero suppression
- Internal clock generator
- Internal encoding for keyboard inputs
- Internal debouncing for keyboard inputs
- Display flash in calculator overflow state

## Typical Keyboard and Connection Diagram




**Top View**
**Order Number MM57459N**
**See Package 22**

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Relative to GND1 | −0.3V to +10V |
| Ambient Operating Temperature | 0°C to +70°C |
| Ambient Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 Seconds) | 300°C |
| Power Dissipation | 0.75 Watt at 25°C |
| | 0.4 Watt at 70°C |

*Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.*

## DC Electrical Characteristics $0°C \leq T_A \leq 70°C$, $4.5V \leq V_{CC} \leq 9.5V$, unless otherwise specified

| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| Operating Voltage ($V_{CC}$) | | 4.5 | | 9.5 | V |
| Operating Supply Current | $V_{CC} = 5V$, $T_A = +25°C$ (all inputs and outputs open) | | | 8 | mA |
| Input Voltage Levels | | | | | |
| CKI, $\overline{RESET}$ | | | | | |
| Logic High ($V_{IH}$) | | $0.7\,V_{CC}$ | | | V |
| Logic Low ($V_{IL}$) | | | | 0.6 | V |
| $\overline{RESET}$ Hysteresis | | 1.0 | | | V |
| All Other Inputs | | | | | |
| Logic High ($V_{IH}$) | $V_{CC} = 9.5V$ | 3.0 | | | V |
| Logic High ($V_{IH}$) | $V_{CC} = 5V \pm 10\%$ | 2.0 | | | V |
| Logic Low ($V_{IL}$) | | | | 0.8 | V |
| Output Current Levels | | | | | |
| Ouptut Sink Current | | | | | |
| $D_0$-$D_3$ ($I_{OL}$) | $V_{CC} = 9.5V$, $V_{OL} = 1.0V$ | 30 | | 150 | mA |
| | $V_{CC} = 4.5V$, $V_{OL} = 1.0V$ | 15 | | 70 | mA |
| $S_a$-$S_g$, $S_p$ ($I_{OL}$) | $V_{CC} = 9.5V$, $V_{OL} = 1.0V$ | 2.0 | | 9.0 | mA |
| | $V_{CC} = 4.5V$, $V_{OL} = 1.0V$ | 1.0 | | 4.5 | mA |
| Output Source Current | | | | | |
| $S_a$-$S_g$, $S_p$ ($I_{OH}$) | $V_{CC} = 9.5V$, $V_{OH} = 2.0V$ | −3.0 | | −30 | mA |
| | $V_{CC} = 6.0V$, $V_{OH} = 2.0V$ | −3.0 | | −20 | mA |

# 1. Key Definition

[0] - [9] - [•]

The first number key in a sequence will clear the display and enter the digit in the LSD of the display. Successive entries will shift the display left and enter data in the LSD. The first decimal point entered is effective. An attempted entry of more than 8 digits or 7 decimal places will be ignored.

### [C] — Clear

Clears the display and constant registers, and the result overflow indicator. Memory register is not affected by key. In the memory overflow condition, this key is operative as a clear memory key.

### [CE] — Clear Entry

Clears the display of a number entry. In the result overflow mode, this key resets the overflow condition and allows calculation to continue; however this key is inoperative during memory overflow.

### [MC] — Memory Clear

Clears the memory.

### [+] — Plus

Stores an addition operation and performs a possible preceding operation. Successive depression of the plus key will not affect the display.

### [−] — Minus

Stores a subtract operation and performs a possible preceding operation. Repeat subtraction by the minus key will not be possible. If this is depressed after a % , + , or = key, subtraction becomes the pending operation. Immediately following a × or ÷ key, this acts as a data entry and −0. is displayed.

### [×] — Multiply

Operates the same as the plus key except that a multiply command is stored. Successive depression of the multiply key will not alter the display.

### [÷] — Divide

Operates the same as the plus key except that a divide command is stored. Successive depression of the divide key will not alter the display.

### [=] — Equal

Executes any previous operation and maintains that operation for possible use in the implied constant mode. The first factor entered for multiplication and the second factor entered for division, subtraction, and addition, are retained for the constant operation. Completes the add-on or discount mode when used following the % key. The first depression of the equal key immediately following a + or − key will not alter the display.

### [%] — Percent

The purpose of the percent key is to allow for the calculation of add-on and discount. Determination of add-on requires the principal amount to be the first enter followed by the + or × key, with the percentage being the second entry. Depression of the percent key yields the amount to be added-on, such as tax or interest. Depression of the = key adds this amount to be principal. Discount is determined in a similar manner using the − key (× and − keys). In the constant mode, new percentages to be added-on may be entered while retaining the principal amount.

### [MR] — Memory Recall

Transfers the contents of the memory register into the display register. Memory is retained except in the memory overflow condition. In this case, memory is cleared and its previous contents are displayed in the result overflow mode.

### [M+] — Memory Plus

Add the current display to the contents of memory. M+ will termniate a number entry.

### [M−] — Memory Minus

Subtracts current display from the contents of memory. M− will terminate a number entry.

# 2. Error Conditions

### Result Overflow

If the result in absolute value exceeds $10^8 - 1$, the display will flash, and only the C and CE keys are operative.

### Memory Overflow

If a M+ or M− operation causes the contents of memory to exceed the above value, the display will flash. In this overflow condition, only the C key is operative.

# 3. Operation Characteristics

### Data Entry

Entry is always floating. On data entry, the data will be right hand justified with the last digit entered always appearing in the least significant digit position. The display register will left shift the display one digit as each new digit is entered.

### Data Output

The output data as a result of a calculation will be right hand justified such that trailing insignificant zeros after the decimal are not displayed. Numbers less than one (1) will be displayed with one leading zero (0.25 for example). Numbers greater than one (1) will not display zeros to the left of the most significant digit.

## Output Display

The output segments are fully decoded for standard seven-segment display. The digit outputs are multiplexed with the segment scan to provide the output.

## Digit and Segment Buffers

The segment buffers provide constant drop and operate in conjunction with the constant current digit buffers to provide display current.

## Constant Operation

The MM57459 has an implied constant mode of operation on +, −, ×, ÷, and % operations. The constant calculation is performed automatically by the = key, % key, or % = keys without a constant switch. The second operand is treated as the constant for add, subtract, and divide and the first operand is the constant for multiplication.

For A ± B%-type calculations, the first operand is treated as the constant with the percentage displayed with the proper sign.

## Decimal Alignment

The results of addition or subtraction remain aligned to the preceding entry having the most decimal places unless a right shift is needed to keep the eight most significant digits (in which case the least significant decimal digits are lost).

## Display Font

The following table shows the required segment outputs as a function of the display. In the truth table, the symbol • is used to indicate a selected segment.

| Character | Display | SA | SB | SC | SD | SE | SF | SG | SP |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | • | • | • | • | • | • | | |
| 1 | 1 | | • | • | | | | | |
| 2 | 2 | • | • | | • | • | | • | |
| 3 | 3 | • | • | • | • | | | • | |
| 4 | 4 | | • | • | | | • | • | |
| 5 | 5 | • | | • | • | | • | • | |
| 6 | 6 | • | | • | • | • | • | • | |
| 7 | 7 | • | • | • | | | | | |
| 8 | 8 | • | • | • | • | • | • | • | |
| 9 | 9 | • | • | • | • | | • | • | |
| Minus Sign | − | | | | | | | • | |
| Dec. Pt. | . | | | | | | | | • |

RESULT OVF: THE DISPLAY WILL FLASH.
MEMORY OVF: THE DISPLAY WILL FLASH.

## Floating Minus Sign

When displaying a negative number the minus indication will be located one digit to the left of the MSD display.

The results of multiplication and division are completely right justified such that only the most significant digits are displayed (the digits not displayed will be truncated).The C key resets decimal alignment.

## Successive Operations

Only the last operation entered is performed unless a − entry follows a × or ÷ which sets up the calculator for numeric entry only.

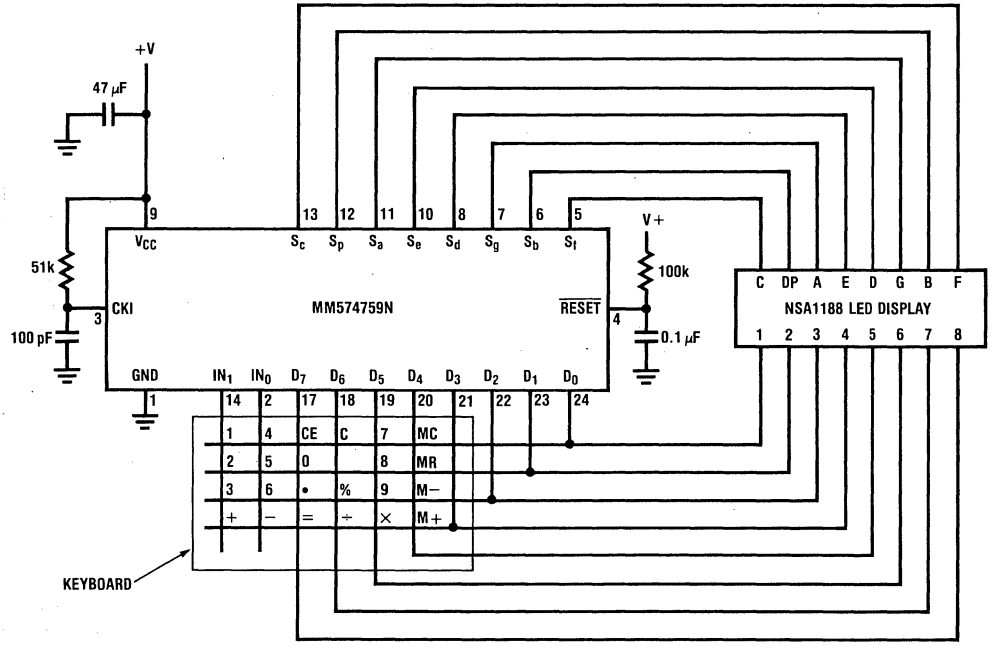



Figure 1. Typical Calculator Application

# Telecommunications

National Semiconductor

Telecommunications

# MM5393, MM5394, MM53143, MM53144 Push Button Pulse Dialer Circuits

## General Description

The MM5393, MM5394, MM53143 and MM53144 are low threshold voltage, ion-implanted, metal-gate CMOS integrated circuits that convert pushbutton inputs into a series of pulses to simulate a telephone rotary dial. Pushbutton inputs require the use of a simple, low cost single contact calculator type keypad. An inexpensive R/C oscillator network is used as the frequency reference. Storage is provided for 21 digits. A redial feature via use of the # key is included. An interdigit pause can be externally selected as either 420 or 840 ms. A mute output is provided to mute receiver noise during outpulsing. No muting occurs during the interdigit pause, thereby allowing the user to hear any busy or error condition arising during the call. The MM5393 and MM53143 provide a pacifier tone of 600 Hz every time a key is depressed. The MM5393 and MM5394 provide a 1.6:1 break/make ratio. The MM53143 and MM53144 provide a 2:1 break/make ratio.

## Features

- Direct line powered operation
- Low voltage operation to 2V
- Low cost R/C oscillator
- Single contact keypad
- 21-digit storage
- Selectable interdigital pause
- Redial of last number
- 600 Hz tone (available in MM5393 and MM53143)

## Block and Connection Diagrams

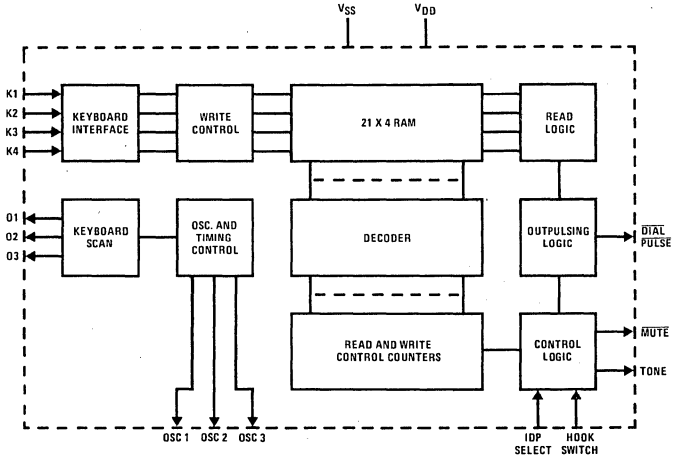
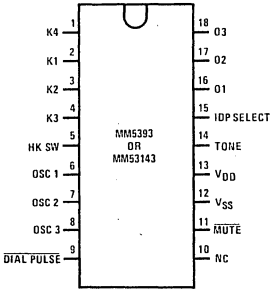



FIGURE 1

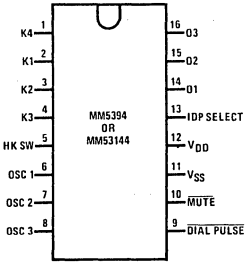
Dual-In-Line Package




TOP VIEW

**Order Number MM5393J or MM53143J**
**See Package 12**

Dual-In-Line Package




TOP VIEW

**Order Number MM5394J or MM53144J**
**See Package 11**

FIGURE 2

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ −0.3V to $V_{DD}$ +0.3V |
| Operating Temperature Range | −30°C to +70°C |
| Storage Temperature Range | −55°C to +150°C |
| $V_{DD}-V_{SS}$ | 6.5V Max |
| Lead Temperature (soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating temperature range, $V_{SS}$ = Gnd, $2V \leq V_{DD} \leq 5.5V$

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Input Voltage Levels at IDP Select, Hook Switch, K1–K4 | | | | | |
| Logical "1" | | $V_{DD}$−0.25 | | $V_{DD}$ | V |
| Logical "0" | | $V_{SS}$ | | $V_{SS}$+0.25 | V |
| Input Pull-Up Resistor Currents at K1–K4, Source | $V_{DD}$ = 3V, $V_{IN}$ = $V_{SS}$ | | 1 | 3 | μA |
| Input Pull-Down Resistor Current at HK SW, Sink | $V_{DD}$ = 3V, $V_{IN}$ = 3V | | 1.5 | 3 | μA |
| Keypad Contact Resistance | | | | 1 | kΩ |
| Output Current Levels | | | | | |
| $\overline{\text{Dial Pulse}}$ | | | | | |
| Logical "1", Source | $V_{DD}$ = 3V, $V_{OUT}$ = $V_{DD}$ −0.9 | 150 | | | μA |
| Logical "0", Sink | $V_{DD}$ = 3V, $V_{OUT}$ = $V_{SS}$ +0.9 | 150 | | | μA |
| $\overline{\text{Mute}}$ | | | | | |
| Logical "1", Source | $V_{DD}$ = 3V, $V_{OUT}$ = $V_{DD}$ −0.9 | 100 | | | μA |
| Logical "0", Sink | $V_{DD}$ = 3V, $V_{OUT}$ = $V_{SS}$ +0.9 | 100 | | | μA |
| Tone | | | | | |
| Logical "1", Source | $V_{DD}$ = 3V, $V_{OUT}$ = $V_{DD}$ −0.5 | 10 | | | μA |
| Logical "0", Sink | $V_{DD}$ = 3V, $V_{OUT}$ = $V_{SS}$ +0.5 | 10 | | | μA |
| 01, 02, 03 | | | | | |
| Logical "1", Source | $V_{DD}$ = 3V, $V_{OUT}$ = $V_{DD}$ −0.5 | 20 | | | μA |
| Logical "0", Sink | $V_{DD}$ = 3V, $V_{OUT}$ = $V_{SS}$ +0.5 | 150 | | | μA |
| Supply Current | $V_{DD}$ = 3.3V, Osc Freq = 20 kHz | | | 100 | μA |
| | $V_{DD}$ = 5.5V, "ON Hook" Osc Stopped | | | 5 | μA |
| Outpulsing Frequency | Osc = 20 kHz | 9 | | 11 | Hz |

## Functional Description

A block diagram of the MM5393, MM5394, MM53143 and MM53144 integrated circuit is shown in *Figure 1* and package connection diagrams for the 2 package options are shown in *Figure 2*.

**Oscillator (Pins 6, 7, and 8):** The time base for the pulse dialer integrated circuit is an R/C-controlled oscillator like that shown in *Figure 3,* typically tuned to 20 kHz by the R1 and C1 combination. Stability of ±10% of typical frequency can be maintained over the voltage range 3.0 – 5.5V and temperature range –30°C to +70°C. At fixed voltage and temperature, part to part variation is less than 5%.

This clock is successively divided to derive the necessary timing for outpulsing and interdigit pause.

**Keyboard (Pins 1–4 and 16–18 or 14–16):** The MM5393, MM5394, MM53143 and MM53144 utilize an inexpensive single contact (Form A, *Figure 7*) keypad. A valid key closure is recorded when a single row ($K_X$ input) is connected to a single column ($O_Y$ input). Key closures are protected from contact bounce for 5 ms.

**$\overline{\text{Dial Pulse}}$ Output (Pin 9):** The $\overline{\text{Dial Pulse}}$ output drives an external bipolar transistor that sequentially opens (breaks) the telephone loop a number of times equal to the input digit selected. For example, key 5 will generate 5 loop current breaks. The break/make ratio of the MM5393 and MM5394 is 1.6:1.0 (i.e. 61.5%:38.5%). The break/make ratio of the MM53143 and MM53144 is 2.0:1.0 (i.e. 67%:33%).

**IDP Select (Pin 15 or 13):** The IDP select input is used to select an interdigit separation of either 420 ms (logic "0" = $V_{SS}$) or 840 ms (logic "1" = $V_{DD}$). An interdigit delay precedes the first digit outpulse sequence.

**$\overline{\text{Mute}}$ (Pin 11 or 10):** The $\overline{\text{Mute}}$ output is used to drive an external bipolar transistor that is used to mute the receiver during the outpulse period. System timing between key closure, mute and dial pulse is shown by the timing diagram in *Figure 4.*

**Tone (Pin 14 MM5393 and MM53143 Only):** The MM5393 and MM53143 provide a tone output to provide audio feedback to the user. The output is a 600 Hz tone that requires an external bipolar driver to activate the telephone receiver.

**Hook Switch Input (Pin 5):** The function of the hook switch input is to properly initialize the circuitry for proper memory and redial operation. In the "ON Hook", logic "0" or $V_{SS}$ condition, the hook switch input

a. Stops the 20 kHz oscillator

b. Sets the memory pointer back to digit 1

c. Clamps the dial pulse and mute outputs to logic "1" or $V_{DD}$

d. Resets all control logic

When the telephone is taken "OFF-Hook", this input must be taken to logic "1" or $V_{DD}$ to release the oscillator and enable the memory and various outputs. For a non-redial application it is necessary to provide an RC delay of approximately 10 $\mu$s to the hook switch input in order to provide a proper power-on clear sequence.

Schematic diagrams for use of the MM5393, MM5394, MM53143 and MM53144 in typical applications are shown in *Figures 5 and 6*.

**Redial Feature**

Pushbutton inputs are accepted at an asynchronous rate. If only 1 key is detected for 5 ms, the decoded key will be loaded into a first-in-first-out memory and outpulsing of the correct number of pulses will immediately begin. After the first digit has been completed, outpulsing will cease unless another key has been entered. This allows use in a PBX system to insure receipt of a dial tone after an access code has been entered and before entering the remainder of the number. If the call was not successful, it can be redialed at a later time by pressing the redial (#) key. If an access code is required, as in a PBX system, it can be manually entered, the dial tone established, and then the redial key pushed to automatically dial the remainder of the number. Only 1 key can be entered before pushing the redial key.

An example of this operation is shown here:

| | KEY INPUTS | OUTPULSES | MEMORY |
|---|---|---|---|
| First Try | 9 P 4087375000 | 94087375000 | 94087375000 |
| Second Try | 9 P # | 94087375000 | 94087375000 |
| Third Try | 9 P # | 94087375000 | 94087375000 |

Where P implies a user pause

## Functional Description (Continued)

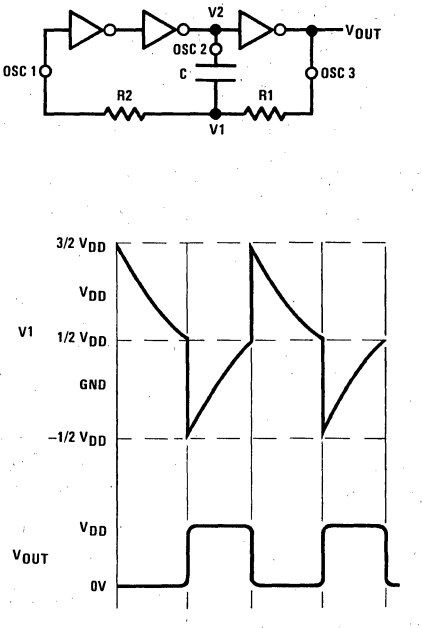



FIGURE 3. Three Gate Oscillator and Waveforms

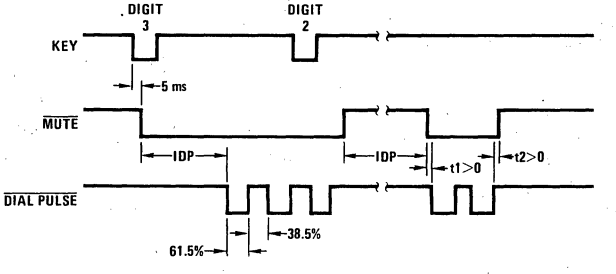
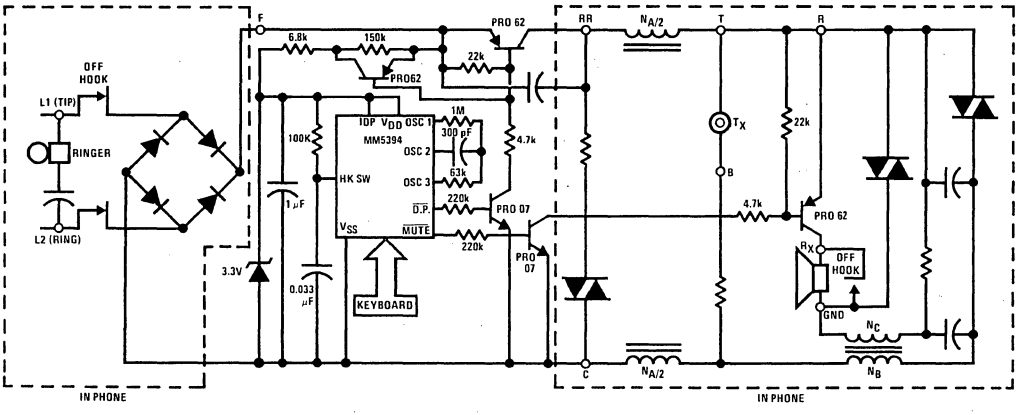


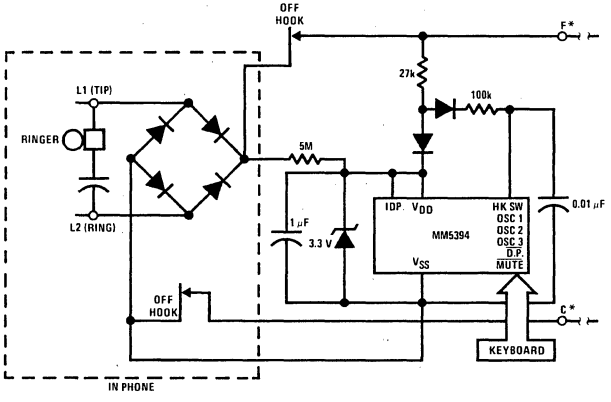

FIGURE 4. Output Timing Waveforms

## Functional Description (Continued)




**Note 1:** No redial.
**Note 2:** Non-valued parts included in instrument.
**Note 3:** Letters refer to instrument terminals.

**FIGURE 5. Typical Application of MM5394 in Type 500D Telephone**




* Remainder of system is same as *Figure 5*.

**FIGURE 6. Typical Application of MM5394 Using Redial Feature**

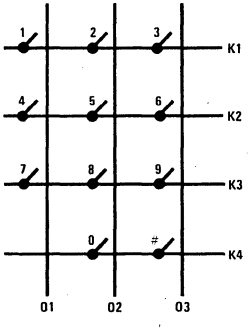



**FIGURE 7. Keypad Matrix**

National Semiconductor

Telecommunications

# MM5395, MM53125 DTMF (TOUCH TONE®) Generators

## General Description

The MM5395 and MM53125 are low threshold voltage, ion-implanted, metal-gate CMOS integrated circuits that generate all dual tone multi-frequency (DTMF) pairs required in tone-dialing systems. The 8 audio output frequencies are generated from an on-chip 3.579545 MHz master oscillator. No external components other than the crystal are required for the oscillator. The MM5395 and MM53125 can be powered directly from telephone lines over wide range loop conditions. The MM53125 interfaces to an inexpensive single-contact calculator type keypad. The MM5395 interfaces to a standard telephone 2-of-8 keypad.

## Features

- Powered directly from telephone line
- Low voltage operation to 3.5V
- Uses inexpensive 3.579545 MHz crystal
- Tone accuracy better than ±1% without tuning
- Operation with either single-contact or 2-of-8 keypads
- Excellent thermal and voltage stability
- High band pre-emphasis
- Multi-key lockout with single tone capability
- Mute switch output
- BCD interface mode

## Block and Connection Diagrams

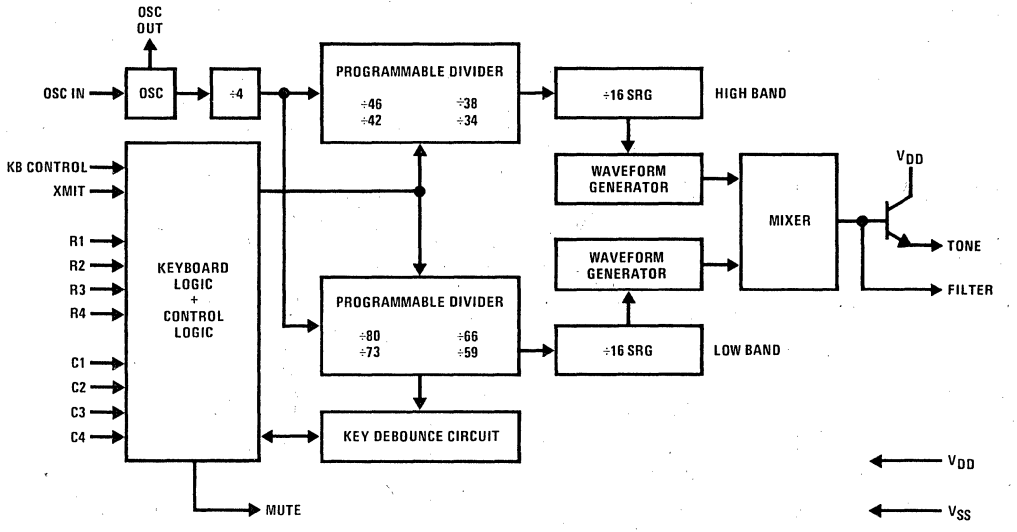
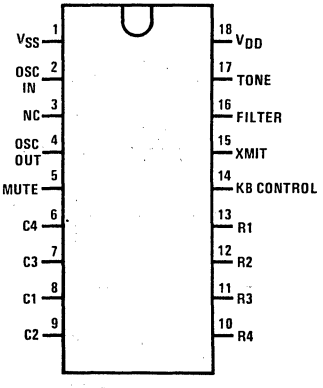



Figure 1a

Dual-In-Line Package




TOP VIEW

Figure 1b

**Order Number MM5395N or MM53125N**
**See Package 20**

®Registered trademark of Bell Telephone

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS} - 0.3V$ to $V_{DD} + 0.3V$ |
| Operating Temperature Range | $-30°C$ to $+70°C$ |
| Storage Temperature Range | $-55°C$ to $+150°C$ |
| $V_{DD} - V_{SS}$ | 6.5V |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating temperature range, $3.5V \leq V_{DD} - V_{SS} \leq 6V$, unless otherwise specified

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Input Pull-Up Resistor at Column Inputs | $V_{IN} = V_{SS}$ | 100 | | 400 | kΩ |
| Input Pull-Down Resistor at "XMIT" | $V_{IN} = V_{DD}$ | 100 | | 400 | kΩ |
| Internal Resistor at Row Inputs | | | | | |
| To $V_{DD}$ (MM5395) | $V_{IN} = V_{SS}$ | 100 | | 400 | kΩ |
| To $V_{SS}$ (MM53125) | $V_{IN} = V_{DD}$ | 100 | | 400 | kΩ |
| Keypad Contact Resistance | | | | 1 | kΩ |
| Input Voltage Levels | | | | | |
| Logical "1" | | $V_{DD}-0.25$ | | $V_{DD}$ | V |
| Logical "0" | | $V_{SS}$ | | $V_{SS}+0.25$ | V |
| Output Voltage Swings at "TONE OUTPUT" | $V_{DD} - V_{SS} = 3.5V$, $R_L \geq 500\Omega$ | | | | |
| Low Band Only | | | 820 | | mVp-p |
| High Band Only | | | 1000 | | mVp-p |
| High Band Pre-Emphasis | | | 2 | | dB |
| Harmonic Distortion | $R_L \geq 500\Omega$, | | | | |
| | No External Filtering | | −19 | | dB |
| | With 1000 pF at Filter | | −27 | | dB |
| Tone Frequency Deviation | | | | 1.0 | % |
| Operating Frequency | | | 3.579545 | | MHz |
| Key Debounce Time | | | 2 | 4 | ms |
| Power Dissipation | $V_{DD} - V_{SS} = 6V$, $R_L = 500\Omega$ | | | 50 | mW |
| Output Current Level at "MUTE" | $V_{DD} - V_{SS} = 3.5V$ | | | | |
| Logical "1" | $V_{OUT} = V_{DD} - 0.2V$ | 20 | | | μA |
| Logical "0" | $V_{OUT} = V_{SS} + 0.5V$ | 2.0 | | | mA |

## Functional Description

A functional block diagram of the MM5395 (or MM53125) is shown in *Figure 1a*, and a connection diagram is shown in *Figure 1b*. The oscillator will start immediately upon power being applied. When a key is pressed, both output tones start from zero on the negative half cycle after a 2 to 4 ms key debounce period. If 2 or more keys are pressed together, one or both tones will be switched OFF according to the functional truth table, *Figure 2a*. Output frequencies and accuracies are shown in *Figure 2b*.

The KB CONTROL input is used to change the interface from keyboard to BCD according to *Figure 3*. In the BCD interface mode, tone pairs are generated corresponding to the input BCD code on the ROW inputs *(Figure 4)* and are enabled during the period XMIT is high. By appropriate use of the COLUMN inputs during this mode, individual tones can be generated for test or signaling purposes.

## Functional Description (Continued)

A MUTE output is provided to electronically control common key functions such as switching out the transmitter and switching a muting resistor to the receiver.

The sum of the 2 sine waves is provided at the TONE output. A FILTER connection is available for access to the base of the output emitter follower for efficient filtering of the output waveform. A 1000 pF capacitor produces a total harmonic distortion 20 dB below the in band power without degrading high band pre-emphasis for operation in the North American telephone system. The TONE output signal amplitude varies directly with the $V_{DD}$ supply. Using a zener diode to clamp this supply near the low end of the line variation and the output circuits shown in *Figures 5 and 6* generate a line current signal amplitude that will remain constant with line voltage variations. Typical performance of this circuit is shown in *Figure 7*. In order to meet all CEPT and BPO guidelines for unwanted frequency components at 10 kHz and above the output, additional external filtering is required as shown in *Figure 8*.

*Figure 9* is a keypad interconnection diagram to indicate row and column connections for both types of keypads. Timing waveforms are shown in *Figure 10*.

| ROW | COLUMN | LOW BAND | HIGH BAND |
|---|---|---|---|
| None | None | DC | DC |
| One | One | $f_L$ | $f_H$ |
| None | One | DC | $f_H$ |
| One | None | $f_L$ | DC |
| Two or more | None | DC | DC |
| Two or more | One | DC | $f_H$ |
| None | Two or more | DC | DC |
| One | Two or more | $f_L$ | DC |
| Two or more | Two or more | DC | DC |

**a. Functional Truth Table**

| INPUTS | DESIRED FREQUENCIES | | ACTUAL FREQUENCY | PERCENT DEVIATION |
|---|---|---|---|---|
| | $f_L$ (Hz) | $f_H$ (Hz) | (Hz) | |
| R1 | 697 | – | 699.1 | 0.306 |
| R2 | 770 | – | 766.2 | −0.497 |
| R3 | 852 | – | 847.4 | −0.536 |
| R4 | 941 | – | 948.0 | 0.741 |
| C1 | – | 1209 | 1215.9 | 0.569 |
| C2 | – | 1336 | 1331.7 | −0.324 |
| C3 | – | 1477 | 1471.9 | −0.35 |
| C4 | – | 1633 | 1645.0 | 0.736 |

**b. Output Frequencies**

**FIGURE 2. Keypad Interface Mode**

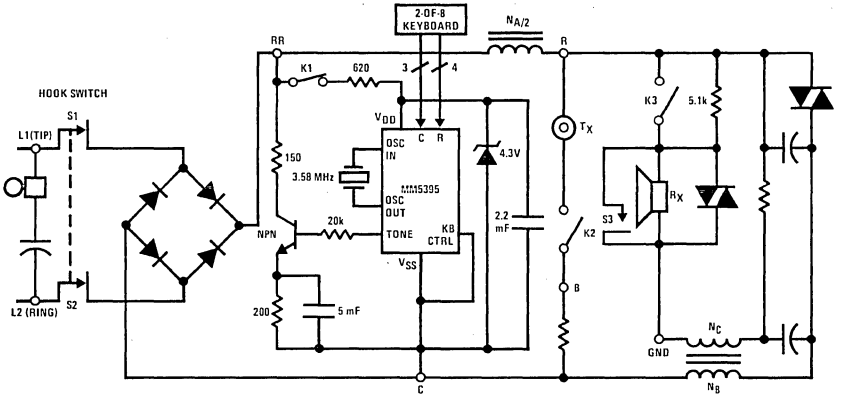
| KB CONTROL | XMIT | INTERFACE MODE |
|---|---|---|
| 0 | Open | Keypad |
| 1 | 0 | Idle |
| 1 | 1 | Send tones |

**FIGURE 3. Interface Mode Control**

| XMIT | C1 | C2 | R1 | R2 | R3 | R4 | FREQUENCIES GENERATED | |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | $f_L$ (Hz) | $f_H$ (Hz) |
| 0 | X | X | X | X | X | X | DC | DC |
| 1 | Open | Open | 0 | 0 | 0 | 0 | 941 | 1336 |
| 1 | Open | Open | 0 | 0 | 0 | 1 | 697 | 1209 |
| 1 | Open | Open | 0 | 0 | 1 | 0 | 697 | 1336 |
| 1 | Open | Open | 0 | 0 | 1 | 1 | 697 | 1477 |
| 1 | Open | Open | 0 | 1 | 0 | 0 | 770 | 1209 |
| 1 | Open | Open | 0 | 1 | 0 | 1 | 770 | 1336 |
| 1 | Open | Open | 0 | 1 | 1 | 0 | 770 | 1477 |
| 1 | Open | Open | 0 | 1 | 1 | 1 | 852 | 1209 |
| 1 | Open | Open | 1 | 0 | 0 | 0 | 852 | 1336 |
| 1 | Open | Open | 1 | 0 | 0 | 1 | 852 | 1477 |
| 1 | 0 | Open | Valid BCD Inputs | | | | $f_L$ | DC |
| 1 | Open | 0 | | | | | DC | $f_H$ |
| 1 | 0 | 0 | | | | | DC | DC |

**FIGURE 4. Functional Truth Table for Signal Interface Mode**

## Functional Description (Continued)




**Note 1:** All S switches are common with hookswitch.
**Note 2:** All K switches are common with KB.
**Note 3:** Switches shown in "OFF Hook" and "KB Depressed" positions.

FIGURE 5. MM5395 Typical Application

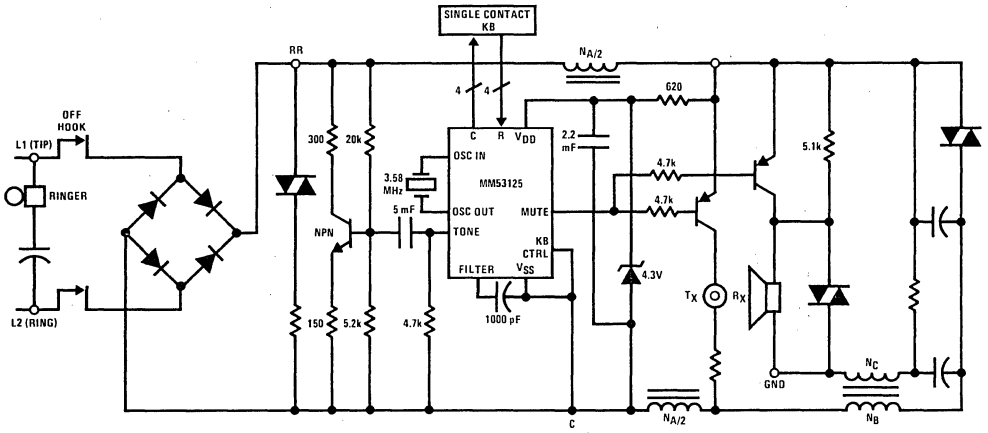



FIGURE 6. MM53125 Typical Application

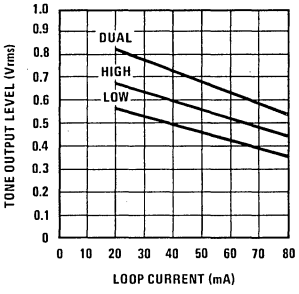



FIGURE 7. Typical Tone Output vs Loop Current

## Functional Description (Continued)

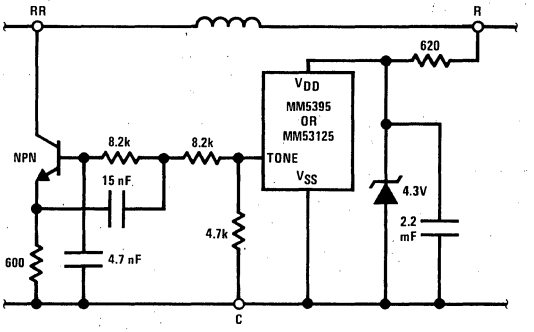
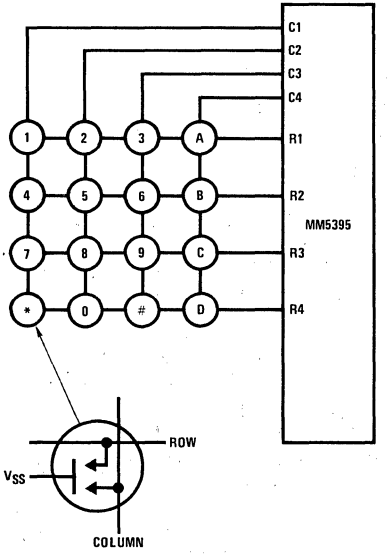
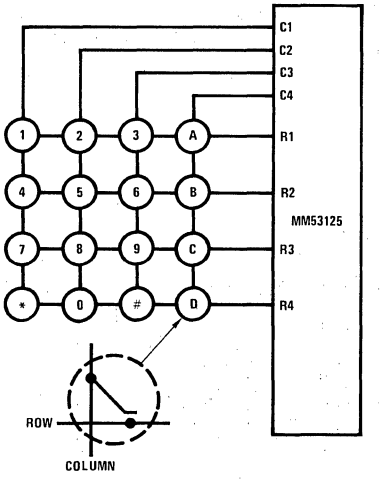



FIGURE 8. Tone Output Circuit for European Application

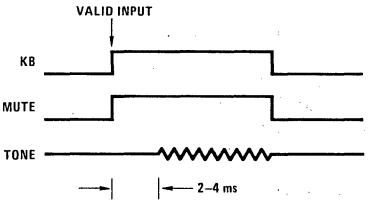



a) Standard Telephone Keypad

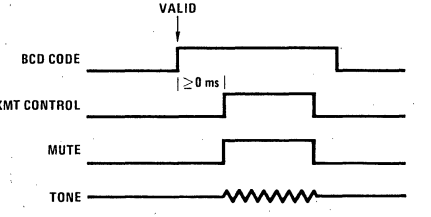



b) Single Contact Keypad

FIGURE 9. Keypad Interconnection Diagrams




a. Keyboard Mode (KB Control = 0)




b. BCD Mode (KB Control = 1)

FIGURE 10

# MM53130 DTMF (TOUCH TONE®) Generator

## General Description

The MM53130 is a low threshold voltage, ion-implanted, metal-gate CMOS integrated circuit that generates all dual tone multi-frequency (DTMF) pairs required in tone-dialing systems. The 8 audio output frequencies are generated from an on-chip 3.579545 MHz master oscillator. No external components other than the crystal are required for the oscillator. The MM53130 can be powered directly from telephone lines over wide range loop conditions. The device can interface directly to an inexpensive single-contact calculator type keyboard or a standard telephone 2-of-8 keypad *(Figure 4)*. The MM53130 is also capable of accepting binary code inputs for microprocessor-controlled systems applications.

## Features

- 3V-8V operating voltage
- On-chip 3.579545 MHz crystal-controlled oscillator
- Tone accuracy better than ± 1% without tuning
- Interface with standard 2-of-8 telephone keypad
- Interface with single-contact low cost keypad
- Input signals can be in binary code
- Multi-key lockout with/without single tone capability
- On-chip high band and low band tone generators and mixer
- High band pre-emphasis
- Low harmonic distortion
- Open emitter-follower low impedance output
- Separate receiver mute and transmitter mute switch outputs
- Powered directly from the telephone line

## Block Diagram

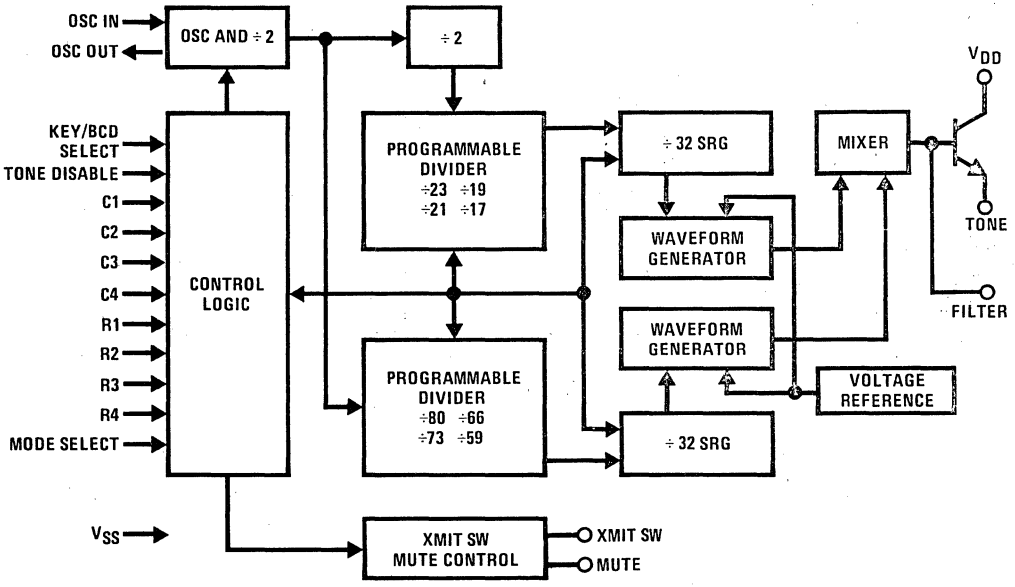



**FIGURE 1**

® Registered trademark of Bell Telephone

## Absolute Maximum Ratings

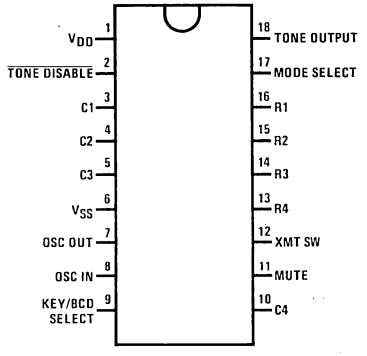
| | |
|---|---|
| Voltage at Any Pin Except XMT SW and MUTE | $V_{SS}-0.3V$ to $V_{DD}+0.3V$ |
| Voltage at XMT SW and MUTE Pins | $V_{SS}-0.3V$ to 15V |
| Operating Temperature Range | −40°C to +70°C |
| Storage Temperature Range | −65°C to +150°C |
| $V_{DD}-V_{SS}$ | 15V |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics $T_A$ within operating temperature range, $3V \le V_{DD} \le 8V$, unless otherwise specified.

| Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| Input Pull-Up Resistor | | | | | |
| Column and Row Inputs | | 25 | 50 | 90 | kΩ |
| Key/BCD Select | | 200 | 650 | 1000 | kΩ |
| Mode Select | | 200 | 650 | 1000 | kΩ |
| Tone Disable | | 200 | 650 | 1000 | kΩ |
| Input Pull-Down Resistor | | | | | |
| Column and Row Inputs | $V_{DD}=3V$ | 650 | | | Ω |
| | $V_{DD}=8V$ | 200 | | | Ω |
| Input Voltage Levels | | | | | |
| Logical "1" | | 80% of $V_{DD}$ | | $V_{DD}$ | V |
| Logical "0" | | $V_{SS}$ | | 20% of $V_{DD}$ | V |
| Operating Frequency | | | 3.579545 | | MHz |
| Output Voltage Swing at Tone Output | | | | | |
| Low Band Alone | $R_L>150\Omega$ | | 820 | | mVp-p |
| High Band Alone | $R_L>150\Omega$ | | 1000 | | mVp-p |
| Harmonic Distortion | $R_L>150\Omega$ | | | −20 | dB |
| Tone Frequency Deviation | | | | 1.0 | % |
| Typical Application Output Level $V_L$ (See *Figure 5*) | $20<I_L<100$ mA | | | | |
| Low Band Tone | $R_L=150\Omega$ | | −7 | | dBV |
| High Band Tone | $R_L=150\Omega$ | | −6 | | dBV |
| THD | $f\le 20$ kHz | | 4 | | % |
| Output Currents | $V_{DD}=3V$ | | | | |
| XMT SW/MUTE | $V_{OUT}=2V$ | 3 | | | mA |
| Idle Current | $R_L=\infty$, $V_{DD}=8.0V$ (No Key Depressed) | | | 1 | mA |
| Operating Current | $R_L=\infty$, $V_{DD}=3.5V$ | | | 2 | mA |
| Key Down to Tone Outputting Time (Debounce) | | | 3 | 4 | ms |
| DC Output | Tone Disable = 0 | | TRI-STATE® | | |

## Connection Diagram

**Dual-In-Line Package**




TOP VIEW

**Order Number MM53130N**
**See Package 20**

**FIGURE 2**

## Functional Description

A functional block diagram of the MM53130 is shown in *Figure 1*, and connection diagram is shown in *Figure 2*. The MM53130 can be operated in the Keyboard Interface Mode and can also be operated in the Signal Interface Mode depending on the logic level at the Key/Binary Select input. In either mode, the device will digitally synthesize the high and low band sine waves of DTMF signaling, when valid signals are applied to row and/or column inputs. The sum of the two sine waves is then provided at the Tone output.

**Tone Disable:** This input has an internal pull-up resistor. When this input is open or at logical high ($V_{DD}$), the XMT SW and MUTE outputs will deliver valid output signals in response to the proper input signals. When Tone Disable is at logical low ($V_{SS}$), the device will be in the inactive mode. Tone output will go to an open circuit state, XMT SW and MUTE outputs will sink current through on-chip open drain N-channel devices and the crystal oscillator will be disabled.

**Key/Binary Select:** When this input is open or at logical high ($V_{DD}$), the device will interface a keyboard. (See Table I.) When Key/Binary Select is low ($V_{SS}$), the device will accept binary inputs on the row signal input lines. (See Table II.)

**Oscillator:** Tone generation and internal timing is dependent on the accurate operation of the crystal oscillator. The oscillator inverter/amplifier and all necessary bias networks are included on-chip. The only external component is a 3.579545 MHz crystal. It should be connected to the device as shown in the typical application diagram *(Figure 5)*. The oscillator is not running unless a valid input signal is applied to the device. The oscillator is also disabled when Tone Disable is tied to logic low ($V_{SS}$). This feature will prevent RF modulation on the telephone line.

**Single Tone Capability:** This is a desirable feature for initial testing. With the device operating in the Keypad Interface Mode, operation of multiple keys in different rows and columns will not generate output tones. However, operation of two or more keys in the same row or column will generate the proper tone for that row or column. During multiple key operation, the XMT SW and MUTE outputs will not change state more than once. With the device operating in the Signal Interface Mode, a logical low at the column 1 input will inhibit the high-band tone output while a logical low at the column 2 input will inhibit the low-band tone output. (See Table I.) Logical low inputs on both column inputs 1 and 2 will disable the device the same way as the Tone Disable input will when set to logical low.

**Mode Select:** This input has an internal pull-up resistor. When open or at logical high, single tone outputs are allowed. When this input is at logical low, single tone outputs are prohibited. XMT SW and MUTE outputs will stay open circuit during a multiple key depression input.

**Tone Output:** Dual-tone output frequencies are generated in response to valid input signals to the device. (See Table III.) Each frequency is synthesized with 32 steps of approximation for low harmonic distortion. The amplitudes of the low and high frequency tones are constant and independent of operating voltages. When tone outputs are present, the Tone output will be the composite of the AC signal superimposed on a DC offset. The DC offset is approximately 1/2 $V_{DD}$. When no tones are present at the Tone output pin, the pin will be open circuit.

**XMT SW (Transmitter Switch) and MUTE Outputs:** In the idle state (no key depressed, no signal interface inputs and Tone Disable at a logical low) both the XMT SW and MUTE outputs will sink current to $V_{SS}$ through on-chip transistors. In the active state, these outputs will be open circuits wherever valid output tones are generated. The MUTE output activates before the XMT SW output as shown in *Figure 3*.

**Signal Inputs (Row and Column Inputs):** These inputs do not have a fixed pull-up or pull-down internal resistor, or a fixed logical level. Logic levels at the inputs are determined by internal states of the device. An input scan technique is used so that the device can directly interface either 2-of-8 keypads with common switch arrangements or the single contact X-Y keypads. (See *Figure 4*.)

## Functional Description (Continued)

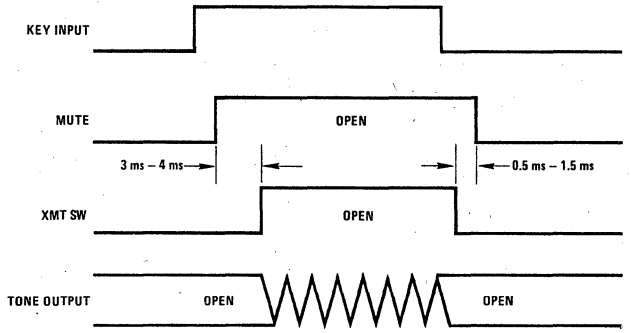



**FIGURE 3. Timing Diagram of MUTE and XMT SW in Relation to Key Input and Tone Output**

**TABLE I. FUNCTIONAL TRUTH TABLE (WITH "MODE SELECT" OPEN)**

| Key/Binary Select | Tone Disable | Row | Column | Tone Output | | XMT SW | MUTE |
|---|---|---|---|---|---|---|---|
| | | | | Low Band | High Band | | |
| X | 0 | X | X | 0 | 0 | 0 | 0 |
| 1 | 1 | One | One | $f_L$ | $f_H$ | 1 | 1 |
| 1 | 1 | One | Two or More | $f_L$ | 0 | 1 | 1 |
| 1 | 1 | Two or More | One | 0 | $f_H$ | 1 | 1 |
| 1 | 1 | Two or More | Two or More | 0 | 0 | 0 | 0 |
| 0 | 1 | Binary | Open | $f_L$ | $f_H$ | 1 | 1 |
| 0 | 1 | Binary | C1 = 0 | $f_L$ | 0 | 1 | 1 |
| 0 | 1 | Binary | C2 = 0 | 0 | $f_H$ | 1 | 1 |
| 0 | 1 | X | C1 and C2 = 0 | 0 | 0 | 0 | 0 |

**TABLE II. FUNCTIONAL TRUTH TABLE FOR SIGNAL INTERFACE**

| Keyboard Inputs | Binary Inputs | | | | | | Frequencies Generated | |
|---|---|---|---|---|---|---|---|---|
| | C1 | C2 | R1 | R2 | R3 | R4 | $f_L$ (Hz) | $f_H$ (Hz) |
| 1 | Open | Open | 0 | 0 | 0 | 1 | 697 | 1209 |
| 2 | Open | Open | 0 | 0 | 1 | 0 | 697 | 1336 |
| 3 | Open | Open | 0 | 0 | 1 | 1 | 697 | 1477 |
| 4 | Open | Open | 0 | 1 | 0 | 0 | 770 | 1209 |
| 5 | Open | Open | 0 | 1 | 0 | 1 | 770 | 1336 |
| 6 | Open | Open | 0 | 1 | 1 | 0 | 770 | 1477 |
| 7 | Open | Open | 0 | 1 | 1 | 1 | 852 | 1209 |
| 8 | Open | Open | 1 | 0 | 0 | 0 | 852 | 1336 |
| 9 | Open | Open | 1 | 0 | 0 | 1 | 852 | 1477 |
| 0 | Open | Open | 1 | 0 | 1 | 0 | 941 | 1336 |
| * | Open | Open | 1 | 0 | 1 | 1 | 941 | 1209 |
| # | Open | Open | 1 | 1 | 0 | 0 | 941 | 1477 |
| A | Open | Open | 1 | 1 | 0 | 1 | 697 | 1633 |
| B | Open | Open | 1 | 1 | 1 | 0 | 770 | 1633 |
| C | Open | Open | 1 | 1 | 1 | 1 | 852 | 1633 |
| D | Open | Open | 0 | 0 | 0 | 0 | 941 | 1633 |
| | 0 | Open | Valid Binary Inputs | | | | $f_L$ | — |
| | Open | 0 | | | | | — | $f_H$ |
| | 0 | 0 | | | | | 1/2 $V_{DD}$ | 1/2 $V_{DD}$ |

## Functional Description (Continued)

TABLE III. OUTPUT FREQUENCIES

| Inputs | Desired Freq. (Hz) | | Actual Frequency (Hz) | Percent Deviation |
|---|---|---|---|---|
| | $f_L$ | $f_H$ | | |
| R1 | 697 | | 699.1 | 0.306 |
| R2 | 770 | | 766.2 | −0.497 |
| R3 | 852 | | 847.4 | −0.536 |
| R4 | 941 | | 948.0 | 0.741 |
| C1 | | 1209 | 1215.9 | 0.569 |
| C2 | | 1336 | 1331.7 | −0.324 |
| C3 | | 1477 | 1471.9 | −0.35 |
| C4 | | 1633 | 1645.0 | 0.736 |

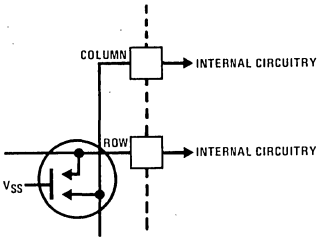



FIGURE 4a. Standard Dual Contact Telephone Key

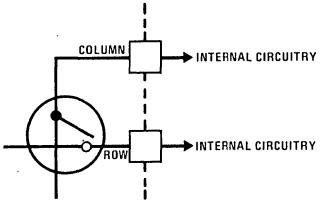
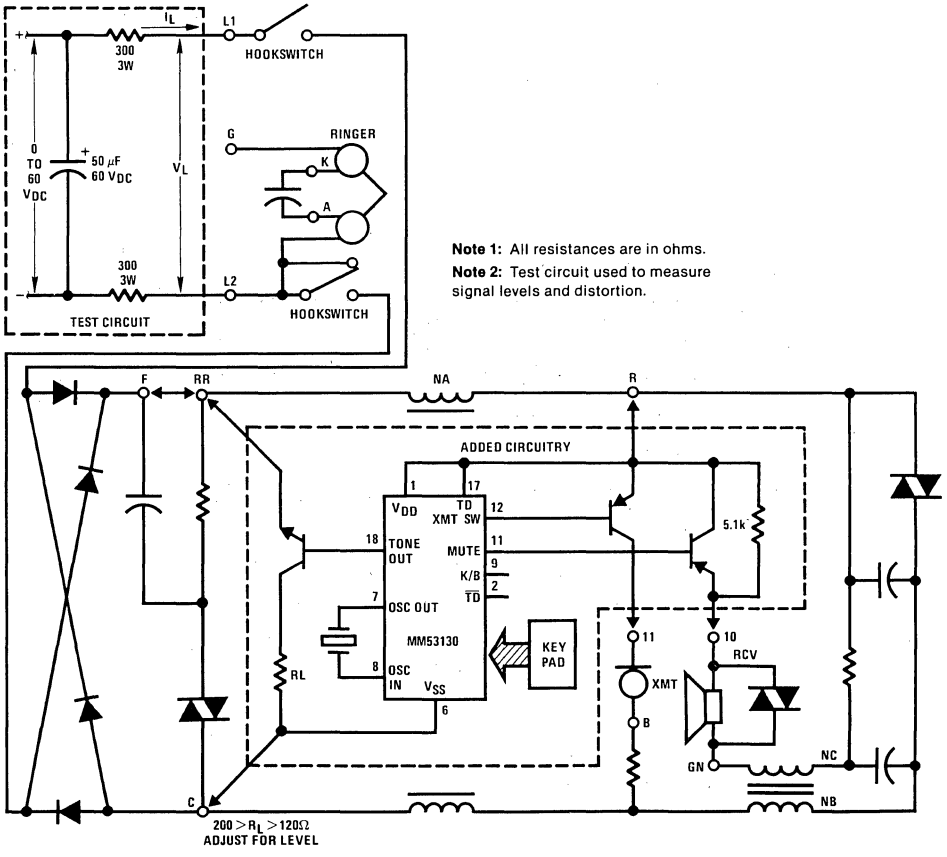



FIGURE 4b. Single Contact Key

## Typical Application




**Note 1:** All resistances are in ohms.

**Note 2:** Test circuit used to measure signal levels and distortion.

FIGURE 5

National Semiconductor

Telecommunications

# MM53190 Push-Button Pulse Dialer

## General Description

The MM53190 is a low threshold voltage, ion implanted, metal-gate CMOS integrated circuit that provides all the logic required to convert a push-button input into a series of pulses suitable for simulating a telephone rotary dial. The circuit works with both calculator type keypad (single-contact) or standard 2-of-7 type keypad. An inexpensive ceramic resonator is used as a frequency reference. When not actually outpulsing, or if there are no keypad entries, the MM53190 consumes only microamperes of current and does not allow any internal oscillators to run.

The MM53190 contains a 16-digit first-in–first-out memory that allows the user to enter digits faster than they are outpulsed. Numbers up to 16 digits may be dialed. After 16 digits have been entered, no more entries will be accepted. The outpulsing rate can be externally selected as either 10 pps or 20 pps. An interdigit pause of 4, 6, 8 or 10 times the dial pulse period is also externally selectable. The break/make ratio (ratio of the time the line is broken to the time the line is looped during outpulsing) is externally selectable to 1/1, 1.5/1, 1.6/1 or 2/1. A mute output is provided to mute receiver noise during outpulsing. No muting occurs during the inter-digit pause, thereby allowing the user to hear any busy or invalid condition arising during the call. The MM53190 provides a pacifier tone of 632 Hz every time a key is depressed. The last number entered may be redialed by use of the # key.

## Features

- Powered directly from the telephone line
- Uses standard calculator type keypad or 2-of-7 type keypad
- Uses inexpensive ceramic resonator for a frequency reference
- Pin-selectable outpulsing rate
- Pin-selectable interdigit pause
- Pin-selectable break/make ratio
- 632 Hz pacifier tone
- Redial of last number

## Block and Connection Diagrams

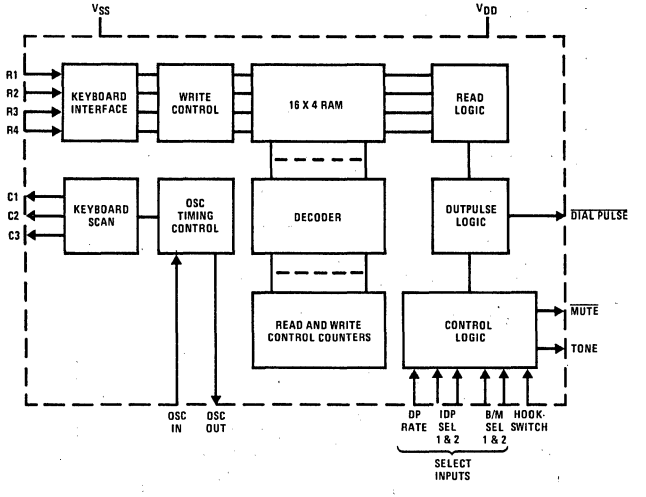
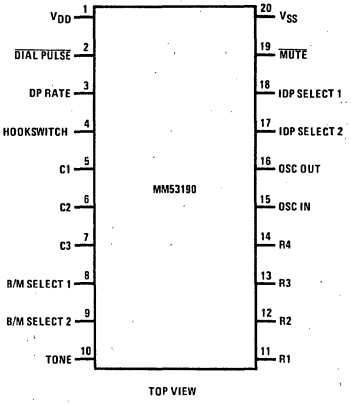



FIGURE 1

**Dual-In-Line Package**




**Order Number MM53190N**
**See Package 20A**

FIGURE 2

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ − 0.3 to $V_{DD}$ + 0.3V |
| Current into $\overline{DP}$ for Voltages Exceeding $V_{DD}$ | ≤ 500 μA |
| Operating Temperature Range | −30°C to +70°C |
| Storage Temperature Range | −40°C to +70°C |
| $V_{DD} - V_{SS}$ | 6.5V |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Operating Voltage Range

$V_{SS}$ = GND, $V_{DD}$ = 2.5V min, 5.5V max

## Electrical Characteristics

$V_{SS}$ = GND, 2.5V ≤ $V_{DD}$ ≤ 5.5V, −30°C ≤ $T_A$ ≤ +70°C unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Input Voltage | | | | | |
| Logical "1" | | $V_{DD}$−0.25 | | $V_{DD}$ | V |
| Logical "0" | | $V_{SS}$ | | $V_{SS}$+0.25 | V |
| Output Current Levels: | | | | | |
| $\overline{\text{Dial Pulse}}$ | | | | | |
| Logical "0", Sink | $V_{DD}$ = 3V, $V_{OUT}$ = 0.7V | 500 | | | μA |
| $\overline{\text{Mute}}$ | | | | | |
| Logical "0", Sink | $V_{DD}$ = 3V, $V_{OUT}$ = 0.7V | 500 | | | μA |
| Tone | | | | | |
| Logical "1" | $V_{DD}$ = 3V, $V_{OUT}$ = 2.75V | 4 | | | μA |
| Logical "0" | $V_{DD}$ = 3V, $V_{OUT}$ = 0.25V | 4 | | | μA |
| C1–C3 | | | | | |
| Logical "1" | $V_{DD}$ = 3V, $V_{OUT}$ = 2.75V | 1 | | | μA |
| Logical "0" | $V_{DD}$ = 3V, $V_{OUT}$ = 0.25V | 18 | | | μA |
| Keypad Resistance | | | | 1 | kΩ |
| Operating Current | $V_{DD}$ = 3V | | | | |
| | Quiescent | | | 1 | μA |
| | Oscillating | | | 300 | μA |
| Outpulsing Frequency | Osc = 488 kHz | 9.5 | | 10.5 | Hz |
| Input Leakages: | | | | | |
| Pins 3, 8, 9, 17, 18 | $V_{DD}$ = 5.5V, $V_{IN}$ = $V_{SS}$ | | | 5 | μA |
| Pins 11, 12, 13, 14 | $V_{DD}$ = 5.5V, $V_{IN}$ = $V_{SS}$ | | | 30 | μA |
| Pin 4 (Hookswitch) | $V_{DD}$ = 5.5V, $V_{IN}$ = $V_{SS}$ | | | 1 | μA |
| Pins 3, 8, 9, 17, 18 | $V_{DD}$ = 5.5V, $V_{IN}$ = $V_{DD}$ | | | 1 | μA |
| Pins 11, 12, 13, 14 | $V_{DD}$ = 5.5V, $V_{IN}$ = $V_{DD}$ | | | 1 | μA |
| Pin 4 (Hookswitch) | $V_{DD}$ = 5.5V, $V_{IN}$ = $V_{DD}$ | | | 5 | μA |

## Functional Description

A block diagram of the MM53190 integrated circuit is shown in *Figure 1* and a package connection diagram is shown in *Figure 2*.

**Oscillator (Pins 15 and 16):** The precision time base of the MM53190 pulse dialer is provided by an internal oscillator circuit which utilizes an inexpensive ceramic resonator as a frequency reference. Two external capacitors, as shown in *Figure 3*, are needed to load the resonator to operate in the anti-resonant mode. A 455 kHz series resonance ceramic resonator will result in a frequency of oscillation of 488 kHz. Ceramic resonators are available from Vernitron Corporation, Murata Corporation, and Radio Materials Company. Frequency stability of ±5% can be maintained for all devices over the voltage and temperature ranges. When the circuit is not outpulsing, or no keys are depressed, the oscillator will be shut down to eliminate noise and minimize dissipation.

**Keypad (Pins 5–7 and 11–14):** Three column scan output pins and four row input pins are provided to utilize a standard single-contact keypad or 2-of-7 type keypad *(Figure 4)*. A valid key closure is recorded when a single row ($R_X$ input) is connected to a single column ($C_X$ output) or when a single row and a single column are brought to $V_{SS}$. Key closures are protected from

## Functional Description (Continued)

contact bounce for 6 ms. Roll-over keyboard inputs will be considered valid.

**$\overline{\text{Dial Pulse}}$ Output (Pin 2):** The $\overline{\text{Dial Pulse}}$ output drives an external bipolar transistor that sequentially opens (breaks) the telephone loop a number of times equal to the input digit selected. For example, key 5 will generate 5 loop current breaks. The $\overline{\text{Dial Pulse}}$ output is an open drain transistor that sinks current only during a break.

**Break/Make Select (Pins 8–9):** The break/make ratio of the MM53190 can be externally selected by the 2 break/make select pins to be 1/1, 1.5/1, 1.6/1 or 2/1. This allows applications in a wide variety of telephone systems (Table I).

**DP Rate Select (Pin 3):** The dial pulse rate select input is used to select an outpulsing rate of either 10 pps or 20 pps (Table II).

**IDP Select (Pins 17 and 18):** The IDP select inputs are used to select an interdigit separation of 400 ms, 600 ms, 800 ms or 1000 ms when the outpulsing rate is 10 pps; and 200 ms, 300 ms, 400 ms, or 500 ms when the outpulsing rate is 20 pps (Table III).

**$\overline{\text{Mute}}$ (Pin 19):** The $\overline{\text{Mute}}$ output is used to drive an external bipolar transistor that is used to mute the receiver during the outpulse period. The $\overline{\text{Mute}}$ output is an open drain transistor that only sinks current while muting. System timing between key closure, mute and dial pulse are shown in the timing diagram in *Figure 5*. For initial key entries, and subsequent key entries made 1 IDP period after the last digit has been outpulsed, mute will occur 1 IDP period before outpulsing begins. For key entries made during outpulsing, or during an IDP, there will be a pre-dial mute of 100 ms when the outpulsing rate is 10 pps, and a pre-dial mute of 50 ms when the outpulsing rate is 20 pps. The post-dial mute is 50 ms when the outpulsing rate is 10 pps and 25 ms when the outpulsing rate is 20 pps.

**Tone (Pin 10):** The MM53190 provides a pacifier tone output to provide audio feedback to the user that a key has been depressed. The output is a 632 Hz tone that can be capacitively coupled in to the telephone receiver.

**Redial:** This feature allows the user to automatically dial the last number that was dialed. This is accomplished by pushing the # key on the next dial attempt. The number to be redialed may be 3 to 16 digits long. If an access code is required, as in a PBX system, up to 2 digits may be entered before the dial tone is established and the redial key is pushed to automatically dial the remainder of the number. To maintain memory information, power must be present to the part while in the ON-HOOK condition. To detect the ON-HOOK condition, the hookswitch input (pin 4) must be left floating. Hookswitch is used to reset the internal control circuitry and memory pointers. To detect the OFF-hook condition, hookswitch must be at a logical "1". An example of the redial operation is shown below.

| | KEY INPUTS | OUTPULSES | MEMORY |
|---|---|---|---|
| First Try | 85P4087375000 | 854087375000 | 854087375000 |
| Second Try | 85P# | 854087375000 | 854087375000 |
| Third Try | 85P# | 854087375000 | 854087375000 |

Note. P indicates a user pause

TABLE I

| BREAK/MAKE RATIO | | |
|---|---|---|
| B/M | SELECT 1 | SELECT 2 |
| 1.5/1 | 0 | 0 |
| 2/1 | 0 | 1 |
| 1/1 | 1 | 0 |
| 1.6/1 | 1 | 1 |

TABLE II

| DIAL PULSE RATE | |
|---|---|
| pps | SELECT |
| 10 | 0 |
| 20 | 1 |

TABLE III

| INTERDIGITAL PAUSE | | | |
|---|---|---|---|
| IDP LENGTH | DIAL PULSE RATE | SELECT 1 | SELECT 2 |
| 800 ms | 10 pps | 0 | 0 |
| 400 ms | 20 pps | 0 | 0 |
| 1000 ms | 10 pps | 0 | 1 |
| 500 ms | 20 pps | 0 | 1 |
| 400 ms | 10 pps | 1 | 0 |
| 200 ms | 20 pps | 1 | 0 |
| 600 ms | 10 pps | 1 | 1 |
| 300 ms | 20 pps | 1 | 1 |

## Functional Description (Continued)

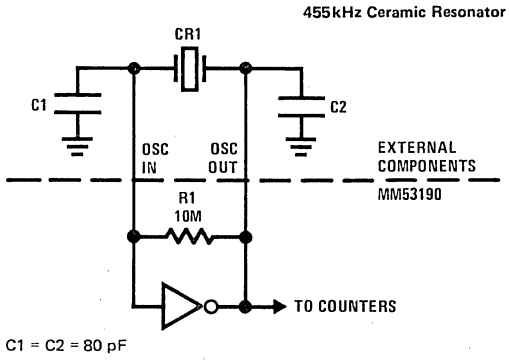



FIGURE 3

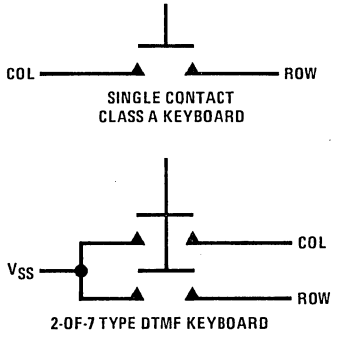
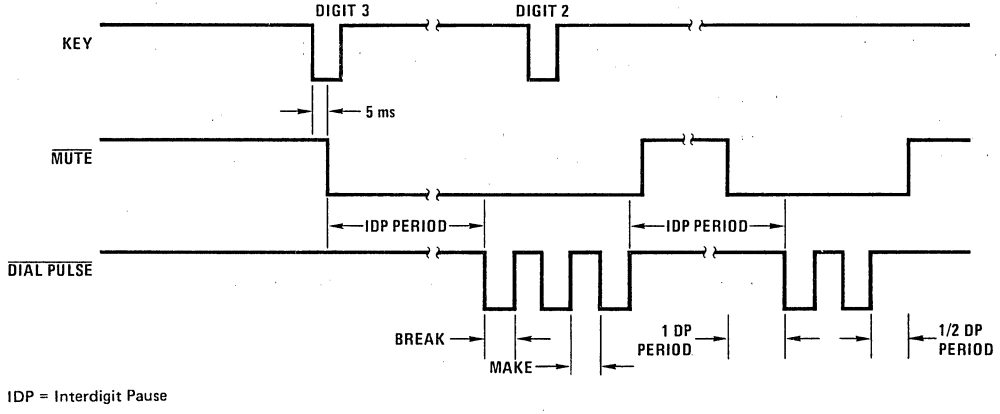



FIGURE 4

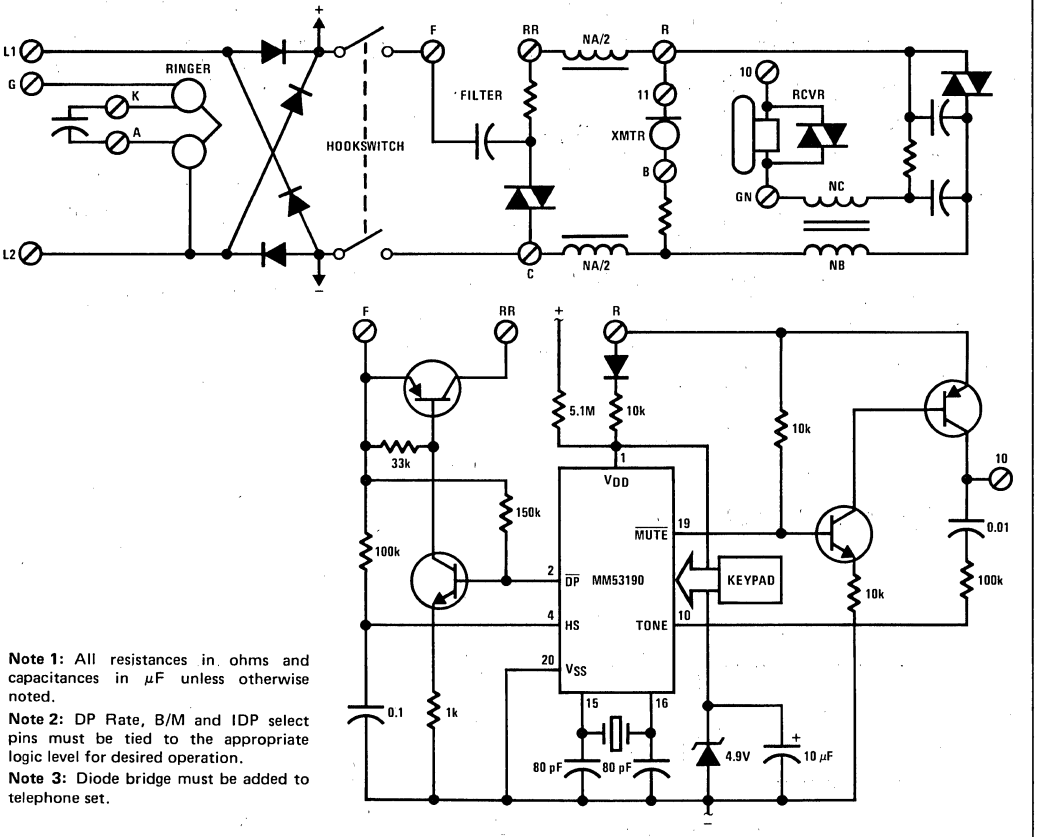
## Output Timing Waveforms




IDP = Interdigit Pause

FIGURE 5. $\overline{\text{Mute}}$ Not Active Between Digits

## Functional Description (Continued)




**Note 1:** All resistances in ohms and capacitances in µF unless otherwise noted.

**Note 2:** DP Rate, B/M and IDP select pins must be tied to the appropriate logic level for desired operation.

**Note 3:** Diode bridge must be added to telephone set.

FIGURE 6. Using the MM53190 Pulse Dialer with Redial Option

# Display Drivers

National Semiconductor

# MM5445, MM5446, MM5447, MM5448 VF Display Drivers

## General Description

The MM5445 through MM5448 are monolithic MOS integrated circuits utilizing P-channel metal gate low threshold, enhancement mode and ion-implanted depletion mode devices. They are available in 40-pin molded dual-in-line packages. Each output can source up to 1 mA at 2.0V maximum output voltage. A single pin controls the VF display brightness by setting the positive output voltage level.

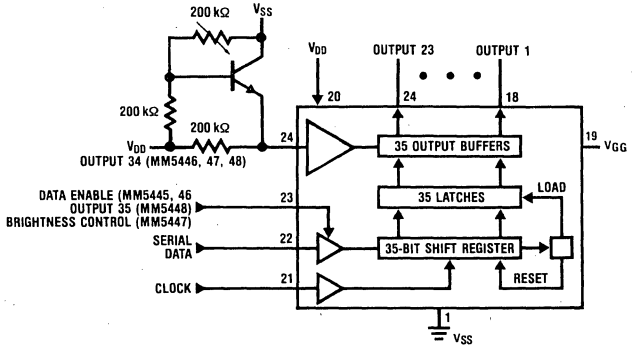
## Features

- Continuous brightness control
- Serial data input
- No load signal required
- Enable (on MM5445 and MM5446)
- Wide power supply operation
- TTL compatibility
- 33, 34 or 35 outputs, 1 mA source capability
- Alphanumeric capability
- Input data format compatible with MM5450, MM5451 LED drivers and MM5452, MM5453 LCD drivers

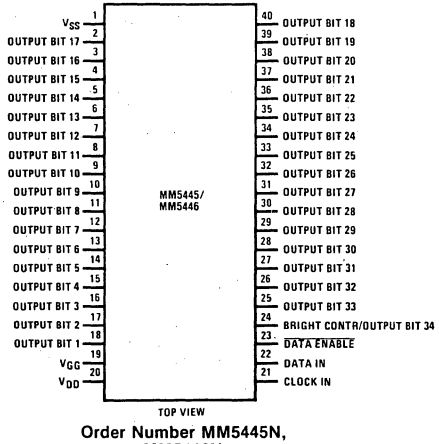
## Applications

- COPS or microprocessor displays
- Industrial control indicator
- Digital clock, thermometer, counter, voltmeter
- Instrumentation readouts

## Block Diagram



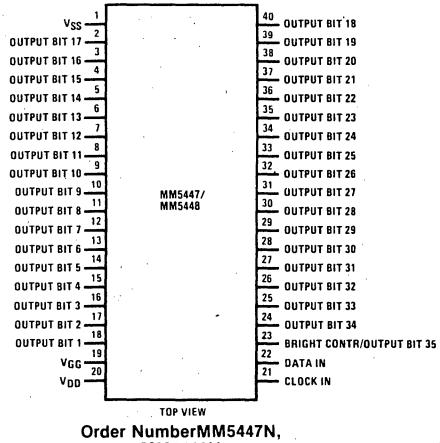

## Connection Diagrams (Dual-In-Line Packages)




TOP VIEW

Order Number MM5445N, MM5446N
See Package 24

Figure 2a




TOP VIEW

Order NumberMM5447N, MM5448N
See Package 24

Figure 2b

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ to $V_{SS}-30V$ |
| Operating Temperature | −40°C to +85°C |
| Storage Temperature | −65°C to +150°C |
| Power Dissipation | 560mW at +85°C<br>1W at +25°C |
| Junction Temperature | +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, $V_{DD}=0V$, $V_{SS}=4.5$ to 5.5V, unless otherwise specified.

| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| Power Supply | | | | | |
| $V_{SS}$ | | 4.5 | 5.0 | 5.5 | V |
| $V_{GG}$ | $V_{SS}=5V$ | −25 | | −7 | V |
| $V_{SS}$ | $V_{DD}=V_{GG}=0$ | 12 | | 18 | V |
| Power Supply Current | | | | | |
| $I_{SS}$ | $V_{SS}=5V$, $V_{GG}=-25V$ | | | 9 | mA |
| $I_{GG}$ | $V_{DD}=0$ | −2 | | | mA |
| Brightness Control | With respect to $V_{SS}$ | $V_{SS}-V_{GG}/2$ | | $V_{SS}$ | V |
| Input Logic Levels | | | | | |
| Logic "0" Level | $-25V \leq V_{GG} \leq -7V$ | −0.3 | | 0.7 | V |
| Logic "1" Level | $-25V \leq V_{GG} \leq -7V$ | 2.2 | | $V_{SS}+0.3$ | V |
| Logic "0" Level | $V_{DD}=V_{GG}=0$ | −0.3 | | 1 | V |
| Logic "1" Level | $V_{DD}=V_{GG}=0$ | $V_{SS}-1$ | | $V_{SS}+0.3$ | V |
| Input Currents | | | | | |
| DATA IN and CLOCK | | −10 | | 10 | μA |
| DATA ENABLE | | −10 | | 35 | μA |
| BRIGHTNESS CONTROL | Excluding Output Loads (Note 2) | | | 2 | mA |
| Output Source Current | | | | | |
| Segment OFF | $V_{OUT}=V_{SS}-V_{GG}/2$ | | | −2 | μA |
| Segment ON | $V_{OUT}=V_{SS}-2V$ (Notes 1 and 2) | 1 | | | mA |
| Input Clock Frequency | | 0 | | 250 | kHz |
| Duty Cycle | | 40 | 50 | 60 | % |
| Output Matching | $I_{OUT}=1mA$ | −0.5 | | 0.5 | V |

**Note 1:** With Brightness Control tied to $V_{SS}$ (MM5445 and MM5447) and $V_{GG}=-25V$.

**Note 2:** All output source current is provided from the Brightness Control input pin (MM5445 and MM5447).

## Functional Description

The MM5445 Series are specifically designed to operate 4 or 5-digit alphanumeric displays with minimal interface with the display and the data source. Character generation is done external to the MM5445 Series. Serial data transfer from the data source to the display driver is accomplished with 2 signals, serial data and clock. Using a format of a leading "1" followed by the 35 data bits allows data transfer without an additional load signal. The 35 data bits are latched after the 36th bit is complete, thus providing non-multiplexed, direct drive to the display. Outputs change only if the serial data bits differ from the previous time. Display brightness is determined by control of the positive output voltage level.

A block diagram is shown in *Figure 1.*

*Figure 2* shows the pin-out of the MM5445 series. Bit 1 is the first bit following the start bit and it will appear on pin 18. A logical "1" at the input will turn on the appropriate VF display segment.

*Figure 4* shows the input data format. A start bit of logical "1" precedes the 35 bits of data. At the 36th clock a LOAD signal is generated synchronously with the high state of the clock, which loads the 35 bits of the shift registers into the latches. At the low state of the clock a RESET signal is generated which clears all the shift registers for the next set of data. The shift registers are static master-slave configuration. There is no clear for the master portion of the first shift register, thus allowing continuous operation.

There must be a complete set of 36 clocks or the shift registers will not clear.

When the chip first powers ON an internal power ON reset signal is generated which resets all registers and all latches. The START bit and the first clock return the chip to its normal operation.

*Figure 3* shows the timing relationships between data, clock and data enable. A maximum clock frequency of 250kHz is assumed.

## Typical Applications

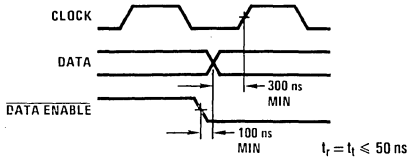



Figure 3

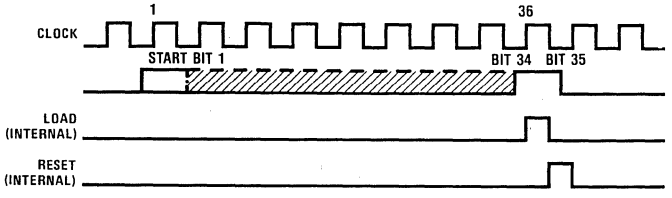
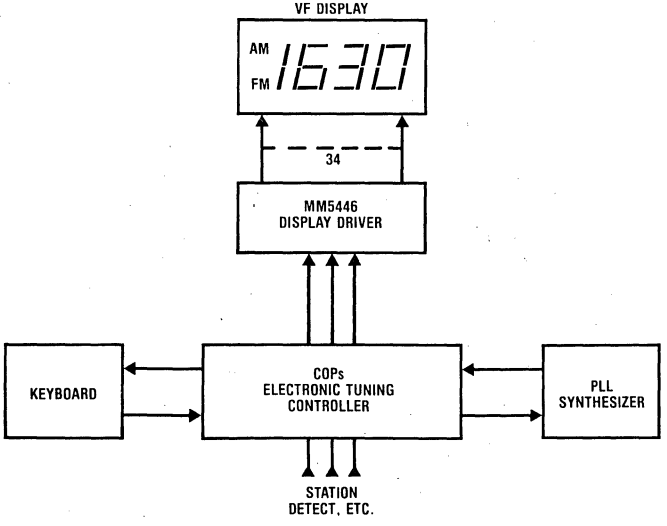



Figure 4. Input Data Format




Basic Electronically Tuned Radio System

National Semiconductor



# MM5450, MM5451 LED Display Drivers

## General Description

The 5450 and MM5451 are monolithic MOS integrated circuits utilizing N-channel metal-gate low threshold, enhancement mode, and ion-implanted depletion mode devices. They are available in 40-pin molded dual-in-line packages. A single pin controls the LED display brightness by setting a reference current through a variable resistor connected to $V_{DD}$.

## Features

- Continuous brightness control
- Serial data input
- No load signal required
- Enable (on MM5450)
- Wide power supply operation
- TTL compatibility
- 34 or 35 outputs, 15mA sink capability
- Alphanumeric capability

## Applications

- COPS or microprocessor displays
- Industrial control indicator
- Relay driver
- Digital clock, thermometer, counter, voltmeter
- Instrumentation readouts

## Block Diagram

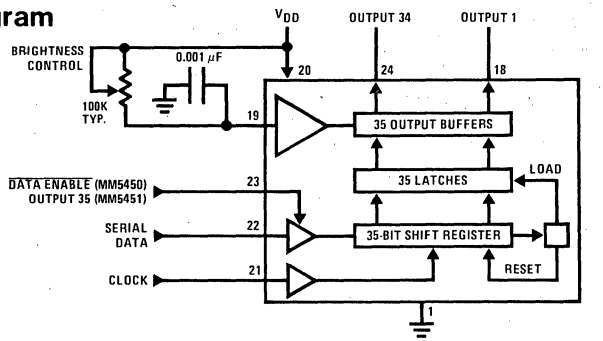
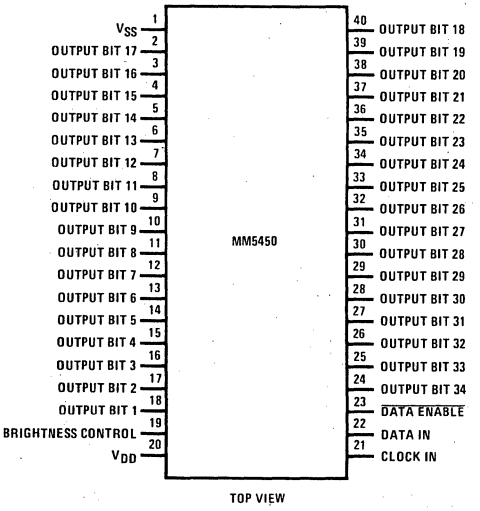



FIGURE 1

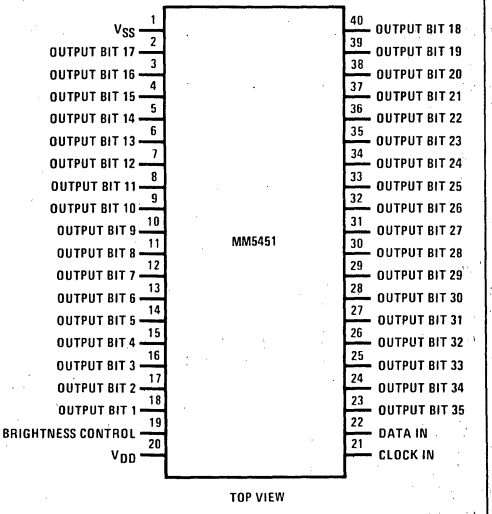
## Connection Diagrams (Dual-In-Line Packages)




TOP VIEW

**Order Number MM5450N**
**See Package 24**

FIGURE 2a




TOP VIEW

**Order Number MM5451N**
**See Package 24**

FIGURE 2b

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ to $V_{SS}$ + 12V |
| Operating Temperature | −25°C to +85°C |
| Storage Temperature | −65°C to +150°C |
| Power Dissipation | 560 mW at +85°C |
| | 1W at +25°C |
| Junction Temperature | +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, $V_{DD}$ =4.75V to 11.0V, $V_{SS}$ = 0V, unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply | | 4.75 | | 11 | V |
| Power Supply Current | Excluding Output Loads | | | 7 | mA |
| Input Voltages | | | | | |
| Logical "0" Level | ±10 μA Input Bias | −0.3 | | 0.8 | V |
| Logical "1" Level | $4.75 \leq V_{DD} \leq 5.25$ | 2.2 | | $V_{DD}$ | V |
| | $V_{DD} > 5.25$ | $V_{DD} - 2$ | | $V_{DD}$ | V |
| Brightness Input (Note 2) | | 0 | | 0.75 | mA |
| Output Sink Current (Note 3) | | | | | |
| Segment OFF | $V_{OUT}$ = 3.0V | | | 10 | μA |
| Segment ON | $V_{OUT}$ = 1V (Note 4) | | | | |
| | Brightness Input = 0 μA | 0 | | 10 | μA |
| | Brightness Input = 100 μA | 2.0 | 2.7 | 4 | mA |
| | Brightness Input = 750 μA | 15 | | 25 | mA |
| Brightness Input Voltage (Pin 19) | Input Current = 750 μA | 3.0 | | 4.3 | V |
| Input Clock Frequency | | 0 | | 0.5 | MHz |
| Duty Cycle | | 40 | 50 | 60 | % |
| Output Matching (Note 1) | | | | ±20 | % |

**Note 1:** Output matching is calculated as the percent variation from $I_{MAX} + I_{MIN}/2$.
**Note 2:** With a fixed resistor on the brightness input pin some variation in brightness will occur from one device to another.
**Note 3:** Absolute maximum for each output should be limited to 40mA.
**Note 4:** The $V_{OUT}$ voltage should be regulated by the user. See Figures 6 and 7 for allowable $V_{OUT}$ vs. $I_{OUT}$ operation.

## Functional Description

Both the MM5450 and the MM5451 are specifically designed to operate 4 or 5-digit alphanumeric displays with minimal interface with the display and the data source. Serial data transfer from the data source to the display driver is accomplished with 2 signals, serial data and clock. Using a format of a leading "1" followed by the 35 data bits allows data transfer without an additional load signal. The 35 data bits are latched after the 36th bit is complete, thus providing non-multiplexed, direct drive to the display. Outputs change only if the serial data bits differ from the previous time. Display brightness is determined by control of the output current for LED displays. A 0.001 capacitor should be connected to brightness control, pin 19, to prevent possible oscillations.

A block diagram is shown in *Figure 1*. For the MM5450 a DATA ENABLE is used instead of the 35th output. The DATA ENABLE input is a metal option for the MM5450. The output current is typically 20 times greater than the current into pin 19, which is set by an external variable resistor. There is an internal limiting resistor of 400Ω nominal value.

*Figure 4* shows the input data format. A start bit of logical "1" precedes the 35 bits of data. At the 36th clock a LOAD signal is generated synchronously with the high state of the clock, which loads the 35 bits of the shift registers into the latches. At the low state of the clock a RESET signal is generated which clears all the shift registers for the next set of data. The shift registers are static master-slave configuration. There is no clear for the master portion of the first shift register, thus allowing continuous operation.

There must be a complete set of 36 clocks or the shift registers will not clear.

When the chip first powers ON an internal power ON reset signal is generated which resets all registers and all latches. The START bit and the first clock return the chip to its normal operation.

*Figure 2* shows the pin-out of the MM5450 and MM5451. Bit 1 is the first bit following the start bit and it will appear on pin 18. A logical "1" at the input will turn on the appropriate LED.

## Functional Description (Continued)

*Figure 3* shows the timing relationships between data, clock and data enable. A max clock frequency of 0.5 MHz is assumed.

For applications where a lesser number of outputs are used, it is possible to either increase the current per output, or operate the part at higher than 1V $V_{OUT}$. The following equation can be used for calculations.

$$T_j = (V_{OUT})\ (I_{LED})\ (\text{No. of segments})\ (124^\circ C/W) + T_A$$

where:

$T_j$ = junction temperature +150°C max

$V_{OUT}$ = the voltage at the LED driver outputs

$I_{LED}$ = the LED current

124°C/W = thermal coefficient of the package

$T_A$ = ambient temperature

The above equation was used to plot *Figure 5, Figure 6, and Figure 7.*

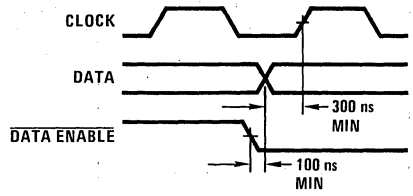



FIGURE 3

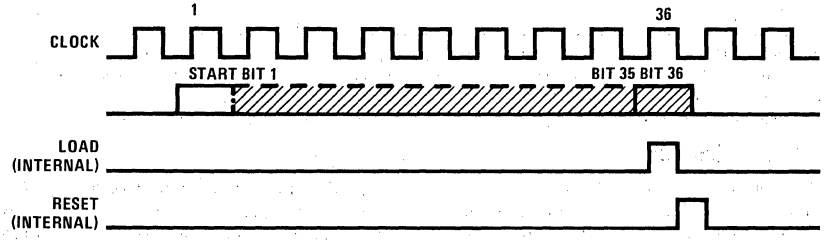



FIGURE 4. Input Data Format

## Typical Applications

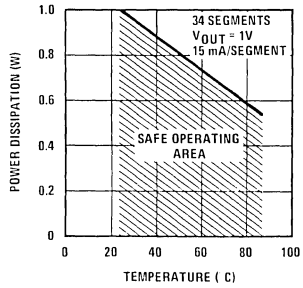



FIGURE 5

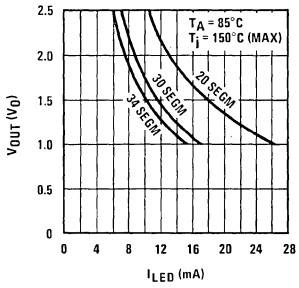



FIGURE 6

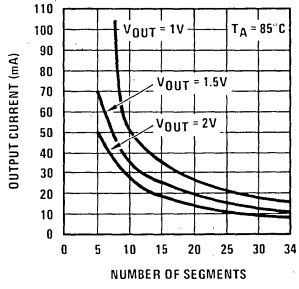
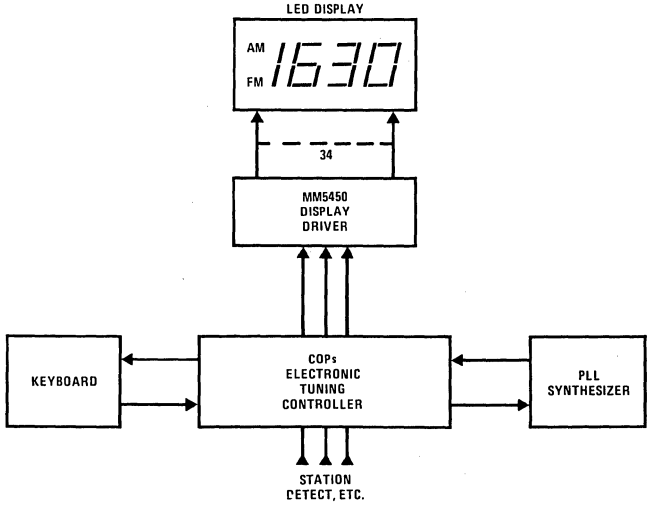



FIGURE 7

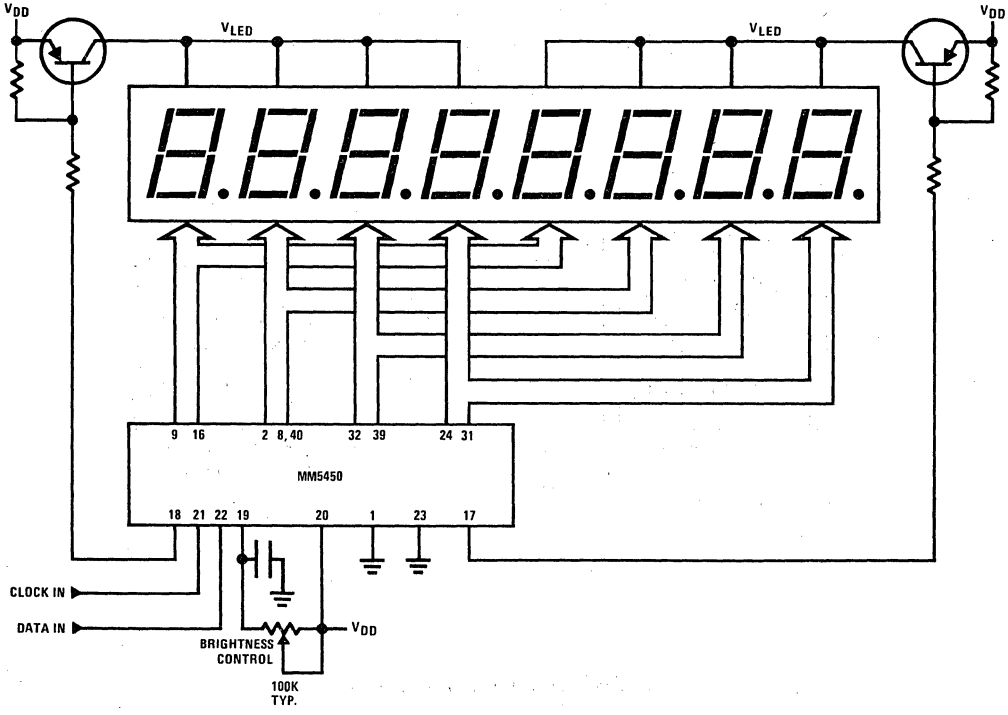
### Basic Electronically Tuned Radio System




5

## Typical Applications (Continued)

Duplexing 8 Digits with One MM5450



National Semiconductor



# MM5452, MM5453 Liquid Crystal Display Drivers

## General Description

The MM5452 is a monolithic integrated circuit utilizing CMOS metal gate, low threshold enhancement mode devices. It is available in a 40-pin molded package. The chip can drive up to 32 segments of LCD and can be paralleled to increase this number. The chip is capable of driving a 4 1/2-digit 7-segment display with minimal interface between the display and the data source.

The MM5452 stores the display data in latches after it is clocked in, and holds the data until new display data is received.

## Features

- Serial data input
- No load signal required
- $\overline{\text{DATA ENABLE}}$ (MM5452)
- Wide power supply operation
- TTL compatibility
- 32 or 33 outputs
- Alphanumeric and bar graph capability
- Cascaded operation capability

## Applications

- COPs or microprocessor displays
- Industrial control indicator
- Digital clock, thermometer, counter, voltmeter
- Instrumentation readouts
- Remote displays

## Block and Connection Diagrams

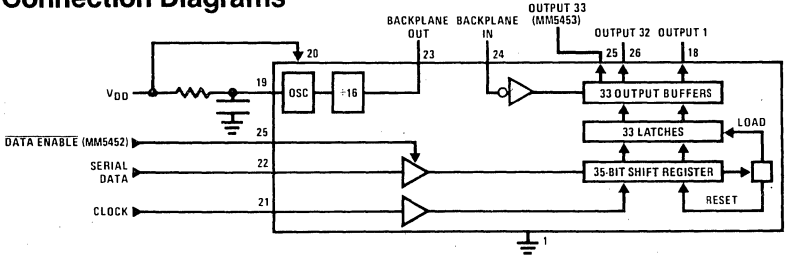
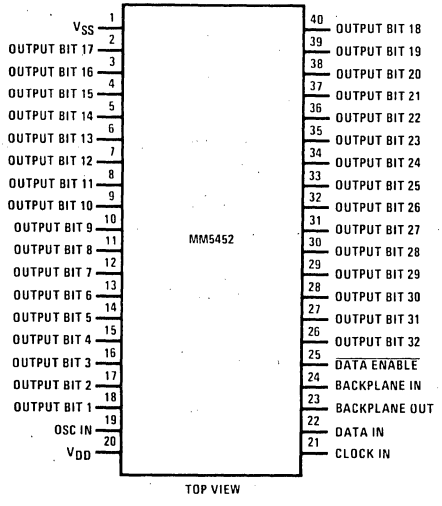



FIGURE 1

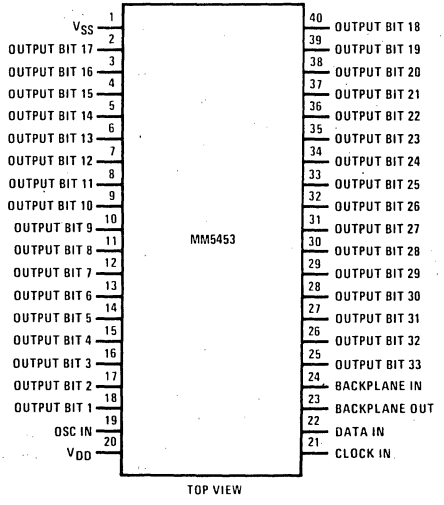
Dual-In-Line Package




Order Number MM5452N
See Package 24

FIGURE 2a

Dual-In-Line Package




Order Number MM5453N
See Package 24

FIGURE 2b

## Absolute Maximum Ratings

| | | | |
|---|---|---|---|
| Voltage at Any Pin | $V_{SS}$ to $V_{SS}+10V$ | Power Dissipation | 300 mW at +70°C<br>350 mW at +25°C |
| Operating Temperature | 0°C to +70°C | | |
| Storage Temperature | −65° to +150°C | Junction Temperature | +150°C |
| | | Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, $V_{DD}=3.0V$ to 10V, $V_{SS}=0V$, unless otherwise specified.

| Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| Power Supply | | 3 | | 10 | V |
| Power Supply Current | Excluding Outputs | | | 40 | μA |
| | OSC = $V_{SS}$, BP IN @ 32 Hz | | | 10 | μA |
| | $V_{DD}=5V$, Open Outputs, No Clock | | | | |
| Clock Frequency | | | | 500 | kHz |
| Input Voltages | | | | | |
| Logical '0' Level | $V_{DD}<4.75$ | −0.3 | | 0.1 $V_{DD}$ | V |
| | $V_{DD}\geq 4.75$ | −0.3 | | 0.8 | V |
| Logical '1' Level | $V_{DD}>5.25$ | 0.9 $V_{DD}$ | | $V_{DD}$ | V |
| | $V_{DD}\leq 5.25$ | 2.0 | | $V_{DD}$ | V |
| Output Current Levels | | | | | |
| Segments | | | | | |
| Sink | $V_{DD}=3V$, $V_{OUT}=0.3V$ | | | −20 | μA |
| Source | $V_{DD}=3V$, $V_{OUT}=V_{DD}-0.3V$ | 20 | | | μA |
| Backplane | | | | | |
| Sink | $V_{DD}=3V$, $V_{OUT}=0.3V$ | | | −320 | μA |
| Source | $V_{DD}=3V$, $V_{OUT}=V_{DD}-0.3V$ | 320 | | | μA |
| Output Offset Voltage | Segment Load 250 pF | | | | |
| | Backplane Load 8750 pF | | | ±50 | mV |

## Functional Description (Continued)

The MM5452 is specifically designed to operate 4 1/2-digit 7-segment displays with minimal interface with the display and the data source. Serial data transfer from the data source to the display driver is accomplished with 2 signals, serial data and clock. Since the MM5452 does not contain a character generator, the formatting of the segment information must be done prior to inputting the data to the MM5452. Using a format of a leading "1" followed by the 32 data bits allows data transfer without an additional load signal. The 32 data bits are latched after the 36th clock is complete, thus providing non-multiplexed, direct drive to the display. Outputs change only if the serial data bits differ from the previous time.

A block diagram is shown in *Figure 1*. For the MM5452 a $\overline{\text{DATA ENABLE}}$ is used instead of the 33rd output. If the $\overline{\text{DATA ENABLE}}$ signal is not required, the 33rd output can be brought out. This is the MM5453 device.

