MITSUBISHI SEMICONDUCTORS 1990

SINGLE-CHIP 16-BIT MICROCOMPUTERS

DATA BOOK

QCI
ASSOCIATES
2216 O'Toole Avenue
San Jose, CA 95131
Phone (408) 432-1070

MITSUBISHI ELECTRIC

All values shown in this catalogue are subject to change for product improvement.

The information, diagrams and all other data included herein are believed to be correct and reliable. However, no responsibility is assumed by Mitsubishi Electric Corporation for their use, nor for any infringements of patents or other rights belonging to third parties which may result from their use.

INDEX

5 DEVELOPMENT SUPPORT TOOLS FOR SERIES MELPS 7700

RASM77	MELPS 7700 Relocatable Macro Assembler	5—3
C77	MELPS 7700 C Compiler	5—5
RTT77	MELPS 7700 Real-Time Tracer	5—7
SDB77	MELPS 7700 Symbolic Debugger	5—8
MR7700KIT	MELPS 7700 Real-Time OS Development KIT	5—10
MR7700SRC	MELPS 7700 Real-Time OS Source KIT	5—12
PC4000E	Debugging Machine	5—13
PC4000P	Power Supply for PC4000E	5—15
PC4816	MELPS 7700 Debugging Machine	5—16
PCA4800M	1M Byte Memory Expansion Board for PC4816	5—20
M37700T-RTT	MELPS 7700 Common Option Board (Real-Time Tracer, Correspond to PC4000E)	5—21
M37700T-POD	MELPS 7700 Emulation Pod (Single-Chip Mode, Correspond to PC4000E)	5—22
M37700TX-POD	MELPS 7700 Emulation Pod (Microprocessor Mode, Correspond to PC4000E)	5—23
M37700T-HPD	MELPS 7700 High Level Emulation POD (Correspond to PC4816)	5—24
M37702T-HPD	MELPS 7700 High Level Emulation POD (Correspond to PC4816)	5—25
M37704T-HPD	MELPS 7700 High Level Emulation POD (Correspond to PC4816)	5—26
M37720T-HPD	MELPS 7700 High Level Emulation POD (Correspond to PC4816)	5—27
M37795T-HPD	MELPS 7700 High Level Emulation POD (Correspond to PC4816)	5—28
M37796T-HPD	MELPS 7700 High Level Emulation POD (Correspond to PC4816)	5—29
PCA4707	Program Writing Adapter for One time PROM Version Microcomputers	5—30
PCA4708	Program Writing Adapter for EPROM Version Microcomputers	5—31
PCA4709	Program Writing Adapter for PROM Version Microcomputers	5—32
PCA4774	Program Writing Adapter for One time PROM Version Microcomputers	5—33
PCA4780	Pitch Converter for 80-pin Flat Package	5—34
PCA4990	Program Writing Adapter for Versatile Memory	5—35
PCA7780G02	Pitch Converter for 80-pin Flat Package	5—36

6 APPENDICES

Series MELPS 7700 Mask ROM Ordering Method ..... 6—3
Series MELPS 7700 PROM Ordering Method ..... 6—16
Mark Specification Form ..... 6—31

Contact Addresses for Further Information

■SERIES MELPS 7700 16-BIT MICROCOMPUTERS (Continue)

Type	Circuit function and organization	Structure	Supply voltage (V)	Electrical characteristics: Typ power dissipation (mW)	Electrical characteristics: Min cycle time (ns)	Electrical characteristics: Max frequency (MHz)	Package	Page
M37705M2-XXXSP ★	16K-Byte Mask-Prog. ROM, 512-Byte RAM 16-Bit Timer, 8-Bit A-D Converter UART	C, Si	5±10%	30	500	8	64P4B	2—162
M37705M2AXXXSP ★	16K-Byte Mask-Prog. ROM, 512-Byte RAM 16-Bit Timer, 8-Bit A-D Converter UART	C, Si	5±10%	60	250	16	64P4B	
M37705S1SP ★	External ROM, 512-Byte RAM 16-Bit Timer, 8-Bit A-D Converter UART	C, Si	5±10%	30	500	8	64P4B	
M37705S1ASP ★	External ROM, 512-Byte RAM 16-Bit Timer, 8-Bit A-D Converter UART	C, Si	5±10%	60	250	16	64P4B	
M37795SJ ★	External ROM, RAM 16-Bit Timer, 10-Bit A-D Converter Three PWM, Serial I/O	C, Si	5±10%	50	500	8	84P0	2—184
M37795STJ ★	External ROM, RAM 16-Bit Timer, 10-Bit A-D Converter Three PWM, Serial I/O	C, Si	5±10%	50	500	8	84P0	

★ : New product

DEVELOPMENT SUPPORT SYSTEMS

Development support systems for series MELPS 7700 (1)

MELPS7700 type name	Assembler	Compiler	Debug system							For evaluate
			PC4000E base				PC4816 base			
			Debugger	Emulation common board	Emulation pod	Control software	Debugger	Emulation pod	Control software	
M37700M2-XXXFP M37700M2AXXXFP M37700M4-XXXFP M37700M4AXXXFP M37700SFP M37700SAFP M37700S4FP M37700S4AFP M37700E2-XXXFP M37700E2AXXXFP M37700E2FS M37700E2AFS M37700E4-XXXFP M37700E4AXXXFP M37700E4FS M37700E4AFS	RASM77	C77	PC4000E	M37700T-RTT	M37700T-POD (Corresponds to single-chip mode)	RTT77	PC4816★	M37700T-★ HPD	SDB77★	M37700E2FS M37700E2AFS M37700E4FS M37700E4AFS
M37701M2-XXXSP M37701M2AXXXSP M37701SSP M37701SASP M37701M4-XXXSP M37701M4AXXXSP M37701S4SP M37701S4ASP M37701E2-XXXSP M37701E2AXXXSP M37701E4-XXXSP M37701E4AXXXSP					M37700TX-POD (Corresponds to microprocessor mode)					M37701E2-XXXSP M37701E2AXXXSP M37701E4-XXXSP M37701E4AXXXSP

Development supports systems for series MELPS 7700 (2)

MELPS7700 type name	Assembler	Compiler	Debug system							For evaluate
			PC4000E base				PC4816 base			
			Debugger	Emulation common board	Emulation pod	Control software	Debugger	Emulation pod	Control software	
M37704M2-XXXFP M37704M2AXXXFP M37704S1FP M37704S1AFP M37704E2-XXXFP M37704E2AXXXFP M37704E2FS M37704E2AFS	RASM77	C77	PC4000E	M37700T-RTT	M37700T-POD (Corresponds to single-chip mode)	RTT77	PC4816★	M37704T-★★ HPD	SDB77★	M37704E2FS M37704E2AFS
M37705M2-XXXSP M37705M2AXXXSP M37705S1SP M37705S1ASP M37705E2-XXXSP M37705E2AXXXSP M37705E2SS M37705E2ASS					M37700TX-POD (Corresponds to microprocessor mode)					M37705E2SS M37705E2ASS
M37795SJ M37795STJ			—	—	—	—		M37795T-★★ HPD		—
M37796E4-XXXJ M37796E4TXXXJ								M37796T-★★ HPD		

★ : New products ★★ : Under development

ORDERING INFORMATION

FUNCTION CODE

Mitsubishi integrated circuit may be ordered using the following simplified alphanumeric type-codes which define the function of the IC/LSIs and the package style.