*Figure 4* shows the input data format. A start bit of logical "1" precedes the 32 bits of data. At the 36th clock a LOAD signal is generated synchronously with the high state of the clock, which loads the 32 bits of the shift registers into the latches. At the low state of the clock a RESET signal is generated which clears all the shift registers for the next set of data. The shift registers are static master-slave configuration. There is no clear for the master portion of the first shift register, thus allowing continuous operation.

If the clock is not continuous, there must be a complete set of 36 clocks otherwise the shift registers will not clear.

*Figure 2a* shows the pin-out of the MM5452. Bit 1 is the first bit following the start bit and it will appear on pin 18.

*Figure 3* shows the timing relationships between data, clock and $\overline{\text{DATA ENABLE}}$.

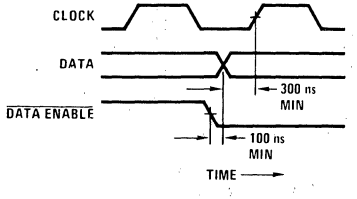



FIGURE 3

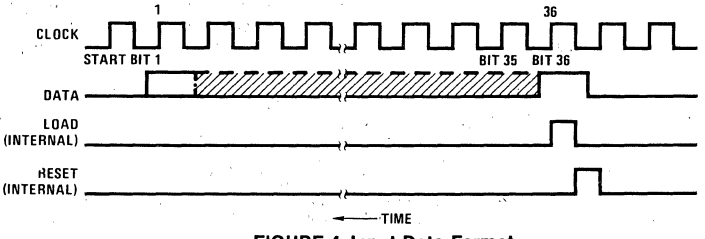
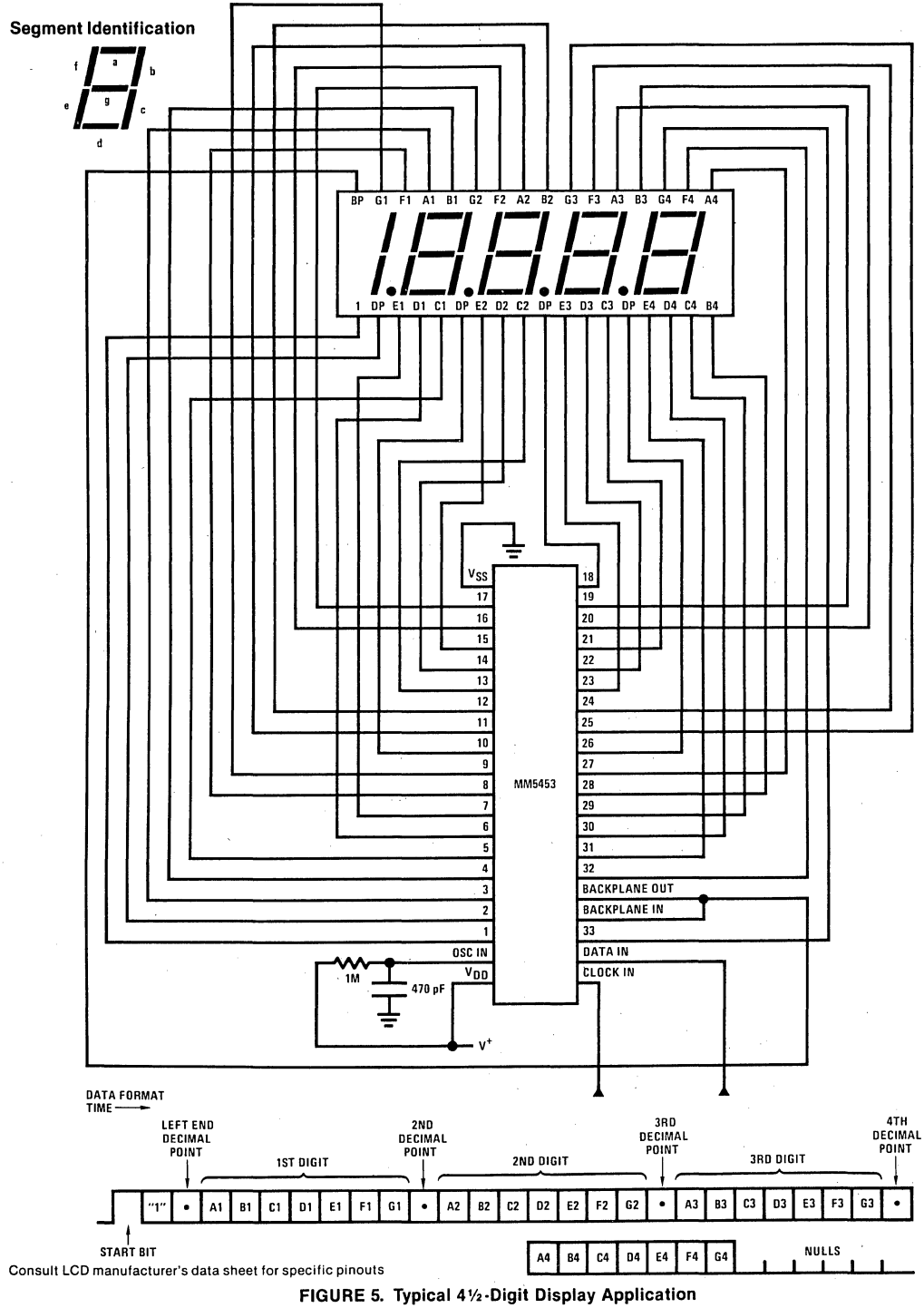



FIGURE 4. Input Data Format

## Functional Description (Continued)

*Figure 5* shows a typical application. Note how the input data maps to the output pins and the display. The MM5452 and MM5453 do not have format restrictions, as all outputs are controllable. This application assumes a specific display pinout. Different display/driver connection patterns will, of course, yield a different input data format.




Consult LCD manufacturer's data sheet for specific pinouts

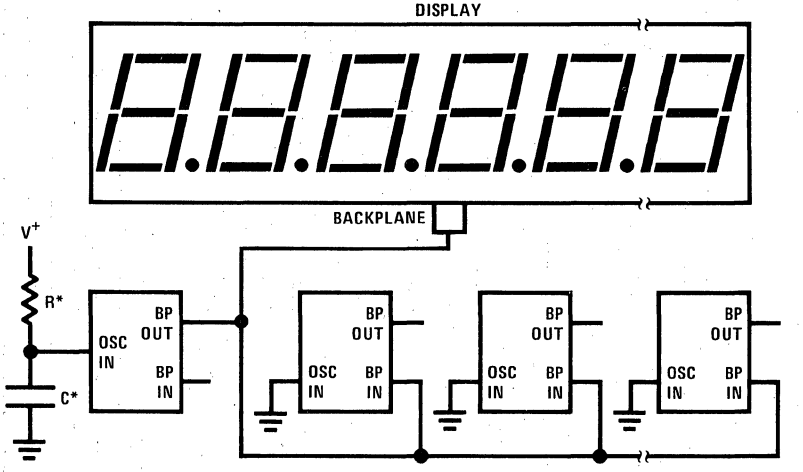
**FIGURE 5. Typical 4½-Digit Display Application**

## Functional Description (Continued)

*Figure 8* shows a four wire remote display that takes advantage of the device's serial input to move many bits of display information on a few wires.

### Using an External Clock

The MM5452, MM5453 LCD Drivers can be used with an externally supplied clock, provided it has a duty cycle of 50%. Deviations from a 50% duty cycle result in an offset voltage on the LCD. In *Figure 7*, a flip flop is used to assure a 50% duty cycle. The oscillator input is grounded to prevent oscillation and reduce current consumption in the chips. The oscillator is not used.




* The minimum recommended value for R for the oscillator input is 9 kΩ. An RC time constant of approximately $4.91 \times 10^{-4}$ should produce a backplane frequency between 30 Hz and 150 Hz.

**FIGURE 6. Parallel Backplane Outputs**

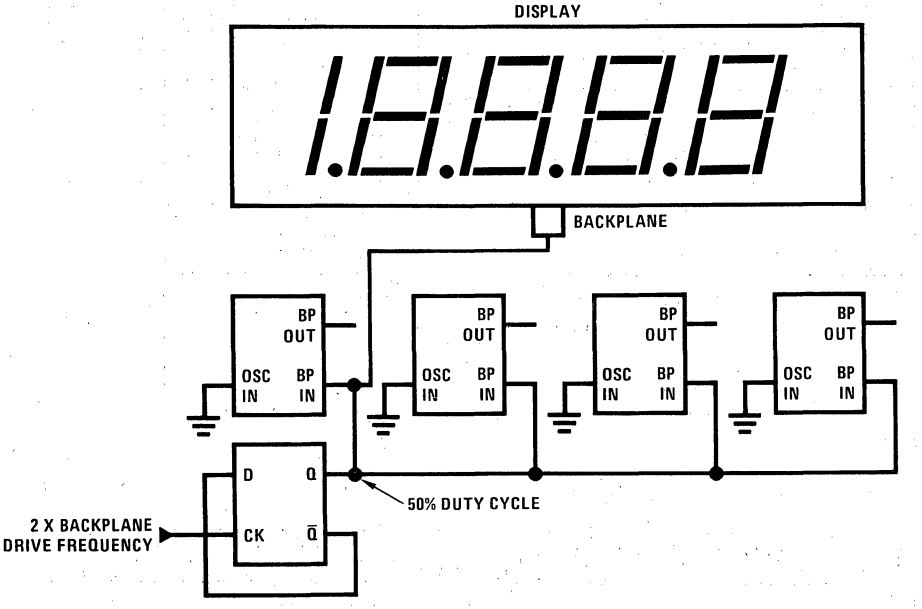



**FIGURE 7. External Backplane Clock**

## Functional Description (Continued)

Using an external clock allows synchronizing the display drive with AC power, internal clocks, or DVM integration time to reduce interference from the display.

*Figure 9* is a general block diagram that shows how the device's serial input can be used to advantage in an analog display. The analog voltage input is compared with a staircase voltage generated by a counter and a digital-to-analog converter or resistor array. The result of this comparison is clocked into the MM5452, MM5453. The next clock pulse increments the staircase and clocks the new data in.

With a buffer amplifier, the same staircase waveform can be used for many displays. The digital-to-analog converter need not be linear; logarithmic or other non-linear functions can be displayed by using weighted resistors or special DACs. This system can be used for status indicators, spectrum analyzers, audio level and power meters, tuning indicators, and other applications.

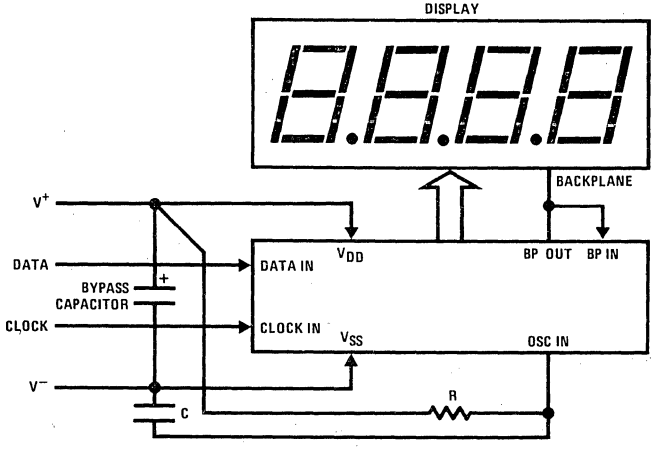
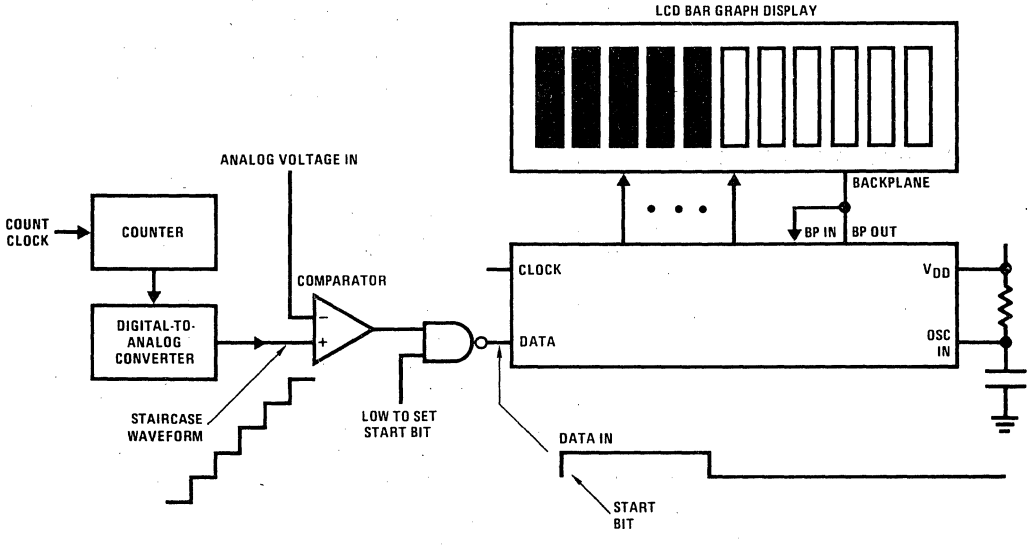



**FIGURE 8. Four Wire Remote Display**




Data is high until staircase>input

**FIGURE 9. Analog Display**

National Semiconductor



# MM5480 LED Display Driver

## General Description

The 5480 is a monolithic MOS integrated circuit utilizing N-channel metal gate low threshold, enhancement mode and ion-implanted depletion mode devices. It utilizes the MM5451 die packaged in a 28-pin package making it ideal for a 3½ digit display. A single pin controls the LED display brightness by setting a reference current through a variable resistor connected either to $V_{DD}$ or to a separate supply of 11V maximum.

## Features

- Continuous brightness control
- Serial data input
- No load signal required
- Wide power supply operation
- TTL compatibility
- Alphanumeric capability
- 3½ digit displays

## Applications

- COPS or microprocessor displays
- Industrial control indicator
- Relay driver
- Digital clock, thermometer, counter, voltmeter
- Instrumentation readouts

## Block Diagram

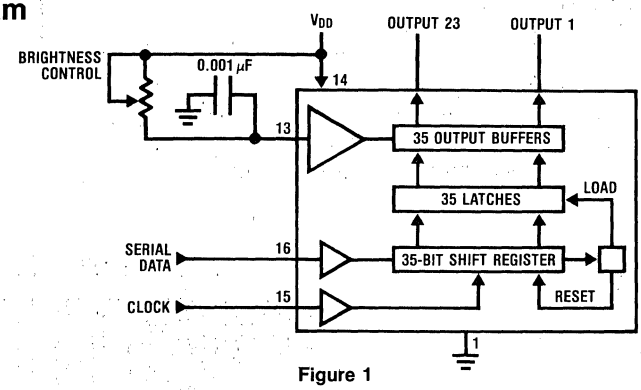
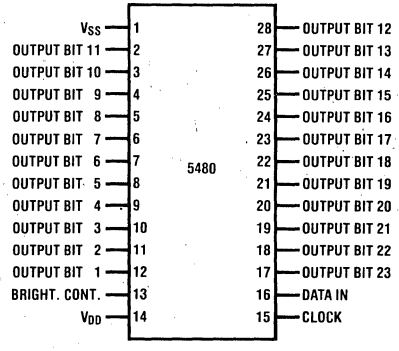



Figure 1

## Connection Diagram (Dual-In-Line Packages)




Order Number MM5480N
See Package 23

Figure 2

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ to $V_{SS}$ + 12V |
| Operating Temperature | −25 °C to +85 °C |
| Storage Temperature | −65 °C to +150 °C |
| Power Dissipation | 490 mW at +85 °C |
| | 940 mW at +25 °C |
| Junction Temperature | +150 °C |
| Lead Temperature (Soldering, 10 seconds) | 300 °C |

## Electrical Characteristics

$T_A$ within operating range, $V_{DD}$ = 4.75 to 11.0V, $V_{SS}$ = 0V, unless otherwise specified.

| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| Power Supply | | 4.75 | | 11.0 | V |
| Power Supply Current | Excluding Output Loads | | | 7.0 | mA |
| Input Voltages | | | | | |
| Logical "0" Level | ±10 μA Input Bias | −0.3 | | 0.8 | V |
| Logical "1" Level | 4.75 ≤ $V_{DD}$ ≤ 5.25 | 2.2 | | $V_{DD}$ | V |
| | $V_{DD}$ > 5.25 | $V_{DD}$ − 2 | | $V_{DD}$ | V |
| Brightness Input (Note 2) | | 0 | | 0.75 | mA |
| Output Sink Current (Note 3) | | | | | |
| Segment OFF | $V_{OUT}$ = 3.0V | | | 10.0 | μA |
| Segment ON | $V_{OUT}$ = 1V (Note 4) | | | | |
| | Brightness Input = 0 μA | 0 | | 10.0 | μA |
| | Brightness Input = 100 μA | 2.0 | 2.7 | 4.0 | mA |
| | Brightness Input = 750 μA | 15.0 | | 25.0 | mA |
| Brightness Input Voltage (Pin 13) | Input Current = 750 μA | 3.0 | | 4.3 | V |
| Input Clock Frequency | | 0 | | 0.5 | MHz |
| Duty Cycle | | 40 | 50 | 60 | % |
| Output Matching (Note 1) | | | | ±20 | % |

**Note 1:** Output matching is calculated as the percent variation from $I_{MAX} + I_{MIN}/2$.

**Note 2:** With a fixed resistor on the brightness input pin some variation in brightness will occur from one device to another.

**Note 3:** Absolute maximum for each output should be limited to 40 mA

**Note 4:** The $V_{OUT}$ voltage should be regulated by the user.

## Functional Description

The MM5480 is specifically designed to operate 3½-digit alphanumeric displays with minimal interface with the display and the data source. Serial data transfer from the data source to the display driver is accomplished with 2 signals, serial data and clock. Using a format of a leading "1" followed by the 35 data bits allows data transfer without an additional load signal. The 35 data bits are latched after the 36th bit is complete, thus providing non-multiplexed, direct drive to the display. Outputs change only if the serial data bits differ from the previous time. Display brightness is determined by control of the output current for LED displays. A 0.001 capacitor should be connected to brightness control, pin 13, to prevent possible oscillations.

A block diagram is shown in *Figure 1.* The output current is typically 20 times greater than the current into pin 13, which is set by an external variable resistor. There is an internal limiting resistor of 400Ω nominal value.

*Figure 4* shows the input data format. A start bit of logical "1" precedes the 35 bits of data. At the 36th clock a LOAD signal is generated synchronously with the high state of the clock, which loads the 35 bits of the shift registers into the latches. At the low state of the clock a RESET signal is generated which clears all the shift registers for the next set of data. The shift registers are static master-slave configuration. There is no clear for the master portion of the first shift register, thus allowing continuous operation.

There must be a complete set of 36 clocks or the shift registers will not clear.

When the chip first powers ON an internal power ON reset signal is generated which resets all registers and all latches. The START bit and the first clock return the chip to its normal operation.

*Figure 5* shows the Output Data Format for the 5480. Because it uses only 23 of the possible 35 outputs, 12 of the bits are 'Don't Cares'.

*Figure 3* shows the timing relationships between data, clock, and data enable. A maximum clock frequency of 0.5 MHz is assumed.

For applications where a lesser number of outputs are used, it is possible to either increase the current per output, or operate the part at higher than 1V $V_{OUT}$. The following equation can be used for calculations.

$T_j = (V_{OUT})(I_{LED})$ (No. of segments) (132 °C/W) + $T_A$

## Functional Description (Continued)

where:

$T_j$ = junction temperature + 150°C max.
$V_{OUT}$ = the voltage at the LED driver outputs
$I_{LED}$ = the LED current
132°C/W = thermal coefficient of the package
$T_A$ = ambient temperature

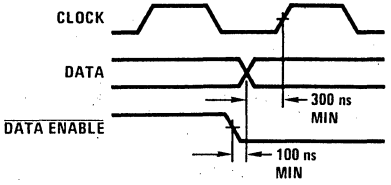



**Figure 3**

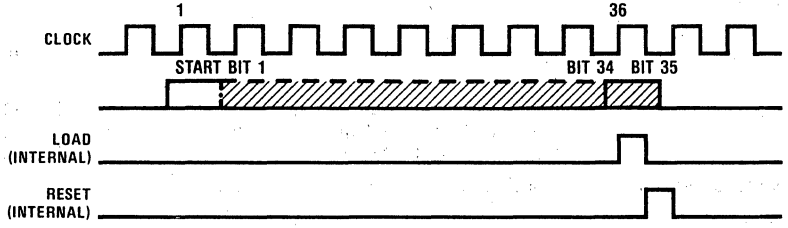
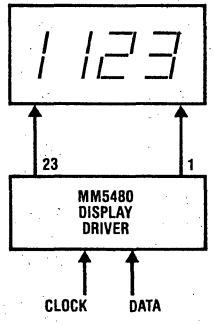



**Figure 4. Input Data Format**

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5451 | 35 | 34 | 33 | 32 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | START |
| 5480 | X | 23 | 22 | 21 | 20 | 19 | X | X | 18 | X | 17 | 16 | 15 | 14 | 13 | 12 | X | X | X | X | 11 | 10 | 9 | 8 | X | X | X | 7 | 6 | 5 | 4 | 3 | 2 | 1 | X | START |

**Figure 5. Output Data Format**




**Basic 3½ Digit Interface**

National Semiconductor

# MM5481 LED Display Driver

## General Description

The 5481 is a monolithic MOS integrated circuit utilizing N-channel metal gate low threshold, enhancement mode and ion-implanted depletion mode devices. It utilizes the MM5450 die packaged in a 20-pin package making it ideal for a 2 digit display. A single pin controls the LED display brightness by setting a reference current through a variable resistor connected either to $V_{DD}$ or to a separate supply of 11V maximum.

- Wide power supply operation
- TTL compatibility
- Alphanumeric capability
- 2 digit LED driver

## Features

- Continuous brightness control
- Serial data input
- No load signal required
- Data enable

## Applications

- COPS or microprocessor displays
- Industrial control indicator
- Relay driver
- Instrumentation readouts

## Block Diagram

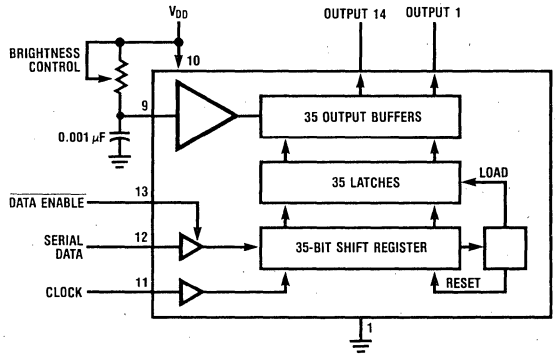
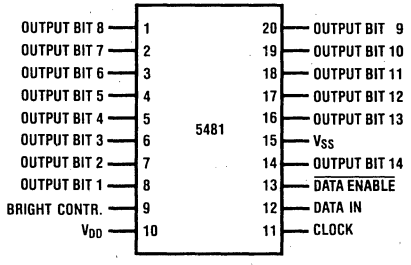



Figure 1

## Connection Diagram

(Dual-In-Line Package)




Top View
Order Number MM5481N
See Package 20A

Figure 2

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ to $V_{SS}+12V$ |
| Operating Temperature | −25°C to +85°C |
| Storage Temperature | −65°C to +150°C |
| Power Dissipation | 450mW at +85°C |
| | 860mW at +25°C |
| Junction Temperature | +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, $V_{DD}=4.75$ to 11.0V, $V_{SS}=0V$, unless otherwise specified.

| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| Power Supply | | 4.75 | | 11 | V |
| Power Supply Current | Excluding Output Loads | | | 7 | mA |
| Input Voltages | | | | | |
| Logical "0" Level | ±10μA Input Bias | −0.3 | | 0.8 | V |
| Logical "1" Level | $4.75 \leq V_{DD} \leq 5.25$ | 2.2 | | $V_{DD}$ | V |
| | $V_{DD} > 5.25$ | $V_{DD}-2$ | | $V_{DD}$ | V |
| Brightness Input (Note 2) | | 0 | | 0.75 | mA |
| Output Sink Current (Note 3) | | | | | |
| Segment OFF | $V_{OUT}=3.0V$ | | | 10 | μA |
| Segment ON | $V_{OUT}=1V$ (Note 4) | | | | |
| | Brightness Input = 0μA | 0 | | 10 | μA |
| | Brightness Input = 100μA | 2.0 | 2.7 | 4.0 | mA |
| | Brightness Input = 750μA | 15 | | 25 | mA |
| Brightness Input Voltage (Pin 9) | Input Current = 750μA | 3.0 | | 4.3 | V |
| Input Clock Frequency | | 0 | | 0.5 | MHz |
| Duty Cycle | | 40 | 50 | 60 | % |
| Output Matching (Note 1) | | | | ±20 | % |

**Note 1:** Output matching is calculated as the percent variation from $I_{MAX}+I_{MIN}/2$.

**Note 2:** With a fixed resistor on the brightness input pin some variation in brightness will occur from one device to another.

**Note 3:** Absolute maximum for each output should be limited to 40mA

**Note 4:** The $V_{OUT}$ voltage should be regulated by the user.

## Functional Description

The MM5481 uses the 5450 die which is packaged to operate 2-digit alphanumeric displays with minimal interface with the display and the data source. Serial data transfer from the data source to the display driver is accomplished with 2 signals, serial data and clock. Using a format of a leading "1" followed by the 35 data bits allows data transfer without an additional load signal. The 35 data bits are latched after the 36th bit is complete, thus providing non-multiplexed, direct drive to the display. Outputs change only if the serial data bits differ from the previous time. Display brightness is determined by control of the output current for LED displays. A 0.001 capacitor should be connected to brightness control, pin 9, to prevent possible oscillations.

A block diagram is shown in *Figure 1.* The output current is typically 20 times greater than the current into pin 9, which is set by an external variable resistor. There is an internal limiting resistor of 400Ω nominal value.

*Figure 4* shows the input data format. A start bit of logical "1" precedes the 35 bits of data. At the 36th clock a LOAD signal is generated synchronously with the high state of the clock, which loads the 35 bits of the shift registers into the latches. At the low state of the clock a RESET signal is generated which clears all the shift registers for the next set of data. The shift registers are static master-slave configuration. There is no clear for the master portion of the first shift register, thus allowing continuous operation.

There must be a complete set of 36 clocks or the shift registers will not clear.

When the chip first powers ON an internal power ON reset signal is generated which resets all registers and all latches. The START bit and the first clock return the chip to its normal operation.

*Figure 5* shows the Output Data Format for the 5481. Because it uses only 14 of the possible 34 outputs, 20 of the bits are 'Don't Cares'. Note that only alternate groups of 4 outputs are used.

*Figure 3* shows the timing relationships between data, clock, and data enable. A maximum clock frequency of 0.5MHz is assumed.

## Functional Description (Continued)

For applications where a lesser number of outputs are used, it is possible to either increase the current per output, or operate the part at higher than 1V $V_{OUT}$. The following equation can be used for calculations.

$T_j = (V_{OUT})\ (I_{LED})$ (No. of segments) (145 °C/W) + $T_A$

where:

$T_j$ = junction temperature + 150 °C max.
$V_{OUT}$ = the voltage at the LED driver outputs
$I_{LED}$ = the LED current
145 °C/W = thermal coefficient of the package
$T_A$ = ambient temperature

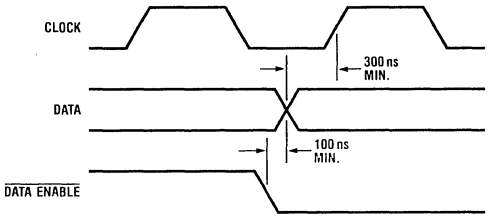



Figure 3

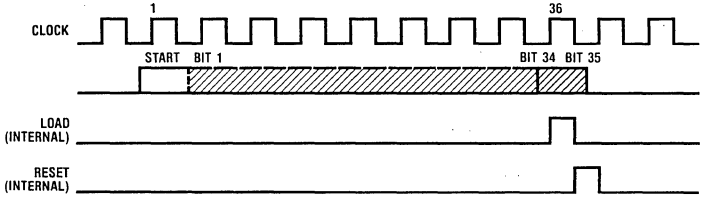
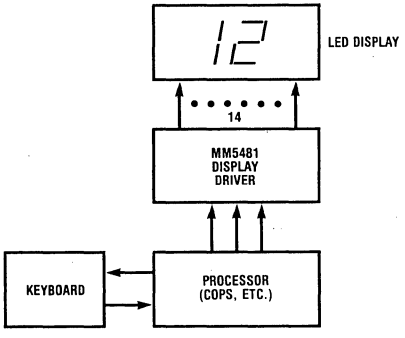



Figure 4. Input Data Format

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5450 | 34 | 33 | 32 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | START |
| 5481 | X | X | X | X | 14 | 13 | X | X | X | X | 12 | 11 | 10 | 9 | X | X | X | X | 8 | 7 | 6 | 5 | X | X | X | X | 4 | 3 | 2 | 1 | X | X | X | X | START |

Figure 5. Output Data Format




Basic Electronically Tuned Television System

National Semiconductor



# MM58201 Multiplexed LCD Driver

## General Description

The MM58201 is a monolithic CMOS LCD driver capable of driving up to 8 backplanes and 24 segments. A 192-bit RAM stores the data for the display. Serial input and output pins are provided to interface with a controller. An RC oscillator generates the timing necessary to refresh the display. The magnitude of the driving waveforms can be adjusted with the $V_{TC}$ input to optimize display contrast. Four additional bits of RAM allow the user to program the number of backplanes being driven, and to designate the driver as either a master or slave for cascading purposes. When two or more drivers are cascaded, the master chip drives the backplane lines, and the master and each slave chip drive 24 segment lines. Synchronizing the cascaded drivers is accomplished by tying the RC OSC pins together and the BP1 pins together.

The MM58201 is packaged in a 40-lead dual-in-line package.

## Features

- Drives up to 8 backplanes and 24 segment lines
- Stores data for display
- Cascadable
- Low power
- Fully static operation

## Applications

- Dot matrix LCD driver
- Multiplexed 7-segment LCD driver
- Serial in/serial out memory

## Block Diagram

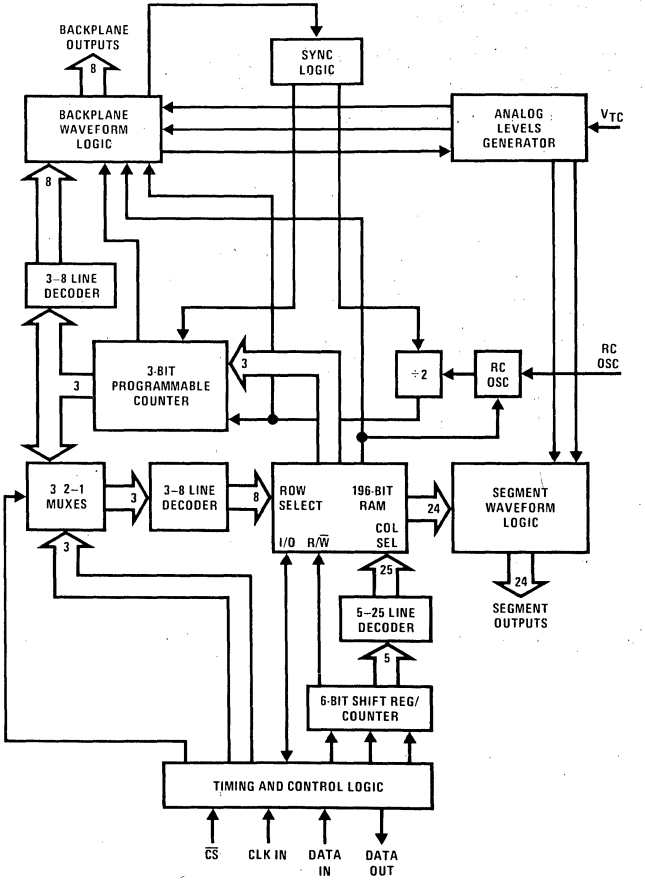
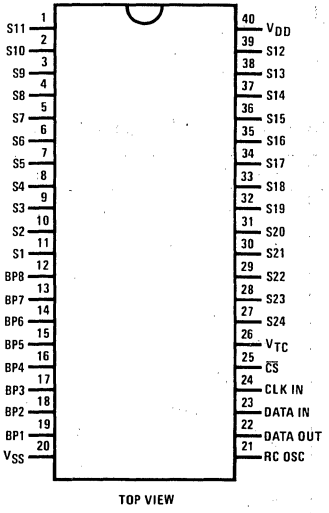



FIGURE 1

## Connection Diagram

Dual-In-Line Package




TOP VIEW

Order Number MM58201N
See Package 24

FIGURE 2

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS} - 0.3V$ to $V_{SS} + 18V$ |
| Operating Temperature Range | 0°C to 50°C |
| Storage Temperature Range | −65°C to +150°C |
| Package Dissipation | 500 mW |
| Operating $V_{DD}$ Range | $V_{SS} + 9.0V$ to $V_{SS} + 18.0V$ |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## DC Electrical Characteristics Min/max limits apply across temperature range unless otherwise noted.

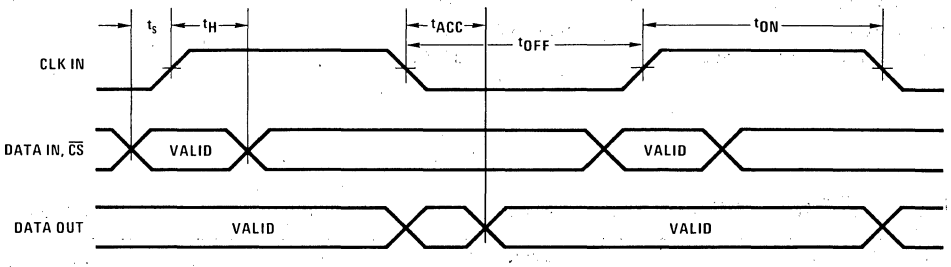
| | Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{CC}$ | Supply Current | $V_{DD} = 18V$ | | | 0.25 | mA |
| $V_{IN(1)}$ | Logical "1" Input Voltage | $V_{DD} = 9V$<br>$V_{DD} = 15V$ | 4.0<br>6.5 | | | V<br>V |
| $V_{IN(0)}$ | Logical "0" Input Voltage | | | | 1.0 | V |
| $V_{OUT(0)}$ | Logical "0" Output Voltage | $I_{SINK} = 2.5$ mA | | | 0.4 | V |
| $I_{OUT(1)}$ | Logical "1" Output Current | $V_{DD} = 18V$ | 0 | | ±10 | μA |
| $I_{IN(1)}$ | Logical "1" Input Current | $V_{IN} = 18V$ | | | 1.0 | μA |
| $I_{IN(0)}$ | Logical "0" Input Current | $V_{IN} = 0V$ | −1.0 | | | μA |
| $V_{TC}$ | Input Voltage | | 4.5 | | | V |
| $V_{TC}$ | Input Impedance | | 15 | | 30 | kΩ |
| $Z_{OUT}$ | Output Impedance | Backplane and Segment Outputs | | | 12.5 | kΩ |
| | DC Offset Voltage | Between Any Backplane and Segment Output | 0 | | ±10 | mV |

## AC Electrical Characteristics $T_A$ and $V_{DD}$ within operating range unless otherwise noted.

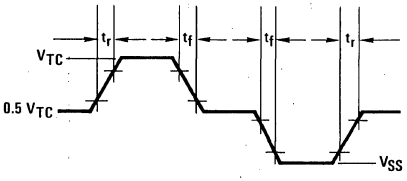
| | Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $f_{OSC}$ | Oscillator Frequency* | | $128\eta$ | | $400\eta$ | Hz |
| $f_{CLK\ IN}$ | Clock Frequency | | DC | | 275 | kHz |
| $t_{ON}$ | Clock Pulse Width | | 1.8 | | | μs |
| $t_{OFF}$ | Clock OFF Time | | 1.8 | | | μs |
| $t_s$ | Input Data Set-Up Time | | 200 | | | ns |
| $t_H$ | Input Data Hold Time | | 100 | | | ns |
| $t_{ACC}$ | Access Time | | 1.0 | | | μs |
| $t_r$ | Rise Time | Backplane, Segment Outputs $C_L = 2000$ pF | | | 60 | μs |
| $t_f$ | Fall Time | Backplane, Segment Outputs $C_L = 2000$ pF | | | 60 | μs |

* $\eta$ is the number of backplanes programmed.

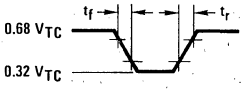
## Switching Time Waveforms




**Backplane Output**




**Segment Output**




## Functional Description

A block diagram of the MM58201 LCD driver is shown in *Figure 1*. A connection diagram is shown in *Figure 2*.

### Serial Inputs and Output

A negative-going edge on the $\overline{CS}$ input initiates a frame. The $\overline{CS}$ input must stay low for one rising edge of CLK IN, however it may not be pulsed low more than once in 31 clocks. At least one clock must occur while $\overline{CS}$ is high.

CLK IN latches data from the DATA IN input on its rising edge. Data from the DATA OUT pin changes on the falling edge of CLK IN and is valid before the next rising edge.

The first five bits of data following $\overline{CS}$ are the address bits *(Figure 3)*. The address selects the column where the operation is to start. Bit 1 is the MSB and bit 5 is the LSB. The sixth bit is the read/write bit. A logic "1" specifies a read operation and a logic "0" specifies a write operation. The next 24 bits are the data bits. The first data bit corresponds to the BP1 row of the display, the second data bit to the BP2 row, and so on. After the eighth and sixteenth data bits, the column pointer is incremented. When starting address 10110 or 10111 is specified, the column pointer increments from 10111 to 00000.

The DATA OUT output is an open drain N-channel device to $V_{SS}$ *(Figure 4)*. With an external pull-up this configuration allows the controller to operate at a lower supply voltage, and also permits the DATA OUT output to be wired in parallel with the DATA OUT outputs from any other drivers in the system.

To program the number of backplanes being driven and the M/$\overline{S}$ bit, load address 11000, a write bit, three bits for the number of backplanes (Table I), and the M/$\overline{S}$ bit. The remaining 20 data bits will be ignored but it is necessary to provide 21 more clocks before initiating another frame.

**TABLE I. BACKPLANE SELECT**

| Number of Backplanes | B2 | B1 | B0 |
|---|---|---|---|
| 2 | 0 | 0 | 1 |
| 3 | 0 | 1 | 0 |
| 4 | 0 | 1 | 1 |
| 5 | 1 | 0 | 0 |
| 6 | 1 | 0 | 1 |
| 7 | 1 | 1 | 0 |
| 8 | 1 | 1 | 1 |

## Functional Description (Continued)

### RC OSC Pin

This oscillator generates the timing required for multiplexing the liquid crystal display. The oscillator operates at a frequency that is $4\eta$ times the refresh rate of the display, where $\eta$ is the number of backplanes programmed. Since the refresh rate should be in the range from 32 Hz to 100 Hz, the oscillator frequency must be:

$$128\eta \leq f_{OSC} \leq 400\eta$$

The frequency of oscillation is related to the external R and C components in the following way:

$$f_{OSC} = \frac{1}{1.25\ RC} \pm 30\%$$

The value used for the external resistor should be in the range from 10 kΩ to 1 MΩ.

The value used for the external capacitor should be less than 0.02 μF.

### $V_{TC}$ Pin

The $V_{TC}$ pin is an analog input that controls the contrast of the segments on the LCD. If eight backplanes are being driven ($\eta = 8$), a voltage of typically 8V is required at 25°C. The voltage for optimum contrast will vary from display to display. It also has a significant negative temperature coefficient.

The voltage source on the $V_{TC}$ input must be of relatively low impedance since the input impedance of $V_{TC}$ ranges from 15 kΩ to 30 kΩ. A suitable circuit is shown in *Figure 5.*

In a standby mode, the $V_{TC}$ input can be set to $V_{SS}$. This reduces the supply current to less than 250 μA per driver.

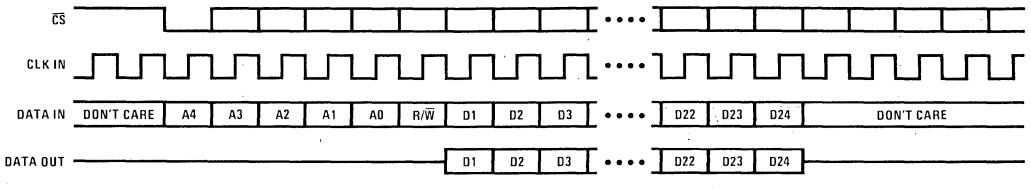
### Backplane and Segment Outputs

Connect the backplane and segment outputs directly to the LCD row and column lines. The outputs are designed to drive a display with a total ON capacitance of up to 2000 pF.

The output structure consists of transmission gates tapped off of a resistor divider driven by $V_{TC}$ *(Figure 6).*

A critical factor in the lifetime of an LCD is the amount of DC offset between a backplane and segment signal. Typically, 50 mV of offset is acceptable. The MM58201 guarantees an offset of less than 10 mV.

The BP1 output is disabled when the M/$\overline{S}$ bit is set to zero. This allows the BP1 output from the master chip to be connected directly to it so that synchronizing signals can be generated. Synchronization occurs once each refresh cycle, so the cascaded chips are assured of remaining synchronized.




| | S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 | S9 | S10 | S11 | S12 | S13 | S14 | S15 | S16 | S17 | S18 | S19 | S20 | S21 | S22 | S23 | S24 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| BP1 | | | | | | | | | | | | | D1 | D9 | D17 | | | | | | | | | | B2 |
| BP2 | | | | | | | | | | | | | D2 | D10 | D18 | | | | | | | | | | B1 |
| BP3 | | | | | | | | | | | | | D3 | D11 | D19 | | | | | | | | | | B0 |
| BP4 | | | | | | | | | | | | | D4 | D12 | D20 | | | | | | | | | | M/$\overline{S}$ |
| BP5 | | | | | | | | | | | | | D5 | D13 | D21 | | | | | | | | | | |
| BP6 | | | | | | | | | | | | | D6 | D14 | D22 | | | | | | | | | | |
| BP7 | | | | | | | | | | | | | D7 | D15 | D23 | | | | | | | | | | |
| BP8 | | | | | | | | | | | | | D8 | D16 | D24 | | | | | | | | | | |

| | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| A3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| A2 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| A1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| A0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |

Diagram above shows where data will appear on display if starting address 01100 is specified in data format.

**FIGURE 3. Data Format**

## Functional Description (Continued)

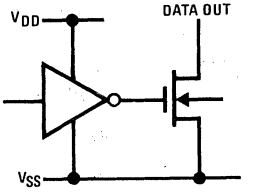



FIGURE 4. DATA OUT Structure

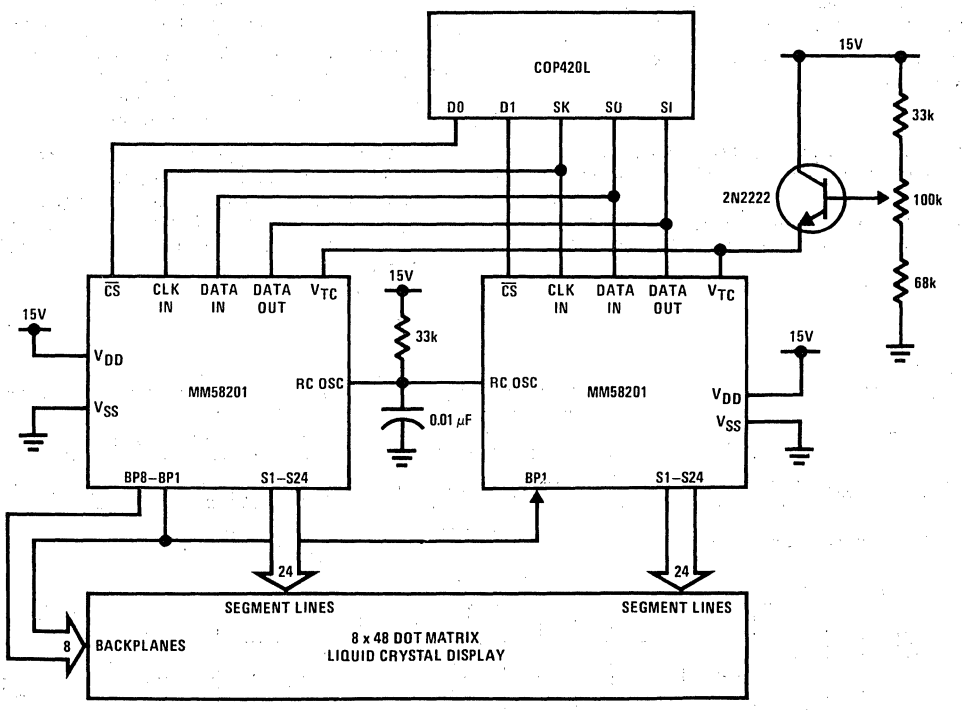



FIGURE 5. Typical Application

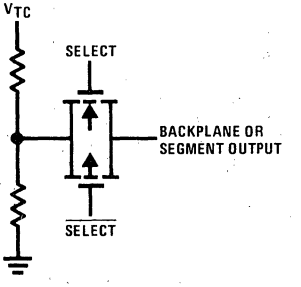



FIGURE 6. Structure of LCD Outputs

# Oscillators

National Semiconductor

# MM5368 CMOS Oscillator Divider Circuit

## General Description

The MM5368 is a CMOS integrated circuit generating 50 or 60 Hz, 10 Hz, and 1 Hz outputs from a 32 kHz crystal (32,768 Hz). For the 60 Hz selected output the input time base is divided by 546.133, for the 50 Hz mode it is divided by 655.36. The 50/60 Hz output is then divided by 5 or 6 to obtain a 10 Hz output which is further divided to obtain a 1 Hz output. The 50/60 Hz select input can be floated for a counter reset.

## Features

- 50/60 Hz output
- 1 Hz output
- 10 Hz output
- Low power dissipation
- Fully static operation
- Counter reset
- 3V–15V supply range
- On-chip oscillator – tuning and load capacitors are the only required external components besides the crystal. (For operation below 5V it may be necessary to use an ~ 1MΩ pullup on the oscillator output to insure start-up.)

## Block Diagram

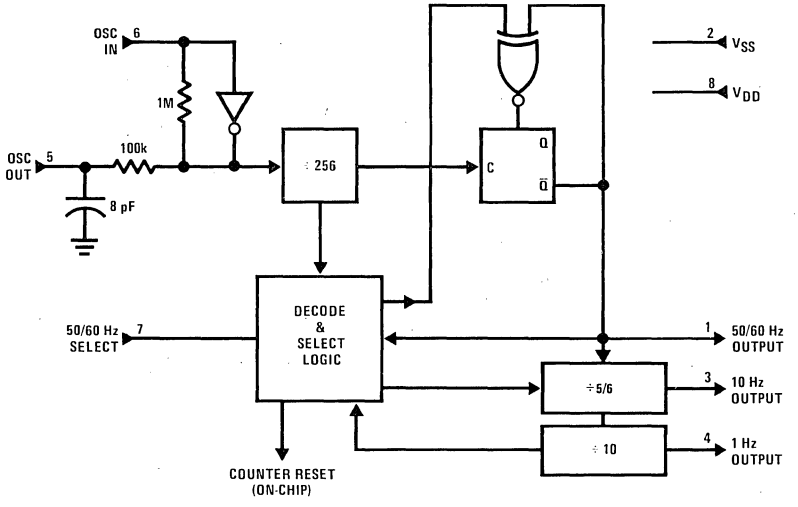
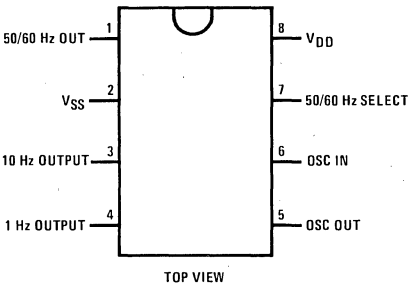



FIGURE 1

## Connection Diagram

Dual-In-Line Package




Order Number MM5368N
See Package 17

FIGURE 2. Pin-Out

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $-0.3V$ to $V_{DD} + 0.3V$ |
| Operating Temperature | $0°C$ to $+70°C$ |
| Storage Temperature | $-65°C$ to $+150°C$ |
| Maximum $V_{DD}$ Voltage | 16V |
| Operating $V_{DD}$ Range | $3V \leq V_{DD} \leq 15V$ |
| Lead Temperature (Soldering, 10 seconds) | $300°C$ |

## Electrical Characteristics

$T_A$ within operating range, $V_{SS} = 0V$

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Quiescent Current Drain | $V_{DD} = 15V$; 50/60 Select Floating | | | 10 | μA |
| Operating Current Drain | $f_{IN} = 32$ kHz, $V_{DD} = 3V$ | | | 50 | μA |
| | $f_{IN} = 32$ kHz, $V_{DD} = 15V$ | | | 1500 | μA |
| Maximum Input Frequency | $V_{DD} = 3V$ | | | 64 | kHz |
| | $V_{DD} = 15V$ | | | 500 | kHz |
| Output Current Levels | $V_{DD} = 5V$ | | | | |
| Logical "1", Source | $V_{OH} = V_{SS} + 2.7V$ | | | −400 | μA |
| Logical "0", Sink | $V_{OL} = V_{SS} + 0.4V$ | 400 | | | μA |
| | $V_{DD} = 9V$ | | | | |
| Logical "1", Source | $V_{OH} = V_{SS} + 6.7V$ | | | −1500 | μA |
| Logical "0", Sink | $V_{OL} = V_{SS} + 0.4V$ | 1500 | | | μA |
| Input Current Levels | 50/60 Select Input | | | | |
| Logical "1" ($I_{IH}$) | $V_{DD} = 3V$, $V_{IN} \geq 0.9V_{DD}$ | | | 50 | μA |
| Logical "1" ($I_{IH}$) | $V_{DD} = 15V$, $V_{IN} \geq 0.9V_{DD}$ | | | 3 | mA |
| Logical "0" ($I_{IL}$) | $V_{DD} = 3V$, $V_{IN} \leq 0.1V_{DD}$ | | | 20 | μA |
| Logical "0" ($I_{IL}$) | $V_{DD} = 15V$, $V_{IN} \leq 0.1V_{DD}$ | | | 1 | mA |

## Functional Description *(Figure 1)*

The MM5368 initially divides the input time base by 256. From the resulting frequency (128 Hz for 32 kHz crystal) 8 clock periods are dropped or eliminated during 60 Hz operation and 28 clock periods are eliminated during 50 Hz operation. This frequency is then divided by 2 to obtain a 50 or 60 Hz output. This output is not periodic from cycle to cycle; however, the waveform repeats itself every second. Straight divide by 5 or 6 and 10 are used to obtain the 10 Hz output and the 1 Hz outputs.

The 60 Hz mode is obtained by tying pin 7 to $V_{DD}$. The 60 Hz output waveform can be seen in *Figure 3.* The 10 Hz and 1 Hz outputs have an approximate 50% duty cycle. In the 50 Hz mode the 50/60 select input is tied to $V_{SS}$. The 50 Hz output waveform can be seen in *Figure 3.* The 10 Hz output has an approximate 40% duty cycle and the 1 Hz output has an approximate 50% duty cycle.

For the 50/60 Hz select input floating, the counter chain is held reset, except for the initial toggle flip-flop which is needed for the reset function. A reset may also occur when the input is switched *(Figure 4).* To insure the floating state, current sourced from the input must be limited to 1.0 μA and current sunk by the input must be limited to 1.0 μA for $V_{DD} = 3V$.

## Timing Diagrams

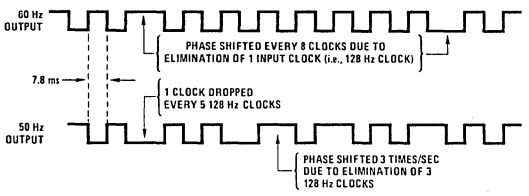



FIGURE 3. 50/60 Hz Output

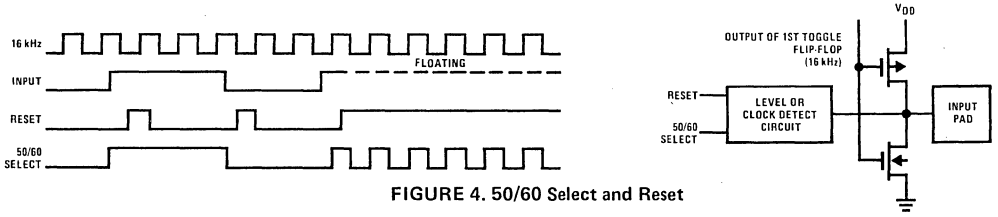
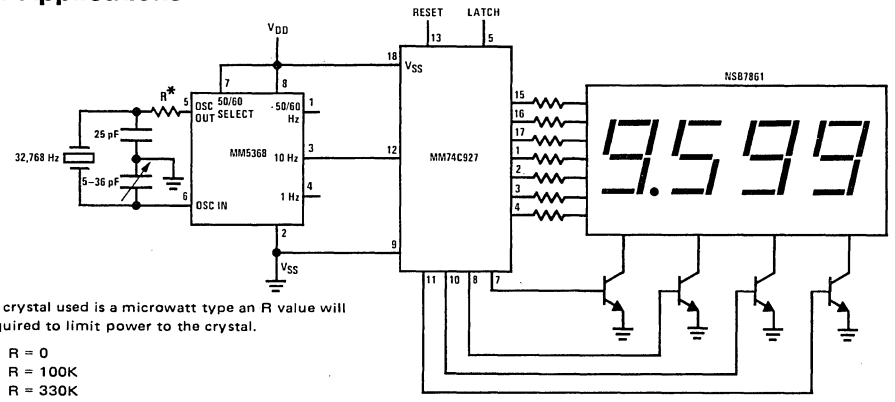



FIGURE 4. 50/60 Select and Reset

## Typical Applications




*If the crystal used is a microwatt type an R value will be required to limit power to the crystal.

3V R = 0
5V R = 100K
10V R = 330K

FIGURE 5. 10 Minute (9:59.9) Timer

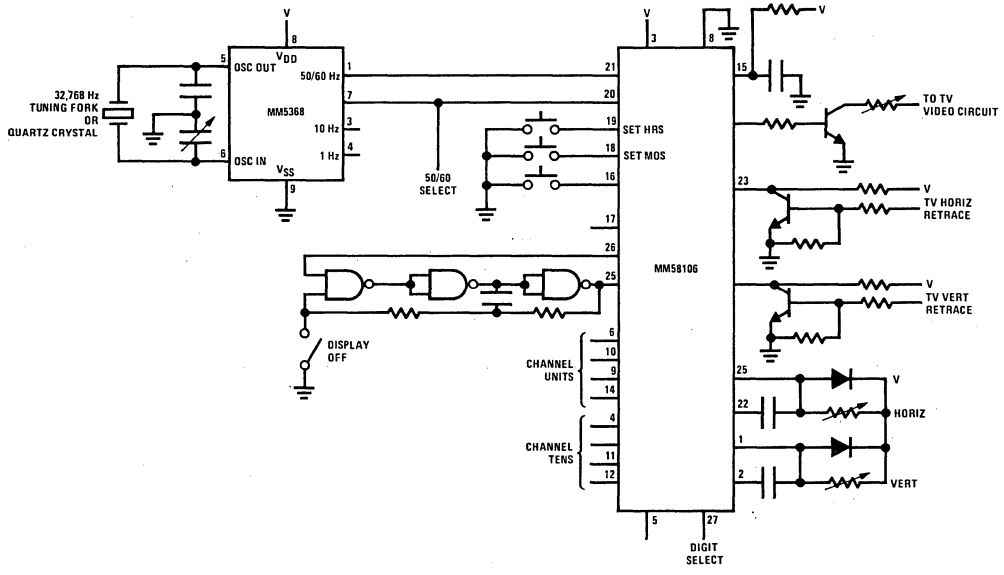



FIGURE 6. TV Time/Channel Display

National Semiconductor



# MM5369 Series 17 Stage Oscillator/Divider

## General Description

The MM5369 is a CMOS integrated circuit with 17 binary divider stages that can be used to generate a precise reference from commonly available high frequency quartz crystals. An internal pulse is generated by mask programming the combinations of stages 1 through 4, 16 and 17 to set or reset the individual stages. The MM5369 is advanced one count on the positive transition of each clock pulse. Two buffered outputs are available: the crystal frequency for tuning purposes and the 17th stage output. The MM5369 is available in an 8-lead dual-in-line epoxy package.

## Features

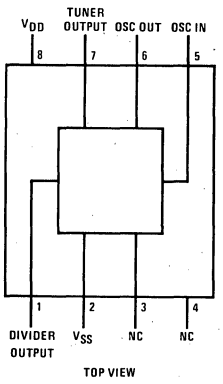
- Crystal oscillator
- Two buffered outputs
  - Output 1 crystal frequency
  - Output 2 full division
- High speed (4 MHz at $V_{DD}$ = 10V)
- Wide supply range 3–15V
- Low power
- Fully static operation
- 8 lead dual-in-line package
- Low current

## Options

| | |
|---|---|
| ■ MM5369AA | 3.58 MHz to 60 Hz |
| ■ MM5369EYR | 3.58 MHz to 50 Hz |
| ■ MM5369EST | 3.58 MHz to 100 Hz |

## Connection Diagram

Dual-In-Line Package




**Order Number MM5369AA/N, MM5369EYR/N, MM5369EST/N**
**See Package 17**

FIGURE 1

## Block Diagram

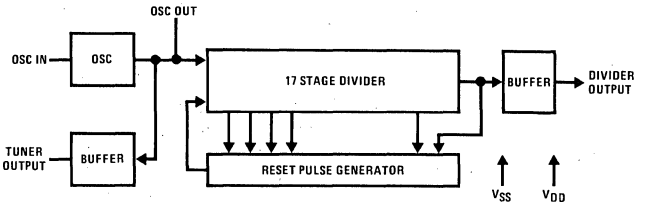



FIGURE 2

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | −0.3V to $V_{DD}$ +0.3V |
| Operating Temperature | 0°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Package Dissipation | 500 mW |
| Maximum $V_{CC}$ Voltage | 16V |
| Operating $V_{CC}$ Range | 3V to 15V |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating temperature range, $V_{SS}$ = GND, 3V $\leq V_{DD} \leq$ 15V unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Quiescent Current Drain | $V_{DD}$ = 15V | | | 10 | μA |
| Operating Current Drain | $V_{DD}$ = 10V, $f_{IN}$ = 4.19 MHz | | 1.2 | 2.5 | mA |
| Frequency of Oscillation | $V_{DD}$ = 10V | DC | | 4.5 | MHz |
| | $V_{DD}$ = 6V | DC | | 2 | MHz |
| Output Current Levels | $V_{DD}$ = 10V | | | | |
| | $V_O$ = 5V | | | | |
| Logical "1" Source | | 500 | | | μA |
| Logical "0" Sink | | 500 | | | μA |
| Output Voltage Levels | $V_{DD}$ = 10V | | | | |
| | $I_O$ = 10 μA | | | | |
| Logical "1" | | 9.0 | | | V |
| Logical "0" | | | | 1.0 | V |

## Functional Description

A connection diagram for the MM5369 is shown in *Figure 1* and a block diagram is shown in *Figure 2*.

### TIME BASE

A precision time base is provided by the interconnection of a 3,579,545 Hz quartz crystal and the RC network shown in *Figure 3* together with the CMOS inverter/amplifier provided between the OSC IN and the OSC OUT terminals. Resistor R1 is necessary to bias the inverter for class A amplifier operation. Capacitors C1 and C2 in series provide the parallel load capacitance required for precise tuning of the quartz crystal.

The network shown provides > 100 ppm tuning range when used with standard crystals trimmed for $C_L$ = 12 pF. Tuning to better than ±2 ppm is easily obtainable.

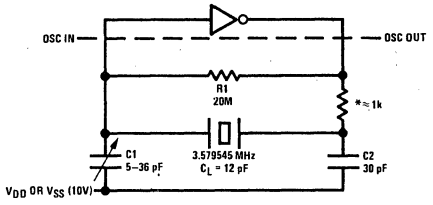
### DIVIDER

A pulse is generated when divider stages 1 through 4, 16 and 17 are in the correct state. By mask options, this pulse is used to set or reset individual stages of the counter. *Figure 4* shows the relationship between the duty cycle and the programmed modulus.

### OUTPUTS

The Tuner Output is a buffered output at the crystal oscillator frequency. This output is provided so that the crystal frequency can be obtained without disturbing the crystal oscillator. The Divide Output is the input frequency divided by the mask programmed number. Both outputs are push-pull outputs. A typical application of the MM5369 is shown in *Figure 5*.

## Functional Description (Continued)




*To be selected based on xtal used

FIGURE 3. Crystal Oscillator Network

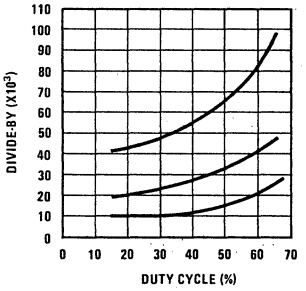



FIGURE 4. Plot of Divide-By vs Duty Cycle

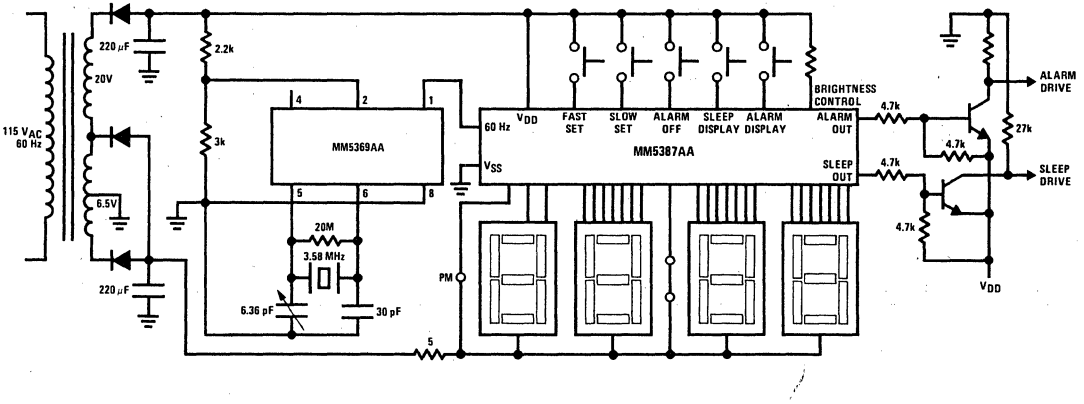



FIGURE 5. Typical Application

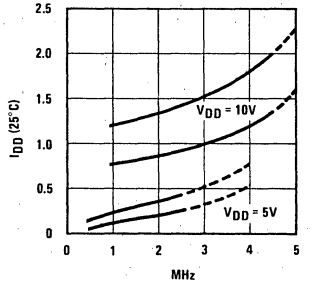



FIGURE 6. Typical Current Drain vs Oscillator Frequency

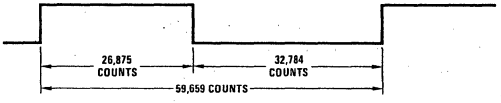



FIGURE 7. Output Waveform for Standard MM5369AA

# MM53107 Series 17-Stage Oscillator/Divider

## General Description

The MM53107 is a low threshold voltage CMOS integrated circuit with 17 binary divider stages that can be used to generate a precise reference from a 2.097152 MHz quartz crystal. An internal pulse is generated by the combinations of stages 1–4, 16 and 17 to set or reset the individual stages. The MM53107 is advanced one count on the positive transition of each clock pulse. One buffered output is available: the 17th stage 60 or 100 Hz output. The MM53107 is available in an 8-lead dual-in-line epoxy package.

**Options**

| | |
|---|---|
| MM53107AA | 2.09 MHz to 60 Hz |
| MM53107FDU | 2.09 MHz to 100 Hz |

## Features

- Input frequency—2.097152 MHz
- Output frequency—60Hz or 100 Hz
- Crystal oscillator
- High speed (2 MHz at $V_{DD}$ = 2.5V)
- Wide supply range 2–6V
- Low power (0.5 mW @ 2 MHz/2.5V)
- Fully static operation
- 8-lead dual-in-line package

## Block and Connection Diagrams

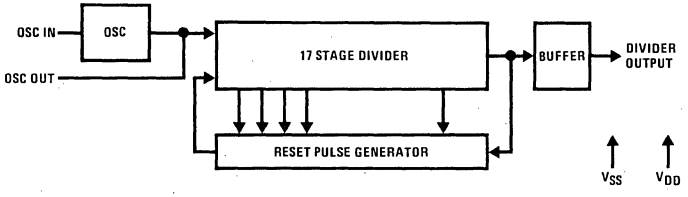



FIGURE 1

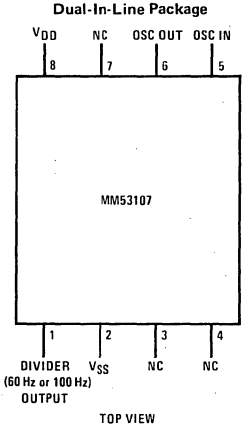
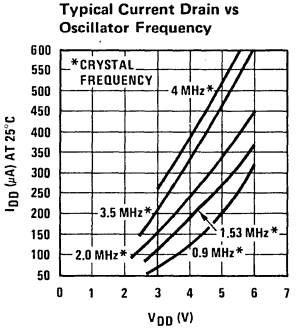



FIGURE 2

## Typical Performance Characteristics

Order Number MM53107AA/N or MM53107FDU/N See Package 17



## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | −0.3V to $V_{CC}$ + 0.3V |
| Operating Temperature | 0°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Package Dissipation | 500 mW |
| Maximum $V_{CC}$ Voltage | 7V |
| Operating $V_{CC}$ Range | 2.5V to 6V |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating temperature range, $V_{SS}$ = Gnd, 2.5V ≤ $V_{DD}$ ≤ 6V unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Quiescent Current Drain | $V_{DD}$ = 6V | | | 10 | μA |
| Operating Current Drain | $V_{DD}$ = 2.5V, $f_{IN}$ = 2.1 MHz | | | 200 | μA |
| Frequency of Oscillation | $V_{DD}$ = 2.4V | dc | | 2.1 | MHz |
| | $V_{DD}$ = 6V | dc | | 4.0 | MHz |
| Output Current Levels | | | | | |
| Logical "1" Source | $V_{DD}$ = 4V, | 100 | | | μA |
| Logical "0" Sink | $V_{OUT}$ = 2V | 100 | | | μA |
| Output Voltage Levels | | | | | |
| Logical "1" | $V_{DD}$ = 6V $I_O$Source = 10μA | 5.0 | | | V |
| Logical "0" | $I_O$Sink = −10μA | | | 1.0 | V |

## Functional Description

A connection diagram for the MM53107 is shown in *Figure 2* and a block diagram is shown in *Figure 1.*

### TIME BASE

A precision time base is provided by the interconnection of a 2,097,152 Hz quartz crystal and the RC network shown in *Figure 3* together with the CMOS inverter/amplifier provided between the Osc In and the Osc Out terminals. Resistor R1 is necessary to bias the inverter for class A amplifier operation. Capacitors C1 and C2 in series provide the parallel load capacitance required for precise tuning of the quartz crystal.

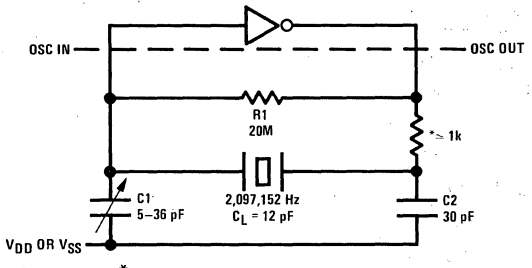
The network shown provides > 100 ppm tuning range when used with standard crystals trimmed for $C_L$ = 12 pF. Tuning to better than ±2 ppm is easily obtainable.

### DIVIDER

A pulse is generated when divider stages 1–4, 16 and 17 are in the correct state. This pulse is used to set or reset individual stages of the counter, the modulus of the counter is 34,952 to provide 60 Hz.

### OUTPUT

The Divide Output is the input frequency divided by 34,952. The output is a push-pull output. A typical application of the MM53107 is shown in *Figure 5.*




*To be selected based on the crystal used

**FIGURE 3. Crystal Oscillator Network**

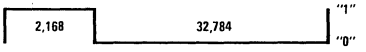



**FIGURE 4. Duty Cycle for MM53107AA**

## Functional Description (Continued)

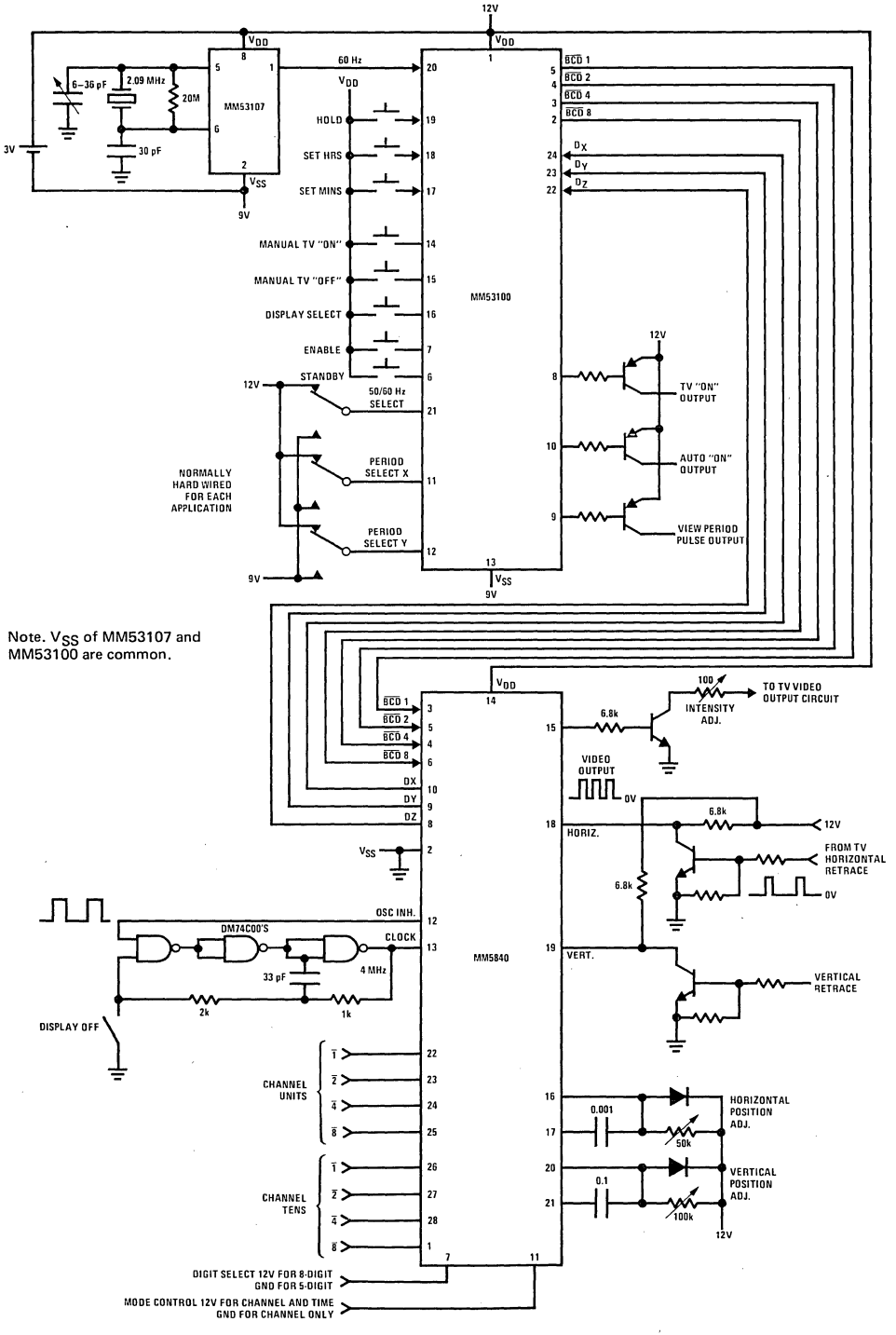



**FIGURE 5. Typical Application TV Channel and Time Display**

# Electronic Data Processing

National Semiconductor

Electronic Data Processing

# MM5034, MM5035 Octal 80-Bit Static Shift Register

## General Description

The MM5034 octal 80-bit shift register is a monolithic MOS integrated circuit utilizing N-channel low threshold enhancement mode and ion-implanted depletion mode devices.

The MM5034 is designed for use in computer display peripherals. All inputs and outputs are TTL compatible. The clocks and recirculate logic are internal to reduce system component count, and TRI-STATE® output buffers provide bus interface. Because of its N-channel characteristics, single 5V power supply operation is required.

Simple interface to the NSC CRT DP8350 controller and character generator to incorporate an entire CRT terminal is feasible with the MM5034.

The MM5034 is available in a 22-lead dual-in-line package.

The MM5035 is a 20-pin version of the MM5034 with the TRI-STATE output select feature omitted, for a simple data in/data out operation.

## Features

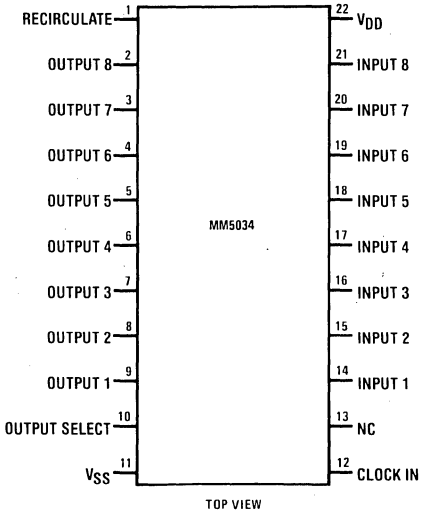
- Single 5V power supply
- Internal clocks
- High speed and static operation
- TRI-STATE output buffer
- Recirculate and output select independent
- TTL compatible

## Applications

- CRT displays
- Computer peripherals

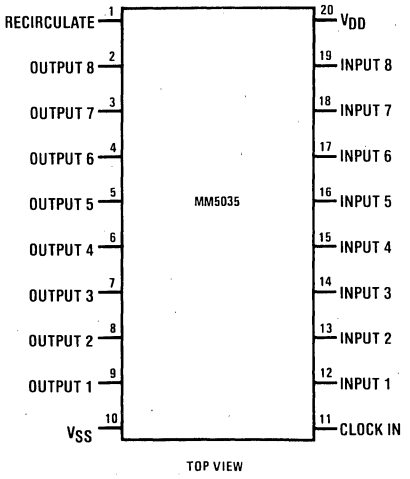
## Connection Diagrams

Dual-in-Line-Package




Order Number MM5034N
See Package 21

Dual-in-Line-Package




Order Number MM5035N
See Package 20A

5

## Absolute Maximum Ratings

| | |
|---|---|
| Supply Voltage | 7 $V_{DC}$ |
| Input Voltage | 7 $V_{DC}$ |
| Power Dissipation | 750 mW |
| Storage Temperature Range | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics $V_{DD}$ = 5V ±5%, $T_A$ = 0°C to +70°C

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Clock Input | | | | | |
| Logical "1" Input Voltage | | 2.2 | | | V |
| Logical "0" Input Voltage | | | | 0.8 | V |
| Data and Control Inputs | | | | | |
| Logical "1" Input Voltage | | 2.2 | | | V |
| Logical "0" Input Voltage | | | | 0.8 | V |
| Data, Clock and Control Inputs | | | | | |
| Logical "1" Input Current | $V_{IN}$ = 5V | | | 5.0 | μA |
| Input Capacitance | $V_{IN}$ = 2.5V | | 5.0 | 8.0 | pF |
| Outputs | | | | | |
| Logical "1" Output Voltage | $I_{OUT}$ = 100 μA | 2.4 | 2.8 | | V |
| Logical "0" Output Voltage | $I_{OUT}$ = 1.6 mA | | 0.25 | 0.4 | V |
| TRI-STATE Output Current | $V_{OUT}$ = 5V | | | −5.0 | μA |
| | $V_{OUT}$ = 0V | | | 5.0 | μA |
| Supply Current | | | 60 | 90 | mA |
| Timing | | | | | |
| Clock Frequency | | 0 | | 3.0 | MHz |
| Clock Pulse Width High | *(Figure 1)* | 125 | | 10,000 | ns |
| Clock Pulse Width Low | (Note 1) | 125 | | ∞ | ns |
| Output Rise and Fall Time ($t_r$, $t_f$) | *(Figure 1)* | | 40 | 50 | ns |
| Set-Up Time | *(Figure 1)* | 100 | | | ns |
| Hold Time | *(Figure 1)* | 0 | | | ns |
| Output Enable Time | *(Figure 1)* | | | 185 | ns |
| Output Disable Time | *(Figure 1)* | | | 185 | ns |
| Clock Rise and Fall Time | *(Figure 1)* | | | 5.0 | μs |
| Output Delay, ($t_{PD}$) | | | 80 | 185 | ns |

**Note 1:** The clock input must be at a low level for DC storage. Minimum pulse width assumes 10 ns $t_r$ and $t_f$.

## MM5034, MM5035 Recirculate and TRI-STATE Operation

Recirculate is used to maintain data in the shift register after it has been loaded. While the shift register is being loaded, Recirculate must be at a logical "0". When the loading is completed, Recirculate should be brought to a logical "1". This disables the data input and feeds the output of the last shift cell back to the input of the first shift cell for each of the 8 registers.

For the output to be in the TRI-STATE mode output-select should be at the logical '1' level.

## AC Test Circuits and Switching Time Waveforms

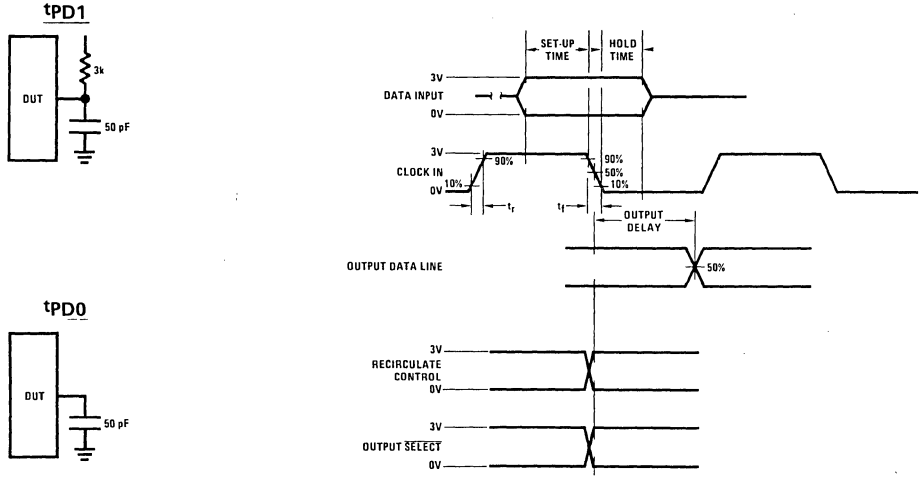



FIGURE 1

## Typical Application

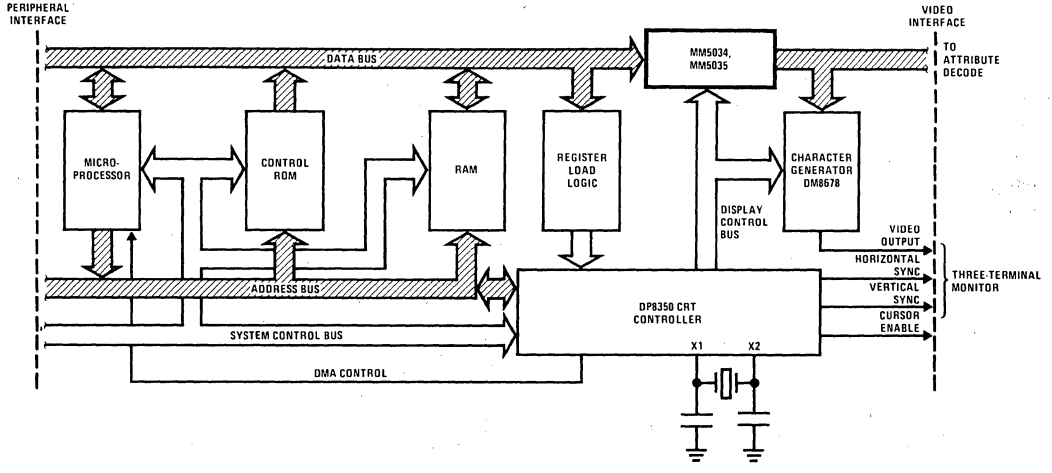



FIGURE 2. CRT System Diagram Using the MM5034, MM5035 as a Line Buffer with DMA

National Semiconductor

Electronic Data Processing

# MM5303 Universal Fully Asynchronous Receiver/Transmitter

## General Description

The MM5303 is a fully asynchronous receiver/transmitter, fabricated with National's metal-gate, depletion load, PMOS technology. All inputs and outputs are fully TTL compatible, requiring no external resistors or level shifting.

This device is a programmable interface between an asynchronous serial data channel and a parallel data channel. The transmitter section converts parallel data into a serial word which includes: start bit, data, parity bit (if selected), and stop bit(s). The receiver converts a serial word of the same format into a parallel one and automatically checks start bit, parity (if selected), and stop bit(s).

Both transmitter and receiver are doubly buffered; in addition, received data out and status words may be TRI-STATED, facilitating bus configurations.

Status conditions are: transmission complete, Tx buffer register empty, Rx data available, parity error, framing error, and over-run error.

The MM5303 is fully programmable. It can operate full or half duplex, transmitting and receiving simultaneously at different baud rates; word length may be 5, 6, 7 or 8 bits; parity generation/checking may be even, odd or inhibited; the number of stop bits may be either 1 or 2, with 1 1/2 bits when transmitting a 5 bit code.

## Features

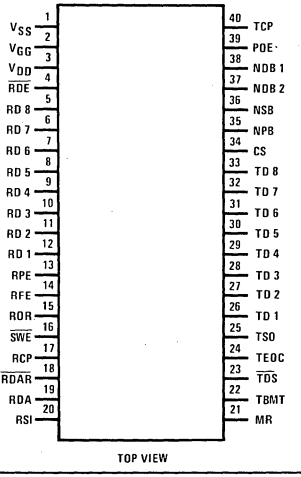
- Low power
- High speed
- Fully externally programmable:
  - Word length
  - Parity mode
  - Number of stop bits
- Fully double buffered eliminating need for precise synchronization
- Full or half duplex operation
- Direct TTL/DTL compatibility
- Automatic data received/transmitted status generation
- TRI-STATE outputs
- Automatic start bit generation/verification
- Internal pull-ups on all inputs

## Applications

- Peripherals
- Terminals
- Mini computers
- Facsimile transmission
- Modems
- Concentrators
- Asynchronous data multiplexers
- Card and tape readers
- Printers
- Data sets
- Controllers
- Keyboard encoders
- Remote data acquisition systems
- Asynchronous data cassettes

## connection diagram

Dual-In-Line Package




TOP VIEW

**Order Number MM5303N**
**See Package 24**

## Absolute Maximum Ratings (Note 1)

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ − 25V/$V_{SS}$ + 0.3V* |
| Operating Temperature Range | −25°C to +70°C |
| Storage Temperature Range | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

*Outputs should not have more than $V_{SS}$ − 15V

## DC Electrical Characteristics

$T_A$ within operating temperature range, $V_{SS}$ = 5V ±5%, $V_{DD}$ = 0V, $V_{GG}$ = −12V ±5% unless otherwise noted.

| | PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|---|
| $V_{IH}$ | High Input Voltage Levels | (Note 3) | $V_{SS}$−1.5 | | $V_{SS}$+0.3 | V |
| $V_{IL}$ | Low Input Voltage Levels | | $V_{DD}$ | | 0.8 | V |
| $V_{OH}$ | High Output Voltage Levels | $I_{OH}$ = −100μA | 2.4 | | | V |
| $V_{OL}$ | Low Output Voltage Levels | $I_{OL}$ = 1.6 mA | | | 0.4 | V |
| $I_{IH}$ | High Level Input Current Levels | $V_{IN}$ = $V_{SS}$ | | | 10 | μA |
| $I_{IL}$ | Low Level Input Current Levels | $V_{IN}$ = 0.4V, $V_{SS}$ = 5.25V | | | 1.6 | mA |
| $I_{OL}$ | Output Leakage Current Level | $\overline{SWE}$ = $\overline{RDE}$ = $V_{IH}$, $0 \leq V_{OUT} \leq 5V$ | | | −1 | μA |
| $I_{OS}$ | Output Short Circuit Current Level | $V_{OUT}$ = 0V, (Note 4) | | | 25 | mA |
| $C_{IN}$ | Input Capacitance | (Note 2) | | | | |
| | All Inputs | $V_{IN}$ = $V_{SS}$, f = 1 MHz | | 5 | 10 | pF |
| $C_{OUT}$ | Output Capacitance | | | | | |
| | All Outputs | $\overline{SWE}$ = $\overline{RDE}$ = $V_{IH}$, f = 1 MHz | | 10 | 20 | pF |
| $I_{SS}$ | Power Supply Current | All Inputs at $V_{SS}$ | | 13 | 25 | mA |
| $I_{GG}$ | Power Supply Current | All Inputs at $V_{SS}$ | | 6 | 15 | mA |

## AC Electrical Characteristics at 25°C

| | PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|---|
| | Clock Frequency | RCP, TCP | dc | | 500 | kHz |
| $t_{PW}$ | Pulse Width | | | | | |
| | Clock | RCP, TCP | 1 | | | μs |
| | Master Reset | MR | 5 | | | μs |
| | Control Strobe | CS | 1 | | | μs |
| | Tx Data Strobe | $\overline{TDS}$ | 300 | | | ns |
| | Rx Data Available Reset | $\overline{RDAR}$, (Note 5) | 200 | | | ns |
| $t_C$ | Coincidence Time | $\overline{TDS}$ | 300 | | | ns |
| | | CS | 1 | | | μs |
| $t_{SET}$ | Input Set-Up Time | TD1–TD8 | 0 | | | ns |
| | | NPB, NSB, NDB, POE | 0 | | | ns |
| $t_{HOLD}$ | Input Hold Time | TD1–TD8 | 300 | | | ns |
| | | NPB, NSB, NDB, POE | 0 | | | ns |
| $t_{pd0}$ | Output Propagation Delay to Low State | $\overline{RDE}$, $\overline{SWE}$ Enable to Outputs Low | | | 500 | ns |
| $t_{pd1}$ | Output Propagation Delay to High State | $\overline{RDE}$, $\overline{SWE}$ Enable to Outputs High | | | 500 | ns |

**Note 1:** "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. Except for "Operating Temperature Range" they are not meant to imply that the devices should be operated at these limits. The table of "Electrical Characteristics" provides conditions for actual device operation.

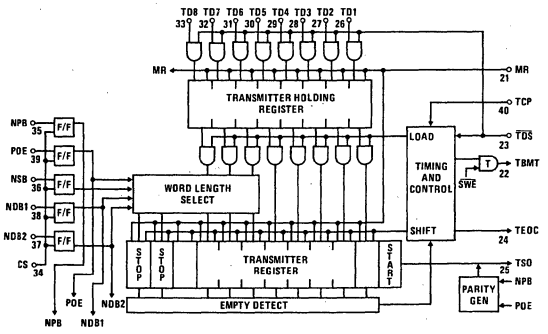
**Note 2:** Capacitance is guaranteed by periodic testing.

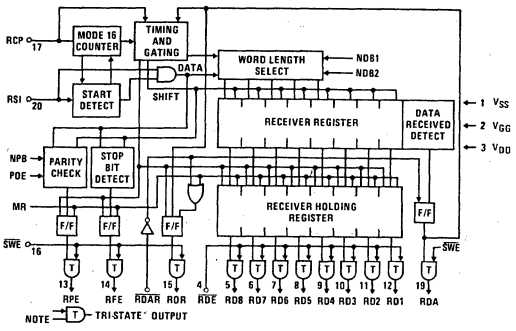
**Note 3:** Positive true logic notation is used:
Logic "1" = most positive voltage level
Logic "0" = most negative voltage level

**Note 4:** Only one output should be shorted at a time.

**Note 5:** Refer to Receiver Timing diagram for detail.

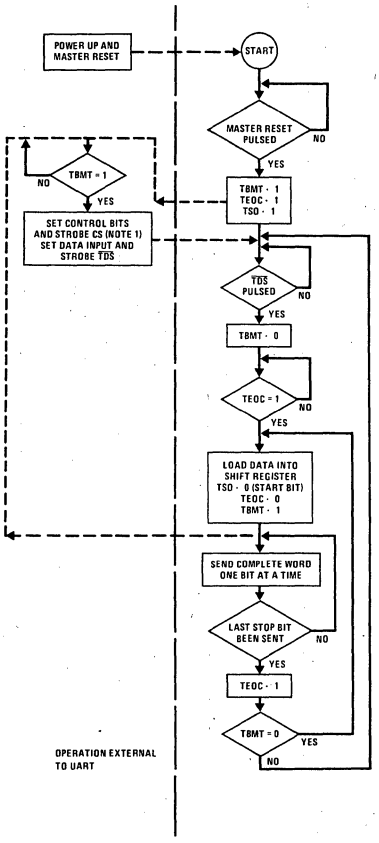
# Functional Block Diagrams



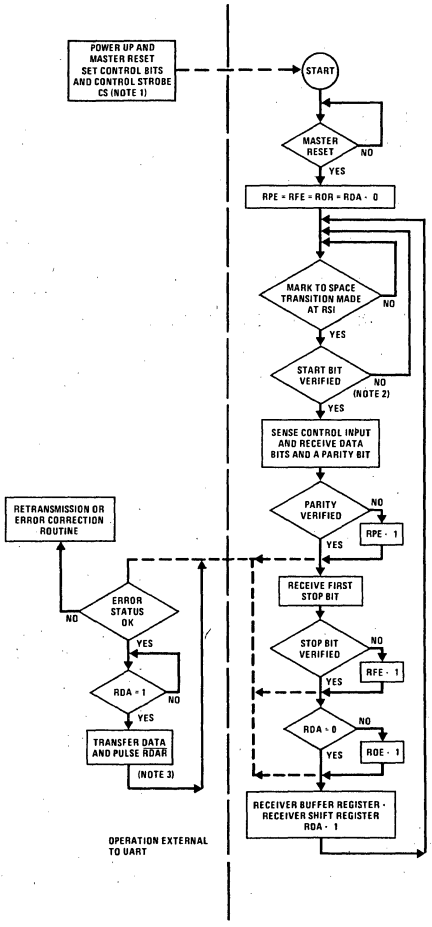




# Flow Charts

## Transmitting Sequence




## Receiving Sequence




**Note 1:** Control Strobe should be made only at the beginning of transmission and remain inactive during transmission. It may be tied high if no change is necessary.

**Note 2:** The line must stay low for 8 RCP pulses to be verified.

**Note 3:** $\overline{RDAR} \leftarrow 0$ will cause RDA ← 0, refer to receiver timing for detail.

## Pin Functions

| PIN NO. | SYMBOL | NAME | FUNCTION |
|---|---|---|---|
| 1 | $V_{SS}$ | Power Supply | +5V supply |
| 2 | $V_{GG}$ | Power Supply | −12V supply |
| 3 | $V_{DD}$ | Ground | Ground |
| 4 | $\overline{RDE}$ | Received Data Enable | A low-level input enables the outputs (RD8–RD1) of the receiver buffer register. |
| 5–12 | RD8–RD1 | Receiver Data Outputs | These are the 8 TRI-STATE data outputs enabled by $\overline{RDE}$. Unused data output lines, as selected by NDB1 and NDB2, have a low-level output, and received characters are right justified, i.e., the LSB always appears on the RD1 output. |
| 13 | RPE | Receiver Parity Error Output | This TRI-STATE output (enabled by $\overline{SWE}$) is at a high-level if the received character parity bit does not agree with the selected parity. |
| 14 | RFE | Receiver Framing Error Output | This TRI-STATE output (enabled by $\overline{SWE}$) is at a high-level if the received character has no valid stop bit. |
| 15 | ROR | Receiver Over Run Output | This TRI-STATE output (enabled by $\overline{SWE}$) is at a high-level if the previously received character is not read (RDA output not reset) before the present character is transferred into the receiver buffer register. |
| 16 | $\overline{SWE}$ | Status Word Enable Input | A low-level input enables the outputs (RPE, RFE, ROR, RDA, and TBMT) of the status word buffer register. |
| 17 | RCP | Receiver Clock | This input is a clock whose frequency is 16 times (16X) the desired receiver baud rate. |
| 18 | $\overline{RDAR}$ | Receiver Data Available Reset Input | A low-level input resets the RDA output to a low-level. |
| 19 | RDA | Receiver Data Available Output | This TRI-STATE output (enabled by $\overline{SWE}$) is at a high-level when an entire character has been received and transferred into the receiver buffer register. |
| 20 | RSI | Receiver Serial Input | This input accepts the serial bit input stream. A high-level (mark) to low-level (space) transition is required to initiate data reception. |
| 21 | MR | Master Reset | This input should be pulsed to a high-level after power turn-on. This sets TSO, TEOC, and TBMT to a high-level and resets RDA, RPE, RFE and ROR to a low-level. |
| 22 | TBMT | Transmitter Buffer Empty Output | This TRI-STATE output (enabled by $\overline{SWE}$) is at a high-level when the transmitter buffer register is empty and may be loaded with new data. |
| 23 | $\overline{TDS}$ | Transmitter Data Strobe Input | A low-level input strobe enters the data bits into the transmitter buffer register. |
| 24 | TEOC | Transmitter End of Character Output | This output appears as a high-level each time a full character is transmitted. It remains at this level until the start of transmission of the next character or for one full TCP period in the case of continuous transmission. |
| 25 | TSO | Transmitter Serial Output | This output serially provides the entire transmitted character. TSO remains at a high-level when no data is being transmitted. |
| 26–33 | TD1–TD8 | Transmitter Data Inputs | There are 8 data input lines (strobed by $\overline{TDS}$) available. Unused data input lines,as selected by NDB1 and NDB2, may be in either logic state. The LSB should always be placed on TD1. |
| 34 | CS | Control Strobe Input | A high-level input enters the control bits (NDB1, NDB2, NSB, POE and NPB) into the control bits holding register. This line may be strobed or hard wired to a high-level. |
| 35 | NPB | No Parity Bit | A high-level input eliminates the parity bit from being transmitted; the stop bit(s) immediately follow the last data bit. In addition, the receiver requires the stop bit(s) to follow immediately after the last data bit. Also, the RPE output is forced to a low-level. See pin 39, POE . |
| 36 | NSB | Number of Stop Bits | This input selects the number of stop bits, 1, 1 1/2, or 2 to be transmitted. A low-level input selects 1 stop bit; a high-level input selects 2 stop bits, except when 5-bit data is selected, then 1 1/2 stop bits will occur. |

## Pin Functions (cont'd)

| PIN NO. | SYMBOL | NAME | FUNCTION |
|---|---|---|---|
| 37–38 | NDB2, NDB1 | Number of Data Bits/ Character | These 2 inputs are internally decoded to select either 5, 6, 7 or 8 data bits/ character as per the following truth table: |

| NDB2 | NDB1 | data bits/character |
|---|---|---|
| L | L | 5 |
| L | H | 6 |
| H | L | 7 |
| H | H | 8 |

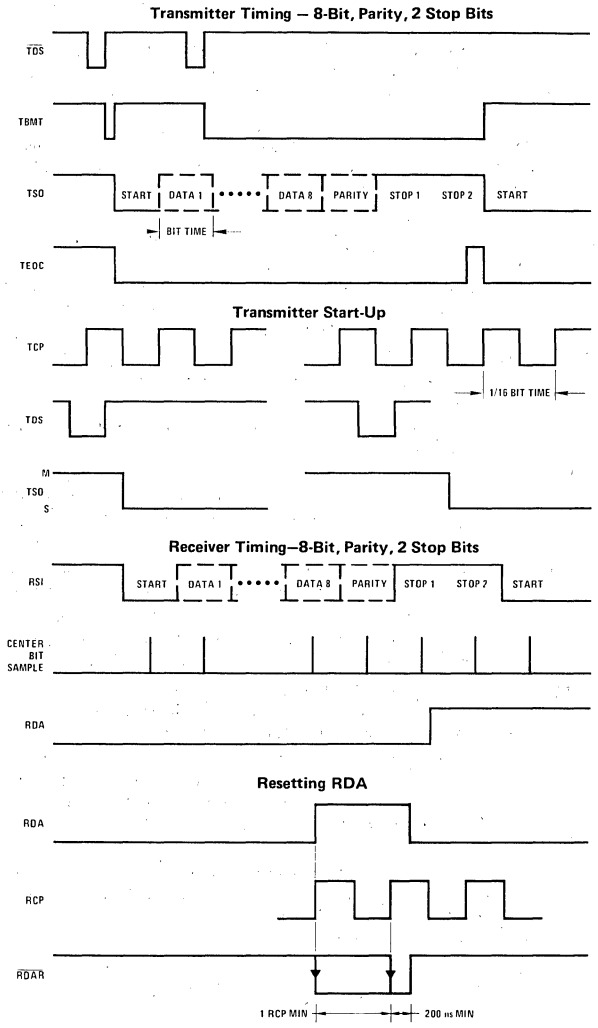
| PIN NO. | SYMBOL | NAME | FUNCTION |
|---|---|---|---|
| 39 | POE | Odd/Even Parity Select | The logic level on this input, in conjunction with the NPB input, determines the parity mode for both the receiver and transmitter, as per the following truth table: |

| NPB | POE | MODE |
|---|---|---|
| L | L | odd parity |
| L | H | even parity |
| H | X | no parity |

X = don't care

| PIN NO. | SYMBOL | NAME | FUNCTION |
|---|---|---|---|
| 40 | TCP | Transmitter Clock | This input is a clock whose frequency is 16 times (16X) the desired transmitter baud rate. |

## Timing Diagrams

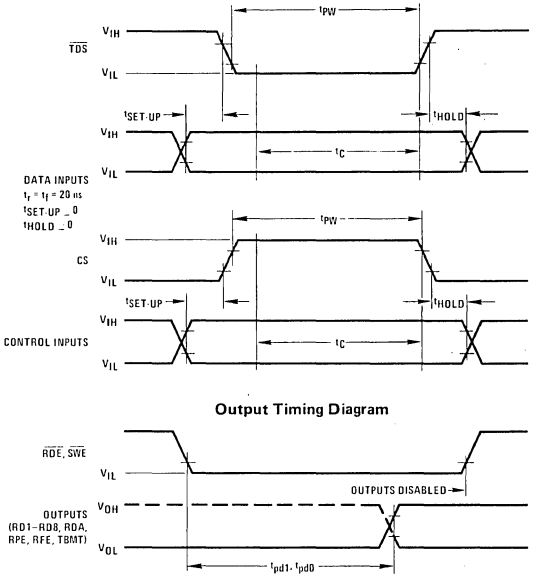



Upon data transmission initiation, or when not transmitting at 100% line utilization, the start bit will be placed on the TSO line at the high to low transition of the TCP clock following the trailing edge of $\overline{TDS}$.

$\overline{RDAR}$ may go low any time after the RDA comes up but must stay low for at least 200 ns after the first clock pulse period. $\overline{RDAR}$ may be hard wired low, in which case RDA will go high and remain high for the duration of the positive clock pulse.

## Timing Diagrams (cont'd)

**Data/Control Timing Diagram**

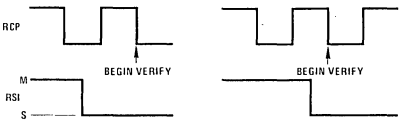



**Output Timing Diagram**




Note: Waveform drawings not to scale for clarity.

**Start Bit Detect/Verify**




If the RSI line remains spacing for 1/2 a bit time, a genuine start bit is verified. Should the line return to a marking condition prior to 1/2 a bit time, the start bit verification process begins again.

National Semiconductor

Electronic Data Processing

# MM5307 Baud Rate Generator/Programmable Divider

## General Description

The National Semiconductor MM5307 baud rate generator/programmable divider is a MOS/LSI P-channel enhancement mode device. A master clock for the device is generated either externally or by an on-chip crystal oscillator (Note 4). An internal ROM controls a divider circuit which produces the output frequency. Logic levels on the four control pins select between sixteen output frequencies. The frequencies are chosen from the following possible divisors: 2N, for $3 \leq N \leq 2048$; 2N + 1 and 2N + 0.5 for $4 \leq N \leq 2048$. Also one of the sixteen frequencies may be gated from the external frequency input. The MM5307AA is supplied with the divisors shown in the Control Table.

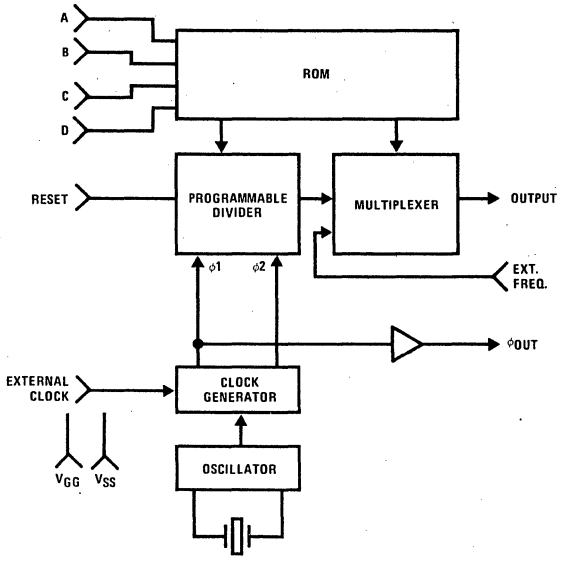
## Features

- On-chip crystal oscillator
- Choice of 16 output frequencies from 1 crystal
- External frequency input pin
- Internal ROM allows generation of other frequencies on order
- Bipolar compatibility
- 0.01% accuracy (typ) exclusive of crystal
- 1 MHz master clock frequency

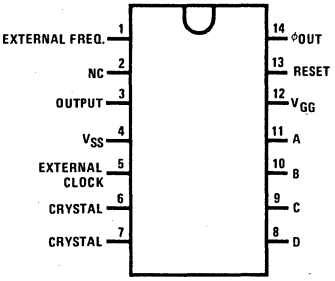
## Applications

- UAR/T clocks
- System clocks
- Electrically programmable counters

## Schematic and Connection Diagrams




Dual-In-Line Package




TOP VIEW

**Order Number MM5307AA/N, MM5307AB/N, MM5307FAG/N**
**See Package 18**

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin With Respect to $V_{SS}$ | +0.3V to $V_{SS}$ − 20V |
| Power Dissipation | 700 mW |
| Storage Temperature Range | −65°C to +150°C |
| Operating Temperature | 0°C to +70°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## DC Electrical Characteristics

$T_A$ within operating range, $V_{SS}$ = 5V ±5%, $V_{GG}$ = −12V ±5%, unless otherwise specified.

| | PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|---|
| | All Inputs (Except Crystal Pins) | | | | | |
| $V_{IH}$ | Logical High Level | | $V_{SS}$−1.5 | | $V_{SS}$+0.3 | V |
| $V_{IL}$ | Logical Low Level | | $V_{SS}$−18 | | $V_{SS}$−4.2 | V |
| | Leakage | $V_{IN}$ = −10V, $T_A$ = 25°C, All Other Pins GND | | | 0.5 | μA |
| | Capacitance | $V_{IN}$ = 0V, f = 1 MHz, All Other Pins GND, (Note 1) | | | 7.0 | pF |
| | External Clock Duty Cycle | | 40% | | 60% | |
| | Capacitance Measured Across Crystal Pins | f = 1 MHz, (Note 3) | | | 5.0 | pF |
| | Output Levels | | | | | |
| $V_{OH}$ | Logical High Level | $I_{SOURCE}$ = −0.5 mA | $V_{SS}$−2.6 | $V_{SS}$ | | V |
| $V_{OL}$ | Logical Low Level | $I_{SINK}$ = 1.6 mA | | | $V_{SS}$−4.6 | V |
| $I_{GG}$ | Power Supply Current | f = 1 MHz | | | 35 | mA |

## AC Electrical Characteristics

$T_A$ within operating range $V_{SS}$ = 5V ±5%, $V_{GG}$ = −12V ±5%, unless otherwise specified.

| | PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|---|
| | Master Frequency | | 0.8 | | 1.0 | MHz |
| $t_A$ | Access Time | $C_L$ = 50 pF, (Note 2) | | | 16 | μs |
| $t_{RD}$ | Reset Delay Time | f = Master Clock Frequency | | | 500 + 4/f | ns |
| $R_{PW}$ | Reset Pulse Width | | 500 + 4/f | | | ns |
| $t_{OD}$ | Output Delay From Reset | | | | 500 + 4/f | ns |
| | Output Duty Cycle = 0.5T ± 1/f | T = Output Period f = Master Frequency | 0.5T−1/f | | 0.5T+1/f | |

**Note 1:** Capacitance is guaranteed by periodic measurement.

**Note 2:** Access time is defined as the time from a change in control inputs (A, B, C, D) to a stable output frequency. Access time is a function of frequency. The following formula may be used to calculate maximum access time for any master frequency: $T_A = 2.8\mu s + 1/f \times 13$, f is in MHz.