For Mitsubishi Original Products

Example : **M 3 77 00 E 4 - 001 FP**

- **M** : Mitsubishi integrated prefix
- **3** : Represent an original single-chip microcomputer
- Series designation using 2 digits
- Circuit function identification code using 2 digits
- Memory identification code using a digit
  - **E** : PROM
  - **P** : Piggyback
  - **M** : Mask ROM
  - **S** : External ROM
  - **N** : Mask ROM+EEPROM
- Memory size identification code using a digit
- Normally, using hyphen.
  When electrical characteristic, or division of quality identification code using alphanumeric character.
  T : For automobile, industrial equipment
- Mask ROM number
- Package style
  - **J** : PLCC, or SOJ package
  - **FP** : Molded plastic flat package
  - **FS** : Ceramic flat package
  - **SP** : Molded plastic shrink package
  - **SS** : Ceramic shrink package

PACKAGE CODE

Package style may be specified by using the following simplified alphanumeric code.

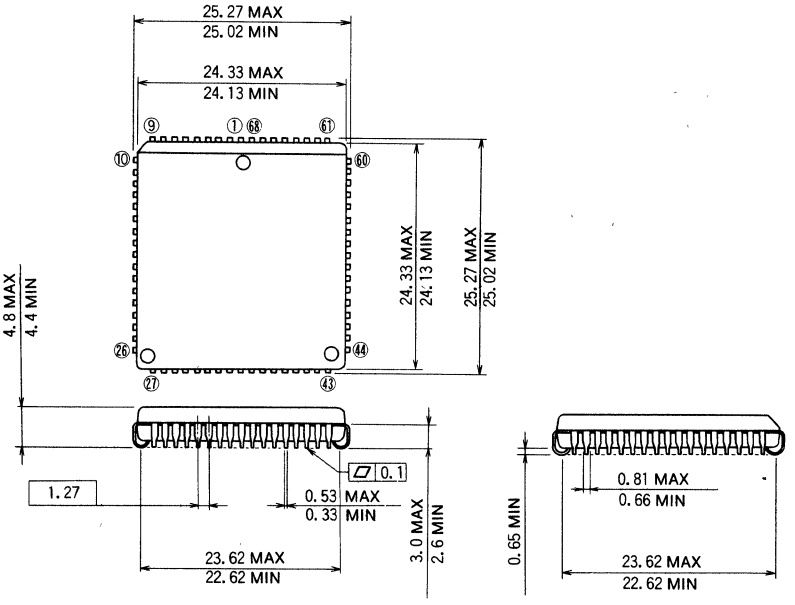
Example : **42 P 4 B**

- Number of pins
- Package structure
  - **K** : Glass-sealed ceramic
  - **P** : Molded plastic
  - **S** : Metal-sealed ceramic
- Package outline
  - **0** : PLCC, SOJ
  - **1** : DIP
  - **2** : SOP
  - **4** : DIP
  - **6** : QFP
- Secondary outline code

  Special-purpose secondary codes describing outline are included as necessary. For details, contact your sales representative.

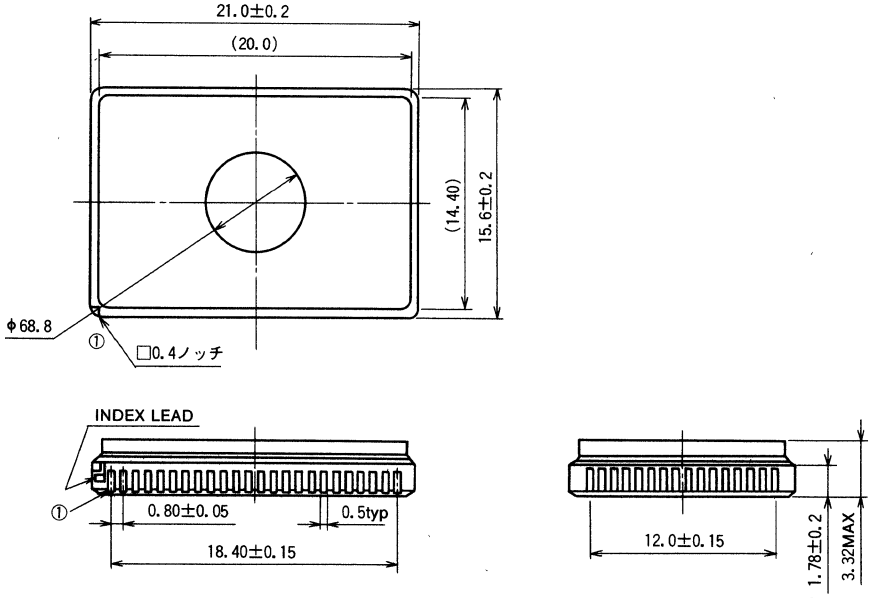
TYPE 68P0 68-PIN MOLDED PLASTIC LEADED CHIP CARRIER

Dimension in mm



TYPE 80D0 80PIN CERAMIC LCC

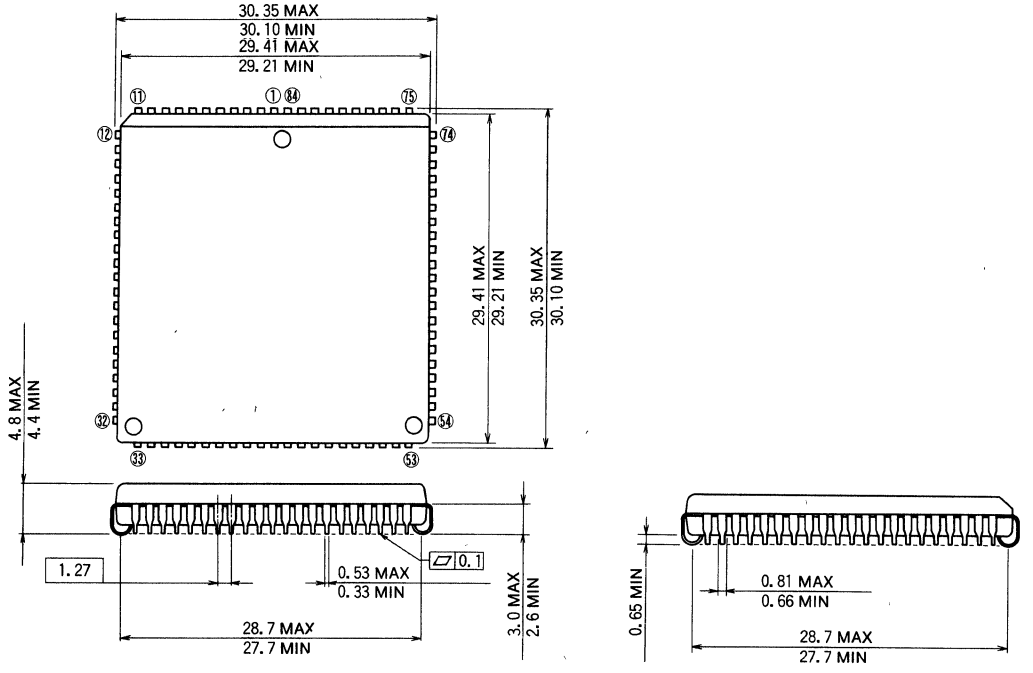
Dimension in mm



MITSUBISHI ELECTRIC

TYPE 84P0 84-PIN MOLDED PLASTIC LEADED CHIP CARRIER

Dimension in mm



MITSUBISHI ELECTRIC

LETTER SYMBOLS FOR THE DYNAMIC PARAMETERS

b) those that are characteristics of the memory.
The letter symbols so far proposed for memory circuits are listed in sub-clauses 3.1 and 3.2 below.
All subscripts A should be in lower-case.