**Note 3:** The MM5307 is designed to operate with a 921.6 kHz parallel resonant crystal. When ordering the crystal a value of load capacitance ($C_L$) must be specified. This is the capacitance "seen" by the crystal when it is operating in the circuit. The value of $C_L$ should match the capacitance measured at the crystal frequency across the crystal input pins on the MM5307. Any mismatch will be reflected as a very small error in the operating frequency. To achieve maximum accuracy, it may be necessary to add a small trimmer capacitor across the terminals.

**Note 4:** If the crystal oscillator is used Pin 5 (external clock) is connected to $V_{SS}$. If an external clock is used Pin 7 is connected to $V_{SS}$.

## Control Table

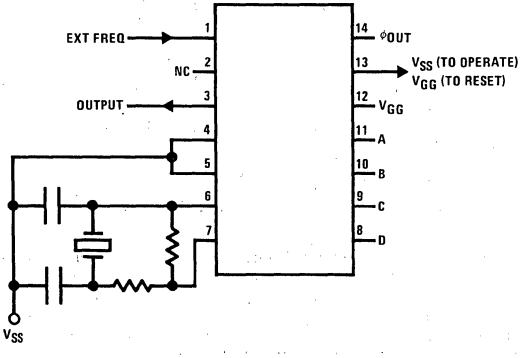
Input Freq: 921.6 kHz Master Clock

| CONTROL PINS | | | | NOMINAL BAUD RATES (OUTPUT FREQUENCY/16) | | | DIVISOR FOR AA |
|---|---|---|---|---|---|---|---|
| A | B | C | D | AA | AB | FAG | |
| 0 | 0 | 0 | 1 | 50 | 50 | 50 | 1152 |
| 0 | 0 | 1 | 0 | 75 | 200 | 75 | 768 |
| 0 | 0 | 1 | 1 | 110 | 110 | 110 | 524 |
| 0 | 1 | 0 | 0 | 134.5 | 134.5 | 134.5 | 428.5 |
| 0 | 1 | 0 | 1 | 150 | 150 | 150 | 384 |
| 0 | 1 | 1 | 0 | 300 | 300 | 300 | 192 |
| 0 | 1 | 1 | 1 | 600 | 600 | 600 | 96 |
| 1 | 0 | 0 | 0 | 900 | 900 | 1050 | 64 |
| 1 | 0 | 0 | 1 | 1200 | 1200 | 1200 | 48 |
| 1 | 0 | 1 | 0 | 1800 | 1800 | 45.5 | 32 |
| 1 | 0 | 1 | 1 | 2400 | 2400 | 2400 | 24 |
| 1 | 1 | 0 | 0 | 3600 | 3600 | 56.9 | 16 |
| 1 | 1 | 0 | 1 | 4800 | 4800 | 4800 | 12 |
| 1 | 1 | 1 | 0 | 7200 | 75 | 66.7 | 8 |
| 1 | 1 | 1 | 1 | 9600 | 9600 | 9600 | 6 |
| 0 | 0 | 0 | 0 | EXTERNAL FREQ | | | |

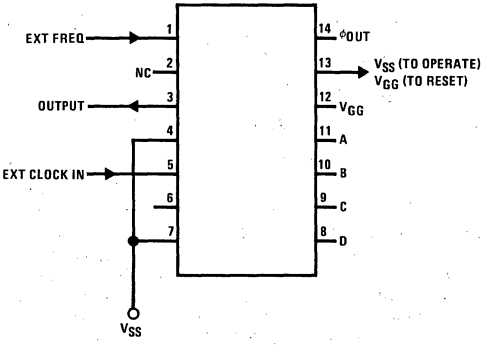
Positive Logic: $1 = V_H$
$0 \equiv V_L$

## Typical Applications

Internal Oscillator



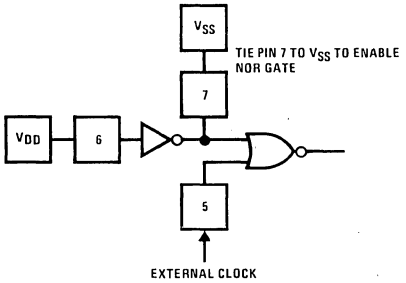

External Clock



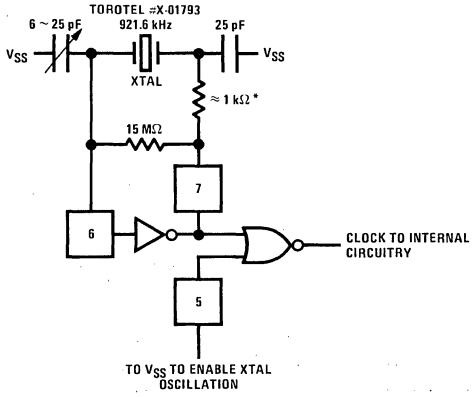
## Application Hints

The external clock is brought in on pin 5 and pin 7 is tied to $V_{SS}$ to enable the external clock input. Pin 6 can be left open; however, this may cause some current flow that can be eliminated by connecting pin 6 to $V_{DD}$.

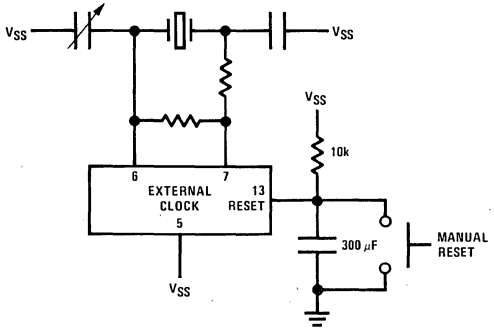
1) To use the MM5307 with an external clock, hook it up as follows:




2) To use a crystal directly:




*Component values should be selected based on crystal used.

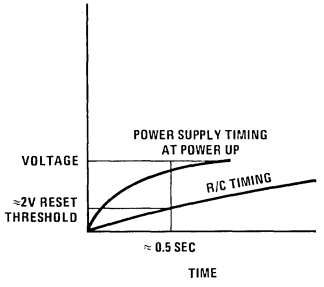



3) Reset (pin 13) must be at $V_{SS}$ to operate. It may be necessary to take this to GND or $V_{GG}$ to reset the ROM select circuit. An option is to tie $\phi$ out (pin 14) to external Freq In (pin 1), if not otherwise used.

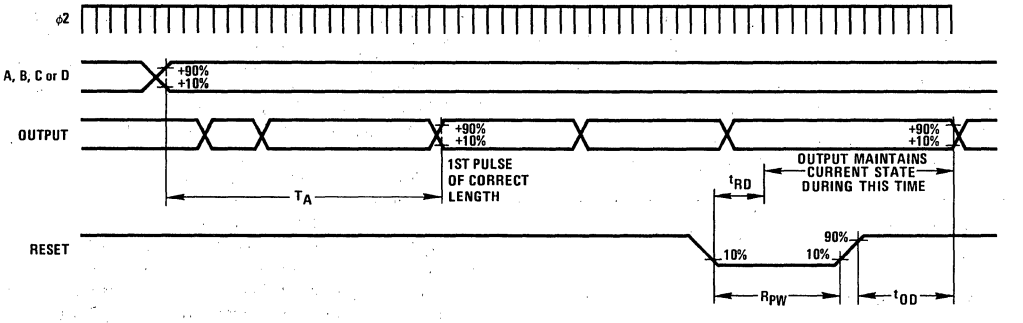
4) An interesting application might use two MM5307's in series to generate additional frequencies, i.e., with one programmed from the 921.6 kHz to 800 Hz out, a second could divide that by 16 to give a 50 Hz crystal controlled signal.

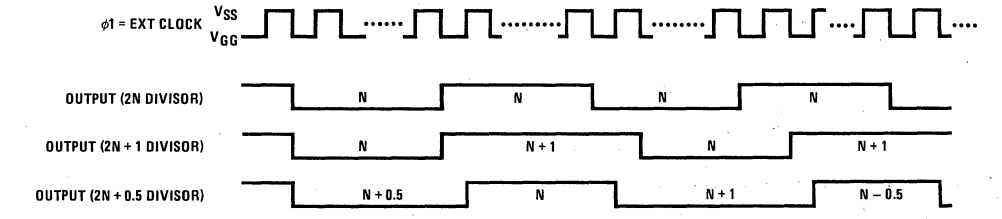
5) MM5307AA divisors are on the data sheet. AB divisors are the same as the AA except: 1) Code 0010 is divided by 288 → 32 kHz out, 200 baud; 2) Code 1110 is divided by 768 → 1.2 kHz, 75 baud.

The MM5307 does not always generate an output when the power is up, even though the oscillator seems to be operating properly. In order to eliminate this problem, it is necessary to reset the chip at power "ON". This can be done manually, with a reset signal by a host system, or automatically by using R/C timing elements. The reset is done internally, when program inputs change. When using an R/C combination for auto resetting, the time constant must be several times larger than that of the power supply. For example, most lab power supplies take at least 0.5 sec for the voltage to reach 90% of full level. A 10 kΩ resistor and 300 μF capacitor combination should be adequate for most applications.



## Timing Diagrams



National Semiconductor

Electronic Data Processing

# MM5330 4½-Digit Panel Meter Logic Block

## General Description

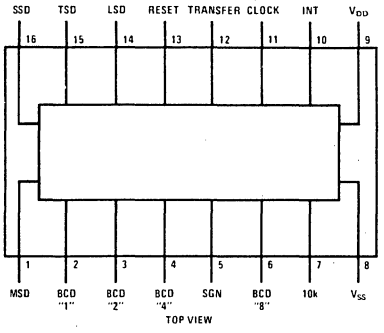
The MM5330 is a monolithic integrated circuit which provides the logic circuitry to implement a 4-1/2 digit panel meter. The MM5330 utilizes P-channel low threshold enhancement mode devices and ion-implanted depletion mode devices. All inputs and outputs are TTL compatible with BCD output for direct interface with various display drivers.

## Features

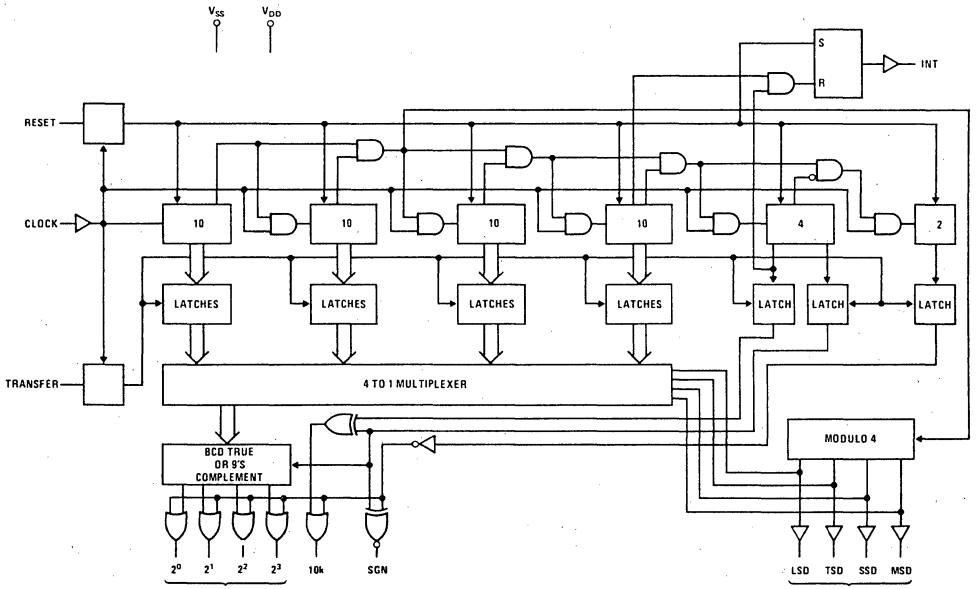
- dc to 400 kHz operation
- TTL compatible inputs and outputs
- BCD output code
- Overrange blanking
- Valid sign bit during overrange
- Standard supply voltages; +5, −15V

## Connection and Block Diagrams

Dual-In-Line Package




**Order Number MM5330N**
**See Package 19**



## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ + 0.3V to $V_{SS}$ − 25V |
| Operating Temperature | 0°C to +75°C |
| Storage Temperature | −40°C to +125°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, $V_{SS}$ = 4.75V to 5.25V, $V_{DD}$ = −16.5V to −13.5V unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage ($V_{SS}$) | | 4.75 | 5 | 5.25 | V |
| Power Supply Voltage ($V_{DD}$) | | −16.5 | −15 | −13.5 | V |
| Power Supply Current ($I_{SS}$) | No Load | | | 30 | mA |
| Input Frequency | | dc | | 400 | kHz |
| Reset or Transfer Pulse Width | | 200 | | | ns |
| Input Voltage Levels | $V_{SS}$ = 5V, $V_{DD}$ = −15V | | | | |
| Logic "1" | Inputs Driven by TTL or Square Waves | 3 | | 5 | V |
| Logic "0" | Inputs Driven by TTL or Square Waves | −15 | | 0.8 | V |
| Clock Input Voltage Levels | | | | | |
| Logic "1" | Driven by Sinewave | $V_{SS}$−0.5 | | $V_{SS}$+0.3 | V |
| Logic "0" | Driven by Sinewave | $V_{SS}$−25 | | $V_{SS}$−4.5 | V |
| Output Current Levels | $V_{SS}$ = 5V, $V_{DD}$ = −15V | | | | |
| Digit Output State | | | | | |
| Logic "1" | $V_O$ Forced To 4.75V | 100 | | | μA |
| Logic "0" | $V_O$ Forced To 4.5V | −5 | | −20 | mA |
| All Other Outputs | | | | | |
| Logic "1" | $V_O$ Forced To 3V | 100 | | | μA |
| Logic "0" | $V_O$ Forced To 0.4V | −2 | | | mA |
| Delay From Digit Output to BCD Output | | 0.1 | | 5 | μs |

### FUNCTIONAL DESCRIPTION

**Counters:** The MM5330 has four ÷10 counters, one ÷4 counter, and one ÷2 for a count of 80,000 clock pulses. A ripple carry is provided and all counter flip-flops are synchronous with the negative transition of the input clock. The last flip-flop in the divider chain (÷2 in the block diagram) triggers with the "0" to "1" transition of the previous flip-flop. The count sequence is shown in the first column of the count diagram.

**Reset:** All counter stages are reset to "0" and the INT flip-flop (driving the INT output) is set to "1" on the first negative clock transition after a "0" is applied to the Reset input. The internal reset is removed on the first negative clock transition after the internal reset has occured and a "1" has been applied to the Reset input. This timing provides an on-chip reset at least one clock cycle wide and a one cycle delay to remove reset before counting begins.

**Transfer:** Data in the counters is transferred to the latches when the Transfer input is at "0." If the Transfer input is held low the state of the counters is continuously displayed (see count diagram). Data will cease to transfer to the latches on the first positive clock transition after the first negative clock transition after a "1" is applied to the Transfer input. This provides a transfer pulse at least one half clock cycle wide and a half clock cycle delay to remove the transfer signal before the counters change state.

**INT:** The integrate output is used to set the charge time on a dual slope integrator. INT is "1" from reset to the 18,000th clock pulse, then "0" until the next reset. The dual slope integrator is the voltage monitoring part of the external circuitry needed for a DPM. It charges a capacitor at a rate proportional to the measured voltage while INT is "1," then discharges at a rate proportional to a fixed reference as shown in the dual slope diagram. When the output of the integrator reaches 0V a pulse is generated and fed into the Transfer input of the chip. As the dual slope diagram indicates, the number in the latches is proportional to the measured voltage.

**Multiplexing:** The modulo 4 multiplex counter is triggered by the carry from the second decade counter, making the multiplex rate one hundredth the counting rate (4 kHz for a 400 kHz clock). The LSD, TSD, SSD and MSD (least significant, third significant, second significant and most significant digits) outputs indicate by a low level which decade latch is displayed at the BCD outputs.

## FUNCTIONAL DESCRIPTION (Continued)

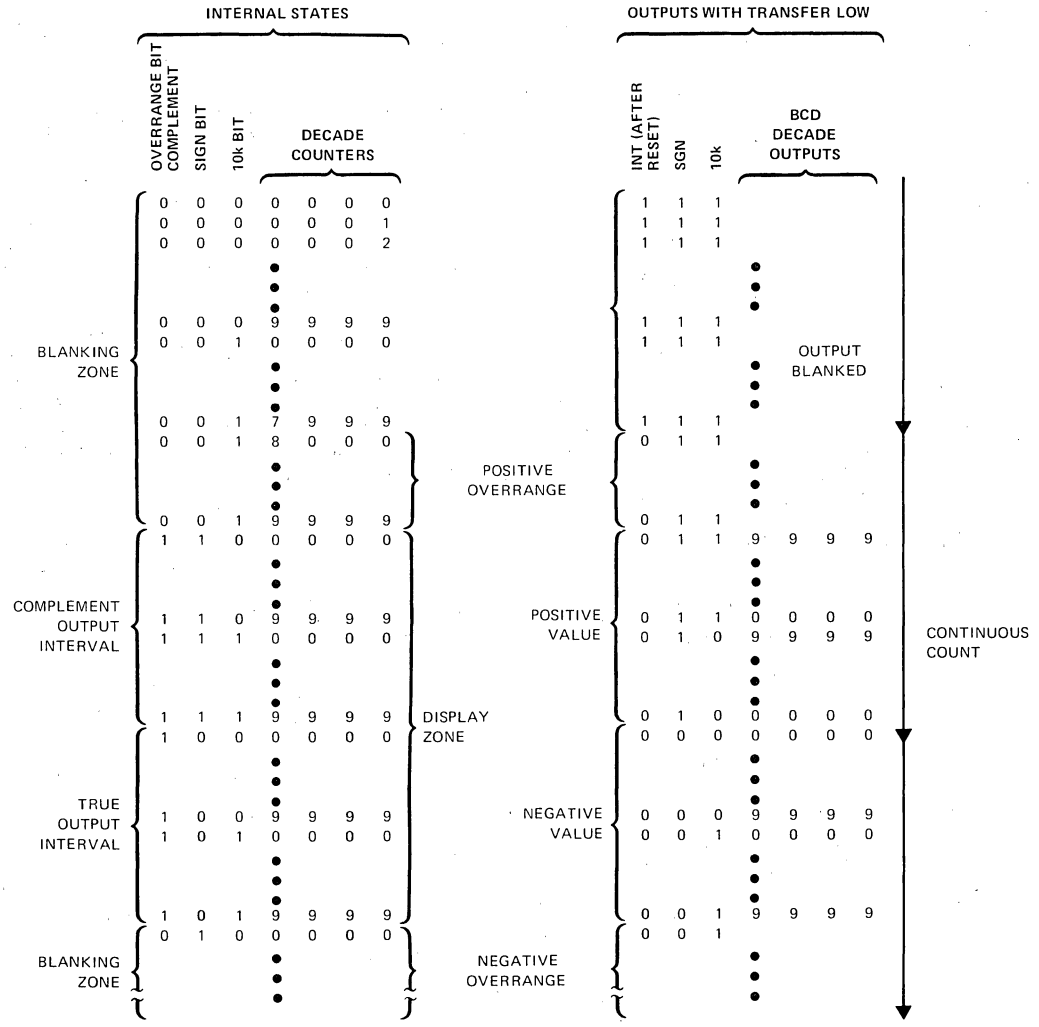
**Overrange Blanking and Sign:** The data in the latch for the ÷2 counter is used to detect an out-of-range voltage. If this latch is "0" the BCD and 10k outputs are forced to all "1's" and the SGN output is inverted. When the data in the overrange latch and the sign bit latch are "1" the sign bit generates the 9's complement of the decade latches and the complement of the 10k latch at the respective outputs. When the overrange bit is "1" and the sign bit is "0" true BCD of the decade latches and the uncomplemented 10k latch appear at the outputs.

## APPLICATIONS INFORMATION

The MM5330 is the display and control for a modified dual slope system. It contains the counters and latches, together with a multiplexing system to provide 4 digits of display with one decoder driver. It also provides a sign digit, either plus or minus, and a ten-thousand counts digit for full display of ±19999. By eliminating the right-most digits it may also be used as a 2-1/2 or 3-1/2 digit DVM chip.

The basic modified dual slope system for which the MM5330 is designed, is shown in *Figure 1.* The integrator is now used in a non-inverting mode and is biased to integrate negatively for all voltages below $V_{MAX}$. Thus if the maximum positive voltage at $V_{IN}$ is 1.9999V, then $V_{MAX}$ would be set at 2.200V. In this way, all voltages measured are below $V_{MAX}$. This eliminates the need for reference switching and provides automatic polarity with no additional components. Also, it can be shown that the amplifier input bias currents which cause errors in conventional dual slope systems are eliminated by merely zeroing the display. Thus low bias current op amps are not necessarily required unless a high input impedance is desired at $V_{IN}$.

## Count Diagram



## APPLICATIONS INFORMATION (Continued)

Secondly, the use of a conventional op amp for a comparator allows zeroing of all voltage offsets in both the op amp and comparator. This is achieved by zeroing the voltage on the capacitor through the use of the comparator as part of a negative feedback loop. During the zeroing period, the non-inverting input of the integrator is at $V_{REF}$. As this voltage is within the active common-mode range of the integrator the loop will respond by placing the integrator and comparator in the active region. The voltage on the capacitor is no longer equal to zero, but rather to a voltage which is the sum of both the op amp and comparator offset voltages. Because of the intrinsic nature of an integrator, this constant voltage remains throughout the integrating cycle and serves to eliminate even large offset voltages.

The waveforms at the output of the integrator are as shown. The voltage at A is the comparator threshold just discussed. Simultaneously, with the opening of switch A, $V_{IN}$ is connected to the input of the integrator via switch B. The output then slews to $V_{IN}$. Integration then begins for the reference period, after which time the reference voltage is again applied to the input. The output again slews the difference between $V_{REF}$ and $V_{IN}$ and integrates for the unknown period until the comparator threshold is crossed. At this point, the accumulated counts are transferred from the counters to the latches and zeroing begins until the next conversion interval.

It may be obvious, however, that while we have eliminated several of the basic dual slope circuits disadvantages, we have created another—the number of counts are no longer proportional to $V_{IN}$ but rather to $(V_{MAX}-V_{IN})$. In fact, when we short $V_{IN}$ to ground we are actually measuring our own 2.2000 $V_{MAX}$.

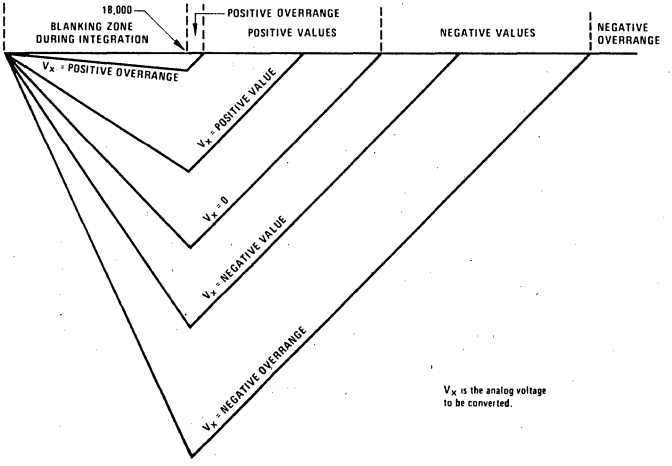
What is done in the MM5330 is to code convert the number of counts as shown in the count diagram. This chart shows a code conversion starting at the time of a reset. The first 18,000 counts are the reference period after which time the integrator changes slope. If a comparator crossing is detected within the next 2000 counts, a plus overrange condition will occur at the display. This condition results in a lit "+" sign, a lit "1" and four blanked rightmost digits. A transfer at 20,000 however, will create a reading of +1.9999, at 20,001 a reading of 19.998 and so on, until at 39,999 a reading of +0000 would be displayed. A transfer occuring at 40,000 would cause a −0000 display and so on until 60,000 counts were entered at which time a −1 with four blanked digits would be displayed indicating a minus overrange condition.

A typical circuit for a low cost 4 1/2 digit DPM is shown in *Figure 2.* The display interface used is a TTL, 7-segment decoder driver and four P-type transistors. The ±1 digit is driven directly by CMOS. The clock-synchronous reset and transfer functions prevent any cyclic digit variations and present a blink-free, flicker-free display. CMOS analog switches are used as reference, zero, and input switches and used also in the comparator slew rate circuit.

A problem with all dual slope systems occurs when short integrating times and high clock frequencies are used. Because of the very slow rise time of the ramp into the comparator, the output of the comparator will normally ramp at approximately 1/10 of its actual slew rate. Thus, a significant number of extra counts are displayed due to the slow rate of rise of the comparator. A technique to improve this consists of capacitor $C_S$ and analog switch four. An unstable positive loop is created by this capacitor when the comparator comes out of saturation. This causes the output to rise at its slew rate to the comparator threshold. As soon as this threshold is reached the analog switch opens and zeroing is initiated as previously discussed.

A simplified approach to performing the modified dual slope function combines the MM5330 and the LF11300 dual slope analog block as in *Figure 3.* The LF11300 provides the front analog circuitry required. This includes a FET input amplifier, analog switches, integrator and comparator. The LF11300 provides auto zero, > 1000 MΩ input impedance, and a ±10V analog range.

## Dual Slope Diagram



## Typical Applications

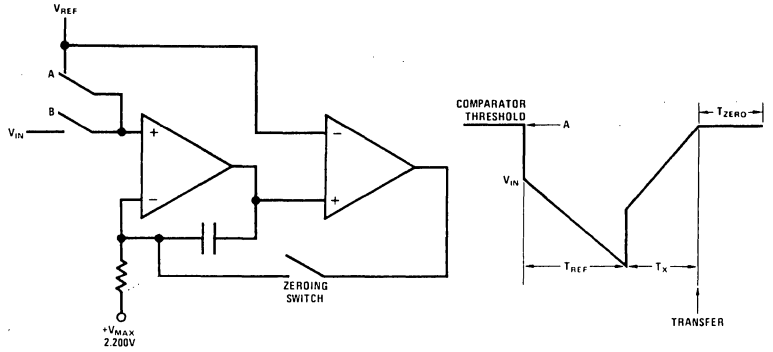



FIGURE 1. Modified Dual Slope

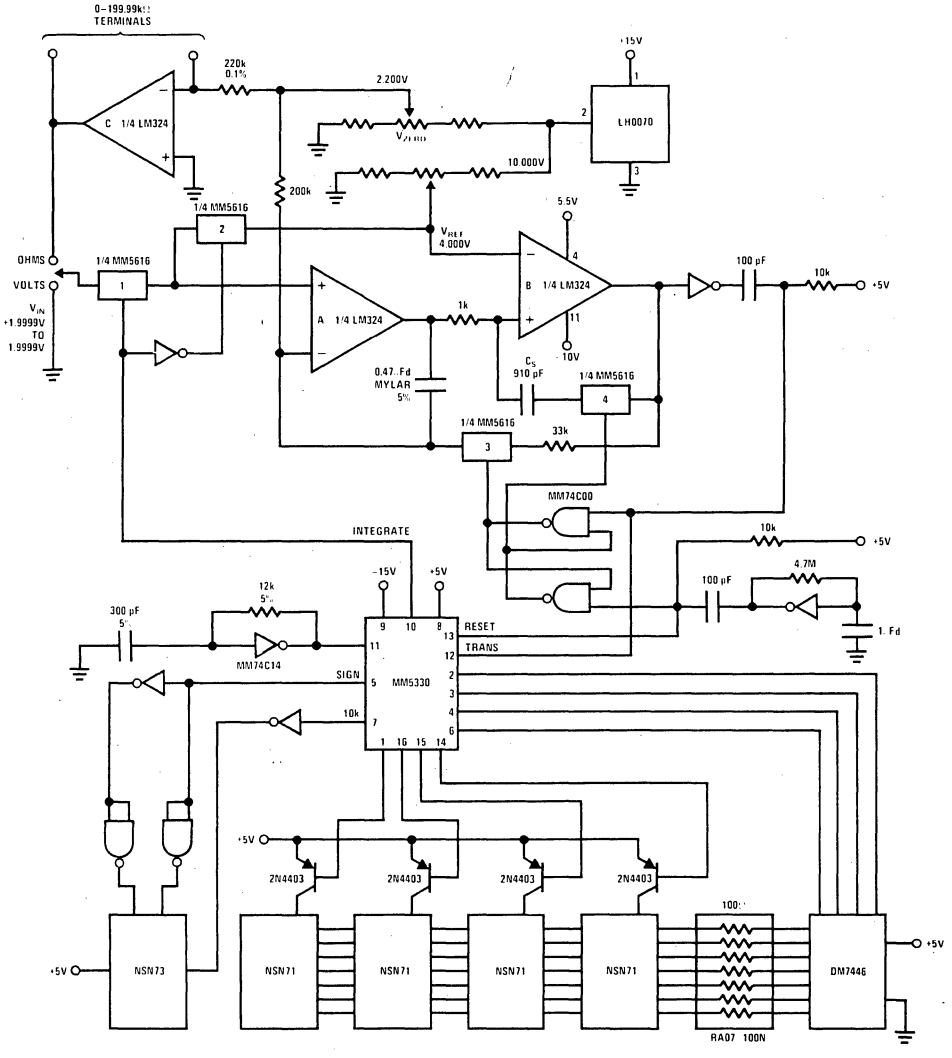
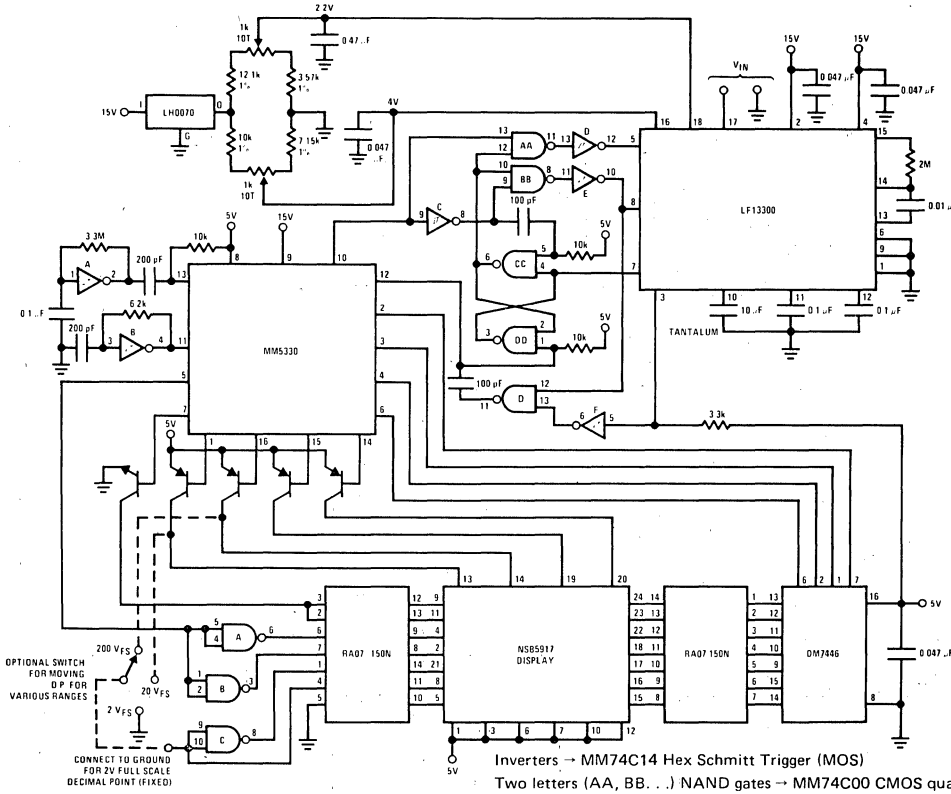
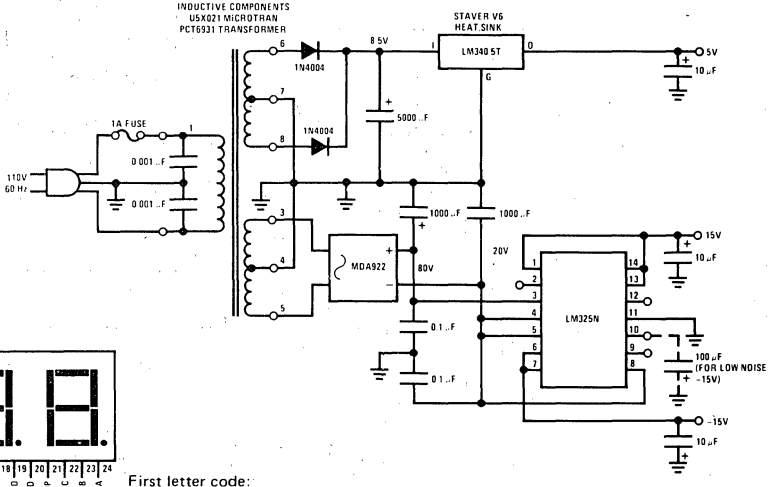



FIGURE 2. Typical Application Low Cost 4 1/2 Digit Volt-Ohm Meter

## Typical Applications (Cont'd)

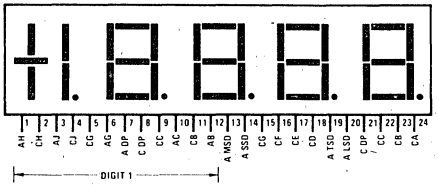



Inverters → MM74C14 Hex Schmitt Trigger (MOS)
Two letters (AA, BB. . .) NAND gates → MM74C00 CMOS quad NAND gates
One letter (A, B. . .) NAND gates → DM7400 TTL quad NAND gates
↓ → analog ground
→ digital ground
All resistors 1/4W, 5% unless otherwise noted.
All capacitor values in µF unless otherwise noted.





(Second letter code)



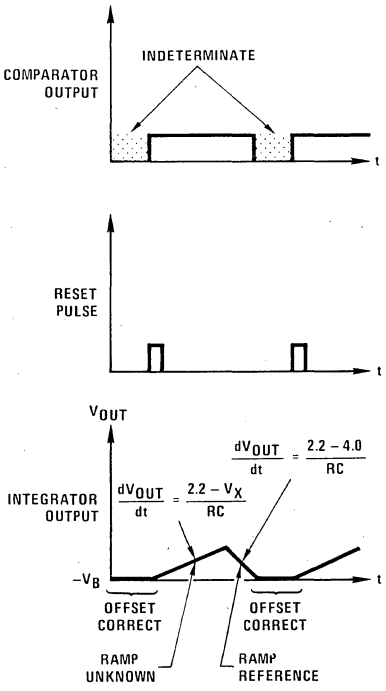

NSB5917 Display (Front View)

First letter code:
A → anode
C → cathode
MSD → digit 2
SSD → digit 3
TSD → digit 4
LSD → digit 5

**FIGURE 3. 4 1/2-Digit DVM**

## Timing Diagrams

4 1/2-Digit DPM




Note. Here the LF13300 always operates as an autozeroed, high input impedance inverting integrator; bipolar input voltages are handled by offsetting the analog ground by 2.2V.

National Semiconductor

Electronic Data Processing

# MM53200 Encoder/Decoder

The MM53200 Encoder/Decoder is an MOS/LSI Digital Code Transmitter — Receiver system.

## Features

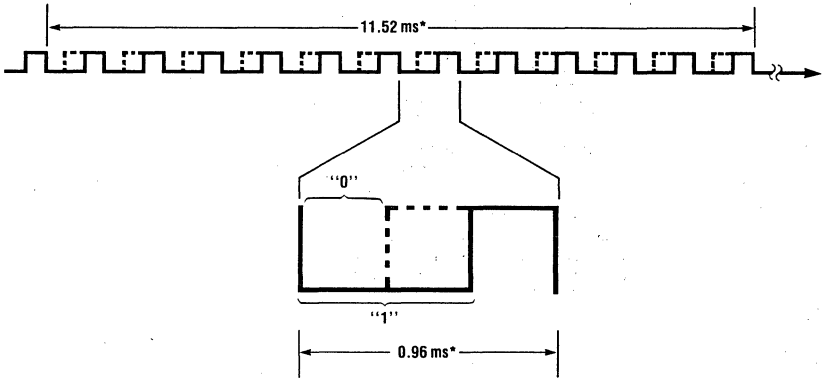
- A single chip contains both the Encoder and Decoder.
- Oscillator stability is non-critical, 5% components may be used.
- Cross interference of receivers in close proximity is virtually eliminated by circuitry which requires 4 valid words to be received, each within 64 ms of the other.

## Operation

In the transmit mode the twelve inputs are scanned sequentially producing the output pattern shown in Figure 1. This code is generated at the rate of 0.96 ms/bit, or 11.52 ms/word with 11.52 ms reset pulse between words.

In the receiver mode, the incoming signal is compared to the local code in a sequential manner; if there is an error, the system is reset and begins its comparison on the next pulse. If all twelve bits are received correctly, a "valid" signal will be generated. This signal clears a 64 ms counter and clocks a 3 stage counter. The 3 stage counter counts the "valid" pulses and when 4 pulses have been received, the transmit/receive output goes low. After the transmit/receive output is enabled, the next "valid" must be received within 128 ms, giving a one valid in 6 requirement to keep the transmit/receive output low.

Connection diagrams for the device in the Receive and Transmit modes are shown in Figures 2 and 3.




*@ 100 kHz Osc.

**Figure 1. Output Waveform**

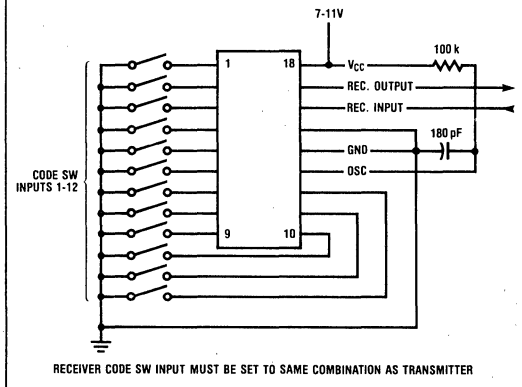



**Figure 2. Pin Connections for Receiver Mode**

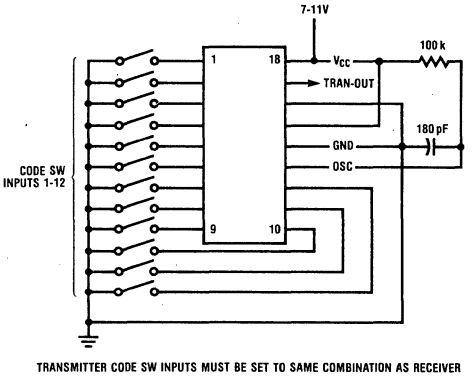



**Figure 3. Pin Connections for Transmitter Mode**

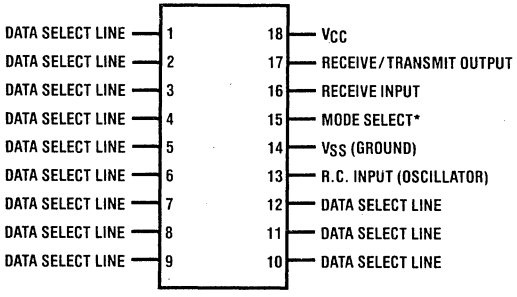
## Design Specifications

| | |
|---|---|
| Storage Temperature | −65 °C to +125 °C |
| Operating Temperature | −25 °C to +70 °C |
| Lead Temperature, Max. (Soldering, 10 seconds) | +300 °C |
| Power Supply | |
| $V_{DD}$ | $V_{SS}$ + 7V to $V_{SS}$ + 11V |
| $I_{DD}$ | 12 mA Max. |

## Electrical Characteristics

| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| Input Voltage Levels | | | | | |
| Schmitt Trigger Input | Level 1 | $V_{SS}+4$ | | | V |
| | Level 0 | | | $V_{SS}+2$ | V |
| All Other Inputs | Level 1 | $V_{DD}-0.5$ | | $V_{DD}$ | V |
| | Level 0 | $V_{SS}$ | | $V_{SS}+0.5$ | V |
| Input Resistor to $V_{DD}$ | | 200k | | 1.2M | Ω |
| Output Voltge (trans/rec) | | | | | |
| Logic High "1" | $I_{SOURCE}$ 5 μA | $V_{DD}-0.5$ | | $V_{DD}$ | |
| Logic Low "0" | $I_{SINK}$ 2 mA | $V_{SS}$ | | $V_{SS}+1.0$ | |
| Oscillator Frequency | ±15% exclusive of external components | | 100 | | kHz |

**18-Pin DIP — Top View**




*a. GROUND CONNECTION IS RECEIVER MODE
b. $V_{DD}$ CONNECTION IS TRANSMITTER MODE

**Order Number MM53200N**
**See Package 20**

### Pin Functions

**Pin #**

1-12 These Data Select lines are used to set the address of the encoder/decoder pair. They have on-chip pull-ups and input switches should pull them to ground.

13 The R.C. Input is the connection point for the single pin Oscillator. A resistor is hooked from this pin to $V_{CC}$ and a capacitor from this pin to GND. The frequency = 2/RC. The frequency may be decreased by increasing the resistor value.

14 $V_{SS}$ is the Ground Pin.

15 The Mode Select pin changes operation of the IC from Receiver to Transmitter. By grounding pin 15 the IC is put in the Receiver mode. By connection to $V_{CC}$ the IC is put in the Transmitter mode.

16 The Receiver input receives the digital PCM waveform from the Detector circuit.

17 The Output pin produces the PCM waveform when in the Transmit mode and is active low in the Receive mode.

18 $V_{CC}$ is the positive supply pin.

National Semiconductor

Electronic Data Processing

PRELIMINARY

# MM54240 Asynchronous Receiver/Transmitter Remote Controller

## General Description

The MM54240 is a monolithic MOS integrated circuit utilizing N-channel low-threshold, enhancement mode and ion-implanted depletion mode devices. The circuit is designed for processor-type remote control applications. The data transmission consists of a pulse-width modulated serial data stream of 18 bits. This stream consists of 7 address bits, 1 command bit, 8 data bits, 1 parity bit and 1 dummy bit in that order.

The MM54240 can be operated in either one of two modes; namely "master" and "slave". The master works directly from a processor bus structure. It is capable of polling and controlling 128 slave circuits. The slave circuits are interfaced to remote data sources and/or data destinations.

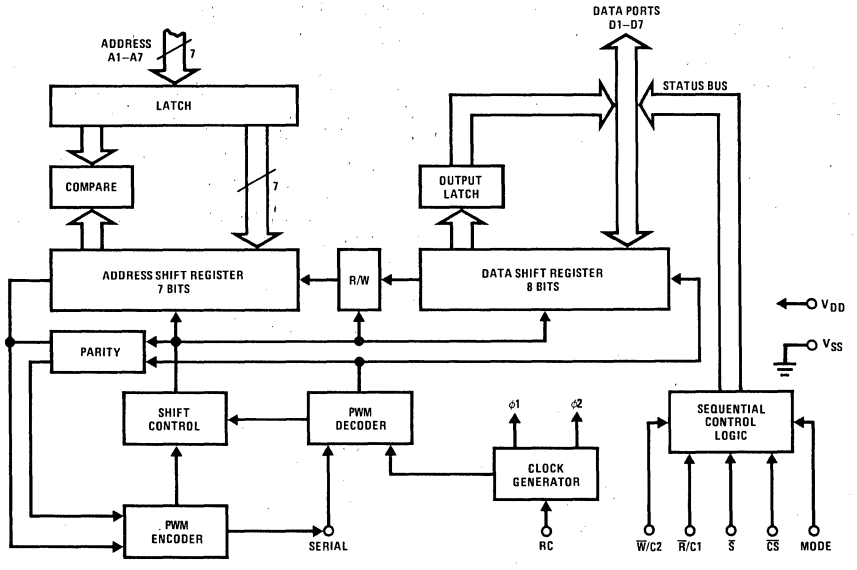
## Applications

The MM54240 finds application in transmitting data to and receiving data from remote A-to-D converters, remote microprocessor units, remote digital transducer or remote data peripheral devices.

## Features

- Supply voltage range — 4.75V to 11.5V single supply
- Low quiescent current — 5.0 mA maximum
- On-chip oscillator based on inexpensive R-C components
- Pulse-width modulation techniques minimize error and maximize frequency tolerance
- Mode input for either master or slave operations
- Chip select ($\overline{CS}$) input in the master mode
- Selectable output port options in the slave mode

## Functional Block Diagram




**Pin Configuration**

| | |
|---|---|
| 1 | $V_{SS}$ |
| 2 | D8 |
| 3 | D7 |
| 4 | D6 |
| 5 | D5 |
| 6 | D4 |
| 7 | D3 |
| 8 | D2 |
| 9 | D1 |
| 10 | Serial |
| 11 | Mode |
| 12 | $\overline{CS}$ |
| 13 | $V_{DD}$ |
| 14 | $\overline{R}$/C1 |
| 15 | $\overline{W}$/C2 |
| 16 | $\overline{S}$ |
| 17 | OSC |
| 18 | A7 |
| 19 | A6 |
| 20 | A5 |
| 21 | A4 |
| 22 | A3 |
| 23 | A2 |
| 24 | A1 |

## Absolute Maximum Ratings

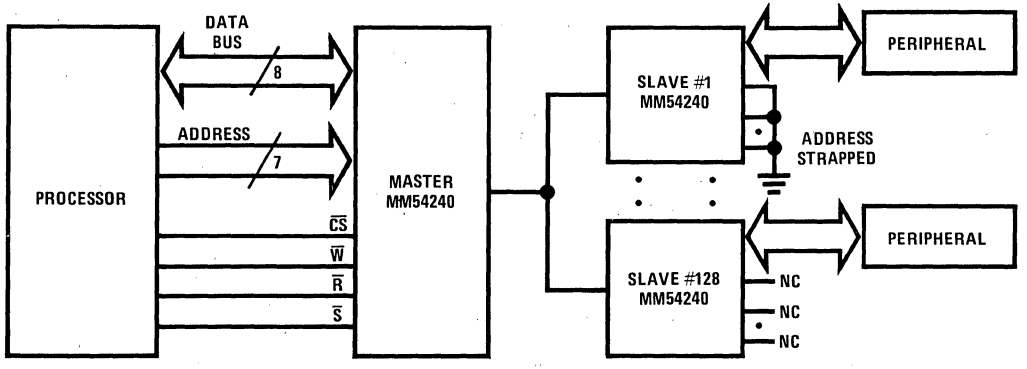
(exceeding these ratings could result in permanent damage to the device)

| | |
|---|---|
| Voltage on Any Pin with Respect to $V_{SS}$ | −0.5V to +12.0V |
| Operating Temperature | −40°C to +85°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics $T_A$ within operating range, $V_{SS} = 0V$

| Symbol | Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{DD}$ | Supply Voltage | | 4.75 | | 11.5 | V |
| $I_{DD}$ | Supply Current, Quiescent | $V_{DD} = 4.75V$ to 11.5V | | | 5.0 | mA |
| | Input Voltage | | | | | |
| $V_{IL}$ | Logic "0" | $V_{DD} = 4.75V$ to 11.5V | 0 | | 0.8 | V |
| $V_{IH}$ | Logic "1" | $V_{DD} = 4.75V$ to 5.25V | 2.4 | | $V_{DD}$ | V |
| $V_{IH}$ | Logic "1" | $V_{DD} = 5.25V$ to 11.5V | $V_{DD}-2.85$ | | $V_{DD}$ | V |
| | Output Current (D1-D8) | | | | | |
| $I_{OL}$ | $V_{OL} = 0.4V$ | $V_{DD} = 4.75V$ to 11.5V | 2.0 | | | mA |
| $I_{OH}$ | $V_{OH} = 2.4V$ | $V_{DD} = 4.75V$ to 5.25V | 200 | | | μA |
| $I_{OH}$ | $V_{OH} = 0.5\ V_{DD}$ | $V_{DD} = 5.25V$ to 11.5V | 200 | | | μA |
| $I_{OH}$ | $V_{OH} = 0.6\ V_{DD}$ (Weak $V_{OH}$) | $V_{DD} = 4.75V$ to 11.5V | 0.5 | | 30 | μA |
| | Output ($\overline{CS}$ Slave) | | | | | |
| $I_{OL}$ | $V_{OL} = 0.5V$ | $V_{DD} = 4.75V$ to 11.5V | 0.4 | | | mA |
| | Frequency RC Input | | | | | |
| F | For a Fixed $(RC)_1$ | $V_{DD} = 4.75V$ to 7.0V | 200 | 400 | 600 | kHz |
| F | For a Fixed $(RC)_2$ | $V_{DD} = 7.0V$ to 11.5V | 200 | 400 | 600 | kHz |
| | Output Current (Serial) | | | | | |
| $I_{OL}$ | $V_{OL} = 0.4V$ | $V_{DD} = 4.75V$ to 11.5V | 2.0 | | | mA |
| $I_{LEAK}$ | Open Drain Leakage | $V_{DD} = 4.75V$ to 11.5V | | | 10 | μA |
| | Input Pull-Up Resistors | | | | | |
| $I_{IL}$ | $V_{IN} = V_{SS}$ | $V_{DD} = 4.75V$ to 11.5V | 15 | | 100 | μA |

## Typical Application



## Circuit Description

The MM54240 consists of four major logic blocks: Sequential Control, Shift Register, PWM Encoder and PWM Decoder.

**Data Ports (D1-D8):** The data ports are for input and output of data and there are three output levels. In addition to the standard high and low states when the outputs are driving an external circuit, the outputs can also be in a weak pull-up state to $V_{DD}$. For the master circuit, the outputs are configured with standard high and low states coincident with properly enabled $\overline{CS}$ and $\overline{R}$. This permits direct interface or buffered interface with the standard bus structure of a processor system. The first three data ports (D1, D2, D3) also serve as status pins coincident with enabled $\overline{CS}$ and $\overline{S}$.* For the slave circuit, specialized input and output options are available by selecting the C1 and C2 inputs.

**Address Ports (A1-A7):** The address ports are for the input of address information into the MM54240. For the master circuit, the input must be valid during the $\overline{R}$ and $\overline{W}$ command strobes. For the slave circuit, a unique hard-wired code must be on the address ports. This code is the address of the slave circuit for addressing purposes.

**Mode:** This input is low for slave and high (or open) for master selections. An internal pull-up resistor is provided.

**Chip Select ($\overline{CS}$):** In the slave mode, this pin will become an output and will indicate logic "0" when the circuit is expecting to receive a transmission. In the master mode, the $\overline{CS}$ input has to be pulled low before the $\overline{R}$, $\overline{W}$ or $\overline{S}$ strobes can be acknowledged. An internal pull-up resistor is provided.

* The other data ports will output logic "0".

**Read/Control 1 ($\overline{R}$/C1):** In the master mode, while $\overline{CS}$ is active low, this input can be used to initiate either of the following three operations depending upon the present status of the circuit.

1. To initiate a read command
2. To enable output ports if transmission received is valid
3. To terminate read command if transmission received is incorrect

In the slave mode, this input, together with $\overline{W}$/C2, selects the specialized output port configuration.

**Write/Control 2 ($\overline{W}$/C2):** In the master mode, while $\overline{CS}$ is active low, this input can be used to initiate a write command. In the slave mode, this input, together with $\overline{R}$/C1, selects the specialized output port configuration.

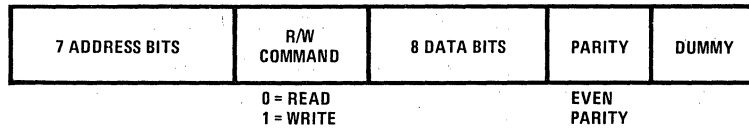
**Status ($\overline{S}$):** In the master mode while $\overline{CS}$ is active low, this input enables circuit status information to be output at the first three data ports. The other five data ports will be at logic "0". In the slave mode, this input sets all the output (D1-D8) latches to the logic "1" state.

**RC:** This input is for connection to a resistor-capacitor circuit for the on-chip oscillator. Frequency tolerance is specified for two voltage ranges. In a master-slave system, if no one circuit has a frequency more than a factor of 2 different from any other circuit, then, valid transmission is guaranteed. Nominal setting is 400 kHz.

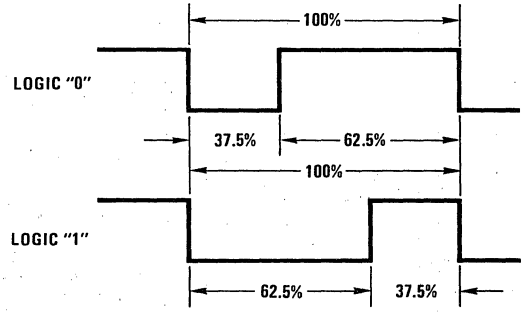
**Serial:** Input and output pin for serial transmission. Output has open drain configuration.

### Data Format

1. Serially transmitted data



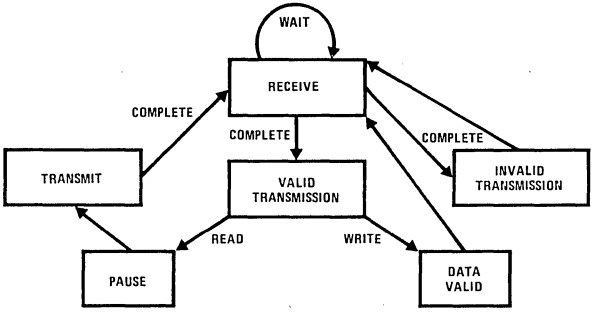

2. Pulse Width Modulation Coding




A bit is equivalent to 96 clocks of the R-C oscillator frequency i.e.; when R-C frequency = 400 kHz, 1 bit = 240 $\mu$s, 1 word = 4.32 ms.

## Circuit Description (Continued)

### Slave Circuit Logic Flow Diagram

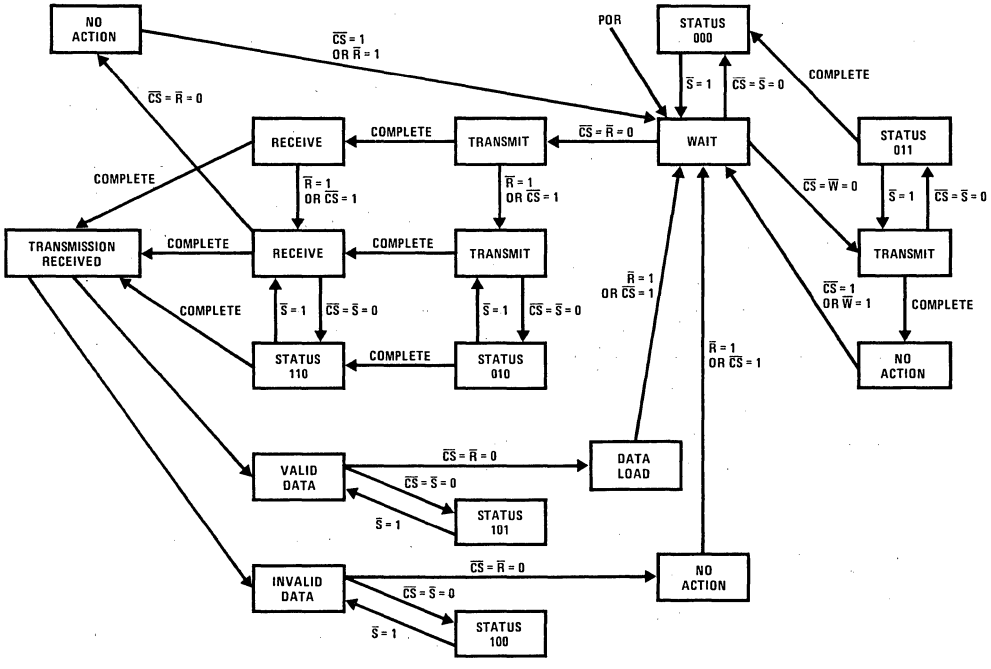



### SPECIALIZED OUTPUT OPTIONS FOR SLAVE CIRCUITS

| C1 | C2 | Description |
|---|---|---|
| 1 | 1 | All 8 pins are high impedance input ports |
| 0 | 1 | All 8 pins are standard low impedance output ports |
| 0 | 0 | D1-D4 are standard low impedance output ports<br>D5-D8 are high impedance input ports |
| 1 | 0 | * Logic "0" outputs are low impedance output ports<br>Logic "1" outputs are weak pull-ups to $V_{DD}$ |

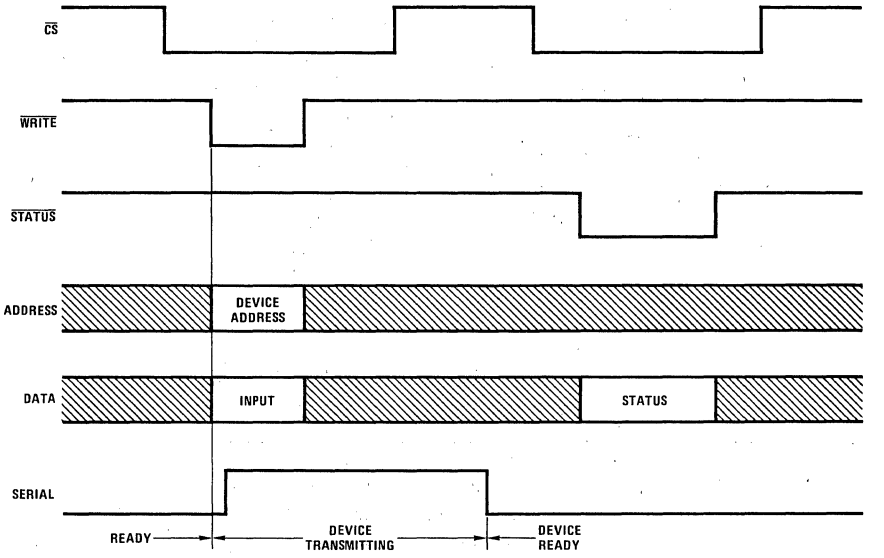
* In this option, the slave data ports can be connected in a wired-OR configuration with open-collector or open-drain outputs on the peripheral.

### Master Circuit Logic Flow Diagram

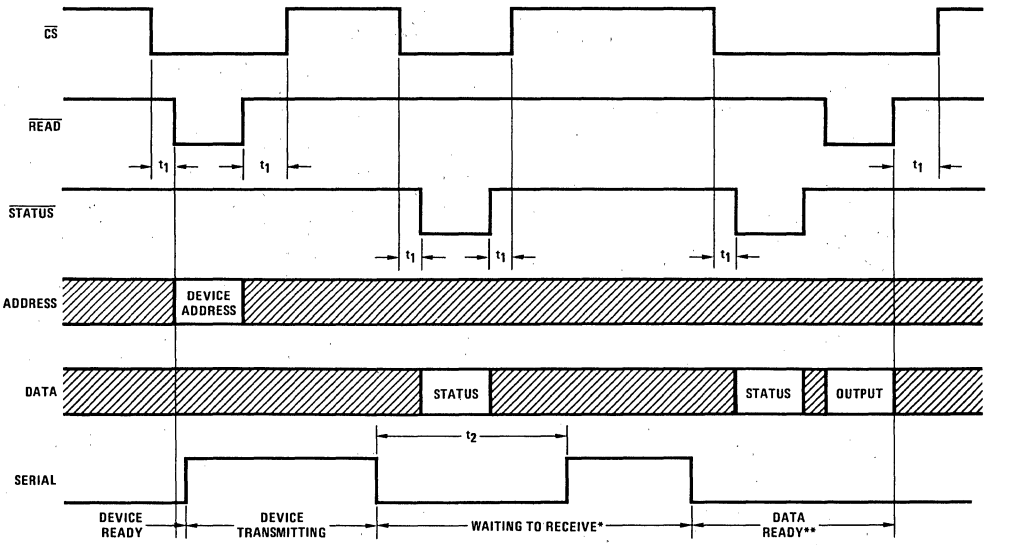


## Switching Time Waveforms

### Master Write Operation




### Master Read Operation




* During "waiting to receive" state, $\overline{CS}$ coupled with $\overline{R}$/C1 will force device into the device ready state

** If address or parity do not match, the data invalid state is entered. $\overline{CS}$ coupled with $\overline{R}$/C1 will force device into the device ready state

$t_1$—there is no overlap requirement for $\overline{\text{CHIP SELECT}}$

$t_2$—delay between master-slave transmission 0.2 mS to 1.4 ms

National Semiconductor

Electronic Data Processing

# MM57109 MOS/LSI Number-Oriented Processor

## General Description

The MM57109 is an MOS/LSI number-oriented processor (actually, a pre-programmed single-chip microcomputer member of National's COP family) intended for use in number processing applications. Scientific calculator functions, test and branch capability, internal number storage, and input/output instructions have been combined in this single chip device. Programming is done in calculator keyboard level language which simplifies software development. Generated code is more reliable because algorithms are preprogrammed in an on-chip ROM. Data or instructions can be synchronous or asynchronous; I/O digit count, I/O notation mode, and error control are user programmable; a sense input and flag outputs are available for single bit control.

The MM57109 can be used as a stand-alone processor with external ROM/PROM and program counter (PC). Alternatively it can be configured as a peripheral device on the bus of a microprocessor or minicomputer.

## Applications

- Instruments
- Microprocessor/minicomputer peripheral
- Test equipment
- Process controllers

## Features

- Scientific calculator instructions (RPN)
  - Up to 8-digit mantissa, 2-digit exponent
  - Four-register stack, one memory register
  - Trigonometric functions, logarithmic functions, $Y^X$, $e^X$, $\pi$, etc.
  - Error flag generation and recovery
- Flexible input/output
  - HOLD input allows asynchronous instructions or single stepping
  - Asynchronous digit input instruction (AIN) with data ready ($\overline{ADR}$) input
  - Multidigit I/O instructions (IN, OUT) with floating point or scientific notations
  - Programmable mantissa digit count for IN, OUT instructions
  - Sense input and flag outputs
- Branch control
  - Conditional and unconditional program branching
  - Increment/decrement branch on non-zero for program loops
- Interface simplicity
  - Single $\phi$ clock
  - Low power operation
  - Generates all I/O control signals
  - Separate digit input, output, and address buses

TABLE I. FEATURE COMPARISON OF LSI NUMBER PROCESSING CHIPS

| FUNCTION | CALCULATOR | MM57109 | MICRO-PROCESSOR |
|---|---|---|---|
| I/O | Keyboard display | Multidigit asynchronous digit single bit | Data bytes single bit |
| Data format | Floating point Scientific Notation | Floating point Scientific Notation | Binary |
| Data length | Fixed | Variable (1 to 8 digit mantissa) | Fixed |
| Program | Key sequence | External ROM/ PC, $\mu$P or FIFO | External ROM Internal PC |
| Speed (math or I/O operations) | 14–1500 ms | 0.5–1000 ms | 0.5–1000 ms |
| Minimum number of chips for CPU and RAM | 1–3 | 1 (external PC and program source) | 2–6 |

Note: This data sheet is complete. It contains all necessary programming information and electrical interconnect details. The user should read this document *thoroughly* before proceeding.

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Relative to $V_{SS}$ (All Other Pins Connected to $V_{SS}$) | $V_{SS}$ + 0.3V to $V_{SS}$ − 12V |
| Ambient Operating Temperature | 0°C to +70°C |
| Ambient Storage Temperature | −55°C to +125°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## DC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $7.9V \leq V_{SS} - V_{DD} \leq 9.5V$ unless otherwise stated

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Operating Voltage ($V_{SS} - V_{DD}$) | | 7.9 | 9.0 | 9.5 | V |
| Operating Supply Current ($I_{DD}$) | $V_{SS} - V_{DD}$ = 9.5V, $T_A$ = 25°C (Excluding Outputs) | | 12 | 18 | mA |
| Osc. Input Voltage Levels | Internal 6k Resistor to $V_{SS}$ | | | | |
| Input High Level ($V_{IH}$) | $V_{SS} - V_{DD}$ = 7.9V | $V_{SS}$−1.0 | | | V |
| Input Low Level ($V_{IL}$) | $V_{SS} - V_{DD}$ = 9.5V | | | $V_{DD}$+1.5 | V |
| HOLD, POR Input Voltage Levels | No Internal Resistors | | | | |
| Input High Level ($V_{IH}$) | | $V_{SS}$−3.0 | | | V |
| Input Low Level ($V_{IL}$) | | | | $V_{DD}$+1.5 | V |
| $I_1$–$I_6$ Input Voltage Levels | Internal Resistors to $V_{SS}$ (No Resistor for $I_6$), (Note 1) | | | | |
| Input High Level ($V_{IH}$) | | $V_{SS}$−1.0 | | | V |
| Input Low Level ($V_{IL}$) | | | | $V_{SS}$−4.0 | V |
| **INTERFACING WITH MOS OR CMOS** | | | | | |
| All Outputs | External Resistor to $V_{DD}$ = 10k–20k | | | | |
| Output High Voltage ($V_{OH}$) | | $V_{SS}$−1 | | $V_{SS}$ | V |
| Output Low Voltage ($V_{OL}$) | | $V_{DD}$ | | $V_{DD}$+1 | V |

## AC Electrical Characteristics $0°C \leq T_A \leq +70°C$, $7.9V \leq V_{SS} - V_{DD} \leq 9.5V$ unless otherwise stated

| PARAMETER | CONDITIONS (Note 2) | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Osc. Input Frequency | | 320 | 400 | 400 | kHz |
| Osc. Duty Cycle | | 46 | 56 | 66 | % |
| Osc. Input | | | | | |
| Rise Time ($t_r$) | $C_{LOAD}$ = 25 pF, $R_{LOAD}$ = 6 kΩ | | | 350 | ns |
| Fall Time ($t_f$) | RC = 0.15 μs | | | 50 | ns |
| Sync. Output Timing | $C_{LOAD}$ = 250 pF | | | | |
| $t_B$ (1 Microcycle) | | 10.0 | | 12.5 | μs |
| $t_{pdsL}$ | | 0.1 | | 1.65 | μs |
| $t_{pdsH}$ | | 0.1 | | 1.25 | μs |
| $t_{HS}$ | | 0.1 | | 0.8 | μs |
| R/$\overline{W}$, ISEL Output Timing | | | | | |
| $t_{pdf}$ | $C_{LOAD}$ = 100 pF | | | 4.4 | μs |
| $\overline{DAS}$ Output Timing | | | | | |
| $t_{pdDAS}$ | $C_{LOAD}$ = 50 pF | | | 4.4 | μs |
| $t_{rDAS}$ | $C_{LOAD} \leq$ 20 pF | 0.3 | | | μs |
| DA1–DA4, $\overline{BR}$ Output Timing | $C_{LOAD}$ = 100 pF (DA1–DA4) | | | | |
| $t_{pdg}$ | $C_{LOAD}$ = 250 pF ($\overline{BR}$) | 0.5 | | 4.0 | μs |
| DO1–DO4, F1, F2, RDY, ERROR Output Timing | | | | | |
| $t_{pdK}$ | | | | 6.0 | μs |

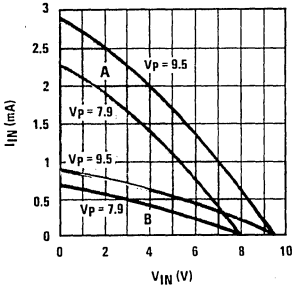
**Note 1:** An external resistor (5k–20k) can be tied at the $I_6$ input to $V_{SS}$ to overcome internal load device to $V_{DD}$.

**Note 2:** See *Figure 2* for timing diagrams of each of the following inputs/outputs.

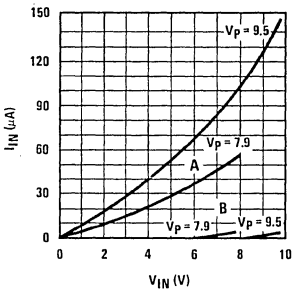
## Typical Performance Characteristics

DC Voltage-Current Curves for Chip Inputs and Outputs

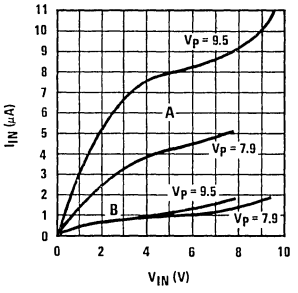
$I_5$–$I_1$



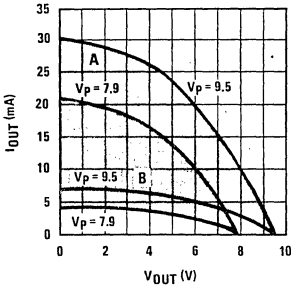

$I_6$



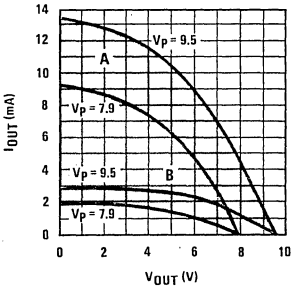

POR



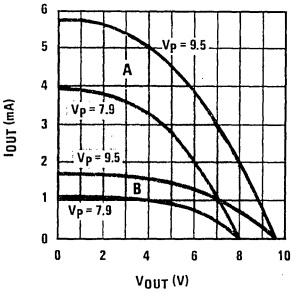

SYNC



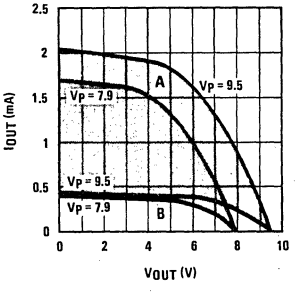

ISEL, R/$\overline{W}$



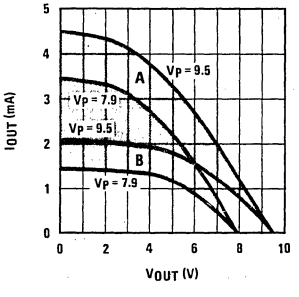

$\overline{DAS}$



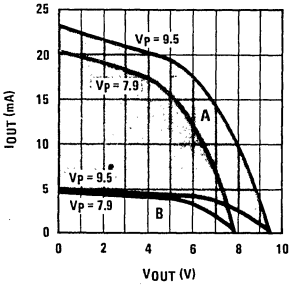

DA1–DA4




$\overline{BR}$




DO1–DO4, F1, F2, ERROR and RDY



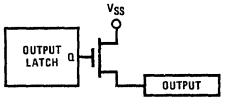

**Note 1:** Curves "A" represent a maximum current device at 0°C. Curves "B" represent a minimum current device at +70°C.

**Note 2:** Vp = $V_{SS}$–$V_{DD}$ (volts)
$V_{IN}$ = Input voltage at pin relative to $V_{DD}$
$I_{IN}$ = Current *into* pin
$V_{OUT}$ = Output voltage at pin relative to $V_{DD}$
$I_{OUT}$ = Current *out of* pin

**Note 3:** HOLD input current is essentially zero at all valid input voltages.

**Note 4:** OSC input loading consists of a resistor to $V_{SS}$ whose minimum possible value is 4k and whose maximum possible value is 8k.

**Note 5:** Output currents ($I_{OUT}$) are shown for logic "1" output conditions only. Any output which is at logic "0" is in a high-impedance state. Equivalent circuit for each output is:



## Pinout and Block Diagrams

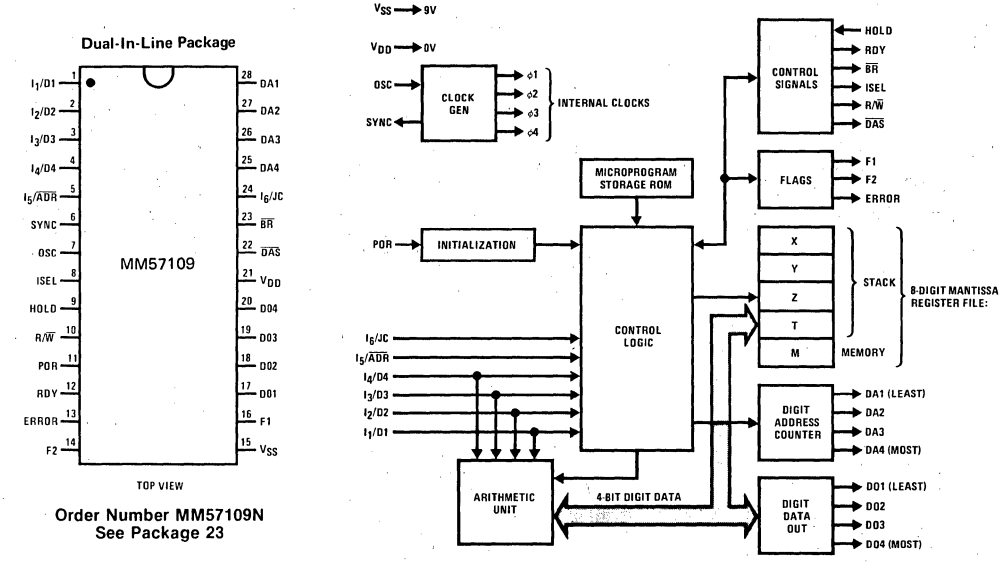
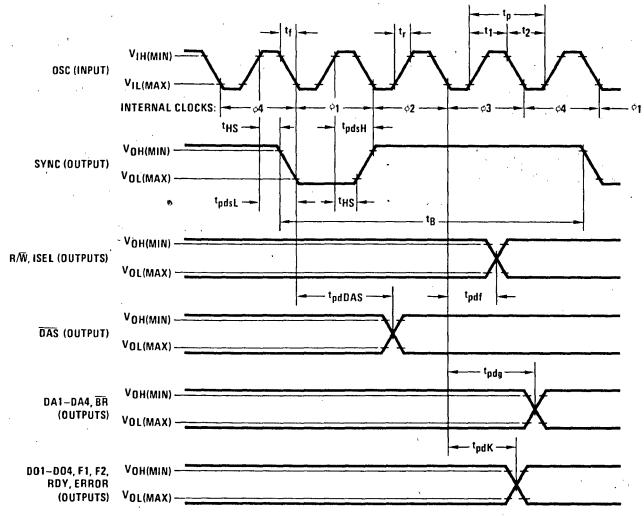



FIGURE 1

## Switching Time Waveforms




**Note 1:** See discussion of timing diagrams on page 5.

**Note 2:** Osc. Duty Cycle = $t_1$ $(t_1 + t_2) = t_1/t_p$.

**Note 3:** The last four timing diagrams indicate that, *if* the output changes, it will change within the indicated time. This is not meant to imply that these signals will change every microcycle. The conditions which will cause the various outputs to change are explained in detail in the functional description section of this manual.

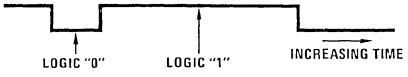
FIGURE 2. Timing Diagrams for Oscillator Input and Chip Outputs

## Timing Diagram Conventions

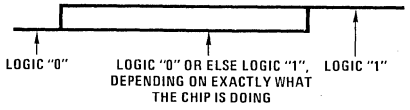
This data sheet makes extensive use of timing diagrams to illustrate electrical and logical characteristics of signal inputs and outputs. To avoid confusion concerning these diagrams, the following conventions have been adopted:

1. Time axis is horizontal, increasing to the right.

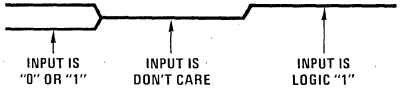
2. Upper side of waveform represents logic "1" ($V_{SS}$). Lower side of waveform represents logic "0" ($V_{DD}$).



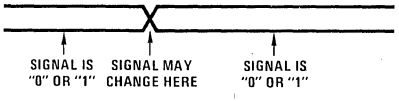

3. Lines appearing simultaneously at both logic "0" and logic "1" indicate that the state of the input or output is either "0" or "1", and does not change during this time. This is used when the logic state depends on exactly what the user is doing with the chip at the time, and thus is unknown to the person drawing the timing waveform.



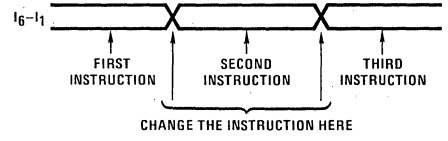

4. Lines located at a level between logic "1" and logic "0" appear on input signals only, and indicate that the state of the input is a don't care during this time.



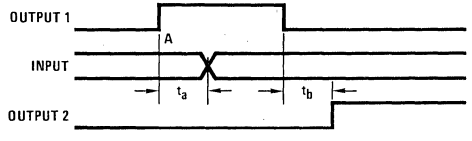

5. An "X" in a waveform indicates that the input or output may change state at this time.




This representation is often used for a group of inputs or outputs whose logic states are unknown, but the time at which a signal may change must be shown. Example:




6. Minimum and/or maximum time values are sometimes specified. The interpretation of these values depends on whether the waveforms are inputs or outputs. Example:



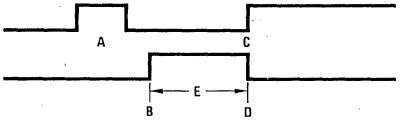

The following table shows three different ways in which these timing diagrams can be interpreted, depending on whether min, max or both are specified.

| PARAMETER | | MIN | MAX | COMMENT |
|---|---|---|---|---|
| (A) | $t_a$ | 3.7 $\mu$s | | Input must be changed later than 3.7 $\mu$s after point "A" |
| | $t_b$ | | 1.9 ms | Output 2 will go high no later than 1.9 ms after output 1 goes low |
| (B) | $t_a$ | | 2.4 ns | Input must be changed before 2.4 ns after point "A" |
| | $t_b$ | 0.7 s | | Output 2 will not go high until at least 0.7 s after output 1 goes low |
| (C) | $t_a$ | 0.6 $\mu$s | 3.0 $\mu$s | Input must be changed between 0.6 and 3.0 $\mu$s after point "A", no sooner and no later |
| | $t_b$ | 0.01 $\mu$s | 0.9 $\mu$s | Output 2 will go high between 0.01 and 0.9 $\mu$s after output 1 goes low |

This illustrates the various meanings that must be attached to time values, depending on whether they refer to inputs or outputs and whether minimum, maximum or min/max limits are placed on the value.

7. Rise and fall times are measured from maximum logic low to minimum logic high. For example, if the maximum logic low ("0") level ($V_{L(MAX)}$) of a signal is $V_{DD}$ + 1V, and if the minimum logic high ("1") level ($V_{H(MIN)}$) is $V_{SS}$ – 1.5V, then the rise time would be the time it takes the signal to go from $V_{DD}$ + 1V to $V_{SS}$ – 1.5V.

Timing diagrams are seldom shown to scale because of space limitations. However, they do show the proper relationship between waveforms. Consequently, the reader, when studying a timing diagram, should exercise care in understanding what information the timing diagram is meant to show, and ignore the time scale distortions that are necessarily introduced. Example:




Waveform relationships are maintained, so pulse A does come before B, and C and D occur at the same time. However, the time axis may be distorted, so pulse width E may not be twice that of pulse A. It may be 10 times, or even 1000 times wider.

## Pinout Description

The MM57109 is intended for microprocessor number processing applications, either as a microcomputer peripheral chip or as a stand-alone processor. *Figure 1* shows a pinout diagram of the MM57109, giving the pin numbers and names of the signal lines. It also shows a functional block diagram illustrating the internal organization of the MM57109 and the origin of the signal lines that are used to communicate with the external world.

The MM57109 operates on a 9V power supply. In order to make it TTL compatible, it can be operated from supplies of 5V and −4V. The signal inputs are designed to respond properly to LPTTL logic levels (with the exception of OSC, HOLD and POR) when the MM57109 is operated in this fashion. (See electrical specifications and *Figure 3* for details on LPTTL interface).

A 400 kHz oscillator operating between $V_{DD}$ and $V_{SS}$ is required. The rise and fall times and frequency of this oscillator are not critical, making it relatively easy to generate. The MM57109 provides a SYNC output, which is a signal that goes active low once every 4 oscillator cycles. A single SYNC pulse corresponds to a single "microcycle" (about 10 $\mu$s). The execution of a single MM57109 instruction involves thousands of microcycles. A later section of this manual will contain a tabulation of instruction execution times listed in microcycles.

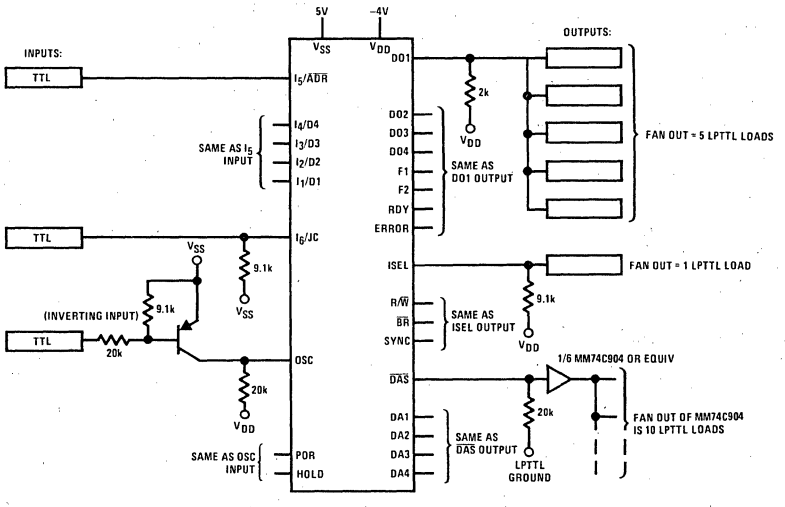
The processor is reset by applying a reset pulse to the POR pin as shown in *Figure 6*. The chip will then set the various outputs to their proper levels and generate three ready pulses (RDY). These ready pulses are designed to provide for automatic processing of an error in stand-alone systems. (See section titled ERROR CONTROL.) A microcomputer system would ignore the first two RDY pulses and use the third one as a "Ready for Instruction" signal.

The MM57109 has 6 instruction inputs ($I_6$–$I_1$) which are used to provide it with a 6-bit instruction code (commonly referred to as an "op code"). Each op code corresponds to one of the MM57109 instructions. A list of instructions, their op codes, and a description of the operations they perform will be given later in this manual. The 6 instruction lines are shared by 6 data lines. The output ISEL identifies which function the 6 lines are performing. When ISEL = 1, the 6 lines are instruction lines ($I_6$–$I_1$). When ISEL = 0, the 6 lines are data lines (JC, $\overline{ADR}$, D4–D1). In many cases the data lines, which are associated with the IN, AIN, and TJC instructions, will not be used. In these instances ISEL can be ignored. If the data lines are used, ISEL is the select input for six 2–1 multiplexers or an enable input to buffers, latches or ROMs. Later in this manual sample systems will be shown illustrating use of ISEL.

A ready output (RDY) goes high when the processor is ready to read a 6-bit instruction code. This output operates in conjunction with the HOLD input. When RDY goes high, it will remain high if HOLD = 1. If processor instructions are not always ready when RDY goes high, some method must be provided to set HOLD = 1. A microprocessor might have a flag output which holds HOLD = 1 until it is ready to pass an instruction to the processor. (If instructions are always ready when RDY goes high, the HOLD input can be tied to "0".) After RDY goes high, it will wait for HOLD = 0 and then go low again. At this time the 6-bit instruction code is read and the instruction is performed.

The branch output ($\overline{BR}$) is a 4-microcycle active low pulse which signals that the result of a test instruction (e.g. TEST X = 0) was true. This pulse starts prior to RDY = 1 for the next instruction, and ends slightly after RDY = 1.

The four signals RDY, HOLD, ISEL and $\overline{BR}$ were carefully chosen to allow the MM57109 to be used as a stand-alone processor or as a microcomputer peripheral. In a stand-alone system, RDY would be a clock for an external program counter (PC) whose outputs would address a ROM containing the MM57109 instructions. $\overline{BR}$ would parallel load the PC, resulting in a program branch. In a microcomputer system, RDY would inform




**Note 1:** $I_5$–$I_1$ have an internal resistor to $V_{SS}$. $I_6$ has an internal resistor to $V_{DD}$.
**Note 2:** All inputs must be driven by TTL or LPTTL outputs with no other loads on them.

**FIGURE 3. Low Power TTL Interface**

## Pinout Description (Continued)

the microcomputer that the MM57109 is ready for a new instruction. HOLD would be used to inform the MM57109 that the microcomputer is not ready to respond to the MM57109.

Several instructions have conditions which will cause an error to occur. Table VI enumerates these conditions. When an error does occur, the MM57109 will set the ERROR output high. This output can be tested with the TERR instruction and cleared with the ECLR instruction.

The two outputs F1 and F2 are flags which are set by the instructions SF1 and SF2. They can also be pulsed active high with the instructions PF1 and PF2. These flag outputs could be used as single bit outputs from the MM57109.

The JC input is a general purpose single bit input. A TJC instruction will branch (i.e., result in a true condition causing a $\overline{BR}$ pulse) if the input JC is high. Otherwise the TJC instruction will do nothing.

Table II summarizes this description of the MM57109 signal lines.

### INPUTTING DATA

As shown in *Figure 1*, the MM57109 has an internal register file. Each of the 5 registers (X, Y, Z, T and M) has 8 mantissa digits, 2 exponent digits, a decimal point position indicator, and mantissa and exponent sign bits. Instructions operate on these registers. The instructions IN and OUT input and output numbers to and from the X register. There are two possible modes of operation for IN and OUT instructions. Floating point mode transfers mantissa digits, a mantissa sign digit, and a decimal point position digit. Scientific notation mode transfers mantissa digits, 2 exponent digits, a digit containing mantissa and exponent sign bits, and a decimal point position indicator. Initially the MM57109 is in the floating point mode. The TOGM instruction toggles to the opposite mode. The number of mantissa digits input or output by an IN or OUT instruction is equal to the mantissa digit count (MDC). The MDC is initially 8 and can be set to any value from 1 to 8 using the SMDC instruction. When an IN or OUT instruction is executed, the four DA outputs will sequence through values indicating which digit is to be input or output. The section of this manual entitled DATA FORMATS gives the values of the DA lines for each of the digits input or output by an IN or OUT instruction. During an OUT instruction, the four DO outputs provide the digit outputs, coded in BCD. The R/$\overline{W}$ output is pulsed active low once for each digit. This R/$\overline{W}$ pulse can be used to write the data into a RAM or clock it into a latch. During an IN instruction, the four I lines ($I_4$–$I_1$), are data input lines for the digits to be input (and so are also named D4–D1). The same data format is used for IN as is used for OUT. The $\overline{DAS}$ output is pulsed active low prior to reading each digit. This $\overline{DAS}$ pulse can be used as a data request signal or to clock data into a latch.

The IN and OUT instructions have been designed to allow easy expansion of the internal register file. A 256 x 4 RAM will add an additional 16 registers for data storage. The DA lines are used to provide part of the RAM address. The rest of the address, which would specify one of the 16 registers, comes from the external instruction storage (ROM, microprocessor, etc). The DO lines are the input to the RAM, while the RAM outputs are multiplexed to the I lines, using ISEL to select between instructions or data. The processor R/$\overline{W}$ line is the RAM R/$\overline{W}$ signal.

There are three ways to input data to the MM57109. The first is the IN instruction which has already been described. Second is the AIN instruction, which inputs a single digit into the X register. Multiple AIN instructions will input more than one digit to the X register, since the AIN instruction does not cause termination of the number entry mode (number entry mode will be fully described later in this manual). The DA lines provide a digit address from 0 to 7 for multiple AIN instructions. The $\overline{ADR}$ input (shared with $I_6$) is a data hold signal for AIN. If $\overline{ADR}$ is high during an AIN instruction, the processor will wait till it goes low, and then read the digit on D4–D1. Finally, the F2 output of the MM57109 will be pulsed active low as a read acknowledge signal. Note that only mantissa digits, not exponents or signs, can be entered with AIN.

For systems using a microcomputer with the MM57109 as a peripheral, it is likely that neither the IN nor the AIN instruction would be used. Instead, the third method of inputting data to the processor would be used. This method involves entering numbers as instructions. Using the instructions "0", "1", "2", . . . "9", the decimal point instruction, etc., a number can be entered directly into the processor in the same manner as one presses keys to enter numbers into a calculator. The decimal point is to the right of the last digit entered unless it was fixed by a DP instruction. The EE instruction causes the next digits to be entered into the exponent. The CS instruction changes the sign of the mantissa (or exponent, if EE instruction was entered).

### 2-WORD INSTRUCTIONS

Several instructions are 2-word instructions, of which there are 4 types. Each type generates two RDY pulses, one for each word. The first type are the inverse instructions (inverse SIN, COS, TAN and inverse +, –, x, / for memory operations). These instructions require that the INV instruction first be executed, followed by the desired instruction (SIN, COS, etc.). The second type is the SMDC instruction. The second word of this instruction is the mantissa digit count, a BCD number from 1 to 8. The third type is the IN and OUT instructions. The second word of these instructions is a high order address for a RAM or a device select code. It is not necessary to use the second word of IN or OUT instructions because the MM57109 ignores it, providing only a RDY pulse that may or may not be used by external hardware. The final type of 2-word instructions are the branch instructions. The second word of these instructions is intended to be a branch address to be loaded into an external program counter in stand-alone systems. For a microprocessor system, the second RDY pulse can be used to clock the $\overline{BR}$ output into a latch. The latch can then be tested to discover if the branch condition was true ($\overline{BR}$ = 0) or false ($\overline{BR}$ = 1). Many microcomputer applications will not use the branch instructions, since testing and branching is often more easily done within the external microprocessor itself.

## Pinout Description (Continued)

TABLE II. MM57109 PIN DESCRIPTION

| ABBREVIATED NAME | PIN NUMBER | FUNCTIONAL NAME | DESCRIPTION |
|---|---|---|---|
| $V_{SS}$, $V_{DD}$ | 15, 21 | $V_{SS}$, $V_{DD}$ (Power Supply) | $V_{SS}$ = $V_{DD}$ + 9V nominally (see electrical specifications) ($V_{SS}$ = Logic "1", $V_{DD}$ = Logic "0"). |
| POR | 11 | Power On Reset (Input) | An 8-microcycle or longer active high pulse at this input will initialize the MM57109. It then sets R/$\overline{W}$ = 1, other outputs = 0, and generates three RDY pulses before reading first instruction. HOLD must be 0 to complete each RDY pulse. |
| OSC | 7 | Oscillator (Input) | Single $\phi$ clock with frequency 4X microcycle time. Typical frequency is 400 kHz. |
| SYNC | 6 | Sync (Output) | Active low output pulse once each microcycle. |
| RDY | 12 | Ready (Output) | Rising edge indicates processor is ready to execute next instruction or get second word of 2-word instruction. If HOLD = 0, RDY goes low again and next instruction is executed. If HOLD = 1, RDY stays high until HOLD = 0. (See *Figure 8*). RDY can be used to clock an external program counter or to request an instruction from another CPU. |
| HOLD | 9 | Hold (Input) | When set high prior to or at the rising edge of RDY, RDY will be held high and instruction execution delayed until HOLD is set low. |
| $\overline{BR}$ | 23 | Branch (Output) | A 4-microcycle active low pulse indicates a program branch. RDY goes high during this pulse. $\overline{BR}$ may be used as a load signal for an external PC or as a sense input to a microprocessor. |
| ISEL | 8 | Instruction Select (Output) | Selects 6 bit instruction code (ISEL = 1) or JC, $\overline{ADR}$, D4–D1 (ISEL = 0) on $I_6$–$I_1$ (the 6 input lines). |
| R/$\overline{W}$ | 10 | Read/$\overline{Write}$ (Output) | Pulsed active low during OUT instruction to write data digits into a RAM or register. Address and data are valid at both edges. R/$\overline{W}$ is also pulsed during a PRW1 or PRW2 instruction. |
| $I_6$, JC | 24 | Input 6, Jump Condition (Input) | Most significant instruction bit when ISEL = 1. Jump condition for TJC instruction when ISEL = 0. (JC = 1 indicates jump condition true.) |
| $I_5$, $\overline{ADR}$ | 5 | Input 5, AIN Data Ready (Input) | Instruction bit 5 when ISEL = 1. AIN Data Ready ($\overline{ADR}$) for AIN instruction when ISEL = 0, ($\overline{ADR}$ = 0 for data ready). |
| $I_4$–$I_1$, D4–D1 | 4, 3, 2, 1 | Inputs 4–1, Data 4–1 (Inputs) | Instruction bits 4–1, or mantissa digit count on second word of SMDC instruction, when ISEL = 1. Digit data (AIN or IN instructions) when ISEL = 0. Bit 4 is the most significant bit. |
| DA4–DA1 | 25, 26, 27, 28 | Digit Address 4–1 (Outputs) | Digit address for AIN, IN, and OUT instructions. Used as multiplex selector (AIN) or as low order address (IN, OUT) for RAM or other I/O device. Bit 4 is the most significant bit. Set to 0 after each IN, OUT or AIN instruction. |
| $\overline{DAS}$ | 22 | Digit Address Strobe (Output) | Active low pulse indicates digit address is changing. New address is valid on second (positive-going) edge. |
| DO4–DO1 | 20, 19, 18, 17 | Digit Outputs 4–1 (Outputs) | BCD digit output for OUT instruction. Set to 0 after each OUT instruction. Bit 4 is the most significant bit. |
| F1 | 16 | Flag 1 (Output) | User controlled flag can be set or pulsed (reset if set). |
| F2 | 14 | Flag 2 (Output) | User controlled flag can be set or pulsed (reset if set). Active low pulse (set if reset) generated after each AIN data read. This can be used as an acknowledge signal to clear a flip-flop. |
| ERROR | 13 | Error Flag (Output) | Set on an arithmetic or OUT error. Reset by ECLR instruction. See ERROR CONTROL for more information. |

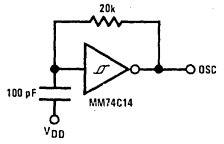
## Functional Description

### OSCILLATOR GENERATION

*Figure 4* shows a simple circuit for generation of the MM57109 oscillator.

### INITIALIZATION SEQUENCE

*Figure 5* shows a flowchart and *Figure 6* a timing diagram of the MM57109 initialization sequence which occurs when the POR input is set high for at least 8 clock periods.




$f_{OSC} \cong 400$ kHz

$T_{OSC} \cong 2.5\ \mu s$

FIGURE 4. MM57109 Oscillator Circuit

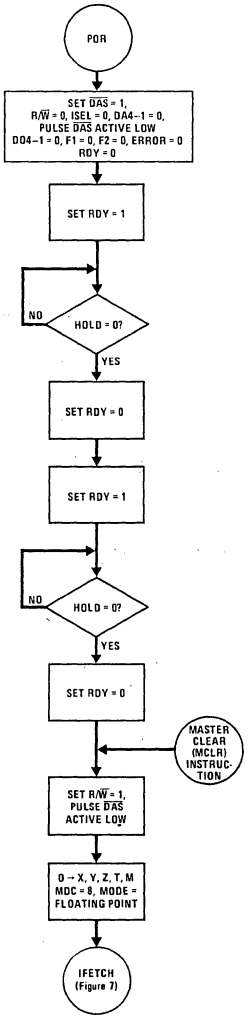
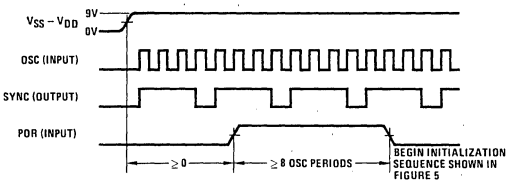
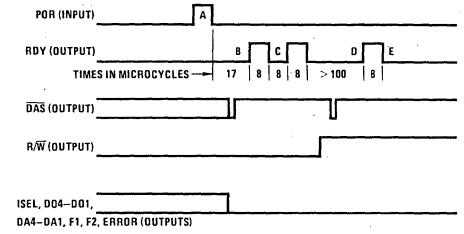



FIGURE 5. MM57109 Initialization Flowchart




(a) POR Sequence for Initialization




Note: Hold = 0

| POINTS | DESCRIPTION (Assume External PC) | |
|---|---|---|
| A | Initialize PC to 0 | Error Recovery Locations |
| B | Advance PC to location 1 | |
| C | Advance PC to location 2 | |
| D | Advance PC to location 3, MM57109 initialized | |
| E | Begin execution of instruction at location 3 | |

(b) Output Timing During Initialization

FIGURE 6. MM57109 Initialization Timing Diagram

## Functional Description (Continued)

### INSTRUCTION FETCH AND EXECUTION

*Figure 7* shows a flowchart and *Figure 8* shows a timing diagram of the instruction fetch and execution sequence.

After initialization (POR) or the completion of an instruction, the processor raises RDY to signal the instruction store device that the processor is ready for the next instruction word. The instruction store device could be a semiconductor memory, host CPU, or an asynchronous device of some kind. If the instruction store is not ready to respond within the required access time (8 microcycles), it must raise the HOLD input to delay the instruction word fetch. HOLD may be set high any time while RDY is low, or at the leading edge of RDY. When HOLD goes low the processor will lower RDY and begin instruction execution. The instruction word must remain valid while RDY is low.

During program branches, skips, or fetching of the second word of a 2-word instruction, the RDY/HOLD sequence is the same as discussed above. (See flowchart in *Figure 7* and timing diagram in *Figure 8(e)*.)

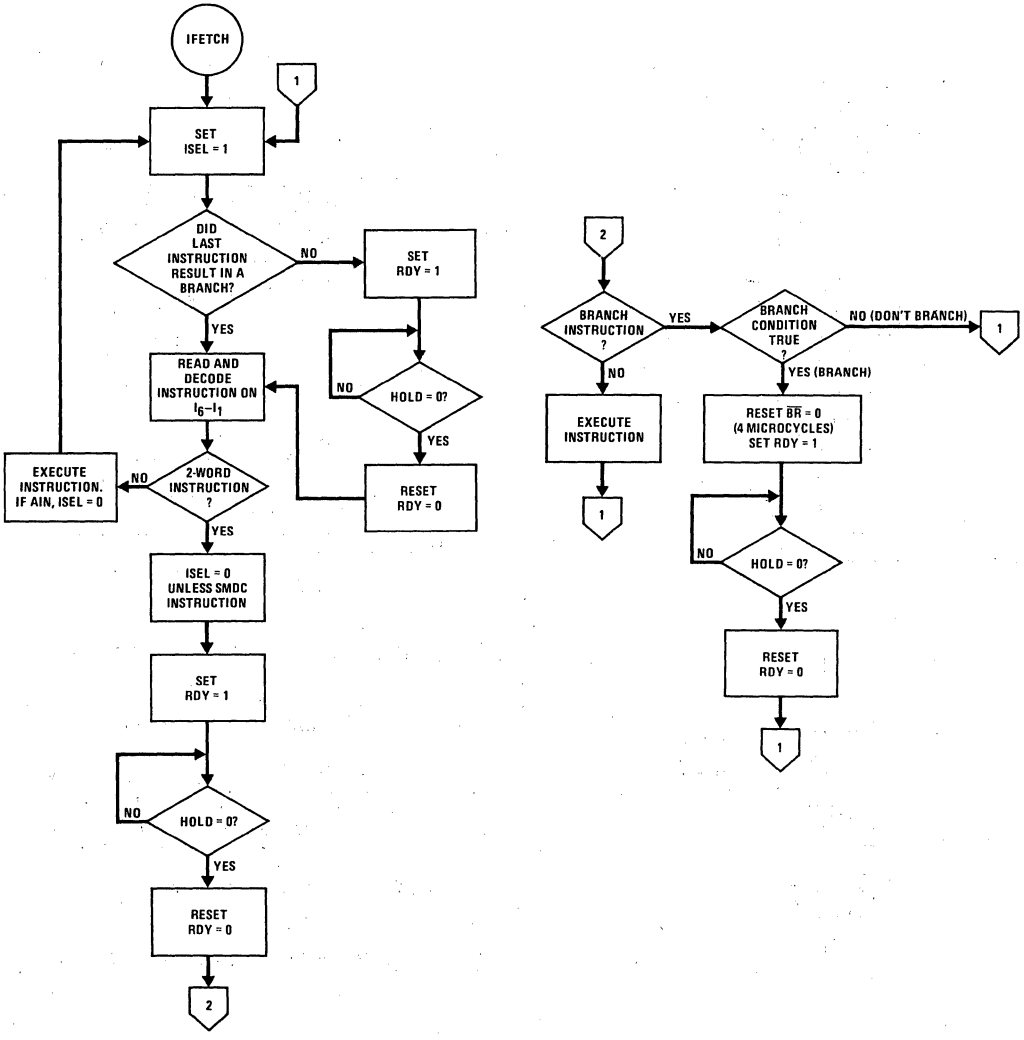
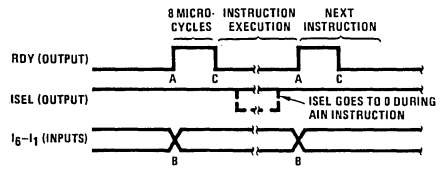


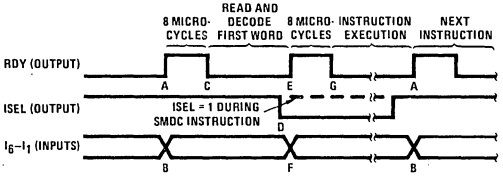

**FIGURE 7. MM57109 Instruction Fetch and Execution Flowchart**

## Functional Description (Continued)



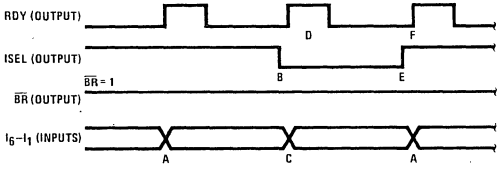

| POINTS | DESCRIPTION |
|---|---|
| A | RDY goes high |
| B | Next instruction becomes available on I lines |
| C | RDY goes low, instruction is read and executed by MM57109 |

(a) 1-Word Instruction with HOLD = 0



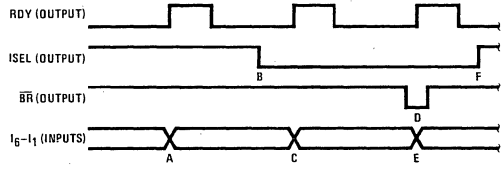

| POINTS | DESCRIPTION |
|---|---|
| A | RDY goes high for first instruction word |
| B | First instruction word becomes available on I lines |
| C | RDY goes low. Instruction is read and decoded by MM57109 |
| D | ISEL goes low for second instruction word |
| E | RDY goes high for second instruction word |
| F | Second instruction word becomes available on I lines |
| G | Instruction is executed by MM57109 |

(b) 2-Word Instruction with HOLD = 0



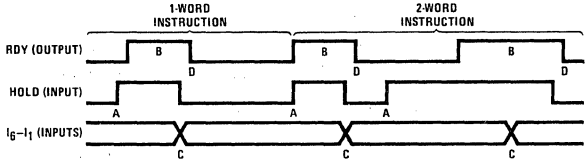

| POINTS | DESCRIPTION |
|---|---|
| A | First word of next instruction becomes available on I lines |
| B | ISEL = 0 for second word |
| C | Second word (branch address) becomes available to external program counter |
| D | Data becomes available on JC for TJC instruction |
| E | ISEL = 1 for next instruction |
| F | RDY pulse for next instruction occurs |

(c) 2-Word Branch Instruction that Doesn't Branch (HOLD = 0)




| POINTS | DESCRIPTION |
|---|---|
| A | First word of branch instruction becomes available on I lines |
| B | ISEL = 0 for second word |
| C | Second word (branch address) becomes available to external program counter |
| D | Branch condition is true so 4-microcycle active low $\overline{BR}$ pulse occurs and RDY is pulsed with leading edge occurring during $\overline{BR}$ pulse. This clocks branch address into program counter |
| E | New instruction at branch address becomes available to MM57109 |
| F | ISEL = 1 for next instruction. RDY pulse that usually follows ISEL = 1 is suppressed |

(d) 2-Word Branch Instruction that Does Branch (HOLD = 0)




| POINTS | DESCRIPTION |
|---|---|
| A | HOLD = 1 at or prior to RDY leading edge (RDY leading edge may be used to set HOLD = 1) |
| B | RDY stays high until HOLD = 0 |
| C | When HOLD goes low, next instruction word must be available on I lines |
| D | RDY goes low after HOLD goes low |

(e) RDY-HOLD Relationship for 1 and 2-Word Instructions

FIGURE 8. MM57109 Instruction Fetch and Execution Timing Diagrams

## Functional Description (Continued)

### MM57109 INSTRUCTION SET

The MM57109 has 70 instructions available to the user. The 70 instructions are classed into digit entry, move, math, clear, branch, input/output, and mode control instructions. Table III contains a detailed description of these instructions. *Figure 9* shows the instruction format. Table IV contains a summary of the instructions. Note that all arithmetic instructions operate on 8-digit mantissa/2-digit exponent numbers, regardless of mantissa digit count or notation mode. Accuracy of all instructions is 7 digits. Computations are performed internally with 9 digits, then rounded to 8. The eighth digit is accurate to ±5.