3.1. Timing Requirements

The letter symbols for the timing requirements of semiconductor memories are as follows:

Term	Subscript
Cycle time	c
Time interval between two signal events	d
Fall time	f
Hold time	h
Precharging time	pc
Rise time	r
Recovery time	rec
Refresh time interval	rf
Setup time	su
Transition time	t
Pulse duration (width)	w

3.2. Characteristics

The letter symbols for the dynamic characteristics of semiconductor memories are as follows :

Characteristic	Subscript
Access time	a
Disable time	dis
Enable time	en
Propagation time	P
Recovery time	rec
Transition time	T
Valid time	v

Note Recovery time for use as a characteristic is limited to sense recovery time

4. SUBSCRIPTS B AND D (For Signal Name or Terminal Name)

The letter symbols for the signal name or the name of the terminal are as given below.
All subscripts B and D should be in upper-case.

Signal or terminal	Subscript
Address	A
Clock	C
Column address	CA
Column address strobe	CAS
Data input	D
Data input/output	DQ
Chip enable	E
Erasure	ER
Output enable	G
Program	PR
Data output	Q
Read	R
Row address	RA
Row address strobe	RAS
Refresh	RF
Read/Write	RW
Chip select	S
Write (write enable)	W

Note 1 In the letter symbols for time intervals, bars over the subscripts, for example CAS, should not be used
2 It should be noted, when further letter symbols are chosen, that the subscript should not end with H, K, V, X, or Z (See clause 5)
3 If the same terminal, or signal, can be used for two functions (for example Data input/output, Read/Write) the waveform should be labelled with the dual function, if appropriate, but the symbols for the dynamic parameters should include only that part of the subscript relevant to the parameter

5. SUBSCRIPTS C AND E (For Transition of Signal)

The following symbols are used to represent the level or state of a signal :

Transition of signal	Subscript
High logic level	H
Low logic level	L
Valid steady-state level (either low or high)	V
Unknown, changing, or 'don't care' level	X
High-impedance state of three-state output	Z

The direction of transition is expressed by two letters, the direction being from the state represented by the first letter to that represented by the second letter, with the letters being as given above.
When no misunderstanding can occur, the first letter may be omitted to give an abbreviated symbol for subscripts C and E as indicated below.
All subscripts C and E should be in upper-case.

Examples	Subscript	
	Full	Abbreviated
Transition from high level to low level	HL	L
Transition from low level to high level	LH	H
Transition from unknown or changing state to valid state	XV	V
Transition from valid state to unknown or changing state	VX	X
Transition from high-impedance state to valid state	ZV	V

Note Since subscripts C and E may be abbreviated, and since subscripts B and D may contain an indeterminate number of letters, it is necessary to put the restriction on the subscripts B and D that they should not end with H, L, V, X, or Z, so as to avoid possible confusion

FOR DIGITAL INTEGRATED CIRCUITS

New symbol	Former symbol	Parameter–definition
C_I		Input capacitance
C_O		Output capacitance
$C_{I/O}$		Input/output terminal capacitance
$C_{I(\phi)}$		Input capacitance of clock input
f		Frequency
$f_{(\phi)}$		Clock frequency
I		Current–the current into an integrated circuit terminal is defined as a positive value and the current out of a terminal is defined as a negative value
I_{BB}		Supply current from V_{BB}
$I_{BB(AV)}$		Average supply current from V_{BB}
I_{CC}		Supply current from Vcc
$I_{CC(AV)}$		Avarage supply current from Vcc
$I_{CC(PD)}$		Power down supply current from Vcc
I_{DD}		Supply current from V_{DD}
$I_{DD(AV)}$		Average supply current from V_{DD}
I_{GG}		Supply current from V_{GG}
$I_{GG(AV)}$		Average supply current from V_{GG}
I_I		Input current
I_{IH}		High level input current–the value of the input current when V_{OH} is applied to the input considered
I_{IL}		Low level input current–the value of the input current when V_{OL} is applied to the input considered
I_{OH}		High level output current–the value of the output current when V_{OH} is applied to the output considered
I_{OL}		Low level output current–the value of the output current when V_{OL} is applied to the output considered
I_{OZ}		Off state (high impedance state) output current–the current into an output having a three-state capability with input condition so applied that it will establish according to the product specification, the off (high-impedance) state at the output
I_{OZH}		Off state (high impedance state) output current, with high level voltage applied to the output
I_{OZL}		Off-state (high impedance state) output current, with low-level voltage applied to the output
I_{OS}		Short-circuit output current
I_{SS}		Supply current from V_{SS}
Pd		Power dissipation
N_{EW}		Number of erase/write cycles
N_{RA}		Number of read access unrefreshed
R_i		Input resistance
R_L		External load resistance
R_{OFF}		Off state output resistance
R_{ON}		On-state output resistance
t_a		Access time–the time interval between the application of a specified input pulse during a read cycle and the availability of valid data signal at an output
$t_{a(A)}$	$t_{a(AD)}$	Address access time–the time interval between the application of an address input pulse and the availability of valid data signals at an output
$t_{a(CAS)}$		Column address strobe access time
$t_{a(E)}$	$t_{a(CE)}$	Chip enable access time
$t_{a(G)}$	$t_{a(OE)}$	Output enable access time
$t_{a(PR)}$		Data access time after program
$t_{a(RAS)}$		Row address strobe access time
$t_{a(S)}$	$t_{a(CS)}$	Chip select access time
t_C		Cycle time
t_{CR}	$t_{C(RD)}$	Read cycle time–the time interval between the start of a read cylce and the start of the next cycle
t_{CRF}	$t_{C(REF)}$	Refresh cycle time–the time interval between successive signals that are intended to restore the level in a dynamic memory cell to its original level
t_{CPG}	$t_{C(PG)}$	Page-mode cycle time
t_{CRMW}	$t_{C(RMR)}$	Read-modify-write cycle time–the time interval between the start of a cycle in which the memory is read and new data is entered, and the start of the next cycle
t_{CW}	$t_{C(WR)}$	Write cycle time–the time interval between the start of a write cycle and the start of the next cycle

New symbol	Former symbol	Parameter—definition
$t_{su(D)}$	$t_{su(DA)}$	Data-in setup time
$t_{su(D-E)}$	$t_{su(DA-CE)}$	Chip enable setup time before data-in
$t_{su(D-W)}$	$t_{su(DA-WR)}$	Write setup time before data-in
$t_{su(E)}$	$t_{su(CE)}$	Chip enable setup time
$t_{su(E-P)}$	$t_{su(CE-P)}$	Precharge setup time before chip enable
$t_{su(G-E)}$	$t_{su(OE-CE)}$	Chip enable setup time before output enable
$t_{su(P-E)}$	$t_{su(P-CE)}$	Chip enable setup time before precharge
$t_{su(PD)}$		Power-down setup time
$t_{su(R)}$	$t_{su(RD)}$	Read setup time
$t_{su(R-CAS)}$	$t_{su(RA-CAS)}$	Column address strobe setup time before read
$t_{su(RA-CAS)}$		Column address strobe setup time before row address
$t_{su(S)}$	$t_{su(CS)}$	Chip select setup time
$t_{su(S-W)}$	$t_{su(CS-WR)}$	Write setup time before chip select
$t_{su(W)}$	$t_{su(WR)}$	Write setup time
t_{THL}		High-level to low-level transition time } the time interval between specified reference points on the edge of the output pulse when the output is going to the low (high) level and when a specified input signal is applied through a specified network and the output is loaded by another specified network
t_{TLH}		Low-level to high level transition time }
$t_{v(A)}$	$t_{dv(AD)}$	Data valid time after address
$t_{v(E)}$	$t_{dv(CE)}$	Data valid time after chip enable
$t_{v(E)PR}$	$t_{v(CE)PR}$	Data valid time after chip enable in program mode
$t_{v(G)}$	$t_{v(OE)}$	Data valid time after output enable
$t_{v(PR)}$		Data valid time after program
$t_{v(S)}$	$t_{v(CS)}$	Data valid time after chip select
t_W		Pulse width (pulse duration) the time interval between specified reference points on the leading and training edges of the waveforms
$t_{w(E)}$	$t_{w(CE)}$	Chip enable pulse width
$t_{w(EH)}$	$t_{w(CEH)}$	Chip enable high pulse width
$t_{w(EL)}$	$t_{w(EL)}$	Chip enable low pulse width
$t_{w(PR)}$		Program pulse width
$t_{w(R)}$	$t_{w(RD)}$	Read pulse width
$t_{w(S)}$	$t_{w(CS)}$	Chip select pulse width
$t_{w(W)}$	$t_{w(WR)}$	Wrtie pulse width
$t_{w(\phi)}$		Clock pulse width
Ta		Ambient temperature
Topr		Operating temperature
Tstg		Storage temperature
V_{BB}		V_{BB} supply voltage
V_{CC}		V_{CC} supply voltage
V_{DD}		V_{DD} supply voltage
V_{GG}		V_{GG} supply voltage
V_I		Input voltage
V_{IH}		High-level input voltage—the value of the permitted high-state voltage at the input
V_{IL}		Low-level input voltage—the value of the permitted low-state voltage at the input
V_O		Output voltage
V_{OH}		High-level output voltage—the value of the guaranteed high-state voltage range at the output
V_{OL}		Low-level output voltage—the value of the guaranteed low-state voltage range at the output
V_{SS}		V_{SS} supply voltage