### NUMBER ENTRY

When a digit, decimal point, or $\pi$ is entered with an AIN, 0–9, DP, or PI instruction, the stack is first pushed and the X register cleared: $Z \rightarrow T$, $Y \rightarrow Z$, $X \rightarrow Y$, $0 \rightarrow X$. This process is referred to as "initiation of number entry." Following this, the entered digit and future digits are loaded into the X mantissa. Subsequent entry of digits or DP, EE, or CS instructions do not cause initiation of number entry. Digits following the eighth mantissa digit are ignored. (CAUTION: An internal error will occur if more than 8 digits are entered with AIN instruction, or if a non-BCD digit is entered. Note that the ERROR flag with *not* be set if this happens. A POR sequence will be necessary to restart the processor.) This number entry mode is terminated by any instruction except 0–9, DP, EE, CS, PI, AIN or HALT. Termination of number entry means two things. First, the number is normalized by adjusting the exponent and decimal point position so that the decimal point is to the right of the first mantissa digit. Second, the next digit, decimal point, or $\pi$ entered will again cause initiation of number entry, as already described. There is one exception to this number entry initiation rule: the stack is *not* pushed if the instruction prior to the entered digit was an ENTER. However, the X register is still cleared and the entered digit put in X.

The IN instruction enters *all* digits of a number. Therefore, IN does not cause initiation of number entry. However, it does terminate number entry mode if the processor is in this mode before the IN instruction is executed. This means the user can mix 0–9, AIN and IN instructions without performing an ENTER before an IN.

The IN instruction will always push the stack prior to inputting digits unless the previous instruction was ENTER. This allows multiple IN instructions to be executed without performing an ENTER between them.

### INSTRUCTION TIMING

Table V shows execution times of each instruction. These times are shown in microcycles, 1 microcycle being equal to 1 SYNC period (approximately 10 $\mu$s). *Figure 10* shows timing diagrams illustrating the dynamic characteristics of execution of each type of instruction, assuming HOLD = 0.

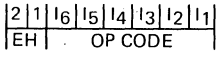
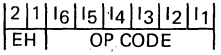
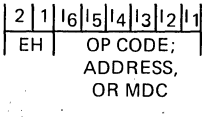
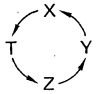
| SINGLE WORD INSTRUCTIONS | 2-WORD INSTRUCTIONS |
|---|---|
|  |  <br>  |
| The "OP CODE" is a 6-bit operation code (see Table III) which specifies which instruction is to be performed.<br><br>The MM57109 requires only the OP CODE to function. However, memory devices with 8-bit word sizes are often used. The extra two bits, here designated "EH" for "external hardware", could be used for device selection on AIN instructions, etc. | The first word has the same format as a 1-word instruction.<br><br>The function and format of the second word depends on the OP CODE of the first word:<br><br>(a) First word = SMDC instruction. The second word contains the MDC (1–8).<br><br>(b) First word = IN or OUT instruction. The second word contains a high-order address for RAM or I/O device (low-order address from DA lines).<br><br>(c) First word = INV instruction. The second word contains the second OP CODE for the instruction (M+, M–, MX, M/, $SIN^{-1}$, $COS^{-1}$, $TAN^{-1}$).<br><br>(d) First word = branch instruction. The second word contains the branch address to be loaded into PC on branch. |

**FIGURE 9. MM57109 Instruction Format**

## Functional Description (Continued)

TABLE III. MM57109 INSTRUCTION DESCRIPTION TABLE (* INDICATES 2-WORD INSTRUCTION)

| CLASS | SUBCLASS | MNEMONIC* | OCTAL OP CODE | FULL NAME | DESCRIPTION |
|---|---|---|---|---|---|
| Digit Entry | | 0 | 00 | 0 | Mantissa or exponent digits. On first digit (d) the following occurs: Z → T, Y → Z, X → Y, d → X. See description of number entry on page 12. |
| | | 1 | 01 | 1 | |
| | | 2 | 02 | 2 | |
| | | 3 | 03 | 3 | |
| | | 4 | 04 | 4 | |
| | | 5 | 05 | 5 | |
| | | 6 | 06 | 6 | |
| | | 7 | 07 | 7 | |
| | | 8 | 10 | 8 | |
| | | 9 | 11 | 9 | |
| | | DP | 12 | Decimal Point | Digits that follow will be mantissa fraction. |
| | | EE | 13 | Enter Exponent | Digits that follow will be exponent. |
| | | CS | 14 | Change Sign | Change sign of exponent or mantissa. Xm = X mantissa. Xe = X exponent. CS causes −Xm → Xm or −Xe → Xe depending on whether or not an EE instruction was executed after last number entry initiation. |
| | | PI | 15 | Constant $\pi$ | 3.1415927 → X, stack not pushed. |
| | | EN | 41 | Enter | Terminates digit entry and pushes the stack. The argument entered will be in X and Y. Z → T, Y → Z, X → Y |
| | | NOP | 77 | No Operation | Do nothing instruction that will terminate digit entry. |
| | | HALT | 17 | Halt | External hardware detects HALT op code and generates HOLD = 1. Processor waits for HOLD = 0 before continuing. HALT acts as a NOP and may be inserted between digit entry instructions since it does not terminate digit entry. |
| Move | | ROLL | 43 | Roll | Roll Stack.   |
| | | POP | 56 | Pop | Pop Stack. Y → X, Z → Y, T → Z, 0 → T |
| | | XEY | 60 | X exchange Y | Exchange X and Y. X ⟷ Y |
| | | XEM | 33 | X exchange M | Exchange X with memory. X ⟷ M |
| | | MS | 34 | Memory Store | Store X in Memory. X → M |
| | | MR | 35 | Memory Recall | Recall Memory into X. Stack is pushed. M → X → Y → Z → T |
| | | LSH | 36 | Left Shift Xm | X mantissa is left shifted while leaving decimal point in same position. Former most significant digit is saved in link digit. Least significant digit is zero. Former link digit is lost. |
| | | RSH | 37 | Right Shift Xm | X mantissa is right shifted while leaving decimal point in same position. Link digit, which is normally zero except after a left shift, is shifted into the most significant digit. Least significant digit is lost. |

## Functional Description (Continued)

TABLE III. MM57109 INSTRUCTION DESCRIPTION TABLE (CONTINUED) (*INDICATES 2-WORD INSTRUCTION)

| CLASS | SUBCLASS | MNEMONIC* | OCTAL OP CODE | FULL NAME | DESCRIPTION |
|---|---|---|---|---|---|
| Math | F (X,Y) | + | 71 | Plus | Add X to Y. X + Y → X. On +, −, x, / and YX instructions, stack is popped as follows: Z → Y, T → Z, O → T. Former X, Y are lost. |
| | | − | 72 | Minus | Subtract X from Y. Y − X → X |
| | | x | 73 | Times | Multiply X times Y. Y x X → X |
| | | / | 74 | Divide | Divide X into Y. Y ÷ X → X |
| | | YX | 70 | Y to X | Raise Y to X power. $Y^X \rightarrow X$ |
| | F (X,M) | INV +* | 40, 71 | Memory Plus | Add X to memory. M + X → M. On INV +, −, x and / instructions, X, Y, Z, and T are unchanged. Former M is lost. |
| | | INV −* | 40, 72 | Memory Minus | Subtract X from memory. M − X → M |
| | | INV x* | 40, 73 | Memory Times | Multiply X times memory. M x X → M |
| | | INV /* | 40, 74 | Memory Divide | Divide X into memory. M ÷ X → M |
| | F (X) Math | 1/X | 67 | One Divided by X | 1 ÷ X → X. On all F (X) math instructions Y, Z, T and M are unchanged and previous X is lost. |
| | | SQRT | 64 | Square Root | $\sqrt{X} \rightarrow X$ |
| | | SQ | 63 | Square | $X^2 \rightarrow X$ |
| | | 10X | 62 | Ten to X | $10^X \rightarrow X$ |
| | | EX | 61 | e to X | $e^X \rightarrow X$ |
| | | LN | 65 | Natural log of X | ln X → X |
| | | LOG | 66 | Base 10 log of X | log X → X |
| | F (X) Trig | SIN | 44 | Sine X | SIN(X) → X. On all F(X) trig functions, Y, Z, T, and M are unchanged and the previous X is lost. |
| | | COS | 45 | Cosine X | COS(X) → X |
| | | TAN | 46 | Tangent X | TAN(X) → X |
| | | INV SIN* | 40, 44 | Inverse sine X | $SIN^{-1}(X) \rightarrow X$ |
| | | INV COS* | 40, 45 | Inverse cosine X | $COS^{-1}(X) \rightarrow X$ |
| | | INV TAN* | 40, 46 | Inverse tan X | $TAN^{-1}(X) \rightarrow X$ |
| | | DTR | 55 | Degrees to radians | Convert X from degrees to radians. |
| | | RTD | 54 | Radians to degrees | Convert X from radians to degrees. |
| Clear | | MCLR | 57 | Master Clear | Clear all internal registers and memory; initialize I/O control signals, MDC = 8, MODE = floating point. (See INITIALIZATION.) |
| | | ECLR | 53 | Error flag clear | O → Error flag |
| Branch | Test | JMP* | 25 | Jump | Unconditional branch to address specified by second instruction word. On all branch instructions, second word contains branch address to be loaded into external PC. |
| | | TJC* | 20 | Test jump condition | Branch to address specified by second instruction word if JC ($I_6$) is true (=1). Otherwise, skip over second word. |
| | | TERR* | 24 | Test error flag | Branch to address specified by second instruction word if error flag is true (=1). Otherwise, skip over second word. May be used for detecting specific errors as opposed to using the automatic error recovery scheme dealt with in the section on Error Control. |
| | | TX = 0* | 21 | Test X = 0 | Branch to address specified by second instruction word if X = 0. Otherwise, skip over second word. |
| | | TXF* | 23 | Test $\|X\| < 1$ | Branch to address specified by second instruction word if $\|X\| < 1$. Otherwise, skip over second word. (i.e. branch if X is a fraction.) |
| | | TXLT0* | 22 | Test $X < 0$ | Branch to address specified by second instruction word if $X < 0$. Otherwise, skip over second word. |

## Functional Description (Continued)

TABLE III. MM57109 INSTRUCTION DESCRIPTION TABLE (CONTINUED) (* INDICATES 2-WORD INSTRUCTION)

| CLASS | SUBCLASS | MNEMONIC* | OCTAL OP CODE | FULL NAME | DESCRIPTION |
|---|---|---|---|---|---|
| Branch | Count | IBNZ | 31 | Increment memory and branch if $M \neq 0$ | $M + 1 \rightarrow M$. If $M = 0$, skip second instruction word. Otherwise, branch to address specified by second instruction word. |
| | | DBNZ | 32 | Decrement memory and branch if $M \neq 0$ | $M - 1 \rightarrow M$. If $M = 0$, skip second instruction word. Otherwise, branch to address specified by second instruction word. |
| I/O | Multi-digit | IN* | 27 | Multidigit input to X | The processor supplies a 4-bit digit address (DA4–$\overline{DA1}$) accompanied by a digit address strobe ($\overline{DAS}$) for each digit to be input. The high order address for the number to be input would typically come from the second instruction word. The digit is input on D4–D1, using ISEL = 0 to select digit data instead of instructions. The number of digits to be input depends on the calculation mode (scientific notation or floating point) and the mantissa digit count (See DATA FORMATS and INSTRUCTION TIMING). Data to be input is stored in X and the stack is pushed ($X \rightarrow Y \rightarrow Z \rightarrow T$). At the conclusion of the input, DA4–DA1 = 0. |
| | | OUT* | 26 | Multidigit output from X | Addressing and number of digits is identical to IN instruction. Each time a new digit address is supplied, the processor places the digit to be output on DO4–DO1 and pulses the R/$\overline{W}$ line active low. At the conclusion of output, DO4–DO1 = 0 and DA4–DA1 = 0. |
| I/O | Single-digit | AIN | 16 | Asynchronous Input | A single digit is read into the processor on D4–D1. ISEL = 0 is used by external hardware to select the digit instead of instruction. It will not read the digit until $\overline{ADR}$ = 0 (ISEL = 0 selects $\overline{ADR}$ instead of $I_5$), indicating data valid. F2 is pulsed active low to acknowledge data just read. |
| I/O | Flags | SF1 | 47 | Set Flag 1 | Set F1 high, i.e. F1 = 1. |
| | | PF1 | 50 | Pulse Flag 1 | F1 is pulsed active high. If F1 is already high, this results in it being set low. |
| | | SF2 | 51 | Set Flag 2 | Set F2 high, i.e. F2 = 1. |
| | | PF2 | 52 | Pulse Flag 2 | F2 is pulsed active high. If F2 is already high, this results in it being set low. |
| | | PRW1 | 75 | Pulse R/$\overline{W}$ 1 | Generates R/$\overline{W}$ active low pulse which may be used as a strobe or to clock extra instruction bits into a flip-flop or register. |
| | | PRW2 | 76 | Pulse R/$\overline{W}$ 2 | Identical to PRW1 instruction. Advantage may be taken of the fact that the last 2 bits of the PRW1 op code are 01 and the last 2 bits of the PRW2 op code are 10. Either of these bits can be clocked into a flip-flop using the R/$\overline{W}$ pulse. |
| Mode Control | | TOGM | 42 | Toggle Mode | Change mode from floating point to scientific notation or vice-versa, depending on present mode. The mode affects only the IN and OUT instructions. Internal calculations are always in 8-digit scientific notation. |
| | | SMDC* | 30 | Set Mantissa Digit Count | Mantissa digit count is set to the contents of the second instruction word (=1 to 8). |
| | | INV* | 40 | Inverse Mode | Set inverse mode for trig or memory function instruction that will immediately follow. Inverse mode is for next instruction only. |

## Functional Description (Continued)

TABLE IV. MM57109 INSTRUCTION SUMMARY TABLE (* INDICATES 2-WORD INSTRUCTION)

| $I_4$–$I_1$ | $I_6I_5$ = 00 | 01 | 10 | 11 |
|---|---|---|---|---|
| 0000 | 0 | TJC* | INV* | XEY |
| 0001 | 1 | TX=0* | EN | EX |
| 0010 | 2 | TXLT0* | TOGM | 10X |
| 0011 | 3 | TXF* | ROLL | SQ |
| 0100 | 4 | TERR* | SIN (SIN$^{-1}$*) | SQRT |
| 0101 | 5 | JMP* | COS (COS$^{-1}$*) | LN |
| 0110 | 6 | OUT* | TAN (TAN$^{-1}$*) | LOG |
| 0111 | 7 | IN* | SF1 | 1/X |
| 1000 | 8 | SMDC* | PF1 | YX |
| 1001 | 9 | IBNZ* | SF2 | + (M+*) |
| 1010 | DP | DBNZ* | PF2 | – (M–*) |
| 1011 | EE | XEM | ECLR | x (MX*) |
| 1100 | CS | MS | RTD | / (M/*) |
| 1101 | PI | MR | DTR | PRW1 |
| 1110 | AIN | LSH | POP | PRW2 |
| 1111 | HALT | RSH | MCLR | NOP |

**Note 1:** HALT is same as NOP except it does not terminate number entry. External hardware must generate HOLD = 1 to halt.

**Note 2:** ISEL = 0 for AIN, all 2-word instructions except SMDC.

**Note 3:** All instructions with $I_6$ $I_5$ = 00, do not terminate number entry. Other instructions do terminate number entry.

TABLE V. INSTRUCTION EXECUTION TIMES

| INSTRUCTION MNEMONIC | EXECUTION TIME (MICROCYCLES) (AVERAGE) | EXECUTION TIME (MICROCYCLES) (WORST-CASE VALUES) | INSTRUCTION MNEMONIC | EXECUTION TIME (MICROCYCLES) (AVERAGE) | EXECUTION TIME (MICROCYCLES) (WORST-CASE VALUES) |
|---|---|---|---|---|---|
| 0–9 | | 238 | OUT | | 583 |
| DP | | 152 | IN | | 395 |
| EE | | 151 | SF1 | | 163 |
| CS | | 166 | PF1 | | 185 |
| PI | | 1312 | SF2 | | 163 |
| HALT | | 134 | PF2 | | 185 |
| AIN | | 284 | PRW1 | | 130 |
| TJC | | 208 | PRW2 | | 130 |
| TX=0 | | 278 | SIN | 56200 | 95900 |
| TXLT0 | | 197 | COS | 56200 | 95900 |
| TXF | | 277 | TAN | 35000 | 97600 |
| TERR | | 191 | INV SIN | 54000 | 93900 |
| JMP | | 186 | INV COS | 54000 | 93900 |
| IBNZ | | 2314 | INV TAN | 30200 | 92900 |
| DBNZ | | 2314 | LN | 24800 | 92000 |
| SMDC | | 163 | LOG | 30700 | 92600 |
| XEM | | 812 | EX | 30800 | 93900 |
| MS | | 839 | 10X | 27400 | 96500 |
| MR | | 1385 | +, – | 2200 | 6600 |
| LSH | | 168 | INV+, INV– | 1700 | 5000 |
| RSH | | 173 | (M+, M–) | | |
| INV | | 166 | x | 3200 | 22700 |
| EN | | 552 | INV x (MX) | 2700 | 21400 |
| TOGM | | 157 | / | 7800 | 22300 |
| ROLL | | 905 | INV / (M/) | 7300 | 21100 |
| ECLR | | 163 | 1/X | 4500 | 22800 |
| POP | | 448 | YX | 55400 | 95500 |
| MCLR | | 734 | SQRT | 7000 | 30200 |
| XEY | | 652 | SQ | 3000 | 21900 |
| NOP | | 122 | DTR, RTD | 9600 | 41700 |

**Note 1:** All times are measured from leading edge of ready for first word of the instruction to leading edge of ready for first word of the next instruction. (Hold = 0).

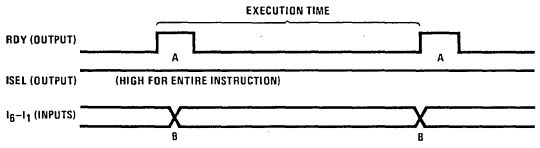
**Note 2:** Add 67 microcycles to the execution time of any instruction which initiates number entry and is preceded by an ENTER instruction.

**Note 3:** Add 282 microcycles to the execution time of any instruction which initiates number entry and is not preceded by an ENTER instruction.

**Note 4:** Add 1003 microcycles to the execution time of any instruction which terminates number entry.

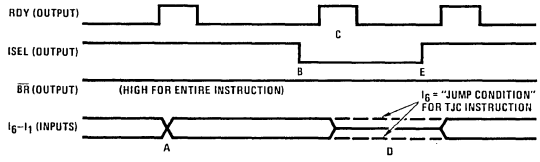
**Note 5:** The execution time of each instruction is a function of the internal state of the device and is not necessarily related to the number of digits in the operand. It is not possible to predict precisely what the execution time will be for any given instruction. This table shows worst-case values for basic instructions, and both average and worst-case values for mathematical instructions.

## Functional Description (Continued)



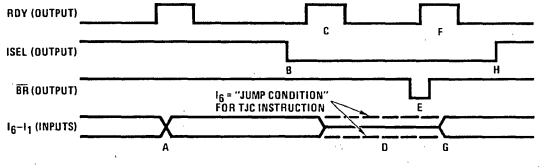

| POINTS | DESCRIPTION |
|---|---|
| A | RDY pulsed to indicate ready for next instruction. |
| B | Instruction is placed on I lines before RDY goes low. |

**(a) Timing Characteristics for All Instructions Except AIN, HALT, TJC, TX = 0, TXLT0, TXF, TERR, JMP, OUT, IN, SMDC, IBNZ, DBNZ, PF1, PF2, SF1, SF2, PRW1, PRW2, and ECLR**



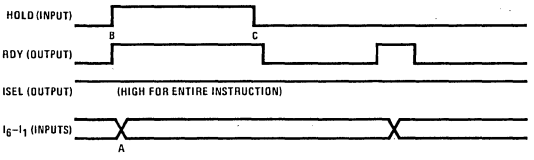

| POINTS | DESCRIPTION |
|---|---|
| A | Instruction is placed on I lines. |
| B | ISEL goes low for second word. |
| C | Second instruction word (= branch address) becomes available to external program counter. |
| D | I lines at this time are don't care, (exception: for TJC instruction $I_6$ (= JC) must contain the jump condition signal during this time). |
| E | ISEL goes high prior to RDY pulse for next instruction. |

**(b) Timing Characteristics for TJC, TX = 0, TXLT0, TXF, TERR, IBNZ, and DBNZ with Branch Condition False**



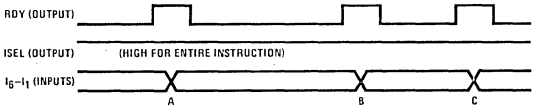

| POINTS | DESCRIPTION |
|---|---|
| A | Instruction is placed on I lines. |
| B | ISEL goes low for second word. |
| C | Second instruction word (= branch address) becomes available to external program counter. |
| D | I lines at this time are don't-care, (exception: for TJC instruction $I_6$ (= JC) must contain the jump condition signal during this time). |
| E | 4-microcycle $\overline{BR}$ active low pulse provides load signal for external program counter. |
| F | RDY leading edge occurs during $\overline{BR}$ pulse, thus loading program counter with branch address. |
| G | Next instruction appears on I lines prior to RDY going low. |
| H | ISEL goes high for first word of next instruction, RDY pulse that usually occurs at this time is suppressed. |

**(c) Timing Characteristics for TJC, TX = 0, TXLT0, TXF, TERR, IBNZ, and DBNZ with Branch Condition True. Also for JMP Instruction.**



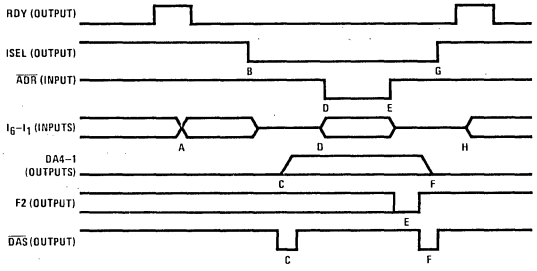

| POINTS | DESCRIPTION |
|---|---|
| A | Halt instruction is placed on I lines. |
| B | External hardware decodes halt instruction and puts HOLD high. (This is necessary only if the user wishes to have a HALT instruction). |
| C | HOLD is released, RDY goes low, HALT instruction is executed as a do-nothing instruction. |

**(d) Timing Characteristics for HALT Instruction**




| POINTS | DESCRIPTION |
|---|---|
| A | SMDC instruction is placed on I lines. |
| B | At second RDY pulse second word of SMDC instruction is placed on I lines. This word is the new MDC. (1–8). |
| C | Next instruction is placed on I lines. |

**(e) Timing Characteristics for SMDC Instruction**

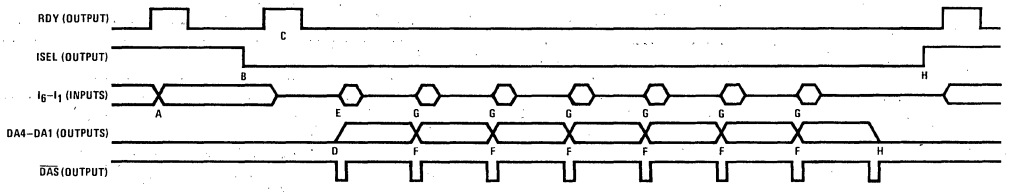



| POINTS | DESCRIPTION |
|---|---|
| A | AIN instruction is placed on I lines. |
| B | ISEL goes low to select data digit on I lines. $I_6$ and $I_4$–$I_1$ are don't care as long as $\overline{ADR}$ ($I_5$) = 1. |
| C | Digit address appears on DA lines. $\overline{DAS}$ provides 1 microcycle active low pulse which frames DA change. (Negative going edge occurs while DA lines are still 0. Positive going edge occurs after DA lines have changed to their new value). The first AIN instruction will have a digit address of 0000. Consecutive AIN instructions have digit addresses of 0001, 0010, etc., up to 0111. Instructions which terminate number entry reset the internal digit address to 0000. |
| D | Data digit is placed on D4–D1. $\overline{ADR}$ goes low indicating valid data digit. |
| E | F2 is pulsed active low to indicate read of data on D4–D1. $\overline{ADR}$ (= $I_5$) must be low for this pulse to occur. If $\overline{ADR}$ is high, the MM57109 will wait till it goes low before reading the I lines. $\overline{ADR}$ may go high again anytime after negative going edge of F2. I lines are don't care after F2 is pulsed. |
| F | DA lines are reset to 0. $\overline{DAS}$ provides a 1 microcycle pulse framing DA change. |
| G | ISEL goes high prior to RDY pulse for next instruction. |
| H | Next instruction appears on I lines. |

**(f) Timing Characteristics for AIN Instruction**

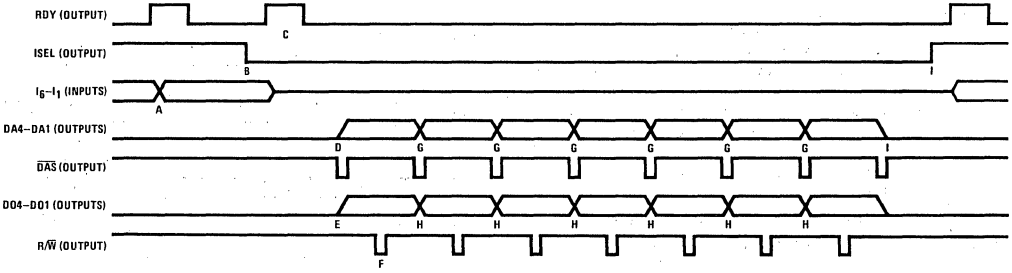
**FIGURE 10. Instruction Timing Diagrams**

## Functional Description (Continued)



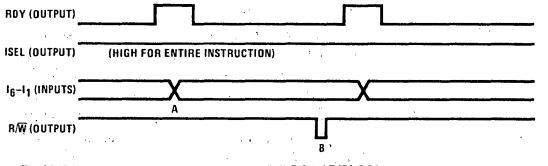

| POINTS | DESCRIPTION |
|---|---|
| A | IN instruction is placed on I lines. |
| B | ISEL goes low for second instruction word. Data digits multiplexed onto I lines when ISEL = 0. I lines are don't care until $\overline{DAS}$ pulse occurs. |
| C | Second word of instruction becomes available to external hardware as a high-order address for a RAM or other device. |
| D | First digit address appears on DA lines. This digit address is 0 or 2 depending on whether mode is scientific notation or floating point, respectively. Each time a new digit address appears, a 1 microcycle active low pulse on $\overline{DAS}$ frames this change so that at the negative going edge of $\overline{DAS}$ the old digit address is valid while at the positive going edge of $\overline{DAS}$ the new digit address is valid. |
| E | Digit data becomes valid on D4–D1 within 1/2 microcycle after $\overline{DAS}$ negative edge. Digit data must remain valid for 1 microcycle. During this time data is read. |
| F | Digit address advances to next digit, i.e., 0, 1, 2, 3, . . ., N scientific notation or 2, 3, 4, . . ., N floating point<br>where N = MDC + 3 scientific notation<br>N = MDC + 1 floating point<br>(See DATA FORMATS). |
| G | Next digit is placed on D4–D1, again within 1/2 microcycle after $\overline{DAS}$ negative edge. |
| H | All digits have been read in. Digit Address goes to 0000. $\overline{DAS}$ pulse occurs. ISEL goes high before RDY pulse for next instruction. Number of digits read depends on notation mode and mantissa digit count (see DATA FORMATS). |

**(g) Timing Characteristics for IN Instruction**



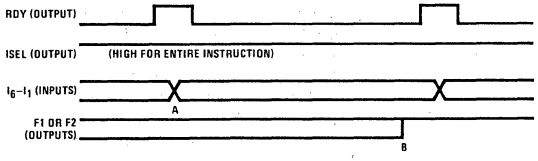

| POINTS | DESCRIPTION |
|---|---|
| A | OUT instruction is placed on I lines. |
| B | ISEL goes low for second instruction word. |
| C | Second word of instruction becomes available to external hardware as a high order address for a RAM or other device. |
| D | First digit address appears on DA lines, the same as for the IN instruction. $\overline{DAS}$ frames DA changes, as with IN instruction. |
| E | First digit output appears on DO4–DO1 within 2 microcycles after the negative edge of $\overline{DAS}$. |
| F | R/$\overline{W}$ active low pulse occurs to write data into RAM or other device. This is a 2-microcycle pulse occurring within 3 microcycles after the negative edge of $\overline{DAS}$. |
| G | Digit address advances to next digit. (See DATA FORMATS). |
| H | Next digit output appears on DO4–DO1, 14 microcycles after the last digit appeared. |
| I | All digits have been output. Digit Address goes to 0000. ISEL goes high before RDY pulse for next instruction. Number of digits read depends on notation mode and mantissa digit count (see DATA FORMATS). |

**(h) Timing Characteristics for OUT Instruction**




| POINTS | DESCRIPTION |
|---|---|
| A | PRW1 or PRW2 instruction is placed on I lines |
| B | 2-microcycle R/$\overline{W}$ active low pulse occurs. |

**(i) Timing Characteristics for PRW1, PRW2 Instructions**

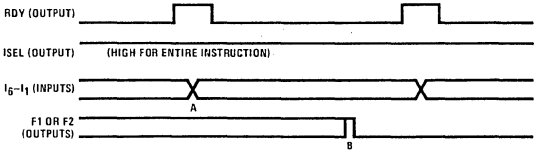



| POINTS | DESCRIPTION |
|---|---|
| A | SF1 or SF2 instruction is placed on I lines |
| B | F1 or F2 is set high. |

**(j) Timing Characteristics for SF1, SF2 Instructions**

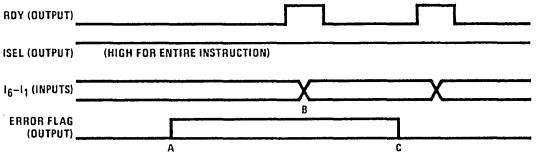
**FIGURE 10. Instruction Timing Diagrams (Continued)**

## Functional Description (Continued)




| POINTS | DESCRIPTION |
|---|---|
| A | PF1 or PF2 instruction is placed on I lines |
| B | F1 or F2 is pulsed active high for 2 microcycles. |

(k) Timing Characteristics for PF1, PF2 Instructions




| POINTS | DESCRIPTION |
|---|---|
| A | If an error occurs, ERROR is set high (see Table VI). |
| B | ECLR instruction is placed on I lines. |
| C | Error flag is set low. |

(l) Timing Characteristics for ECLR Instruction and Error Occurrences

FIGURE 10. Instruction Timing Diagrams (Continued)

### DATA FORMATS

#### IN/OUT Instructions

Mantissa digit count and notation mode determine data format. Table VII shows the contents of the D, D0 and DA lines for an IN or OUT instruction. Anywhere from 4 to 11 digits will be input or output by a single instruction.

#### AIN Instruction

One digit is input per AIN instruction. A maximum of 8 digits may be entered into the X mantissa by using consecutive AIN instructions. Digit entry is terminated by EN or any function instruction, (see NUMBER ENTRY). Table VII shows the DA lines for consecutive AIN instructions.

### ERROR CONTROL

The error flag, which can drive an LED indicator, is set high upon detection of an arithmetic or output error. (See Table VI).

TABLE VI. ERROR CONDITIONS

Error flag is set when:

1. LN X when $X \leq 0$
   LOG X when $X \leq 0$
2. Any result $< 10^{-99}$
   Any result $\geq 10^{100}$
3. TAN 90°, 270°, 450°, etc.
4. SIN X, COS X, TAN X when $|X| \geq 9000°$
5. $SIN^{-1}$ X, $COS^{-1}$ X when $|X| > 1$ or $|X| \leq 10^{-50}$
6. SQRT X when $X < 0$
7. /, INV/, 1/X when X = 0
8. In floating point mode OUT instruction if number of mantissa digits to left of decimal point is greater than the mantissa digit count.

The error flag can be tested by the TERR instruction (which branches if ERROR = 1) or it can be used to clear the external program counter, resulting in a hardware jump to location 0, the error recovery location. In either case, an ECLR instruction must be executed to clear the error flag.

The occurrence of an error does not affect the operation of the processor in any way. The OUT instruction will not output digits if error condition 8 is true, but otherwise will output digits, even if the error flag is set.

For automatic error recovery, ERROR is wired to the asynchronous clear input of the external program counter (PC). The instruction at location 0 is an ECLR to clear ERROR so that the next RDY pulse will advance the PC to location 1. A JMP instruction at location 1 with the branch address at location 2 of an error routine is then executed, which results in a transfer of program control to the error routine. These first 3 error recovery locations are skipped over upon reset (POR) as can be seen in the initialization and instruction fetch flowcharts. The program shown in Table VIII shows typical error recovery coding.

### SAMPLE SYSTEMS

*Figures 11–14* show sample systems using the MM57109. *Figure 11* shows a simple demonstrator system using switches to enter instructions. An LED display is used to demonstrate the OUT instruction, with a switch to force an OUT instruction on the I lines and to hold the HOLD input low for 1 second for repeated execution of the OUT instruction, resulting in a multiplexed display. A flip-flop latches the $\overline{BR}$ pulse which occurs when a test and branch instruction is true. LED lamps provide visual indication of the various flags. An enter button allows single instruction words to be entered one at a time in the ENTER mode and causes the display to light for 1 second in the DISPLAY mode.

*Figure 12* shows a stand-alone system with external program counter and a RAM to expand memory.

*Figure 13* shows the MM57109 used as a microprocessor peripheral. Latches contain instructions for the MM57109 and digit data for the microprocessor.

*Figure 14* shows a data acquisition system which obtains data from a 3-digit A/D converter. *Figure 14(b)* shows a program which reads data from the A/D converter. This coding should be studied as a general example of an MM57109 program.

*Figure 15* shows a microprocessor to MM57109 interface using 2 FIFO's for instruction and data buffering.

These sample systems are not intended to be detailed drawings of a complete system (except *Figure 11*). Their purpose is to provide the designer with some ideas as to how to use the MM57109 in an actual system.

## Functional Description (Continued)

### TABLE VII. DATA FORMATS

#### IN/OUT INSTRUCTIONS (A) MODE = SCIENTIFIC NOTATION

| DA4–DA1 | IN: / OUT: | D4 / DO4 | D3 / DO3 | D2 / DO2 | D1 / DO1 |
|---|---|---|---|---|---|
| 0 | | Most significant exponent digit | | | |
| 1 | | Least significant exponent digit | | | |
| 2 | | Sm | 0 | 0 | Se |
| 3 | | Not used | | | |
| 4 | | Most significant mantissa digit (Decimal point follows this digit) This digit *must* be non-zero on the AIN instruction unless the entire number is zero. Failure to do this will result in errors in calculations. This digit will *always* be non-zero on the OUT instruction, for non zero numbers. | | | |
| 5 | | Second most significant mantissa digit. | | | |
| . | | . | | | |
| . | | . | | | |
| . | | . | | | |
| MDC + 3 | | Least significant mantissa digit | | | |

#### IN/OUT INSTRUCTIONS (B) MODE = FLOATING POINT

| DA4–DA1 | DPX | IN: / OUT: | D4 / DO4 | D3 / DO3 | D2 / DO2 | D1 / DO1 |
|---|---|---|---|---|---|---|
| 2 | | | Sm | 0 | 0 | 0 |
| 3 | | | DP POS | | | |
| 4 | 11 | | Most significant mantissa digit = 0–9. On the OUT instruction, this digit will be non-zero unless $\|X\| < 1$, in which case it will be zero and DP POS will be 11. Leading zeroes are not blanked. | | | |
| 5 | 10 | | Second most significant mantissa digit | | | |
| . | . | | . | | | |
| . | . | | . | | | |
| . | . | | . | | | |
| MDC + 3 | 12 − MDC | | Least significant mantissa digit = 0–9 | | | |

**Notes:**

MDC = Mantissa digit count, set by SMDC instruction, initially = 8
Sm = Sign of mantissa, 0 = positive, 1 = negative
Se = Sign of exponent (Se = 0 in floating point mode)
DP POS = Decimal point position indicator is a value in the range from 11 down to 12 − MDC, which indicates a digit, as given by the DPX column in the table. The decimal point is located immediately following this digit. Example: If DP POS = 10, then the decimal point follows the second most significant mantissa digit (DPX = 10).

#### AIN INSTRUCTION

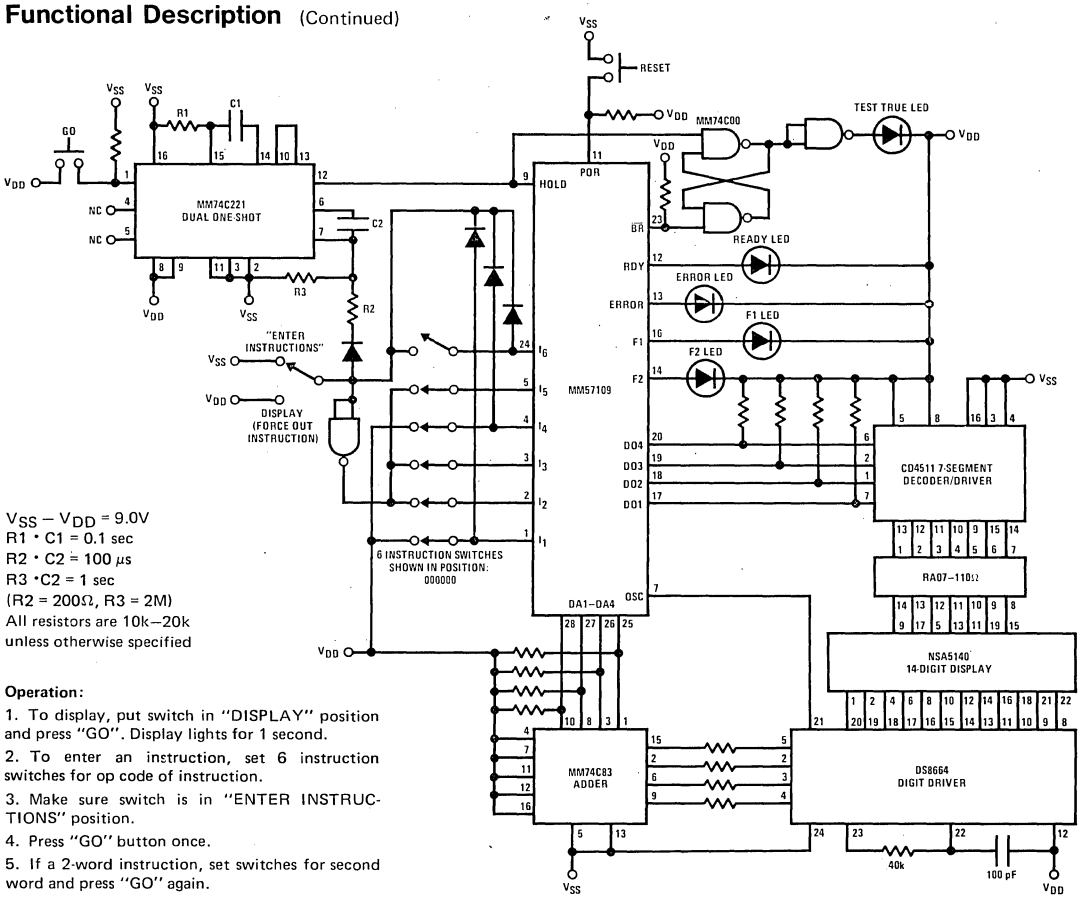
| DA4–DA1 | |
|---|---|
| 0 | Most significant digit Xm (first AIN instruction) |
| . | . |
| . | . |
| . | . |
| 7 | Least significant digit Xm (eighth AIN instruction) |

Note. $X_m$ = X register mantissa. Decimal point follows last digit entered. An irrecoverable internal error will occur if more than 8 digits are entered in a row with AIN, or if a non-BCD digit is entered.

### TABLE VIII. ERROR RECOVERY CODING

| OCTAL ADDRESS | OCTAL OP CODE | LABEL | INSTRUCTION MNEMONIC | OPERAND | COMMENT |
|---|---|---|---|---|---|
| 00 | 53 | | ECLR | | Clear error flag |
| 01 | 25 | | JMP | ERROR | Jump to error routine |
| 02 | 75 | | | | Address of label 'ERROR' |
| 03 | . | | User Program | | |
| . | . | | . | | |
| . | . | | . | | |
| . | . | | . | | |
| . | . | | . | | |
| . | . | | . | | |
| 75 | . | ERROR | — | — | User error recovery routine |
| . | . | | . | . | . |
| . | . | | . | . | . |

## Functional Description (Continued)




$V_{SS} - V_{DD} = 9.0V$
$R1 \cdot C1 = 0.1$ sec
$R2 \cdot C2 \doteq 100\ \mu s$
$R3 \cdot C2 = 1$ sec
(R2 = 200Ω, R3 = 2M)
All resistors are 10k–20k unless otherwise specified

**Operation:**

1. To display, put switch in "DISPLAY" position and press "GO". Display lights for 1 second.
2. To enter an instruction, set 6 instruction switches for op code of instruction.
3. Make sure switch is in "ENTER INSTRUCTIONS" position.
4. Press "GO" button once.
5. If a 2-word instruction, set switches for second word and press "GO" again.

FIGURE 11. MM57109 Sample System with Switches for Instruction Entry

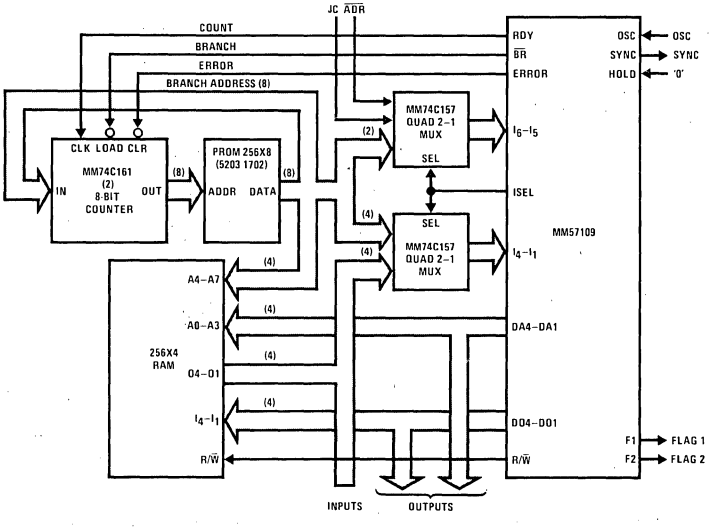



FIGURE 12. MM57109 Stand-Alone System

## Functional Description (Continued)

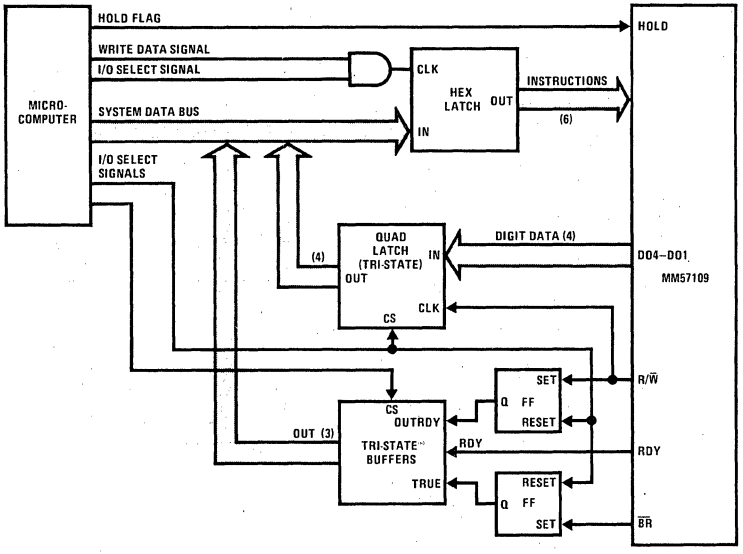



FIGURE 13(a). MM57109 as a Microcomputer Peripheral

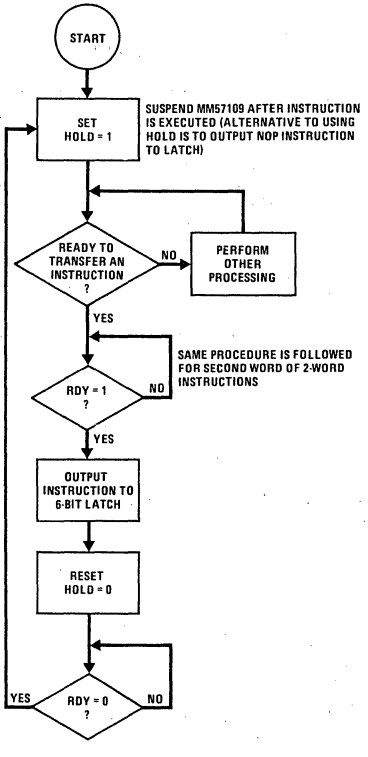



FIGURE 13(b). Microcomputer Software for MM57109 Peripheral Interface

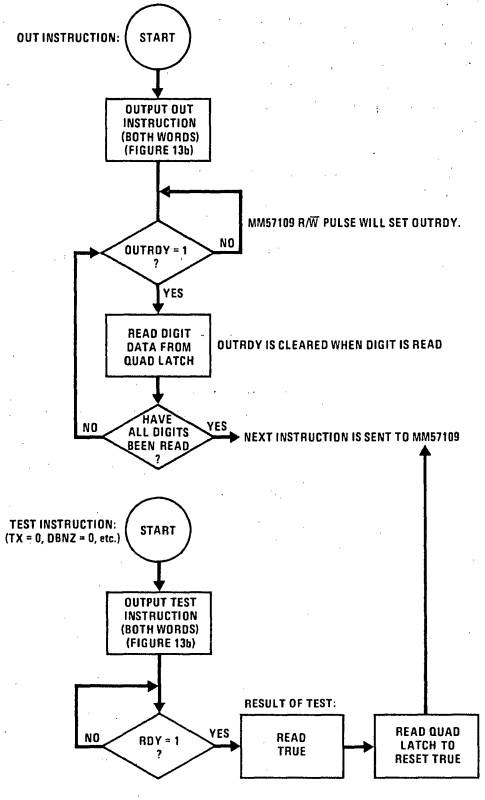



FIGURE 13(c). Microcomputer Software for MM57109 OUT, Test Instructions

## Functional Description (Continued)

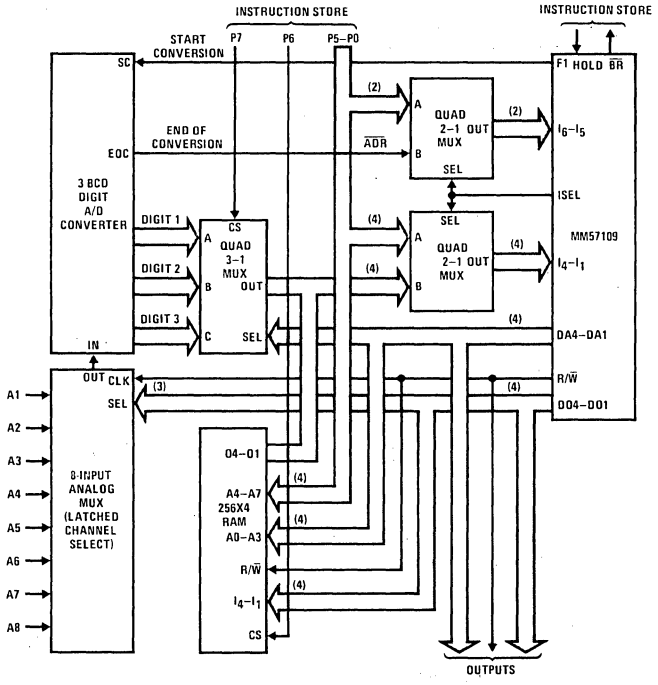



FIGURE 14(a). MM57109 Analog Data Acquisition System Block Diagram

### Acquisition System Instruction Format

| TYPE OF INSTRUCTION | $P_7$ | $P_6$ | $P_5P_4P_3P_2P_1P_0$ |
|---|---|---|---|
| Select Analog Channel | 1 | 1 | OUT instruction |
| A/D Input | 0 | 1 | AIN instruction |
| RAM I/O | 1 | 0 | IN/OUT instructions for second word. Second word $P_0$–$P_3$ are high order RAM addresses |
| Others | 1 | 1 | Other instructions |

### Acquisition System Coding

| | $P_7$ | $P_6$ | $P_5$–$P_0$ | COMMENT |
|---|---|---|---|---|
| | 1 | 1 | 4 | Number of channels to be input |
| | 1 | 1 | MS | Store in M |
| | 1 | 1 | SMDC | Mantissa Digit Count = 1 |
| | 1 | 1 | 1 | |
| LOOP | 1 | 1 | MR | Retrieve channel number |
| | 1 | 1 | OUT | Select analog channel |
| | 1 | 1 | 0 | |
| | 1 | 1 | PF1 | Start A/D converter |
| | 1 | 1 | EN | Push stack |
| | 0 | 1 | AIN | Read A/D converter digit 1 when EOC = 0 |
| | 0 | 1 | AIN | Read A/D converter digit 2 |
| | 0 | 1 | AIN | Read A/D converter digit 3 |
| | 1 | 1 | DBNZ | Update channel number and check if 0 |
| | 1 | 1 | LOOP | |
| | 1 | 1 | X | Channel 1 times channel 2 (C1 x C2) |
| | 1 | 1 | / | (C1 x C2) ÷ C3 |
| | 1 | 1 | COS | COSINE ((C1 x C2) ÷ C3) |
| | 1 | 1 | + | C4 + COSINE ((C1 x C2) ÷ C3) |
| | 1 | 1 | SMDC | Mantissa Digit Count = 3 |
| | 1 | 1 | 3 | |
| | 1 | 0 | OUT | Result to RAM (0) |
| | 1 | 0 | 0 | |

FIGURE 14(b). MM57109 Analog Data Acquisition System Input Coding

## Functional Description (Continued)

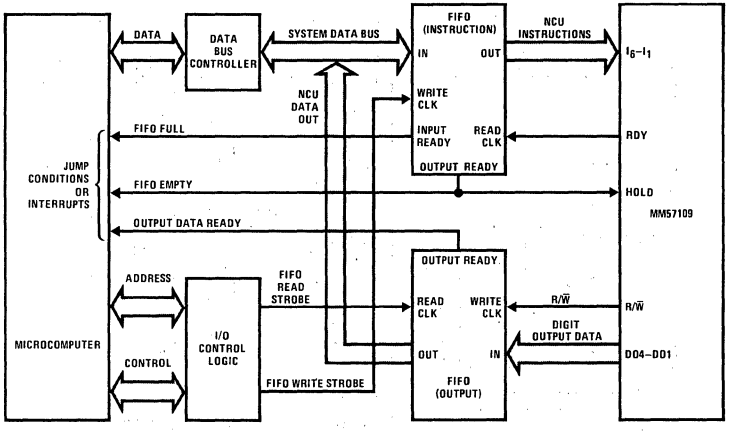



FIGURE 15. MM57109 Microcomputer Interface Using 2 FIFO's

### Getting Your MM57109 Going

After wiring up a system using an MM57109, the following steps should be followed to verify that the processor is operating properly:

1. Check power supply for proper level, polarity, and absence of noise.
2. Check oscillator frequency, levels, duty cycle and rise and fall times.
3. Verify presence of SYNC output.
4. Check POR reset pulse duration, levels and rise and fall times.
5. Put HOLD input high and verify that RDY stays high.
6. Put HOLD input low and verify that RDY is pulsing active high.
7. Check that the system places the proper instructions on the I lines.
8. Force an OUT instruction on the I lines, put HOLD low, apply a reset pulse, and verify that DO4, DO2, DO1 DA–DA1, $\overline{DAS}$ and R/$\overline{W}$ are changing.

National Semiconductor

Electronic Data Processing

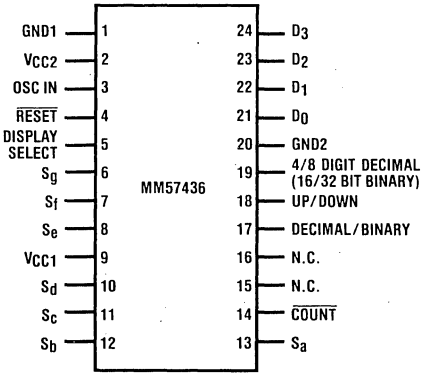
# MM57436 Decimal/Binary Up/Down Counter

## General Description

The MM57436 Counter, an NMOS silicon gate technology device, is designed to be a minimal solution Decimal/Binary Up/Down counter with display capability. The counter length is user selectable at 4 digits decimal (16 bits binary) or 8 digits decimal (32 bits binary). The device has the capability of direct drive of a 4 digit multiplexed LED display. In the 8-digit (32-bit) mode, the user may direct either the top four digits or lower four digits to the display. The MM57436 will run off an internal RC oscillator or the user may supply an external oscillator for greater precision in the count rate.

## Features

- Decimal or binary count
- Up or down count
- 4 or 8 digit (16 or 32 bit) counter length
- 4 digit, seven segment multiplexed LED display drive
- User display control
- Single supply operation
- Wide supply range (4.5V-9.5V)
- TTL compatible on inputs




Top View
Order Number MM57436N
See Package 22

| Pin | Description |
|---|---|
| OSC IN | Oscillator Input — External Oscillator or RC |
| Display Select | Control line to display upper or lower 4 digits (16 bits) of 8-digit (32-bit) counter |
| $S_A$-$S_G$ | Multiplexed 7-segment outputs |
| $\overline{\text{COUNT}}$ | Input for signal to be counted |
| Decimal/ Binary | Counter mode control |
| Up/Down | Up-down count control |
| 4/8 Digit (16/32 Bit Binary) | Counter length control |
| $D_0$-$D_3$ | Display digit strobes |
| $V_{CC1}$, $V_{CC2}$ | Power supply |
| GND1, GND2 | Ground |

**Figure 1. Connection Diagram**

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin Relative to $GND_1$ | −0.3V to +10V |
| Ambient Operating Temperature | 0°C to +70°C |
| Ambient Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 Seconds) | 300°C |
| Power Dissipation | 0.75 Watt at 25°C<br>0.4 Watt at 70°C |

*"Absolute Maximum Ratings" indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.*

## DC Electrical Characteristics $0°C \leq T_A \leq 70°C$, $4.5V \leq V_{CC} \leq 9.5V$, unless otherwise specified

| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| Operating Voltage ($V_{CC}$) | | 4.5 | | 9.5 | V |
| Operating Supply Current | $V_{CC} = 5V$, $T_A = +25°C$ (all inputs and outputs open) | | | 5.0 | mA |
| Input Voltage Levels | | | | | |
| OSC IN, $\overline{RESET}$ Levels | | | | | |
| Logic High ($V_{IH}$) | | $0.7V_{CC}$ | | | V |
| Logic Low ($V_{IL}$) | | | | 0.6 | V |
| $\overline{RESET}$ Hysteresis | | 1.0 | | | V |
| All Other Inputs | | | | | |
| Logic High ($V_{IH}$) | $V_{CC} = 9.5V$ | 3.0 | | | V |
| Logic High ($V_{IH}$) | $V_{CC} = 5V \pm 10\%$ | 2.0 | | | V |
| Logic Low ($V_{IL}$) | | | | 0.8 | V |
| Output Current Levels | | | | | |
| Output Sink Current | | | | | |
| $D_0$-$D_3$ ($I_{OL}$) | $V_{CC} = 9.5V$, $V_{OL} = 1.0V$ | 30 | | 150 | mA |
| | $V_{CC} = 4.5V$, $V_{OL} = 1.0V$ | 15 | | 70 | mA |
| $S_A$-$S_G$ ($I_{OL}$) | $V_{CC} = 9.5V$, $V_{OL} = 1.0V$ | 2.0 | | 9.0 | mA |
| | $V_{CC} = 4.5V$, $V_{OL} = 1.0V$ | 1.0 | | 4.5 | mA |
| Output Source Current | | | | | |
| $S_A$-$S_G$ ($I_{OH}$) | $V_{CC} = 9.5V$, $V_{OH} = 2.0V$ | −3.0 | | −30 | mA |
| | $V_{CC} = 6.0V$, $V_{OH} = 2.0V$ | −3.0 | | −20 | mA |

## AC Electrical Characteristics $0°C \leq T_A \leq 70°C$, $4.5V \leq V_{CC} \leq 9.5V$, unless otherwise specified

| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| OSC IN | | | | | |
| Frequency | | 100 | | 266.67 | kHz |
| Duty Cycle | | 40 | | 60 | % |
| Rise Time | | | | 1 | μs |
| Fall Time | | | | 1 | μs |
| Internal Time Base | | | | | |
| (= 4/Frequency) | | 15 | | 40 | μs |
| OSC IN Using RC | $R = 56k\Omega \pm 5\%$, | | | | |
| Frequency | $C = 100pF \pm 10\%$ | 140 | | 266.67 | kHz |
| Internal Time Base | | | | | |
| (= 4/Frequency) | | 15 | | 28 | μs |
| Inputs | | | | | |
| Up/Down, Display Select | | | | | |
| $t_{SETUP}$ | | | | 8 | μs |
| $t_{HOLD}$ | | | | 1 | μs |
| $\overline{Count}$ | | | | | |
| $t_{SETUP}$ | | | | 2 | μs |
| $t_{HOLD}$ | | | | 1 | μs |

## AC Electrical Characteristics (continued) $0\,°C \leq T_A \leq 70\,°C$, $4.5V \leq V_{CC} \leq 9.5V$, unless otherwise specified

| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| $\overline{\text{Count}}$ Input Frequency | 4 Digit Decimal Up Count | | | | |
| | OSC IN = 266.67 kHz | | | 14.4 | kHz |
| | OSC IN = 100 kHz | | | 5.43 | kHz |
| | 4 Digit Decimal Down Count | | | | |
| | OSC IN = 266.67 kHz | | | 13.6 | kHz |
| | OSC IN = 100 kHz | | | 5.13 | kHz |
| | 8 Digit Decimal Up Count | | | | |
| | OSC IN = 266.67 kHz | | | 9.52 | kHz |
| | OSC IN = 100 kHz | | | 3.57 | kHz |
| | 8 Digit Decimal Down Count | | | | |
| | OSC IN = 266.67 kHz | | | 9.17 | kHz |
| | OSC IN = 100 kHz | | | 3.44 | kHz |
| | 16 Bit Binary Up Count | | | | |
| | OSC IN = 266.67 kHz | | | 16.3 | kHz |
| | OSC IN = 100 kHz | | | 6.14 | kHz |
| | 16 Bit Binary Down Count | | | | |
| | OSC IN = 266.67 kHz | | | 15.3 | kHz |
| | OSC IN = 100 kHz | | | 5.76 | kHz |
| | 32 Bit Binary Up Count | | | | |
| | OSC IN = 266.67 kHz | | | 11.2 | kHz |
| | OSC IN = 100 kHz | | | 4.21 | kHz |
| | 32 Bit Binary Down Count | | | | |
| | OSC IN = 266.67 kHz | | | 10.3 | kHz |
| | OSC IN = 100 kHz | | | 3.86 | kHz |
| Pulse Width (= 8/OSC IN Frequency) | OSC IN = 100kHz | 80 | | | μs |
| | OSC IN = 266.67 kHz | 30 | | | μs |
| $\overline{\text{RESET}}$ Input Pulse Width | Resetting device while device running | | | | |
| | OSC IN = 100 kHz | 160 | | | μs |
| | OSC IN = 266.67 kHz | 60 | | | μs |

## Functional Description

The MM57436 will count pulses at its count input and will display 4 digits of the resultant count. Under user control the device will count in either decimal or binary and will either count up or count down. The user may also select which group of 4 digits (16 bits) is to be displayed.

The display is standard, seven-segment for the decimal counter. In the binary mode, hex characters are displayed as follows:

0-9, A, b, C, d, E, F

The mode controls of the MM57436 are as follows:

**Decimal/Binary —** With this pin left open or tied to $V_{CC}$, the MM57436 is a decimal counter. Connecting this pin to output D1 converts the MM57436 to a binary counter. This mode is a strap option and may *not* be changed while the device is running.

**4/8-Digit Decimal (16/32-Bit Binary) —** With this pin left open or tied to $V_{CC}$ the MM57436 is a 4-digit decimal or 16-bit binary counter. Connecting this pin to ground converts the MM57436 to an 8-digit decimal or 32 bit binary counter. The counter length is a strap option and may *not* be changed while the device is running.

**Up/Down —** With this pin left open or at a logic "1" (positive logic) the MM57436 will increment its internal counter by 1 with every pulse input at the $\overline{\text{COUNT}}$ input. With this pin connected to ground or to a logic "0" (positive logic), the MM57436 will *decrement* its internal counter by 1 with every pulse at the $\overline{\text{COUNT}}$ input. This input may be tied high or low, may come from a switch or may be controlled by a logic signal. It may be changed by the user at any time. Note, if this input is to be controlled by a mechanical switch some external debounce protection may be required depending on the application. There is no debounce protection internally on this input.

**Display Select —** With this input tied to $V_{CC}$ or at a logic "1", the MM57436 will display the upper 4 digits (16 bits) of the 8 digit (32 bit) counter. Connecting this pin to ground or to a logic "0" will cause the lower 4 digits of the 8 digit counter to be displayed. If the MM57436 is operating as a 4-digit counter (pin 19 open or at $V_{CC}$) the Display Select input is ignored and has no effect whatsoever on the display. This input may be hard wired to either $V_{CC}$ or ground; may be controlled by a switch or may be controlled by a logic signal. The input may be changed at any time by the user without impairing the operation of the device.

# General Operation

## Initialization

The $\overline{\text{RESET}}$ logic will clear the MM57436 if the power supply rise time is between 1 ms and 1 $\mu$s. If the power supply rise time is greater than 1 ms, the user must provide an external RC network and diode to the $\overline{\text{RESET}}$ pin as shown below (Figure 2). The $\overline{\text{RESET}}$ input is configured as a Schmitt trigger input. The user may control this with an external signal if desired as long as the proper levels are maintained. The $\overline{\text{RESET}}$ pin is the means by which the user may clear the counter. $\overline{\text{RESET}}$ may be brought low at any time. The MM57436 will be cleared whenever the proper "0" level is applied at the $\overline{\text{RESET}}$ input provided the input stays low for at least 16 clock cycles. If the reset pin is not used it should be connected to $V_{CC}$.

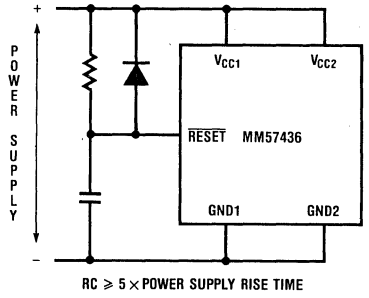



**Figure 2. Power-Up Clear Circuit**

## Oscillator

The user has the option of connecting an RC network to the OSC IN pin and using the internal oscillator or he may supply an external oscillator to the OSC IN pin. The OSC IN input is a Schmitt trigger input and the user must insure that the proper levels are met when supplying an external clock.

The external oscillator is recommended when the counting speed and/or the stability of the counting speed is critical. The internal RC oscillator is only accurate to about ±15% to ±20%. However, if practical in the application, the RC network can be tuned for the desired operating frequency. Some typical RC values that place the operating speed at near the maximum are shown below (Figure 3).

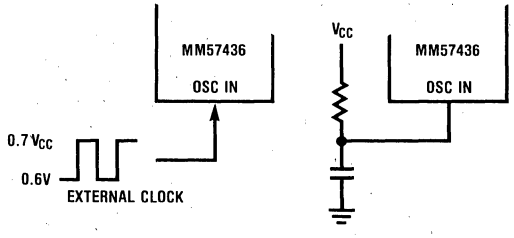
## Power Supply

The MM57436 has two $V_{CC}$ pins: $V_{CC1}$ and $V_{CC2}$ — and two ground pins: GND1 and GND2. Both $V_{CC1}$ and $V_{CC2}$ *must* be connected to the positive supply ($V_{CC}$). Both GND1 and GND2 *must* be connected to ground. Failure to do this will result in improper operation of the MM57436.

## $\overline{\text{Count}}$ Input

The MM57436 counts negative-going pulses at the $\overline{\text{Count}}$ Input. The width of the negative-going (logic "1" to logic "0") must be at least 8 times the oscillator cycle time.

In order to maximize the counting speed and not to miss any pulses, during the display cycles, the MM57436 has a 4-bit register at the $\overline{\text{COUNT}}$ input which will accumulate up to 15 counts. This register is added/subtracted from the counter. Therefore at the higher input count speeds, when the counter is changed from an up counter to a down counter or vice versa, there is a window of up to 15 counts — the maximum value in the input register — in the count. This effect is completely unobservable at slow input count speeds and gradually becomes more noticeable as the repetition rate of the count pulse increases. If the up/down mode is not changed during operation, the only observable effect of the input register is that the display may appear to increment or decrement by values greater than 1.




| RC Controlled Oscillator | | |
|---|---|---|
| R(kΩ) | C(pF) | OSC IN Period (μs) |
| 51 | 100 | 4.75 ± 15% |
| 82 | 56 | 4.75 ± 13% |

**Figure 3. MM57436 Oscillator**

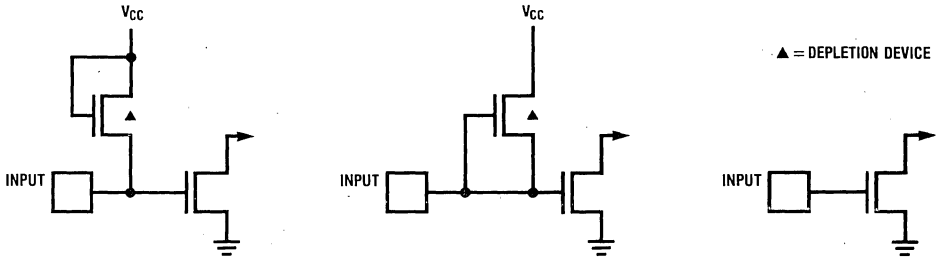
## Input/Output Characteristics

### Inputs

The MM57436 has three types of inputs. Figure 4a is the input with a depletion load to $V_{CC}$ found on pins 17, 18, and 19 (Decimal/Binary, Up/Down, 4/8 Digit). Figure 4b is a slightly different type of input with a depletion load to $V_{CC}$ found on pins 4 and 14 ($\overline{RESET}$, $\overline{COUNT}$). The remaining input, pin 5-Display Select, has no load device (Figure 4c).

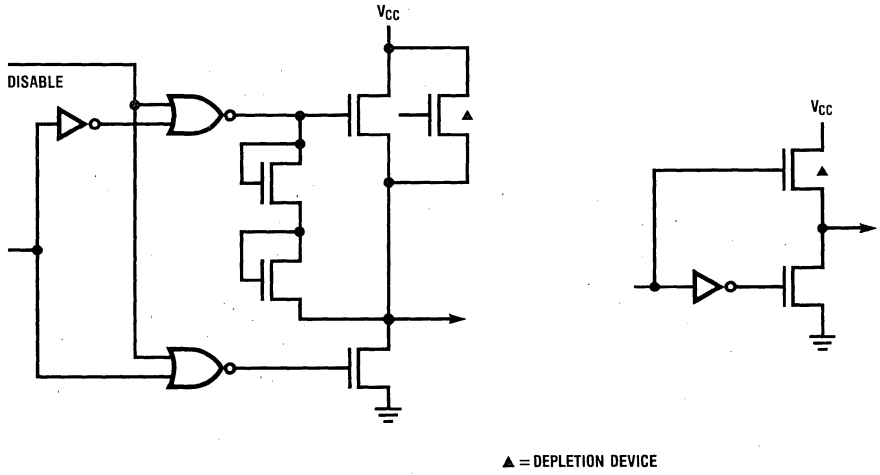
### Outputs

There are only two types of outputs on the MM57436: the segment drivers (Figure 5a) and the digit drivers (Figure 5b).




a. Pins 17, 18, 19 b. Pins 4, 14 c. Pin 5

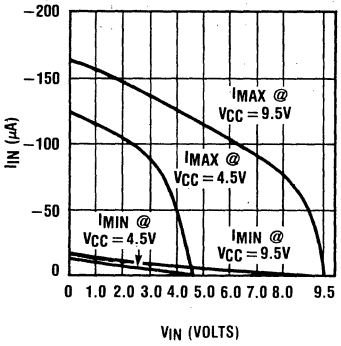
Figure 4. Input Configurations



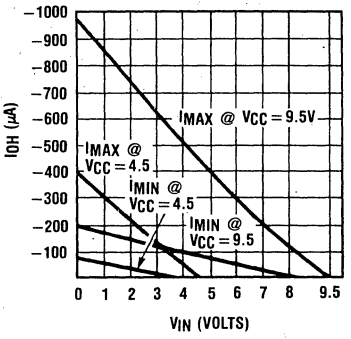

a. Segment Driver Outputs b. Digit Driver Outputs

Figure 5. Output Configurations

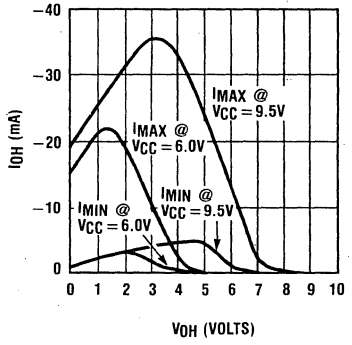
Input Current $\overline{\text{RESET}}$, Count



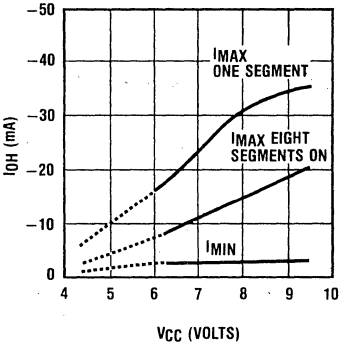

Input Current Decimal/ Binary, Up/Down, 4/8 Digit



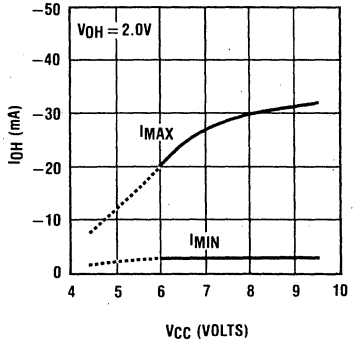

$S_a$-$S_g$ LED Output Source Current




LED Output Direct Segment and Digit Drive




LED Output Direct Segment Drive




Output Sink Current for $D_0$-$D_3$

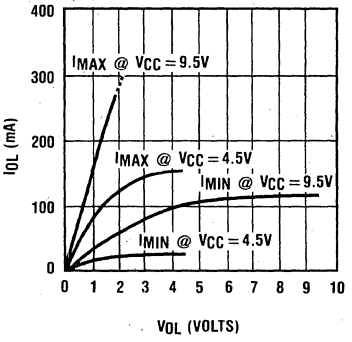



**Figure 6. I/O DC Current Characteristics**

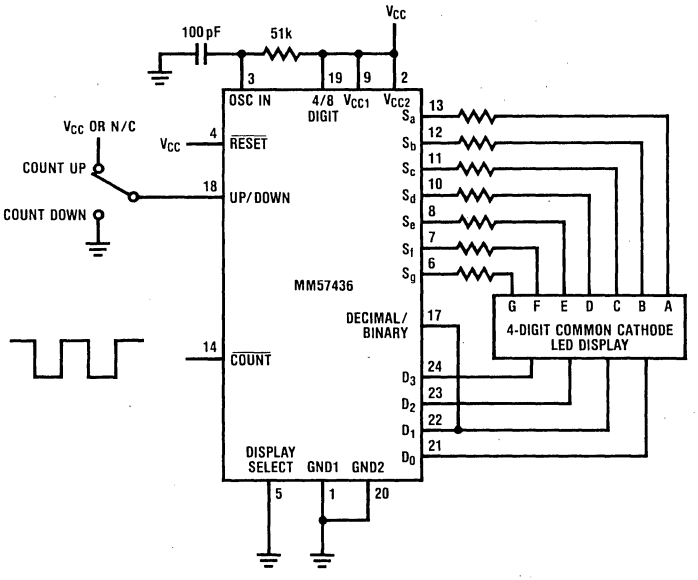



Figure 7. MM57436 as 16-Bit Binary Counter with RC Oscillator and Switch-Controlled Up/Down Mode

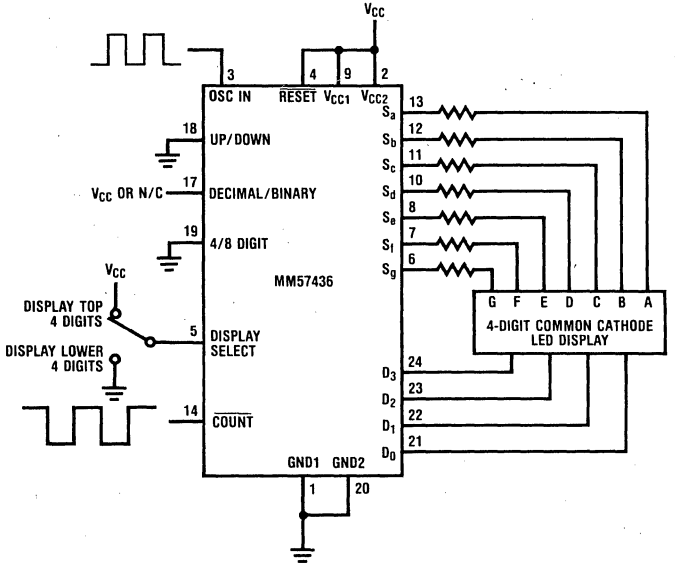



Figure 8. MM57436 as 8-Digit Decimal Down Counter with Extenal Oscillator

National Semiconductor

Electronic Data Processing

# MM57499 96 or 144-Key Serial Keyboard Interface (SKI)

## General Description

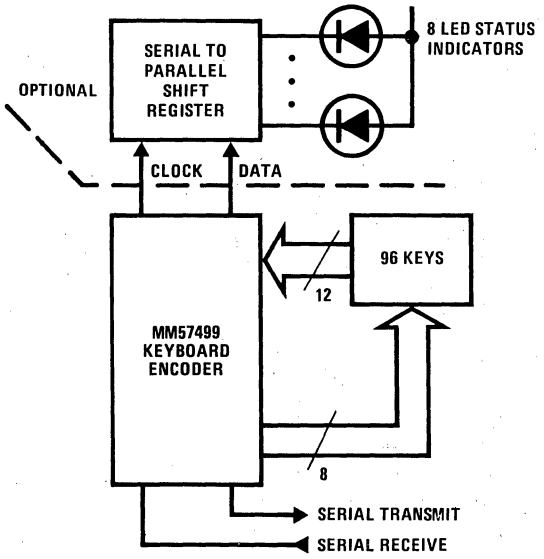
The MM57499 keyboard interface, an NMOS silicon gate technology device, is designed to be a minimum IC solution for the purpose of interfacing detached keyboards to terminals. It can reduce the usual 18 to 24-wire keyboard to terminal interconnection to a 5-wire connection.

The 96-key operation is a simple direct interface to a 12 × 8 matrix keyboard. The additional capability of a 144-key option can be obtained by implementing an inexpensive 4 to 12-line decoder IC between the MM57499 and a 12 × 12 matrix keyboard. If fewer than 96 or 144 keys are used, no connection is required in the matrix at the unused key locations.

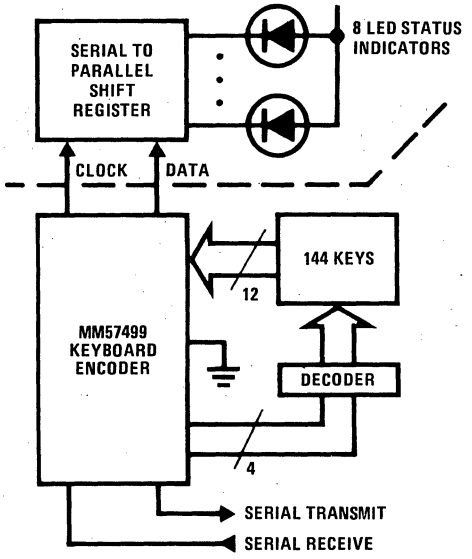
## Features

- Full upper and lower case ASCII codes, numeric pad & function encoding on-chip
- On-chip oscillator utilizes the standard 3.58 MHz color burst crystal
- On-chip baud rate generator
- Serial transmit and receive
- 400 WPM burst rate (typical)
- 2-key lockout
- Auto repeat on all keys
- Manual repeat key
- Programmable phrase storage
- Shift, cap loc, control, modes
- 144-key strap option
- Status information for up to 8 indicators
- Single 5V supply
- 2.5 kΩ maximum ON resistance
- TTL compatible
- 28-pin dual-in-line package

## Basic Application




96-Key Operation




144-Key Operation

## Absolute Maximum Ratings (Note 1)

| | |
|---|---|
| Voltage at Any Pin Relative to GND | −0.5V to +7V |
| Ambient Operating Temperature (Note 1) | 0°C to +70°C |
| Ambient Storage Temperature | −65°C to +150°C |
| Power Dissipation | 0.75 W at 25°C<br>0.4 W at 70°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## DC Electrical Characteristics $0°C \le T_A \le +70°C$, $4.5V \le V_{CC} \le 6.3V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Operating Voltage ($V_{CC}$) | | 4.5 | 6.3 | V |
| Operating Supply Current | $V_{CC} = 5V$, $T_A = 25°C$<br>(all inputs and outputs open) | | 30 | mA |
| Input Voltage Levels | | | | |
| Crystal Input | | | | |
| Logic High ($V_{IH}$) | | 2.0 | | V |
| Logic Low ($V_{IL}$) | | | 0.4 | V |
| $\overline{RESET}$ Input Levels | | | | |
| Logic High | | 0.7 $V_{CC}$ | | V |
| Logic Low | | | 0.6 | V |
| $\overline{RESET}$ Hysteresis | | 1.0 | | V |
| All Other Inputs | | | | |
| Logic High | $V_{CC} = max$ | 3.0 | | V |
| Logic Low | | | 1.2 | V |
| Output Voltage Levels | | | | |
| Standard Output | | | | |
| TTL Operation | $V_{CC} = 5V \pm 5\%$ | | | |
| Logic High ($V_{OH}$) | $I_{OH} = 100\ \mu A$ | 2.4 | | V |
| Logic Low ($V_{OL}$) | $I_{OL} = -1.6$ mA | | 0.4 | V |
| CMOS Operation | | | | |
| Logic High ($V_{OH}$) | $I_{OH} = 10\ \mu A$ | $V_{CC} - 1$ | | V |
| Logic Low ($V_{OL}$) | $I_{OL} = -10\ \mu A$ | | 0.2 | V |

**Note 1:** Absolute maximum ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications are not ensured when operating the device at absolute maximum ratings.

## AC Electrical Characteristics $0°C \le T_A \le +70°C$, $4.5V \le V_{CC} \le 6.3V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| Cycle Time | | | 4.469 | μs |
| Input Frequency | | | 3.579 | MHz |
| Duty Cycle | | 30 | 55 | % |
| Outputs | | | | |
| MM57499 to CMOS Propagation Delay Clock Output | $4.5V \le V_{CC} \le 6.3V$, $C_L = 50$ pF,<br>$V_{OH} = 0.7\ V_{CC}$, $V_{OL} = 0.3\ V_{CC}$ | | | |
| $t_{PD1}$ | | | 1.1 | μs |
| $t_{PD0}$ | | | 0.3 | μs |
| Pin 17 Data Output | | | | |
| $t_{PD1}$ | | | 1.4 | μs |
| $t_{PD0}$ | | | 0.3 | μs |
| $t_{PD1}$ | $V_{OH} = 2V$ | | 0.7 | μs |

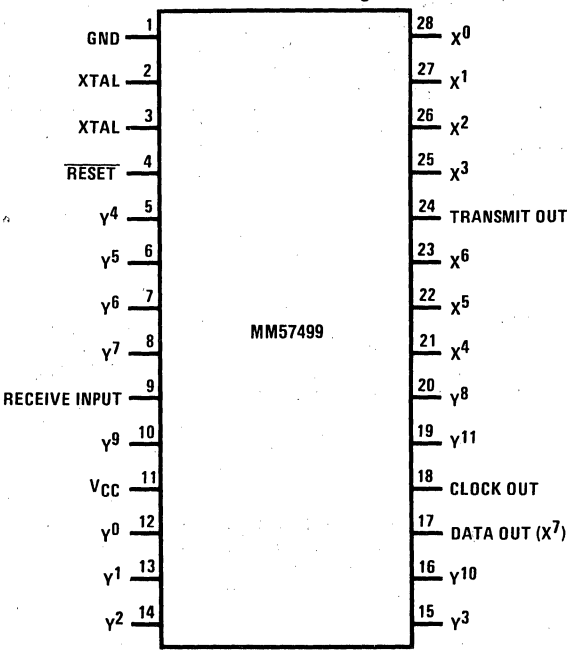
## AC Electrical Characteristics (Continued) $0°C \le T_A \le +70°C$, $4.5V \le V_{CC} \le 6.3V$ unless otherwise noted.

| Parameter | Conditions | Min | Max | Units |
|---|---|---|---|---|
| MM57499 to TTL Propagation Delay | Fanout = 1 Standard TTL Load<br>$V_{CC} = 5V \pm 5\%$, $C_L = 50$ pF,<br>$V_{OH} = 2.4V$, $V_{OL} = 0.4V$ | | | |
| Pin 18 Clock Output | | | | |
| $t_{PD1}$ | | | 0.8 | μs |
| $t_{PD0}$ | | | 0.8 | μs |
| Pin 17 Data Output | | | | |
| $t_{PD1}$ | | | 1.0 | μs |
| $t_{PD0}$ | | | 1.0 | μs |
| Row Outputs, Transmit Output | | | | |
| $t_{PD1}$ | | | 1.3 | μs |
| $t_{PD0}$ | | | 1.3 | μs |
| Key Cycle Timing | | | | |
| Down Debounce | | | | |
| $t_d$ | See *Figure 9* for Timing Sequence | 96-Key | 11.5 | ms |
| | | 144-Key | 14.4 | ms |
| Up Debounce | | | | |
| $t_u$ | | 96-Key | 7.5 | ms |
| | | 144-Key | 9.2 | ms |
| Transmit Time | | 96/144-Key | 9.1 | ms |
| Decode Time | | 96/144-Key | 0.3 | ms |
| $t_e$ | | | | |
| Burst Rates | | 96-Key | 423 | WPM† |
| | | 144-Key | 327 | WPM† |
| Auto Repeat Rate | | 96-Key | 15 | CPS |
| | | 144-Key | 12 | CPS |
| Manual Repeat Rate | | 96-Key | 66 | CPS |
| | | 144-Key | 61 | CPS |

† 5-character words

## Connection Diagram

Dual-In-Line Package




TOP VIEW
Order Number MM57499N
See Package 23

# Functional Description

## KEY SCAN

The MM57499 interfaces to a standard X-Y keyboard matrix. The strobe lines "walk" down the keyboard X matrix lines (or external decoder) and are detected on the Y inputs if a key is pressed. The sequential strobe/scan characterizes timing in many of the MM57499 functions. The key function matrix is shown in *Figure 1* and the complete code assignment is given in Table I.

Diode isolation is required in the key matrix to guarantee that if two keys and a control key are simultaneously pressed the MM57499 will process the correct key sequence. This maintains 2-key lockout and insures that an erroneous control, shift, or repeat key is not encountered.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| SHIFT KEY | CONTROL | REPEAT | CAP LOC | SHIFT LOC | Z z | Y y | X x | $Y^{11}$ |
| W w | V v | U u | T t | S s | R r | Q q | P p | $Y^{10}$ |
| O o | N n | M m | L l | K k | J j | I i | H h | $Y^{9}$ |
| G g | F f | E e | D d | C c | B b | A a | □ @ | $Y^{8}$ |
| ? / | > . | = − | < , | + ; | * : | ) 9 | ( 8 | $Y^{7}$ |
| ' 7 | & 6 | % 5 | $ 4 | # 3 | " 2 | ! 1 | 0 | $Y^{6}$ |
| BREAK | · | RTN | SP | ESC | LF | 9 | 8 | $Y^{5}$ |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | $Y^{4}$ |
| DEL — | ^ ~ | ] } | \ \| | [ { | BS | TAB | ← | $Y^{3}$ |
| → | ↓ | ↑ | FMT | IL | DC | DL | FS | $Y^{2}$ |
| EOL | EOS | CLEAR | SC | B TAB | DE | ADM | IC | $Y^{1}$ |
| LS | FN7 | FN6 | FN5 | FN4 | FN3 | FN2 | FN1 | $Y^{0}$ |
| $X^{7}$ | $X^{6}$ | $X^{5}$ | $X^{4}$ | $X^{3}$ | $X^{2}$ | $X^{1}$ | $X^{0}$ | |

**FIGURE 1. Key Function Matrix**

## Functional Description (Continued)

**TABLE I. CODE ASSIGNMENTS**

| X | Y | Code | Control | Control & Shift or Shift Loc | Shift | Shift Loc | Shift Loc & Cap Loc | Cap Loc | Key |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 80 | 80 | 80 | 80 | 80 | 80 | 80 | FN1 |
| 1 | 0 | 81 | 81 | 81 | 81 | 81 | 81 | 81 | FN2 |
| 2 | 0 | 82 | 82 | 82 | 82 | 82 | 82 | 82 | FN3 |
| 3 | 0 | 83 | 83 | 83 | 83 | 83 | 83 | 83 | FN4 |
| 4 | 0 | 84 | 84 | 84 | 84 | 84 | 84 | 84 | FN5 |
| 5 | 0 | 85 | 85 | 85 | 85 | 85 | 85 | 85 | FN6 |
| 6 | 0 | 86 | 86 | 86 | 86 | 86 | 86 | 86 | FN7 |
| 7 | 0 | 87 | 87 | 87 | 87 | 87 | 87 | 87 | LS |
| 0 | 1 | 88 | 88 | 88 | 88 | 88 | 88 | 88 | IC |
| 1 | 1 | 89 | 89 | 89 | 89 | 89 | 89 | 89 | ADM |
| 2 | 1 | 8A | 8A | 8A | 8A | 8A | 8A | 8A | DE |
| 3 | 1 | 8B | 8B | 8B | 8B | 8B | 8B | 8B | BTAB |
| 4 | 1 | 8C | 8C | 8C | 8C | 8C | 8C | 8C | SC |
| 5 | 1 | 8D | 8D | 8D | 8D | 8D | 8D | 8D | CLEAR |
| 6 | 1 | 8E | 8E | 8E | 8E | 8E | 8E | 8E | EOS |
| 7 | 1 | 8F | 8F | 8F | 8F | 8F | 8F | 8F | EOL |
| 0 | 2 | 90 | 90 | 90 | 90 | 90 | 90 | 90 | BS |
| 1 | 2 | 91 | 91 | 91 | 91 | 91 | 91 | 91 | DL |
| 2 | 2 | 92 | 92 | 92 | 92 | 92 | 92 | 92 | DC |
| 3 | 2 | 93 | 93 | 93 | 93 | 93 | 93 | 93 | IL |
| 4 | 2 | 94 | 94 | 94 | 94 | 94 | 94 | 94 | FMT |
| 5 | 2 | 95 | 95 | 95 | 95 | 95 | 95 | 95 | ↑ |
| 6 | 2 | 96 | 96 | 96 | 96 | 96 | 96 | 96 | ↓ |
| 7 | 2 | 97 | 97 | 97 | 97 | 97 | 97 | 97 | → |
| 0 | 3 | 98 | 98 | 98 | 98 | 98 | 98 | 98 | ← |
| 1 | 3 | 09 | 09 | 09 | 09 | 09 | 09 | 09 | TAB |
| 2 | 3 | 08 | 08 | 08 | 08 | 08 | 08 | 08 | BS |
| 3 | 3 | 7B | 1B | 1B | 5B | 5B | 5B | 7B | { |
| 4 | 3 | 7C | 1C | 1C | 5C | 5C | 5C | 7C | : |
| 5 | 3 | 7D | 1D | 1D | 5D | 5D | 5D | 7D | } |
| 6 | 3 | 7E | 1E | 1E | 5E | 5E | 5E | 7E | ~ |
| 7 | 3 | 5F | 1F | 1F | 7F | 7F | 7F | 5F | — |
| 0 | 4 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 0 |
| 1 | 4 | 31 | 31 | 31 | 31 | 31 | 31 | 31 | 1 |
| 2 | 4 | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 2 |
| 3 | 4 | 33 | 33 | 33 | 33 | 33 | 33 | 33 | 3 |
| 4 | 4 | 34 | 34 | 34 | 34 | 34 | 34 | 34 | 4 |
| 5 | 4 | 35 | 35 | 35 | 35 | 35 | 35 | 35 | 5 |
| 6 | 4 | 36 | 36 | 36 | 36 | 36 | 36 | 36 | 6 |
| 7 | 4 | 37 | 37 | 37 | 37 | 37 | 37 | 37 | 7 |
| 0 | 5 | 38 | 38 | 38 | 38 | 38 | 38 | 38 | 8 |
| 1 | 5 | 39 | 39 | 39 | 39 | 39 | 39 | 39 | 9 |
| 2 | 5 | 0A | 0A | 0A | 0A | 0A | 0A | 0A | LF |
| 3 | 5 | 1B | 1B | 1B | 1B | 1B | 1B | 1B | ESC |
| 4 | 5 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | SP |
| 5 | 5 | 0D | 0D | 0D | 0D | 0D | 0D | 0D | RTN |
| 6 | 5 | 2E | 2E | 2E | 2E | 2E | 2E | 2E | . |
| 7 | 5 | FF | FF | FF | FF | FF | FF | FF | BREAK |
| 0 | 6 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 0 |

# Functional Description (Continued)

## TABLE I. CODE ASSIGNMENTS (Continued)

| X | Y | Code | Control | Control & Shift or Shift Lock | Shift | Shift Loc | Shift Loc & Cap Loc | Cap Loc | Key | Additional Codes for 144-Key Option | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | X | Y | Code |
| 1 | 6 | 31 | 31 | 21 | 21 | 21 | 21 | 31 | 1 | 8 | 0 | 99 |
| 2 | 6 | 32 | 32 | 22 | 22 | 22 | 22 | 32 | 2 | 8 | 1 | 9A |
| 3 | 6 | 33 | 33 | 23 | 23 | 23 | 23 | 33 | 3 | 8 | 2 | 9B |
| 4 | 6 | 34 | 34 | 24 | 24 | 24 | 24 | 34 | 4 | 8 | 3 | 9C |
| 5 | 6 | 35 | 35 | 25 | 25 | 25 | 25 | 35 | 5 | 8 | 4 | 9D |
| 6 | 6 | 36 | 36 | 26 | 26 | 26 | 26 | 36 | 6 | 8 | 5 | 9E |
| 7 | 6 | 37 | 37 | 27 | 27 | 27 | 27 | 37 | 7 | 8 | 6 | 9F |
| 0 | 7 | 38 | 38 | 28 | 28 | 28 | 28 | 38 | 8 | 8 | 7 | A0 |
| 1 | 7 | 39 | 39 | 29 | 29 | 29 | 29 | 39 | 9 | 8 | 8 | A1 |
| 2 | 7 | 3A | 3A | 2A | 2A | 2A | 2A | 3A | : | 8 | 9 | A2 |
| 3 | 7 | 3B | 3B | 2B | 2B | 2B | 2B | 3B | ; | 8 | 10 | A3 |
| 4 | 7 | 2C | 2C | 3C | 3C | 3C | 3C | 2C | , | 8 | 11 | A4 |
| 5 | 7 | 2D | 2D | 3D | 3D | 3D | 3D | 2D | - | 9 | 0 | A5 |
| 6 | 7 | 2E | 2E | 3E | 3E | 3E | 3E | 2E | . | 9 | 1 | A6 |
| 7 | 7 | 2F | 2F | 3F | 3F | 3F | 3F | 2F | / | 9 | 2 | A7 |
| 0 | 8 | 40 | 00 | 00 | 60 | 60 | 60 | 40 | @ | 9 | 3 | A8 |
| 1 | 8 | 61 | 01 | 01 | 41 | 41 | 41 | 41 | A | 9 | 4 | A9 |
| 2 | 8 | 62 | 02 | 02 | 42 | 42 | 42 | 42 | B | 9 | 5 | AA |
| 3 | 8 | 63 | 03 | 03 | 43 | 43 | 43 | 43 | C | 9 | 6 | AB |
| 4 | 8 | 64 | 04 | 04 | 44 | 44 | 44 | 44 | D | 9 | 7 | AC |
| 5 | 8 | 65 | 05 | 05 | 45 | 45 | 45 | 45 | E | 9 | 8 | AD |
| 6 | 8 | 66 | 06 | 06 | 46 | 46 | 46 | 46 | F | 9 | 9 | AE |
| 7 | 8 | 67 | 07 | 07 | 47 | 47 | 47 | 47 | G | 9 | 10 | AF |
| 0 | 9 | 68 | 08 | 08 | 48 | 48 | 48 | 48 | H | 9 | 11 | B0 |
| 1 | 9 | 69 | 09 | 09 | 49 | 49 | 49 | 49 | I | 10 | 0 | B1 |
| 2 | 9 | 6A | 0A | 0A | 4A | 4A | 4A | 4A | J | 10 | 1 | B2 |
| 3 | 9 | 6B | 0B | 0B | 4B | 4B | 4B | 4B | K | 10 | 2 | B3 |
| 4 | 9 | 6C | 0C | 0C | 4C | 4C | 4C | 4C | L | 10 | 3 | B4 |
| 5 | 9 | 6D | 0D | 0D | 4D | 4D | 4D | 4D | M | 10 | 4 | B5 |
| 6 | 9 | 6E | 0E | 0E | 4E | 4E | 4E | 4E | N | 10 | 5 | B6 |
| 7 | 9 | 6F | 0F | 0F | 4F | 4F | 4F | 4F | O | 10 | 6 | B7 |
| 0 | 10 | 70 | 10 | 10 | 50 | 50 | 50 | 50 | P | 10 | 7 | B8 |
| 1 | 10 | 71 | 11 | 11 | 51 | 51 | 51 | 51 | Q | 10 | 8 | B9 |
| 2 | 10 | 72 | 12 | 12 | 52 | 52 | 52 | 52 | R | 10 | 9 | BA |
| 3 | 10 | 73 | 13 | 13 | 53 | 53 | 53 | 53 | S | 10 | 10 | BB |
| 4 | 10 | 74 | 14 | 14 | 54 | 54 | 54 | 54 | T | 10 | 11 | BC |
| 5 | 10 | 75 | 15 | 15 | 55 | 55 | 55 | 55 | U | 11 | 0 | BD |
| 6 | 10 | 76 | 16 | 16 | 56 | 56 | 56 | 56 | V | 11 | 1 | BE |
| 7 | 10 | 77 | 17 | 17 | 57 | 57 | 57 | 57 | W | 11 | 2 | BF |
| 0 | 11 | 78 | 18 | 18 | 58 | 58 | 58 | 58 | X | 11 | 3 | C0 |
| 1 | 11 | 79 | 19 | 19 | 59 | 59 | 59 | 59 | Y | 11 | 4 | C1 |
| 2 | 11 | 7A | 1A | 1A | 5A | 5A | 5A | 5A | Z | 11 | 5 | C2 |
| 3 | 11 | | ON→FC | | | OFF→FB | | | Cap Loc | 11 | 6 | C3 |
| 4 | 11 | | ON→FE | | | OFF→FD | | | Shift Loc | 11 | 7 | C4 |
| 5 | 11 | | NO CODE | | | NO CODE | | | RPT | 11 | 8 | C5 |
| 6 | 11 | | | | | | | | CNTR | 11 | 9 | C6 |
| 7 | 11 | | | | | | | FD* | SHIFT | 11 | 10 | C7 |
| CNTR | ESC | | ON→FA | | | | | | PGM | 11 | 11 | C8 |
| CNTR | ; | | OFF→F9 + CHARACTER STRING† | | | | | | | | | |

* If Shift Loc is ON, Shift will transmit FD and end Shift Loc ON mode.

† First time only.

## Functional Description (Continued)

### KEY CYCLE TIMING

Valid key closures are detected by the MM57499 by recurring strobe/scan events. The MM57499 strobes rows of the matrix at rates unique to the configuration (depending on either the 96 or 144-key mode option) of the MM57499 and the number of keys down.

The MM57499 processes a key if the minimum debounce requirements are met. To insure debounce the MM57499 verifies the key down closure. (Timing is summarized in the Electrical Characteristics table.) After the key has been verified down, the MM57499 recognizes the key as being valid and processes the ASCII code. Before the next key is processed, the previous key pressed must have been up for three scan times. If sufficient dwell on the key is encountered the MM57499 will go into the automatic repeat mode until the key is detected to be up. Strobe/scan times are dependent on the keyboard situation. With no key pressed the full matrix scan is accomplished in 2.5 ms (3.4 ms)*. Under normal operating conditions, burst rates of 423 words per minute (327)* typical can be realized.

### TRANSMIT

Designated as $T_t$ in the key cycle timing diagram, the transmit chain is made up of 1 stop bit, 1 start bit, 8 data bits, and 1 stop bit, in that order. The timing is 0.833 ms per bit, which is 9.16 ms (1200 baud) for the complete transmit cycle.

* 144-key mode

The only situation in which this timing would occur is programmable phrase mode, where the 8-bit data words are separated by 2 stop and 1 start bits. Under normal operating conditions debounce time will stretch the stop bits by transmitting a continuous logical "1."

### RECEIVE STATUS

The addition of an external serial in-parallel out shift register permits status indicator drive capability. This status information is inputted to pin 9 of the MM57499. The serial data chain must have a valid start bit and at least 1 stop bit or the MM57499 will not accept the status change. The status is an 8-bit data word, and is clocked into the status latch 0.178 ms after detecting a stop bit. The data chain into the receive input is sampled 0.1 ms into the start bit and every 0.833 ms thereafter for the next 9 bits (to include 8 data bits and 1 stop bit).

The status word read by the MM57499 encoder is complemented. The external serial to parallel shift register LED driver will also do a complement of the data word. Therefore the status indicator device (LED) is on with a Logical "1" data bit received.

Data is transmitted to the status latch by a serial process. The status data transfer is completed in 8 cycle times (see *Figure 3* input/output timing diagram).

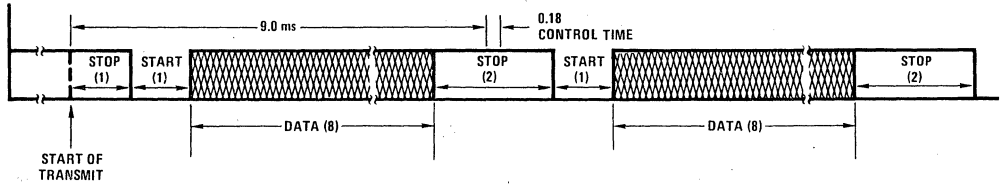



**FIGURE 2. Recurring Transmit Stream**

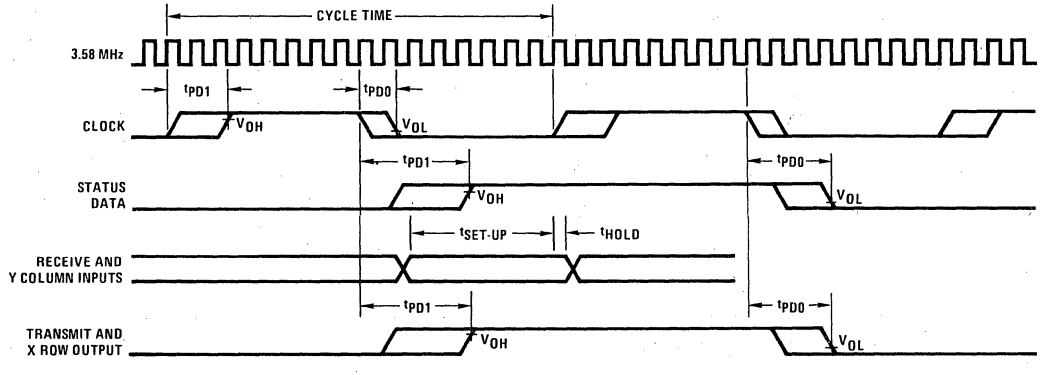



**FIGURE 3. Input/Output Timing Diagrams**

## Functional Description (Continued)

### TRANSMIT/RECEIVE INTERRUPT

In the event the MM57499 is transmitting a character and, at some time during that process a status word update is sent to the MM57499, an interrupt in the transmit stream will occur. The transmit output pin will drop to a logical low and remain in that state until the received word is processed. Once a break has been detected, the processor can determine that the data is not valid. The MM57499 will process the received word and retransmit the interrupted character. If the receiver status option is not utilized, normal operation (without interrupts) will occur (see *Figure 4* for transmit out and interrupt timing).

### SHIFT LOC AND CAP LOC KEYS

Both the *SHIFT LOC* and *CAP LOC* are software latching keys. When either is depressed they transmit 8-bit codes to indicate a mode change. When the *SHIFT LOC* is pressed, a shift loc code FE is transmitted and all appropriate characters are shifted. A second depression of the *SHIFT LOC* key will cause a shift loc OFF code FD to be transmitted and lower case ASCII is again transmitted.

The *SHIFT KEY* (not the *SHIFT LOC* key) will not transmit a mode change unless the *SHIFT LOC* is ON. Keeping the *SHIFT KEY* depressed accomplishes the same function as the *SHIFT LOC,* much the same as most common typewriters. (The shift key has a momentary action, the shift loc key locks the keyboard until it is deliberately released via the *SHIFT* or *SHIFT LOC* key). Relieving the *SHIFT KEY* returns the character transmit to lower case ASCII. Depressing the *SHIFT KEY* while the *SHIFT LOC* key is ON causes an FD code to be transmitted and the shift loc is terminated.

The *CAP LOC* is similar in function to the *SHIFT LOC* in that a cap loc ON code FC is transmitted upon a depression of the *CAP LOC* key. The *CAP LOC* mode will capitalize alphabet and appropriate keys; i.e., if a "B" key is depressed, a capital B is transmitted. The *SHIFT LOC* key accommodates all other key secondary functions. A second depression of the *CAP LOC* key transmits a cap loc OFF code FB and the keyboard is returned to normal. When more than one mode is entered coincidentally, refer to the key codes for mode dominance.

### 96-KEY OR 144-KEY SELECTION

The MM57499 can be configured to either a 96-key or 144-key arrangement.

In the 96-key mode, a standard 8 × 12 matrix keyboard is required, and the X-Y matrix lines are connected directly to the keyboard as shown in the minimal configuration (*Figure 5*). In this configuration pin 21 is used as a strobe line. Note the diode isolation requirements also shown in *Figure 7*.

If the 144-key mode is desired, pin 21 must be strapped to ground to actuate the 144-key codes. An external 4 to 12-line decoder (12 lines of a 4 to 16-line decoder) must be interfaced between the MM57499 and the keyboard on the X matrix lines, as shown in *Figure 6*.

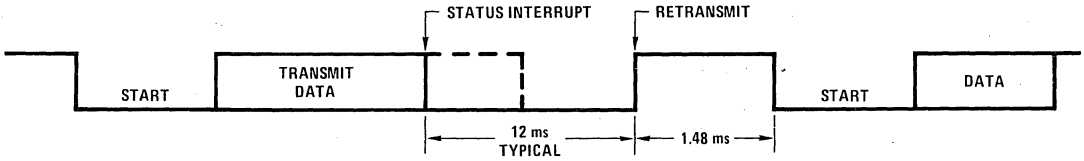



**FIGURE 4. Transmit Output and Interrupt Timing**

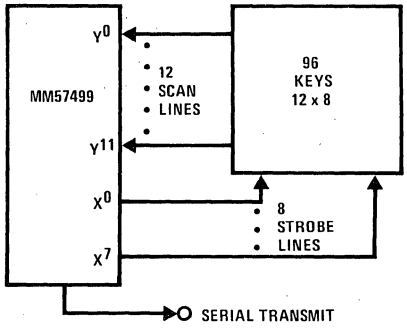



**FIGURE 5. Minimal Configuration—96-Key**

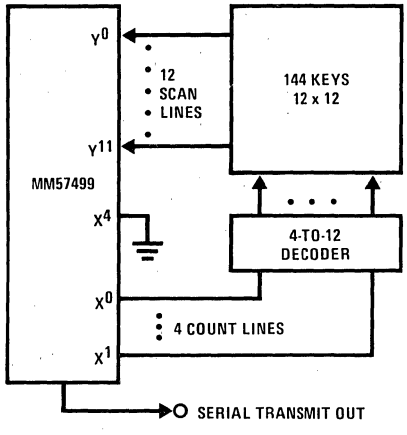



**FIGURE 6. 144-Key Configuration**

## Functional Description (Continued)

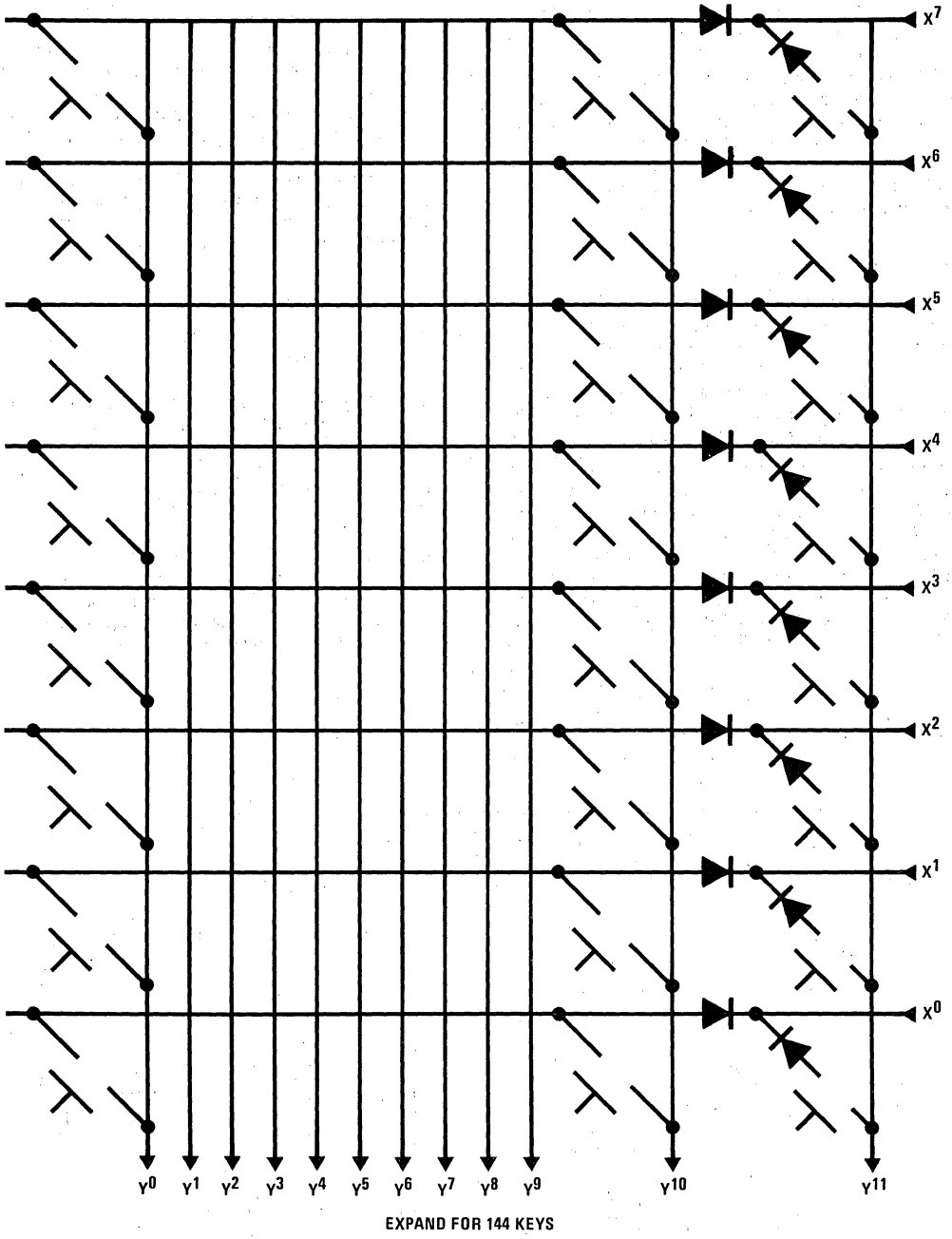



**FIGURE 7. 96-Key Board Matrix (Note diode direction)**

## Functional Description (Continued)

### CHARACTER REPEAT

As explained previously in the key cycle timing paragraphs, normal depression of a character key initiates a transmission of the character after a minimal scan/debounce time. Keeping that same character key depressed for one second will cause an automatic repeat of that character, followed by successive transmits.

These repeat rates are summarized in the AC Electrical Characteristics table. A secondary method of successively repeating the character is by use of the *REPEAT KEY*. In this case the desired character and the repeat key are depressed simultaneously. The character repeat begins immediately, with no initial pause.

### PROGRAMMABLE PHRASE

In many terminal applications a certain word, phrase, name, title, etc. is required periodically. It may also be necessary that indent spacing or a predetermined tab sequence be recalled. The MM57499 has the unique capability of storing up to 14 characters of key data, whether they actually be key characters or control codes. These 14 key strokes can be stored for later use.

To program this memory first press the *CONTROL ESC* key. This causes the hex code FA to be transmitted and indicates the programming mode is active. This FA code could be used to enable a status indicator (see status applications for precautions). The next 1 to 14 key strokes will be stored in the MM57499 memory for recall upon command. Keying the *CONTROL SEMICOLON* key will cause the programmed characters to be transmitted at 1200 baud.

The *first* time this stored instruction or phrase is transmitted, a hex code F9 is also included at the beginning of the transmit data stream to indicate the termination of the programming mode. (The status indicator could now be turned off if a status change command is given.) Additional keying of the *CONTROL SEMICOLON* keys retransmits the stored characters or control codes (programmed phrase) as many times as recalled and until the MM57499 memory is reprogrammed (via the same steps as described above) with a new phrase. A power down or a *RESET* operation will also clear the memory. Summarizing, the programming steps are:

1. *CONTROL ESC*
2. Program—up to 14 key strokes
3. *CONTROL SEMICOLON*
4. For additional recalls of memory key *CONTROL SEMICOLON*
5. For reprogramming, repeat steps 1, 2, 3 above

Until the *CONTROL SEMICOLON* is keyed, the MM57499 will remain in the programming mode, regardless of how many programming keys have been pressed, and even though only the first 14 key strokes are stored. The phrase is programmable from 1 to 14 key strokes, therefore it is not necessary to program all 14 strokes prior to keying the *CONTROL SEMICOLON*. If the 14 key stroke limit is inadvertently exceeded and additional key strokes are entered, the MM57499 will transmit an 07 bell code after the 14th key stroke and for every additional key stroke thereafter as a warning device until the *CONTROL SEMICOLON* is keyed.

If the *CONTROL SEMICOLON* is keyed and the device memory is unprogrammed (empty), the MM57499 will ignore the keying.

### STATUS LATCH APPLICATIONS

The status latches may be used for various applications. An 8-bit word with start and stop bits is received and then clocked into the status latch immediately. If an invalid word is received (i.e., no stop bit) the MM57499 will revert to the previous valid status word and clock it into the status latch. The detection of the leading edge of a start bit on the receive line (pin 15) causes all other operations within the MM57499 to cease until the status word has been received and latched. Should the MM57499 be processing a key when a status word is sent, the operation is restarted after the status word is received by the MM57499. If the MM57499 is transmitting a word when a valid status start bit is received, the transmit line drops to a logical "0" (low) to denote a break (00). After the receive is completed, the MM57499 will retransmit the interrupted character in its entirety (see *Figure 4*).

One status application would be to indicate the state of the keyboard. If *SHIFT LOC* is pressed a hex FE is transmitted to the CPU. The CPU at this time can send back a status word to illuminate a single LED to be the *SHIFT LOC* indicator. Upon the second depressing of *SHIFT LOC* the MM57499 transmits a hex FD. At this time the CPU can send back a status word to turn off the *SHIFT LOC* indicator.

When using the status indicators in conjunction with the programmable phrase option, care must be taken to guarantee the integrity of the character stream. If it is desired to indicate the programming active state with the keyboard status latch, some guidelines must be followed. When entering the programming mode a hex FA is transmitted to the CPU. In order to insure the integrity of the following key strokes (to be stored as the programmed phrase) it is necessary to initiate transmission of the status word within 10 ms from the time the FA code is received. *No other status changes should be sent from the CPU during "PROGRAMMING MODE ON" sequence.* There is a small probability that a status word interrupt may cause a key stroke to be inadvertently ignored. The minimum time to press the next key plus 10 ms is the maximum allowable delay. In most applications this is more than sufficient time to start the status correction. To indicate the termination of the programming mode, care must also be taken to send the status change within 10 ms after receiving a mode change from the keyboard to assure that a conflict of send or receive data does not occur. During normal key entry the keyboard encoder is capable of processing a status word at any time.

### INITIALIZATION

The reset logic, internal to the MM57499, will initialize (clear) the device upon power-up if the power supply rise

## Functional Description (Continued)

time is less than 1 ms and greater than 1 $\mu$s. If the power supply rise time is greater than 1 ms, the user must provide an external RC network and diode to the $\overline{\text{RESET}}$ pin as shown below. The $\overline{\text{RESET}}$ pin is configured as a Schmitt trigger input. If the $\overline{\text{RESET}}$ pin is not used it should be connected to $V_{CC}$. Initialization will occur whenever a logic "0" is applied to the $\overline{\text{RESET}}$ input, provided it stays low for at least 10 $\mu$s.

Table II is a routine showing how to read from the serial keyboard encoder with the INS8250 ACE using the INS8060 SCAMP II Microprocessor.

**TABLE II**

```
START:   LD     005 (P3)      ; READ ACE STATUS REG.

         XAE                   ; MOVE STATUS TO E REG.
         LDE

         ANI    008            ; IS FRAMING ERR SET
         JNZ    ERR            ; FE IS SET, JUMP

         LDE                   ; FE NOT SET, IS RECEIVER READY?
                               ; WE COULD HAVE REREAD THE STATUS
                               ; REG. BECAUSE RECEIVER READY ONLY
                               ; CLEARS UPON READING THE RECEIVER
                               ; OR WRITING A 000 TO THE STATUSER
                               ; REG. IF WE WISH TO RING BEL UPON
                               ; RECEIPT OF AN OVER RUN ERROR, WE
                               ; CANNOT REREAD THE ACE STATUS BECAUSE
                               ; THE FIRST READ CLEARED OUT THE ERRORS
         ANI    001            ;
         JNZ    RECEIV         ; RECEIVER IS READY, JUMP

         JMP    START          ; RECEIVER NOT READY, REPEAT LOOP

ERR:     LD     000 (P3)       ; READ ACE RECEIVER, THROW AWAY DATA
         JMP    START          ; GO BACK TO SCAN BEGINNING
```

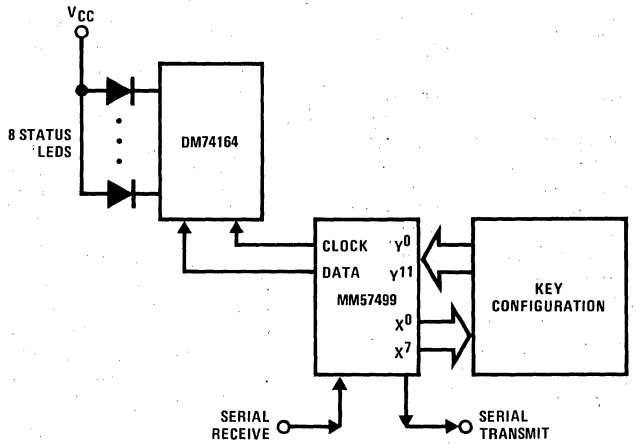



**FIGURE 8. Status Indicator Configuration**

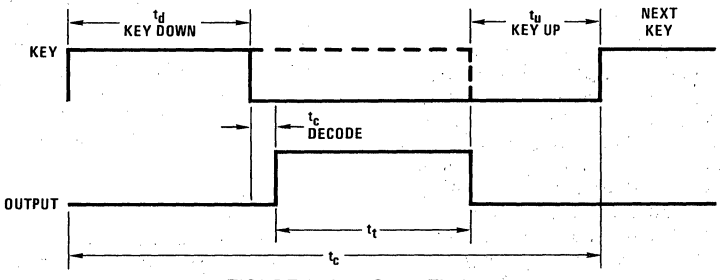



**FIGURE 9. Key Cycle Timing**

## Functional Description (Continued)

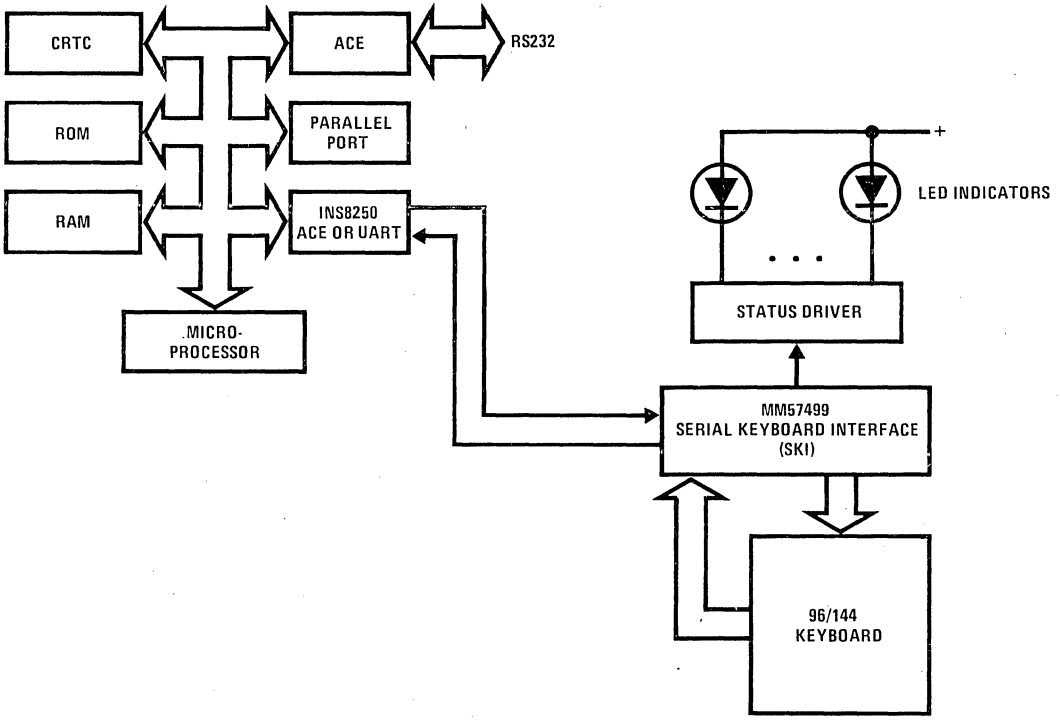



FIGURE 10. MM57499 System Concept

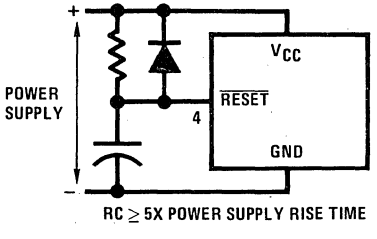



FIGURE 11. Power-Up Reset Circuit

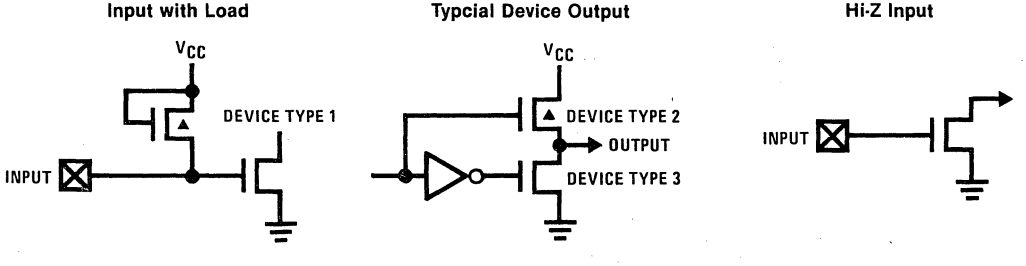



FIGURE 12. Input/Output Devices

## Functional Description (Continued)

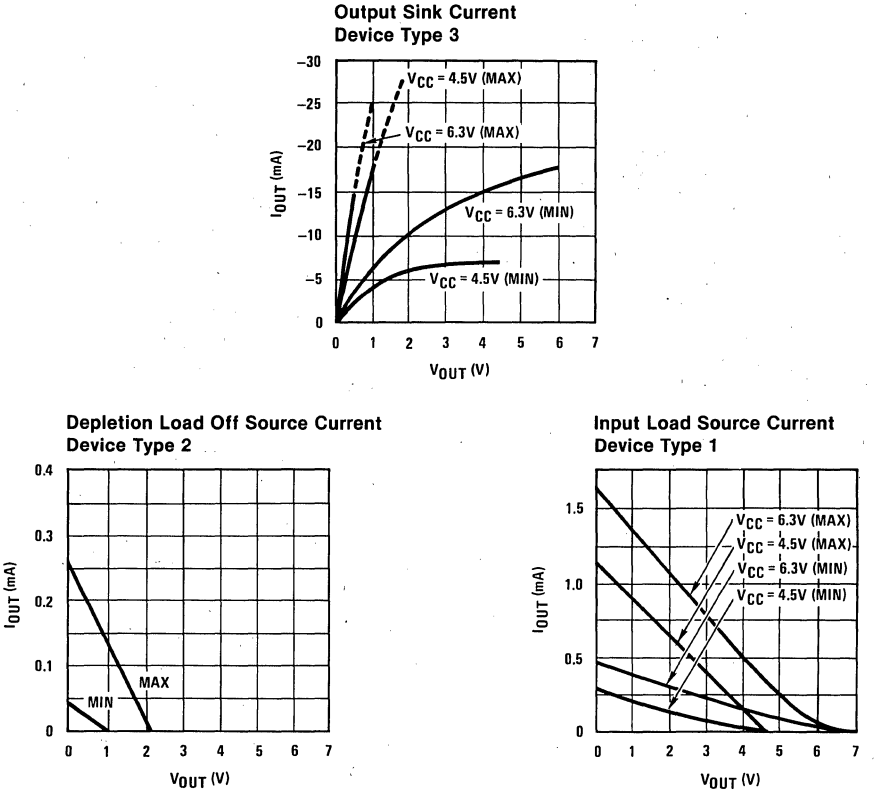



**FIGURE 13. Input/Output Device Characteristics**

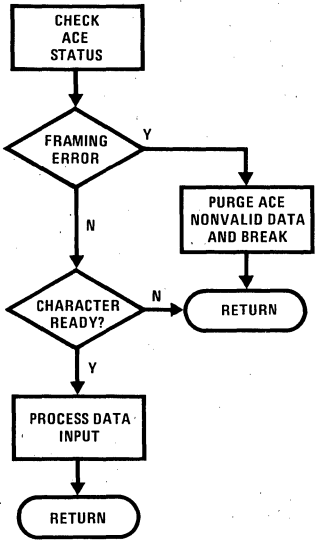
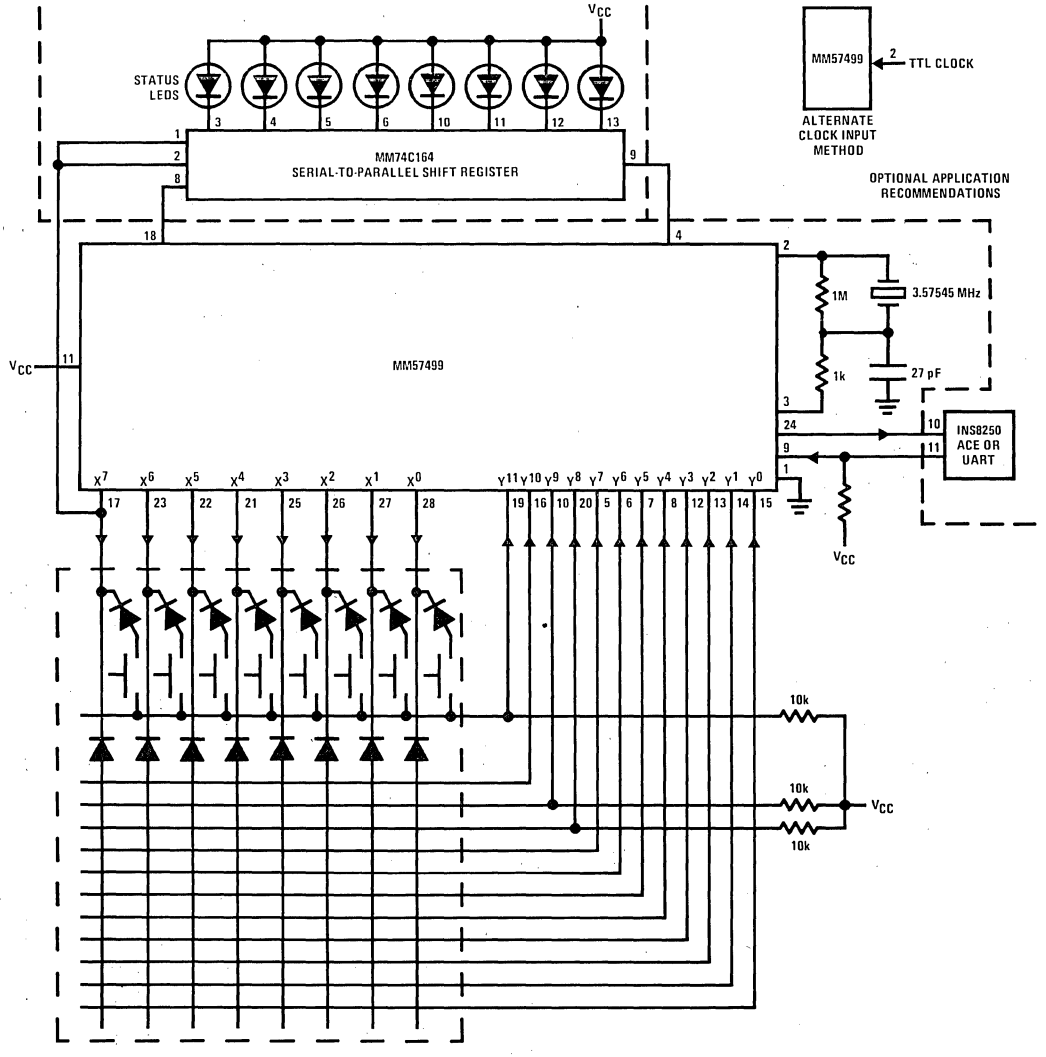


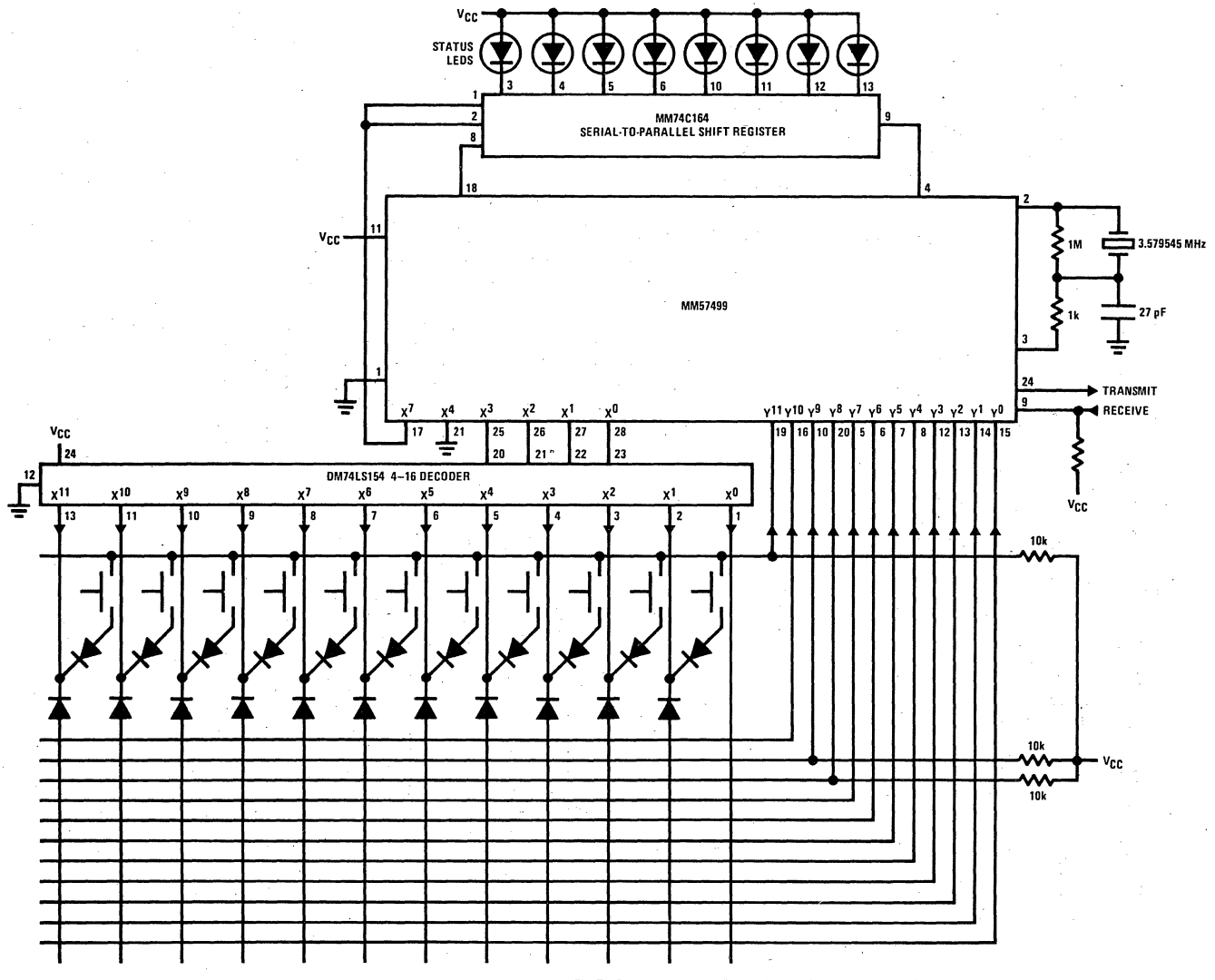

**FIGURE 14. INS8250 ACE Receive Flow When Utilizing Status Latches (Interrupt I/O Mode)**

## Functional Description (Continued)

### Functional Application 96-Key Mode



## Functional Application 144-Key Mode



National Semiconductor

Electronic Data Processing

# MM5863 12-Bit Binary A/D Building Block

## General Description

The MM5863 is the digital controller for the LF13300D* analog building block. Together they form an integrating 12-bit A/D converter. The MM5863 provides all the necessary control functions, plus features like auto zeroing, polarity and overrange indication, as well as continuous conversion. The 12-bit plus sign parallel and serial outputs are TRI-STATE® TTL level compatible. The device also includes output latches to simplify data bus interfacing.

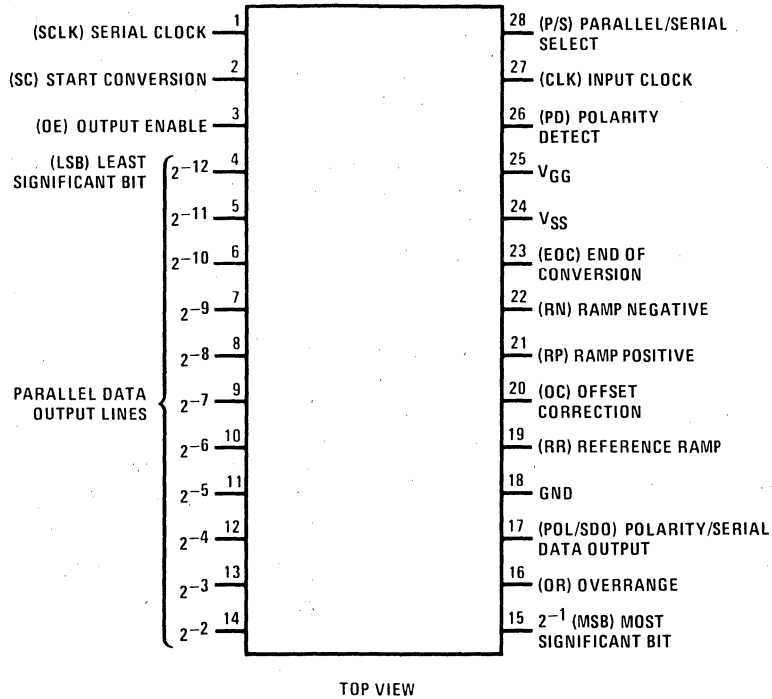
*See LF13300D data sheet for more information

## Features

- 12-bit binary output
- Parallel or serial output
- Parallel TRI-STATE output
- Polarity indication
- Overrange indication
- Continuous conversion capability
- 100% overrange capability
- 5V, −15V power requirements
- TTL compatible
- Clock frequency to 500 kHz

## Connection Diagram

Dual-In-Line Package




TOP VIEW

Order Number MM5863N
See Package 23

## Absolute Maximum Ratings

| | |
|---|---|
| Supply Voltage ($V_{SS}$) | 5.25V |
| Supply Voltage ($V_{GG}$) | −16.5V |
| Voltage at Any Input | 5.25V |
| Operating Temperature | 0°C to +70°C |
| Storage Temperature | −40°C to +150°C |
| Clock Frequency | 500 kHz |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$V_{SS}$ = 5V, $V_{GG}$ = −15V, 0°C to +70°C, unless otherwise specified.

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Power Supply Voltage ($V_{SS}$) | | 4.75 | 5.00 | 5.25 | V |
| Power Supply Voltage ($V_{GG}$) | | −13.5 | −15.00 | −16.5 | V |
| Power Supply Current ($I_{SS}$) | | | | 28 | mA |
| Power Supply Current ($I_{GG}$) | | | | 34 | mA |
| Logic "1" Input Voltage | | 3.4 | | | V |
| Logic "0" Input Voltage | | | | 0.8 | V |
| Logic "1" Output Voltage | $V_{SS}$ = 4.75, $I_{OH}$ = 100 μA | 3.8 | | | V |
| Logic "0" Output Voltage | $V_{SS}$ = 5.25, $I_{OL}$ = −1.6 mA | | | 0.4 | V |
| Width of EOC | Auto Cycle | 5/f | | | Sec |
| Prop. Delay PD to EOC | | 4/f | | 5/f+1 μs | Sec |
| Output Enable Time | OE to Any Data Output, SC = 1, P/S = 0 | | | 1.0 | μs |
| Output Disable Time | OE to Any Data Output, SC = 1, P/S = 0 | | | 2.4 | μs |
| Output Enable Time | P/S to Any Data Output Except Polarity, SC = 1, OE = 0 | | | 0.9 | μs |
| Output Disable Time | P/S to Any Data Output Except Polarity, SC = 1, OE = 0 | | | 2.2 | μs |
| Output Enable Time | SC to Any Data Output, OE = 0, P/S = 0 | | | 1.0 | μs |
| Output Disable Time | SC to Any Data Output, OE = 0, P/S = 0 | | | 2.4 | μs |
| Prop. Delay Serial Clock | SCLK to POL/SDO | | | 0.6 | μs |
| Conversion Time | Full Scale | | | 8966/f | Sec |
| Conversion Time | 100% Overrange | | | 13062/f | Sec |

# Functional Description

## OPERATION

The MM5863 is designed for use with the LF13300 analog front end. Four control signals are supplied to the LF13300 and 1 control signal is required from the LF13300. The conversion cycle is composed of 5 distinct phases. They are: Phase I — Offset Correct; Phase II — Polarity Detect; Phase III — Offset Correct; Phase IV — Ramp Unknown; Phase V — Ramp Reference.

**Phase I — Offset Correct (256 Clock Periods)**

This phase is initiated by taking the Start Conversion (SC) and the Output Enable (OE) lines to a logic "1". At this time, Offset Correct (OC) will be a logic "1". The LF13300 requires this phase to correct any intrinsic offset voltage errors prior to the polarity detect phase.

**Phase II — Polarity Detect (256 Clock Periods)**

This phase is used to determine polarity of the analog input. At the midpoint of this phase, PD from the LF13300 is examined for polarity. If PD = logic "1", then the input voltage is positive. If PD = logic "0", then the input is negative. The Ramp Positive signal (RP) will be a logic "1", and Offset Correct will be logic "0" for the entire phase of 256 clock periods. The above operation is also necessary to determine which integrator input (positive or negative) of the LF13300 should be used for proper A/D conversion (see LF13300 data sheet).

**Phase III — Offset Correct (256 Clock Periods)**

This phase is identical to Phase I and is used by the LF13300 to eliminate any offsets induced as a result of the Polarity Detect Phase. Offset Correct (OC) will be at a logic "1".

**Phase IV — Ramp Unknown (4096 Clock Periods)**

The unknown input voltage is integrated for a fixed time during this phase. The result of the Phase II Polarity Detect Cycle determines whether RP or RN will be at logic "1". If Phase II indicates a positive input, the RP signal will be a logic "1". If phase II indicates a negative input, Ramp Negative (RN) will be a logic "1". These 2 signals will never be at logic "1" simultaneously.

**Phase V — Ramp Reference**

This phase is a variable length phase depending on the magnitude of the analog input voltage. During this time, Ramp Reference (RR) will be in the logic "1" state. When PD goes to a logic "0" state, or when the internal counter reaches 100% of full scale (8192 clock periods), the Ramp Reference (RR) signal goes to the logic "0" state, the counter output is loaded into the output register, and the End of Conversion, (EOC) signal goes to a logic "1". The Polarity Bit will reflect whatever value was determined during Phase II. The output register will hold the data until a new conversion is completed and new data is loaded into the register. The OE line must be low in the logic "0" state and SC must be high in the logic "1" state to enable the outputs.

## DATA OUTPUTS

Both serial and parallel outputs are available. In either case, OE must be low and SC must be high to enable the outputs. For parallel output, the P/S line must be low in the logic "0" state. For serial outputs, the P/S line must be high. In the serial mode, the data is shifted out of the Polarity/Serial Output POL/SDO line and all other data outputs are in the high impedance state. Each Serial Clock (SCLK) will right shift the output register one bit. Thus, 13 clock pulses are required to fully shift out the data. The data will be shifted out in the following order: Polarity, Overrange, MSB, 2SB, 3SB, . . ., LSB. If OE and P/S are in the logic "0" state and SC in the logic "1" state, all outputs will momentarily go to the logic "1" state for 1 clock period immediately preceding EOC.

## CONTINUOUS CONVERT MODE

In this mode, the End of Conversion (EOC) output is connected to the OE input. As long as SC is in the logic "1" state, then each EOC will initiate a new conversion. The data outputs will be disabled for the first 5 clock cycles after EOC goes high.

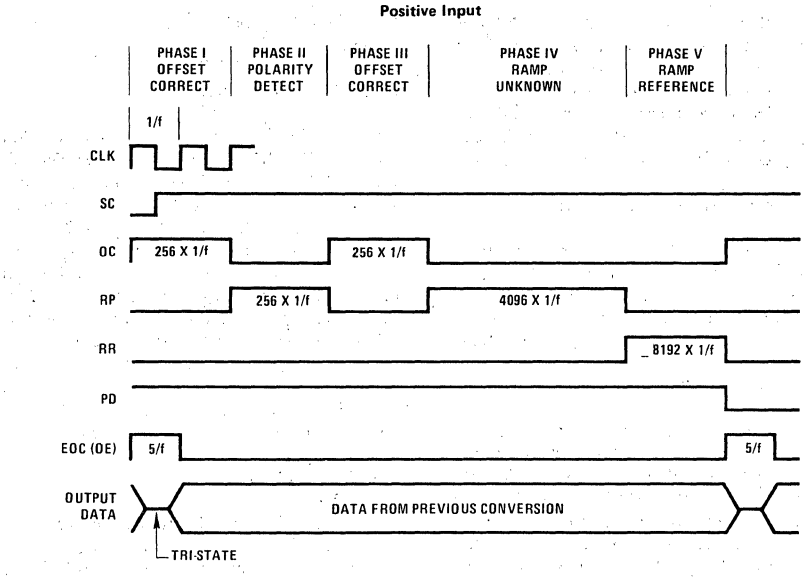
# Truth Table

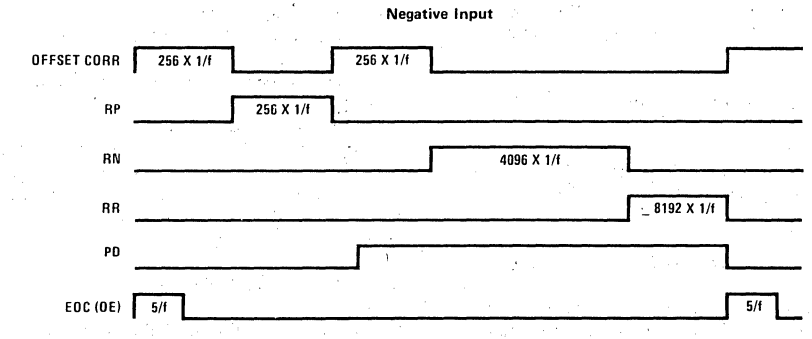
| INPUT | SC | OE | P/S | LSB | | | | | | | | | | | MSB | OVER-RANGE | POLARITY |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 100% Full Scale | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Full Scale | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| Zero | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Zero | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| −Full Scale | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| −100% Full Scale | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| Any | 1 | 1 | X | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z |
| Any | 1 | 0 | 1 | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z | Serial Output |
| Any | 0 | X | X | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z | Z |

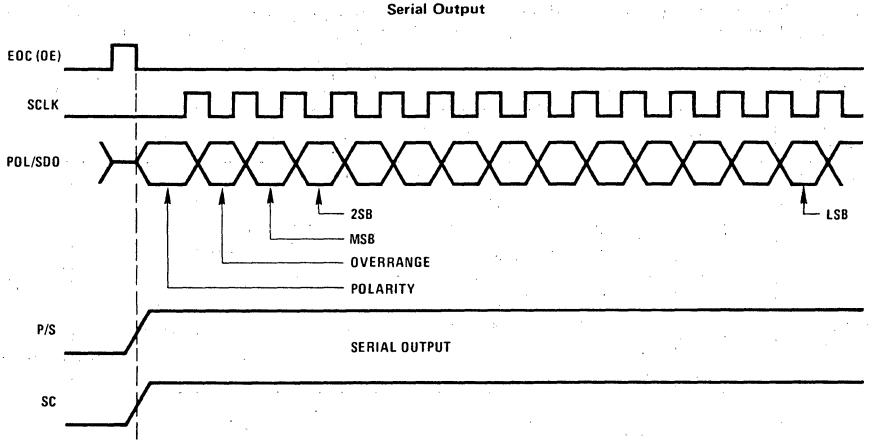
1 = High
0 = Low
Z = High Impedance
X = Don't Care

## Timing Diagrams

The following timing diagrams are shown for the MM5863 connected in the auto-cycle mode.

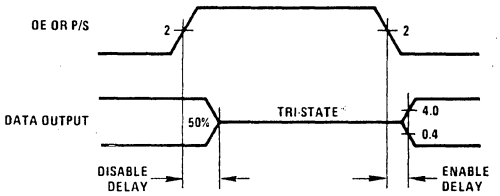


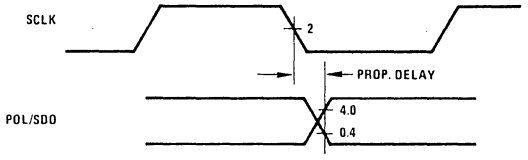



## Timing Diagrams (Cont'd)

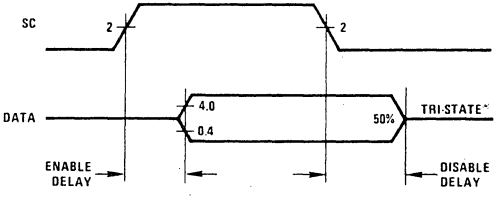
**Output Enable/Disable Time**




**Serial Clock Delay**

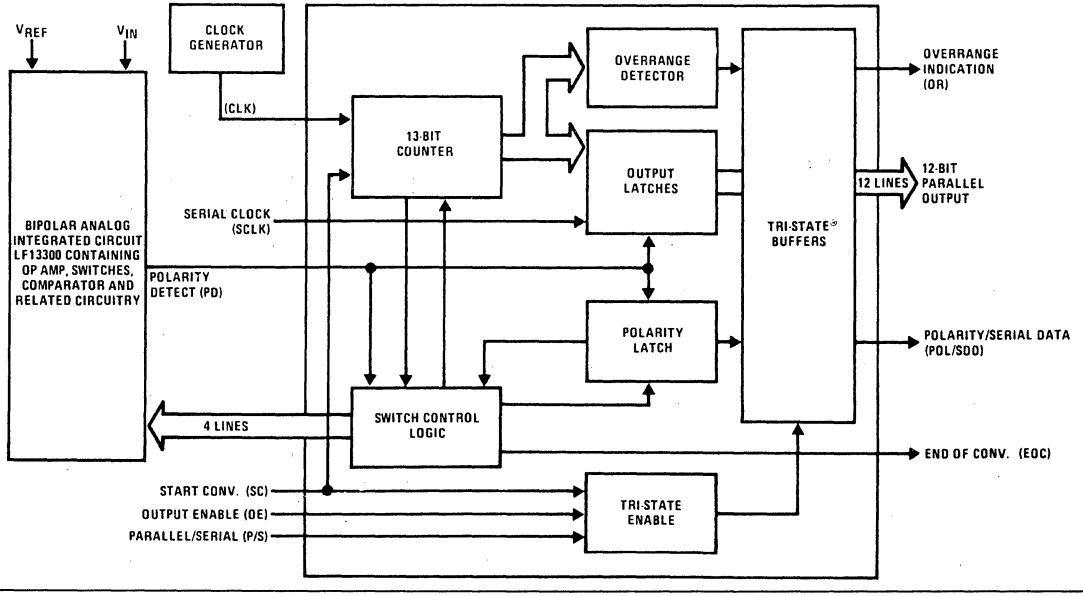



**Output Enable/Disable Time**



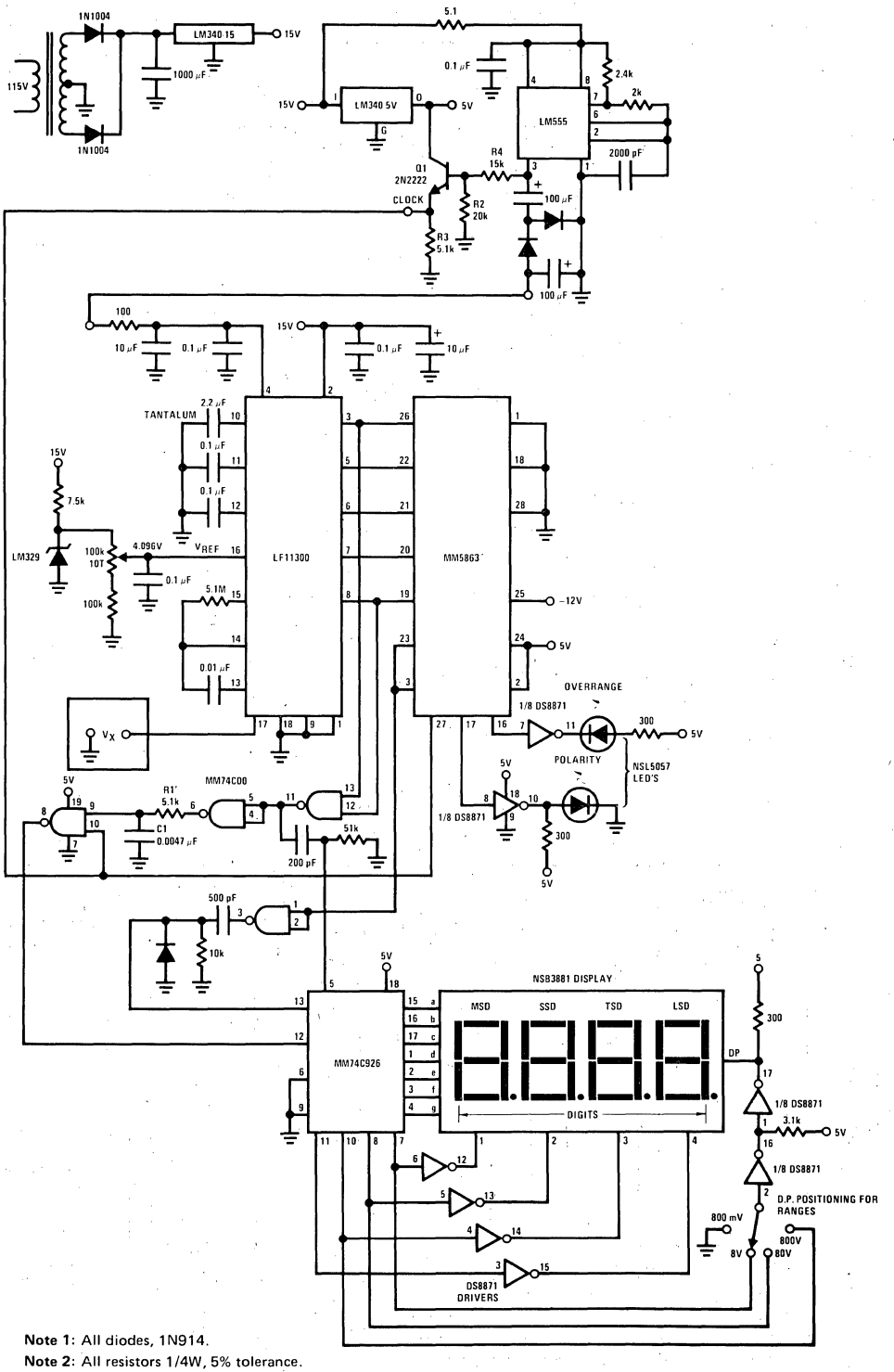

## Block Diagram

**Digital Control Integrated Circuit**



## Typical Application

3 3/4–3 1/2-Digit DPM




**Note 1:** All diodes, 1N914.

**Note 2:** All resistors 1/4W, 5% tolerance.

**Note 3:** Circuit drawn for 8V full scale operation input scaling not shown.

National Semiconductor

# MM5865 Universal Timer

## General Description

The MM5865 Universal Timer is a monolithic MOS integrated circuit utilizing P-channel low-threshold, enhancement mode and ion-implanted depletion mode devices. The chip contains all the logic required to control the two 4-digit counters, blank leading zeros, compare the two counters and to cascade with another MM5865. Input pins start, stop, reset and set the counters, determine which of the 7 functions is performed, the resolution of the display (0.01 sec, 0.1 sec, 1 sec, or external clock) and what modulo the counters divide by. Outputs include the comparator output, multiplexed BCD outputs and digit enables. The BCD outputs interface directly with MM14511, a BCD to 7-segment decoder, which interfaces with a LED display. The digit enable outputs of 2 cascaded MM5865's interface directly with a DM8863 LED 8-digit driver. A DS8877 or DS75492 Hex Digit Driver may be used with a single MM5865. The digit enable outputs interface directly with a DM8863, a LED digit driver. The 7 functions include start-stop with total elapsed time, start-stop with accumulative event time, split, sequential with total elapsed time, rally with total elapsed time, program up count and program down count. The circuit uses a 32.8 kHz crystal or an external clock and is packaged in a 40-lead dual-in-line package.

## Applications

- Stop watch
- Kitchen timer
- Oven timer
- Event timer/counter
- Rally timer
- Navigational timer
- Industrial timer/counter

## Features

- Function 1: Standard Start-Stop with total elapsed time memory
- Function 2: Standard Start-Stop with total accumulative event time
- Function 3: Sequential with total elapsed time memory
- Function 4: Standard split
- Function 5: Rally with total elapsed time memory
- Function 6: Programmable up count. Repeatable upon command
- Function 7: Programmable down count
- Comparator output
- Crystal controlled oscillator (32.8 kHz)
- External clock input (option)
- Provides external clock
- Select resolution
- Select count up or down
- Select modulo 6 or 10 for digits 2, 3 and 4
- Blanking between digits
- Leading-zero blanking
- Multiplex rate output
- External multiplex rate input (option)
- Can be cascaded
- Waiting state indicator
- Simple interface to LED display
- Elimination of illegal time display at turn-on
- Wide power supply range 7V–20V

## Block and Connection Diagrams

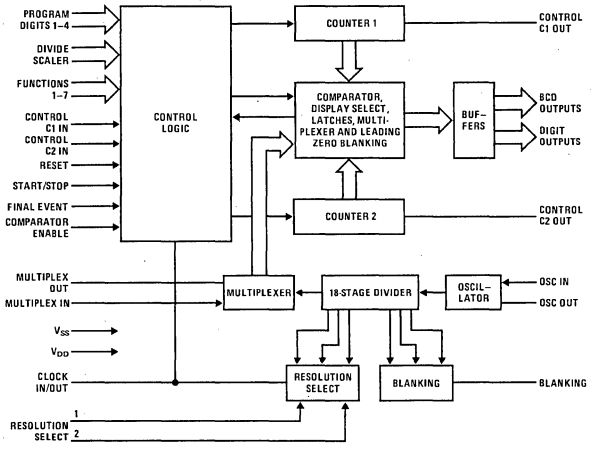
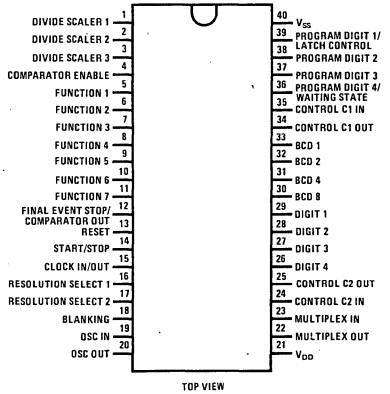



FIGURE 1.

**Dual-In-Line Package**




**Order Number MM5865N**
**See Package 24**

FIGURE 2.

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at Any Pin | $V_{SS}$ + 0.3V to $V_{SS}$ − 25V |
| Operating Temperature | −25°C to +70°C |
| Storage Temperature | −65°C to +150°C |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics

$T_A$ within operating range, 7V ≤ $V_{SS}$ ≤ 20V, $V_{DD}$ = 0V, unless otherwise specified.

| | PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|---|
| $I_{DD}$ | Power Supply Current | | | 7 | 15 | mA |
| | Input Frequency at OSC IN | $V_{SS} \geq 10V$ | dc | 32.8 | 80 | kHz |
| | Multiplex Frequency | $V_{SS} \geq 10V$ | dc | 0.4 | 80 | kHz |
| | Blanking Frequency | $V_{SS} \geq 10V$ | dc | 0.8 | 10 | kHz |
| | Clock Frequency | $V_{SS}$ = 7V | dc | 0.1 | 10 | kHz |
| | | $V_{SS}$ = 10V | dc | | 100 | kHz |
| | Input Levels | | | | | |
| | Input Logic Low | Internal Resistor | $V_{DD}$ | | $V_{DD}$+1 | V |
| | Input Logic High | ~100k to $V_{DD}$ | $V_{SS}$− 1 | | $V_{SS}$ | V |
| | **OUTPUT CURRENTS** | | | | | |
| | Digit and BCD Outputs | $V_{SS}$ = 7V | | | | |
| | Source Current | $V_{OUT}$ = $V_{SS}$ −2V | 1 | | | mA |
| | Sink Current | $V_{OUT}$ = $V_{SS}$ − 6.3V | 1 | | | μA |
| | Blanking Output | $V_{SS}$ = 7V | | | | |
| | Source Current | $V_{OUT}$ = $V_{SS}$ − 2V | 1 | | | mA |
| | Sink Current | $V_{OUT}$ = $V_{SS}$ − 6.3V | 1 | | | μA |
| | Multiplex Output | $V_{SS}$ = 7V | | | | |
| | Source Current | $V_{OUT}$ = $V_{SS}$ − 2.5V | 500 | | | μA |
| | Sink Current | $V_{OUT}$ = $V_{SS}$ − 6.3V | 8 | | | μA |
| | Clock Output | $V_{SS}$ = 7V | | | | |
| | Source Current | $V_{OUT}$ = $V_{SS}$ − 4V | 10 | | | μA |
| | Sink Current | $V_{OUT}$ = $V_{SS}$ − 6.3V | 5 | | | μA |
| | Control C1, C2 Outputs | $V_{SS}$ = 7V | | | | |
| | Source Current | $V_{OUT}$ = $V_{SS}$ − 2.5V | 500 | | | μA |
| | Control C1, C2 Inputs | $V_{SS}$ = 7V | | | | |
| | Sink Current | $V_{IN}$ = $V_{SS}$ − 6.3V | 8 | | | μA |
| | Comparator Output | $V_{SS}$ = 7V | | | | |
| | Source Current | $V_{OUT}$ = $V_{SS}$ − 2V | 1 | | | mA |
| | Sink Current | $V_{OUT}$ = $V_{SS}$ − 6.3V | 1 | | | μA |
| | Waiting State Indicator | $V_{SS}$ = 7V | | | | |
| | Source Current | $V_{OUT}$ = $V_{SS}$ − 2V | 1 | | | mA |
| | Sink Current | $V_{OUT}$ = $V_{SS}$ − 6.3V | 1 | | | μA |

## Functional Description

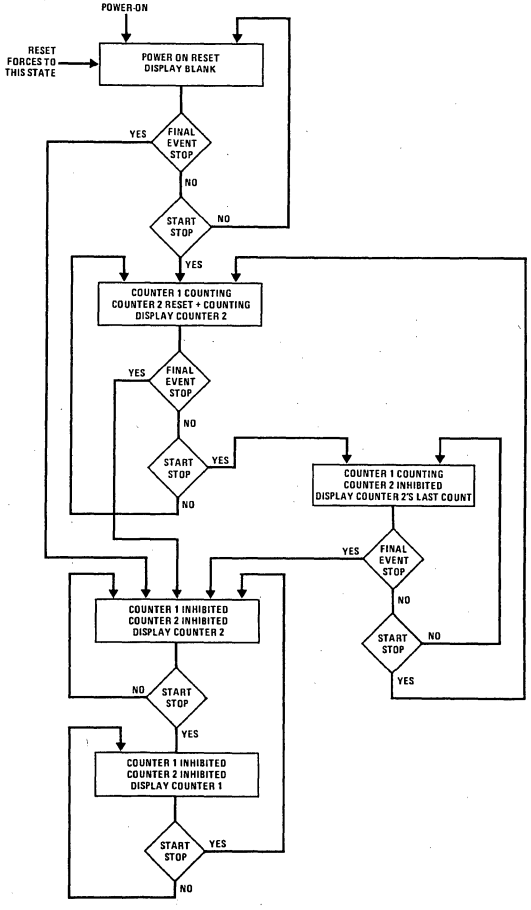
A block diagram of the MM5865 Universal Timer is shown in *Figure 1.* A connection diagram is shown in *Figure 2.* Unless otherwise indicated, the following discussions are based on *Figure 1.*

### Function 1

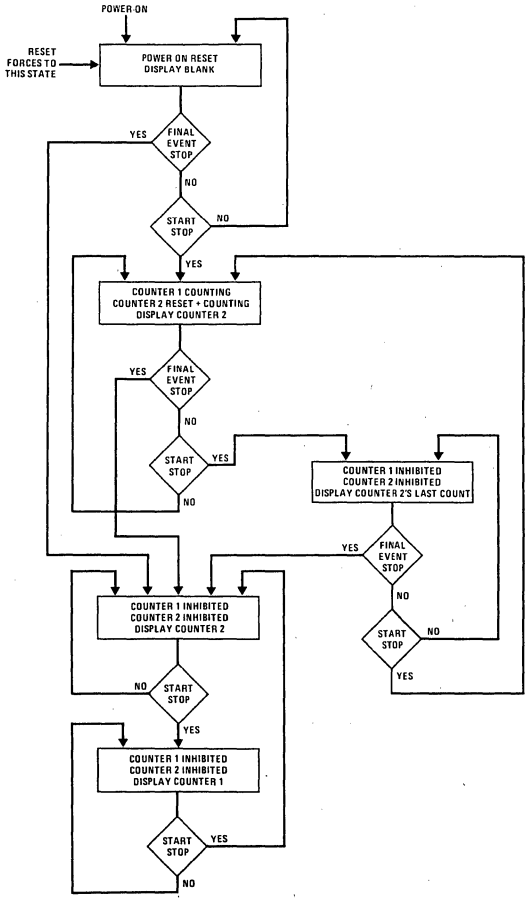
In Function 1, counters 1 and 2 count up beginning with a transition on the Start-Stop pin from $V_{DD}$ to $V_{SS}$. Counter 2 is shown counting. A second transition from $V_{DD}$ to $V_{SS}$ on the Start-Stop pin inhibits the clock pulses to counter 2, stores and displays the contents of counter 2. Counter 1 continues to count. The third transition from $V_{DD}$ to $V_{SS}$ on the Start-Stop pin resets counter 2, enables clock pulses to counter 2 and displays counter 2 counting. Subsequent Start-Stop transitions repeat this sequence, all this time counter 1 continues to count. At the conclusion of the last event to be timed, a Final Event Stop transition from $V_{DD}$ to $V_{SS}$ inhibits the clock to both counters and displays counter 2. A Start-Stop transition from $V_{DD}$ to $V_{SS}$ switches the display from counter 2 to counter 1. Repetitive Start-Stop transitions switch the display between counter 2 and counter 1.

### Function 2

In Function 2, counter 1 and 2 count up beginning with a transition on the Start-Stop pin. Counter 2 is displayed counting. A second transition on the Start-Stop pin inhibits the clock pulses to both counter 1 and counter 2, stores and displays the contents of counter 2. The third transition on the Start-Stop pin resets counter 2, enables the clock to both counters and displays counter 2 counting. Subsequent Start-Stop transitions repeat this sequence. At the conclusion of the last event to be timed, a Final Event Stop transition inhibits the clock to both counters and displays counter 2. A Start-Stop transition switches the display from counter 2 to counter 1. Repetitive Start-Stop transitions switch the display between counter 2 and counter 1. The control inputs (C1, C2) must be pulsed at Start-Stop command. Pulse width is a function of clock period.




Flow Chart for Function 1

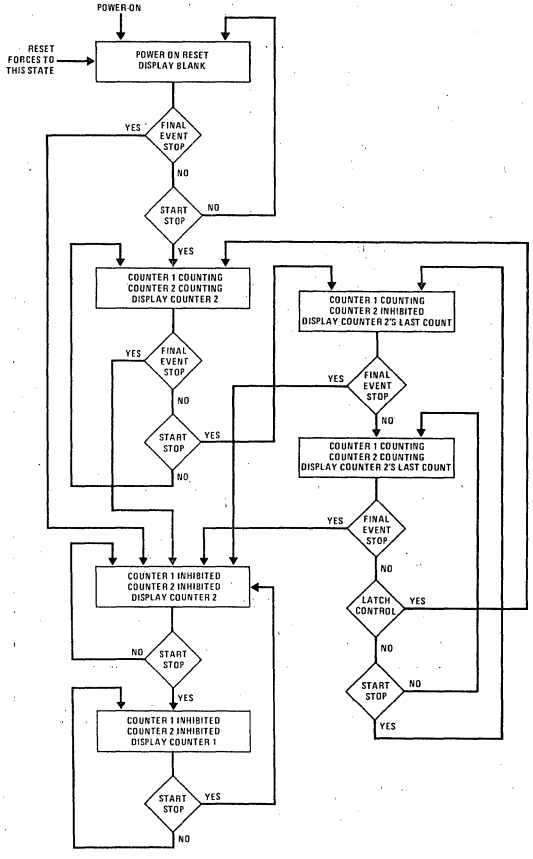



Flow Chart for Function 2

## Functional Description (cont'd)

### Function 3

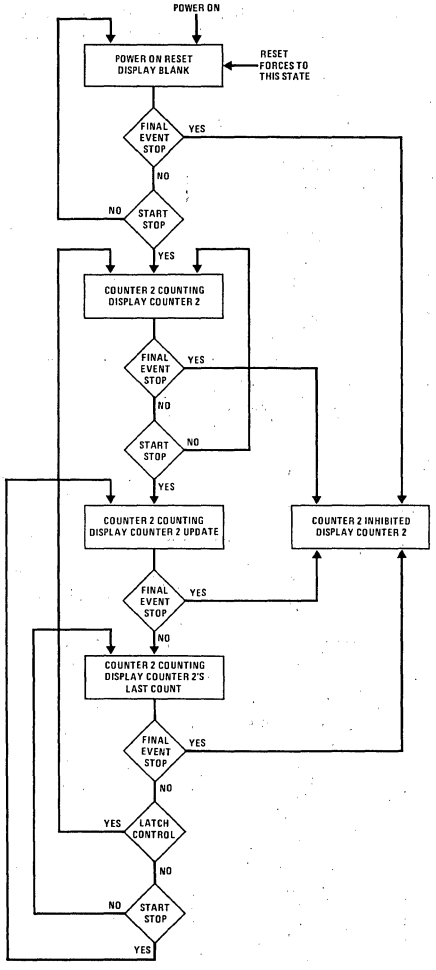
In Function 3, counter 1 and 2 count up beginning with a transition on the Start-Stop pin. Counter 2 is displayed counting. A second transition on the Start-Stop pin stores and displays the contents of counter 2, resets counter 2, and initiates a new up-count in counter 2; however, the new up-count is not displayed. Counter 1 continues to count. A transition on the Latch Control pin will display counter 2 counting until another transition on the Start-Stop pin. A Final Event Stop transition inhibits the clock pulses to both counters 1 and 2 and displays the contents of counter 2. A Start-Stop transition after the Final Event transition switches the display from counter 2 to counter 1. Repetitive Start-Stop transitions switch the display between counter 2 and counter 1.




Flow Chart for Function 3

### Function 4

In Function 4, counter 2 counts up beginning with a transition on the Start-Stop pin. Counter 2 is displayed counting. A second transition on the Start-Stop pin stores and displays the contents of counter 2. Subsequent Start-Stop transitions update the display of counter 2. A transition on the Latch Control pin will display counter 2 counting until a transition on the Start-Stop pin. A Final Event Stop transition inhibits the clock pulses to counter 2 and displays the contents of counter 2.




Flow Chart for Function 4

## Functional Description (cont'd)

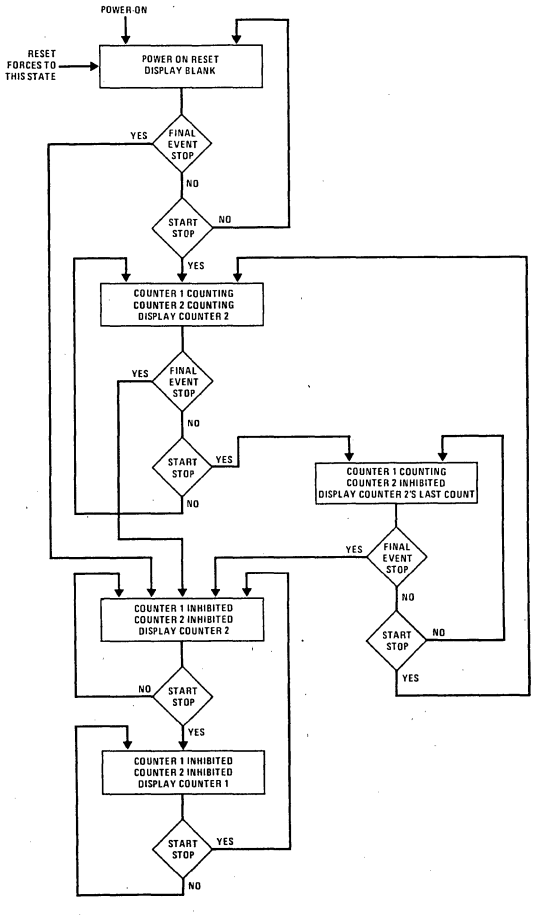
### Function 5

In Function 5, counter 1 and 2 count up beginning with a transition on the Start-Stop pin. Counter 2 is displayed counting. A second transition on the Start-Stop pin inhibits the clock pulses to counter 2, and the contents of counter 2 are displayed. Counter 1 continues counting. The third Start-Stop transition enables the clock pulses to counter 2 and counter 2 is displayed counting. Subsequent Start-Stop transitions repeat this sequence, all the time counter 1 continues counting. At the conclusion of the last event to be timed, a Final Event Stop inhibits the clock pulses to both counters 1 and 2, and displays counter 2. A Start-Stop transition switches the display from counter 2 to counter 1. Repetitive Start-Stop transitions switch the display between counter 2 and counter 1.

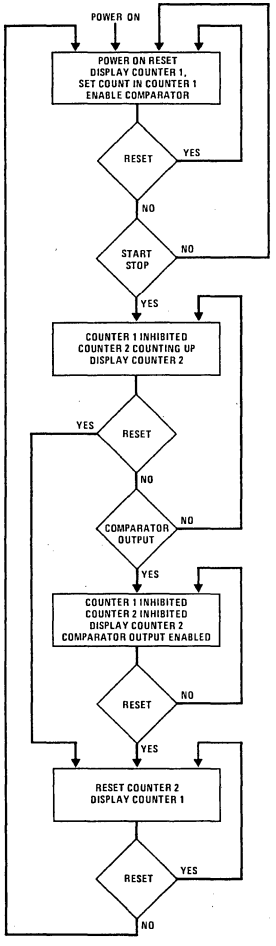
### Function 6

In Function 6, counter 1 is displayed at power-on or reset. Counter 1 is set to a specific count by Program Digit 1–4 pins. Then the comparator is enabled. Counter 2 is displayed counting up beginning with a transition on the Start-Stop pin. When counter 2 is coincident with counter 1, the clock pulses to counter 2 are inhibited, the contents of counter 2 are displayed and the Comparator Output is enabled. Upon the transition of Reset, counter 1 is again displayed with the time that was set, and the Comparator Output is disabled. Counter 1 can be reprogrammed by the Program Digit 1–4 pins if desired. A Start-Stop transition repeats the sequence.

If the Comparator Output pin is connected to the Reset pin, Automatic Reset will occur; however, this connection must be broken during digit programming.




Flow Chart for Function 5

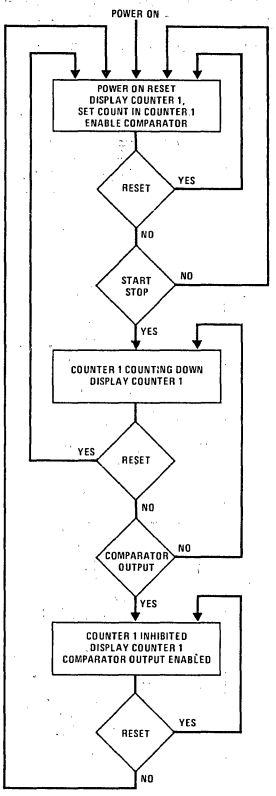



Flow Chart for Function 6

# Functional Description (cont'd)

### Function 7

In Function 7, counter 1 is displayed all the time. Counter 1 is set to a specific count by Program Digit 1–4 pins. Then the comparator and Control C1 In are enabled. Pin 4 and pin 35 must be floating or connected to $V_{DD}$ during digit programming. Counter 1 counts down from the set count beginning with a transition on the Start-Stop pin. When counter 1 counts down to zero, the clock pulses to counter 1 are inhibited and the comparator Output is enabled. This is not repeatable without setting a new count into counter 1. The comparator and Control C1 In must be inhibited and a reset pulse must occur before the new count may be entered.




Flow Chart for Function 7

### Reset

This input will reset all logic and counters in Functions 1–5 and Function 7. In Function 6, Reset will reset logic but not counter 1. Reset is internally pulled to $V_{DD}$, or a logic zero. For a reset to occur, the Reset pin must be held to $V_{SS}$, a logic one.

### Start-Stop

This input is used to control the counters. How it affects the counters is explained in each function. For Start-Stop to affect the counters, it must be held to $V_{SS}$, a logic one. Logic zero results when the pin is tied to $V_{DD}$ or left floating (internal pull-up to $V_{DD}$).

### Final Event Stop/Comparator Output

This pin is used to indicate to the circuit that no more events will be timed or counted. Final Event Stop affects the circuit when it is held to $V_{SS}$. There is an internal pull-up to $V_{DD}$. This pin is also an output pin, $V_{SS}$ indicates comparison between the two counters.

### Divide Scale Inputs

These three inputs are used to determine whether the counters will count in Modulo 6 or Modulo 10. Table I shows the code for which digit will count in Modulo 6 or Modulo 10. A logic one is when the pin is held to $V_{SS}$. When the pin is tied to $V_{DD}$ or left floating (internal pull-up to $V_{DD}$), a logic zero results.

TABLE I. Divide Scaler Code

| DIVIDE SCALER | | | COUNTER 1 | | | | COUNTER 2 | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | D4 | D3 | D2 | D1 | D4 | D3 | D2 | D1 |
| 0 | 0 | 0 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 1 | 0 | 0 | 6 | 10 | 10 | 10 | 6 | 10 | 10 | 10 |
| 0 | 1 | 0 | 10 | 6 | 10 | 10 | 10 | 6 | 10 | 10 |
| 1 | 1 | 0 | 10 | 10 | 6 | 10 | 10 | 10 | 6 | 10 |
| 0 | 0 | 1 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 1 | 0 | 1 | 10 | 10 | 10 | 10 | 6 | 10 | 10 | 10 |
| 0 | 1 | 1 | 10 | 10 | 10 | 10 | 10 | 6 | 10 | 10 |
| 1 | 1 | 1 | 10 | 10 | 10 | 10 | 10 | 10 | 6 | 10 |

### Comparator Enable

This input enables the comparator. To enable the comparator, the pin is held to $V_{SS}$ or logic one. To disable the comparator, the pin is tied to $V_{DD}$ or left floating (internal pull-up to $V_{DD}$).

### Resolution Select Inputs

These two inputs are used to select the frequency of the clock pulses to the counters, Table II shows the code for each frequency. A logic one is when the pin is held to $V_{SS}$. A logic zero results when the pin is tied to $V_{DD}$ or left floating (internal pull-up to $V_{DD}$).

TABLE II. Resolution Select Code

| RESOLUTION SELECT 1 | 2 | FREQUENCY OF CLOCK TO COUNTERS | DISPLAY RESOLUTION |
|---|---|---|---|
| 0 | 0 | 100 Hz | 0.01 sec |
| 0 | 1 | 10 Hz | 0.1 sec |
| 1 | 0 | 1 Hz | 1 sec |
| 1 | 1 | External | |

# Functional Description (cont'd)

### Clock In/Out

This pin is either an input or output depending on the code at the Resolution Select inputs. If the pin is used as an output pin, it will output the clock frequency the Resolution Select inputs have selected. When used as an input, an external clock is used to clock the counters.

### Blanking Output

This output is used to blank the display at the beginning and end of each digit time to allow for internal delay between two cascaded chips, see *Figure 3.* The display is blanked when the Blanking Output is at $V_{DD}$.

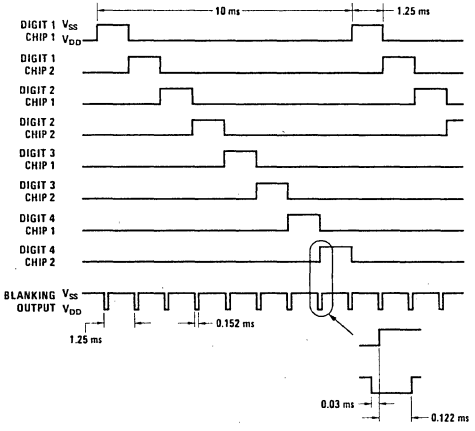



FIGURE 3. Blanking Output

### Oscillator In and Out

A quartz crystal, resonant at 32.8 kHz, two capacitors and one resistor, together with the internal MOS circuits form a crystal controlled oscillator as shown in *Figure 4.* Varying one of the capacitors allows precise frequency settings. For test purposes, OSC IN is the input and OSC OUT is the output of an inverting amplifier.

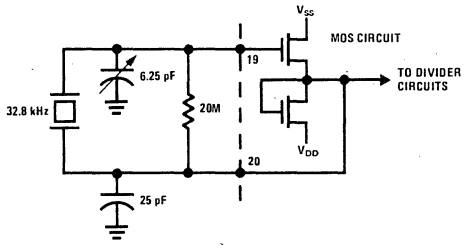



FIGURE 4. Crystal Oscillator

### Multiplex Input and Output

The Multiplex Input pin allows an external multiplex rate to be used in the chip. The multiplex rate inside the chip is one fourth the Multiplex Input and Multiplex Output rate. When using the Multiplex Input pin, the Multiplex Output pin must be tied to $V_{SS}$. The Multiplex Output pin is four times the internal multiplex rate. To use the Multiplex Output pin, the Multiplex Input pin must be tied to $V_{DD}$. The Multiplex Input must be used if the oscillator pins are not used. If the Multiplex Input pin is used, OSC IN, OSC OUT and the blanking output are not used.

### Control C1, C2 In and Control C1, C2 Out

These four input pins are used to cascade two chips together. When the Control C1 In pin is floating (internal pull-up to $V_{DD}$) or tied to $V_{DD}$, the clock pulses to counter 1 are inhibited. When Control C1 In is at $V_{SS}$, counter 1 is enabled. Control C1 Out is at $V_{SS}$ when counter 1 is at its maximum count, and it is floating at all other times. The Control C1 In pin must be floating (or connected to $V_{DD}$) while digit programming in Function 7. Control C2 pins operate on counter 2 in a similar manner.

### Program Digits 1–4

These four input pins are used to program or set any count desired in counter 1 in Functions 6 and 7. When Program Digit 1 is at $V_{SS}$, the least significant digit of counter 1 advances at a 2.5 Hz rate. There is no carry-over from digit to digit. Program Digit 1 has no effect if tied to $V_{DD}$ or left floating (internal pull-up to $V_{DD}$). Only one Program Digit input may be held to $V_{SS}$ at a time.

### Program Digit 1/Latch Control

This input has two functions; besides setting a count in digit 1 of counter 1 in Functions 6 or 7, it also affects Functions 3 and 4. In Functions 3 and 4, this input allows the display to show counter 2 counting as described in Functions 3 and 4.

### Program Digit 4/Waiting State Indicator

This input besides setting a count in digit 4 of counter 1 in Functions 6 and 7, also indicates that the chip has been reset and is in the stand-by mode at power-on. In Functions 1–5, the Waiting State Indicator is at $V_{SS}$ until a Start-Stop transition has occured. Once a Start-Stop transition has occured, the output remains at $V_{DD}$.

### Leading Zero Blanking

In Functions 1–5, leading zeros are blanked for both counters 1 and 2. In Functions 6 and 7, counter 2 has leading zero blanking. At power-on, the display is blank in Functions 1–5, and all zeros are displayed in Functions 6 and 7.

### Output Circuits

For BCD and Digit Outputs, $V_{SS}$ is a logic one. *Figure 5* illustrates the circuit used for all outputs except for Control C1, C2 Out. The Control C1, C2 Out circuit is illustrated in *Figure 6. Figure 7* illustrates the simple interface needed for an 8-digit stop-watch. *Figure 8* illustrates the MM5865 being used to count how many events occur in a specified time. *Figure 9* shows the MM5865 as a simple industrial counter when the input clock is a constant frequency above 400 Hz.

## Functional Description cont'd)

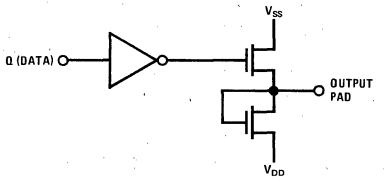



FIGURE 5. Output Circuit

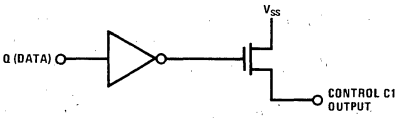



FIGURE 6. Control C1 Out Circuit

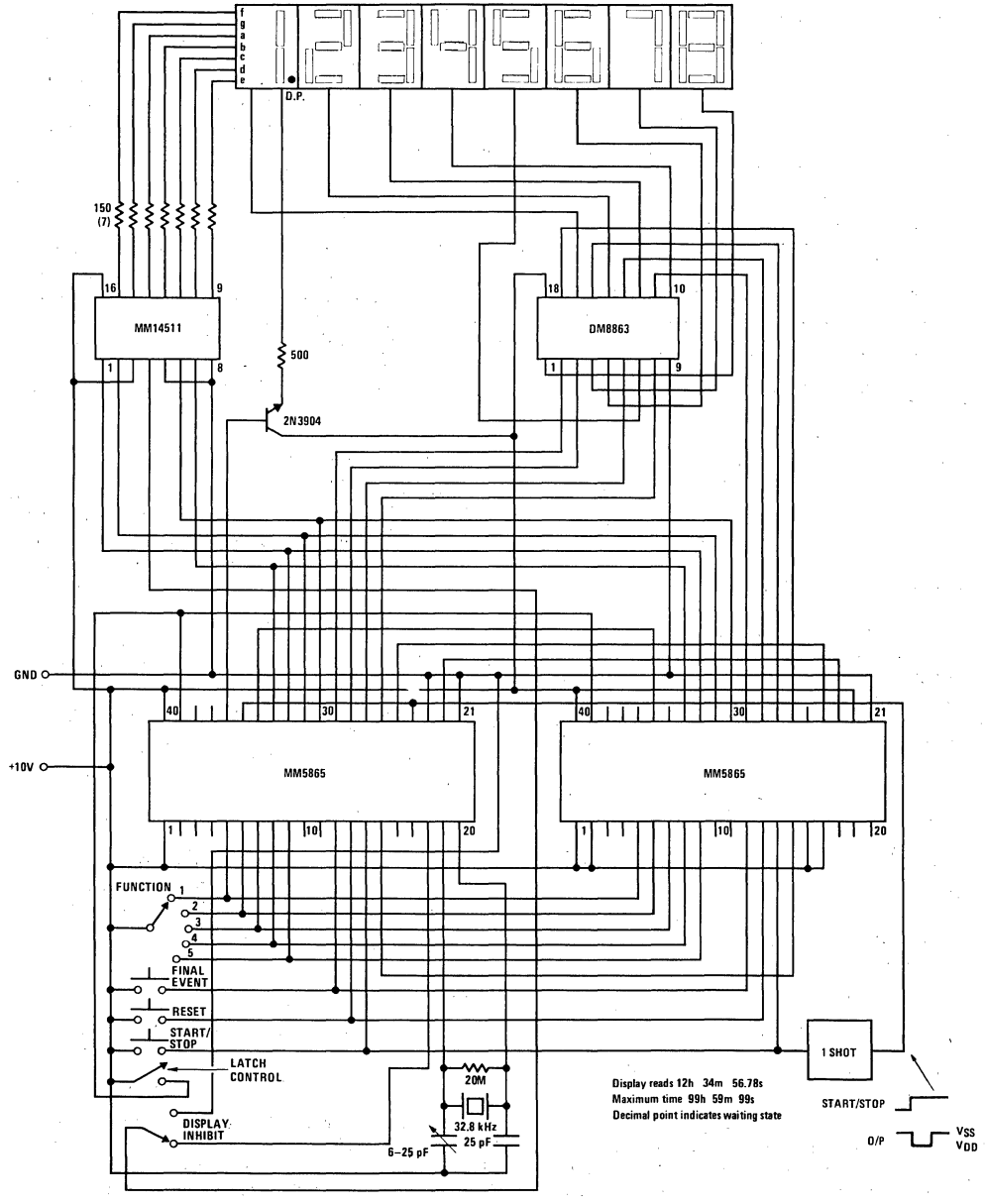



FIGURE 7. Stop Watch Application

## Functional Description (cont'd)

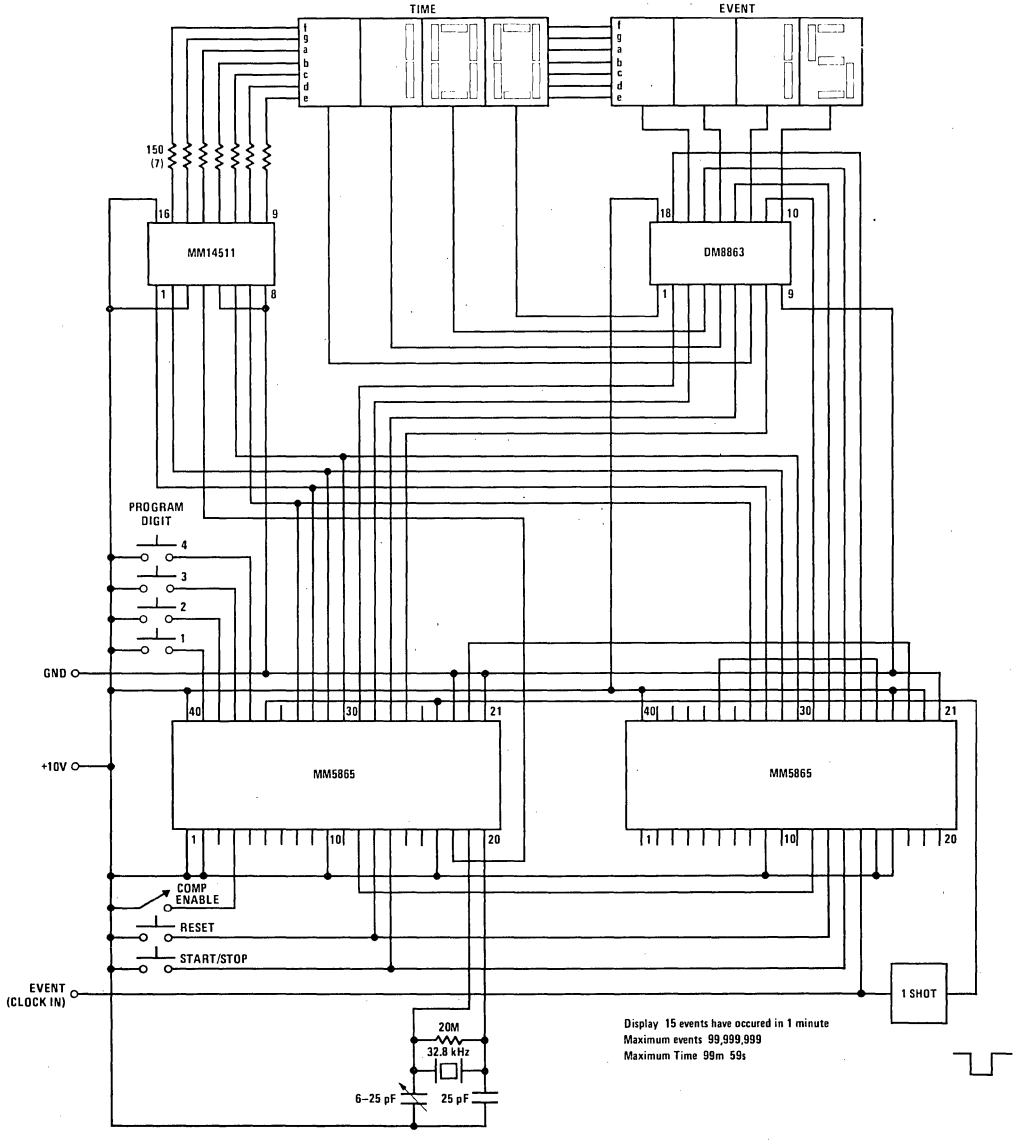



FIGURE 8. Application of MM5865 to Count Events in a Specified Time

## Functional Description (cont'd)

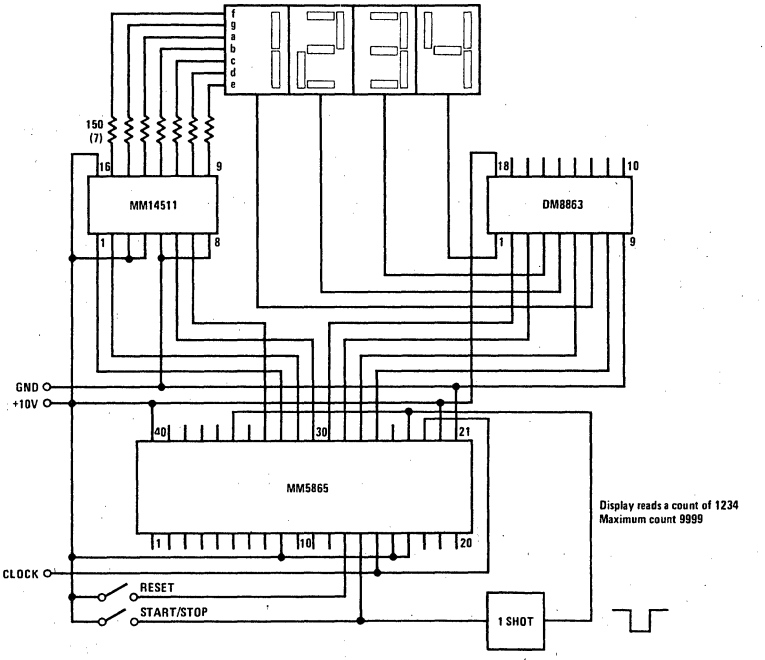



**FIGURE 9. Industrial Counter**

National Semiconductor

Electronic Data Processing

# MM58167 Microprocessor Compatible Real Time Clock

## General Description

The MM58167 is a low threshold metal-gate CMOS circuit that functions as a real time clock calendar in bus-oriented microprocessor systems. The device includes an addressable counter, addressable latch for alarm-type functions, and 2 interrupt outputs. A power-down input allows the chip to be disabled from the outside world for standby low power operation. The time base is generated from a 32,768 Hz crystal-controlled oscillator.

## Features

- Microprocessor compatible
- Thousandths of seconds, hundredths of seconds, tenths of seconds, seconds, minutes, hours, day of the week, day of the month, and month counters with corresponding latches for alarm-type functions
- Interrupt output (maskable) with 8 possible interrupt signals:
  - Latch and counter comparison
  - Every tenth of a second
  - Every second
  - Every minute
  - Every hour
  - Every day
  - Every week
  - Every month
- Power-down mode that disables all outputs except for an interrupt output that occurs on a counter latch comparison. This is not the same as the maskable interrupt output
- Don't care states in the latches
- Status bit to indicate clock rollover during a read
- 32,768 Hz crystal reference, with only the input tuning capacitor and load capacitor needed externally
- Four year calendar

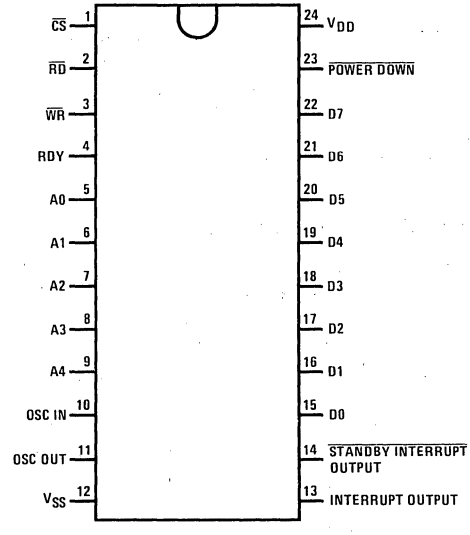
## Functional Description

The MM58167 is a microprocessor oriented real time clock. The circuit includes addressable real time counters and addressable latches, each for thousandths of seconds through months. The counter and latch are divided into bytes of 4 bits each. When addressed, 2 bytes will appear on the data I/O bus. The data, in binary coded decimal, can be transferred to and from the counters via the data I/O bus so that each set of 2 bytes (1 word) can be accessed independently as grouped in Table I.

If either of the bytes in the above 8-bit counter words do not legally reach 4-bit lengths (e.g., day of the week uses only the 3 least significant bits) the unused bits will be unrecognized during a write and held at $V_{SS}$ during a read. If any illegal data is entered into the counters during a write cycle, it may take up to 4 clocks (4 months in the case of the month counter) to restore legal BCD data to the counter during normal counting. The latches will read and write all 4 bits per byte. Each of the counter and latch words can be reset with the appropriate address and data inputs. The counter reset is a write function. The latches can be programmed to compare with the counters at all times by writing 1's into the 2 most significant bits of each latch, thus establishing a don't care state in the latch. The don't care state is programmable on the byte level, i.e., tens of hours can contain a don't care state, yet unit hours can contain a valid code necessary for a comparison.

## Connection Diagram

**Dual-In-Line Package**




TOP VIEW

**Order Number MM58167N**
**See Package 22**

5

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at All Inputs and Outputs | $V_{DD}$ + 0.3 to $V_{SS}$ − 0.3 |
| Operating Temperature | −25°C to +85°C |
| Storage Temperature | −65°C to +150°C |
| $V_{DD} - V_{SS}$ | 6V |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics $T_A = -25°C$ to $+85°C$, $V_{SS} = 0V$

| PARAMETER | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| Supply Voltage | | | | | |
| $V_{DD}$ | Outputs Enabled | 4.0 | | 5.5 | V |
| $V_{DD}$ (Note 1) | Power Down Mode | 2.0 | | 5.5 | V |
| Supply Current | | | | | |
| $I_{DD}$, Static | Outputs TRI-STATE, $f_{IN}$ = DC, $V_{DD}$ = 5.5V | | | 10 | μA |
| $I_{DD}$, Dynamic | Outputs TRI-STATE, $f_{IN}$ = 32 kHz, $V_{DD}$ = 5.5V, $V_{IH} \geq V_{DD} - 0.3V$, $V_{IL} \leq V_{SS} + 0.3V$ | | | 20 | μA |
| $I_{DD}$, Dynamic | Outputs TRI-STATE, $f_{IN}$ = 32 kHz, $V_{DD}$ = 5.5V, $V_{IH}$ = 2.0V, $V_{IL}$ = 0.8V | | | 12 | mA |
| Input Voltage | | | | | |
| Logical Low | | 0.0 | | 0.8 | V |
| Logical High | | 2.0 | | $V_{DD}$ | V |
| Input Leakage Current | $V_{SS} \leq V_{IN} \leq V_{DD}$ | | | 1 | μA |
| Output Impedance | (I/O and Interrupt Output) | | | | |
| Logical Low | $V_{DD}$ = 4.75V, $I_{OL}$ = 1.6 mA | | | 0.4 | V |
| Logical High | $V_{DD}$ = 4.75V, $I_{OH}$ = −400 μA, | 2.4 | | | V |
| | $I_{OH}$ = −10 μA | 0.8 $V_{DD}$ | | | V |
| TRI-STATE® | $V_{OUT}$ = 0V, | | | −1 | μA |
| | $V_{OUT}$ = $V_{DD}$ | | | 1 | μA |
| Output Impedance | (Ready and Standby Interrupt Output) | | | | |
| Logical Low, Sink | $V_{DD}$ = 4.75V, $I_{OL}$ = 1.6 mA | | | 0.4 | V |
| Logical High, Leakage | $V_{OUT} \leq V_{DD}$ | | | 10 | μA |

**Note 1:** To insure that no illegal data is read from or written into the chip during power up, the power down input should be enabled only after all other lines (Read, Write, Chip Select, and Data Bus) are valid.

## Functional Description (Continued)

TABLE I

| COUNTER ADDRESSED | UNITS D0 | D1 | D2 | D3 | MAX USED BCD CODE | TENS D4 | D5 | D6 | D7 | MAX USED BCD CODE |
|---|---|---|---|---|---|---|---|---|---|---|
| Ten Thousandths of a Second | 0 | 0 | 0 | 0 | 0 | I/O | I/O | I/O | I/O | 9 |
| Tenths and Hundredths of Seconds | I/O | I/O | I/O | I/O | 9 | I/O | I/O | I/O | I/O | 9 |
| Seconds | I/O | I/O | I/O | I/O | 9 | I/O | I/O | I/O | 0 | 5 |
| Minutes | I/O | I/O | I/O | I/O | 9 | I/O | I/O | I/O | 0 | 5 |
| Hours | I/O | I/O | I/O | I/O | 9 | I/O | I/O | 0 | 0 | 2 |
| Day of the Week | I/O | I/O | I/O | 0 | 7 | 0 | 0 | 0 | 0 | 0 |
| Day of the Month | I/O | I/O | I/O | I/O | 9 | I/O | I/O | 0 | 0 | 3 |
| Month | I/O | I/O | I/O | I/O | 9 | I/O | 0 | 0 | 0 | 1 |

## Functional Description (Continued)

TABLE II. ADDRESS CODES AND FUNCTIONS

| A4 | A3 | A2 | A1 | A0 | FUNCTION |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | Counter — Thousandths of Seconds |
| 0 | 0 | 0 | 0 | 1 | Counter — Hundredths and Tenths of Seconds |
| 0 | 0 | 0 | 1 | 0 | Counter — Seconds |
| 0 | 0 | 0 | 1 | 1 | Counter — Minutes |
| 0 | 0 | 1 | 0 | 0 | Counter — Hours |
| 0 | 0 | 1 | 0 | 1 | Counter — Day of the Week |
| 0 | 0 | 1 | 1 | 0 | Counter — Day of the Month |
| 0 | 0 | 1 | 1 | 1 | Counter — Months |
| 0 | 1 | 0 | 0 | 0 | Latches — Thousandths of Seconds |
| 0 | 1 | 0 | 0 | 1 | Latches — Hundredths and Tenths of Seconds |
| 0 | 1 | 0 | 1 | 0 | Latches — Seconds |
| 0 | 1 | 0 | 1 | 1 | Latches — Minutes |
| 0 | 1 | 1 | 0 | 0 | Latches — Hours |
| 0 | 1 | 1 | 0 | 1 | Latches — Day of the Week |
| 0 | 1 | 1 | 1 | 0 | Latches — Day of the Month |
| 0 | 1 | 1 | 1 | 1 | Latches — Months |
| 1 | 0 | 0 | 0 | 0 | Interrupt Status Register |
| 1 | 0 | 0 | 0 | 1 | Interrupt Control Register |
| 1 | 0 | 0 | 1 | 0 | Counter Reset |
| 1 | 0 | 0 | 1 | 1 | Latch Reset |
| 1 | 0 | 1 | 0 | 0 | Status Bit |
| 1 | 0 | 1 | 0 | 1 | "GO" Command |
| 1 | 0 | 1 | 1 | 0 | Standby Interrupt |
| 1 | 1 | 1 | 1 | 1 | Test Mode |

All others unused.

TABLE III. COUNTER AND LATCH RESET FORMAT

| D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | COUNTER OR LATCH RESET |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Thousandths of Seconds |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | Hundredths and Tenths of Seconds |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | Seconds |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | Minutes |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Hours |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Days of the Week |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Days of the Month |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Months |

FOR COUNTER RESET A4–A0 MUST BE 10010

FOR LATCH RESET A4–A0 MUST BE 10011

## Functional Description (Continued)

Following a read of any real time counter a status bit read should be done. If during a counter read cycle the clock rolls over, the data read out could be invalid. Thus, during a read if the clock rolls over the status bit will be set. The status bit will appear on D0 when read, D1 through D7 will be zeros.

To synchronize the clock with real time a "GO" command exists which can be used to reset the thousandths of seconds, hundredths and tenths of seconds, and seconds counters. After setting the lower frequency counters (minutes through months), the appropriate address and a write pulse can be sent to reset all counters mentioned above. This allows the clock to be started at an exactly known time. It can also be used as a stopwatch function. The "GO" command is the start and a counter read is the stop point. The clock does not stop during or following a read, so each read would be a split time.

A second special command will enable the standby interrupt output. The standby interrupt output is the only input or output enabled during the power down or standby mode. Power down occurs when the power down input goes to a logical zero level. In this mode the outputs are TRI-STATED and the inputs ignored regardless of the state of the chip select. The standby interrupt is enabled by writing a 1 on the D0 line with the standby interrupt address selected. On the next counter-latch comparison the open drain output device turns on, sinking current. The output will be turned on immediately upon writing a 1 on D0 if the comparison occurred before the write, yet is still in effect. To disable the output a zero on D0 is written at the standby interrupt address. The write cycles must occur during normal operation, but the output can become active during power down. This feature can be used to turn the power back on during a power down mode (see *Figure 4* for a typical application). Refer to Tables II and III for the address input codes and functions and for the counter and latch reset format.

The interrupt output is controlled by the interrupt status register (8 bits) and the interrupt control register (8 bits). The status register contains the present state of the comparator (compares the counters and latches) and the outputs (1 bit each) of the tenths of seconds, seconds, minutes, hours, week, day of the month, and month counters *(Figure 1)*. The interrupt status register can only be read. The interrupt control register is a mask register that regulates which of the 8 bits in the status register goes out as an interrupt. The control register cannot be read from. A 1 is written into the control register to select the appropriate interrupt output. If more than a single 1 exists in the control register each selected bit will come out as an interrupt. This will appear as an interrupt occurring at the highest frequency selected. The interrupt is acknowledged by addressing and reading the status register. Once acknowledged the interrupt output and status register are reset. The only way to disable the interrupt output is to write all 0's into the control register or to enable the power down input.

The I/O bus is controlled by the read, write, ready and chip select lines. During a read cycle ($\overline{RD}$ = 0, $\overline{WR}$ = 1, $\overline{CS}$ = 0, RDY = 0) the data on the I/O bus is the data contained in the addressed counter or latch. During a write cycle ($\overline{RD}$ = 1, $\overline{WR}$ = 0, $\overline{CS}$ = 0, RDY = 0) the data on the I/O bus is latched into the addressed counter or latch. At the start of each read or write cycle the RDY signal goes low and will remain low until the clock has placed valid data on the bus or until it has completed latching data in on a write. The chip select line is used to enable or disable the device outputs. When the chip is selected the device will drive the I/O bus for a read or use the I/O bus as an input for a write. The I/O bus will not be affected when the chip is deselected. The outputs driving the bus will go to the TRI-STATE or high impedance state. The chip will not respond to any inputs when deselected. Refer to *Figures 2 and 3* for read and write cycle timing.

The clock's time base is a 32,768 crystal controlled oscillator. Externally, the crystal, the input tuning capacitor, and the output load capacitor are required. Included internally are a high gain inverter, an RC delay, and the bias resistor. To tune the oscillator a constant read can be done on one of the higher frequency counters. For example, a constant read of the thousandths of seconds counter will place an average 500 Hz signal on the D4 bus line. The period varies slightly due to disable of latches during counter roll.