QUALITY ASSURANCE AND RELIABILITY TESTING

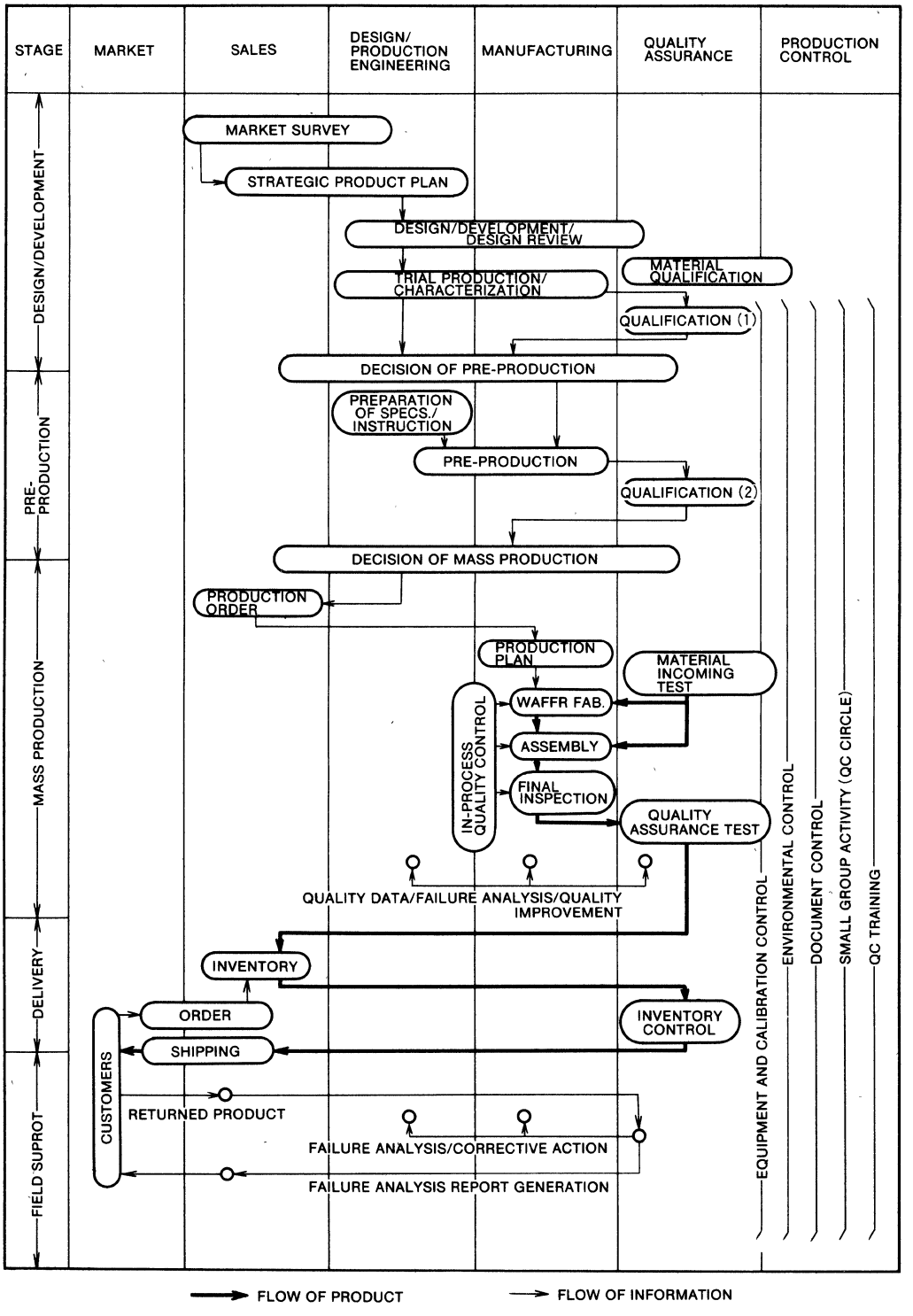


Fig.1 FLOW CHART OF QUALITY ASSURANCE SYSTEM

3 RELIABILITY TEST RESULTS

The reliability test results for Mitsubishi Single-chip 16-bit Microcomputers are shown in Table 2.
Table 2 shows the result of endurance tests of high temperature operation life and high temperature storage life test and the results of the environment tests of thermal stress, high temperature/high humidity and pressure cooker test for the single-chip 16-bit Microcomputer.

Table 2 ENDURANCE and ENVIRONMENTAL TEST RESULTS

Test	Series	Type Number	Test Condition	Number of Samples	Device Hours (Hours)	Number of Failures
High Temperature Operation Life	MELPS 7700	M37700E4-XXXFP M37700M2-XXXFP M37700M4-XXXFP	125℃ 7 V	266	266000	0
High Temperature Storage Life	MELPS 7700	M37700E4-XXXFP M37700M2-XXXFP M37700M4-XXXFP	150℃	228	228000	0
Low Temperature Storage Life	MELPS 7700	M37700E4-XXXFP M37700M2-XXXFP M37700M4-XXXFP	−55℃	66	66000	0
High Temperature High Humidity Life	MELPS 7700	M37700E4-XXXFP M37700M2-XXXFP	85℃85%RH 5.5V	228	228000	0
		M37700M4-XXXFP	85℃85%RH	22	22000	0

Test	Series	Type Number	Test Condition	Number of Samples	Number of Failures		
					96Hours	240Hours	500Hours
Pressure Cooker	MELPS 7700	M37700E4-XXXFP M37700M2-XXXFP M37700M4-XXXFP	121℃ 100%	176	0	0	0

Test	Series	Type Number	Test Condition	Number of Samples	Number of Failures		
					10Cycles	100Cycles	500Cycles
Temperature Cycling	MELPS 7700	M37700M2-XXXFP M37700M4-XXXFP	−65℃ 30min 150℃ 30min	76	0	0	0

(3) Destructive Failure by Electrical Overstress

Surge voltage marginal tests have been performed to reproduce the electrical overstress failure in field uses. Figure 9 and Figure 10 are examples of failure observed by surge voltage test. The trace of destruction is verified as the aluminum bridge by X-ray micro analysis.