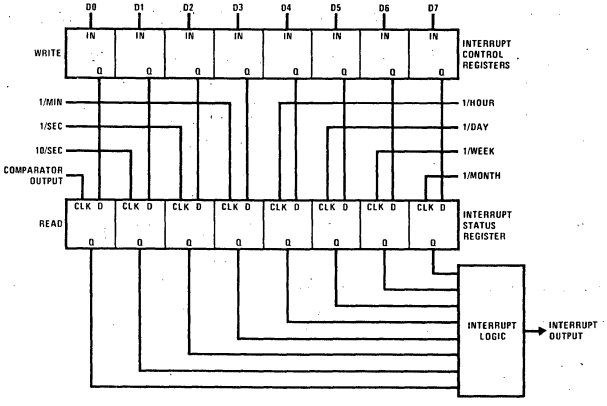



**FIGURE 1. Interrupt Register Format**

## Read Cycle Timing Characteristics $T_A = -25°C$ to $+85°C$, $V_{DD} = 4.0V$ to $5.5V$, $V_{SS} = 0V$

| | PARAMETER | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| $t_{AR}$ | Address Bus Valid to Read Strobe | 100 | | | ns |
| $t_{CSR}$ | Chip Select ON to Read Strobe | 0 | | | ns |
| $t_{RRY}$ | Read Strobe to Ready Strobe | | | 150 | ns |
| $t_{RYD}$ | Ready Strobe to Data Valid | | | 800 | ns |
| $t_{AD}$ | Address Bus Valid to Data Valid | | | 1050 | ns |
| $t_{RH}$ | Data Hold Time from Trailing Edge of Read Strobe | 0 | | | ns |
| $t_{HZ}$ | Trailing Edge of Read Strobe to TRI-STATE Mode | | | 250 | ns |
| $t_{RYH}$ | Read Hold Time After Ready Strobe | 0 | | | ns |
| $t_{RA}$ | Address Bus Hold Time from Trailing Edge of Read Strobe | 50 | | | ns |

Data bus loading is 100 pF
Ready output loading is 50 pF
Input and output AC timing levels are:
Logical "1" = 2.0V
Logical "0" = 0.8V

## Write Cycle Timing Characteristics $T_A = -25°C$ to $+85°C$, $V_{DD} = 4.0V$ to $5.5V$, $V_{SS} = 0V$

| | PARAMETER | MIN | TYP | MAX | UNITS |
|---|---|---|---|---|---|
| $t_{AW}$ | Address Valid to Write Strobe | 100 | | | ns |
| $t_{CSW}$ | Chip Select ON to Write Strobe | 0 | | | ns |
| $t_{DN}$ | Data Valid Before Write Strobe | 100 | | | ns |
| $t_{WRY}$ | Write Strobe to Ready Strobe | | | 150 | ns |
| $t_{RY}$ | Ready Strobe Width | | | 800 | ns |
| $t_{RYH}$ | Write Hold Time After Ready Strobe | 0 | | | ns |
| $t_{WD}$ | Data Hold Time After Write Strobe | 110 | | | ns |
| $t_{WA}$ | Address Hold Time After Write Strobe | 50 | | | ns |

Data bus loading is 100 pF
Ready output loading is 50 pF
Input and output AC timing levels are:
Logical "1" = 2.0V
Logical "0" = 0.8V

## Switching Time Waveforms

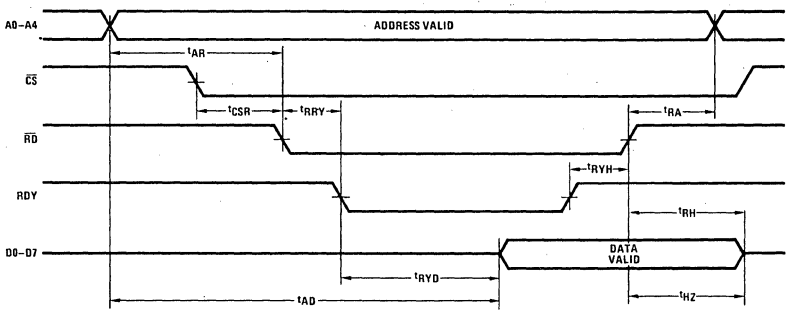



**FIGURE 2. Read Cycle Waveforms**

## Switching Time Waveforms (Continued)

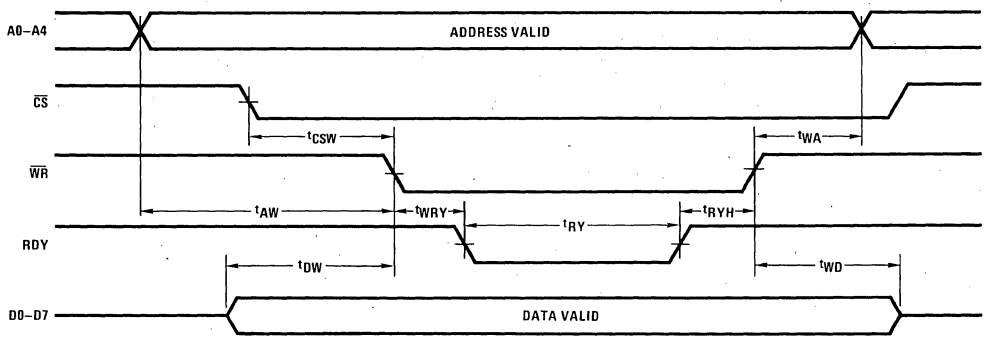



FIGURE 3. Write Cycle Waveforms

## Typical Application

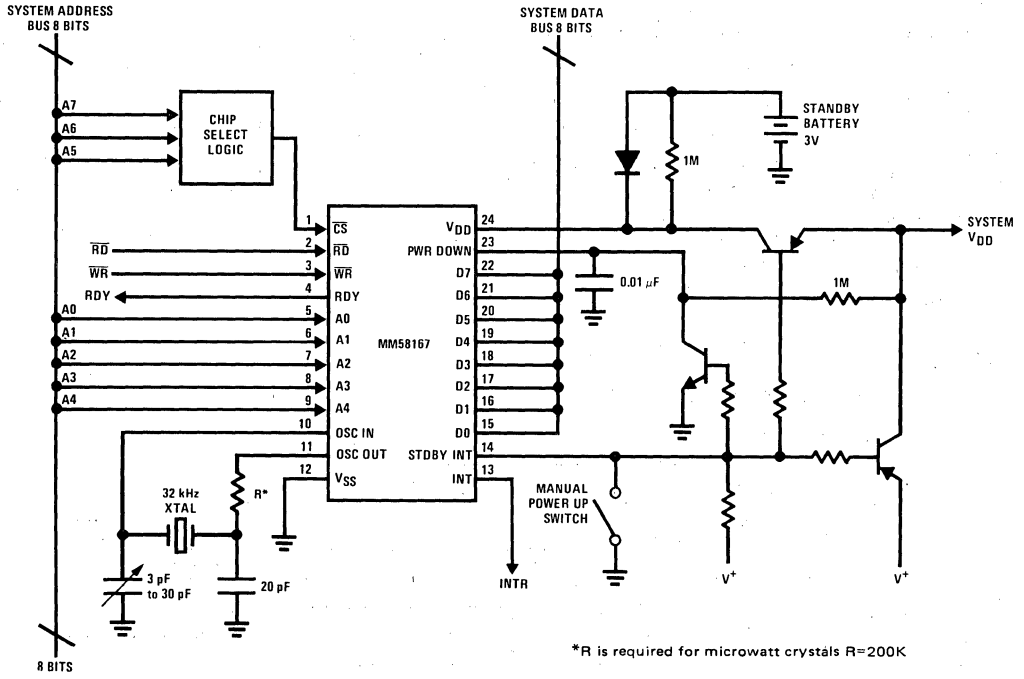



FIGURE 4. Standby Interrupt is Enabled (ON) for Normal Operation and Disabled for Standby Operation

National Semiconductor

# MM58174 Microprocessor-Compatible Real-Time Clock

## General Description

The MM58174 is a low threshold metal gate CMOS circuit that functions as a real-time clock and calendar in bus-oriented microprocessor systems. The device includes an interrupt timer which may be programmed to one of three times. Time-keeping is maintained down to 2.2V to allow low power standby battery operation. The timebase is generated from a 32768 Hz crystal controlled oscillator.

## Features

- Microprocessor-compatible
- Tenths of seconds, seconds, tens of seconds, minutes, tens of minutes, day of week, days, tens of days, months, tens of months, independent registers
- Automatic leap year calculation
- Internal pull-ups to safeguard data
- Protection for read during data changing
- Fast access time (500 ns)
- Independent interrupt system with free drain output
- TTL compatible
- Low power standby operation (2.2V, 10 $\mu$A)
- Low cost internally biased oscillator
- Low cost 16-pin dual-in-line package

## Applications

- Point of sale terminals
- Word processors
- Teller terminals
- Event recorders
- Microprocessor controlled instrumentation
- Microprocessor time clock
- TV/VCR reprogramming
- Intelligent telephone

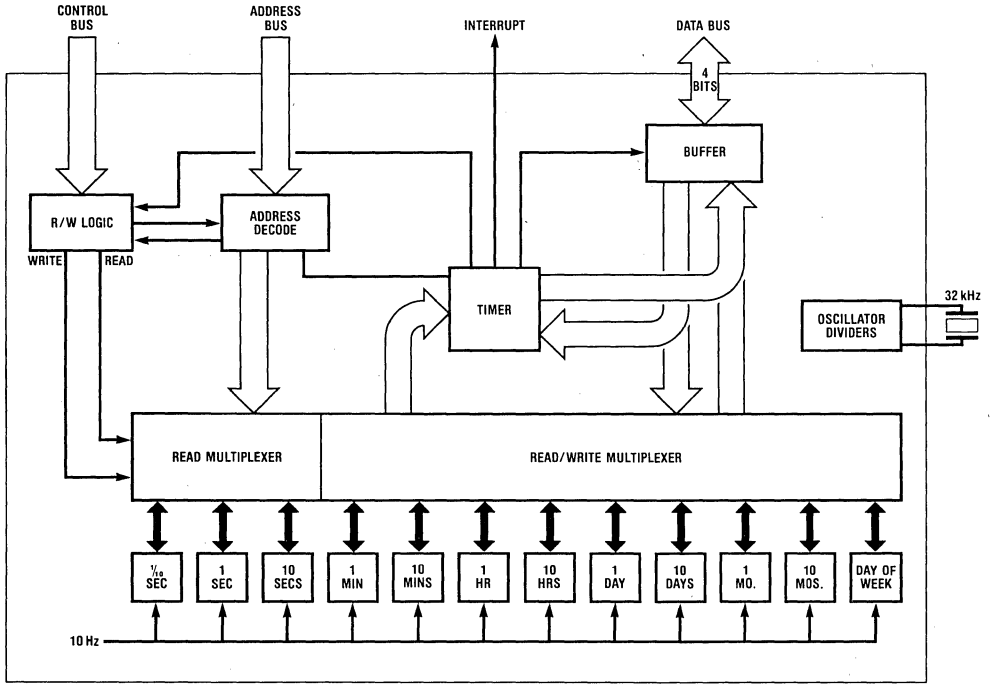



**Figure 1. Block Diagram**

## Absolute Maximum Ratings

| | |
|---|---|
| Voltage at All Inputs and Outputs | $V_{DD} + 0.3$ to $V_{SS} - 0.3$ |
| Operating Temperature | 0°C to 70°C |
| Storage Temperature | −65°C to +150°C |
| $V_{DD} - V_{SS}$ | 6.5V |
| Lead Temperature (Soldering, 10 seconds) | 300°C |

## Electrical Characteristics $T_A = 0°C$ to $+70°C$, $V_{SS} = 0V$

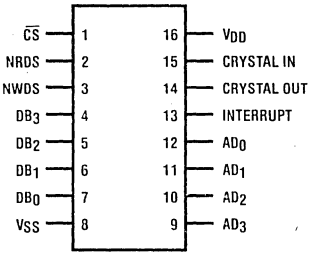
| Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| Supply Voltage, $V_{DD}$ | Stand-by mode (no READ or WRITE instructions) | 2.2 | | | V |
| Supply Current, $I_{DD}$ | Operational mode | 4.0 | | 6.0 | V |
| | $V_{DD} = 2.5V$ | | 4.0 | 10 | μA |
| | $V_{DD} = 5.0V$ | | 1.0 | | mA |
| Input Logic Levels | $V_{DD} = 5.0V$ | | | | |
| for signals: $AD_0$-$AD_3$, $DB_0$-$\overline{DB_3}$, NWDS, NRDS, $\overline{CS}$ | | | | | |
| Logic "1" | | 2.0 | | | V |
| Logic "0" | | | | 0.8 | V |
| Input Capacitance | | | | 10 | pF |
| Input Current Levels | $V_{DD} = 5.0V$ | | | | |
| Current to $V_{SS}$ for signals: $AD_0$-$AD_3$, $DB_0$-$DB_3$, NRDS | $V_{IN} = V_{DD}$ | | | 20 | μA |
| Internal Resistor to $V_{DD}$ for signals: | | | | | |
| NWDS | | 50 | 100 | | kΩ |
| $\overline{CS}$ | | 10 | 100 | | kΩ |
| Output Logic Levels | $V_{DD} = 5V$ | | | | |
| for signals: $DB_0$-$DB_3$ | | | | | |
| Logic "1" | $I_{OH} = 0.1$ mA | 2.4 | | | V |
| Logic "0" | $I_{OL} = 1.6$ mA | | | 0.4 | V |
| INTERRUPT | | | | | |
| Logic "0" | For $I_{DS} = 5$ mA | | | 1.0 | V |
| Off Leakage | $V_{OUT} = 5V$ | | | 5 | μA |

## Functional Description

The MM58174 is a microprocessor bus-oriented real-time clock. The circuit includes addressable real-time counters for tenths of seconds through months and a write only register for leap year calculation. The counters are arranged as bytes of four bits each. When addressed a byte will appear on the data I/O bus so that each word can be accessed independently. If any byte does not contain four bits (e.g. days of the week uses only 3 bits), the unused bits will be unrecognized during a write operation and tied to $V_{SS}$ during a read operation.

The addressable reset latch causes the pre-scaler, tenths of seconds, seconds, and tens of seconds to be held in a reset condition. If a register is updated during a read operation the I/O data is prevented from updating and a subsequent read will return the illegal b.c.d. code '1111'. The interrupt timer may be programmed for intervals of 0.5 second, 5 seconds, or 60 seconds and may be coded as a single or repeated operation. The open drain interrupt output is pulled to $V_{SS}$ when the timer times out and reading the interrupt register provides the status and internal selected information.

## Connection Diagram




Order Number MM58174N
See Package 19

## Circuit Description

The block diagram shown in Figure 1 shows the structure of the CMOS clock chip. A 16-pin DIL package is used.

### Crystal Oscillator

This consists of a CMOS inverter/amplifier with on-chip bias resistor and capacitors. A single 6-36 pF trimmer is all that is required to fine tune the crystal (see Figure 2). The output of the oscillator is blocked by the start/stop F/F.

### Non-Integer Divider

This counter divides the incoming 32,768 Hz frequency by 15/16 down to 30,720 Hz.

### Fixed Divider (512)

This is a standard 9 stage binary ripple counter. Output frequency is 60 Hz. This counter is reset to zero by start/stop F/F.

### Fixed Divider (6)

This is a three stage Johnson counter with a 10 Hz output signal. This counter is reset to zero state by the start/stop F/F.

### Synchronization Stage

Both 10 Hz and 32,768 Hz clocks are fed into this section. It is used to generate a pulse of 15.25 s width on the rising edge of each 10 Hz pulse.

This pulse is used to increment all the seconds, minutes, hours, days, months, and year counter and also to set the data changed F/F.

### Data Changed F/F

This is set by the rising edge of each 10 Hz pulse to indicate that the clock value has changed since the last read operation. It is reset by any clock read command. The flip flop sets all data bus bits to a "1" during NRDS time indicating that a register has been updated.

### Seconds Counters

There are three counters for the seconds:

a) tenths of seconds

b) units of seconds

c) tens of seconds

The outputs of all three counters can be separately multiplexed on to the command 4-bit output bus. Table 1 shows the address decoding for each counter. All three counters are reset to zero by the start/stop F/F.

### Minutes Counters

There are two Minutes counters:

a) units of minutes

b) tens of minutes

Both counters are parallel loaded with data from the 4-bit input bus when addressed by the microprocessor and a Write Data Strobe pulse given. Similarly, the output of both counters can be read separately onto the common 4-bit output bus (Table 1).

### Hours Counters

There are two Hours counters which will count in a 24 hour mode:

a) units of hours

b) tens of hours

Both counters have identical parallel load and read multiplex features to the Minutes counters.

### Seven Day Counter

There is a seven state counter which increments every 24 hours. It will have identical parallel load and read multiplex capabilities to the Minutes and Hours counters. The counter counts cyclically from 1-7.

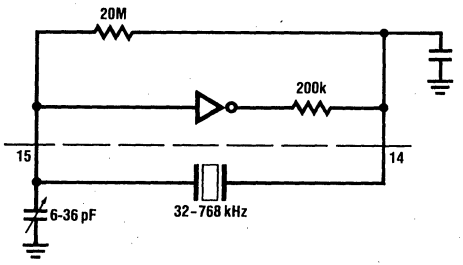



Figure 2. Crystal Oscillator

### Days Counter

There are two Days Counters:

a) units of days

b) tens of days

The Days counters will count up to 28, 29, 30, or 31 days depending on the state of the Months counters and the Years Status Register. Days counters have parallel load and read multiplex capabilities.

### Months Counters

There are two Months counters:

a) units of months

b) tens of months

The Months counters have parallel load and read multiplex capabilities.

### Years Status Register

The Years Status register is a shift register of 4 bits. It will be shifted every year on December 31st. The status register must be set in accordance with Table 3. No readout capability is provided.

### Chip Select (CS)

An external chip select is provided. The chip enable is active low.

### Counter and Register Selection

Table 1 shows the coding on the address lines $AD_0$–$AD_3$ which select the registers in the circuit to be either parallel loaded or read on to the output bus.

### Interrupt Output

An exclusive address selects the interrupt latches (address 15). These latches enable the interrupt output and dictate the frequency of the interrupt as shown in Table 2. The interrupt output flip flop is reset by reading the interrupt register. Writing $DB_3$ at chip address 15 (F) selects single or repeated interrupt.

The contents of the interrupt register are read onto the data bus by reading the interrupt status of the circuit. Table 2 gives the interrupt bits corresponding to data bus bits. $DB_3$ indicates that an interrupt has occurred. The trailing edge of the NRDS pulse that reads the interrupt status automatically reset $DB_3$ to zero. The next

Table 1. Address Decoding for Internal Registers

| Selected Counter | Address Bits $AD_3$ | $AD_2$ | $AD_1$ | $AD_1$ | Mode |
|---|---|---|---|---|---|
| 0 Test Only | 0 | 0 | 0 | 0 | Write only |
| 1 Tenths of secs. | 0 | 0 | 0 | 1 | Read only |
| 2 Units of secs. | 0 | 0 | 1 | 0 | Read only |
| 3 Tens of secs. | 0 | 0 | 1 | 1 | Read only |
| 4 Units of mins. | 0 | 1 | 0 | 0 | Read or Write |
| 5 Tens of mins. | 0 | 1 | 0 | 1 | Read or Write |
| 6 Units of hours | 0 | 1 | 1 | 0 | Read or Write |
| 7 Tens of hours | 0 | 1 | 1 | 1 | Read or Write |
| 8 Units of days | 1 | 0 | 0 | 0 | Read or Write |
| 9 Tens of days | 1 | 0 | 0 | 1 | Read or Write |
| 10 Day of week | 1 | 0 | 1 | 0 | Read or Write |
| 11 Units of months | 1 | 0 | 1 | 1 | Read or Write |
| 12 Tens of months | 1 | 1 | 0 | 0 | Read or Write |
| 13 Years | 1 | 1 | 0 | 1 | Write Only |
| 14 Stop/Start | 1 | 1 | 1 | 0 | Write Only |
| 15 Interrupt & Status | 1 | 1 | 1 | 1 | Read or Write |

Table 2a. Interrupt Selection Data

| Mode: Address 15, Write Mode | | | | |
|---|---|---|---|---|
| **Function** | **$DB_3$** | **$DB_2$** | **$DB_1$** | **$DB_0$** |
| No Interrrupt | X | 0 | 0 | 0 |
| Int. at 6.0 sec. intervals* | 0/1 | 1 | 0 | 0 |
| Int. at 5.0 sec. intervals* | 0/1 | 0 | 1 | 0 |
| Int. at 0.5 sec. intervals* | 0/1 | 0 | 0 | 1 |

* ±16.6 ms

$DB_3 = 0$, single interrupt $DB_3 = 1$, repeated interrupt

Table 2b. Interrupt Read Back (Status)

| Mode: Address 15, Read Mode | | | | |
|---|---|---|---|---|
| **Interrupt Status** | **$DB_3$** | **$DB_2$** | **$DB_1$** | **$DB_0$** |
| Reset | 0 | 0 | 0 | 0 |
| 60 sec. signal | 0/1 | 1 | 0 | 0 |
| 5.0 sec. signal | 0/1 | 0 | 1 | 0 |
| 0.5 sec. signal | 0/1 | 0 | 0 | 1 |

$DB_3 = 0$, no interrupt $DB_3 = 1$, interrupt from timer

Table 3 Years Status Register

| Mode: Address 13, Write Mode | | | | |
|---|---|---|---|---|
| | **$DB_3$** | **$DB_2$** | **$DB_1$** | **$DB_0$** |
| Leap year | 1 | 0 | 0 | 0 |
| Leap year + 1 | 0 | 1 | 0 | 0 |
| Leap year + 1 | 0 | 0 | 1 | 0 |
| Leap year + 1 | 0 | 0 | 0 | 1 |

system NRDS pulse after that which has read the interrupt status automatically restarts the interrupt timer if in continuous mode.

When $DB_3$ is set to zero at chip address 15 (F) the timer is reset at the completion of the selected timing period and must be set by software if a subsequent interrupt is required. Setting $DB_3$ to 1 allows automatic repeated timer interrupts, starting after the next system NRDS pulse after that which has read the interrupt register.

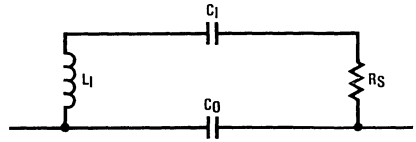
Interrupt should be initialized by applying the reset condition and reading three times at address 15 (F).

### Start/Stop

A logic "1" on $DB_0$ at chip address 14 (E) will start the clock running, a logic "0" will stop the clock. This function allows the loading of time data into the clock and its precise starting.

### Test Mode

This mode is incorporated to facilitate production testing of the circuit. For normal operation, the circuit must be set to the non-test mode as part of the system initialization. This is accomplished by writing a logic "0" to $DB_3$ at $AD_0$.




$C_I = 0.003\,pF$ $R_S \leq 35\,k\Omega$
$C_O = 3.0\,pF$ $Q \leq 40\,k\Omega$

**Figure 3. Typical Crystal Parameters**

**Table 4. Timing: Data from Peripheral to Microprocessor**

| Symbol | Parameter | Min. | Typ. | Max. | Units | Comm. |
|---|---|---|---|---|---|---|
| $t_{ACS0}$ | Address. Bus Valid to Chip Select ON (CS = 0) | | 40 | | ns | $V_{DD} = 5V$ |
| $t_{CSR}$ | Chip Select ON to Read Strobe | 70 | | | ns | |
| $t_{RD}$ | Read Cycle Access Time from Read Strobe to Data Bus Valid | | 450 | 500 | ns | CL = 100 pF |
| $t_{RH}$ | Data hold time from trailing edge of Read Stobe | 0 | | 250 | ns | |
| $t_{RA}$ | Address Bus hold time from trailing edge of Read Strobe | 50 | 500 | | ns | |
| $t_{ACS1}$ | Address change to Chip Select OFF | | 40 | | ns | |
| $t_{AD}$ | Address Bus Valid to Data Valid | | 850 | 1200 | ns | CL = 100 pF |
| $t_{HZ}$ | Time from trailing edge of Read Stobe until interface device bus drivers are in Tri-State mode | 0 | | 250 | ns | |

**Table 5. Timing: Data from Microprocessor to Peripheral**

| Symbol | Parameter | Min. | Typ. | Max. | Units | Comm. |
|---|---|---|---|---|---|---|
| $t_{ACS0}$ | Address Bus Valid to Chip Select ON (CS = 0) | | 40 | | ns | $V_{DD} = 5V$ |
| $t_{CSW}$ | Chip Select ON to Write Probe | 310 | 450 | | ns | |
| $t_{AW}$ | Address Bus Valid to Write Strobe | 350 | | | ns | |
| $t_{WW}$ | Write Strobe Width | 430 | | | ns | |
| $t_{DW}$ | Data Bus Valid before Write Stobe | 50 | | | ns | |
| $t_{WD}$ | Address Bus hold time following Write Strobe | 100 | | | ns | |
| $t_{WA}$ | Data Bus hold time following Write Strobe | 50 | | | ns | |
| $t_{ACS1}$ | Address change to Chip Select OFF (CS = 1) | | 40 | | ns | |

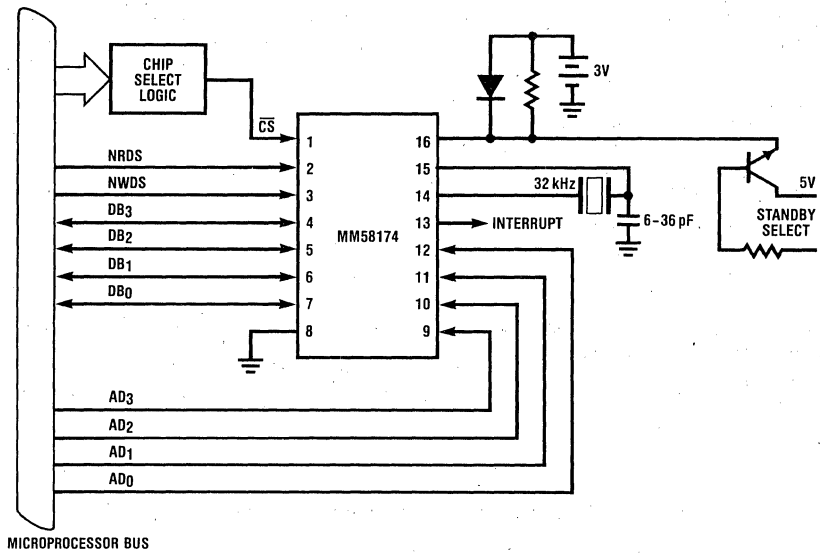



Figure 4. Typical Microprocessor Interface

## Timing Waveforms

### Read Mode

Figure 5 gives detailed timing in accordance with the Microbus Specification for Microprocessors for the transfer of data from peripheral to microprocessor. See Table 4.

All times are measured from (or to) valid logic "0" level = 0.8V or valid logic "1" level = 2.0V.

### Write Mode

Figure 6 gives detailed timing in accordance with the Microbus Specification for Microprocessors for the transfer of data from Microprocessor to peripheral. See Table 5.

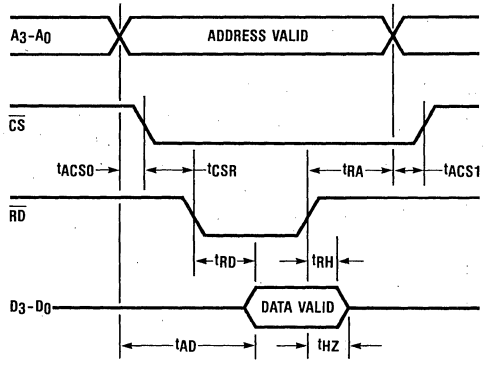



Figure 5. Read Cycle Waveforms

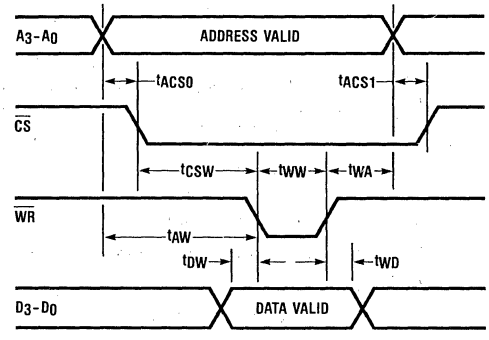



Figure 6. Write Cycle Waveforms

Section 6

# Application Notes

# Using National Clock Integrated Circuits in Timer Applications







## INTRODUCTION

The following is a description of a technique which allows the use of the National MM5309, MM5311, MM5312 and MM5315 clock integrated circuits as timers in industrial and consumer applications. What will be presented is the basic technique along with some simple circuitry and applications.

## BASIC TECHNIQUE

When first approaching the problem of using clock chips for timers, the most obvious technique is to attempt to compare the display data with preset BCD numbers. Because of the multiplexing and number of data bits this technique, while possible, is unwieldy and requires a large number of components.

An easier method is to use one or more demultiplexed BCD lines as control waveforms whose edges determine timer data. In *Figure 1* we examine the 1-bit of the BCD data of the units second time.

From this waveform we observe a one second wide pulse every two seconds. If we look at the 4-bit of the 10 minutes digit we find a pulse which is 20 minutes wide and occurs once each hour.

*Figure 3* is a chart showing the various pulses and their widths for all digits and the useful BCD lines.

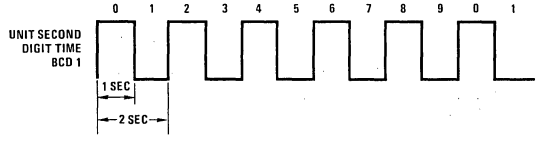



**FIGURE 1. 1 Second Pulse Every 2 Seconds**

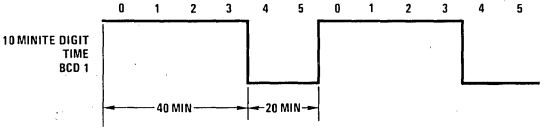



**FIGURE 2. 20 Minute Pulse Every Hour**

| BCD | PULSE RATE | PULSE WIDTH | BCD | PULSE RATE | PULSE WIDTH |
|---|---|---|---|---|---|
| | 1 Sec Digit | | | 10 Sec Digit | |
| 1 | 1 every 2 sec | 1 sec* | 1 | 1 every 20 sec | 10 sec* |
| 2 | | | 2 | 1 every min | 20 sec |
| 4 | 1 every 10 sec | 4 sec | 4 | 1 every min | 20 sec |
| 8 | 1 every 10 sec | 2 sec | 8 | | |
| | 1 Min Digit | | | 10 Min Digit | |
| 1 | 1 every 2 min | 1 min* | 1 | 1 every 20 min | 10 min* |
| 2 | | | 2 | 1 every hr | 20 min |
| 4 | 1 every 10 min | 4 min | 4 | 1 every hr | 20 min |
| 8 | 1 every 10 min | 2 min | 8 | | |
| | Units Hrs Digit (12 Hr Mode) | | | Units Hrs Digit (24 Hr Mode) | |
| 1 | 1 every 2 hrs | 1 hr* | 1 | 1 every 2 hrs | 1 hr* |
| 2 | | | 2 | | |
| 4 | 1 every 12 hrs | 4 hrs | 4 | | |
| 8 | 1 every 12 hrs | 4 hrs | 8 | | |
| | 10 Hrs Digit (12 Hr Mode) | | | 10 Hrs Digit (24 Hr Mode) | |
| 1 | | | 1 | 1 every 24 hrs | 10 hrs |
| 2 | 1 every 12 hrs | 9 hrs | 2 | 1 every 24 hrs | 4 hrs |
| 4 | 1 every 12 hrs | 9 hrs | | | |
| 8 | 1 every 12 hrs | 9 hrs | | | |

*Square waves

FIGURE 3

## SIMPLE DEMULTIPLEXING

In the simple case where, for example, a four hour wide pulse each day is desired, perhaps to turn on lights in the evening, a simple demultiplexing scheme using one diode is shown in *Figure 4*. When power is applied, the internal multiplex circuitry will strobe each digit until the digit with the diode connected is accessed. This digit will sink the multiplex charging current and stop the multiplex scanning. Thus, the BCD outputs now present the data from the selected digit. The waveforms as previously discussed are presented at the BCD lines. Note that these pulses are negative true for all BCD outputs.

An advantage of this type of timer over mechanical types is the elimination of line power drop outs. The circuit shown in *Figure 5* will maintain timing to within a few percent during periods of power line failure, but automatically return to the 60 Hz line for timing as soon as power is restored.

## MORE COMPLEX APPLICATIONS

Where it is desired to maintain the display, or in more complex timing of the "10 seconds every two hours" variety, external demultiplexing shown in *Figure 6* can be used. In this figure the BCD lines are demultiplexed with MM74C74 flip-flops. Examining the waveforms of these circuits we see two edges which allow the 10 second each two hours timing. These are differentiated by the NAND and INVERTERS and the first edge sets and the second resets the S-R flip-flop. The output of the flip-flop is ten seconds wide every two hours. By examining the edges of the *Figure 3* entries any combination of timings can be obtained with the circuit of *Figure 6*.

## LOW FREQUENCY WAVEFORM GENERATION

The asterisked BCD lines in *Figure 3* are those waveforms which are symmetric. By the use of the simple diode demultiplexing scheme previously discussed we

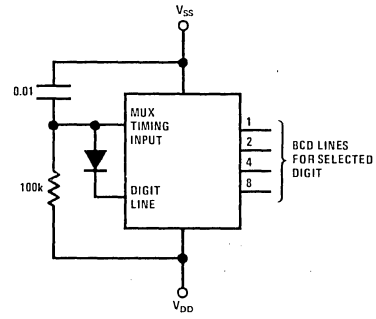



FIGURE 4

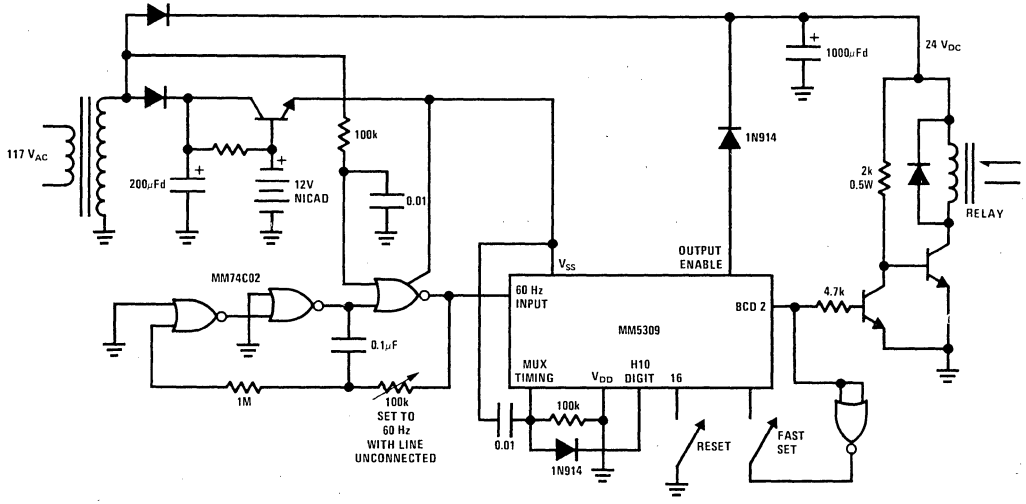



**FIGURE 5. Fail-Safe Automatic Lights Timer. Four Hours Each 24 Hours**

easily obtain square waves with periods of two seconds, two minutes, twenty minutes and two hours. In other cases, where the waveforms are asymmetric, a simple flip-flop can square, while dividing by two, these waveforms producing other low frequency square waves as long as one per two days.

## SUMMARY

We have shown some simple low cost timer and waveform generating examples using National clock integrated circuits. Because of the vast number of timing applications possible, this can in no way be looked at as the limit of clock-timer circuits. Use of the Reset on the MM5309 and MM5315 or the use of clocks in conjunction with programmable counters such as the MM74C161 allows other possibilities to meet specific applications. Also the clock chips themselves can run on frequencies other than 50 or 60 Hz (actually from dc to 10 kHz) which can allow scaling of the waveforms presented in *Figure 3* to different timing rates.

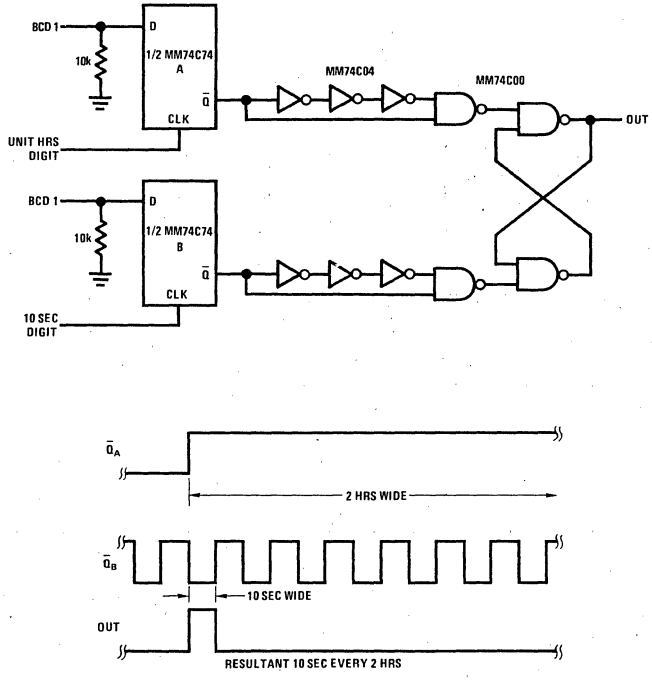



**FIGURE 6. More Universal Demultiplexing Technique**

# Digital Voltmeters and the MM5330







## INTRODUCTION

The first of what could be called the modern digital voltmeter began to appear in the early sixties. Prior to that time a few laboratory types were available, but they were plagued by inaccuracy, temperature drifts, and other problems inherent in vacuum tube technology.

One of the first successful, relatively low cost DMVS was a gated voltage-controlled oscillator configuration. The components of this technique consist of a high gain amplifier, a dc-to-frequency converter, and a linear, accurate frequency to-dc-converter developed from the reference voltage, which supplies the summing voltage at the input node. The amplifiers used were of the chopper stabilized type, that is, the error voltage is chopped to from an ac component which is amplified by ac coupled amplifiers then reconverted to dc. The choppers were made with light sensitive resistors, neon bulbs and light pipes.

They were built as the only method possible to avoid the drifts and offsets which were unavoidable in early transistor technology. Obviously the low current op amps so readily available today, are a significant advantange over these old systems.

The gate voltage was developed from the 60 Hz line. A problem which occurs when the gate is asynchronous with the frequency fed to the display counter, is also shown in *Figure 1*. A beat frequency effect is developed between the gate and the dc to frequency converter and produces a cyclic one digit error. These early voltmeters allowed this phenomenon to occur, today cyclic display errors are unacceptable.

A second display characteristic of these early voltmeters, was to use the ripple counters as the display storage, that is, the rippled counts would move through the display until the gate closed and the final value would be displayed. This was done primarily because of the number of discrete devices required to perform counting and latching. With the coming of integrated circuits, displays were improved, latches were employed, and blink-free displays were adopted.

Polarity selection was made by a front panel switch which internally rearranged references and other circuitry.

An example of today's use of the VCO technique is shown in *Figure 2*. This is a low cost digital thermometer, which, while not a DVM, still employs the basic components of the voltage-controlled oscillator system. These are the high gain amplifiers contained in the LM5700, the dc-to-frequency converter consisting of the transistor source and LM555 timer, and the frequency-to-dc converter consisting of the CMOS inverters and reference voltage. This brings up a characteristic of CMOS most useful in DVM's and other analog-to-digital converters, the ability to switch directly to the supply and ground without offsets. In this case the fixed width negative-going pulses, when filtered, produce a feedback voltage directly proportional to the number of pulses—frequency-to-dc conversion.

The early counter storage display system previously mentioned, is shown in *Figure 3*. Because the best display available was the gaseous tube, no attempt was made to blank displays during the counting period. When the gate closed, the counters had reached a certain count and these counts were displayed.

After the development of the integrated circuit, displays took on a configuration as in *Figure 4*. Between the counters and display, latches were placed to display previous data while new counts were accumulated. The cost and pack count of this scheme made another display technique popular, that of multiplexing.

Briefly, this technique consists of connecting, sequentially, each of the latches to a single decoder driver which drove the display digit which corresponded to that latch. When sequenced at a 50 Hz rate or greater, a flicker-free display results. For this type of display system, TRI-STATE® counter-latches were developed *(Figure 5)*. This technique is still used today in many DVM's.

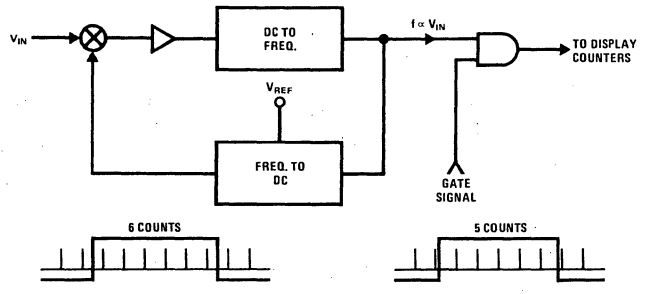



**FIGURE 1. Basic VCO Scheme**

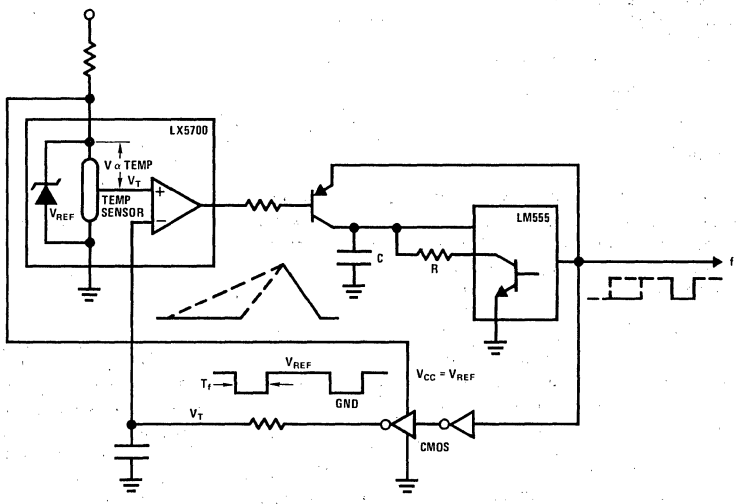



FIGURE 2. Typical VCO Circuit

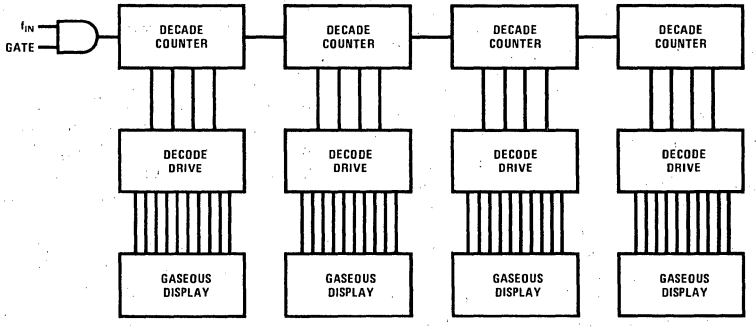



FIGURE 3. Early Display Configuration

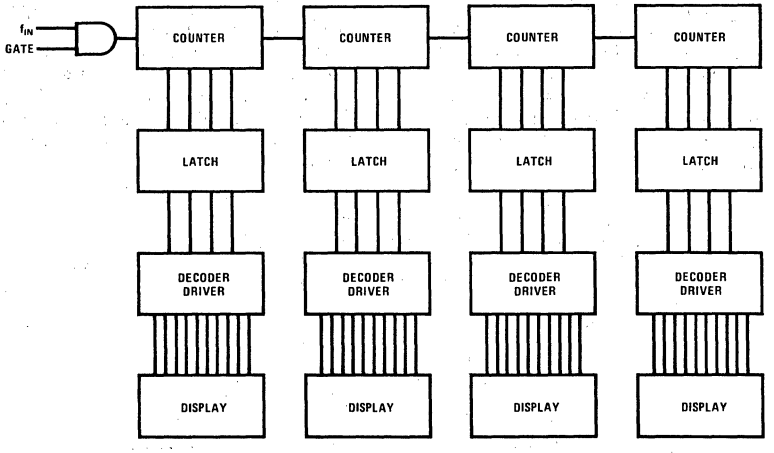



FIGURE 4. Integrated Circuit Display

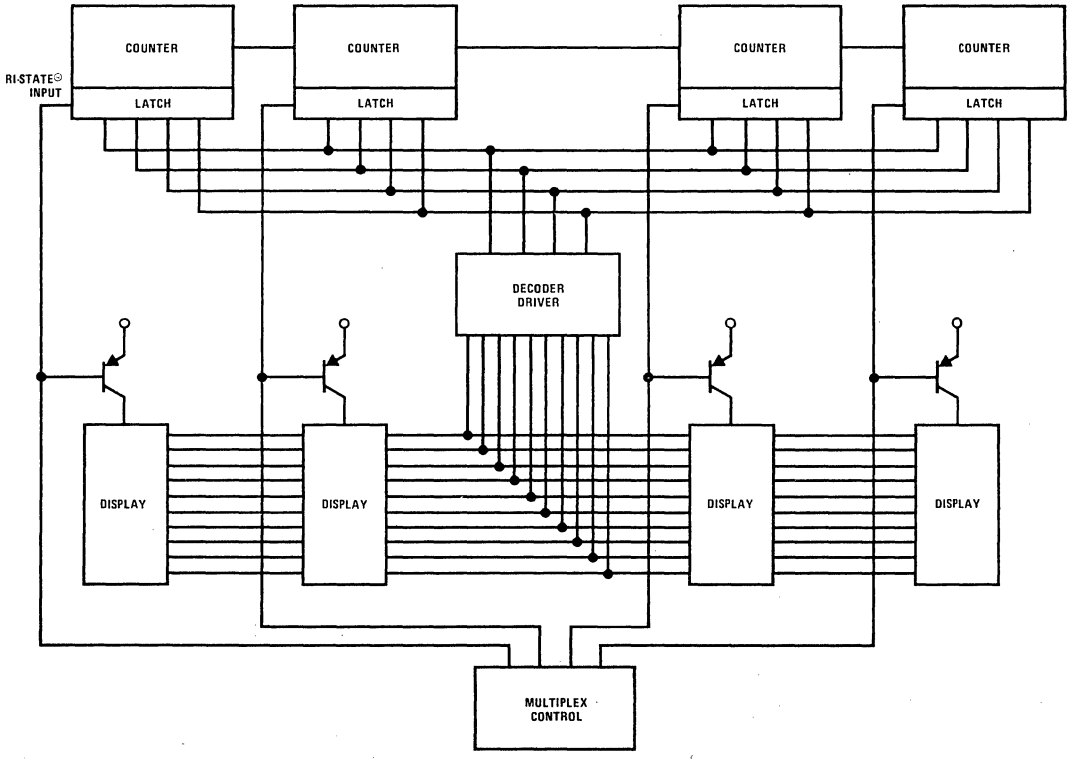



**FIGURE 5. Multiplexed Display**

While multiplexing cuts display costs considerably, the series connection of counters required to accumulate the counts proportional to voltage, could not be multiplexed to do the very nature of VCO or dual slope voltmeter schemes.

The recirculating remainder circuitry to be discussed next is unique in that the data is both derived and displayed on multiplexed, that is sequential digit basis (as seen in *Figure 6.*)

The technique used in the recirculating remainder circuit is to subtract digit valued voltage steps from the input voltage, until ten times the difference between these two voltages is less than ten times the digit valued steps. The number of voltage steps required is the display data and the ten times the difference voltage becomes the new voltage input for the next digit conversion. An example is shown in *Figure 7.*

An analog input of 6.903V is applied to the $[(V_{IN}-V_{STEP}) \times 10]$ amplifier. The ÷12 and decade counters are clocked simultaneously until a (difference x 10) less than $V_{REF}$ is detected by the comparator. At this time, the decade counter stops counting. In this example, the decade counter ceases counting on a six during the digit one period, thus a six is latched in the display. When the digit period ends, both counters are reset and the (difference x 10) voltage is recirculated via the CMOS switch and sample and hold capacitor to become the digit two input voltage (9.03V). The process is then repeated for the next digit. At a repetition rate of 50 Hz or greater, this produces a flicker-free, blink-free display. As such the recirculating remainder system has but one counter, one latch, and one decoder driver for as many digits as are desired. Once again CMOS is used for its capability to swing directly to the supply rail and controls the R-2R ladder directly from the reference voltage.

Some disadvantages of the system are the difficulties in reading voltages of both polarities and an unusual sort of error characteristic when slight ladder or reference drifts occur. While both VCO and dual slope techniques have gradual slope or linearity errors, the recirculating remainder errors are step-like in response to gradual input voltage changes. Lastly, the update rate is fixed by display flicker requirements and thus measurements of noisy voltages cause an annoying inability to read the last digits. It was however, an accurate low-cost technique used successfully in pre-LSI digital voltmeters.

The most widely used system for analog-to-digital conversion is the dual slope circuit. The basic dual slope system appears in *Figure 8.* Assuming the integrator output at zero when $V_x$ is applied, the integrator begins to ramp with an output voltage $V = I_x\ t/C$ where $I_x = -V_x/R$. Simultaneously with the beginning of this ramping, counts from an oscillator are fed into the display counters. At some fixed time, usually counter overflow, $V_x$ would be disconnected and the reference voltage connected to the resistor. The integrator now ramps at $V = I_{REF}\ t/C$ where $I_{REF} = V_{REF}/R$.

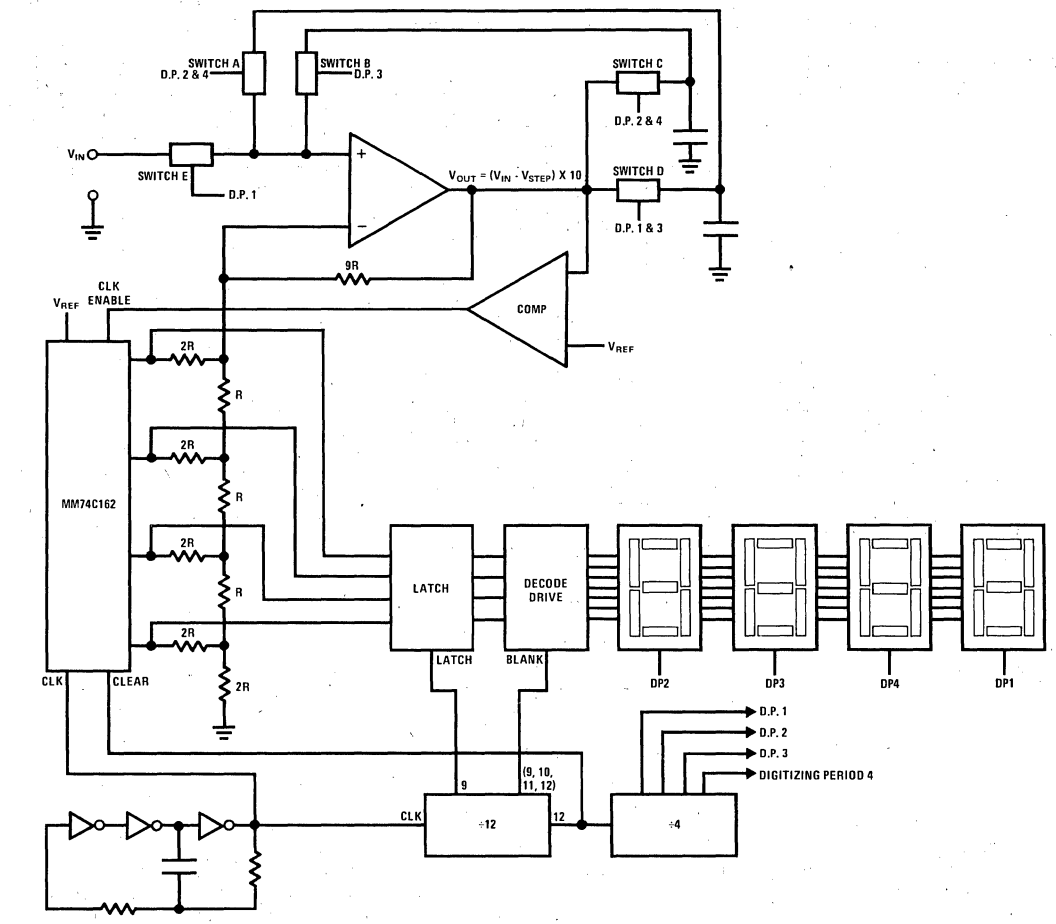
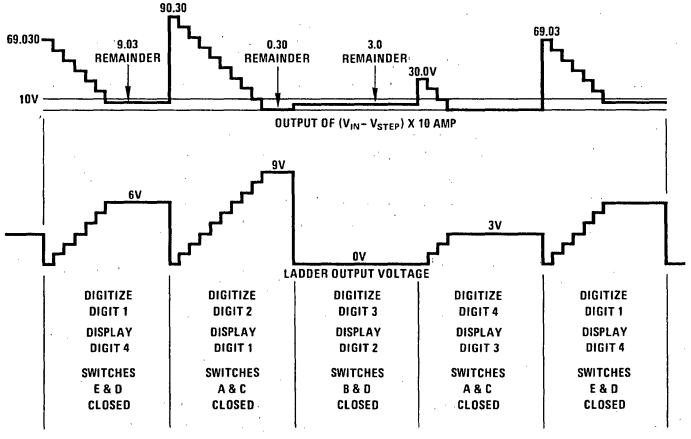



FIGURE 6. Basic Recirculating Remainder System




Example: (Voltage in = 6.903V)

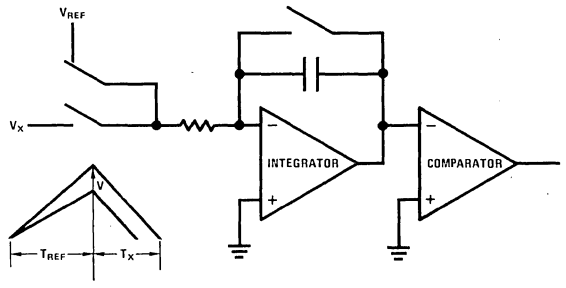
6.903
−6.000 — 6 1.000V steps—display 6
0.903 x 10 = 9.03

9.030
−9.000 — 9 1.000V steps—display 9
0.03 x 10 = 0.3

0.300
−0.000 — 0 1.000V steps—display 0
0.3 x 10 = 3

3.000
−3.000 = — 3 1.000V steps—display 3

FIGURE 7. Recirculating Remainder Waveforms




$$V = \frac{I_X t_{REF}}{C} \qquad I_X = -\frac{V_X}{R}$$

$$-V = \frac{I_{REF} T_X}{C} \qquad I_{REF} = \frac{V_{REF}}{R}$$

$$\frac{I_X T_{REF}}{C} = -\frac{I_{REF} T_X}{C}$$

$$\therefore V_X T_{REF} = V_{REF} T_X$$

**FIGURE 8. Basic Dual Slope**

When the integrator crosses the comparator threshold, the counters are latched to the number of counts accumulated from T to $T_x$. Clearly the voltage at $T_{REF}$ was $I_x\ T_{REF}/C$ and the voltage integrated from $T_{REF}$ to $T_x$ was $-I_{REF}\ T_x/C$ and these two voltages are equal. Therefore,

$$\frac{I_x\ T_{REF}}{C} = \frac{-I_{REF}\ T_x}{C} \text{ or}$$

$$V_x\ T_{REF} = V_{REF}\ T_x$$

Thus, the number of counts accumulated in the display from $T_{REF}$ to $T_x$ is proportional to the unknown voltage. Thus, the basic dual slope system has no gate, and requires stability of the R, C and count frequency only over one conversion period.

The technique for insuring that the ramp begins at zero on each conversion cycle, is to short the capacitor with a switch after each conversion is made. This, of course, forces the integrator output to zero until the next conversion period begins. It is also necessary to start each conversion cycle synchronously with the counter input frequency, or cyclic display errors like that of the gated VCO will appear in the display.

To measure both polarities in conventional dual slope systems, $V_{REF}$ must change in polarity. A problem which can occur is that bias currents which will add to the slope in one polarity, will subtract from the slope in the other. The usual solution, is to use op amps of very low input bias current. Also offset voltages in either the op amps or comparator can cause significant error unless carefully controlled.

Hence, while conventional dual slope has many advantages, its use requires considerable care in op amp, and comparator selection. Also, the measurement of either polarity requires two reference voltages which are, in accurate systems, quite expensive.

The MM5330 is the display and control for a modified dual slope system. It contains, as shown in *Figure 9*, the counters and latches, together with a multiplexing system to provide four digits of display with one decoder driver. It also provides a sign digit, either plus or minus, and a ten-thousand counts digit for a full display of ±19999. By eliminating the right-most digits it may also be used as a 2 1/2 or 3 1/2 digit DVM chip.

The basic modified dual slope system for which the MM5330 is designed, is shown in *Figure 10*. The integrator is now used in a non-inverting mode and is biased to integrate negatively for all voltages below $V_{MAX}$. Thus, if the maximum positive voltage at $V_{IN}$ is 1.9999V, the $V_{MAX}$ would be set at 2.2000V. In this way, all voltages measured are below $V_{MAX}$. This eliminates the need for reference switching and makes the system automatic polarity, with no additional components. Also, it can be shown that the amplifier input bias currents which cause the aforementioned errors in conventional dual slope systems, are eliminated by merely zeroing the display. Thus, low bias current op amps are not necessarily required unless a high input impedance is desired at $V_{IN}$.

Secondly, the use of a conventional op amp for a comparator, allows zeroing of all voltage offsets in both the op amp and comparator. This is achieved by zeroing the voltage on the capacitor through the use of the comparator as part of a negative feedback loop. During the zeroing period, the non-inverting input of the integrator is at $V_{REF}$. As this voltage is within the active common-mode range of the integrator the loop will respond by placing the integrator and comparator in the active region. The voltage on the capacitor is no longer equal to zero, but rather to a voltage which is the sum of both the op amp and comparator offset voltages. Because of the intrinsic nature of an integrator, this constant voltage remains throughout the integrating cycle and serves to eliminate even large offset voltages.

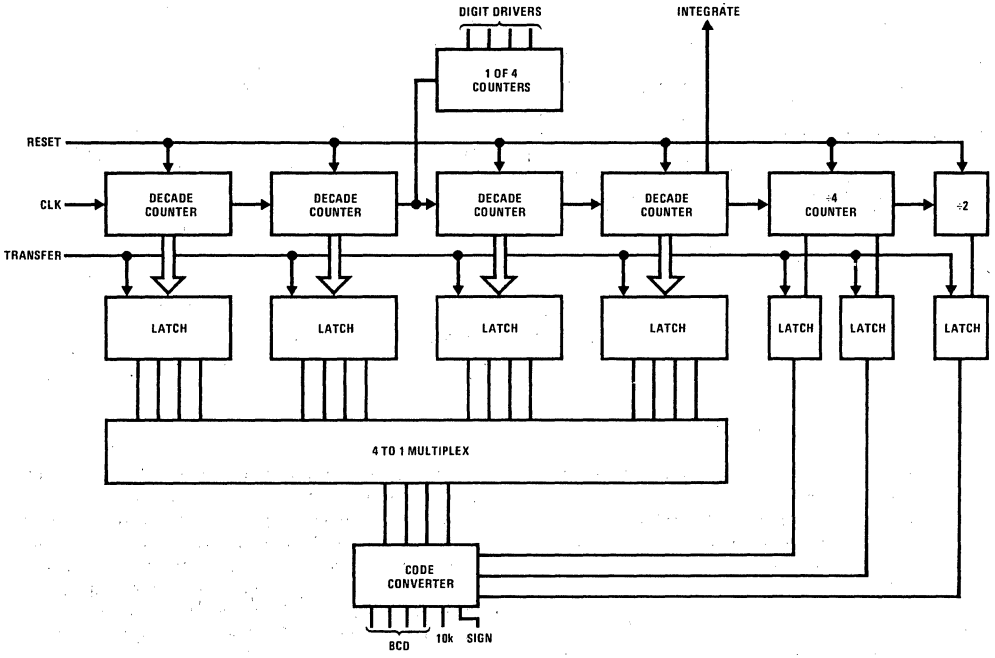



FIGURE 9. Block Diagram MM5330

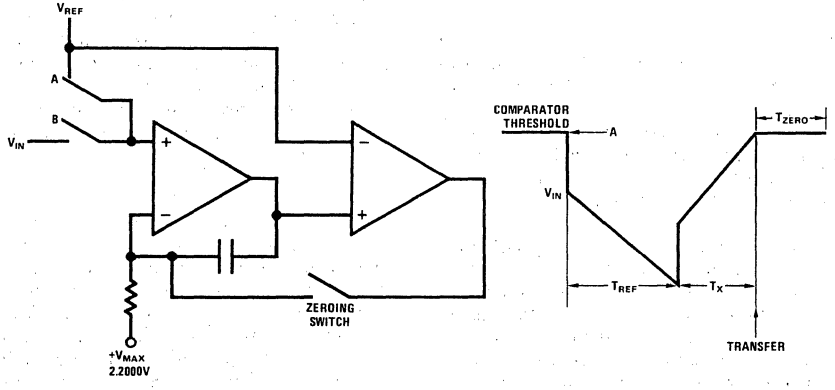



FIGURE 10. Modified Dual Slope

The waveforms at the output of the integrator are as shown. The voltage at A is the comparator threshold just discussed. Simultaneously, with the opening of switch A, $V_{IN}$ is connected to the input of the integrator via switch B. The output then slews to $V_{IN}$. Integration then begins for the reference period, after which time, the reference voltage is again applied to the input. The output again slews the difference between $V_{REF}$ and $V_{IN}$ then integrates for the unknown period until the comparator threshold is crossed. At this point, the accumulated counts are transferred from the counters to the latches and zeroing begins until the next conversion interval.

It may be obvious, however, that while we have eliminated several of the basic dual slope circuits disadvantages, we have created another—the number of counts are no longer proportional to $V_{IN}$ but rather to $(V_{MAX}-V_{IN})$. In fact, when we short $V_{IN}$ to ground we are actually measuring our own 2.2000V $V_{MAX}$.

What is done in the MM5330 is to code convert the number of counts as shown in *Figure 11*. This chart shows a code conversion starting at the time of a reset. The first 18,000 counts are the reference period after which time the integrator changes slope. If a comparator crossing is detected within the next 2000 counts, a plus overrange condition will occur at the display. This condition results in a lit plus sign, a lit one and four blanked right-most digits. A transfer at 20,000, however, will create a reading of +1.9999, at 20,001 a reading of 19,998 and so on, until at 40,000 a reading of +0000 would be displayed. A transfer occuring at 40,001 would cause a −0001 display and so on until 60,000 counts were entered at which time a −1 with four blanked zeros would be displayed indicating a minus overrange condition.

A typical circuit for a low cost 4 1/2 digit circuit is shown in *Figure 12*. The display interface used is a TTL, seven-segment decoder driver and four PNP transistors. The ±1 digit is driven directly by CMOS. The clock-synchronous reset and transfer functions prevent any cyclic digit variations and present a blink-free flicker-free display. CMOS analog switches are used as reference, zero, and input switches and used also in a comparator slew rate circuit.

A problem with all dual slope systems occurs when short integrating times and high clock frequencies are used. Because of the very slow rise time of the ramp into the comparator, the output of the comparator will normally ramp at approximately 1/10 of its actual slew rate. Thus a significant number of extra counts are displayed due to the slow rate of rise of the comparator. A technique to improve this consists of capacitor $C_S$ and analog switch section four. An unstable positive loop is created by this capacitor when the comparator comes out of saturation. This causes the output to rise at its slew rate to the comparator threshold. As soon as this threshold is reached the analog switch opens and zeroing is initiated as previously discussed.

The rapid improvement in display and LSI technology has allowed considerable improvement in digital voltmeters. The modified dual slope technique together with the simplified display interface of the MM5330 are felt to be a much improved technique when compared to circuits of just a short time ago. While DVM chips do not by themselves solve all inherent problems, their careful use allows low cost, high accuracy units, with excellent display characteristics.

| COUNTS AFTER RESET | DISPLAY |
|---|---|
| 0 | |
| . | |
| . | +1---- |
| . | |
| 18,000 | |
| . | |
| . | +1---- |
| . | |
| 19,999 | |
| 20,000 | +19999 |
| 20,001 | +19998 |
| . | |
| . | |
| . | |
| 40,000 | + 0000 |
| 40,001 | − 0001 |
| . | |
| . | |
| . | |
| 59,999 | −19999 |
| 60,000 | −1---- |

**FIGURE 11. Code Conversion Table MM5330**

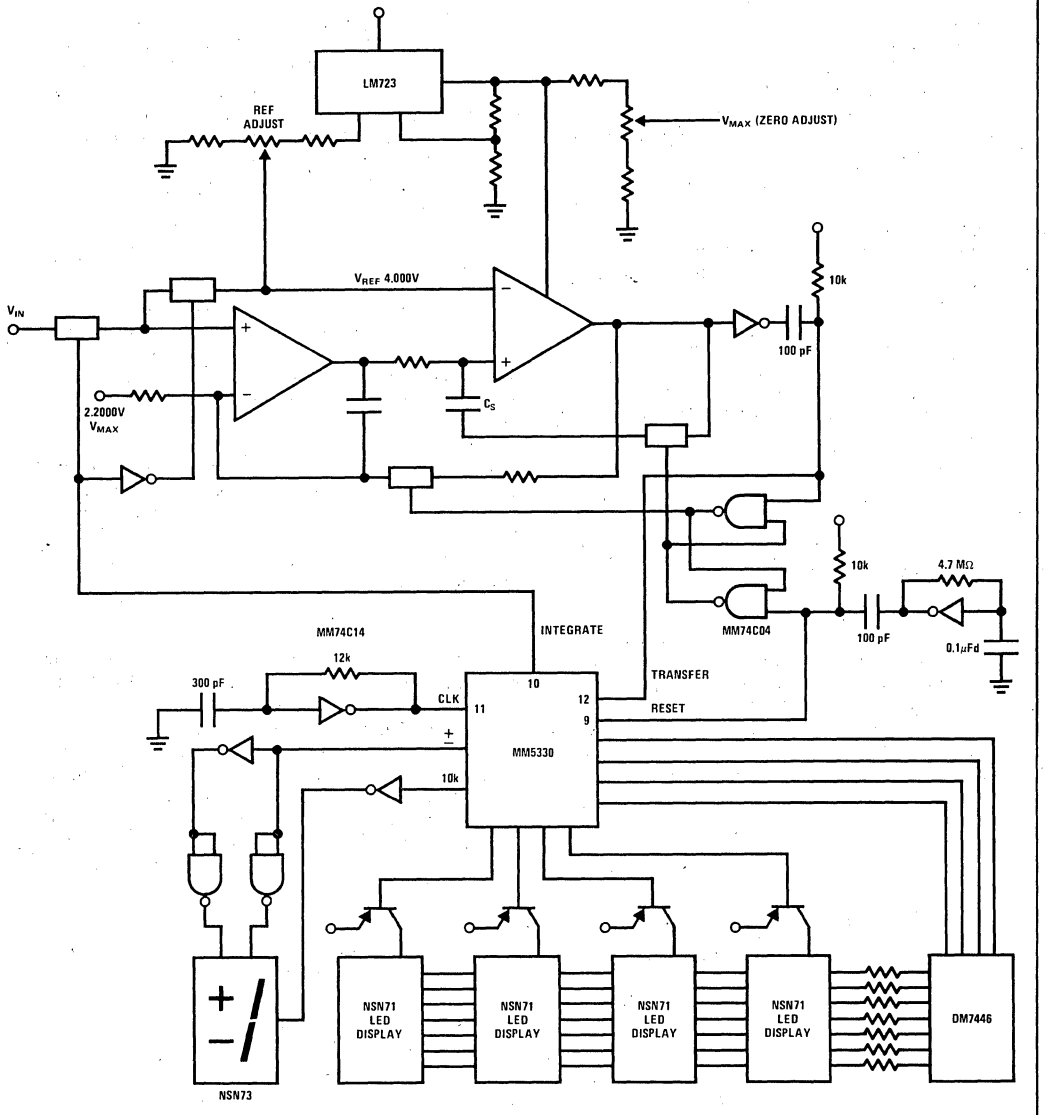



FIGURE 12. Typical Application MM5330

# Specifying A/D and D/A Converters







The specification or selection of analog-to-digital (A/D) or digital-to-analog (D/A) converters can be a chancey thing unless the specifications are understood by the person making the selection. Of course, you know you want an accurate converter of specific resolution; but how do you insure that you get what you want? For example, 12 switches, 12 arbitrarily valued resistors, and a reference will produce a 12-bit DAC exhibiting 12 quantum steps of output voltage. In all probability, the user wants something better than the expected performance of such a DAC. Specifying a 12-bit DAC or an ADC must be made with a full understanding of accuracy, linearity, differential linearity, monotonicity, scale, gain, offset, and hysteresis errors.

This note explains the meanings of and the relationships between the various specifications encountered in A/D and D/A converter descriptions. It is intended that the meanings be presented in the simplest and clearest practical terms. Included are transfer curves showing the several types of errors discussed. Timing and control signals and several binary codes are described as they relate to A/D and D/A converters.

## MEANING OF PERFORMANCE SPECS

**Resolution** describes the smallest standard incremental change in output voltage of a DAC or the amount of input voltage change required to increment the output of an ADC between one code change and the next adjacent code change. A converter with n switches can resolve 1 part in $2^n$. The least significant increment is then $2^{-n}$, or one least significant bit (LSB). In contrast, the most significant bit (MSB) carries a weight of $2^{-1}$. Resolution applies to DACs and ADCs, and may be expressed in percent of full scale or in binary bits. For example, an ADC with 12-bit resolution could resolve 1 part in $2^{12}$ (1 part in 4096) or 0.0245% of full scale. A converter with 10V full scale could resolve a 2.45mV input change. Likewise, a 12-bit DAC would exhibit an output voltage change of 0.0245% of full scale when the binary input code is incremented one binary bit (1 LSB). Resolution is a design parameter rather than a performance specification; it says nothing about accuracy or linearity.

**Accuracy** is sometimes considered to be a non-specific term when applied to D/A or A/D converters. A linearity spec is generally considered as more descriptive. An accuracy specification describes the worst case deviation of the DAC output voltage from a straight line drawn between zero and full scale; it includes all errors. A 12-bit DAC could not have a conversion accuracy better than ±½ LSB or ±1 part in $2^{12+1}$ (±0.0122% of full scale due to finite resolution). This would be the case in figure 1 if there were no errors. Actually, ±0.0122% FS represents a deviation from 100% accuracy; therefore accuracy should be specified as 99.9878%. However, convention would dictate 0.0122% as being an accuracy spec rather than an inaccuracy (tolerance or error) spec.

Accuracy as applied to an ADC would describe the difference between the actual input voltage and the full-scale weighted equivalent of the binary output code; included are quantizing and all other errors. If a 12-bit ADC is stated to be ±1 LSB accurate, this is equivalent to ±0.0245% or twice the minimum possible quantizing error of 0.0122%. An accuracy spec describes the maximum sum of all errors including quantizing error, but is rarely provided on data sheets as the several errors are listed separately.

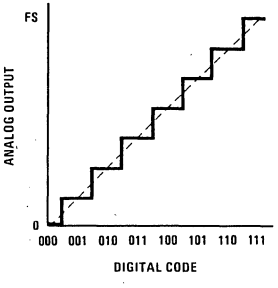



**FIGURE 1. Linear DAC Transfer Curve Showing Minimum Resolution Error and Best Possible Accuracy**

**Quantizing Error** is the maximum deviation from a straight line transfer function of a perfect ADC. As, by its very nature, an ADC quantizes the analog input into a finite number of output codes, only an infinite resolution ADC would exhibit zero quantizing error. A perfect ADC, suitably offset ½ LSB at zero scale as shown in figure 2, exhibits only ±½ LSB maximum output error. If not offset, the error will be $\frac{+1}{-0}$ LSB as shown in figure 3. For example, a perfect 12-bit ADC will show a ±½ LSB error of ±0.0122% while the quantizing error of an 8-bit ADC is ±½ part in $2^8$ or ±0.195% of full scale. Quantizing error is not strictly applicable to a DAC; the equivalent effect is more properly a resolution error.

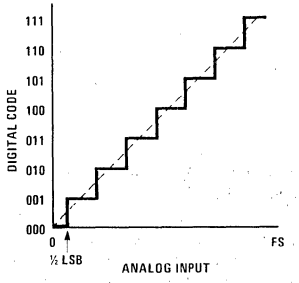



FIGURE 2. ADC Transfer Curve, ½ LSB Offset at Zero

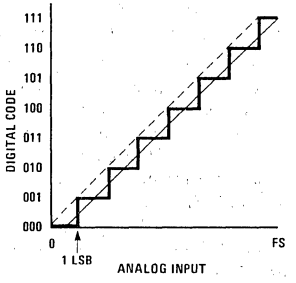



FIGURE 3. ADC Transfer Curve, No Offset

**Scale Error** (full scale error) is the departure from design output voltage of a DAC for a given input code, usually full-scale code. (See figure 4.) In an ADC it is the departure of actual input voltage from design input voltage for a full-scale output code. Scale errors can be caused by errors in reference voltage, ladder resistor values, or amplifier gain, *et. al.* (See **Temperature Coefficient.**) Scale errors may be corrected by adjusting output amplifier gain or reference voltage. If the transfer curve resembles that of figure 7, a scale adjustment at ¾ scale could improve the overall ± accuracy compared to an adjustment at full scale.

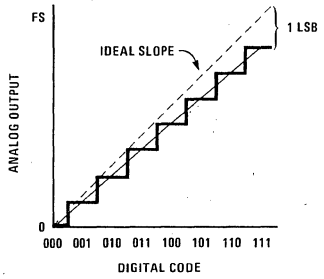
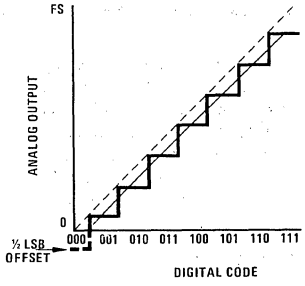



FIGURE 4. Linear, 1 LSB Scale Error

**Gain Error** is essentially the same as scale error for an ADC. In the case of a DAC with current and voltage mode outputs, the current output could be to scale while the voltage output could exhibit a gain error. The amplifier feedback resistors would be trimmed to correct the gain error.

**Offset Error** (zero error) is the output voltage of a DAC with zero code input, or it is the required mean value of input voltage of an ADC to set zero code out. (See figure 5.) Offset error is usually caused by amplifier or comparator input offset voltage or current; it can usually be trimmed to zero with an offset zero adjust potentiometer external to the DAC or ADC. Offset error may be expressed in % FS or in fractional LSB.




**Hysteresis Error** in an ADC causes the voltage at which a code transition occurs to be dependent upon the direction from which the transition is approached. This is usually caused by hysteresis in the comparator inside an ADC. Excessive hysteresis may be reduced by design; however, some slight hysteresis is inevitable and may be objectionable in converters if hysteresis approaches ½ LSB.

**Linearity**, or, more accurately, non-linearity specifications describe the departure from a linear transfer curve for either an ADC or a DAC. Linearity error does not include quantizing, zero, or scale errors. Thus, a specifi-

cation of ±½ LSB linearity implies error in addition to the inherent ±½ LSB quantizing or resolution error. In reference to figure 2, showing no errors other than quantizing error, a linearity error allows for one or more of the steps being greater or less than the ideal shown.

Figure 6 shows a 3-bit DAC transfer curve with no more than ±½ LSB non-linearity, yet one step shown is of zero amplitude. This is within the specification, as the maximum deviation from the ideal straight line is ±1 LSB (½ LSB resolution error plus ½ LSB non-linearity). With any linearity error, there is a differential non-linearity (see below). A ±½ LSB linearity spec guarantees monotonicity (see below) and ⩽ ±1 LSB differential non-linearity (see below). In the example of figure 6, the code transition from 100 to 101 is the worst possible non-linearity, being the transition from 1 LSB high at code 100 to 1 LSB low at 110. Any fractional non-linearity beyond ±½ LSB will allow for a non-monotonic transfer curve. Figure 7 shows a typical non-linear curve; non-linearity is 1¼ LSB yet the curve is smooth and monotonic.

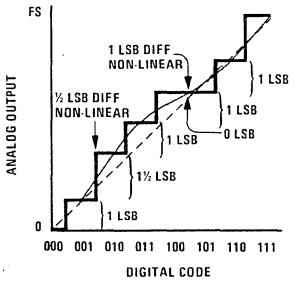



**FIGURE 6. ±½ LSB Non-Linearity (Implies 1 LSB Possible Error), 1 LSB Differential Non-Linearity (Implies Monotonicity)**

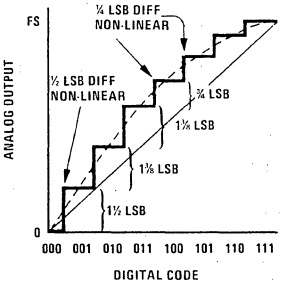



**FIGURE 7. 1¼ LSB Non-Linear, ½ LSB Differential Non-Linearity**

Linearity specs refer to either ADCs or to DACs, and do not include quantizing, gain, offset, or scale errors. Linearity errors are of prime importance along with differential linearity in either ADC or DAC specs, as all other errors (except quantizing, and temperature and long-term drifts) may be adjusted to zero. Linearity errors may be expressed in % FS or fractional LSB.

**Differential Non-Linearity** indicates the difference between actual analog voltage change and the ideal (1 LSB) voltage change at any code change of a DAC. For example, a DAC with a 1.5 LSB step at a code change would be said to exhibit ½ LSB differential non-linearity (see figures 6 and 7). Differential non-linearity may be expressed in fractional bits or in % FS.

Differential linearity specs are just as important as linearity specs because the apparent quality of a converter curve can be significantly affected by differential non-linearity even though the linearity spec is good. Figure 6 shows a curve with a ±½ LSB linearity and ±1 LSB differential non-linearity while figure 7 shows a curve with +1¼ LSB linearity and ±½ LSB differential non-linearity. In many user applications, the curve of figure 7 would be preferred over that of figure 6 because the curve is smoother. The differential non-linearity spec describes the smoothness of a curve; therefore it is of great importance to the user. A gross example of differential non-linearity is shown in figure 8 where the linearity spec is ±1 LSB and the differential linearity spec is ±2 LSB. The effect is to allow a transfer curve with grossly degraded resolution; the normal 8-step curve is reduced to 3 steps in figure 8. Similarly, a 16-step curve (4-bit converter) with only 2 LSB differential non-linearity could be reduced to 6 steps (a 2.6-bit converter?). The real message is, "Beware of the specs." Do not ignore or omit differential linearity characteristics on a converter unless the linearity spec is tight enough to guarantee the desired differential linearity. As this characteristic is impractical to measure on a production basis, it is rarely, if ever, specified, and linearity is the primary specified parameter. Differential non-linearity can always be as much as twice the non-linearity, but no more.

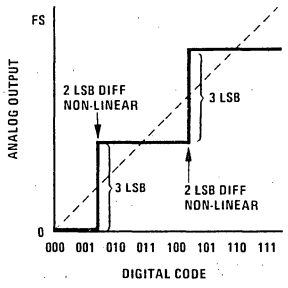



**FIGURE 8. ±1 LSB Linear, ±2 LSB Differential Non-Linear**

**Monotonicity.** A monotonic curve has no change in sign of the slope; thus all incremental elements of a monotonically increasing curve will have positive or zero, but never negative slope. The converse is true for decreasing curves. The transfer curve of a monotonic DAC will contain steps of only positive or zero height, and no negative steps. Thus a smooth line connecting all output voltage points will contain no peaks or dips. The transfer function of a monotonic ADC will provide no decreasing output code for increasing input voltage.

Figure 9 shows a non-monotonic DAC transfer curve. For the curve to be non-monotonic, the linearity error must exceed ±½ LSB no matter by how little. The greater the linearity error, the more significant the negative step might be. A non-monotonic curve may not be a special disadvantage in some systems; however, it is a disaster in closed-loop servo systems of any type (including a DAC-controlled ADC). A ±½ LSB maximum linearity spec on an n-bit converter guarantees monotonicity to n bits. A converter exhibiting more than ±½ LSB non-linearity may be monotonic, but is not necessarily monotonic. For example, a 12-bit DAC with ±½ bit linearity to 10 bits (not ±½ LSB) will be monotonic at 10 bits but may or may not be monotonic at 12 bits unless tested and guaranteed to be 12-bit monotonic.

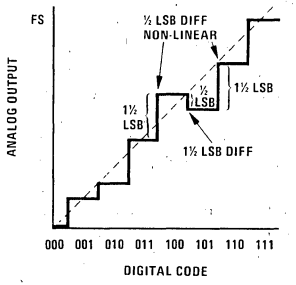
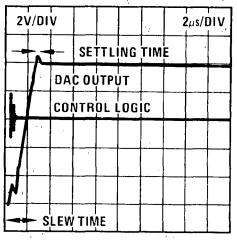
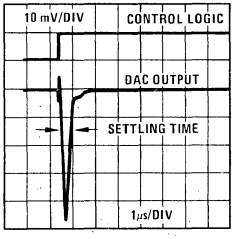



FIGURE 9. Non-Monotonic (Must be > ±½ LSB Non-Linear)




(a) Full-Scale Step




(b) 1 LSB Step

FIGURE 10. DAC Slew and Settling Time

**Settling Time** is the elapsed time after a code transition for DAC output to reach final value within specified limits, usually ±½ LSB. (See also **Conversion Rate** below.) Settling time is often listed along with a slew rate specification; if so, it may not include slew time. If no slew rate spec is included, the settling time spec must be expected to include slew time. Settling time is usually summed with slew time to obtain total elapsed time for the output to settle to final value. Figure 10 delineates that part of the total elapsed time which is considered to be slew and that part which is settling time. It is apparent from this figure that the total time is greater for a major than for a minor code change due to amplifier slew limitations, but settling time may also be different depending upon amplifier overload recovery characteristics.

**Slew Rate** is an inherent limitation of the output amplifier in a DAC which limits the rate of change of output voltage after code transitions. Slew rate is usually anywhere from 0.2 to several hundred volts/μs. Delay in reaching final value of DAC output voltage is the sum of slew time and settling time as shown in figure 10.

**Overshoot and Glitches** occur whenever a code transition occurs in a DAC. There are two causes. The current output of a DAC contains switching glitches due to possible asynchronous switching of the bit currents (expected to be worst at half-scale transition when all bits are switched). These glitches are normally of extremely short duration but could be of ½ scale amplitude. The current switching glitches are generally somewhat attenuated at the voltage output of the DAC because the output amplifier is unable to slew at a very high rate; they are, however, partially coupled around the amplifier via the amplifier feedback network and seen at the output. The output amplifier introduces overshoot and some non-critically damped ringing which may be minimized but not entirely eliminated except at the expense of slew rate and settling time.

**Temperature Coefficient** of the various components of a DAC or ADC can produce or increase any of the several errors as the operating temperature varies. Zero scale offset error can change due to the TC of the amplifier and comparator input offset voltages and currents. Scale error can occur due to shifts in the reference, changes in ladder resistance or non-compensating RC product shifts in dual-slope ADCs, changes in beta or reference current in current switches, changes in amplifier bias current, or drift in amplifier gain-set resistors. Linearity and monotonicity of the DAC can be affected by differential temperature drifts of the ladder resistors and switches. Overshoot, settling time, and slew rate can be affected by temperature due to internal change in amplifier gain and bandwidth. In short, every specification except resolution and quantizing error can be affected by temperature changes.

**Long-Term Drift**, due mainly to resistor and semiconductor aging can affect all those characteristics which temperature change can affect. Characteristics most commonly affected are linearity, monotonicity, scale, and offset. Scale change due to reference aging is usually the most important change.

**Supply Rejection** relates to the ability of a DAC or ADC to maintain scale, offset, TC, slew rate, and linearity when the supply voltage is varied. The reference must, of course, remain constant unless considering a multiplying DAC. Most affected are current sources (affecting linearity and scale) and amplifiers or comparators (affecting offset and slew rate). Supply rejection is usually specified only as a % FS change at or near full scale at 25°C.

**Conversion Rate** is the speed at which an ADC or DAC can make repetitive data conversions. It is affected by propagation delay in counting circuits, ladder switches and comparators; ladder RC and amplifier settling times; amplifier and comparator slew rates; and integrating time of dual-slope converters. Conversion rate is specified as a number of conversions per second, or conversion time is specified as a number of microseconds to complete one conversion (including the effects of settling time). Sometimes, conversion rate is specified for less than full resolution, thus showing a misleading (high) rate.

**Clock Rate** is the minimum or maximum pulse rate at which ADC counters may be driven. There is a fixed relationship between the minimum conversion rate and the clock rate depending upon the converter accuracy and type. All factors which affect conversion rate of an ADC limit the clock rate.

**Input Impedance** of an ADC describes the load placed on the analog source.

**Output Drive Capability** describes the digital load driving capability of an ADC or the analog load driving capacity of a DAC; it is usually given as a current level or a voltage output into a given load.

## CODES

Several types of DAC input or ADC output codes are in common use. Each has its advantages depending upon the system interfacing the converter. Most codes are binary in form; each is described and compared below.

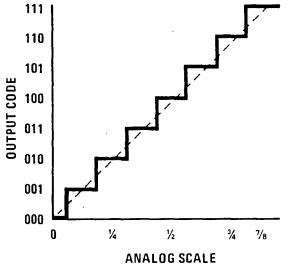
**Natural Binary** (or simply Binary) is the usual $2^n$ code with 2, 4, 8, 16, . . . , $2^n$ progression. An input or output high or "1" is considered a signal, whereas a "0" is considered an absence of signal. This is a positive true binary signal. Zero scale is then all "zeros" while full scale is all "ones."

**Complementary Binary** (or Inverted Binary) is the negative true binary system. It is identical to the binary code except that all binary bits are inverted. Thus, zero scale is all "ones" while full scale is all "zeros."

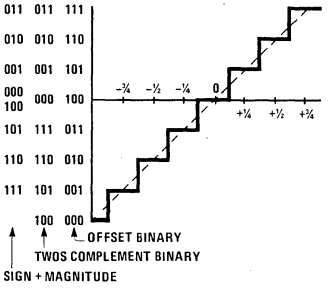
**Binary Coded Decimal** (BCD) is the representation of decimal numbers in binary form. It is useful in ADC systems intended to drive decimal displays. Its advantage over decimal is that only 4 lines are needed to represent 10 digits. The disadvantage of coding DACs or ADCs in BCD is that a full 4 bits could represent 16 digits while only 10 are represented in BCD. The full-scale resolution of a BCD coded system is less than that of a binary coded system. For example, a 12-bit BCD system has a resolution of only 1 part in 1000 compared to 1 part in 4096 for a binary system. This represents a loss in resolution of over 4:1.

**Offset Binary** is a natural binary code except that it is offset (usually ½ scale) in order to represent negative and positive values. Maximum negative scale is represented to be all "zeros" while maximum positive scale is represented as all "ones." Zero scale (actually center scale) is then represented as a leading "one" and all remaining "zeros." The comparison with binary is shown in figure 11.

**Twos Complement Binary** is an alternate and more widely used code to represent negative values. With this code, zero and positive values are represented as in natural binary while all negative values are represented in a twos complement form. That is, the twos complement of a number represents a negative value so that interface to a computer or microprocessor is simplified. The twos complement is formed by complementing each bit and then adding a 1; any overflow is neglected. The decimal number –8 is represented in twos complement as follows: start with binary code of decimal 8 (off scale for ± representation in 4 bits so not a valid code in the ± scale of 4 bits) which is 1000; complement it to 0111; add 0001 to get 1000. The comparison with offset binary is shown in figure 11. Note that the offset binary representation of the ± scale differs from the twos complement representation only in that the MSB is complemented. The conversion from offset binary to twos complement only requires that the MSB be inverted.




(a) Zero to + Full-Scale




(b) ± Full-Scale

**FIGURE 11. ADC Codes**

**Sign Plus Magnitude** coding contains polarity information in the MSB (MSB = 1 indicates a negative sign); all other bits represent magnitude only. This code is compared to offset binary and twos complement in figure 11. Note that one code is used up in providing a double code for zero. Sign plus magnitude code is used in certain instrument and audio systems; its advantage is that only one bit need be changed for small scale changes in the vicinity of zero, and plus and minus scales are symmetrical. A DVM might be an example of its use.

## CONTROL

Each ADC must accept and/or provide digital control signals telling it and/or the external system what to do and when to do it. Control signals should be compatible with one or more types of logic in common use. Control signal timing must be such that the converter or connected system will accept the signals. Common control signals are listed below.

**Start Conversion** (SC) is a digital signal to an ADC which initiates a single conversion cycle. Typically, an SC signal must be present at the fall (or rise) of the clock waveform to initiate the cycle. A DAC needs no SC signal; however, such could be provided to gate digital inputs to a DAC.

**End of Conversion** (EOC) is a digital signal from an ADC which informs the external system that the digital output data is valid. Typically, an EOC output can be connected to an SC input to cause the ADC to operate in continuous conversion mode. In non-continuous conversion systems, the SC signal is a command from the system to the ADC. A DAC does not supply an EOC signal.

**Clock** signals are required or must be generated within an ADC to control counting or successive approximation registers. The clock controls the conversion speed within the limitations of the ADC. DACs do not require clock signals.

## CONCLUSION

Once the user has a working knowledge of DAC or ADC characteristics and specifications, he should be able to select a converter to suit a specific system need. The likelihood of overspecification, and therefore an unnecessarily high cost, is likewise reduced. The user will also be aware that specific parameters, test conditions, test circuits, and even definitions may vary from manufacturer to manufacturer. For practical production reasons, parameters may not be tested in the same manner for all converter types, even those supplied by the same manufacturer. Using information in this note, the user should, however, be able to sort out and understand those specifications (from any manufacturer) pertinent to his needs.

# MM5865 Universal Timer Applications



## Introduction

A single chip universal counter and timer is now available from National Semiconductor Corporation through distributors of their products.

The MM5865 universal timer contains, in one 40-pin package, two 4-digit counters, oscillator, 18-stage divider, multiplexer, and all the logic required to control the counters, blank leading zeros, compare the two counters, program one of the counters, and cascade two MM5865 integrated circuits.

The MM5865 provides input pins for seven modes of timing and/or counting operations. When the chip is used as a timer, two input pins may be programmed to provide a display resolution of 0.01 second, 0.1 second, 1 second, or external clock. In addition, the modulo by which the counters divide may be programmed using three divide scaler input pins.

The outputs include the comparator output, multiplexed BCD segment outputs, and digit enable. The BCD segment outputs interface directly with the MM14511 (CD4511), a BCD to 7-segment latch/decoder/driver which interfaces with an LED display. The digit enable outputs of cascaded MM5865s interface directly with a DS8863 (DM8863), an MOS to LED 8-digit driver. A single MM5865 interfaces directly with a DS8877 or DS75492 6-digit driver.

When a suitable crystal is used with the MM5865 oscillator, the counters of a single chip (or those of two chips cascaded) may be used as timers with the following functions:

1. Counter 2: Start-Stop timing
   Counter 1: Total elapsed time
2. Counter 2: Start-Stop timing
   Counter 1: Total accumulated time
3. Counter 2: Sequential event timing
   Counter 1: Total elapsed time
4. Counter 2: Split-timing with total elapsed time
   Counter 1: Not actively used
5. Counter 2: Total accumulated time
   Counter 1: Total elapsed time
6. Counter 2: Up counter
   Counter 1: Programmable counter
7. Counter 2: Programmable down counter
   Counter 1: Not actively used

Therefore, one or two MM5865s along with two other integrated circuits and a 4- or 8-digit display may be used in the following applications:

1. Photographic enlarger timer, with each digit individually programmable
2. Stopwatch
3. General purpose timer
4. Event timer/counter
5. Rally timer
6. Navigational timer
7. Industrial timer/counter

The MM5865 may also be used as a frequency counter, or it may be used as the time reference of a larger frequency counter. The maximum oscillator frequency of the MM5865 is 80kHz; the maximum clock input frequency is 100kHz.

## How the MM5865 Operates

As can be assumed from the brief description above, the MM5865 is a very powerful integrated circuit, capable of many applications. Therefore, in order to fully stimulate the imagination of readers, its repertoire will be presented in detail.

A block diagram of the MM5865 universal timer is shown in *Figure 1*, and the connection diagram is shown in *Figure 2*. As nearly as possible, all technical terms in the following discussion conform to definitions presented in the *Radio Shack Dictionary of Electronics*, edited by Rudolf F. Graf.

### Multiplexer

Because of the internal multiplexer, only one BCD to 7-segment latch/decoder/driver need be used to interface one or two MM5865s to a suitable display. The multiplexer may be controlled in three ways.

An externally generated multiplex frequency may be applied to the Multiplex Input pin of the MM5865. An external clock is then applied to the Clock Input pin. (For example, an LM555C may be used as a square-wave oscillator to provide the necessary input to pin 23.)

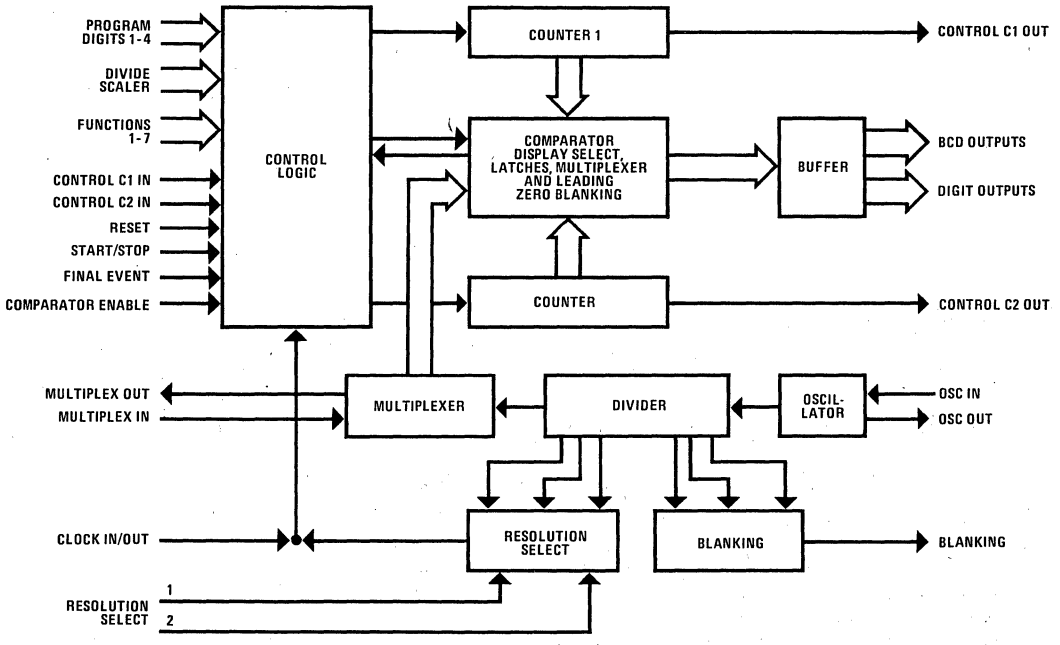



Figure 1. Internal block diagram of the MM5865 Universal Timer.

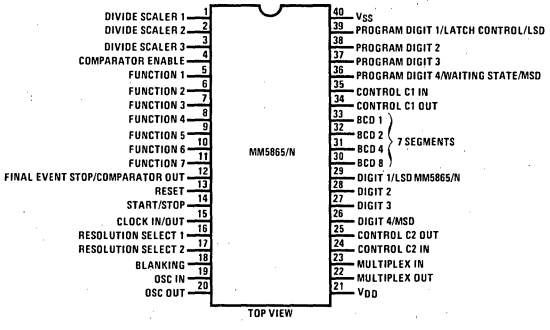



Figure 2. MM5865 connection diagram.

When an external multiplex rate is applied to the Multiplex Input pin, the Multiplex Output pin must be connected to $V_{SS}$, and the Oscillator In, Oscillator Out, and Blanking pins should be floating. The multiplex rate inside the chip is one fourth the frequency applied to the Multiplex Input pin. In this mode of operation two MM5865s may not be cascaded. In fact, to make use of the Multiplex Output pin, the Multiplex Input pin must be connected to $V_{DD}$. The frequency at the Multiplex Output pin is the same as that applied to the Multiplex Input pin.

The multiplexer may also be controlled by using internal MOS circuits to form a crystal controlled oscillator. To form the oscillator a crystal, two capacitors, and one resistor must be added externally. One of the capacitors should be variable to allow precise frequency settings. When these external components are connected to the Oscillator Input and Oscillator Output pins, the Multiplex input pin must be connected to $V_{DD}$.

When the input clock is at a constant frequency above 400Hz the Multiplex Input pin may be connected to the Clock Input pin. In this mode of operation the input clock which is being counted is also used as the externally generated multiplex frequency. The multiplex rate inside the chip will be one fourth the clock input frequency as described above.

### Oscillator

*Figure 3* shows how external components may be connected to the Oscillator Input and Output pins. A frequency counter used to adjust the frequency of the oscillator may be connected to the Oscillator Output pin through a 50pF capacitor.

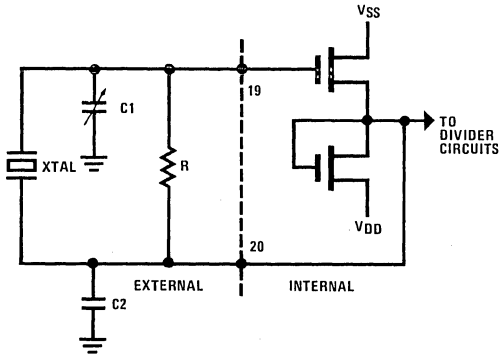



**Figure 3. Crystal oscillator connections.**

### Divider

The divider stages produce the blanking output by dividing the oscillator input frequency by 41. This output is used to blank the display at the beginning and end of each digit time to allow for internal delay between two cascaded chips. The display is blanked when the Blanking Output is at $V_{DD}$.

The divider stages then divide the blanking output by 2 to generate the Multiplex Output. The frequency which appears at the Multiplex Output pin is further reduced in frequency by the divider stages so that the Resolution Select pins may be used to program the resolution of the display. *Table I* shows how these two inputs are used to select the frequency of the internal clock pulses to be applied to the two counters. The frequencies and display resolutions for an oscillator frequency of 32.8kHz are given.

**Table I. Resolution Select Code. A zero indicates that the pin is left floating (or connected to $V_{DD}$); a one indicates that the pin is connected to $V_{SS}$. Note that when an external clock is applied to pin 15, pins 16 and 17 must be connected to $V_{SS}$.**

| Resolution Select | | Frequency of Clock to Counters | Display Resolution |
|---|---|---|---|
| Pin 16 | Pin 17 | | |
| 0 | 0 | 100Hz | 0.01sec |
| 0 | 1 | 10Hz | 0.1 sec |
| 1 | 0 | 1Hz | 1 sec |
| 1 | 1 | External | – |

The Clock Input/Output pin is either an input or an output depending on the code at the Resolution Select input pins. If the pin is used as an output it will output the clock frequency selected by the program applied to pins 16 and 17. When it is used as an input an external clock must be used to clock the counters.

### Control Logic

The block labeled "Control Logic" contains the logic required to select one of the seven functions, reset all logic and counters, start and stop the counters, indicate that a final event has occurred, and display counter 2 in Functions 3 and 4.

The selection of a function is accomplished by connecting one of the seven function pins to $V_{SS}$; the other six function pins are left floating.

The Reset Input will reset all logic and counters in Functions 1-5 and Function 7. In Function 6, Reset will reset logic and counter 2, but not counter 1. For reset to occur the Reset pin must be momentarily connected to $V_{SS}$. Internal control logic provides power-on reset, however, to insure proper power-on resetting of all logic and the counters a 10μF, 35V Solid Tantalum Capacitor (Allied #852-5680) should be used across the $V_{SS}$-$V_{DD}$ power busses.

In Function 6, the Reset Input pin may be connected to the Comparator Output pin in order to automatically reset logic and counter 2. When this connection is made, a Start/Stop transition is all that is needed to repeat the up count of counter 2.

The Start/Stop Input is used to control the counters by momentarily connecting pin 14 to $V_{SS}$. The manner in which this input affects the counters during the execution of each function will be explained as the descriptions of the functions are given.

The Final Event Stop/Comparator Output pin is used to indicate to the circuit that no more events will be timed or counted. Final Event Stop affects the circuit when it is momentarily connected to $V_{SS}$. When this pin is used as the comparator output, a $V_{SS}$ level at the pin indicates comparison between the two counters.

### Additional Control Logic

The three Divide Scaler inputs permit the counters to be programmed to count in Modulo 6 or Modulo 10. *Table II* shows the possible codes which may be applied to the Divide Scaler pins. A zero indicates that the pin is left floating (or connected to $V_{DD}$); a one indicates that the pin is connected to $V_{SS}$.

**Table II. Divide Scaler Code**

| Divide Scalers | | | Modulo | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Counter 1 | | | | Counter 2 | | | |
| Pin | | | Digit | | | | Digit | | | |
| 1 | 2 | 3 | 4 | 3 | 2 | 1 | 4 | 3 | 2 | 1 |
| 0 | 0 | 0 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 1 | 0 | 0 | 6 | 10 | 10 | 10 | 6 | 10 | 10 | 10 |
| 0 | 1 | 0 | 10 | 6 | 10 | 10 | 10 | 6 | 10 | 10 |
| 1 | 1 | 0 | 10 | 10 | 6 | 10 | 10 | 10 | 6 | 10 |
| 0 | 0 | 1 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 1 | 0 | 1 | 10 | 10 | 10 | 10 | 6 | 10 | 10 | 10 |
| 0 | 1 | 1 | 10 | 10 | 10 | 10 | 10 | 6 | 10 | 10 |
| 1 | 1 | 1 | 10 | 10 | 10 | 10 | 10 | 10 | 6 | 10 |

A zero indicates that the pin is left floating (or connected to $V_{DD}$); a one indicates that the pin is connected to $V_{SS}$.

For example, if the Resolution Select pins are programmed to give a 1 second display resolution (code "10") in a stopwatch application, and if the Divide Scaler code is "110," then the maximum possible count for both counters 1 and 2 would be 9959 (99 min, 59 sec). This means that the unit minutes display will advance by one digit every 60 seconds.

Connecting pin 4 to $V_{SS}$ enables the comparator. In functions 1-5 the Comparator Enable pin must be left floating (or connected to $V_{DD}$). In function 6 the Comparator Enable pin must be connected to $V_{SS}$ after digit programming; if the Comparator Enable pin is connected to $V_{SS}$ (comparator enabled) at power on, the Reset pin must be momentarily connected to $V_{SS}$ before a Start/Stop transition will begin the counter 2 count-up.

In function 7, if the Comparator Enable pin is floating (or connected to $V_{DD}$) when power is applied to the chip, or when the function switch is switched to function 7, the Comparator Enable pin must be connected to $V_{SS}$ after digit programming as in function 6; however, in function 7, if the Comparator Enable pin is connected to $V_{SS}$ (comparator enabled) at power on (or when the function switch is switched to function 7), the comparator must be disabled by 1) disconnecting the Comparator Enable pin from $V_{SS}$, and 2) momentarily connecting the Reset pin to $V_{SS}$; this must be done before the digits are programmed. This is necessary, of course, because connecting the Reset pin to $V_{SS}$ after digit programming will simply reset counter 1 to "0000." In function 6, a Reset transition after digit programming does not reset counter 1 to "0000."

In addition, the Control C1 In pin (pin 35) must be floating (or connected to $V_{DD}$) during digit programming in function 7. After digit programming, the Control C1 In pin must be connected to $V_{SS}$ before the count-down begins. A DPDT, Center "OFF" switch connected as shown in *Figure 4*, may be used to control both the Comparator Enable pin and the Control C1 In pin. In one position the DPDT switch connects the Control C1 In pin to $V_{SS}$ for functions 1-5. Digit programming may be accomplished in function 7 by placing the switch in the Center "OFF" position. In the third position both the Comparator Enable and the Control C1 In pins are connected to $V_{SS}$ for functions 6 and 7.

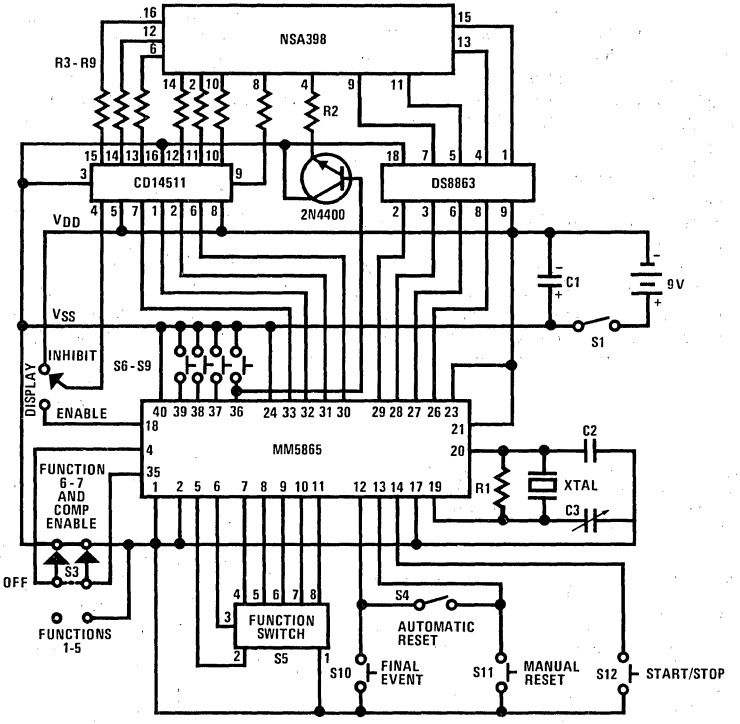



**Figure 4. Stopwatch/Timer application showing the connections for a single MM5865. Two cascaded MM5865s may also be used, as described in the text.**

Pins 36 - 39, the Program Digit 1 - 4 pins, are used to program a desired count into counter 1 when using functions 6 and 7. When any of the four Program Digit pins are connected to $V_{SS}$, the display digit of counter 1 associated with that pin advances at a 2.5Hz rate (assuming the oscillator frequency is 32.8kHz). The Program Digit 1 pin advances the least significant digit of counter 1; the Program Digit 4 pin advances the most significant digit. There is no carry over from digit to digit, and only one Program Digit Input may be connected to $V_{SS}$ at a time.

The Program Digit 1 pin also functions as a counter 2 latch control in functions 3 and 4. In functions 3 and 4, momentarily connecting the Program Digit 1/Latch Control pin to $V_{SS}$ permits the display to show counter 2 counting.