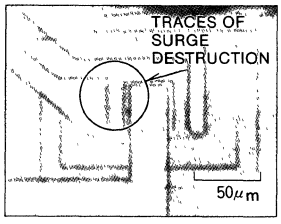


Fig.9 Micrograph of surge voltage destruction

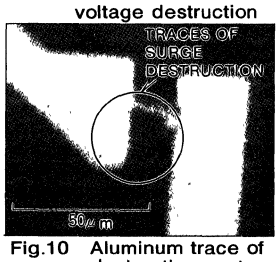


Fig.10 Aluminum trace of destructive spot

(4) Aluminum Electromigration

Figure 11 shows an open circuit of aluminum metallization in high current density region caused by accelerated operation life test. This failure is due to aluminum electromigration. Voids and hillock have been formed in aluminum metallization by high current density operation.

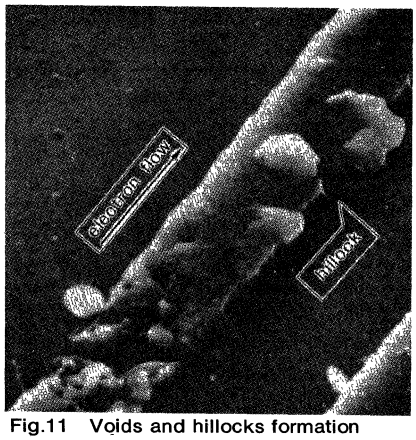


Fig.11 Voids and hillocks formation by Aluminum electromigration

5 SUMMARY

The Mitsubishi quality assurance system and examples of reliability control have been discussed. The customer's interests and requirements for high reliability IC & LSI are increasing significantly. To satisfy customer's expectancy. Mitsubishi as an IC vendor, would like to make perpetual efforts in the following areas.

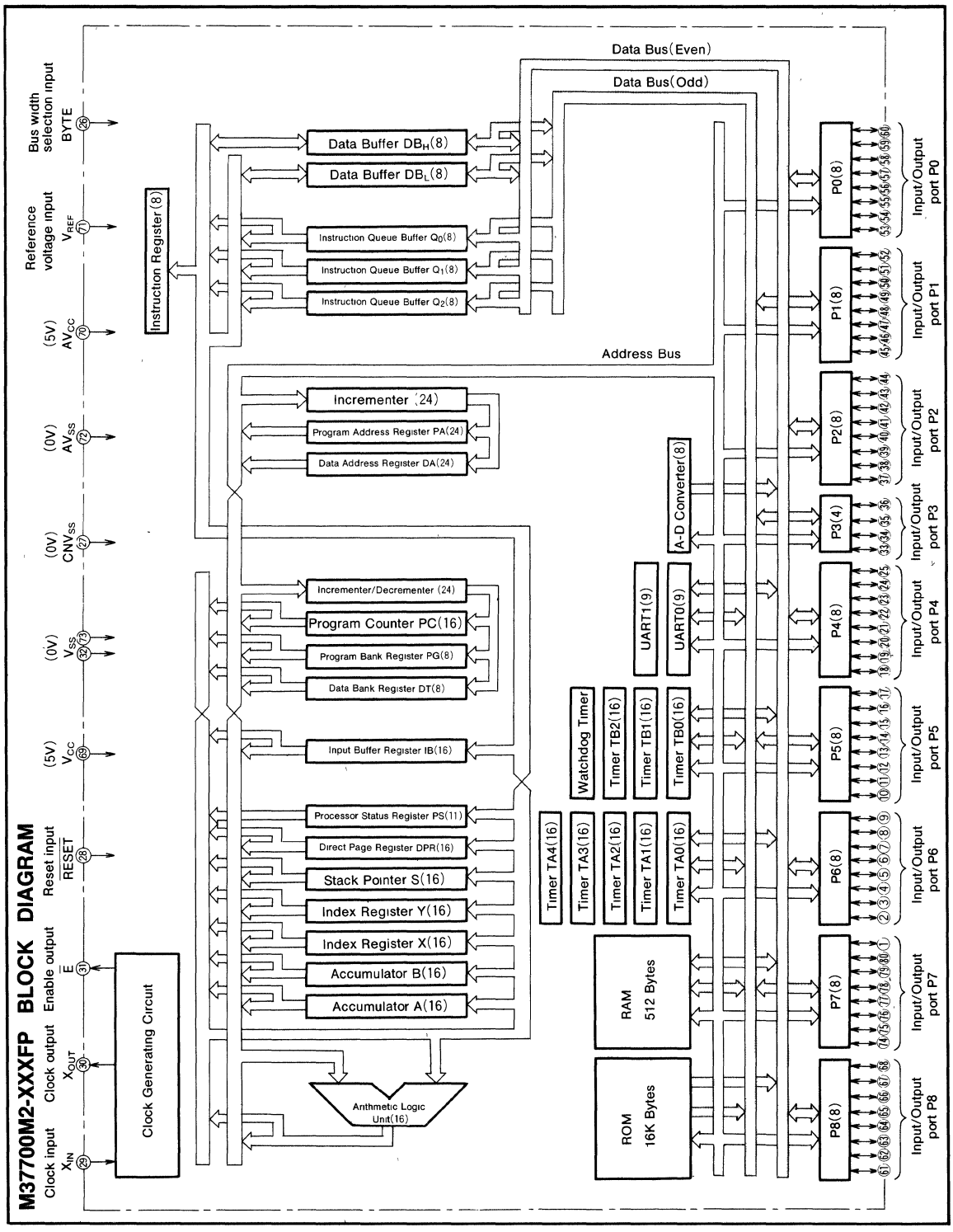
(1) Emphasis on built-in reliability at design stage and reliability evaluation to investigate latent failure modes and acceleration factors.

(2) Execution of periodical endurance, environment and mechanical test to verify reliability target and realize higher reliability.

(3) Focus on development of advanced failure analysis techniques. Detail failure analysis, intensive corrective action and quick response to customer's analysis request.

(4) Collection of customer's quality data in qualification, incoming inspection, production and field use to improve PPM, fraction defective and FIT, failure rate.

Mitsubishi would highly appreciate if the customer would provide quality and reliability data of incoming inspection or field failure rate essential to verify and improve the quality/reliability of IC & LSI.


M37700M2-XXXFP BLOCK DIAGRAM
Clock input
Clock output
Enable output
Reset input
Bus width selection input
Reference voltage input
Data Bus(Even)
Data Bus(Odd)
Address Bus
Instruction Register(8)
Data Buffer DBH(8)
Data Buffer DBL(8)
Instruction Queue Buffer Q0(8)
Instruction Queue Buffer Q1(8)
Instruction Queue Buffer Q2(8)
Incrementer (24)
Program Address Register PA(24)
Data Address Register DA(24)
Incrementer/Decrementer (24)
Program Counter PC(16)
Program Bank Register PG(8)
Data Bank Register DT(8)
Input Buffer Register IB(16)
Processor Status Register PS(11)
Direct Page Register DPR(16)
Stack Pointer S(16)
Index Register Y(16)
Index Register X(16)
Accumulator B(16)
Accumulator A(16)
Arithmetic Logic Unit(16)
Clock Generating Circuit
ROM 16K Bytes
RAM 512 Bytes
Timer TA4(16)
Timer TA3(16)
Timer TA2(16)
Timer TA1(16)
Timer TA0(16)
Watchdog Timer
Timer TB2(16)
Timer TB1(16)
Timer TB0(16)
UART1(9)
UART0(9)
A-D Converter(8)
P0(8)
P1(8)
P2(8)
P3(4)
P4(8)
P5(8)
P6(8)
P7(8)
P8(8)
Input/Output port P0
Input/Output port P1
Input/Output port P2
Input/Output port P3
Input/Output port P4
Input/Output port P5
Input/Output port P6
Input/Output port P7
Input/Output port P8