The Program Digit 4 pin also serves two purposes; in functions 1 - 5 this pin indicates that the chip has been reset and is in the standby mode at power-on. Visual indication of this condition may be accomplished by connecting a transistor between the Program Digit 4/ Waiting State Indicator pin and the Segment DP Anode of a multiplexed display. With the transistor connected as shown in *Figure 4*, the Waiting State Indicator pin will be at $V_{SS}$ at power-on until a Start/Stop transition occurs. After a Start/Stop transition occurs, the Waiting State Indicator pin will remain at $V_{DD}$ until power is removed from the chip.

**Leading Zero Blanking**

In functions 1 - 5, leading zeros are blanked for both counters. In functions 6 and 7, counter 2 has leading zero blanking but counter 1 does not. At power-on the display is blank (or all decimal points if the Waiting State Indicator pin is used) in functions 1 - 5; all zeros are displayed in functions 6 and 7.

**Control C1, C2 In and Control C1, C2 Out**

These four pins are used to cascade two chips together. In this mode of operation the primary MM5865, which is directly controlled by the crystal oscillator, connects to another MM5865 in the following manner: the Control C1 In pin of the primary chip is connected to $V_{SS}$ except during digit programming in function 7; the Control C1 Out pin connects to the Control C1 In pin of the other MM5865; the Control C2 In pin of the primary chip is connected to $V_{SS}$; the Control C2 Out pin connects to the Control C2 In pin of the other MM5865; the Control C1 Out and the Control C2 Out pins of the second chip are left floating.

When the Control C1 In pin is floating (or connected to $V_{DD}$), the clock pulses to counter 1 are inhibited. When the Control C1 In pin is connected to $V_{SS}$, counter 1 is enabled. Control C1 Out is at $V_{SS}$ when counter 1 is at its maximum count, and it is floating at all other times. The Control C2 pins affect counter 2 in a similar manner.

Other possible connections between the two chips are: 1) all function pins connected together, 2) pins 12, 13, 14, and 15 connected together, 3) all BCD pins connected together, and 4) pins 39 connected together in functions 1 - 5 only.

When two MM5865s are cascaded as described above, eight momentary switches or individual electrical signals must be provided if every digit of the display is to be programmable. In addition, another switch would have to be provided to break the pin 39 connection between the two chips in functions 6 and 7. Of course, all of the switching action could be provided by one ganged rotary switch if desired; even the function 6 Reset to Comparator Out connection could be accomplished if the proper switch were used.

**Electrical Characteristics**

The maximum supply voltage which may be connected between $V_{SS}$ and $V_{DD}$ ($V_{DD}$ = 0V) is 20V. National specifies that the minimum voltage at which the chip will operate is 7V; however, some chips will operate well down to $V_{SS}$ = 5V. With a 9V transistor battery used as the power supply, and display inhibited, the power supply current will be approximately 7mA to 15mA for a one-chip stopwatch.

The maximum input frequency at the oscillator is 80kHz; however, the oscillator and dividers are designed for stopwatch applications using a 32.8kHz crystal. (A 32.768kHz crystal, available from Quest Electronics, P.O. Box 4430 E, Santa Clara, CA 95054, may be used without much loss in accuracy.)

Drivers must be provided for the Digit and BCD Outputs. Two MM5865s interface directly with the MM14511 Segment Driver and the DS8863 Digit Driver. A DS8877 or DS75492 Hex Digit Driver may be used with a single MM5865.

**The Seven Functions**

The one-chip circuit shown in *Figure 4* indicates all connections necessary to employ the MM5865 as a 4-digit stopwatch/timer. The seven available functions will be described using this figure, in which the desired function is selected by switching S5. When necessary, refer also to *Figures 1* through *3*.

**Function 1**

In function 1, at power-on (S1 closed) four decimal points are visible on the display, indicating that the counters have been reset, but not necessarily all logic. If the Comparator Enable pin is connected to $V_{SS}$ (S3 in Function 6 - 7 position) at power-on, a Start/Stop transition (obtained by momentarily closing S12) will cause the decimal points to disappear from the display; however, the chip will not begin counting. First it is necessary to place S3 in the Functions 1 - 5 position, then to reset the logic (by momentarily closing S11).

Once all logic is reset (either by applying power with S3 in the Functions 1 - 5 position or by the method discussed above), a Start/Stop transition will cause both counters to begin counting up. The up-count of counter 2 is displayed, the least significant digit advancing at a 1Hz rate. A second Start/Stop transition inhibits the clock pulses to counter 2 and stores and displays the contents of counter 2; however, counter 1 continues to count. A third Start/Stop transition resets counter 2, enables clock pulses to counter 2 and, again, displays counter 2 counting up. Subsequent Start/Stop transitions repeat this sequence. Counter 1 continues to count, from the time of the first Start/Stop transition, until the occurrence of a Final Event Stop transition (obtained by momentarily closing S10). A Final Event

Stop transition inhibits the clock pulses to both counters and displays counter 2. After this Final Event Stop transition has occurred, a Start/Stop transition switches the display from counter 2 to counter 1. Each subsequent Start/Stop transition alternately displays one of the counters.

To summarize, in function 1 both counters start counting up with an initial Start/Stop transition. Counter 1 continues to count (recording total elapsed time) until a Final Event Stop transition. Counter 2 (alternately) starts, then stops counting with each Start/Stop transition (timing as many intervals as desired), until a Final Event Stop transition. Any time a Reset transition occurs both counters are reset to "0000" and the display blanks.

**Function 2**

The only difference between functions 1 and 2 is that in function 2, whenever a Start/Stop transition inhibits the clock pulses to counter 2, the clock pulses to counter 1 are also inhibited. Start/Stop transitions which reset counter 2 and enable clock pulses to counter 2 also enable clock pulses to counter 1; counter 1 does not reset, however. The up-count in counter 1 resumes at the stored count; therefore, counter 1 records total accumulated time.

**Function 3**

In function 3 the power-on conditions are the same as those in functions 1 and 2. Once all logic is reset a Start/Stop transition causes both counters to begin counting up Counter 2 is displayed counting. A second Start/Stop transition stores and displays the contents of counter 2, resets counter 2, and initiates a new up-count. However, the new up-count is not displayed. Counter 1 continues to count. The initial count remains displayed until a third Start/Stop transition. This third Start/Stop transition and subsequent Start/Stop transitions repeat the sequence described above, indicating the length of time between successive Start/Stop transitions.

The occurrence of a Latch Control transition (obtained by momentarily closing S5) any time after the second Start/Stop transition will cause counter 2 to be displayed while counting. The count will continue to be displayed until a Start/Stop transition. This Start/Stop transition also stores and displays the contents of counter 2 and then resets counter 2. As before, counter 1 continues to count, but counter 2 begins a new count.

A Final Event Stop transition inhibits the clock pulses to both counters and displays the contents of counter 2. A Start/Stop transition occurring after the Final Event Stop transition switches the display from counter 2 to counter 1. Repetitive Start/Stop transitions switch the display between counter 2 and counter 1. Any time a Reset transition occurs, both counters are reset to "0000" and the display blanks.

**Function 4**

In function 4 the power-on conditions are the same as those in functions 1-3. Once all logic is reset a Start/Stop transition causes counter 2 to begin up-counting. Counter 2 is displayed counting. A second Start/Stop transition stores and displays the contents of counter 2. Subsequent Start/Stop transitions update the display of counter 2. A Latch Control transition will display counting until the occurrence of a Start/Stop transition. This Start/Stop transition, following the Latch Control transition, does not reset counter 2 as it does in function 3. Rather, counter 2 continues to count up. A Final Event Stop transition inhibits the clock pulses to counter 2 and displays the contents of counter 2. A Reset transition at any time resets counter 2 to "0000."

**Function 5**

Again, in function 5 the power-on conditions are the same as those in functions 1-4. Once all logic is reset a Start/Stop transition causes both counters to begin counting up. Counter 2 is displayed counting. A second transition on the Start/Stop pin inhibits the clock pulses to counter 2, and the contents of counter 2 are displayed. Counter 1 continues to count. A third Start/Stop transition enables the clock pulses to counter 2; counter 2 resumes counting where it left off, and counter 2 is displayed counting.

Subsequent Start/Stop transitions repeat this sequence with counter 1 counting continuously. A Final Event Stop transition inhibits the clock pulses to both counters and displays counter 2. A Start/Stop transition switches the display from counter 2 to counter 1. Repetitive Start/Stop transitions switch the display between counter 2 and counter 1. A Reset transition at any time resets both counters to "0000."

**Function 6**

At power-on in function 6, counter 1 is displayed with "0000." If the comparator is enabled (S3 in the Function 6-7 position) at power on, a Reset transition (obtained by momentarily closing S11) is necessary before a Start/Stop transition can begin the counter 2 count-up.

Counter 1 is programmed to the desired count by holding each of the four Digit Programming Switches Closed in turn. The comparator must then be enabled by placing S3 in the Function 6-7 position (unless it was already enabled at power-on). Counter 2 is displayed counting up beginning with a Start/Stop transition. When counter 2 is coincident with counter 1, the clock pulses to counter 2 are inhibited, the contents of counter 2 are displayed, and the Comparator Output is enabled. A Reset transition after the counter 2/counter 1 coincidence disables the Comparator Output and displays counter 1 with the programmed time. The Reset transition can be obtained either by momentarily closing S11 or by connecting the reset Input pin to the Comparator Output pin after Digit Programming so that logic and counter 2 are reset automatically whenever counter 2 is coincident with counter 1.

After each Reset transition, subsequent Start/Stop transitions repeat the sequence. Counter 1 may be reprogrammed after any Reset transition, if desired. If a Reset transition occurs while counter 2 is counting up, the clock pulses to counter 2 are inhibited, counter 2 is reset, and counter 1 is displayed with the programmed time.

If a Start/Stop transition occurs while counter 2 is counting up, the clock pulses to counter 2 are inhibited and counter 1 is displayed with the programmed time. With the next Start/Stop transition, counter 2 resumes counting where it was stopped.

If the Reset Input pin is not connected to the Comparator Output pin and if a Final Event Stop transition occurs while counter 2 is counting up, the clock pulses to counter 2 are inhibited and the contents of counter 2 are displayed. The next Start/Stop transition displays counter 1 with the programmed time. Repetitive Start/Stop transitions switch the display between counter 2 and counter 1. A Reset transition followed by a Start/Stop transition starts the counter 2 up-count sequence again.

In function 6, and also in function 7, the digit which is preprogrammed to count in Modulo 6 cannot, of course, be programmed to a digit greater than 5.

**Function 7**

In function 7 counter 1 is displayed with "0000" at power-on. If S3 is in the Function 6-7 position at power-on, it must be placed in the "OFF" position; then S11 must be momentarily closed. Counter 1 is set to a specific count by holding each of the four Digit Programming Switches closed in turn; then the Comparator must be enabled by placing S3 in the Function 6-7 position.

Counter 1 counts down from the set count beginning with a Start/Stop transition. When counter 1 counts down to zero the clock pulses to counter 1 are inhibited and the Comparator Output is enabled. This is not repeatable without a new count being entered into counter 1. A Final Event transition halts the counter 1 down-count, and subsequent Start/Stop transitions have no effect on counter 1 or counter 2. A Reset transition resets counter 1 to "0000."

**Peripheral**

The other components shown in *Figure 4* consist of input/output interfaces between the user and the MM5865. The crystal used in this stopwatch/timer circuit is a watch crystal cut to oscillate at 32.768kHz. (A 32.8kHz crystal would be best.) This means that the blanking frequency is 799.2Hz, the multiplex frequency is 399.6Hz, and the clock frequency to the counters is 0.99902Hz.

The oscillator frequency may be adjusted by connecting a counter to pin 20 of the MM5865 through a 50pF capacitor and then varying the capacitance of C3. Any attempt to alter the values of R1, C2, or C3 will probably fail; that is, the oscillator will probably not oscillate.

Most of the switches which control the MM5865 are momentary push-buttons which are available from many sources. The function switch, however, is a very small 8-position switch in a TO-5 package; it is available from James Electronics, P.O. Box 822, Belmont, CA 94002.

The 2N4400 (a 2N3904 can also be used) drives the decimal point anode of the display and is itself driven by the Waiting State output of the MM5865.

The MM14511 provides the functions of a 4-bit storage latch, an 8421 BCD-to-seven segment decoder, and an output drive capability of 25mA. The DS8863 is an 8-digit driver; each driver is capable of sinking up to 75mA. The MM14511 may be operated at supply voltages up to 15V; however, the DS8863 cannot be operated with supply voltage greater than 10V. For operation with supplies up to 18V, the DS8963 is a direct replacement for the DS8863.

The NSA398 is a 9-digit common cathode LED numeric display with a 1/8-inch character height. Eight inputs are provided for selection of the appropriate segments and decimals (anodes) and nine inputs for digit (cathodes) selection. The anodes are internally interconnected for multiplexing. The NSA398 has a red faceplate which provides excellent visual contrast and ease of visibility over a wide angle. *Figure 5* shows the physical dimensions and pin connections of the NSA398.

## Practical Applications of the Stopwatch/Timer

Now that the basic operation of the MM5865 has been presented, it is possible to examine practical applications of the seven function universal timer shown in *Figure 4*. This timer, as shown, has a maximum timing capability of 99 minutes, 59 seconds. If another MM5865 is added to the circuit, this timing capability may be extended to 99 hours, 59 minutes, 99.99 seconds. For very accurate timing, the crystal should be cut to oscillate at 32.8kHz, and the oscillator frequency should be precisely tuned to 32.8kHz.

When the stopwatch/timer is being used to time any event, the display should be disabled with S2 as much as possible so that battery power will be conserved.

Function 1 may be used to time two events occurring simultaneously in the following manner. A driver often travels from his home to a city some hours away. On the way he passes a small town about halfway between his home and the city. He wishes to know how long it takes him to travel from his home to the small town, how long it takes to travel from the town to the city, and finally, how long it takes him to travel from his home to the city.

At the beginning of the trip the driver presses the Start/Stop switch. The display begins to record the time accumulating in counter 2. As he passes through the small town he presses the Start/Stop switch again and records the traveling time from his home to the town. Then he presses the Start/Stop switch again. As he arrives at the city he presses the Final Event Stop switch and records the time shown in the display as being the traveling time from the town to the city. He then presses the Start/Stop switch and sees in the display the traveling time from his home to the city.

Function 2 may be used to record the total accumulated time of several events while each event is being timed individually. For example, a television repairman spends his day ordering parts, talking to customers, and repairing televisions on the bench. He wants to record the time he spends repairing each set so that customers may be properly billed, and he wishes to record his total bench time for the day.

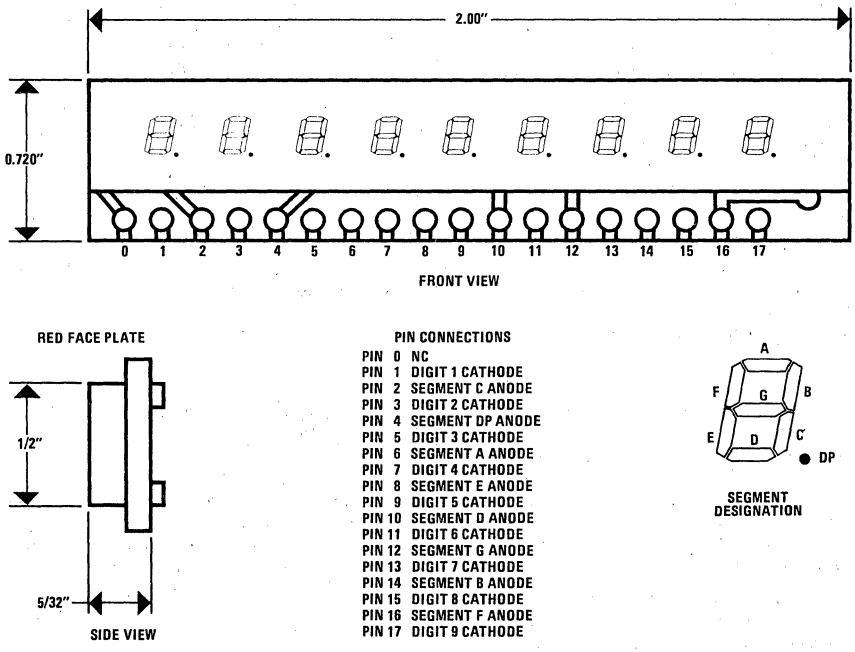



**Figure 5. Physical dimensions and pin connections of the NSA398.**

At the beginning and end of every bench job he presses the Start/Stop switch to record the time for each job. At the end of his day he presses the Final Event Stop switch, then the Start/Stop switch to record his total bench time.

As an example of a function 3 application, consider an assembly line position at which a worker must fasten three parts to a piece of equipment. A supervisor wishes to record the time it takes the worker to fasten each part and the amount of time the equipment spends at this position.

As the worker receives the piece of equipment, the supervisor presses the Start/Stop switch. The display begins counting up. As the worker finishes with the first part, the supervisor presses the Start/Stop switch. This time will remain in the display until the next Start/Stop transition; the supervisor therefore has a chance to record the first event time.

As the worker finishes with the second part, the supervisor presses the Start/Stop switch again and records the time of the second event. After the worker finishes with the third part the supervisor presses the Final Event Stop switch. The display will show the third event time. The supervisor can then press the Start/Stop switch to record the total time this worker handled the equipment.

With function 4, the total time of an event may be accumulated, and the display may be updated while counter 2 is accumulating the total time. For example, a long distance runner desires to pace himself over a 5-mile run. As he starts out he presses the Start/Stop switch. Then, as he passes known checkpoints, he presses the Start/Stop switch to update the display and note the time of arrival at each check point. At the end of the 5-mile run he presses the Final Event Stop switch to record the total time for the run.

Function 5 may be used to record both total accumulated time and total elapsed time. As an example of an application of function 5, consider a pilot who wants to record total flying time as well as total trip time.

As the pilot starts out he presses the Start/Stop switch. He then presses the Start/Stop switch each time he lands and each time he resumes flying. At the end of his trip he presses the Final Event Stop switch and records total flying time. He then presses the Start/Stop switch to record total trip time.

With proper interfacing, function 6 can be used as an enlarger timer. A photographer programs the desired printing time into the display with the Digit Programming switches, closes the Comparator Enable switch, and closes the Automatic Reset switch. For each print he

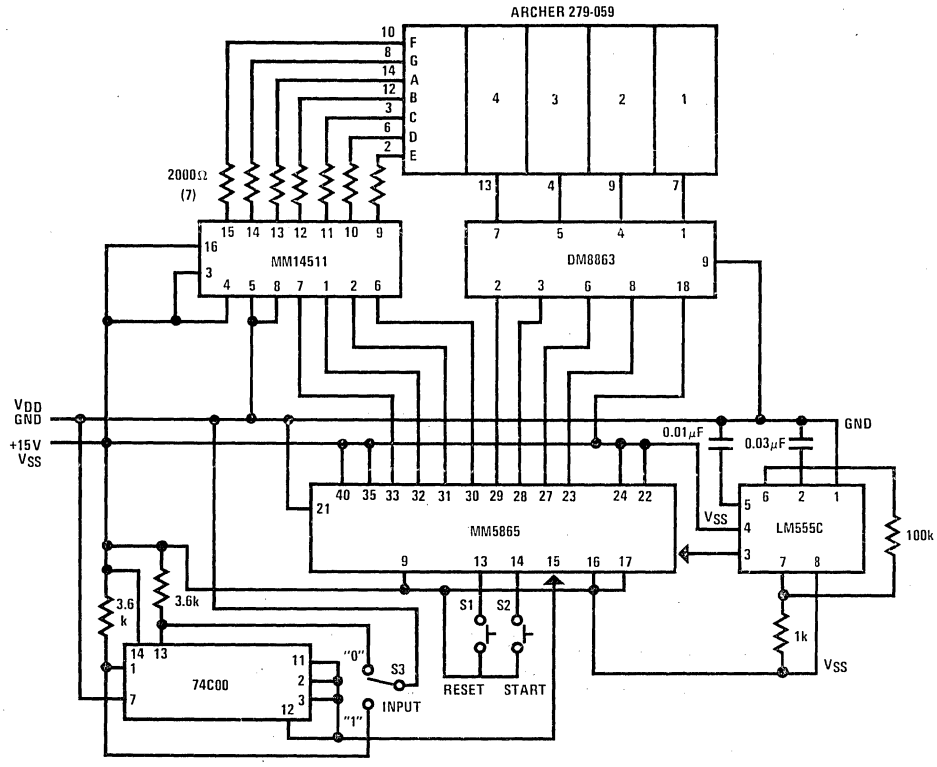



**Figure 6. The MM5865 used in a simple counting circuit.**

simply presses the Start/Stop switch to turn on the enlarger for the desired length of time.

It is not necessary to enable the display while operating the timer. The display must be enabled only to program counter 1. The Reset switch may be pressed at any time to turn off the enlarger. The enlarger may be turned on for adjusting negatives by pressing the Start/Stop switch without enabling the comparator.

With proper interfacing, function 7 may be used as a down-count timer for many applications, including cooking and washing. The desired time is simply programmed into counter 2, the comparator is enabled, and then the Start/Stop switch is pressed. Counter 2 will count down to zero and turn off the appliance.

A few applications (some for which two MM5865s are required) have been presented to illustrate the utility of the MM5865. The Stopwatch/Timer discussed above is but one general application for which the MM5865 may be used.

*Figure 6* shows a simple manual counting circuit in which the MM5865 is used to count the closures of a manual switch. Of course, the manual clock could be replaced by electrical pulses.

The 74C00 in this circuit debounces the switch used as a clock, S3. An LM555 is used to provide a multiplexer input frequency of 233Hz.

The MM5865 is operating in function 5, and displays the up-count of counter 2. After an initial Start/Stop transition, each closure of the manual switch advances the displayed digits by one count. A Reset transition resets counter 2 to "0000."

## Conclusion

The emphasis of this presentation has been on the general timing and programmable capabilities of the MM5865 rather than on specific applications. Because so many functions are available in one package, it is possible to use the MM5865 as a general purpose chip, adding another MM5865 when it is necessary. In most applications only one or several of the seven functions need be used; however, because of its general purpose nature, the MM5865 lends itself well to the concept of quantity purchasing.

A final note: Unless the start pulse is externally synchronized to the clock (available at pin 15 of the MM5865, if the internal oscillator is used), the amount of time which will elapse between the arrival of the start pulse at pin 14 of the MM5865 and the appearance of the first digit in the display will not be equal to the programmed display resolution. It is possible to develop a start pulse that is synchronized to the clock using an MM74C221 Dual Monostable Multivibrator as shown in *Figure 20*. The time constant of R1 - C1 should be equal to the display resolution, the time constant of R2 - C2 should be less than the programmed display resolution, and the time constant of R3 - C3 should be less than the time constant of R2 - C2.

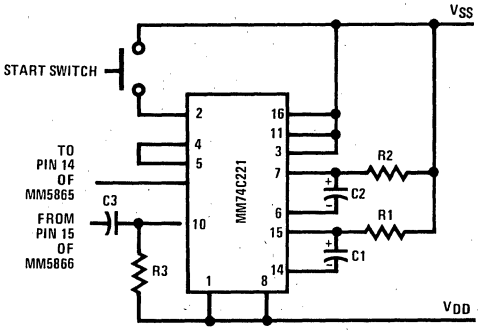



**Figure 21. Start-Pulse Synchronizer. Time constant of R1 - C1 = display resolution. Time constant of R2 - C2 < display resolution. Time constant of R3 - C3 < time constant of R2 - C2.**

# A 4-Digit, 7-Function Stop Watch/Timer







## Introduction

This construction article is the second of a series which is to concentrate on applications of the MM5865 universal timer. The first article, "MM5865 Universal Timer Applications," presented in detail the programmable and functional characteristics of the MM5865.

This second article illustrates the construction and use of a 4-digit, 7-function stopwatch/timer in which the display resolution and counter modulo may be programmed with printed circuit board jumper wires.

Other than switches, all components of the stopwatch/timer are mounted on a glass-epoxy or glass-polyester board which is laminated with 1-ounce copper foil on one side. The board is mounted in the attractive instrument/clock case available from James Electronics.

This instrument/clock case has provisions for the display, precut holes for four calculator-type switches, and a precut line cord hole. In addition, the case is sold with a red display bezel, four rubber feet, and a flip-top to conceal the four switches which may be assembled in the precut holes.

A display resolution of 1 second, 0.1 second, or 0.01 second may be programmed by on-board jumpers or a suitable switch. Furthermore, the counters may be programmed to count in modulo 6 or modulo 10.

When used as a photographic enlarger timer or as an appliance timer, each digit is individually programmable with one of four pushbutton switches. The comparator output of the timer may be coupled to an enlarger/appliance control circuit that can be permanently mounted to the enlarger or appliance.

Applications for the stopwatch/timer include, but are not limited to, the following:

- Laboratory reaction and interval timer
- Photographic enlarger and chemical processing timer
- Stopwatch
- Event timer
- Appliance timer

A simple listing of possible applications for the timer does not adequately describe the enormous power of the instrument. A tabulation of the seven functions which includes a break-out of the functions performed simultaneously by counters 1 and 2 of the MM5865 is much more revealing, and is presented below:

1. Counter 2: Start-stop timing
   Counter 1: Total elapsed time
2. Counter 2: Start-stop timing
   Counter 1: Total accumulated time
3. Counter 2: Sequential event timing
   Counter 1: Total elapsed time
4. Counter 2: Split-timing with total elapsed time
   Counter 1: Not actively used
5. Counter 2: Total accumulated time
   Counter 1: Total elapsed time
6. Counter 2: Up counter
   Counter 1: Programmable counter
7. Counter 2: Programmable down counter
   Counter 1: Not actively used

## Operation

The switches which control the operation of the stopwatch/timer are visible on top of the case shown in the photographs of *Figures 1a* and *1b*. Each switch is indicated in the schematic drawing of *Figure 2*.

In *Figure 1a*, the switch in the rear right hand corner of the case is a 7-position rotary Function Switch (F). At the front of the case the switches are, from left to right, Digit 4 Programming Switch (D4), Digit 3 Programming Switch (D3), Comparator Switch (C), Digit 2 Programming Switch (D2), and Digit 1 Programming (D1)/Latch Control (LC) Switch. Digit 1 is the least significant digit (LSD); Digit 4 is the most significant digit (MSD).

There are four switches under a center flip-cover. These are shown in *Figure 1b*. From left to right they are Final Event Switch (FE), Reset Switch (R), Start/Stop Switch (SS), and Automatic Reset Enable Switch (ARE).

The ARE switch is used only in function 6; it must be OFF for all other functions. The C switch has three positions: Comparator/Count Enable (CCE), used for functions 6 and 7; Program Enable (PE), used for function 7; and Count Enable (CE), used for functions 1 through 5. The D1/LC switch is a dual purpose switch; for functions 3 and 4 it serves as the latch control switch, and for functions 6 and 7 it serves as the Digit 1 programming switch. There is no ON-OFF switch. Power is applied to the stopwatch/timer by plugging the line cord into a 120VAC/60Hz outlet.

*Table I* is a tabulation of the abbreviations used for the switches and the functions to which they apply. If the F switch is set to any of the stop watch functions (1 through 5) when power is initially applied to the stopwatch/timer, the display will remain blank. See "MM5865 Universal Timer Applications" for information on using pin 39 as a power on indicator.

To operate the stopwatch/timer in any of the stopwatch functions, rotate the F switch to one of the stopwatch function positions, place the ARE switch in the OFF position, place the C switch in the CE position, and press the R switch.


(a) (b) (c)

Figure 1. External Photographs of Stopwatch/Timer. a) View of Function Switch, Comparator Switch, and Digit Programming Switches. b) With flip-cover raised, four additional switches are seen. The flip-cover is designed so that a press of the closed cover closes the Start/Stop Switch. c) A miniature jack is mounted at the rear of the case so that a cable may be run to the appliance control box.

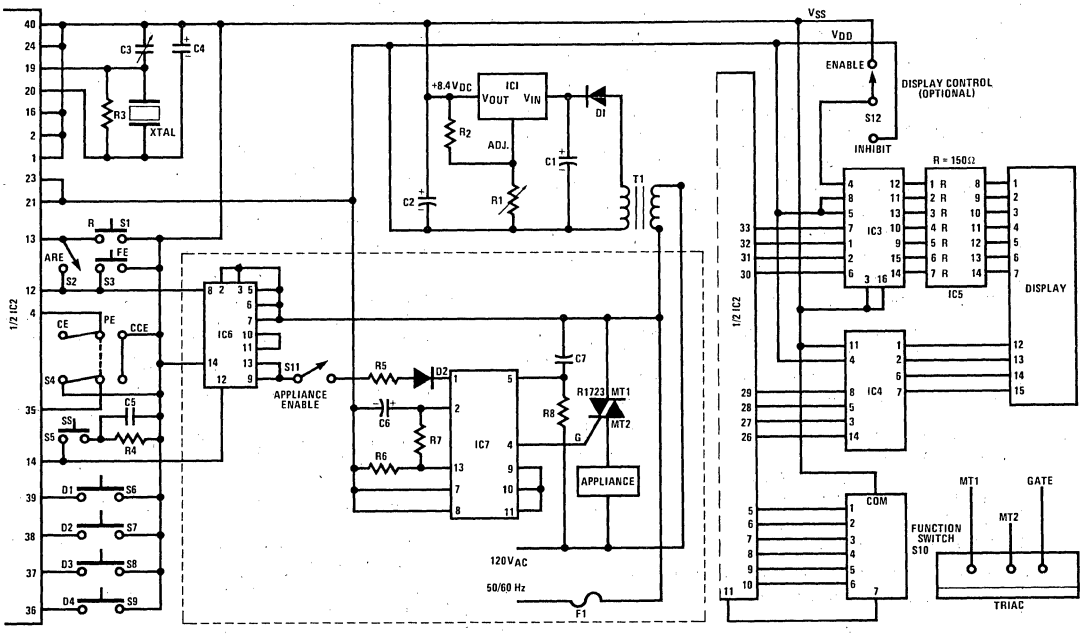



Figure 2. Schematic Diagram of the 4-Digit, 7-Function Stopwatch/Timer. As drawn, the display resolution is 1 second. A SPST switch may be included between pin 16 of IC2 and $V_{SS}$ to provide a display resolution of 0.01 second or 1 second. Another option, shown in the figure, is the Display Control Switch, which may be used to inhibit the display.

Table I. Switch Abbreviations

| Abbreviation | Switch | Functions |
|---|---|---|
| ARE | Automatic Reset Enable | 6 |
| C | Comparator | 1-7 |
| D1 | LSD Programming | 6, 7 |
| D2 | Digit 2 Programming | 6, 7 |
| D3 | Digit 3 Programming | 6, 7 |
| D4 | MSD Programming | 6, 7 |
| F | Function | 1-7 |
| FE | Final Event | 1-5 |
| LC | Latch Control | 3, 4 |
| R | Reset | 1-7 |
| SS | Start/Stop | 1-7 |

Table II. Resolution Select Code. A zero indicates that the pin is left floating (or connected to $V_{DD}$); a one indicates that the pin is connected to $V_{SS}$. Note that when an external clock is applied to pin 15, pins 16 and 17 must be connected to $V_{SS}$.

| Resolution Select | | Frequency of Clock to Counters | Display Resolution |
|---|---|---|---|
| Pin 16 | Pin 17 | | |
| 0 | 0 | 100 Hz | 0.01 sec |
| 0 | 1 | 10 Hz | 0.1 sec |
| 1 | 0 | 1 Hz | 1 sec |
| 1 | 1 | External | – |

Table III. Divide Scaler Code

| Divide Scalers | | | Modulo | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Counter 1 | | | | Counter 2 | | | |
| Pin | | | Digit | | | | Digit | | | |
| 1 | 2 | 3 | 4 | 3 | 2 | 1 | 4 | 3 | 2 | 1 |
| 0 | 0 | 0 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 1 | 0 | 0 | 6 | 10 | 10 | 10 | 6 | 10 | 10 | 10 |
| 0 | 1 | 0 | 10 | 6 | 10 | 10 | 10 | 6 | 10 | 10 |
| 1 | 1 | 0 | 10 | 10 | 6 | 10 | 10 | 10 | 6 | 10 |
| 0 | 0 | 1 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 1 | 0 | 1 | 10 | 10 | 10 | 10 | 6 | 10 | 10 | 10 |
| 0 | 1 | 1 | 10 | 10 | 10 | 10 | 10 | 6 | 10 | 10 |
| 1 | 1 | 1 | 10 | 10 | 10 | 10 | 10 | 10 | 6 | 10 |

A zero indicates that the pin is left floating (or connected to $V_{DD}$); a one indicates that the pin is connected to $V_{SS}$.

Press the SS switch to initiate a sequence of timing series. Press the SS switch again to end a serial (functions 1, 2, 3, 5) and simultaneously initiate a new serial while freezing the display (function 3), or to freeze the display during a continuous count sequence (function 4).

Press the SS switch a third time to initiate a new timing serial (functions 1, 2, 3, 5) or to update the display during a continuous count sequence (function 4). Subsequent presses of the SS switch will repeat the action described above.

Press the LC switch to display a continuing, undisplayed count (functions 3 and 4). Press the FE switch to end a sequence. A final press of the SS switch at the end of a sequence is required to display total elapsed time (functions 1, 3, 5) or total accumulated time (function 2). Subsequent presses of the SS switch after the end of a sequence simply repeat the display of the final serial time, then the total elapsed or total accumulated time.

The operations which may be performed in each function are shown in the flow charts of *Figures 3* through *8*. The first line of type in each PROCESS rectangle indicates a switch or the display upon which an action may be performed. The second line of type indicates the position in which the switch must be placed or the action to be performed. The parallelograms in the flow charts indicate points at which a DECISION must be made. The operation of each function is detailed in the first article of this series.

To operate the timer in function 6, rotate the F switch to function 6, place the C switch in the CCE position, and press the R switch. The display will show four zeros when the R switch is pressed.

The count-up time is programmed into the timer by pressing D1 through D4, one switch at a time, until the desired count-up time appears in the display.

After digit programming, place the ARE switch in the ON position if automatic resetting is desired. The initial press of the SS switch will cause the display to blank, then to indicate the count-up to the programmed time. During the up-count the CA3059 will be enabled, allowing the appliance to be turned on. When the count-up reaches the programmed time, the comparator output will go from 0 volts to 8.4 volts. At this time the CA3059 will be inhibited, and the appliance will turn off. Pressing the R switch any time after the digits have been programmed causes the comparator and counter 2 to reset. Switching the C switch to OFF causes the comparator output pin to go to $V_{DD}$ as long as it is OFF. If the C switch is again placed in the CCE position (before the R switch is pressed), the comparator output pin will go back to $V_{SS}$. Of course, any time the FE switch is pressed the comparator output will go to $V_{SS}$.

If the ARE switch is ON, the count-up sequence may be repeated by pressing the SS switch again. Nothing need be changed until it is necessary to reprogram the digits. When reprogramming is necessary, simply change the time shown in the display to the new time, with the ARE switch in the OFF position, using the digit programming switches. Then press the SS switch to start the up-count. If the ARE switch is OFF, it is necessary to press the reset before starting a new count-up.

To operate the timer in function 7, rotate the F switch to function 7, place the ARE switch in the OFF position, place the C switch in the PE position, and press the R switch. The count-down time is programmed into the timer by pressing D1 through D4, one switch at a time, until the desired count-down time appears in the display. The C switch must then be placed in the CCE position.

Pressing the SS switch will cause counter 1 to begin its down-count from the programmed time to "0000" and will cause the CA3059 to be enabled, turning on the appliance as in function 6. When counter 1 reaches "0000" the CA3059 will be inhibited, turning the appliance off. The down-count is displayed, and may be halted at any time by pressing the FE switch; the down-count may not be resumed. Pressing the R switch any time after digit programming will reset counter 1.

When using function 7, the comparator must be disabled and the R switch must be pressed before digit programming. Then the comparator must be enabled. This is unlike function 6, in which digit programming is allowed at any time, regardless of the state of the comparator. In addition, the ARE switch must not be used in function 7.

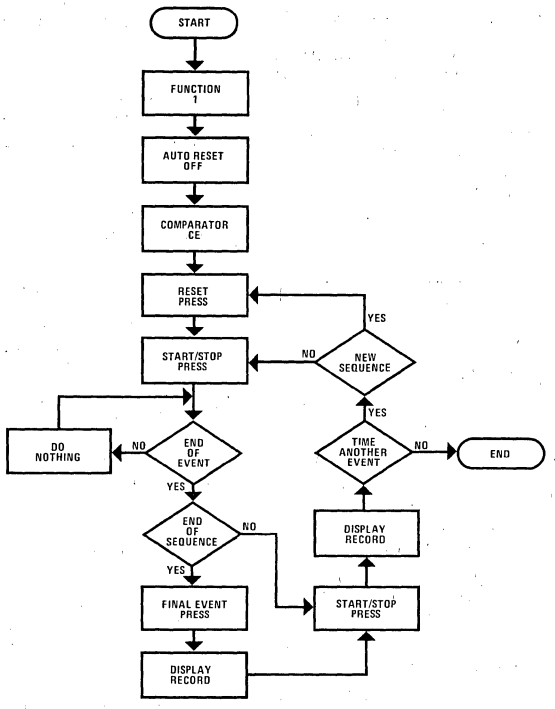



Figure 3. Functions 1 and 2. Pressing START/STOP after FINAL EVENT has been pressed gives Total Elapsed Time in Function 1, Total Accumulated Time in Function 2.

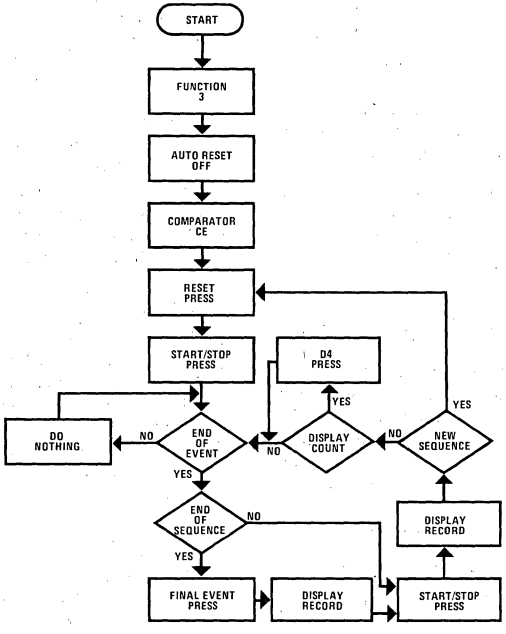



Figure 4. Function 3. Pressing START/STOP after FINAL EVENT has been pressed gives Total Accumulated Time.

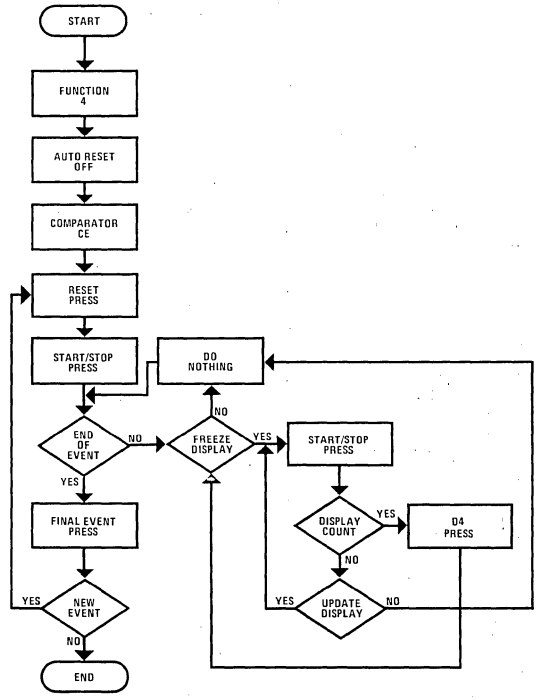



Figure 5. Function 4.

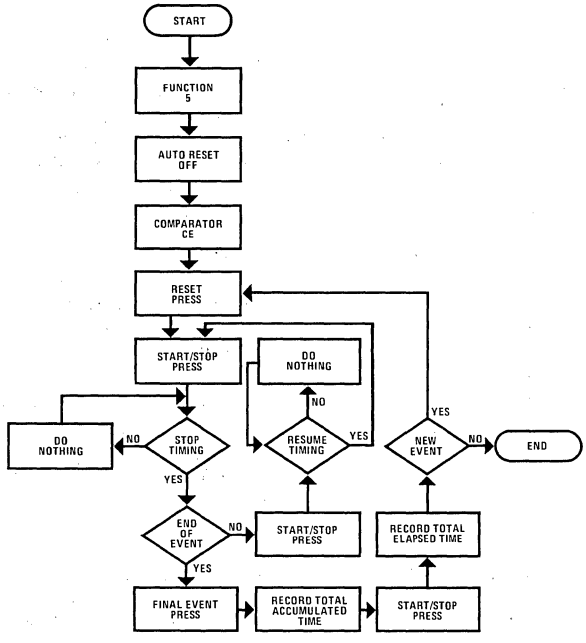



Figure 6. Function 5.

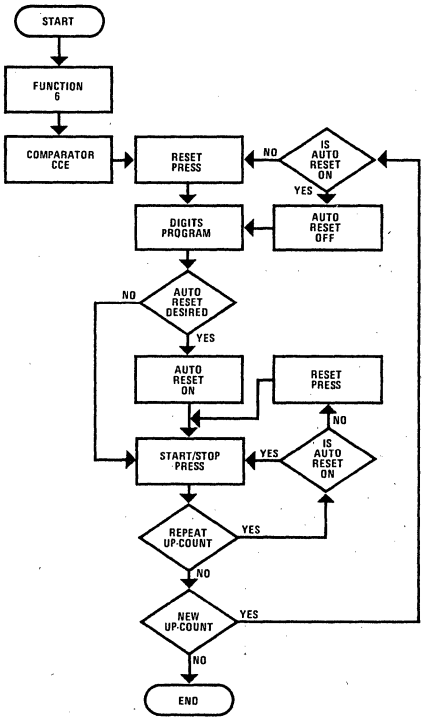



Figure 7. Function 6.

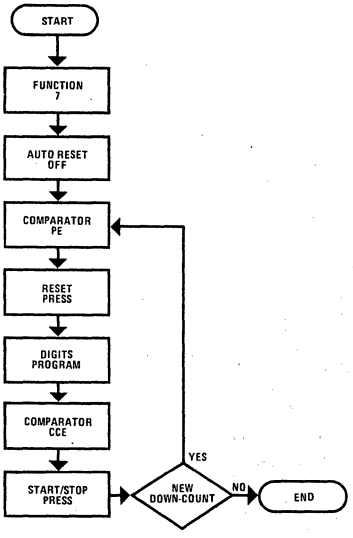



Figure 8. Function 7.

## Interfacing the Stopwatch/Timer with an Appliance Circuit

There are many ways to interface the comparator output with an appliance control circuit. One method of interfacing the MM5865 with an appliance control circuit is shown enclosed in dotted lines in *Figure 2*. *Figure 2* is the schematic diagram of the stopwatch/timer.

The 74C02 has been included as the interfacing element between the comparator output pin and the trigger circuit of a triac. *Figure 9* is a detailed schematic of the 74C02 connections which form a NOR latch.

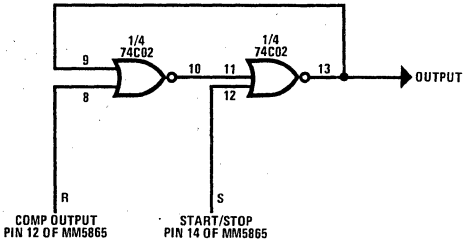



**Figure 9. Detail of the 74C02 NOR Latch. The latch interfaces the MM5865 to the CA3059.**

The appliance control circuit does not cause RFI because the triac is triggered by a zero-voltage switch. Triac firing can be inhibited by the application of a positive (up to 10V) voltage to pin 1 of the CA3059.

When power is initially applied to the stopwatch/timer the S and R inputs of the latch are both "0." When the R switch is pressed, the output of the latch will go to $V_{SS}$, inhibiting the CA3059 pulses to the triac.

When the SS switch is pressed (after digit programming) the output of the latch will go to $V_{DD}$ and the CA3059 will be enabled, turning on the appliance. As the programmed time is reached by counter 2 of the MM5865 (function 6), or as counter 1 reaches "0000" (function 7), the comparator output will go to $V_{SS}$, the output of the latch will go to $V_{SS}$, and the CA3059 will be inhibited, turning off the appliance.

The inhibit level provided by the latch may be removed from the CA3059 by opening the Appliance Enable Switch. This allows the appliance to be turned on for adjustments. For example, when the timer is used with an enlarger, the Appliance Enable Switch permits enabling of the enlarger lamp for focusing and magnification adjustments.

The output of the latch is connected to the appliance control circuit via a tape recorder cable which plugs into a jack mounted at the rear of the stopwatch/timer case and a jack mounted on the appliance control circuit housing. The housing for the appliance control circuit should also have a socket into which the appliance may be plugged, unless a direct connection is desired.

As shown in *Figure 2*, the appliance control circuit consists of a triac and its trigger circuit. When the CA3059 zero voltage switch is enabled, the trigger circuit applies a brief gate signal to the triac for every alternation of the AC line voltage. After the triac is turned on by the gate signal, it remains on for the complete half cycle until the zero-crossing point is reached at the end of the alternation. The appliance receives the full AC line voltage under these conditions.

If the NOR latch inhibits the trigger circuit while the triac is conducting, the triac cuts off when the line voltage approaches zero. It remains off until another gate signal is applied. Therefore, the NOR latch controls the AC input to the appliance.

With the heat sink specified the triac can safely handle appliances rated up to 100 watts (0.83 Amp). For greater appliance loads a larger heat sink should be used. The specified triac is able to handle appliance loads up to 10 Amps. Of course, the fuse must be large enough to handle the current drawn by the appliance. Use a fast blow fuse if possible.

## Construction

The printed circuit board was designed specifically for the James Electronics' instrument/clock case only after assurance that the company has a permanent source for the cases; however, the board may be mounted in any case of sufficient size.

Because the layout of the PC board requires that some traces be proximate, the board must be inspected while it is being etched. During these inspections proper resolution of the traces is maintained, if necessary, by rinsing the board in water and carefully scraping the photoresist from any copper which forms a short circuit between adjacent traces. The scraping is done best with an X-Acto blade. Etching should be continued with frequent inspections.

If the exposure time, the amount of light, and the development time are exactly correct, trace resolution is usually not a problem. However, it is difficult to compute and control these variables without performing many experiments. The inspection method described above can save many boards which otherwise would be lost because of trace resolution defects.

In addition to the care which must be given to the PC board during the etching process, excessive solder should be avoided when soldering to the pads. In case of difficulty with timer operation during the checkout procedure, suspect the board immediately.

Furthermore, no thought should be given to the idea of not using sockets for the integrated circuits. James Electronics has four socket styles. All are adequate except the wire wrap sockets. (The diameter of the wire wrap leads is too large.) However, it is easier to insert and remove ICs from the standard tin and gold sockets.

The drilling guides shown in *Figure 10* indicate all drill sizes for the parts shown in the parts list. Every effort has been made to allow the board to accommodate a variety of components. For this reason, there are extra pads and punch guides on the drilling guides. Refer also to the component layouts shown in *Figure 11*. The boards may be prepared using the X1 positives shown in *Figure 12*.

The bottom half of the James case should be prepared for the board by removing the 6 plastic pegs at the front of the case if they are present. The pegs may be removed by grasping them in the jaws of a long-nose pliers and shaking them from side to side while pulling on the pliers.

The earphone socket should be drilled out from the outside of the bottom half of the case with a 31/64-inch drill bit. This will allow a 7-function rotary switch to be mounted in the right hand (facing the display) corner of the rear section of the top half of the case. When doing this, first press the bit to the 3/8-inch hole in the bottom half of the case, *then* turn on the drill. The bit should slice the earphone socket off with 4 or 5 turns of the chuck.

FIG. 10A

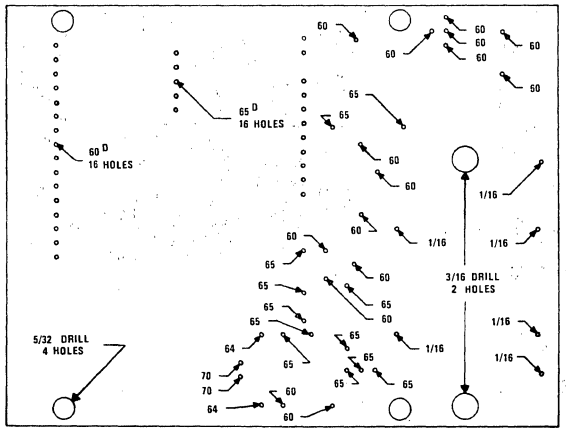



FIG. 10B

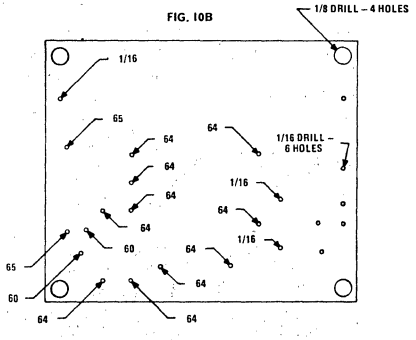



**Figure 10. Drilling Guides for the Printed Circuit Boards. a) Drill sizes and hole locations for the Stopwatch/Timer PC board. b) Drill sizes and hole locations for the appliance control circuit. Dimensions are in inches.**

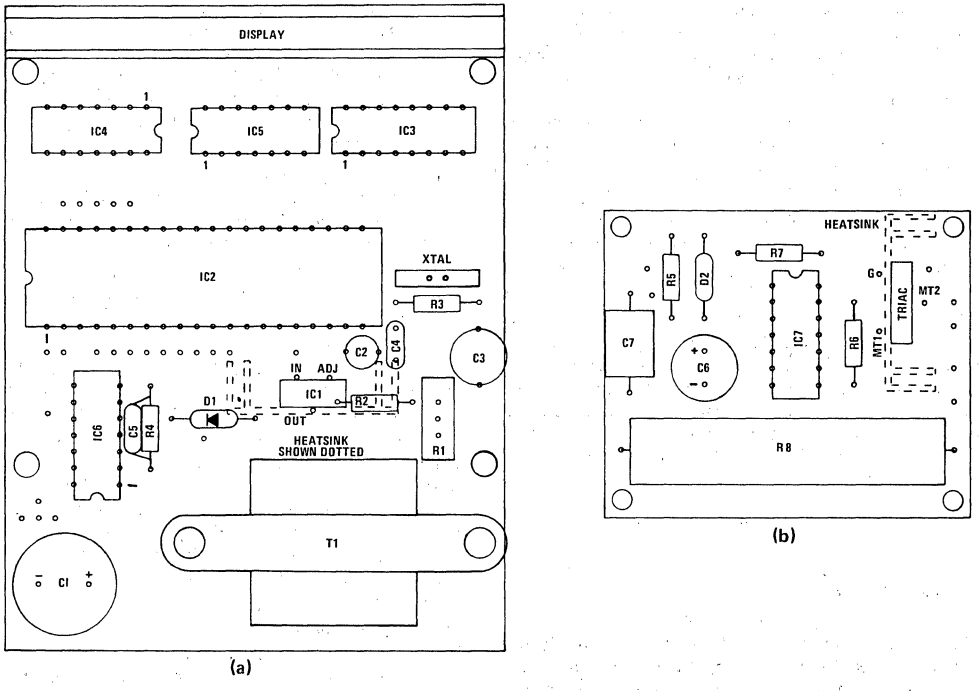



Figure 11. Printed Circuit Board Component Layouts. a) Layout for the Stopwatch/Timer PC board. b) Layout for the Appliance Control PC board.

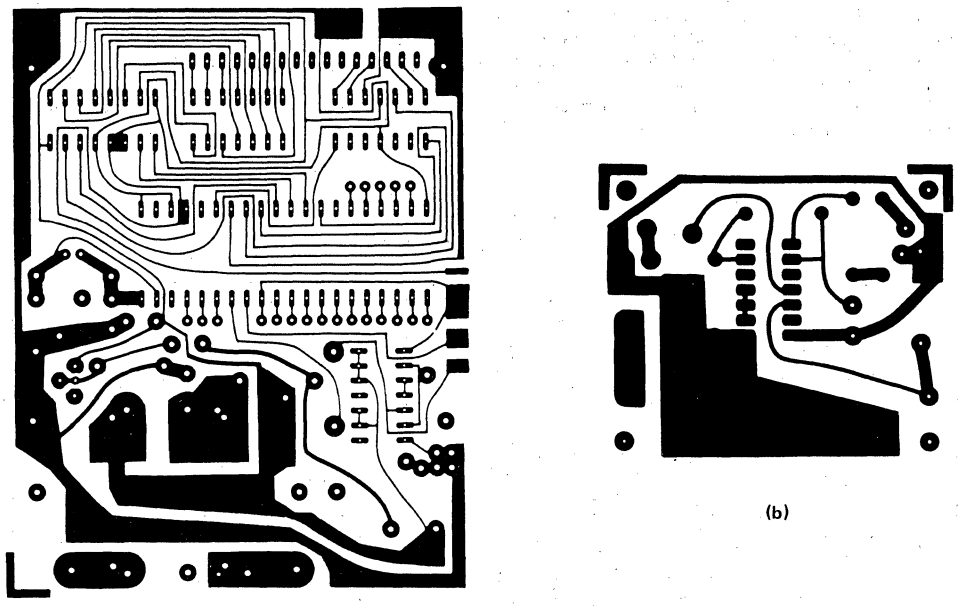



Figure 12. X1 Positives for the Printed Circuit Boards. a) Positive for the Stopwatch/Timer. b) Positive for the Appliance Control circuit.

The center portion of the top half of the case has been designed for a switch assembly composed of three push-button switches and one slide switch. The assembly is made of calculator-type switches and a flex-circuit; however, James Electronics provides neither the switches nor the flex-circuit.

*Figure 13a* shows the layout of the flex-circuit; *Figure 13b* is a view of the flex-circuit after it has been folded over the thin plastic insulator which is shown in *Figure 13c*. The insulator must be oriented so that the circular cutouts are between the two sets of four copper hexagons. The copper trace through each hexagon forms one contact of a SPST switch.

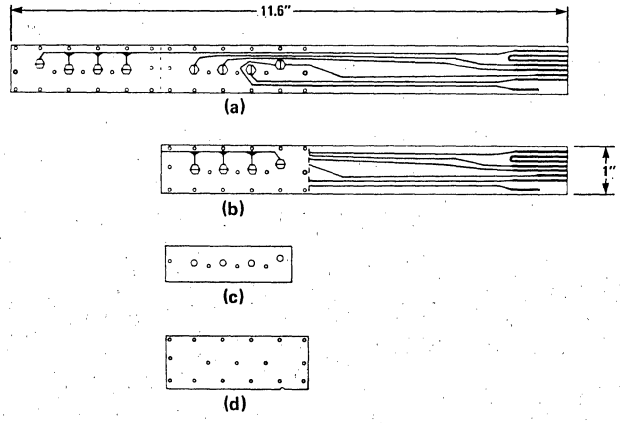



**Figure 13. Flex-Circuit Assembly. a) Layout of the flex-circuit. b) Layout of the flex-circuit after it has been folded to form the contacts of three SPST momentary pushbutton switches and one SPST slide switch. c) Thin plastic insulator which must be inserted between the folded portions of the flex-circuit. d) Plastic cover which fits over the flex-circuit assembly to hold it in place in the top of the case.**

If the automatic reset feature for function 6 is to be included, cut the slide switch hexagon connection to $V_{SS}$ as shown in *Figure 14* and cut a little square piece from the thin insulator. This small square should be just large enough to allow a solder connection to be made between the trace going to the slide switch hexagon and the trace going to the FE switch hexagon. To solder the traces together, pretin both traces slightly, fold the flex-circuit as shown in *Figure 13b*, and apply a small soldering iron tip to the trace going to the slide switch hexagon at a point above the insulator cutout.

The switches should then be placed in the top of the box in the spaces provided. The flex-circuit is then placed over the switches. Finally, the plastic cover fits over the entire assembly as shown in *Figure 15*. Holding the plastic cover firmly in place, touch a clean soldering iron tip to each of the plastic pegs protruding through the holes in the plastic switch assembly cover until the assembly cover is sealed to the top of the case. Cut the single tall plastic peg to the rear of the switch assembly cutout if there is one.



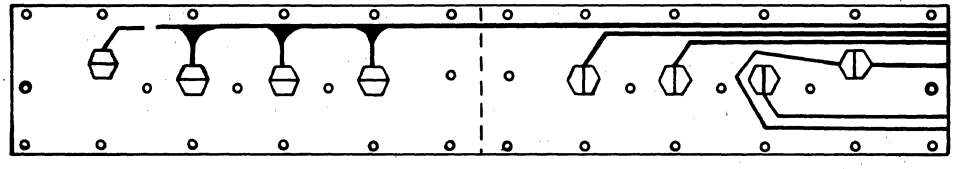


**Figure 14. Full-Size Partial Drawing of the Flex-Circuit Layout Showing the Trace which Must be Cut if the Auto Reset Option is Desired.**

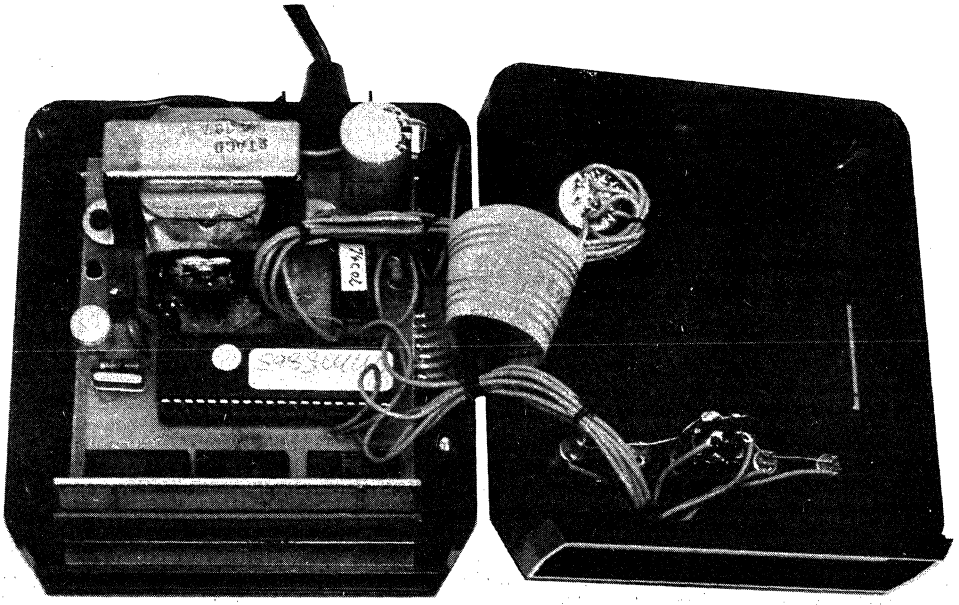


**Figure 15. Photograph Showing the Internal Construction of the Stopwatch/Timer. Note how flex-circuit runs from the top of the case to the trace-side of the printed circuit board.**

Drill the holes for the rotary function switch, the comparator switch, and the four programming switches as shown in the drilling guide of *Figure 16*. The drilling guide must be modified as shown in *Figure 17* if the Centralab PS-101 switch is used. The holes for the rotary switch must be marked and drilled precisely. In addition, if the Centralab PS-101 switch is used the filter capacitor, C1, must lie on its side to make room for the function switch. Mounting the top of the case to the bottom is easier if the Centralab PS-101 switch is used. If desired, a jack may be mounted in the bottom half of the case in the right hand rear corner, behind C1, to provide a quick connection to an enlarger or appliance control circuit. The fit will be tight, but a miniature jack can be mounted without much difficulty. This completes the case preparations.

Before parts are mounted to the PC board, the fit of the board to the case should be checked. It may be necessary to adjust the mounting holes slightly with a small round file. Try not to completely break the traces surrounding the mounting holes. There are six mounting holes in the PC board. These holes match six plastic pegs in the bottom of the case. Two of the pegs are to be inserted through the transformer mounting flanges if a transformer of the correct size is used. If the Radio Shack, or some other transformer which does not fit precisely, is used, it may not be possible to fit the pegs through the transformer mounting flanges.

After the IC sockets are mounted, the transformer and C1 should be mounted. If the Centralab PS-101 switch is used, the filter capacitor should be attached to the board

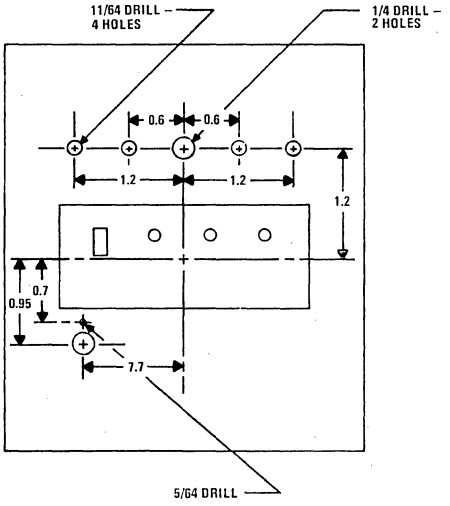



**Figure 16. Drilling Guide for the Case Top if the MRC-1-10 Rotary Function Switch is Used. (Dimensions in inches.)**

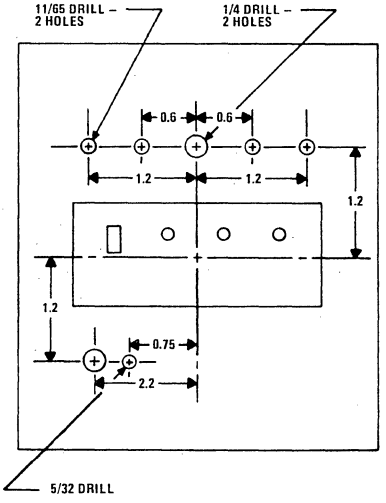



**Figure 17. Drilling Guide for the Case Top if the PS-101 Rotary Function Switch is Used. (Dimensions in inches.)**



with leads that are long enough to permit the capacitor to lie on its side. The diameter of C1 must not be greater than 0.7 inch and its length must not be greater than 1.2 inch.

The display mounting pins should be soldered to the display before the display is mounted to the board. Be careful not to lift the display pin pads when soldering.

Wires must be soldered to the board and connected to the switches mounted to the top of the case. Refer to the wiring diagram shown in *Figure 18*.

Wire jumpers may be used to program the display resolution and the modulo of the counters using the charts shown in *Tables I* and *II*. The connections shown in *Figure 2* cause the display to read in tens of minutes, minutes, tens of seconds, and seconds; maximum time is 99 min 59 sec. A pad which allows a connection to an external clock is available at pin 15 of the MM5865.

After all components have been mounted and all wire connections have been made, proceed to the preliminary checkout and adjustments section before applying power to the board.

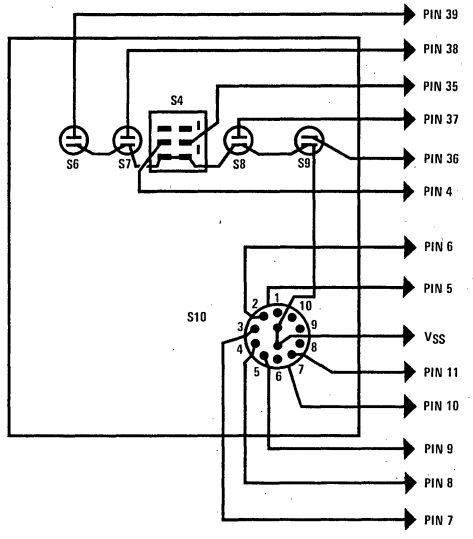



**Figure 18. Wiring Diagram for the Switches Mounted in the Case Top.**

## Preliminary Checkout and Adjustments

The following tests and adjustments should be carefully completed before power is applied to the stopwatch/timer or the appliance control circuit.

Rotate the F switch to function 7, place the ARE switch in the OFF position, place the C switch in the CCE position, and disconnect the tape recorder plug from the jack at the rear of the stopwatch/timer case. Adjust R1 for minimum resistance. Do not connect any appliance to the appliance control circuit, but do place a fuse in the fuse holder.

Measure the following points for the indicated amount of resistance:

1. Across the stopwatch/timer line cord plug > 50 ohms
2. Across C1, with VOM on X1K scale and common probe to $V_{DD}$, > 5k ohms, after C1 charges
3. Across R1 < 15 ohms
4. Across C2 > 100 ohms
5. Across the appliance control circuit line plug > 10k ohms

If these values of resistance cannot be found at the points indicated, check the PC boards for opens or shorts as necessary. Then, with a VOM connected across C2, apply power to the stopwatch/timer; the VOM should read slightly more than 1 volt. Increase the resistance of R1 until the VOM reads 8.4 volts. Slightly under 8.4 volts is better than slightly over. Pressing the reset switch should cause "0000" to appear on the display, unless the display already reads "0000."

If the display is blank or indicates only one or two zeroes, the oscillator is probably not oscillating. Rotate C3, 360 degrees if necessary, while observing the display. If the display still fails to respond properly, check the voltage at pin 20 of the MM5865; it is very close to 6 volts when the oscillator is functioning.

After oscillation has been confirmed the display should be examined for segment and digit defects. If any segment or digit does not appear in the display (The g segment does not appear when the display reads all zeroes.), the board and the display mounting pin connections must be checked.

When handling the stopwatch/timer before it is mounted in its case, extreme care must be used to not break the connections between the flex-circuit and the printed circuit board. However, these connections need not be made until the oscillator and display have been checked out.

After the oscillator and display checkout, the frequency of the oscillator should be adjusted to the crystal frequency using C3. Then the board may be placed in the bottom of the case. The balance of the preliminary checkout consists of stepping through the operational flow diagrams in *Figures 3-8*; a VOM should be connected to the output jack during the functions 6 and 7 checkout. If any of the switches under the flip cover fail to respond, check to see if the flex-circuit is broken at the point where it connects to the board.

## Final Assembly and Checkout

The board may be fastened to the bottom of the case by forcing #6 tinnerman nuts over the plastic pegs which appear through the holes indicated in *Figure 11*. This may be done easily with a 5/16-inch nutdriver. Then force the line cord in the cutout provided.

The top of the case may then be carefully fitted to the bottom, with the red plastic filter partially in place.

A slot in each half of the case retains the filter when the case halves are fastened. If the MRC-1-10 switch is used, the fit will be tight because of its proximity to C1. The cutout for the line cord in the top half of the case must be forced over the line cord.

Once the two halves are fitted properly, fasten them together using the four screws provided with the case. Install the rubber feet and proceed with the final checkout.

The final checkout is a repetition of the operational checks using the flow diagrams. Each option at each decision point in every flow diagram should be exercised.

## Resolution and Accuracy

If a crystal is used for the time base of the stopwatch/timer, the accuracy of the displayed count will, of course, depend upon the particular crystal used. In addition, because the MM5865 begins to count on the leading edge of the start/stop pulse, the width of this pulse becomes important when the event time is very short.

For example, when coupling the timer to an appliance, if the width of the start/stop pulse is longer than the event time, the appliance will not turn off at the end of the programmed time.

This is why C5 and R4 have been included. Together they insure that the start/stop pulse will not be longer than 0.01 second. This pulse width should be adequate for most users. C5 and R4 may be omitted if the length

of time the start/stop switch is to be held closed will always be less than any timed event. When C5 and R4 are omitted, the SS switch simply connects to $V_{SS}$.

As to crystal accuracy, the stopwatch/timer will lose 0.001 sec/sec if a 32.768kHz crystal is used instead of a 32.8kHz crystal. This should be insignificant for most users.

Also, the display resolutions which may be programmed by on board jumper wires will be adequate for most users. *Figure 2* illustrates the connections to the MM5865 which will cause the display to read in tens of minutes, minutes, tens of seconds, and seconds.

When it becomes desirable to achieve a display resolution which allows the timing of events that are hours in length, it is necessary to provide the MM5865 with an external time base. This may be done by cascading two MM5865s or by using a simple timing circuit built around an LM555 timer or a digital clock. *Figure 19* shows how an MM5315 digital clock may be used as a time base for the MM5865. The MM5315 itself uses the line frequency as a time base. The MM5315 is shown as it would be connected for a 60 Hz line frequency.

When an external time base is provided for the MM5865 in this manner, an external multiplexer must also be provided. The oscillator formed with the 74C14 supplies the desired multiplex frequency as shown in *Figure 19*.

A final note: Unless the start pulse is externally synchronized to the clock (available at pin 15 of the MM5865, if the internal oscillator is used), the amount of time which will elapse between the arrival of the start pulse at pin 14 of the MM5865 and the appearance of the first digit in the display will not be equal to the programmed display resolution. It is possible to develop a start pulse that is synchronized to the clock using an MM74C221 Dual Monostable Multivibrator as shown in *Figure 20*. The time constant of R1 - C1 should be equal to the display resolution, the time constant of R2 - C2 should be less than the programmed display resolution, and the time constant of R3 - C3 should be less than the time constant of R2 - C2.

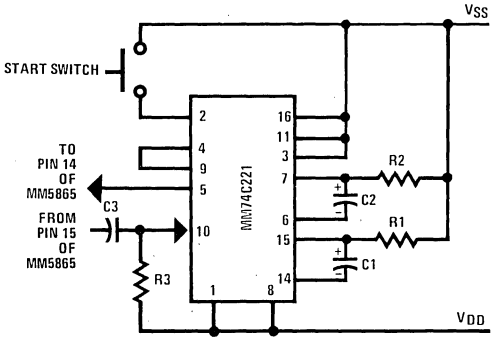



**Figure 20. Start-Pulse Synchronizer. Time constant of R1 - C1 = display resolution. Time constant of R2 - C2 < display resolution. Time constant of R3 - C3 < time constant of R2 - C2.**

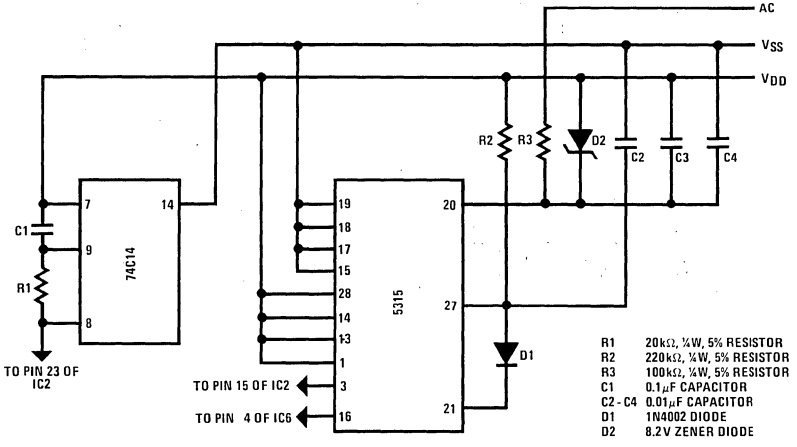
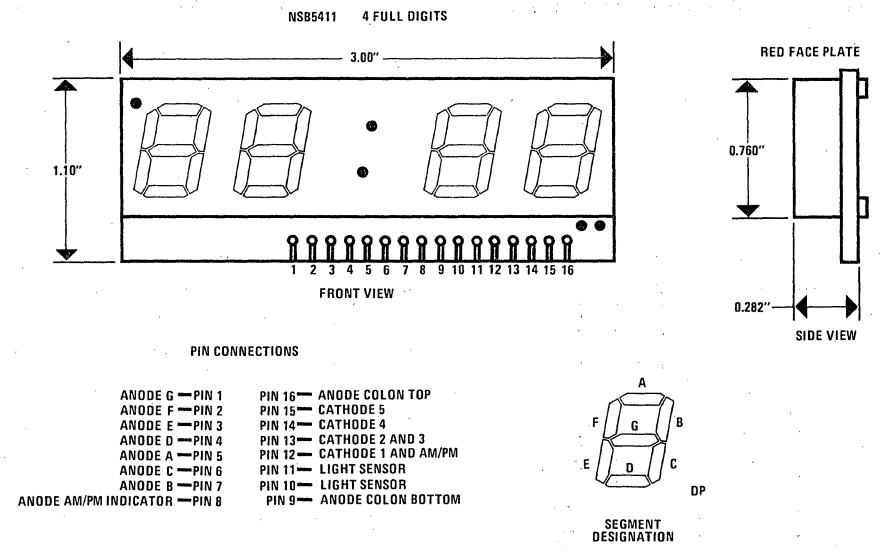



**Figure 19. Using an MM5315 Digital Clock and an External Multiplexer to Provide an External Time Base for the MM5865 to Generate a Display Resolution of 1 Minute.**

## PARTS LIST

| | |
|---|---|
| R1 | 5kΩ trimpot |
| R2 | 240Ω, ¼W, 5% resistor |
| R3 | 20MΩ, ¼W, 5% resistor |
| R4 | 1MΩ, ¼W, 5% resistor |
| R5 | 100kΩ, ¼W, 5% resistor |
| R6 | 5.1kΩ, ¼W, 5% resistor |
| R7 | 4.7kΩ, ¼W, 5% resistor |
| R8 | 10kΩ, 1W, 5% resistor |
| C1 | 470 - 1000mF, 25V capacitor |
| C2 | 10mF, 25$WV_{DC}$ solid tantalum capacitor |
| C3 | 6 - 25pF variable capacitor. Sprague QT1-18 4 - 30pF may be used. |
| C4 | 25 - 27pF, disc ceramic capacitor |
| C5 | 0.01mF disc ceramic capacitor |
| C6 | 100mF, 25$WV_{DC}$ capacitor |
| C7 | 0.05mF, 200$WV_{DC}$ capacitor |
| $D_1$, $D_2$ | IN4003 |
| T1 | 10 - 16.5$V_{AC}$ @ 300mA transformer |
| IC1 | LM317T voltage regulator |
| IC2 | MM5865 universal timer |
| IC3 | CD14511 decoder/driver/latch |
| IC4 | DS8877 or DS75492 digit driver |
| IC5 | RA07 - 150 resistor array |
| IC6 | 74C02 quad 2-input NOR gate |
| IC7 | CA3059 zero voltage switch |
| Triac | HEP R1723 |
| F1 | 1A fast or normal blow fuse |
| XTAL | 32.8kHz crystal (32.768kHz can be substituted. Timer will lose about 35 sec in 11 hr 20 min of use.) |
| S1, S3, S5 | SPST, NO, momentary pushbutton switches; part of flex-circuit switch assembly. |
| S2 | SPST slide switch; part of flex-circuit switch assembly. |
| S4 | DPDT, center OFF toggle switch |
| S6 - S9 | SPST, NO, momentary pushbutton switches |
| S10 | 7 - 12 position rotary switch — Centralab PS-101 or Alcoswitch MRC-1-10. |
| S11 | SPST toggle switch |
| S12 | SPDT toggle switch (optional) |
| Display | National Semiconductor NSB5411 4-digit multiplexed display. |
| Heat Sink | TO-220 heat sink. Two needed. |
| Misc. | 16 display mounting pins (strip of 16 pins); 1 case; Clock/Instrument (available from James Electronics); 1 flex-circuit; 1 flex-circuit insulator; 2 Tinnerman nuts, #6; fuseholder; appliance control box, #LMB C.R.-234; 115$V_{AC}$ chassis mounting socket; miniature jacks; phone cable (shielded); IC sockets. |



# Programmable TV Timer/ Time-Channel Display







This application note describes an on screen TV real time and channel display based on the MM53107, MM53100 and MM5840 MOS/LSI integrated circuits.

The sample was assembled in a small (15 x 6 x 8 cm) box, connected to the TV chassis with 10 wires. A Grundig portable black and white Triumph set was used.

The highlight features of the clock are:

- 8, or optionally 5-digit time display
- Channel number display and remote channel select
- Programmable TV ON timer
- Choice of ON durations, and safety turn-OFF
- Remote manual TV ON and OFF
- Display option with or without frame
- Low power consumption
- Battery back up for power failure or OFF TV mode

## BASIC DESCRIPTION

The time base for the on screen clock is a pulse generator, consisting of a 2.097152 MHz crystal and an MM53107 divide-by-34952 oscillator/divider which delivers a buffered 60 Hz pulse. A CMOS inverter/amplifier is provided between the oscillator in (pin 5) and oscillator out (pin 6) terminals. The 20 MΩ resistor R1 is required, to bias the inverter for Class A amplifier operation. Capacitors C1 and $C_{T1}$ provide the parallel load capacitance needed for precise tuning of the quartz crystal *(Figure 1)*.

The 60 Hz pulse frequency enters the MM53100 at pin 20. An internal prescaler divides by 60 to obtain seconds of time of day. (The MM53100 will also accept 50 Hz input at pin 20 by utilizing pin 21 50/60 Hz select pin.) In the present application this pin 21 input must be a logical "1" in order to divide the 60 Hz output from the MM53107 to 1 Hz. An internal pull-down resistor is common to pin 21 and when left unconnected the prescale divider pin 20 is programmed to divide 50 Hz to 1 Hz.

Should one decide to utilize 50 or 60 Hz AC input directly from the AC power line in place of the MM53107 oscillator/divider, a simple RC filter should be used at pin 20 to remove possible line voltage transients that could cause the clock circuit MM53100 to gain time or otherwise be damaged. The input on pin 20 should swing between $V_{DD}$ and $V_{SS}$.

## MM53100, MM5840 INTERCONNECTION
(Also see *Figure 1*)

On the MM53100, pins 2, 3, 4 and 5 are the clock real time outputs. The format is an inverted multiplexed BCD-code and goes directly to the display IC MM5840.

As the MM53100 and MM5840 are situated on different $V_{SS}$ levels, 4 pull-down resistors R2–R5 are required to provide the display IC with the necessary full swing from 0V to 12V at its BCD inputs. The digital select inputs at the MM53100, $D_X$ (pin 24), $D_Y$ (pin 23), and $D_Z$ (pin 22) are directly driven by the display IC. The code for each displayed digit is shown in the table below:

| DIGIT SELECT LINES | DIGIT DISPLAYED | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | S1 | S10 | x | M1 | M10 | x | H1 | H10 |
| $D_X$ | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| $D_Y$ | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| $D_Z$ | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

## TIME SETTING

To set real time as well as TV ON time, 3 inputs are assigned: pin 17 for SET MINUTES, pin 18 for SET HOURS, and pin 19 for HOLD. Internal pull-down resistors provide logical zeros on all pins, which in turn force normal timekeeping function of the clock. Switching any one of these inputs (one at a time) to logical "1" by means of the pushbuttons B1, B2 or B3 results in the desired time-setting function. SET HOURS advances hours information at 1 hour per second and SET MINUTES advances minutes information at 1 minute per second, without carryover into the hours counter. SET MINUTES also resets the seconds counter to all 00's. Logical "1" at the HOLD input stops the clock (pulses) to the minutes counter and resets the seconds counter. Activating SET MINUTES and SET HOURS simultaneously resets the displayed counters to all 0's (Midnight).

In addition to its function as a clock, the MM53100 functions as an alarm clock. It is possible to preset a time at which, when it coincides with real time, the device will activate certain outputs. This is done by switching the display control input, pin 16 with S4. When taken to logical "1", real time is disabled from the display. It is now possible to set an alarm time (displayed on the screen) at which the TV will be automatically switched ON. This TV ON time setting is controlled by SET HOURS and SET MINUTES switches. The display control input has internal pull-down resistors to $V_{SS}$. When taken to logical "0" or in open circuit condition, real time is displayed and the SET HOURS and SET MINUTES inputs operate the real time counters.

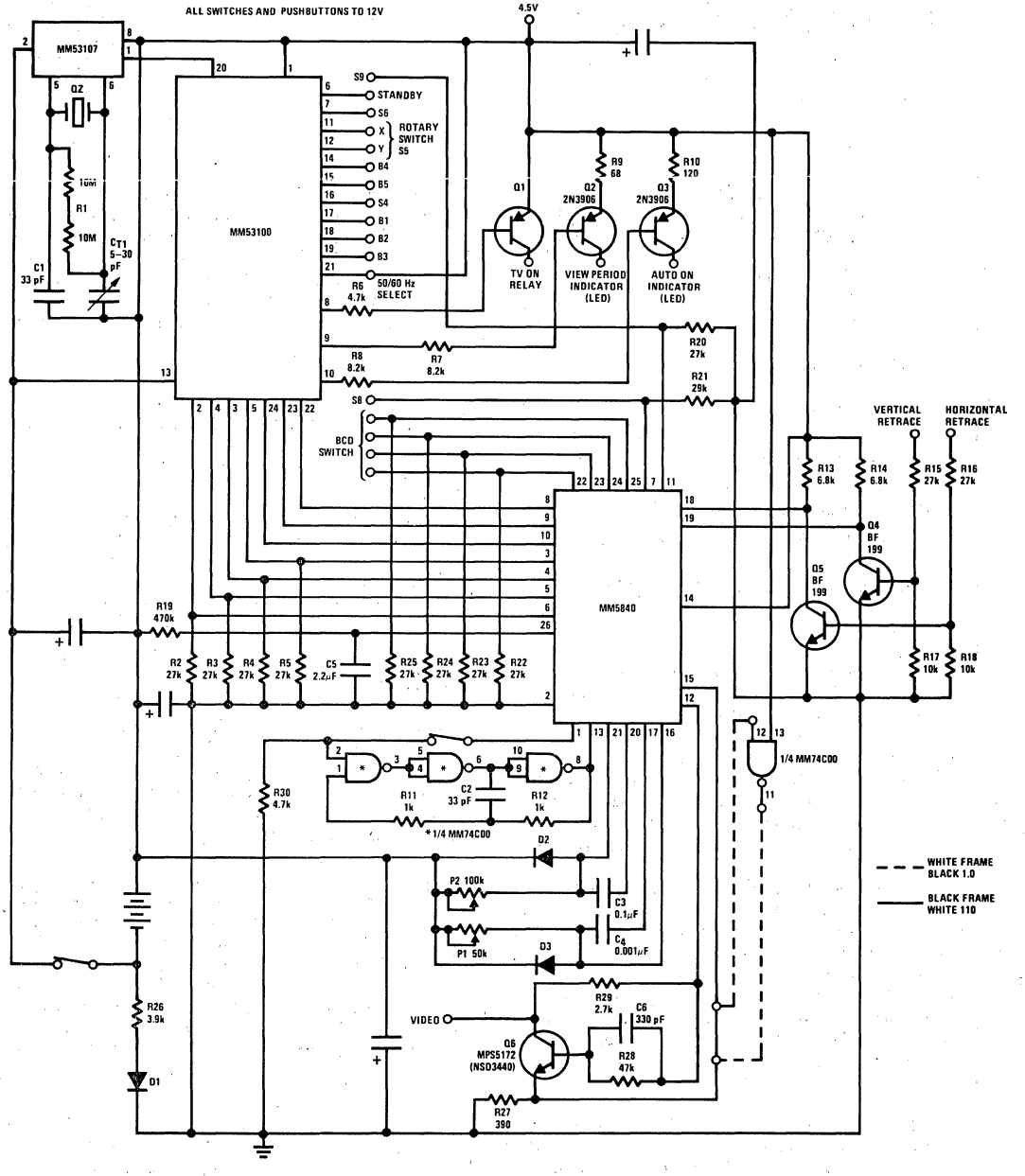



FIGURE 1. Clock Schematic

## DURATION OF TV ON TIME

It is possible to determine the TV ON duration with connections to the period select inputs X (pin 11) and Y (pin 12). The inputs are either permanently wired or connected to a 2-pole, 4-way switch (S5 on *Figure 1*) according to the following table:

| Period Select Inputs | | View Period Programmed |
|---|---|---|
| X | Y | |
| 0 | 0 | 5 Mins |
| 0 | 1 | 10 Mins |
| 1 | 0 | 20 Mins |
| 1 | 1 | 30 Mins |

Internal pull-down resistors at the Period Select inputs make it unnecessary to wire both $V_{SS}$ and $V_{DD}$ on the switch S5.

## ALARM ENABLE

Should the user decide against viewing the TV program originally desired and had consequently programmed to start via the alarm program enable, it is possible to disable the alarm enable by placing a logical "1" on pin 7 of the MM53100 (S6 of *Figure 1*). Pin 7 has an internal pull-down resistor to $V_{SS}$ keeping the input to logical "0" thus enabling the TV alarm enable or turn-ON time.

## TV ON OUTPUT

Pin 8 of the MM53100 provides the ON/OFF signal to the TV chassis. If pin 8 is in a logical "1" state the TV is in a standby-OFF mode. There are 2 possible ways this output can be taken to logical "0" to switch the TV to ON.

If the alarm enable is ON (logical "0") and real time coincides with the programmed ON time

Or if pin 14 (manual ON input) is taken to logical "0" (B4 on *Figure 1*).

Either of the above conditions will switch transistor Q1 and its corresponding relay. The TV ON output will be at logical "1" and the relay will open if either the programmed view period is over or the manual OFF input (pin 15) is brought to logical "1" (pushbutton B5 of *Figure 1*). The manual ON input as well as the manual OFF input are provided with internal pull-down resistors to $V_{SS}$.

## INDICATORS

The MM53100 has 2 indicator outputs, pins 9 and 10, and are "Auto" ON and "View Period" indicators respectively. "Auto" ON indicates that the TV is ON in an automatic mode and is receptive to the alarm enable and disable programming modes.

The "View Period" indicator provides a 1 Hz output during the ON time suitable for driving an ON or OFF screen indicator. Both pins are normally a logical "1" until the TV is switched ON in the automatic mode. Pin 10 stays on logical "0" for the duration of the automatically affected ON time while pin 9 alternates between a logical "1" and logical "0" at a 1 Hz rate.

In this application 2 LEDs driven by transistors Q2 and Q3 indicate that the TV is in the automatic mode and that the set would be turned OFF after 5, 10, 20 or 30 minutes if there is no manual ON signal. (B4 on *Figure 1*). If this were the case, pins 9 and 10 will go back to logical "1". Resistors R7 and R8, as well as R6 for the relay circuit, limit the base currents while R9 and R10 provide for equal light intensity of the LEDs. Pin 9 with its switching transistor may also be used to make the TV display flash at a 1 Hz rate as an indicator that the set is in an "On Time Viewing Period". This is accomplished by means of Q2 and a quad NAND gate MM74C00. A 4 MHz oscillator is switched ON or OFF at pin 1 *(Figure 2)*.

## STANDBY

The standby control input, pin 6, has an internal resistor to $V_{SS}$. Its function is to sense when the line generated 12V supply within the TV is turned OFF by the power-switch S0 and in turn disables the LED and relay-output (pins 8, 9, and 10). This input prevents the batteries from being drained by the base-current of transistors 1 through 3, in case the manual ON switch is accidently pushed while the TV set is disconnected from the power AC line.

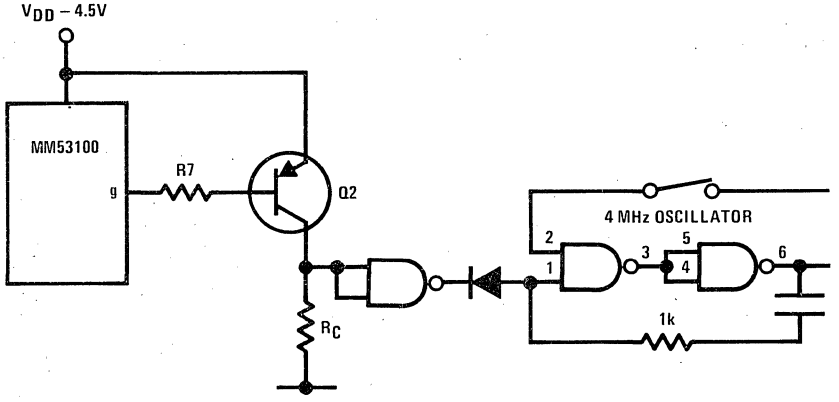



FIGURE 2. TV-Display "Flasher"

### MM5840

MM5840 is the channel number and time display circuit. It operates with a 4 MHz clock input generated by a 3-stage CMOS inverter oscillator MM74C00. Variations of frequency are achieved by selecting C2, R11 and R12 according to the following expression:

$$f \cong \frac{1}{2\ R12\ C2\ \frac{0,4\ R11}{R11 + R12} + 0,7}$$

This oscillator serves as a time base for the display duration of 1 line scan. Sixty 4 MHz pulses are internally counted by the horizontal counter. The display is effected only during this counting time, and therefore, an increase of this frequency will result in a narrower display format (and vice versa). Because the duration of 1 line sweep is about 60 $\mu$s [0.25 $\mu$s (4 MHz) x 60 (count till 60) = 15 $\mu$s] the width of the display is 15 $\mu$s ÷ 60 $\mu$s, about one fourth of the screen width. In contrast to the horizontal counter which uses an external 4 MHz oscillator, the vertical counter makes use of the TV generated line pulses to enable display during 1 picture period. Both counters are enabled by 2 corresponding monostable flip-flops. The pulse width and the display position on the screen can be adjusted by an external RC network. Potentiometer P1 defines the horizontal position and potentiometer P2 defines the vertical. Pin 1 of MM5840 is the oscillator inhibit output. It is normally low, but will go high for the display duration within 1 line, (i.e. the above mentioned 15 $\mu$seconds and 14 times per 1 picture sweep). The height of the display format is 14 lines, defined by the vertical counter. With the logical "1" trigger signal from pin 1 of the MM5840 at the first NAND gate, the oscillator is triggered and enabled during the real readout time. This trigger is important, because it guarantees a stable display. In this way the display start per line and picture sweep is always exactly the same. Also, the oscillation period and thereby power consumption of the CMOS oscillator is reduced to a minimum.

Pins 18 and 19 of the display chip get the horizontal and vertical trigger signals from the TV via the switching amplifiers Q4 and Q5. Care should be taken that these input pulses have sharp edges for a clean and stable display as they represent the intrinsic time base for the display.

### DISPLAY MODE

By means of pins 7, 11 and 26 it is possible to control the display mode. If the mode control input, pin 11, is switched with S9 to logical "1", time and channel number are displayed, while a logical "0" will display in a channel number only.

A logical "1" at pin 7, (digit select input) results in an 8-digit time display, namely hours tens, hours units, colon, minutes tens, minutes units, colon, seconds tens, and seconds units. The last 2 seconds digits and the colon are switched OFF by logical "0".

Pin 26 is a monostable output, operated by selecting the channel number. Leaving it unconnected means permanent channel number display. With a logical "1" the channel number is disabled. With the help of an external RC network, (the resistor R19 going to 12V), it is possible to display the channel number for a duration defined by the time constant. Resistors R10 and R21 are the necessary pull-down resistors which provide the logical "0" information when the respective switches are opened.

Pins 22, 23, 24 and 25 are the channel number inputs. The MM5840 accepts a binary plus 1 coded input signal. It is more convenient to employ a directly coded BCD switch instead of wiring a rotary switch. Pull-down resistors R22—R25 prevent the inputs from floating due to the high input impedance.

### BCD TO 1 OF 10 DECODER

Parallel to the channel number inputs of the MM5840 are the inputs of the MM74C42, which decodes the BCD signal to 1 of 10. *(Figure 3.)* The sensor keys on the TV used, require ground potential for switching the channels. Because of its normally high output the MM74C42 is ideal for this application.

### VIDEO DRIVE TO CRT

Pin 15 of the MM5840 provides the video signal. By using a high voltage ($V_{CE}$) transistor like the NSD3440 it is possible to drive the cathode of the picture tube directly in either black and white or color. However, since the cathode is typically supplied with an internal driver, it is more simple to feed the clock video signal at the base of this driver, taking advantage of the TV internal filter devices between collector and cathode. In this case any fast switching low Beta transistor can fill the position of NSD3440. (The details are shown in *Figure 4.*)

### DISPLAY FRAME

Pin 12 provides square pulses synchronous to the video pulse sequence per line in order to generate a frame around the display. By interconnection of different pins on the PC board *(Figure 5)* it is possible to effect 4 main options: black numbers, white numbers, dark frame with bright numbers, or bright frame and dark numbers, by using the fourth gate of the quad NAND gate MM74C00 (as indicated by the dotted line in *Figures 1 and 4*), Q6 fulfills the function of a differential stage as the 2 positive signals are fed in at the base and emitter. A capacitor C6 speeds up Q6. In this mode of operation, the display is never brighter than the brightest screen picture, because the collector of Q6 can not go higher than the output level of the picture IF device. However, by means of R29, the base of the cathode driver transistor can be adjusted for the display time per sweep, for a good balance between clock display and screen picture.

## POWER SUPPLY

The TV's transformer, rectifier and load capacitor are in action only in the standby position. All the other TV devices are interrupted by S1 of the relay. The DC power at the load capacitor, which varies from 15V when the TV is ON to 22V in standby, is regulated by means of an LM340-12 to a constant 12V for the clock supply. As the TV is OFF (in standby) it is only necessary to run the clock IC and 2 MHz oscillator/divider which are the circuits responsible for real timekeeping. Channel display and channel select, 4 MHz oscillator, display IC MM5840, channel number decoder MM74C42, BCD channel switch and the switching transistors (with the exception of Q1) can all be turned OFF. The $V_{SS}$ of all these devices is interrupted from the TV's ground by means of S2, operated in parallel with S1 by the relay. As it can be seen, only 3 main circuits are closed in standby: the voltage regulator circuit, the circuit of Q1 which switches the coil of the relay, and the charging circuit for the batteries *(Figure 6)*.

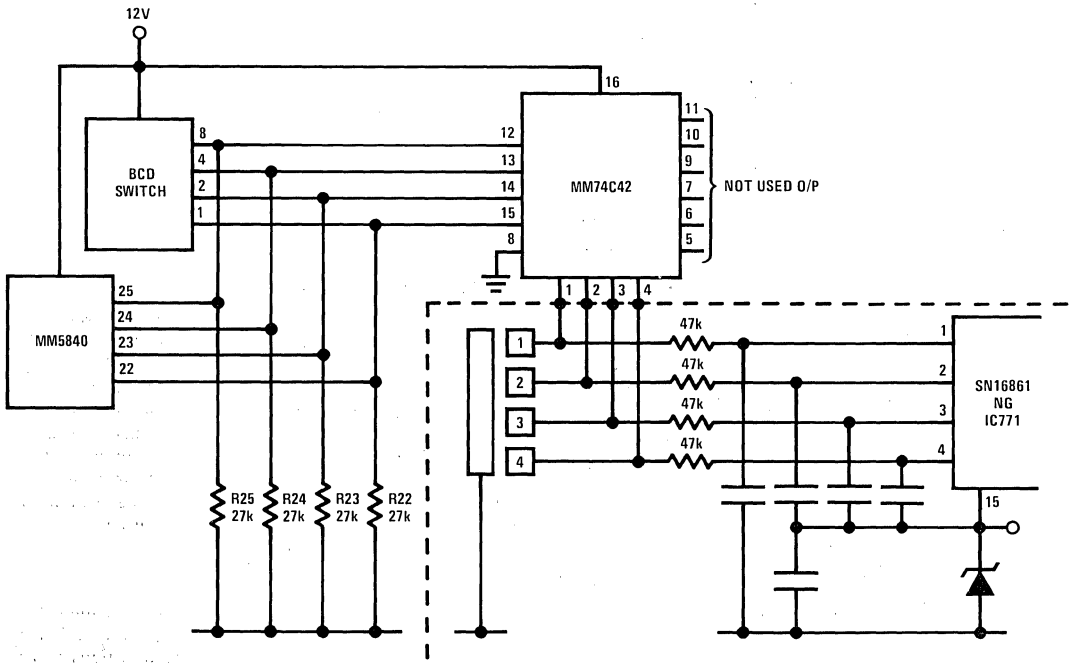



**FIGURE 3. Channel Switching Interface with MM74C42**

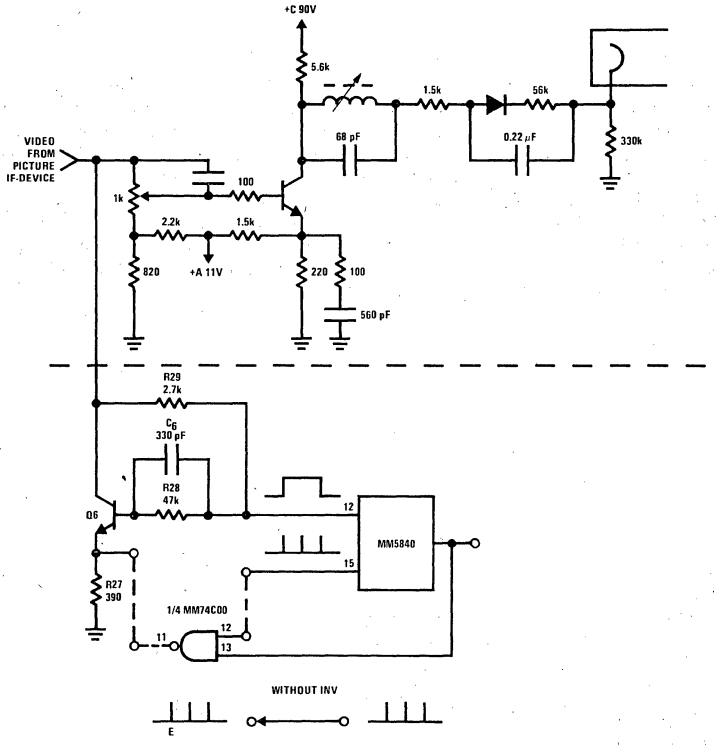
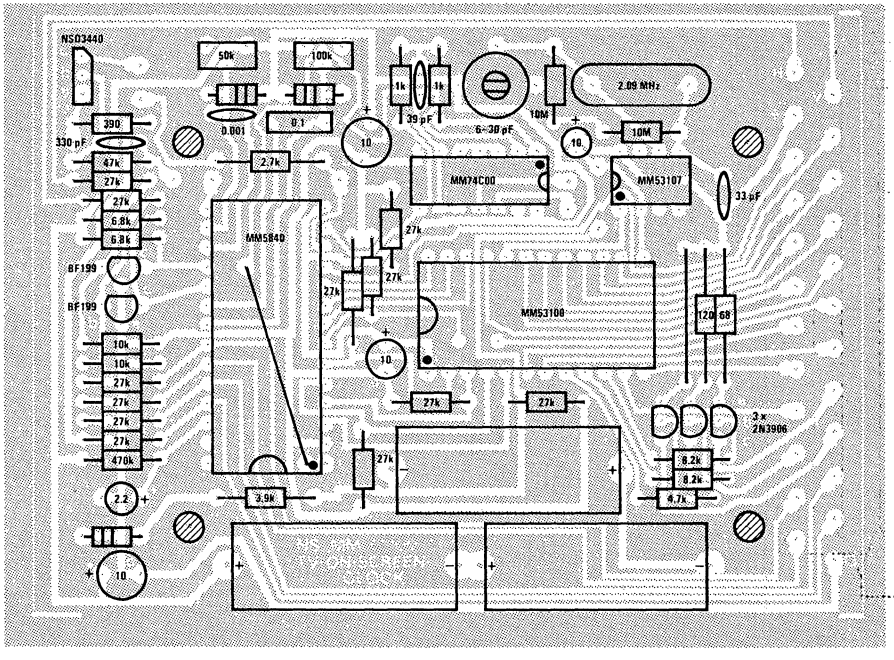



FIGURE 4. Video Drive to CRT and Display Frame Circuit



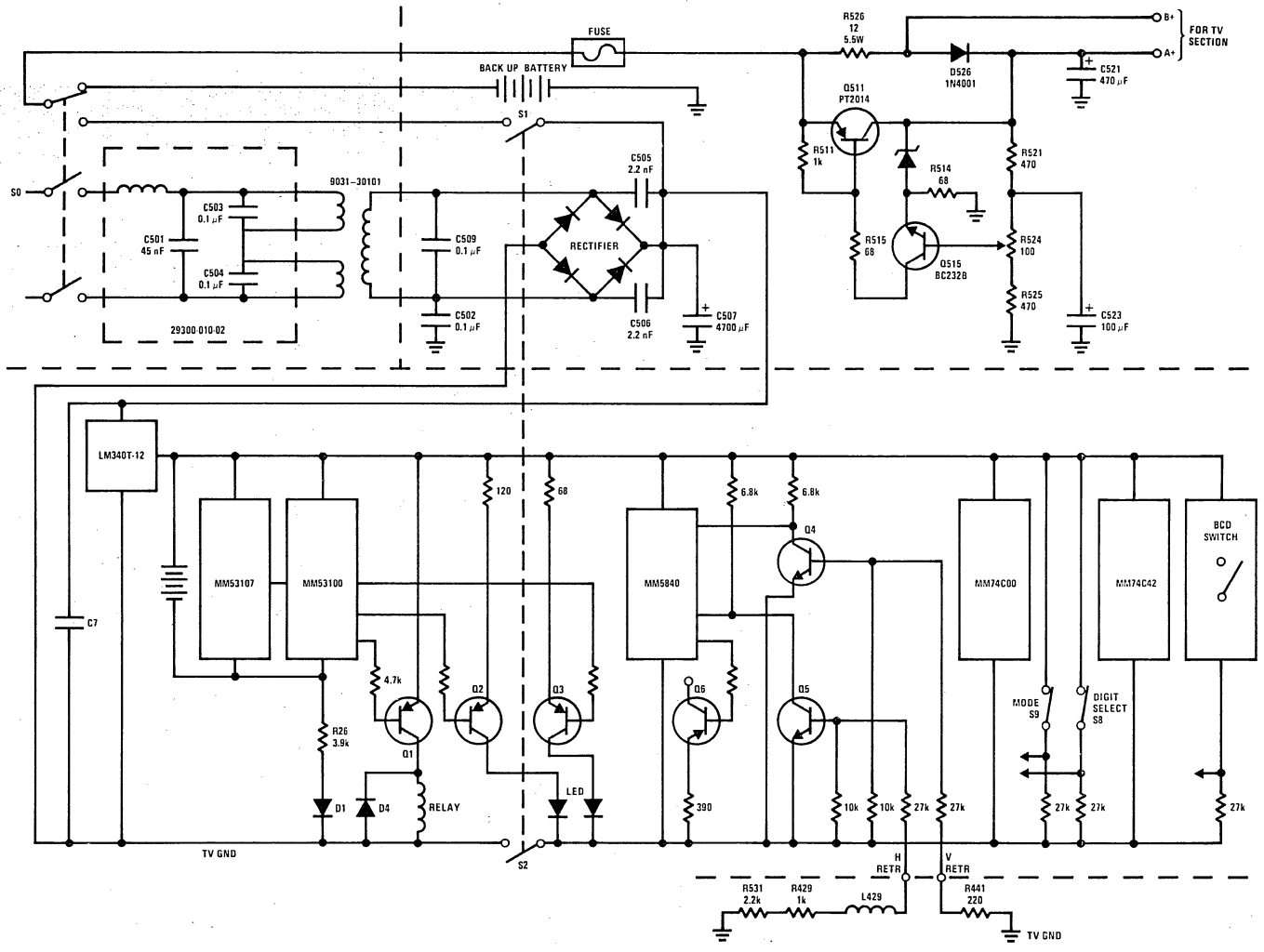



FIGURE 6. Power Supply Organization

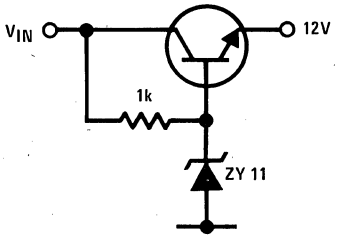



FIGURE 7. Voltage Regulator

The voltage regulator circuit provides a constant 12V with the set in standby or in ON position. A simpler method also provides the necessary 12V *(Figure 7)*.

A zener diode alone would not be suitable. The total supply current of the clock system will rise to about 80–100 mA when the TV switches ON in the automatic mode of operation because Q2 and Q3 will switch, powering the LEDs. To guarantee a clean switching of the relay without a drop down of the positive voltage, this circuit needs a quiescent current to run through the zener diode during standby. This current is only about 8 mA using the LM340T12.