PIN DESCRIPTION

Pin	Name	Input/Output	Functions
V_{CC}, V_{SS}	Power supply		Supply 5 V±10% to V_{CC} and 0 V to V_{SS}.
CNV_{SS}	CNV_{SS} input	Input	This pin controls the processor mode. Connect to V_{SS} for single-chip mode, and to V_{CC} for external ROM types.
$\overline{RESET}$	Reset input	Input	To enter the reset state, this pin must be kept at a "L" condition should be maintained for the required time.
X_{IN}	Clock input	Input	These are I/O pins of internal clock generating circuit Connect a ceramic or quartz crystal resonator between X_{IN} and X_{OUT} When an external clock is used, the clock source should be connected to the X_{IN} pin and the X_{OUT} pin should be left open.
X_{OUT}	Clock output	Output	
$\overline{E}$	Enable output	Output	Data or instruction read and data write are performed when output from this pin is "L"
BYTE	Bus width selection input	Input	In memory expansion mode or microprocessor mode, this pin determines whether the external data bus is 8-bit width or 16-bit width The width is 16 bits when "L" signal inputs and 8 bits when "H" signal inputs
AV_{CC}, AV_{SS}	Analog supply input		Power supply for the A-D converter. Connect AV_{CC} to V_{CC} and AV_{SS} to V_{SS} externally.
V_{REF}	Reference voltage input	Input	This is reference voltage input pin for the A-D converter
$P0_0$~$P0_7$	I/O port P0	I/O	In single-chip mode, port P0 becomes an 8-bit I/O port. An I/O directional register is available so that each pin can be programmed for input or output These ports are in input mode when reset Address(A_7~A_0)is output in memory expansion mode or microprocessor mode
$P1_0$~$P1_7$	I/O port P1	I/O	In single-chip mode, these pins have the same functions as port P0. When the BYTE pin is set to "L" in memory expansion mode or microprocessor mode and external data bus is 16-bit width, high-order data (D_{15}~D_8)is input or output when $\overline{E}$ output is "L" and an address (A_{15}~A_8) is output when $\overline{E}$ output is "H" If the BYTE pin is "H" that is an external data bus is 8-bit width, only address(A_{15}~A_8)is output
$P2_0$~$P2_7$	I/O port P2	I/O	In single-chip mode, these pins have the same functions as port P0. In memory expansion mode or microprocessor mode low-order data(D_7~D_0) is input or output when $\overline{E}$ output is "L" and an address(A_{23}~A_{16})is output when $\overline{E}$ output is "H".
$P3_0$~$P3_3$	I/O port P3	I/O	In single-chip mode , these pins have the same functions as port P0 In memory expansion mode or microprocessor mode, R/$\overline{W}$, $\overline{BHE}$, ALE, and $\overline{HLDA}$ signals are output
$P4_0$~$P4_7$	I/O port P4	I/O	In single-chip mode, these pins have the same functions as port P0. In memory expansion mode or microprocessor mode, $P4_0$ and $P4_1$ become $\overline{HOLD}$ and $\overline{RDY}$ input pin respectively Functions of other pins are the same as in single-chip mode. Port $P4_2$ can be programed for ϕ_1 output pin divided the clock to X_{IN} pin by 2
$P5_0$~$P5_7$	I/O port P5	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as I/O pins for timer A0, timer A1, timer A2, and timer A3.
$P6_0$~$P6_7$	I/O port P6	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as I/O pins for timer A4, external interrupt input $\overline{INT_0}$, $\overline{INT_1}$, and $\overline{INT_2}$ pins, and input pins for timer B0, timer B1, and timer B2.
$P7_0$~$P7_7$	I/O port P7	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as analog input AN_0~AN_7 input pins. $P7_7$ also has an A-D conversion trigger input function.
$P8_0$~$P8_7$	I/O port P8	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as R_XD, T_XD, CLK, $\overline{CTS}$/$\overline{RTS}$ pins for UART 0 and UART 1.

Address (Hexadecimal notation)	
000000	
000001	
000002	Port P0
000003	Port P1
000004	Port P0 data direction register
000005	Port P1 data direction register
000006	Port P2
000007	Port P3
000008	Port P2 data direction register
000009	Port P3 data direction register
00000A	Port P4
00000B	Port P5
00000C	Port P4 data direction register
00000D	Port P5 data direction register
00000E	Port P6
00000F	Port P7
000010	Port P6 data direction register
000011	Port P7 data direction register
000012	Port P8
000013	
000014	Port P8 data direction register
000015	
000016	
000017	
000018	
000019	
00001A	
00001B	
00001C	
00001D	
00001E	A-D control register
00001F	
000020	A-D register 0
000021	
000022	A-D register 1
000023	
000024	A-D register 2
000025	
000026	A-D register 3
000027	
000028	A-D register 4
000029	
00002A	A-D register 5
00002B	
00002C	A-D register 6
00002D	
00002E	A-D register 7
00002F	
000030	UART 0 transmit/receive mode register
000031	UART 0 bit rate generator
000032	UART 0 transmission buffer register
000033	
000034	UART 0 transmit/receive control register 0
000035	UART 0 transmit/receive control register 1
000036	UART 0 receive buffer register
000037	
000038	UART 1 transmit/receive mode register
000039	UART 1 bit rate generator
00003A	UART 1 transmission buffer register
00003B	
00003C	UART 1 transmit/receive control register 0
00003D	UART 1 transmit/receive control register 1
00003E	UART 1 receive buffer register
00003F	

Address (Hexadecimal notation)	
000040	Count start flag
000041	
000042	One-shot start flag
000043	
000044	Up-down flag
000045	
000046	Timer A0
000047	
000048	Timer A1
000049	
00004A	Timer A2
00004B	
00004C	Timer A3
00004D	
00004E	Timer A4
00004F	
000050	Timer B0
000051	
000052	Timer B1
000053	
000054	Timer B2
000055	
000056	Timer A0 mode register
000057	Timer A1 mode register
000058	Timer A2 mode register
000059	Timer A3 mode register
00005A	Timer A4 mode register
00005B	Timer B0 mode register
00005C	Timer B1 mode register
00005D	Timer B2 mode register
00005E	Processor mode register
00005F	
000060	Watchdog timer
000061	Watchdog timer frequency selection flag
000062	
000063	
000064	
000065	
000066	
000067	
000068	
000069	
00006A	
00006B	
00006C	
00006D	
00006E	
00006F	
000070	A-D conversion interrupt control register
000071	UART0 transmission interrupt control register
000072	UART0 receive interrupt control register
000073	UART1 transmission interrupt control register
000074	UART1 receive interrupt control register
000075	Timer A0 interrupt control register
000076	Timer A1 interrupt control register
000077	Timer A2 interrupt control register
000078	Timer A3 interrupt control register
000079	Timer A4 interrupt control register
00007A	Timer B0 interrupt control register
00007B	Timer B1 interrupt control register
00007C	Timer B2 interrupt control register
00007D	$\overline{INT_0}$ interrupt control register
00007E	$\overline{INT_1}$ interrupt control register
00007F	$\overline{INT_2}$ interrupt control regiter

Fig.2 Location of peripheral devices and interrupt control registers

STACK POINTER (S)

Stack pointer (S) is an 16-bit register. It is used during a subroutine call or interrupts. It is also used during stack, stack pointer relative, or stack pointer relative indirect indexed Y addressing mode.