With Q1 constantly connected to 12V, the necessary 3 mA is guaranteed for switching the relay. D24 eliminates voltage peaks at the moment of switching. In standby, the current through this circuit is 0.

This arrangement insures that the batteries are always fully charged to run the MM53107 and MM53100 totally and independently from the AC power line in case the TV is switched OFF by the power switch. The quiescent current of the MM53107 and MM53100 is about 170 $\mu$A. This current will rise to 0.6 mA from the battery if the manual ON switch is set, because the MM53100 provides a base current of about 0.4 mA to Q1.

D1 prevents the batteries from being drained by components going from 12V to ground. This is true for the BCD switch together with the 4 pull-down resistors, or the mode and digital select switch with its corresponding pull-down resistors. After switching the TV from OFF to standby position, the voltage between $V_{DD}$ and the switched OFF $V_{SS}$ will not be zero, but approximately 3V depending on the position of the BCD switch and how many $V_{DD}$ switches with pull-down resistors are closed *(Figure 1)*.

The small amount of current of about 2 mA, plus the 3V on the inputs of the ICs is of no real consequence as they are controlled by the voltage regulator and TV power supply, except that the 3V may cause the MM74C00 to oscillate and drain more battery current. Whether it oscillates in this state is a matter of chance depending on the logical state of the oscillator inhibit output when the $V_{SS}$ is interrupted by S2. If the 2 mA can be tolerated a 4.7k pull-down resistor R30 at the oscillator inhibit output will keep the MM74C00 from oscillating. Otherwise diodes can be added to all signal lines to the TV circuitry. The leaking circuit is generally interrupted without affecting the normal signal path.

The total quiescent current during standby consists of 10 mA, 8 mA voltage regulator current and 2 mA battery charging current. C7 across the input of the LM340T12 is important, to avoid oscillation caused by the long wire from stray capacitance within the TV to the voltage regulator.

## CLOCK ACCURACY

How precise should the 60 Hz pulse frequency (adjusted with C1) be in order to guarantee the usual accuracy in consumer clock devices of less than 1 minute per year? The measurable reference is the buffered 60 Hz divider output at pin 1. A direct measurement of the HF oscillator, say at pin 5 or 6, would influence the frequency by the inevitable load capacitance of the frequency counter employed for this measurement.

A separate high frequency buffered output was not made available in the MM53107 because of the increased battery drain that would result. A coupling loop over the crystal could be used to pick up the frequency. A deviation of +1 minute per year means a deviation of +3600 pulses at pin 1. More than 1,892,160,000 (60 x 60 x 60 x 24 x 365 = $1.89216 \times 10^9$ pulses per year), the number of pulses with a 60 Hz frequency. As a

frequency counter down to 60 Hz is not available everywhere, it is more reasonable to measure the period of 60 Hz, which is 16,6666 ms.

The period has to be adjusted to values between 16,66664 ms (60,00009 Hz), or 1.892162700 pulses per year, and 16,66669 ms (59,99991 Hz): 1,892,150,700 pulses per year. In the first case the clock will be 0.75 minute in advance of real time, in the second case it will be 0.83 minute late per year. With a capacity range of 5–36 pF for C1 it is not too difficult to achieve this result.

The pulse frequency is still dependent on 2 parameters: the supply voltage of the MM53107 and ambient temperature. The deviation of the clock in terms of minutes per year versus $V_{Batt}$ is shown in *Figure 8*.

It is to be seen that with a higher battery voltage, e.g. 4.5V, the deviation of the clock per V = 0.5V is almost half the deviation at lower voltage operation.

The influence of temperature falls within a similar range. The average value, when the whole clock device was heated from 22°C up to 50°C, showed a typical period shifting of +0.0001 ms. This represents about 3 minutes of deviation per year, assuming that the clock was adjusted at 22°C and that the TV reached a maximum temperature of 50°C. It is also assumed that the TV remains in constant operation.

This error can be greatly reduced however, if the calculation is based upon the fact that the TV is not in continuous use. With a typical ON-time of only 3 or 4 hours per day, the error due to temperature would only be about half a minute per year.

## THE MM53100 OFF-SCREEN CLOCK

The flexibility of the programmable TV timer really becomes clear in *Figure 9* where it is shown in a LED display application. The MM5840 display circuit in *Figure 1* is traded with a DS8664 decoder driver. Two additional standard low cost CMOS ICs convert the clock from ON screen to OFF screen LED, maintaining all timer features as previously described.

The DS8664 is in this approach the workhorse. The internal oscillator, tuned for high frequency (130 kHz) with low cost small capacitors, is fed to the MM74C90 counter. This counter terminates the count at Code 8, and is used to step the X, Y, Z input of the timer chip, also driving the decoder port of the DS8664. The digit drive capability of the IC can easily handle the peak current of 350 mA for a 0.8 inch display. The interdigit blanking is taken care of by adding the clock to pin 6 (this avoids any ghost effect).

The multiplexed BCD clock information is inverted and 7-segment encoded by MM74C04 and MM74C48.

All level shifting between the battery operated and the power supply 12V driven parts of the circuit is taken care of by resistors.

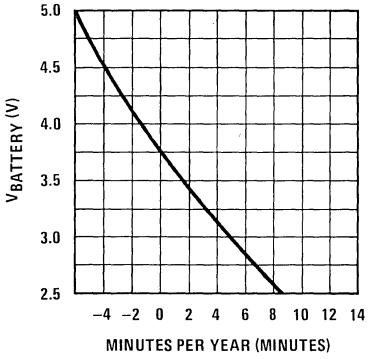



FIGURE 8. Drift vs $V_{BATTERY}$

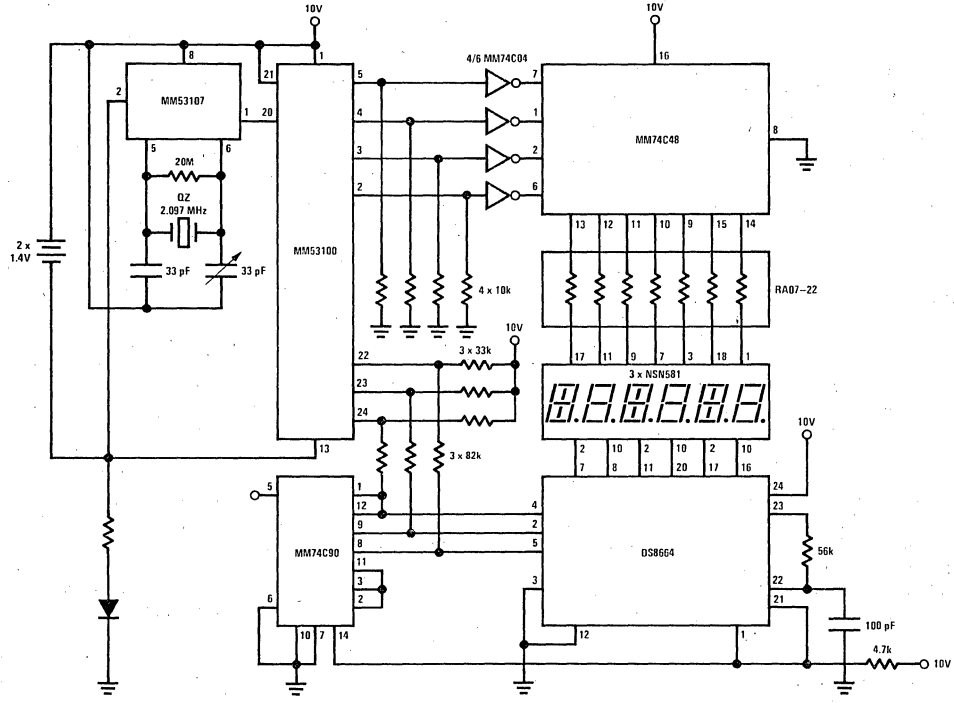



**FIGURE 9. Off Screen Clock Schematic**

# MM54240 Asynchronous Receiver/Transmitter Remote Controller Applications

National Semiconductor
Application Note 249
J. Hong
May 1980



## Introduction

The MM5420 Asynchronous Receiver/Transmitter Remote Controller is a low cost, easy-to-use circuit for serial data transmission applications. The circuit is fabricated in the N-channel metal gate process which gives it a wide supply voltage range ($V_{DD}$ = 4.75V-11.50V) and TTL compatibility.

A typical application would consist of an information handling center and up to 128 information gathering and information supplying stations. The information handling center would be composed of one MM54240 circuit interfaced to a microprocessor I/O system. The MM54240 in this instance is called the "master" circuit. An information gathering and supplying station would be one MM54240 interfaced to a A-to-D converter/D-to-A converter system or a digital peripheral system or any information source/destination. The MM54240 in this instance is the "slave" circuit.

The simplest way to interface such a system is by means of a twisted pair or a coaxial cable. A pull-up resistor is necessary on this communication line since the circuit drivers are open drain outputs. Care should be taken to reduce capacitance and resistance on this line. With the use of pulse width modulation techniques, frequency tolerance between the circuits is broadened. This feature is extremely desirable since the need for an expensive crystal controlled oscillator or ceramic resonator is eliminated. Furthermore, critical timing schemes with start and stop bits are not used. In addition, a debounce circuit is incorporated which contributes greatly to the noise immunity feature of the circuit.

## Circuit Description

A functional block diagram of the MM54240 is shown in Figure 2. The Control Logic section consists of the switching functions of the circuit. The PWM encoder/decoder encodes and decodes the pulse width modulation data format. The Shift Registers store and shift the data.

## Temperature Control and Security Application

The MM54240 can be used in many different types of low to medium speed processor controlled applications. For a system with 128 "slave" circuits, the time that it takes to interrogate all "slave" circuits can range from 1 to 2.5 seconds depending on the oscillator frequency of each individual circuit.

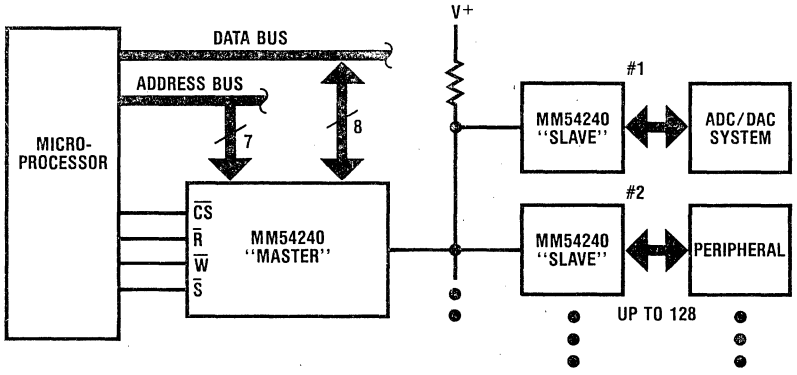



**Figure 1. Typical System Block Diagram**

The following example illustrates a possible in-the home use of such a system. A set of MM54240 circuits are used for controlling the temperature of the various rooms inside the house, the security of the windows and entrances, and also for turning the lights on and off when certain events take place. The processor controlling the system is a COP421L and it is directly interfaced to the MM54240 "master" circuit. Three address inputs of the "master" circuit are not used and they are tied directly to the power supply. A maximum of 16 "slave" circuits are then possible. They will start with address 112 and go up to 127. The "master" circuit has the Data I/O ports interfaced to the L I/O ports, the Address inputs interfaced to the D I/O ports, and the Control inputs interfaced to the G output ports of the COP421L. The Mode input is tied to $V_{DD}$ to select "master" operation and the only external components are the R-C's connected to each circuit. The power supply terminals are shared between the two circuits.

The heating system of the house consists of a furnace with a multi-speed air blower. A thermocouple thermometer installed in each room supplies the temperature information to the processor. The amount of air flowing into a room is controlled by a variable ventilation grating. When the temperature of the room falls, the ventilator opens further to let in increasing amounts of warm air. When all the rooms are sufficiently heated, the furnace is turned off. The temperature in different rooms may not be the same since the processor can control and adjust them to a programmed setting.

The first "slave" circuit has hard-wired address 112 and it is used for furnace control. The control ($C_1$, $C_2$) inputs are set up for (0,1) the low impedance output port selection. $D_1$ of the D outputs is used for furnace ignition. The other D outputs are used for controlling the air blower's variable speed. The second "slave" circuit has hard-wired address 113 and it is used for temperature sensing and ventilation opening control. The control ($C_1$, $C_2$) inputs are set up for (0,0) 4 in/4 out selection. $D_1$-$D_4$ are low impedance output ports for controlling the ventilator opening; $D_5$-$D_8$ are high impedance input ports for receiving temperature information from the thermocouple.

The third "slave" circuit has hard-wired address 114 and it is used for security purposes. The control ($C_1$, $C_2$) inputs are set up for (1,0) the weak pull-up option. The processor will have to initialize the output latches by loading logic ones into them. The $D_6$ I/O port is for Arming and Disarming the alarms. This is accomplished by a locking switch shorting the I/O port to $V_{SS}$ when armed. The $D_1$-$D_5$ I/O ports are connected to doors and windows. When they are shut, the I/O ports are shorted to $V_{SS}$. When a window or door is opened, the voltage level of that I/O port will increase. When the COP421L processor detects this change, it will enable the alarm at the $D_8$ I/O port and turn the light on at the $D_7$ I/O port. The processor can also be programmed to turn the light — $D_7$ I/O port — on when one or certain doors are open.

The second and third "slave" circuit configurations can be duplicated for other rooms. Each "slave" circuit has an optional clear switch in case the processor circuit fails and the "slave" circuit outputs have to be overridden.

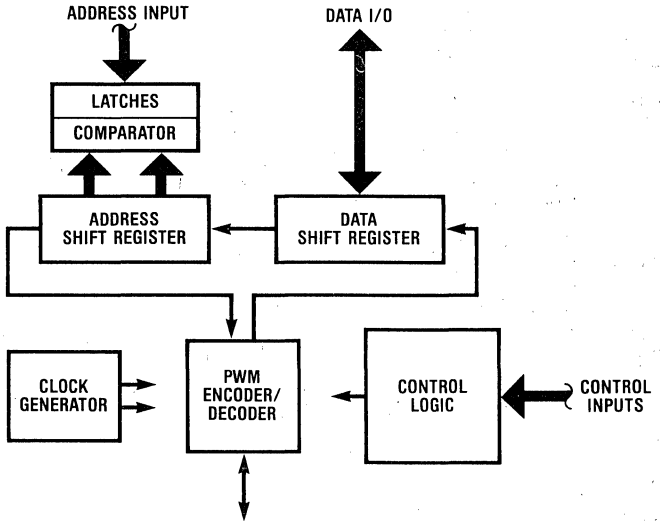



**Figure 2. Circuit Functional Block Diagram**

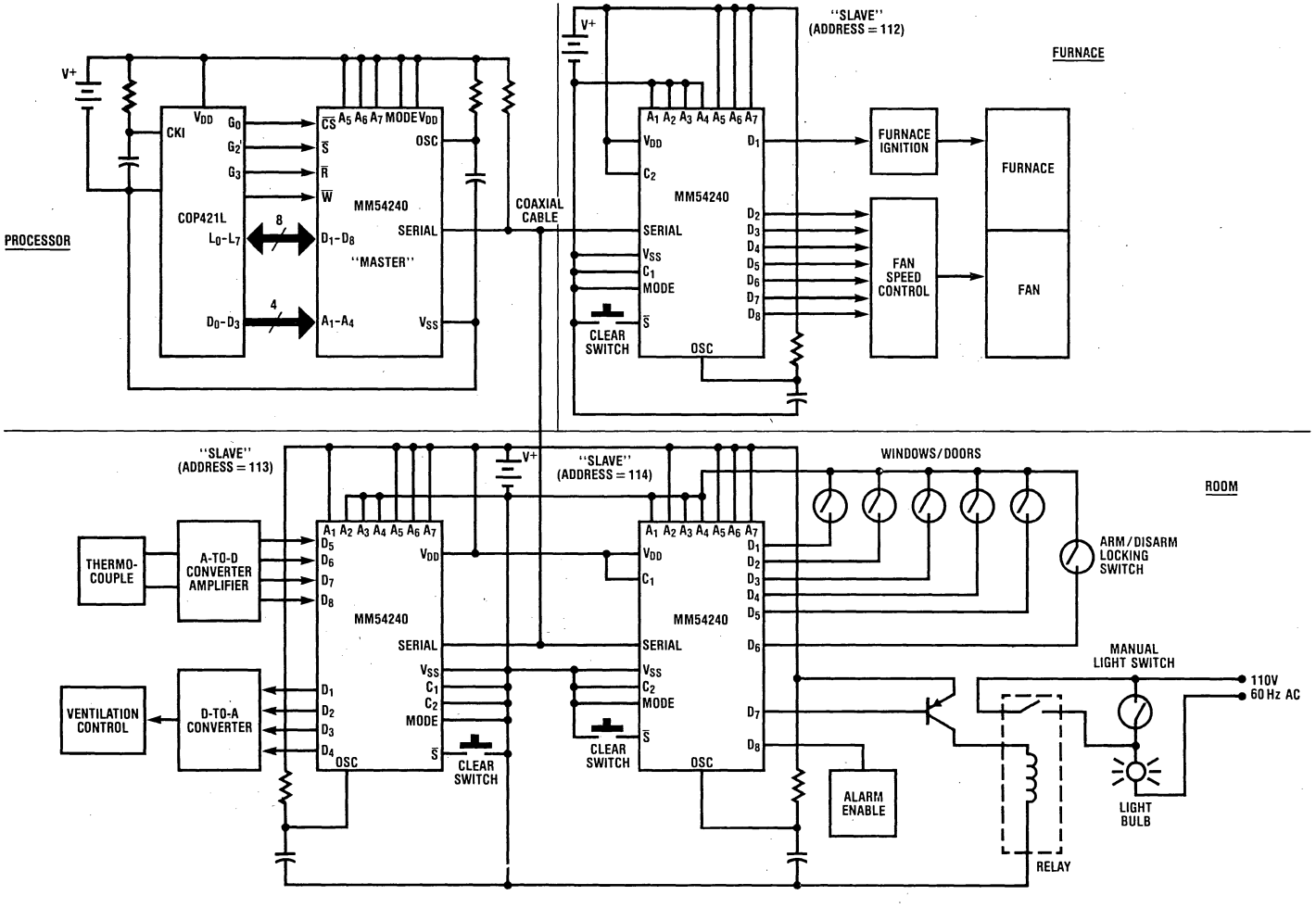



**Figure 3. Temperature Control and Security Application**

## Power Line Transmission

A MM54240 system can be interfaced using other techniques. The pulse width modulated information can be transmitted by carriers like Radio Frequency, infra-red waves, power line transmission or any other suitable medium. For power line transmission applications, an interfacing circuit is used to modulate the information on the 60 Hz AC lines. For the "master" circuit, the processor must generate a signal to control the direction of transmission of the interface circuit. For the "slave" circuit, the chip select ($\overline{CS}$) output is designed for this purpose and can be used directly to control the direction of transmission of the interface circuit.

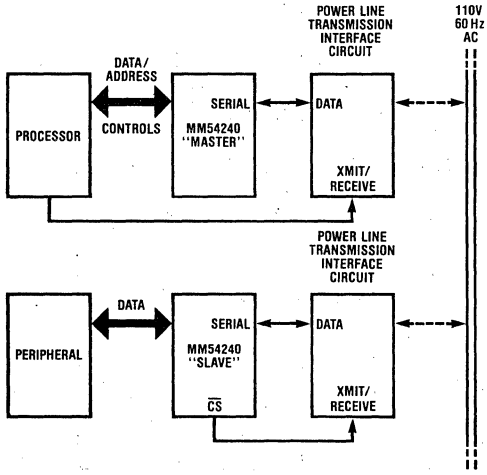



Figure 4. Power Line Transmission Application

## Radio Frequency Transmission

A Radio Frequency transmission system can be built in a similar structure. An I/O multiplexing circuit has to be designed to direct the flow of the transmitted data.

## Conclusion

The MM54240 is a flexible, easy-to-use, and adaptable circuit. It can be used in any application where a serial data transmission is desired. The transmitted data is pulse width modulated. This gives it desirable features such as a low cost oscillator, wide frequency tolerance, and excellent noise immunity. Up to 129 MM54240 circuits can be used for any one system. The circuit is fabricated in the N-channel metal gate process. Designed with National's MICROBUS™ structure in mind, the circuit is easily and directly interfaced to most microprocessor systems.

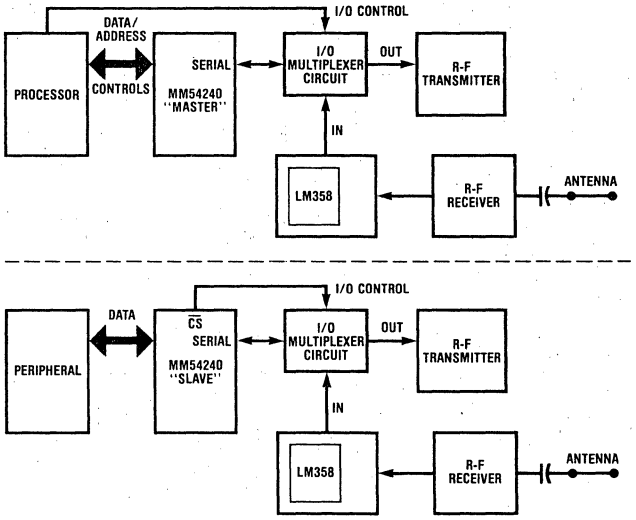



Figure 5. Radio Frequency Transmission Application

# Applications and Uses of the MM5321 TV Camera Sync Generator

National Semiconductor
Application Note 250
Edwin Schoell
May 1980

## 1. Introduction

The MM5321 has been introduced to replace the older MM5320 and correct some difficulties associated with that part. It is a plug-in replacement in almost all applications.

**Major Differences and Improvements are:**

- Horizontal reset control allows resetting to beginning or center of horizontal line.
- Vertical (field) index pulse with both 1.26 and 2.045 MHz clock.
- Improved clocking characteristics.
- Vertical interval always generated after vertical reset pulse at pin 5.
- Vertical sync separator included.

### Power Supplies

The MM5321 is designed to operate from a total supply voltage of 17 volts, or various combinations to supply a total of 17 volts. Interfacing to TTL or CMOS is best accomplished using +5V and −12V as shown in Figure 1. Note that no ground is needed for the MM5321, but it is used as the power return for peripheral chips.

### Input Interfacing

Since the MM5321 is a P-channel device, input switching thresholds are with respect to the most positive ($V_{SS}$) power supply voltage. For this reason, it is important to use the same regulators, or insure the 5V supply to the driver chip is the same as the 5V supply to the MM5321. Problems with poor clocking can often be traced to drive levels not coming to within 1.5 volts of the 5V supply to the MM5321. In some cases, the addition of a 470Ω pull-up resistor from clock input to $V_{SS}$ will solve the problem.

Input clocking problems have also been found with low duty cycle waveforms of the 2.048 MHz clock. A look at the data sheet will reveal that a 50 percent duty cycle is best, but 3 to 4 on/off (43%-57%) ratio is satisfactory. This can be obtained from a properly configured divide by seven counter from a 14.31818 MHz clock.

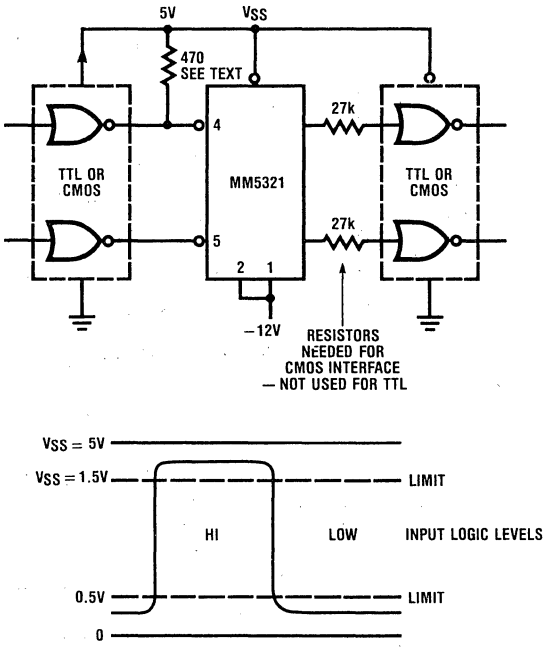



**Figure 1. Input/Output Interfacing**

Two schemes for driving the MM5321 are shown in Figure 2. The first shows a CMOS gate oscillator/buffer, while the second uses a transistor crystal oscillator and a TTL up-counter programmed to divide by seven and produce a 3 to 4 on/off ratio for the MM5321 clock input. In color applications the 14.32 MHz is also divided by four in a shift register type of counter to produce quadrature 3.58 MHz for the systems chroma modulators.

All other inputs should be tied high or low depending on the application of the part, with the exception of the horizontal reset, which is internally pulled down to $V_{GG}$.

### Output Interfacing

The MM5321 will drive 1 TTL load when operated with +5V and −12V supplies as shown in Figure 1. The output structure is an active device to $V_{SS}$ and a current source load to $V_{DD}$, Figure 3. When interfacing to CMOS, a 27kΩ current limit resistor must be used to prevent damage to the CMOS inputs when the protection diode is turned "on".

## 2. Some Applications

The most basic application of the MM5321 is a TV camera sync generator, where it will generate all the usual drive signals as well as high quality composite sync. In this application, little interfacing is needed.

### Genlocking

In some systems, it is necessary to lock the sync generator to another source of sync. This is commonly done by a process called "genlocking" in which the two generators are either fed from the same master (2.045 MHz) clock, or the crystal oscillator of one generator is phase-locked to vertical or horizontal sync of the master generator.

The vertical divider may be reset either by feeding composite sync to the vertical reset control, or by feeding differentiated (short pulses) of vertical sync to the vertical reset pin of the MM5321.

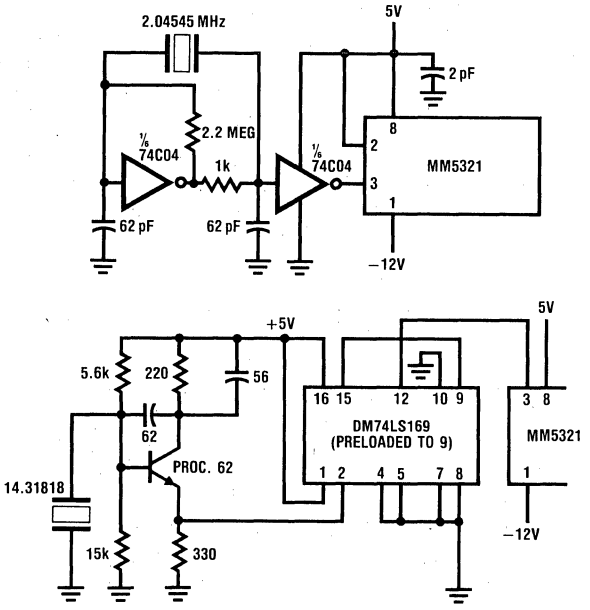



**Figure 2. Input Clock Generating Schemes**

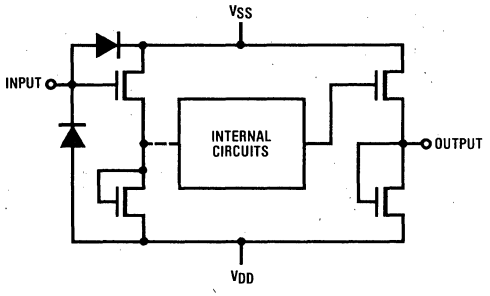



**Figure 3. MM5321 Input/Output Circuits**

Figure 4 shows a genlock application. External horizontal sync is used as a reference to a phase detector which compares to the horizontal sync generated by the MM5321. The error signal controls the clock to the MM5321. In order to insure exact horizontal and vertical timing, external composite sync is fed to the Vertical Reset Control of the MM5321. Circuitry inside the MM5321 decodes the vertical interval of the external sync generating a vertical sync pulse to reset the vertical counter of the MM5321. In effect then, this input acts as a vertical sync separator.

A disadvantage of this approach is that there is a one frame delay between the source and the MM5321, so that serration pulses of the slave will begin one half line earlier than that of the source.

A more precise genlock application is shown in Figure 5. Phase locking is done at 3.58 MHz, vertical is reset with the leading edge of a vertical sync pulse derived from incoming video with the vertical control of the MM5321 resetting to the eleventh half line of generated sync. Composite sync resets the horizontal divider chain for clock periods before horizontal sync resulting in a $\mu$s delay between master and slave. Horizontal drive may be used to reset the horizontal counter to within one clock period or 0.5$\mu$s of master sync. In order not to produce double frequency horizontal drive during the vertical interval, horizontal sync from the master is gated with vertical drive from the slave.

It should be noted in the above application that if the phase lock were omitted, and the 2.048 MHz OSC were allowed to free run, the MM5321 would still maintain vertical and horizontal lock with the source. However, timing errors will build up with time and since the counters in the MM5321 can only be reset in 1 divided by 2.045 MHz or 500 ns intervals, the effect will be to shift the position of horizontal sync 500 ns (1% of 1 horizontal line) every few lines depending on how far apart the two clock sources are.

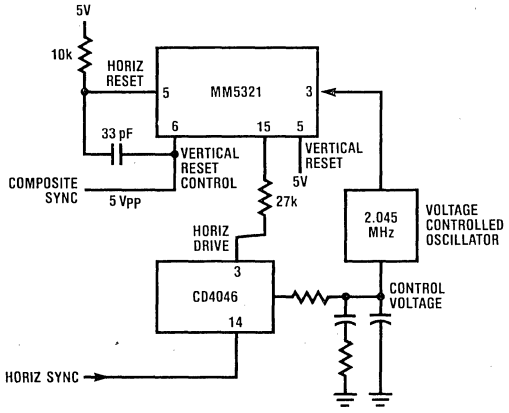



**Figure 4. Genlocking**

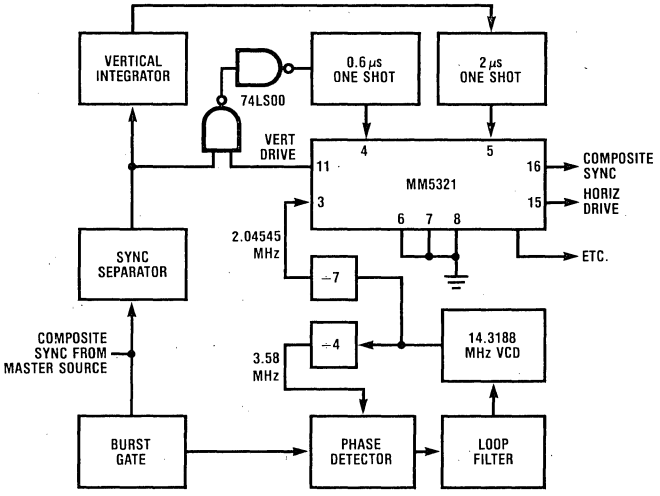



**Figure 5. Genlocking with Color-Burst Phaselock**

As an example, assuming a 0.1% error in clock frequency, after 1000 clock cycles, the error between clocks will be 1 clock cycle or 489 ns and the horizontal counter will reset one line earlier. 1000 clock cycles occur in 489 $\mu$s, or once every 7.7 horizontal lines.

The effect will be a sawtooth appearance on a vertical line displayed on a display.

## 3. PAL and Other Non-Standard Applications

The MM5321 may be used with some external circuitry to generate approximately correct PAL sync. By using the vertical control on the even line after 50 Horizontal lines are counted, the divide by 525 is reset to 5½ lines from zero. The field reference is slightly reduced to give horizontal rate of 50 Hz. It can be seen that for 819 lines, 147 and 152 extra pulses are needed.

Figure 6 shows a realizable circuit, the MM5321 provides all the necessary wave forms as in the 525 lines case. A divide by two provides odd and even field identification. A programmable counter is toggled at half field rate between the counts of 50 and 55; The MM5321 is reset at these times, the combinational logic around the 74C93 is toggled to detect alternately the 10th and 11th counts.

The ensure reliable reset of the 74C93 counter an extra delay was provided by a 74C14 inverter. The vertical drive sets an R-S flip flop at 0 time. The divide by 10/divide by 11 counter is enabled to count five, ten, or eleven pulses, the R-S flip flop is reset by the output of the divide by 5 counter.

The circuit now awaits the next vertical blanking pulse.

### Specifications

| | |
|---|---|
| Line Period | 64 $\mu$s |
| Line Blanketing | 11.2 $\mu$s |
| Front Porch | 1.6 $\mu$s |
| Horizontal Pulse | 4.8 $\mu$s |
| Back Porch | 4.8 $\mu$s |
| Burst | 2.4 $\mu$s |
| Field | 50 Hz/312 = 313 |
| Duration of Field Sequence | 91 H |
| Field Blank. Int. | 21 H lines |
| No. of Serrations | 6 |

In conclusion it can be seen that this sync generator does not exactly conform with CCIR specs. However, there should be a few applications where these waveforms would be ample.

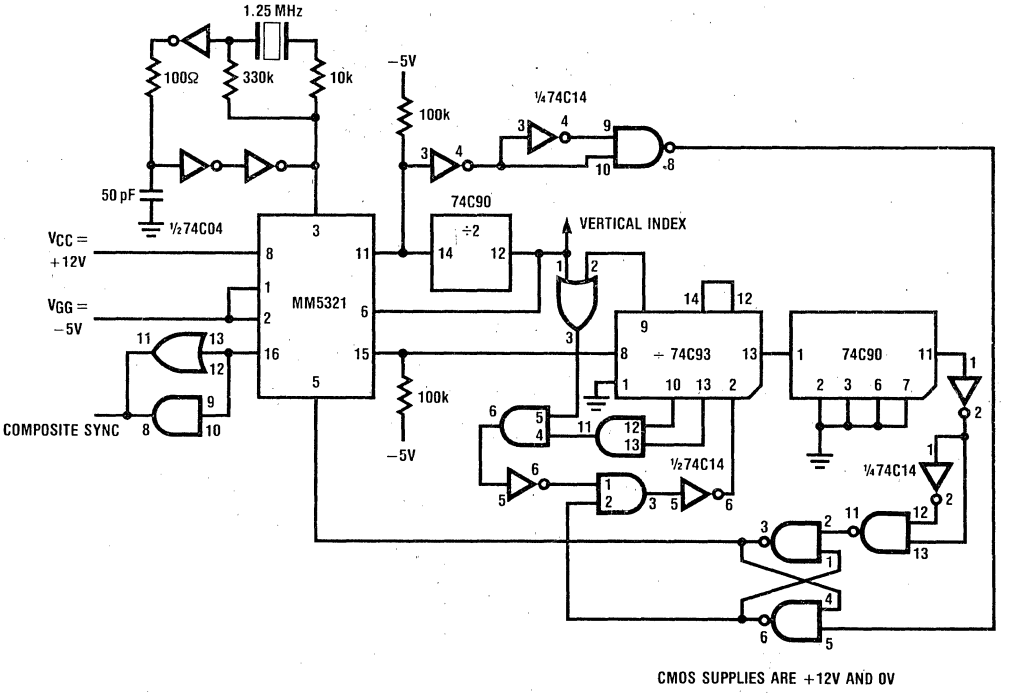



Figure 6. PAL Application

# A Broadcast Quality TV Sync Generator Made Economical through LSI

National Semiconductor
Application Note 251
Robert B. Johnson and
Eugene H. Campbell
May 1980

The growing number of applications of video tape recorders and TV cameras in the consumer market have resulted in the need for a single-chip LSI integrated circuit TV camera sync generator. The National Semiconductor MM5321 TV Camera Sync Generator has been developed to economically provide the basic sync functions for color and monochrome, 525 line, 60 Hz, interlaced applications — and provide it with the reliability and accuracy of a digital IC system. A Metal-Oxide-Semiconductor (MOS) technology was chosen as the most economical method of obtaining the necessary circuit density and speed.

Figure 1 shows the simplified block diagram and Figures 2 through 5 are the timing diagrams of the generator.

All inputs and outputs of the 14-pin device are TTL compatible without the use of external components. Two supplies are required, with the nominal difference between them 17 volts. Ambient temperature may be varied between −25°C and +70°C.

The output functions provided are Horizontal Drive, Vertical Drive, Composite Blanking, Composite Sync, and Color Burst Gate. In addition, a Field Index output function identifies a particular field, and a Color Burst Sync output presents a pulse at half the horizontal rate, but otherwise identical to the Color Burst Gate, and may be used to synchronize the color burst with the generator.

All output functions are derived from the clock applied to the Master Clock input. The user may select either of two input frequencies by selecting the proper horizontal divider, which is accomplished by hard-wiring the Divider Control pin to either the $V_{SS}$ (most positive) or $V_{GG}$ (most negative) power supply.

In color applications, a frequency four times the color burst is usually available to generate the 0°C and 90°C color sub-carrier signals. Dividing that frequency by seven results in 2.04545 MHz, which is the input clock signal to be used when the Divider Control pin is connected to $V_{SS}$. With the control pin wired to $V_{GG}$, the horizontal divider is programmed to acept an input signal eighty times the horizontal rate, or 1.260 MHz.

The horizontal divider is essentially a 65-bit shift register which can be shortened to 40-bits with the Divider Control logic. Control logic also selects the proper set of register taps used for decoding the horizontal timing edges.

One of the outputs of the horizontal divider is a signal used to drive the ten-stage vertical counter and a 42-bit shift register, which together provide the verical division and timing edges.

Shift registers are usually very efficient logic blocks in MOS designs, which is why they were selected for many of the counters in this product. Parasitic capacitances may be used to store charge for periods of time that are

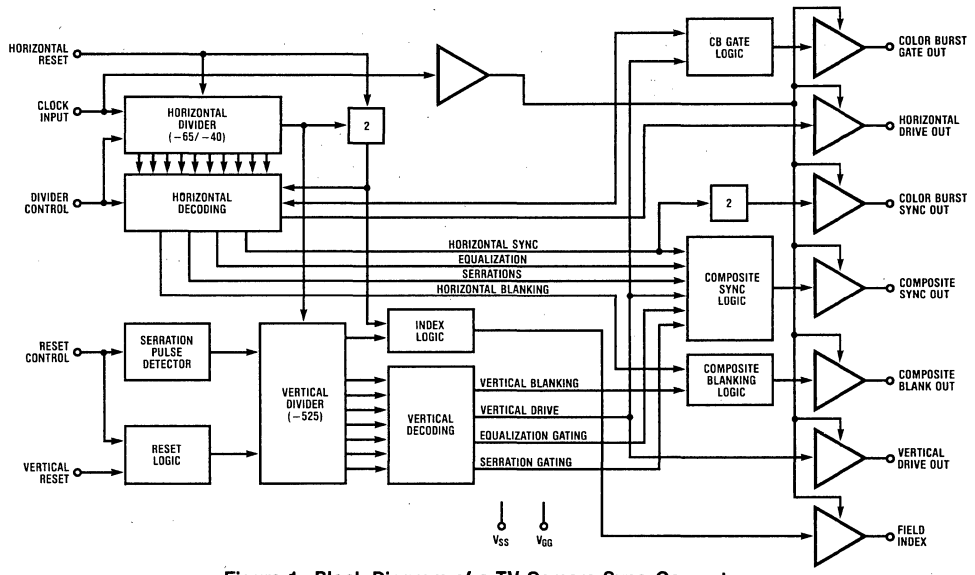



**Figure 1. Block Diagram of a TV Camera Sync Generator Fabricated with MOS/LSI Techniques**

essentially dependent only on semiconductor junction characteristics. Thus, in MOS it is possible to design both dynamic and static shift registers. Dynamic registers were used for both the vertical and horizontal counters because in each case the clock frequency is well above any minimum limitation due to leakage current considerations, and they offer a layout/size advantage over static type cells. The configuration selected uses ten transistors and is capable of being reset to either a "1" or "0" logic state.

The vertical divider is comprised of DC flip-flops configured as a ten-stage short-cycled, modulus 525, ripple counter. Each stage is resettable, and to accomodate additional vertical reset versatility, stages 1, 2, and 8 can be set or reset.

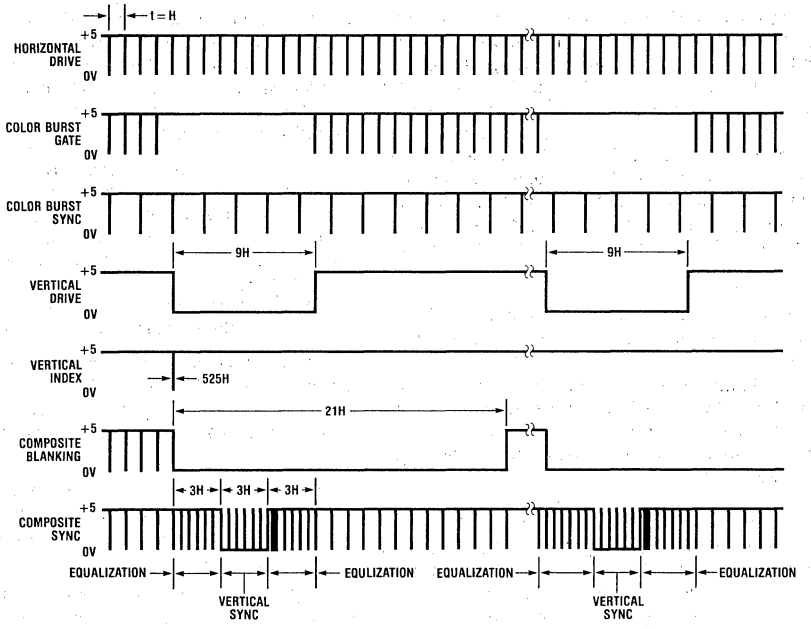



**Figure 2. Sync Generator Output Waveforms**

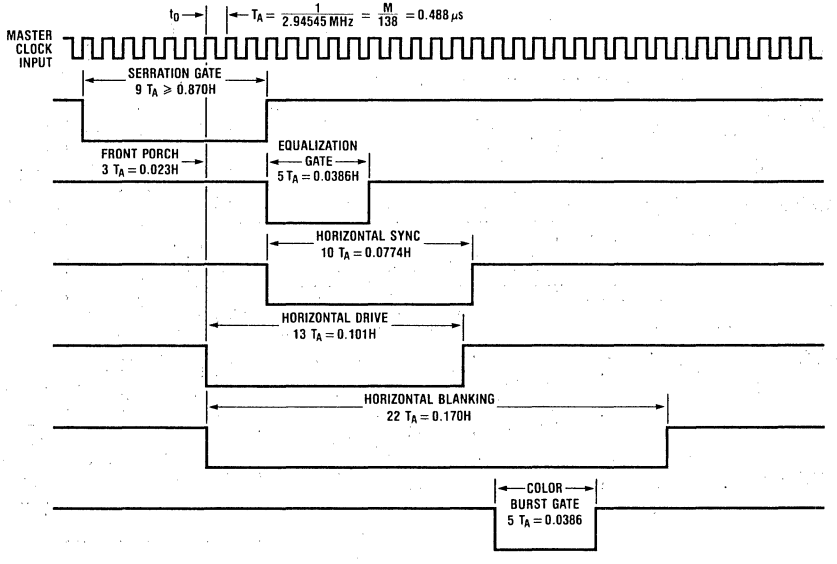



**Figure 3. Horizontal Timing Diagram with the Input Clock Frequency Equal to 2.04545 MHz**

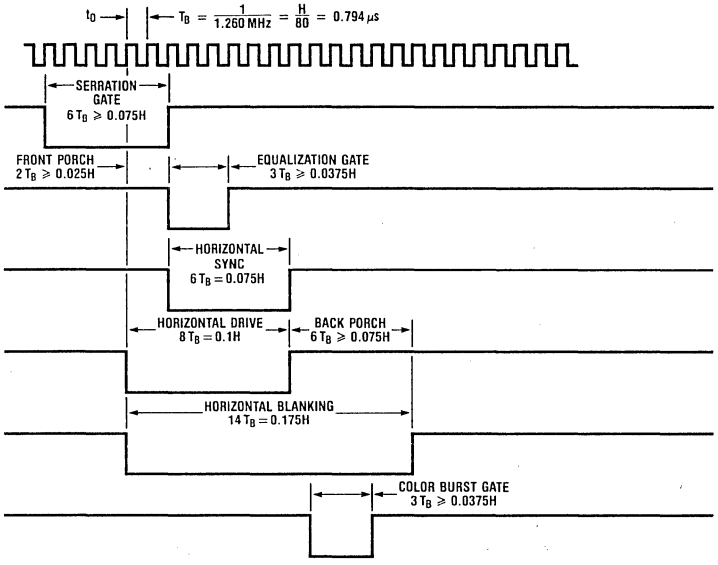



**Figure 4. Horizontal Timing Diagram with the Input Clock Equal to 1.260 MHz**

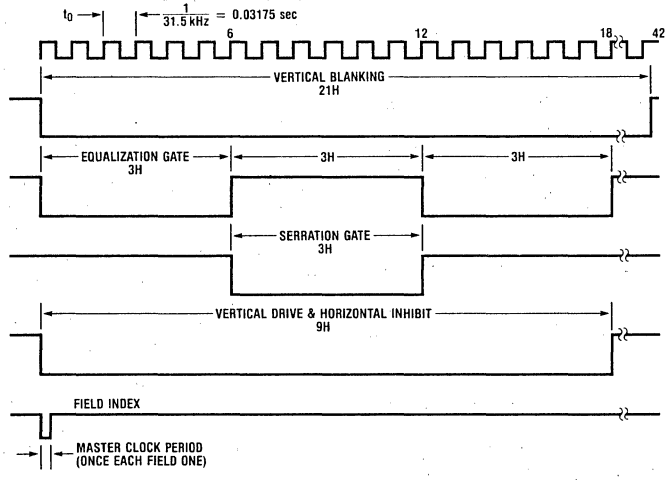



**Figure 5. Vertical Timing Diagram**

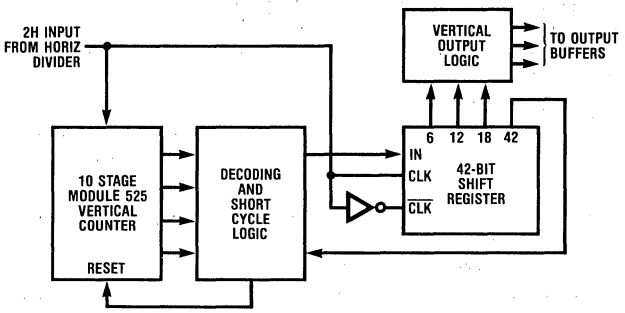



**Figure 6. Simplified Vertical Timing Logic**

Figure 6 indicates the method of generating the vertical output functions. Decoding logic detects the 525th state and short cycles the counter by resetting it to zero. Simultaneously, the input of the 42-bit shift register is set to zero and the vertical blanking and equalization gates are initiated. Six register clock periods later, the equalization gate is terminated and the serration pulse is initiated by the arrival of a zero state at the sixth bit of the shift register. Similarly, the serration gate is terminated and the equalization gate reinitiated when a zero is detected at the 12th tap and, finally, the equalization gate is terminated when the 18th tap changes to a zero. The vertical drive pulse is also initiated when the register input goes to a zero, and is terminated when the zero reaches the 18th bit. The vertical blanking pulse lasts until the zero propagates to the 42nd bit, at which time the register input is reset to a logical "1" level.

In some applications, particularly video recorder tape editing, it is necessary to identify which field of the vertical frame the system is in. For that purpose, the generator derives a Field Index pulse which identified field one by occurring for two input clock periods at the leading edge of the vertical blanking pulse of field one. Field one is defined as the field with a whole scanning line interval between the equalizing pulse and the last line sync pulse of the preceding field.

When designing MOS circuits, one must be aware of the effects of power supply variations, ambient temperature excursions, and process variables on circuit performance. This is the case in design of most circuits of course, but MOS tends to be more sensitive than bipolar circuits due to increased parasitic capacitance and limited current drive capabilities. The speed of any MOS product is essentially dependent upon how fast critical capacitive nodes can be charged and discharged. The charging or discharging current is in turn a function of the size, the voltages applied to, and the threshold and gain factor of the transistor(s) supplying the current. Threshold and gain factor are functions of process variables such as gate oxide thickness, the type of substrate material and its impurity concentration. They are also affected by temperature, which reduce the fermi potential (decreasing threshold), and modifies the carrier mobility in the transistor channel (which lowers the gain factor). the reduction in gain factor generally has more effect than the change in threshold, resulting in an overall reduction in speed with increasing temperature.

As far as the sync generator is concerned, this variation in performance as a function of environmental and power supply conditions could cause skewing of individual output timing edges, reducing the accuracy of the sync functions. Careful design essentially eliminates this problem in the MM5321. First, all output functions were matched for total logic delay by simulating circuit performance for all environmental and process variations, and then optimizing the delays to the output buffers. Second, all output functions are resynchronized at the outputs by an internal clock signal running at the input clock rate, with its own optimized delay characteristics with respect to the horizontal divider clock. For all worst-case conditions the output functions reach the synchronizing point before the synchronizing clock. Third, all the output buffers themselves are identical and therefore have matched delays. Thus, the design results in output functions whose timing delays are matched with respect to each other, but will have differences in delay with respect to the input clock on a part to part basis (due to variations in process variables). Even on a part to part basis, maximum differences in delay between two parts with the maximum allowed process variation should be less than 200ns, or 0.003H, at similar temperature and power supply values.

The output buffers are push-pull using the circuit configuration shown in Figure 7. The output transistors Q1 and Q2 provide the sink and source characteristics shown in Figure 8. When interfacing directly with TTL, the 800Ω resistor serves to limit the excess sink current supplied to the TTL clamp diode, by reducing the gate drive to Q2. This minimizes excessive power dissipation on the chip and protects the TTL diode. Q8 is the logic transfer device driven by the synchronizing clock.

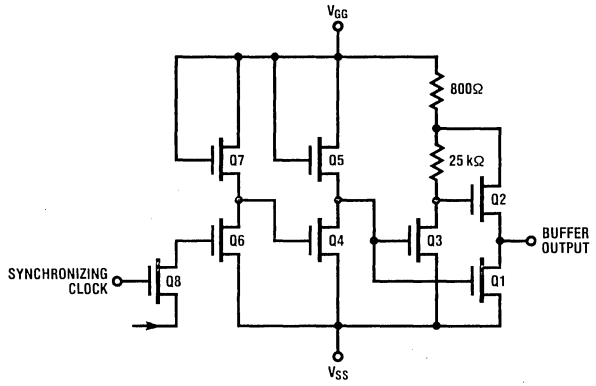



**Figure 7. Schematic of TTL Compatible Push-Pull Buffer used on All Outputs of the Sync Generator**

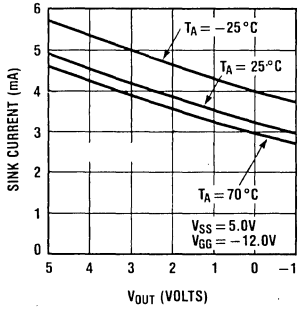



**Figure 8a. Typical Output Sink Current as a Function of Output Voltage**

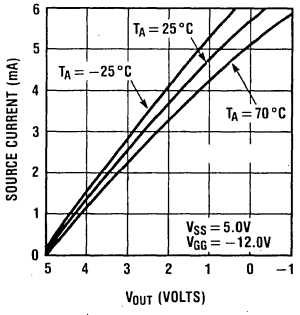



**Figure 8b. Typical Output Source Current as a Function of Output Voltage**

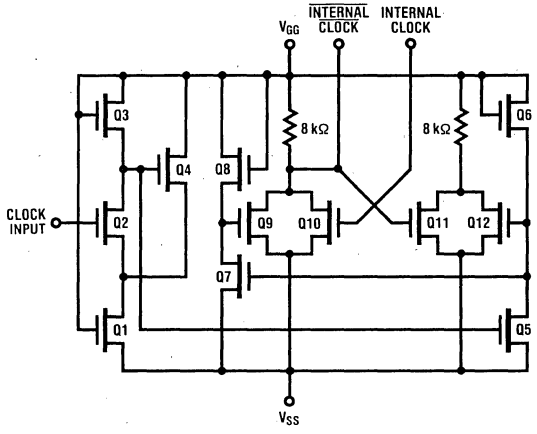



**Figure 9. Schematic of Input Clock Buffer**

The most critical circuitry in the generator, from the standpoint of speed, is the input clock buffer (Figure 9). The buffer is designed to generate a two-phase, full power supply amplitude clock signal from the single-phase low amplitude input signal. Q1 through Q4 constitute a Schmitt trigger type input stage that guarantees a trip-point range of $V_{SS}$ – 4.2V maximum for "0" levels, and $V_{SS}$ – 2.0V minimum for TTL "1" levels. When interfacing directly with TTL, the normal supplies will be +5 volts connected to $V_{SS}$, and −12V connected to the $V_{GG}$ pin. For a tolerance of 5% on the $V_{SS}$ supply, the guaranteed trip-points decipher to a required input level more negative than 4.75V-4.2V, or 0.55V, for the "0" level, and a required level more positive than 4.75V-2.0V, or 2.75, for the "1" level. These levels are obtainable from standard TTL without any external interface components. Q10 and Q11 are feedback latches which eliminate internal clock overlap problems.

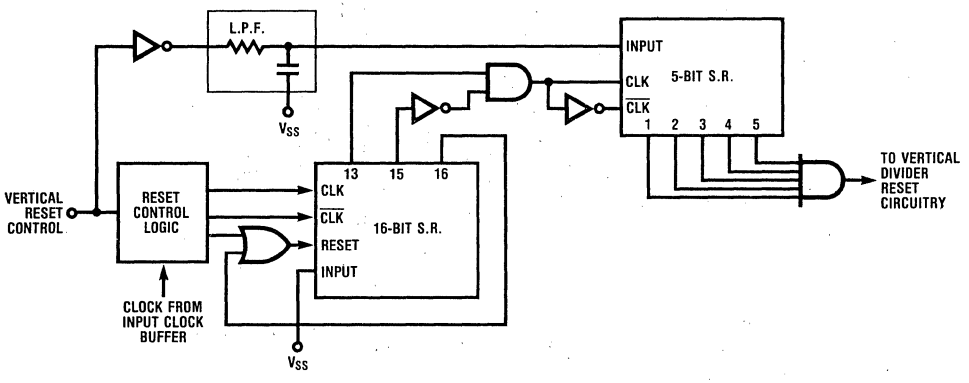



**Figure 10. Basic Logic for Detecting Proper State of the Composite Sync Input Signal for Resetting the Vertical Divider in "Gen-Lock" Operation**

To provide as much versatility as possible, a variety of divider reset ("gen-clock") features have been included. The horizontal and vertical dividers have individual Vertical and Horizontal Reset inputs which allow independent resetting of the appropriate divider. With the inputs tied together, both dividers may be reset simultaneously.

The vertical divider may be reset to either of two states, depending upon the DC level of the Reset Control pin. If the Reset Control is tied to $V_{SS}$, the most positive supply, a TTL "1" to the "0" level transition on the Vertical Reset pin will reset the vertical divider to all zeros, which is time zero as defined by the vertical timing diagram. With the Reset Control returned to $V_{GG}$, a Vertical Reset pulse will reset the vertical divider to the fifth serration pulse (eleven 0.5H time intervals from time zero). This allows the reset pulse to be generated by analog detection of a composite sync or video signal, and used to gen-lock the slave sync generator within the same field interval. The horizontal divider is always reset to zero, as defined by the horizontal timing diagram.

The Field Index output pulse occurs once during each field one at time zero and last for two master clock periods. It can be used to gen-lock similar sync generator chips by connecting it to their Vertical Reset inputs and wiring the Reset Control to the $V_{SS}$ supply.

Another method of resetting the vertical divider is provided by using the Reset Control pin as an input for a composite sync signal from which gen-locking is desired. The slaved generator detects the fifth serration pulse and resets the vertical divider to the proper state (Figure 10).

The reset control logic generates a two-phase clock with a frequency equal to the input clock rate anytime the composite sync input signal is more negative than the Reset Control trip-point. A 16-bit dynamic shift register with its input connected to $V_{SS}$ is driven by the modulated clock signal. When the composite sync input becomes more positive than the Reset Control trip-point, or if the 16th bit becomes a "1", all sixteen bits of the register are reset to zeros. If the composite sync signal remains low for fifteen master clock periods, another two-phase signal is generated which acts as the clock for a 5-bit shift register used to store the sampled state of the inverted (and filtered) composite sync signal. The sample is the average value of the filtered signal during an approximately 200 ns sampling window occurring just before the fifteenth master clock time after the composite sync input signal initially went low. The input trip-point of the 5-bit register determines whether the sampled signal is stored as a "1" or "0" logic state.

Fifteen input clock periods equal a time of 7.3 $\mu$s at an input clock frequency of 2.04545 MHz, and 11.9 $\mu$s when the input rate is 1.260 MHz. The only interval of the composite sync waveform which is legitimately low during this time is the vertical sync pulse. In the present design, the first five serrated intervals must be successfully detected before the vertical divider is reset to the proper state. The limitation in this design may be the difficulty in acutally acquiring legitimate detection due to excessive noise and missing pulses in the composite sync input signal. If this proves to be the case, it is possible to eliminate the second and/or fourth bits of the 5-bit register as detection requirements. This should improve the statistical probability of getting an initial gen-lock condition within a reasonable time.

As illustrated above, the Reset Control input has a dual function. It selects the reset state of the vertical divider when hardwired to either $V_{SS}$ or $V_{GG}$, and acts as a dynamic input when gen-locking is to be established using a composite sync input signal. When using the Reset Control as the input for a composite sync signal, the Vertical Reset pin should be hardwired to $V_{SS}$.

The MM5321 TV Sync Generator has been designed with both versatility and economy as the primary objectives. We feel it exemplifies the role of MOS/LSI standard products can play in providing useful consumer products in a manner that both large *and* small volume users will find attractive.

# Integrated Circuit Combination Provides Digital Frequency Readout with Digital Clock for Radios

*A digital frequency readout is one feature which can easily upgrade existing radio receivers, as well as provide the focal point for new designs of clock radios, 3-in-1 receivers, mid-fi and high-fi products. It is even more desirable when it can provide a full feature clock radio function using a common display.*

The MM5430 from National Semiconductor Corporation is a radio frequency counter integrated circuit which drives a common anode LED direct-drive display to provide a digital frequency presentation of the frequency to which a radio receiver is being tuned. It is an add-on circuit; no changes in the basic receiver circuitry are required. It may either replace existing dial pointer frequency indicators or be used in conjunction with them.

The companion clock IC, the MM5402N also drives a common anode LED direct-drive display, so both ICs, with few additional components, can drive a common display.

For those existing receiver designs which already have a digital clock, the retrofit is relatively simple. The clock and clock display are replaced by the MM5402 and NSB584 displays. The MM5430 and a few other components are added.

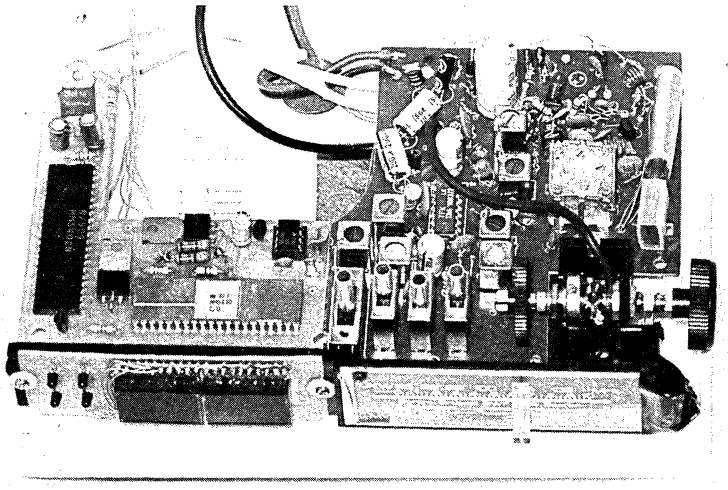
The MM5430 counter accepts an input signal from the FM or AM local oscillator, offsets it by the IF frequency, and displays it. Using the pin selectable options, IF offsets for 10.6 MHz, 10.7 MHz or 10.8 MHz in FM and 262.5 kHz, 455 kHz or 460 kHz in AM may be chosen.

The frequency base for the counter can be derived from the 50 Hz or 60 Hz mains, or by using a commonly available 4.194 MHz crystal. If the crystal option is used, the MM5430 provides a 60 Hz output which can be used for the frequency input of the MM5402 clock. This allows the clock function to have quartz crystal accuracy and allows the system to be operated from DC sources, so the system can be used in automobile radios.

A pin select option provides for U.S. or European channel spacing, and LW band may be displayed in the AM mode.

When the frequency remains fixed for longer than four seconds, the display automatically shifts to the time display. A change in frequency (when tuning to another station) causes the system to read the frequency automatically.

Connection to the tuner is simple. Take-off points can be an additional winding on the local oscillator coil or from some hot point of the oscillator circuit. A divide-by-100 prescaler, such as the DS8629N must be used for the FM input. Based on factors such as low LO level or oscillator pulling, additional buffering may be required, as well as some realignment of the frequency and tracking.



**The MM5430, MM5402 Circuit Applied to an Existing Clock Radio Chassis; Note that the Original Dial Pointer is Still Used.**

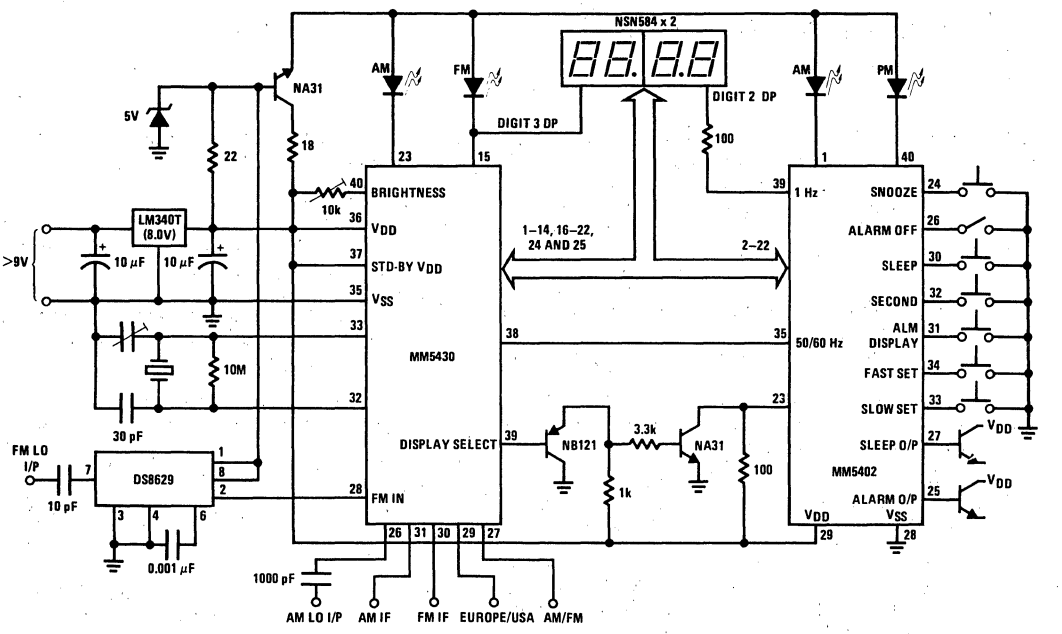



**FIGURE 1. MM5430 and MM5402 DC Operation**

Section 7

# Custom MOS/LSI

# National Semiconductor your choice for Custom MOS/LSI Circuits

National Semiconductor's custom MOS/LSI circuits have helped many companies meet the increasing demands for compact, yet sophisticated electronics. We're ready to put our vast knowledge and experience to work in designing and producing high-quality, reliable, cost-effective custom circuitry for your new product. With your success as our goal, we'll dedicate our resources to your needs and hope that you will consider us your own semiconductor facility.

National Semiconductor has acquired broad capabilities in the design and fabrication of semiconductors since our founding in 1959. As one of the world's largest suppliers of semiconductor products, we have a line of over 6000 standard circuits, including linear IC's, microprocessors, memory components, digital logic circuits, hybrid circuits, converter products, optoelectronic components and displays, discrete transistors, modules and transducers, as well as MOS/LSI circuits.

Our custom MOS/LSI design and manufacturing capabilities serve a variety of worldwide markets. We've developed circuits for timing devices, electronic organs, television receivers, A/D and D/A converters, calculators, keyboard encoders, medical electronics, and many others.

As your custom MOS/LSI supplier, we are keenly aware of the direct relationship between your success and ours. Your timely profitable product introduction remains our primary objective throughout the development cycle.

**A Working Partnership**

As the source of your custom MOS/LSI circuit, National Semiconductor will play a key role in the development and introduction of your product. Ideally, we'll act as one of your in-house design groups, establishing active communications with you early in the program and maintaining them throughout the development cycle. If your product is still in the "idea" stage, we can help you to clearly define it before circuit selection even begins.

**Working Alternatives**

National Semiconductor's custom design group is prepared to work with your organization on any level, from defining your system from concept to fabricating a circuit from your tooling. Our design specialists are available for consultation — with either your staff or an independently-contracted design firm — at any point in the development cycle.

Depending on your in-house capabilities, you may prefer to design your own circuit and supply us with either pattern generation tapes or working plates. Under these circumstances, access to your test tapes and close coordination between your logic designers and our test engineers will assure rigid adherence to acceptance procedures.

# Working together to get the job done Custom MOS/LSI Circuits

**Cost Considerations**

Once designed, a custom MOS/LSI circuit is naturally more cost-effective than standard circuits because you do not pay for unused capabilities. However, projected sales volume should be sufficient to justify the initial design investment. As demonstrated in the break-even graph, the custom MOS/LSI approach becomes cost-effective at around 40 to 50 thousand units. Nonetheless, a careful analysis of increased operational costs associated with multiple components may indicate a single, dedicated custom chip for your product — even at lower volumes.

**Costs that can rise with the number of components**

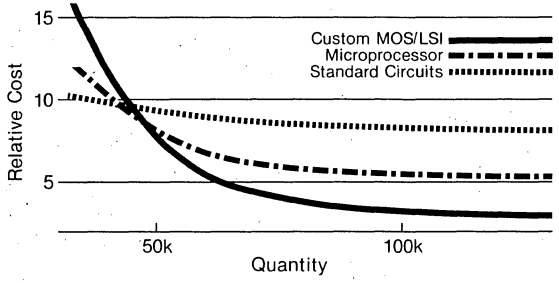
**Parts handling** — incoming inspections, inventory carrying, inventory control

**External components** — interconnects, PC boards, cabinetry

**Assembly** — lead insertions, solderings, initial checkout inspections

**Warranty service** — field trips, troubleshooting procedures, parts replacement

**Comparative Unit Cost Graph**

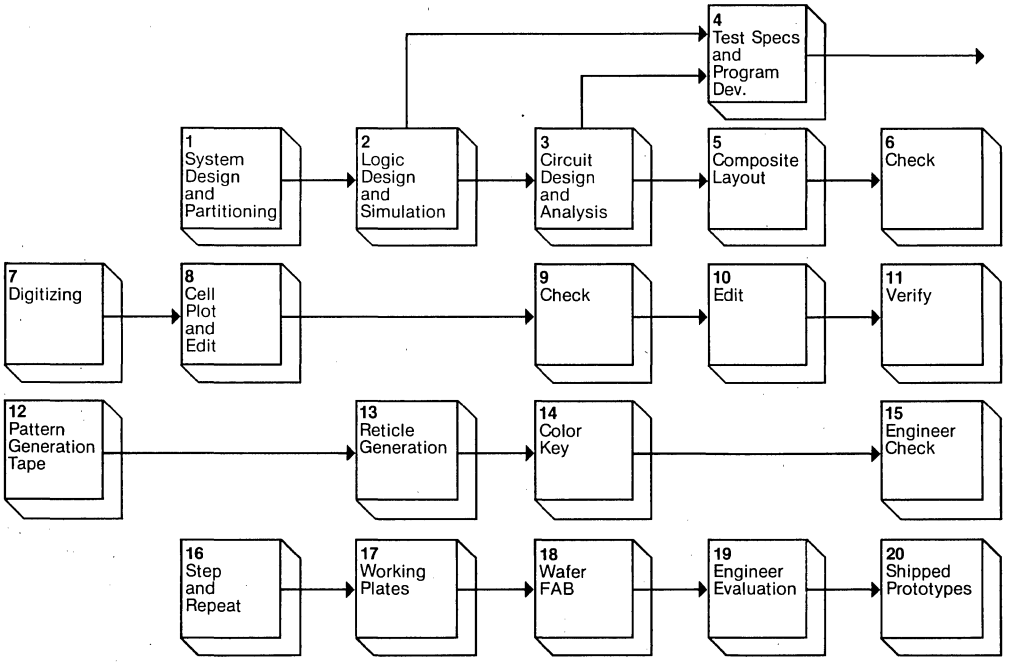



**Competitive Considerations**

Even when standard components appear to be more cost-effective than custom MOS/LSI, the competitive advantages of custom circuitry are often well worth the initial investment.

- **Increased market share** — Because custom MOS/LSI is on the forefront of technology, it can enable you to achieve new, exciting product features that might otherwise be impossible. Reduced costs allow you strategic pricing advantages.
- **Extended product life cycle** — An exclusive circuit design can extend your product's lead in the market place by a year or more and shorten your investment payback period.
- **Confidentiality** — At National Semiconductor, the proprietary nature of your custom design is carefully safeguarded — an important consideration in today's highly competitive environment. A custom circuit would be difficult to copy as compared to the purchase of standard parts.

# The Custom MOS/LSI Development* Cycle... Bringing Your Ideas to Market.




*Our design specialists are available for consultation — with either your staff or an independently-contracted design firm — at any point in the development cycle.

**Product Considerations**

A custom MOS/LSI circuit is designed specifically to carry out the functions that meet your product objectives, with no excess circuitry. Often, a single chip is adequate to achieve the desired result. As product function becomes more complex, the effect of such dedicated circuitry on product marketability and customer confidence increases.

- **Product reliability** is enhanced with fewer circuit packages and fewer interconnections between circuits.
- **Product styling options** are wider because a single, dedicated circuit minimizes space requirements.
- **Energy efficiency** is increased over electromechanical or discrete circuitry. With reduced power requirements, potential heat problems are also reduced.

# A Working Partnership

**Fabrication**

The key to providing your product with an optimal chip design is being able to select a fabrication process that will achieve the desired performance characteristics. National Semiconductor offers you maximum versatility by having capabilities for seven different MOS processes. A summary of those processes, their key features, and most common applications is provided on the following MOS Process Chart.

**National Semiconductor's Available MOS Processes for Design**

| MOS Process | Key Features | Maximum Clock Frequency (MHz) | Supply Voltage Range (volts) | Applications |
|---|---|---|---|---|
| Low-voltage, metal-gate CMOS | highly complex, high speed, noise immunity, low power consumption and bipolar compatibility | 3 | 1.5-5.5 | circuits that must operate from low-voltage cells, i.e., watch and camera circuits |
| N-channel, silicon-gate | high speed and packing density, full bipolar compatibility | 4 | 4-8 | large, high-speed memories with on-chip clocks and drivers, such as 2102 and controller-oriented processors and RAM-intensive circuits |
| N-channel, metal-gate | high voltage, speed, packing density | 3.5 | 4-12 | ROM-intensive circuits, fast-turnaround fabrication, tele-communications and security system circuits |
| High-voltage, metal-gate CMOS | high voltage, speed, noise immunity and low power | 5 | 3-15 | remote-control processors, 9-volt-battery operating systems such as hand-held games, telephone operating systems, automotive |
| P-channel, metal-gate | relatively simple, wide operating voltage range, high noise immunity | 2 | 5-29 | uses include calculator and clock chips, VF-drive, minimum cost products |
| High-density, double poly-silicon-gate CMOS | high-speed, packing density, high perfor-mance, low power and wide operating voltage range | 15 | 2-8 | memories, advanced telecommunications circuits, filters |
| High-density scaled N-channel silicon-gate XMOS | very high density, high-performance | 6 | 4-8 | memories, micro-processors, controllers, general products |

# National Semiconductor your choice for Custom MOS/LSI Circuits

**Let's get together now — during your initial design stages — to develop the right program for your company. Call your National Semiconductor sales representative or one of our field applications engineers — or contact us directly at one of the offices listed on the back cover — to arrange a preliminary meeting.**

**For further information send for our Custom Booklet which outlines the expertise, resources, and support available for your Custom Circuit.**

# A Working Partnership For Customer Tooling

**National Semiconductor's Available MOS Processes for Customer Designed Circuits**

| MOS Process | Key Features | Maximum Clock Frequency (MHz) | Supply Voltage Range (volts) | Applications |
|---|---|---|---|---|
| Low-voltage, metal-gate CMOS | highly complex, high speed, noise immunity, low power consumption and bipolar compatibility | 3 | 1.5-5.5 | circuits that must operate from low-voltage cells, i.e., watch and camera circuits |
| N-channel, silicon-gate | high speed and packing density, full bipolar compatibility | 4 | 4-8 | large, high-speed memories with on-chip clocks and drivers, such as 2102 and controller-oriented processors and RAM-intensive circuits |
| N-channel, metal-gate | high voltage, speed, packing density | 3.5 | 4-12 | ROM-intensive circuits, fast-turnaround fabrication, tele-communications and security system circuits |
| High-voltage, metal-gate CMOS | high voltage, speed, noise immunity and low power | 5 | 3-15 | remote-control processors, 9-volt-battery operating systems such as hand-held games, telephone operating systems, automotive |
| P-channel, metal-gate | relatively simple, wide operating voltage range, high noise immunity | 2 | 5-29 | most widely used process in the industry; uses include calculator and clock chips, VF-drive, minimum cost products |

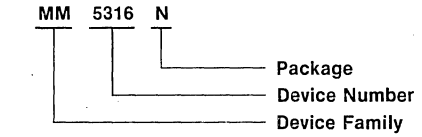
**Let's get together now — during your initial design stages — to develop the right tools to optimize you design for your company. Call your National Semiconductor sales representative or one of our field applications engineers — or contact us directly at one of the offices listed on the back cover — to arrange the key preliminary meeting.**

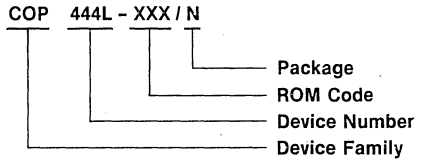
**For further information send for our Custom Booklet which outlines the resources and support available for your own design.**

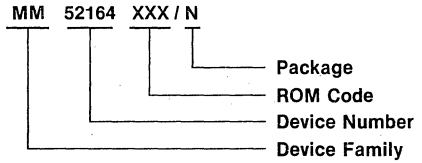
Section 8

# Ordering Information/ Physical Dimensions

# Ordering Information/Physical Dimensions




**Package**

D — Glass/Metal Dual-In-Line Package
J — Ceramic Dual-In-Line Package
N — Epoxy Dual-In-Line Package

**ROM Code**

COPS — Magnetic Disk, PROM, or Tape
MAXI-ROM — PROM or Tape
Contact your local sales office for submittal procedures.

**Device Number**

4-, 5-, or 6-Digit Number Suffix Indicators

**Device Family**

MM — MOS Monolithic
COP— Controller Processor

## Packages

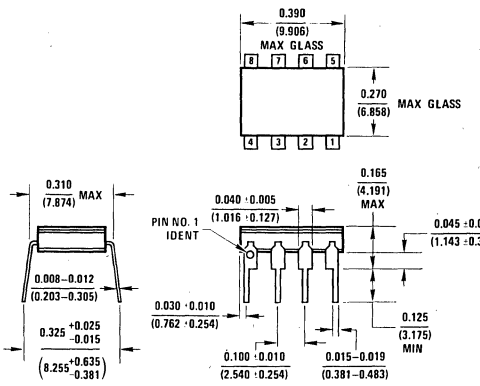
### Dual-In-Line Packages

(N) Devices ordered with "N" suffix are supplied in molded dual-in-line packages. Molding material is EPOXY B, a highly reliable compound suitable for military as well as commercial temperature range applications. Lead material is Alloy 42 with a hot solder dipped surface to allow for ease of solderability.

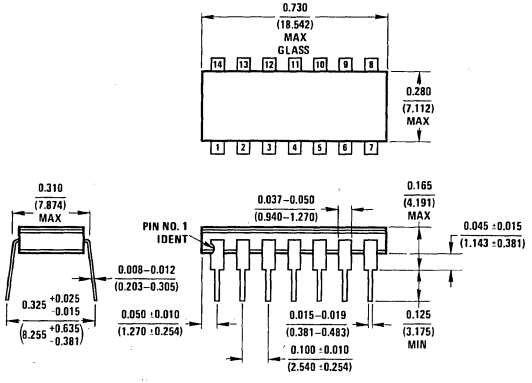
(J) Devices ordered with the "J" suffix are supplied in either 14-pin, 16-pin, or 24-pin ceramic dual-in-line packages. The body of the package is made of ceramic and hermeticity is accomplished through a high temperature sealing of the package. Lead material is tin-plated kovar.

(D) Devices ordered with the "D" suffix are supplied with glass/metal dual-in-line packages. The top and bottom of the package are gold-plated kovar, as are the leads. The side walls are glass, through which the leads extend, forming a hermetic seal.

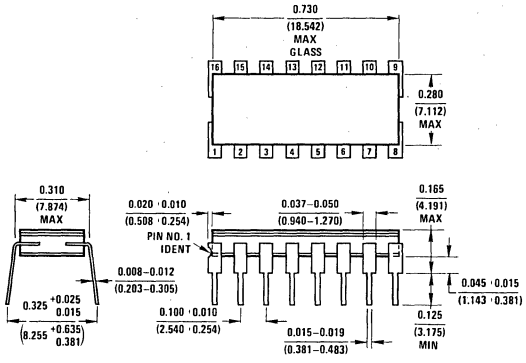
# Physical Dimensions inches (millimeters)



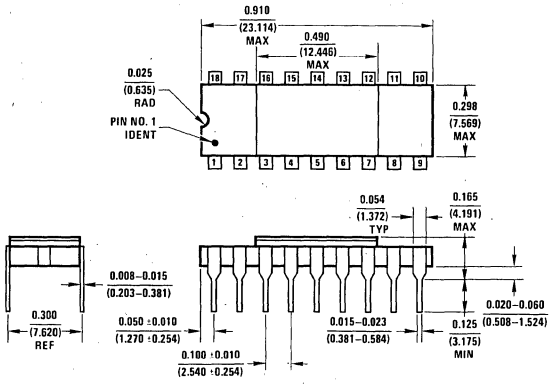

**Package 1**
**8-Lead Cavity DIP (D)**
**NS Package Number D08C**



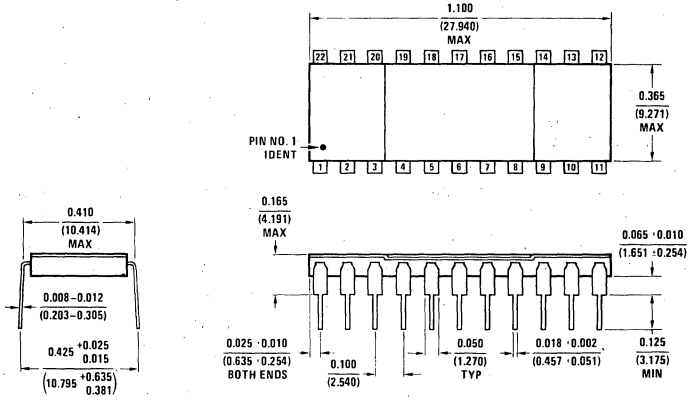

**Package 2**
**14-Lead Cavity DIP (D)**
**NS Package Number D14A**




**Package 3**
**16-Lead Cavity DIP (D)**
**NS Package Number D16A**

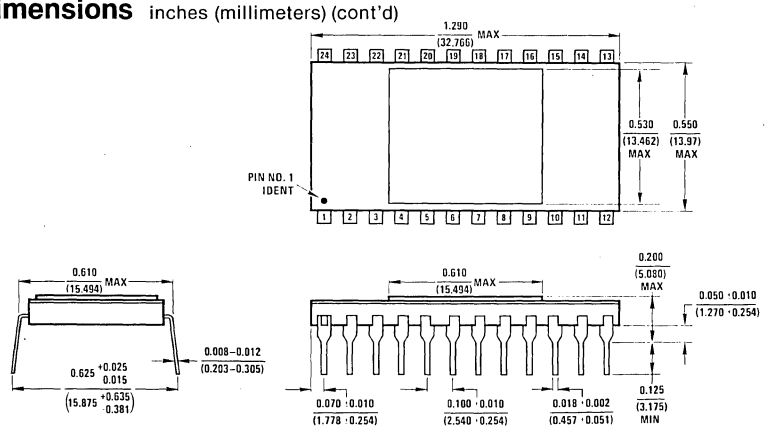



**Package 4**
**18-Lead Cavity DIP (D)**
**NS Package Number D18A**

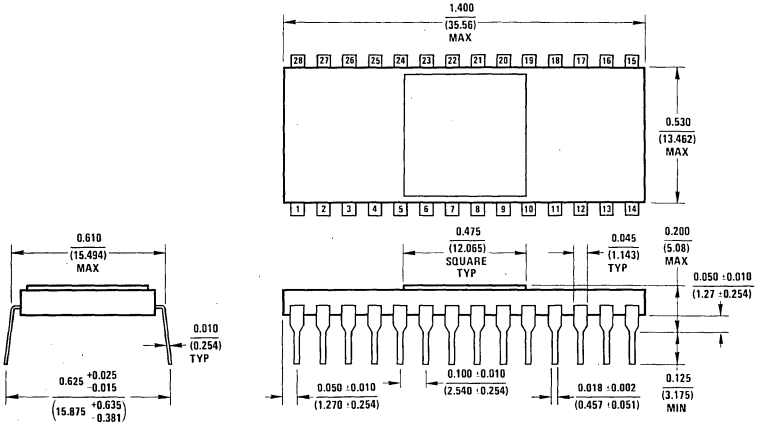



**Package 5**
**22-Lead Cavity DIP (D)**
**NS Package Number D22A**

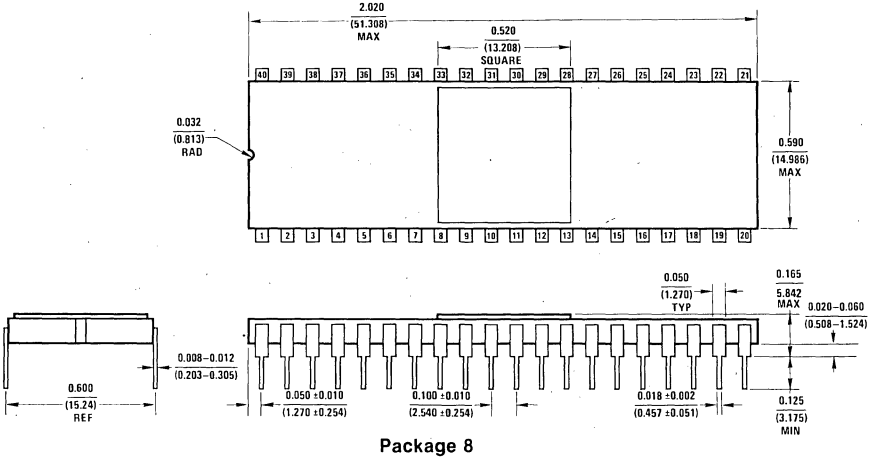
## Physical Dimensions inches (millimeters) (cont'd)




**Package 6**
**24-Lead Cavity DIP (D)**
**NS Package Number D24A**



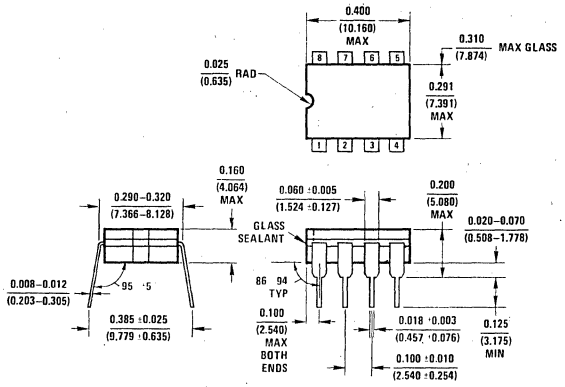

**Package 7**
**28-Lead Cavity DIP (D)**
**NS Package Number D28A**



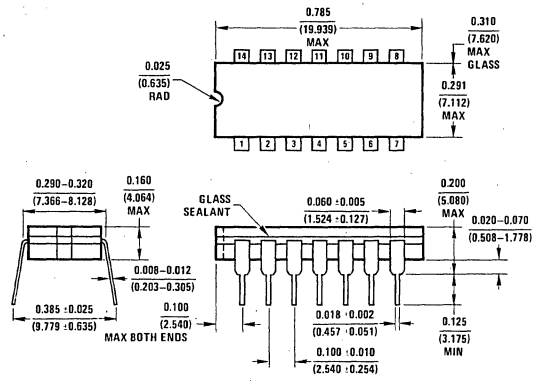

**Package 8**
**40-Lead Cavity DIP (D)**
**NS Package Number D40C**

8

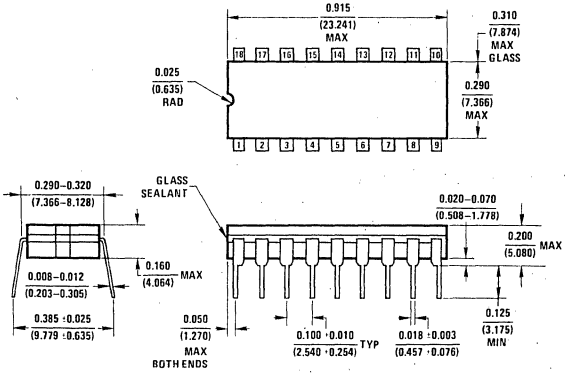
# Physical Dimensions inches (millimeters) (cont'd)



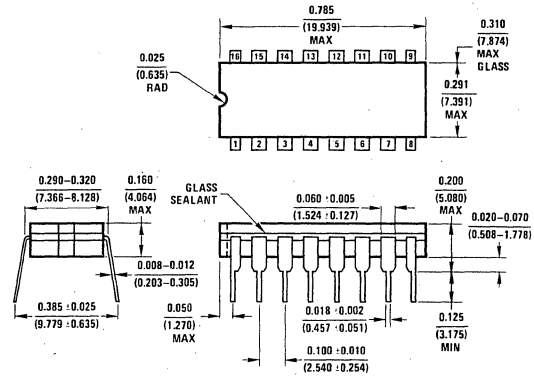

**Package 9**
**8-Lead Cavity DIP (J)**
**NS Package Number J08A**



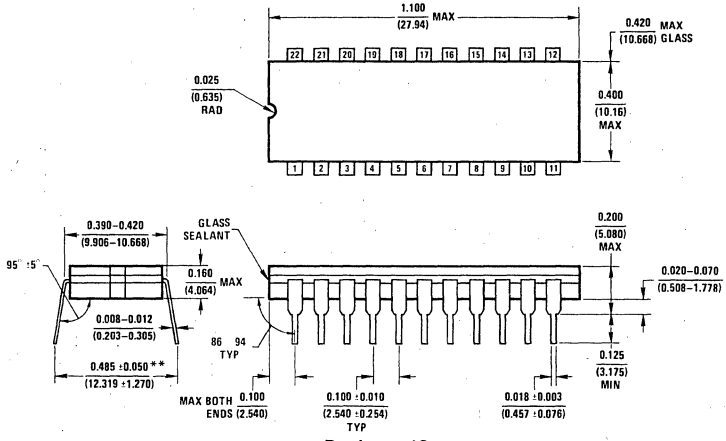

**Package 10**
**14-Lead Cavity DIP (J)**
**NS Package Number J14A**




**Package 11**
**16-Lead Cavity DIP (J)**
**NS Package Number J16A**

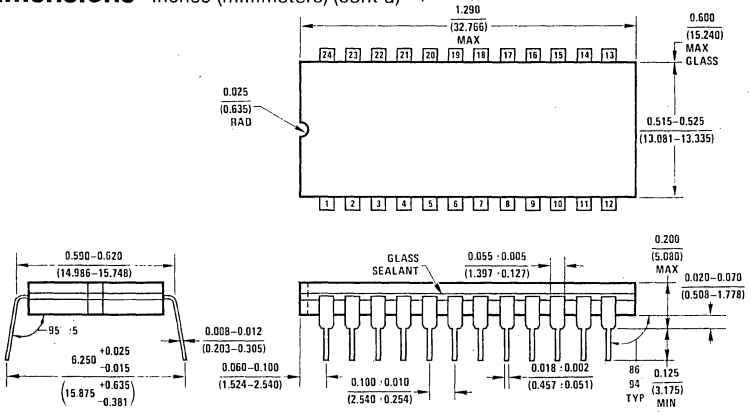



**Package 12**
**18-Lead Cavity DIP (J)**
**NS Package Number J18A**

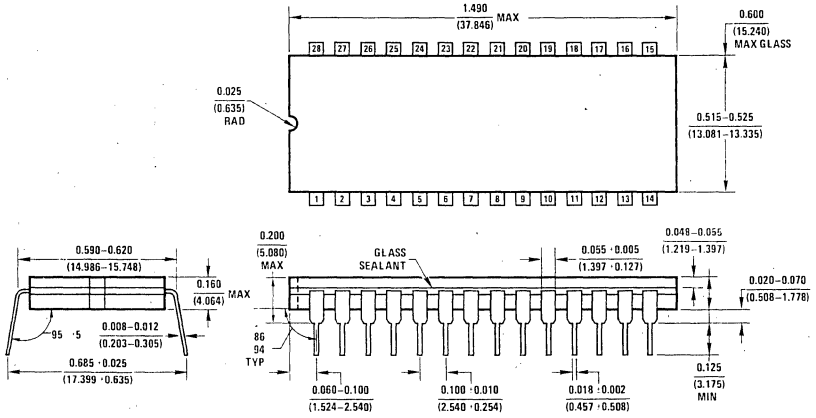



**Package 13**
**22-Lead Cavity DIP (J)**
**NS Package Number J22A**

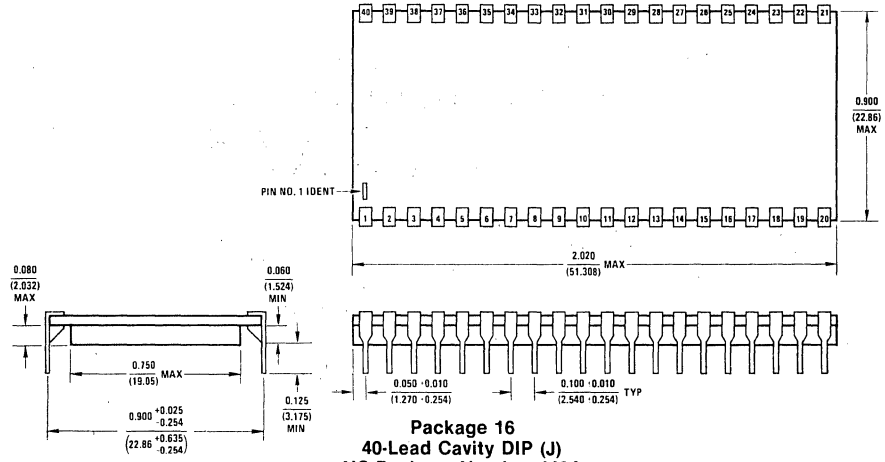
## Physical Dimensions inches (millimeters) (cont'd)




**Package 14**
**24-Lead Cavity DIP (J)**
**NS Package Number J24A**



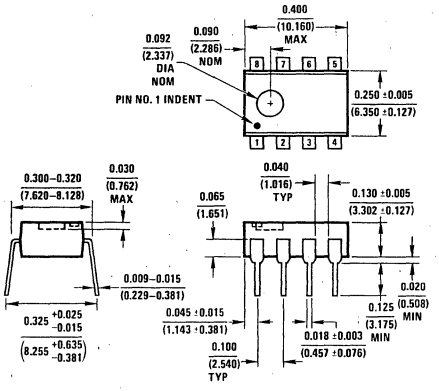

**Package 15**
**28-Lead Cavity DIP (J)**
**NS Package Number J28A**



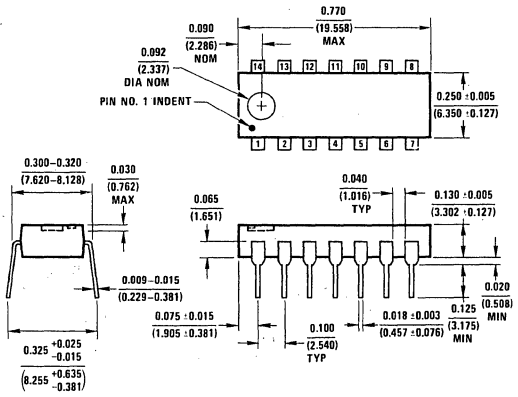

**Package 16**
**40-Lead Cavity DIP (J)**
**NS Package Number J40A**

8

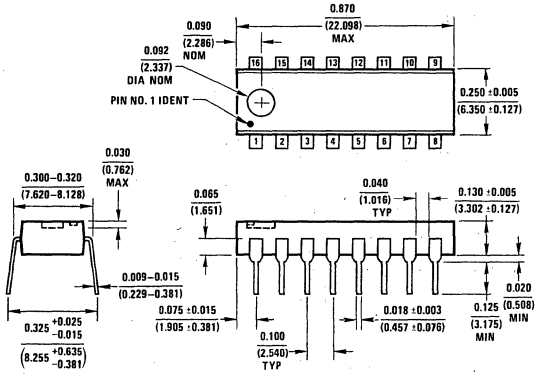
## Physical Dimensions inches (millimeters) (cont'd)



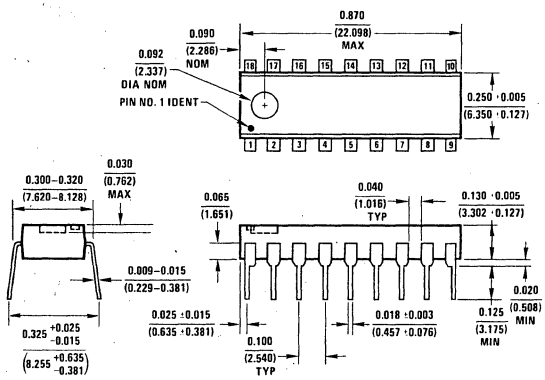

**Package 17**
**8-Lead Molded DIP (N)**
**NS Package Number N08A**



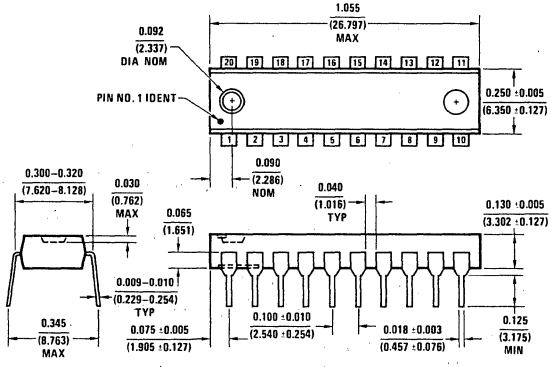

**Package 18**
**14-Lead DIP (N)**
**NS Package Number N14A**



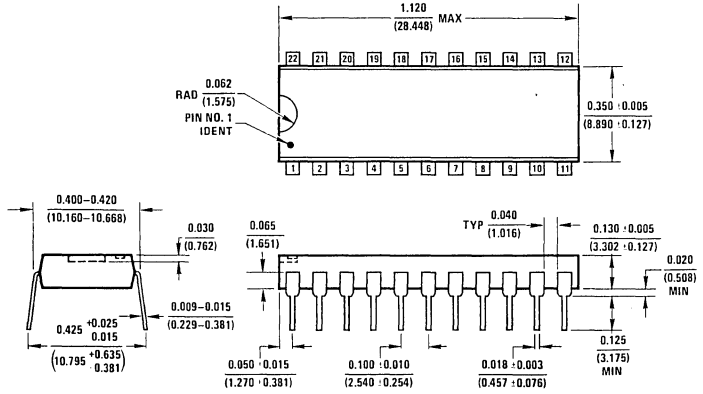

**Package 19**
**16-Lead Molded DIP (N)**
**NS Package Number N16A**



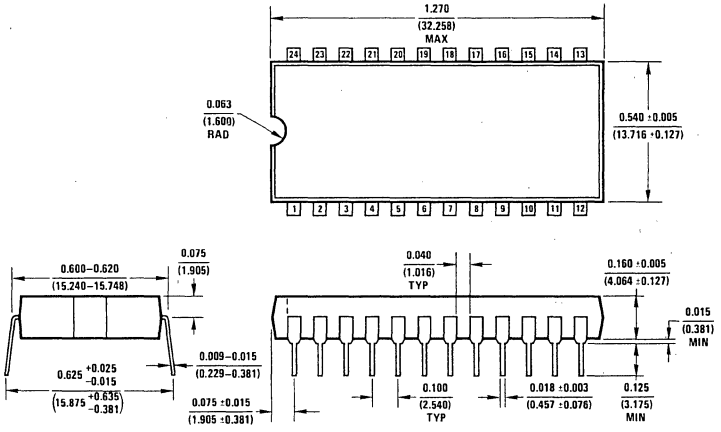

**Package 20**
**18-Lead Molded DIP (N)**
**NS Package Number N18A**



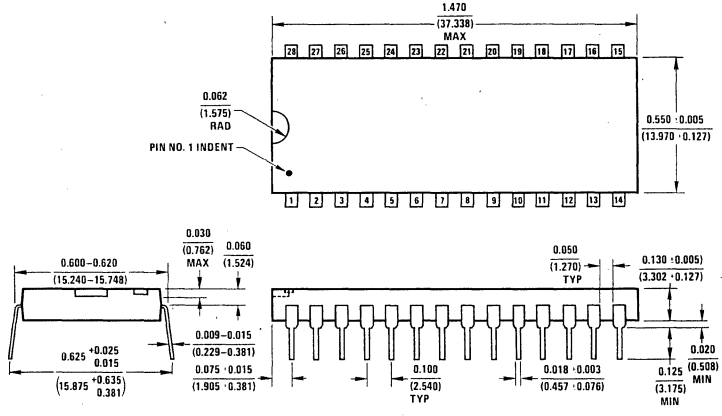

**Package 20A**
**20-Lead Molded DIP (N)**
**NS Package Number N20A**




**Package 21**
**22-Lead Molded DIP (N)**
**NS Package Number N22A**

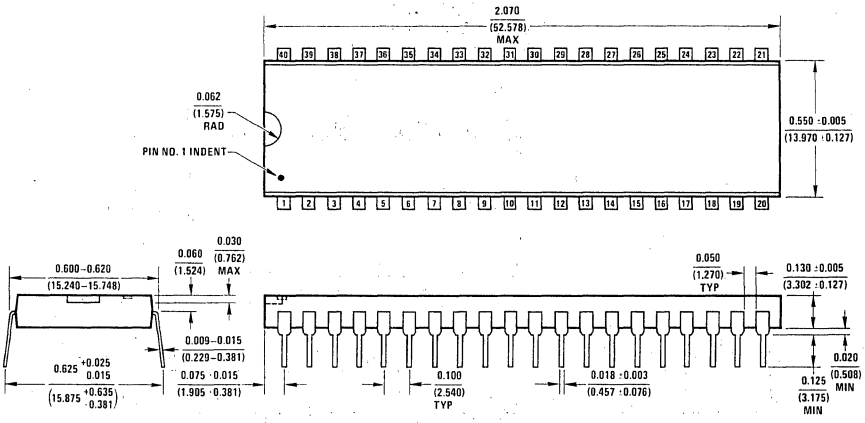



**Package 22**
**24-Lead Molded DIP (N)**
**NS Package Number N24A**




**Package 23**
**28-Lead Molded DIP (N)**
**NS Package Number N28A**

## Physical Dimensions inches (millimeters) (cont'd)




**Package 24**
**40-Lead Molded DIP (N)**
**NS Package Number N40A**


**National Semiconductor Corporation**
2900 Semiconductor Drive
Santa Clara, California 95051
Tel: (408) 737-5000
TWX: (910) 339-9240

**National Semiconductor**
District Sales Office
345 Wilson Avenue, Suite 404
Downsview, Ontario M3H 5W1
Canada
Tel: (416) 635-7260

**Mexicana de Electronica**
Industrial S.A.
Tlacoquemecatl No. 139-401
Esquina Adolfo Prieto
Mexico 12, D.F.
Tel: 575-78-68, 575-79-24

**NS Electronics Do Brasil**
Avda Brigadeiro Faria Lima 844
11 Andar Conjunto 1104
Jardim Paulistano
Sao Paulo, Brasil
Telex: 1121008 CABINE SAO PAULO

**National Semiconductor GmbH**
Eisenheimerstrasse 61/2
8000 München 21
West Germany
Tel: 089/9 15027
Telex: 522772

**National Semiconductor (UK) Ltd.**
301 Harpur Centre
Horne Lane
Bedford MK40 1TR
United Kingdom
Tel: 0234-47147
Telex: 826 209

**National Semiconductor Benelux**
789 Ave. Houbal de Strooper
1020 Bruxelles
Belgium
Tel: (02) 4783400
Telex: 61007

**National Semiconductor Ltd.**
Vodroffsvej 44
1900 Copenhagen V
Denmark
Tel: (01) 356533
Telex: 15179

**National Semiconductor**
Expansion 10000
28, Rue de la Redoute
92 260 Fontenay-aux-Roses
France
Tel: (01) 660-8140
Telex: 250956

**National Semiconductor S.p.A.**
Via Solferino 19
20121 Milano
Italy
Tel: (02) 345-2046/7/8/9
Telex: 332835

**National Semiconductor AB**
Box 2016
12702 Skarholmen
Sweden
Tel: (08) 970190
Telex: 10731

**National Semiconductor**
Calle Nunez Morgado 9
Esc. DCHA. 1-A
Madrid 16
Spain
Tel: (01) 215-8076/215-8434
Telex: 46642

**National Semiconductor Switzerland**
Alte Winterthurerstrasse 53
Postfach 567
CH-8304 Wallisellen-Zürich
Tel: (01) 830-2727
Telex: 59000

**NS International Inc., Japan**
47F Shinjuku Center Building
1-25-1 Nishishinjuku, Shinjuku-ku
Tokyo, Japan
Tel: (04) 355-5711
TWX: 232-2015 JSCJ-J

**National Semiconductor (Hong Kong) Ltd.**
8th Floor,
Cheung Kong Electronic Bldg.
4 Hing Yip Street
Kwun Tong
Kowloon, Hong Kong
Tel: 3-411241-8
Telex: 73866 NSEHK HX
Cable: NATSEMI

**NS Electronics Pty. Ltd.**
Cnr. Stud Rd. & Mtn. Highway
Bayswater, Victoria 3153
Australia
Tel: 03-729-6333
Telex: 32096

**National Semiconductor (Pty.) Ltd.**
10th Floor
Pub Building
Somerset Road
Singapore 0923
Tel: 7379338/7379339
Telex: NAT SEMI RS 21402

**National Semiconductor (Taiwan) Ltd.**
Rm. B, 3rd Floor
Ching Lin Bldg.
No. 9, Ching Tao E. Road
P.O. Box 68-332 or 39-1176 Taipei
Tel: 3917324-6
Telex: 22837 NSTW
Cable: NSTW TAIPEI

**National Semiconductor (Hong Kong) Ltd.**
Korea Liaison Office
6th Floor, Kunwon Bldg.
1-2 Mookjung-Dong
Choong-Ku, Seoul
C.P.O. Box 7941 Seoul
Tel: 267-9473
Telex: K24942


 DA-TM30RRD40M60/Printed in U.S.A.