PROGRAM COUNTER (PC)

Program counter (PC) is a 16-bit counter that indicates the low-order 16 bits of the next program memory address to be executed. These is a bus interface unit between the program memory and the CPU, so that the program memory is accessed through bus interface unit. This is described later.

PROGRAM BANK REGISTER (PG)

Program bank register is an 8-bit register that indicates the high-order 8 bits of the next program memory address to be executed. When a carry occurs by incrementing the contents of the program counter, the contents of the program bank register (PG) is incremented by 1. Also, when a carry or borrow occurs after adding or subtracting the offset value to or from the contents of the program counter (PC) using branch instruction, the contents of the program bank register (PG) is incremented or decremented by 1 so that programs can be written without worrying about bank boundaries.

DATA BANK REGISTER (DT)

Data bank register (DT) is an 8-bit register. With some addressing modes, a part of the data bank register (DT) is used to specify a memory address. The contents of data bank register (DT) is used as the high-order 8 bits of a 24-bit address. Addressing modes that use the data bank register (DT) are direct indirect, direct indexed X indirect, direct indirect indexed Y, absolute, absolute bit, absolute indexed X, absolute indexed Y, absolute bit relative, and stack pointer relative indirect indexed Y.

DIRECT PAGE REGISTER (DPR)

Direct page register (DPR) is a 16-bit register. Its contents is used as the base address of a 256-byte direct page area. The direct page area is allocated in bank 0, but when the contents of DPR is $FF01_{16}$ or greater, the direct page area spans across bank 0 and bank 1. All direct addressing modes use the contents of the direct page register (DPR) to generate the data address. If the low-order 8 bits of the direct page register (DPR) is "00_{16}", the number of cycles required to generate an address is minimized. Normally the low-order 8 bits of the direct page register (DPR) is set to "00_{16}".

PROCESSOR STATUS REGISTER (PS)

Processor status register (PS) is an 11-bit register. It consists of a flag to indicate the result of operation and CPU interrupt levels.

Branch operations can be performed by testing the flags C, Z, V, and N.

The details of each processor status register bit are described below.

1. Carry flag (C)

The carry flag contains the carry or borrow generated by the ALU after an arithmetic operation. This flag is also affected by shift and rotate instructions. This flag can be set and reset directly with the SEC and CLC instructions or with the SEP and CLP instructions.

2. Zero flag (Z)

This zero flag is set if the result of an arithmetic operation or data transfer is zero and reset if it is not. This flag can be set and reset directly with the SEP and CLP instructions.

3. Interrupt disable flag (I)

When the interrupt disable flag is set to "1", all interrupts except watchdog timer, $\overline{\text{DBC}}$, and software interrupt are disabled. This flag is set to "1" automatically when these is an interrupt. It can be set and reset directly with the SEI and CLI instructions or SEP and CLP instructions.

4. Decimal mode flag (D)

The decimal mode flag determines whether addition and subtraction are performed as binary or decimal. Binary arithmetic is performed when this flag is "0". If it is "1", decimal arithmetic is performed with each word treated as two or four digit decimal. Arithmetic operation is performed using four digits when the data length flag m is "0" and with two digits when it is "1". (Decimal operation is possible only with the ADC and SBC instructions.) This flag can be set and reset with the SEP and CLP instructions.

The bus interface unit operates using one of the waveforms (1) to (6) shown in Figure 5. The standard waveforms are (1) and (2).

The ALE signal is used to latch only the address signal from the multiplexed signal containing data and address.

The $\overline{E}$ signal becomes "L" when the bus interface unit reads an instruction code or data from memory or when it writes data to memory. Whether to perform read or write is controlled by the R/$\overline{W}$ signal. Read is performed when the R/$\overline{W}$ signal is "H" state and write is performed when it is "L" state.

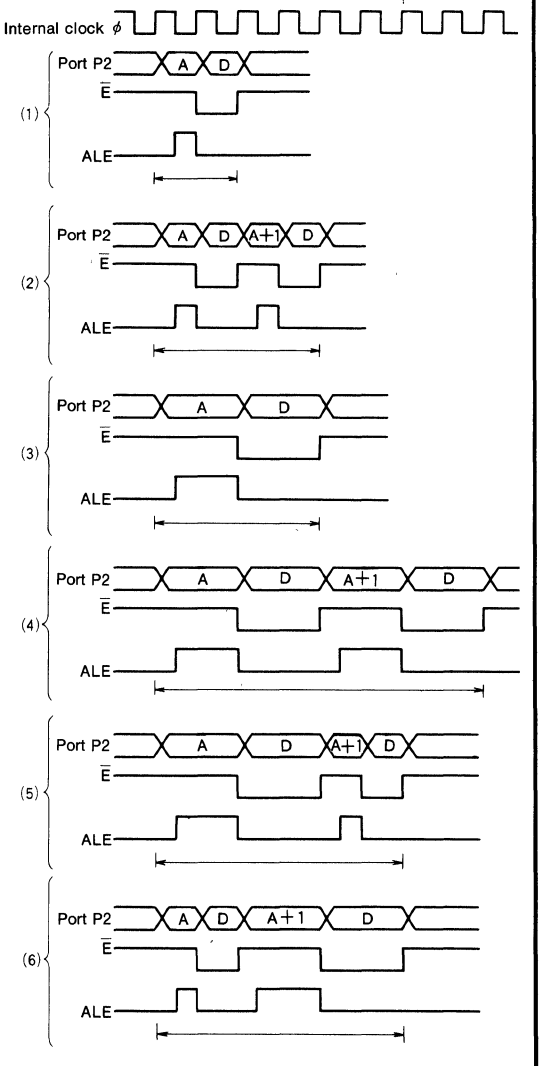
Waveform (1) in Figure 5 is used to access a single byte or two bytes simultaneously. To read or write two bytes simultaneously, the first address accessed must be even. Furthermore, when accessing an external memory area in memory expansion mode or microprocessor mode, set the bus width selection input pin BYTE to "L". (external data bus width to 16 bits) The internal memory area is always treated as 16-bit bus width regardless of BYTE.

When performing 16-bit data read or write, if the conditions for simultaneously accessing two bytes are not satisfied, waveform (2) is used to access each byte one by one. However, when prefetching the instruction code, if the address of the instruction code is odd, waveform (1) is used, and only one byte is read in the instruction queue buffer.

The signals A_0 and $\overline{BHE}$ in Figure 5 are used to control these cases: 1-byte read from even address, 1-byte read from odd address, 2-byte simultaneous read from even and odd addresses, 1-byte write to even address, 1-byte write to odd address, or 2-byte simultaneous write to even and odd addresses. The A_0 signal that is the address bit 0 is "L" when an even number address is accessed. The $\overline{BHE}$ signal becomes "L" when an odd number address is accessed.

The bit 2 of processor mode register (address $5E_{16}$) is the wait bit. When this bit is set to "0", the ALE signal and $\overline{E}$ signal are extended and the access time is doubled when accessing an external memory area in memory expansion mode or microprocessor mode. However, these signals are not extended when an internal memory area is accessed. When the wait bit is "1", the access time is not extended for any access. Waveform (3) is an expansion of waveform (1). Waveform (4),(5), and (6) are expansion of the entire waveform (2), first half of waveform (2), and the last half of waveform (2) respectively.

Instruction code read, data read, and data write are described below.



A : Address
D : Data

These waveforms are at the memory expansion mode and the microprocessor mode

Signal \ Access method	Access 2-byte simultaneously	Access even address 1-byte	Access odd address 1-byte
A_0	"L"	"L"	"H"
$\overline{BHE}$	"L"	"H"	"L"

Fig.5 Relationship between access method and signals A_0 and $\overline{BHE}$

INTERRUPTS

Table 1 shows the interrupt types and the corresponding interrupt vector addresses. Reset is also treated as a type of interrupt and is discussed in this section, too.

$\overline{DBC}$ is an interrupt used during debugging.

Interrupts other than reset, $\overline{DBC}$, watchdog timer, zero divide, and BRK instruction all have interrupt control registers. Table 2 shows the addresses of the interrupt control registers and Figure 6 shows the bit configuration of the interrupt control register.

The interrupt request bit is automatically cleared by the hardware during reset or when processing an interrupt. Also, interrupt request bits other than $\overline{DBC}$ and watchdog timer can be cleared by software.

$\overline{INT_2}$ to $\overline{INT_0}$ are external interrupts and whether to cause an interrupt at the input level (level sense) or at the edge (edge sense) can be selected with the level sense/edge sense selection bit. Furthermore, the polarity of the interrupt input can be selected with polarity selection bit.

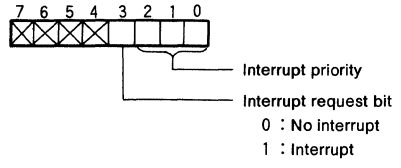
Timer and UART interrupts are described in the respective section.

The priority of interrupts when multiple interrupts are caused simultaneously is partially fixed by hardware, but, it can also be adjusted by software as shown in Figure 7. The hardware priority is fixed the following:

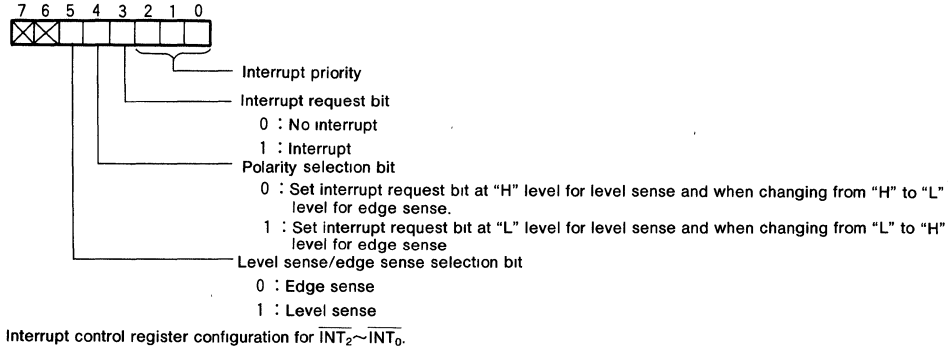
reset > $\overline{DBC}$ > watchdog timer > other interrupts

Table 1. Interrupt types and the interrupt vector addresses

Interrupts	Vector addresses
A-D conversion	$00FFD6_{16}$ $00FFD7_{16}$
UART1 transmit	$00FFD8_{16}$ $00FFD9_{16}$
UART1 receive	$00FFDA_{16}$ $00FFDB_{16}$
UART0 transmit	$00FFDC_{16}$ $00FFDD_{16}$
UART0 receive	$00FFDE_{16}$ $00FFDF_{16}$
Timer B2	$00FFE0_{16}$ $00FFE1_{16}$
Timer B1	$00FFE2_{16}$ $00FFE3_{16}$
Timer B0	$00FFE4_{16}$ $00FFE5_{16}$
Timer A4	$00FFE6_{16}$ $00FFE7_{16}$
Timer A3	$00FFE8_{16}$ $00FFE9_{16}$
Timer A2	$00FFEA_{16}$ $00FFEB_{16}$
Timer A1	$00FFEC_{16}$ $00FFED_{16}$
Timer A0	$00FFEE_{16}$ $00FFEF_{16}$
$\overline{INT_2}$ external interrupt	$00FFF0_{16}$ $00FFF1_{16}$
$\overline{INT_1}$ external interrupt	$00FFF2_{16}$ $00FFF3_{16}$
$\overline{INT_0}$ external interrupt	$00FFF4_{16}$ $00FFF5_{16}$
Watchdog timer	$00FFF6_{16}$ $00FFF7_{16}$
$\overline{DBC}$ (unusable)	$00FFF8_{16}$ $00FFF9_{16}$
Break instruction	$00FFFA_{16}$ $00FFFB_{16}$
Zero divide	$00FFFC_{16}$ $00FFFD_{16}$
$\overline{Reset}$	$00FFFE_{16}$ $00FFFF_{16}$



Interrupt control register configuration for A-D converter, UART0, UART1, timer A0 to timer A4, and timer B0 to timer B2



Interrupt control register configuration for $\overline{INT_2}$~$\overline{INT_0}$.

Fig. 6 Interrupt control register configuration

As shown in Figure 9, there are three different interrupt priority resolution time from which one is selected by software. After the selected time has elapsed, the highest priority is determined and is processed after the currently executing instruction has been completed.

The time is selected with bits 4 and 5 of the processor mode register (address $5E_{16}$) shown in Figure 10. Table 4 shows the relationship between these bits and the number of cycles. After a reset, the processor mode register is initialized to "00_{16}" and therefore, the longest time is selected.

However, the shortest time may be selected by software.

Table 3. Value set in processor interrupt level (IPL) during an interrupt

Interrupt types	Setting value
Reset	0
$\overline{DBC}$	7
Watchdog timer	7
Zero divide	Not change value of IPL.
BRK instruction	Not change value of IPL.

Table 4. Relationship between priority level evaluation time selection bit and number of cycles

Priority level resolution time selection bit		Number of cycles
Bit 5	Bit 4	
0	0	7 cycles of ϕ
0	1	4 cycles of ϕ
1	0	2 cycles of ϕ

ϕ : internal clock

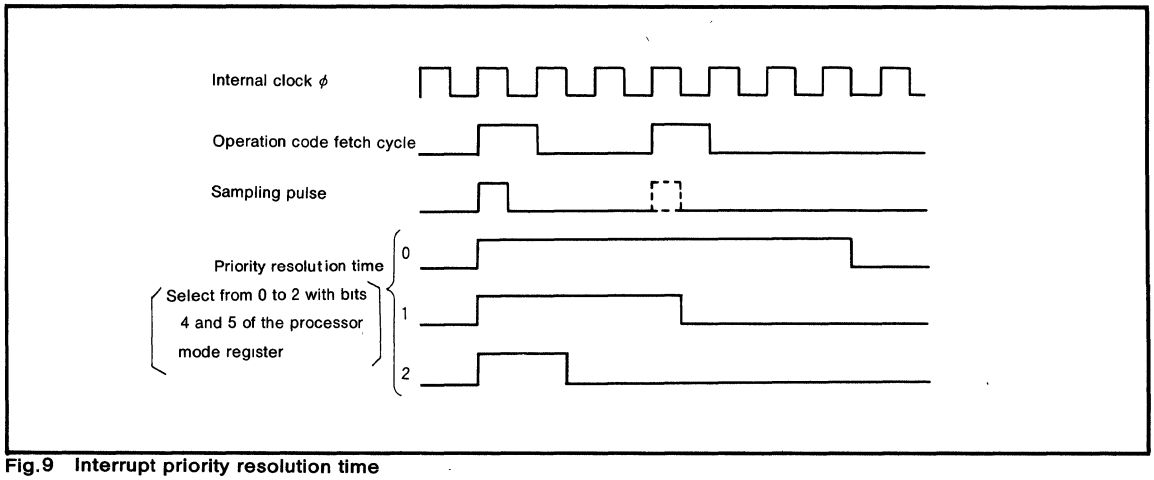


Fig.9 Interrupt priority resolution time

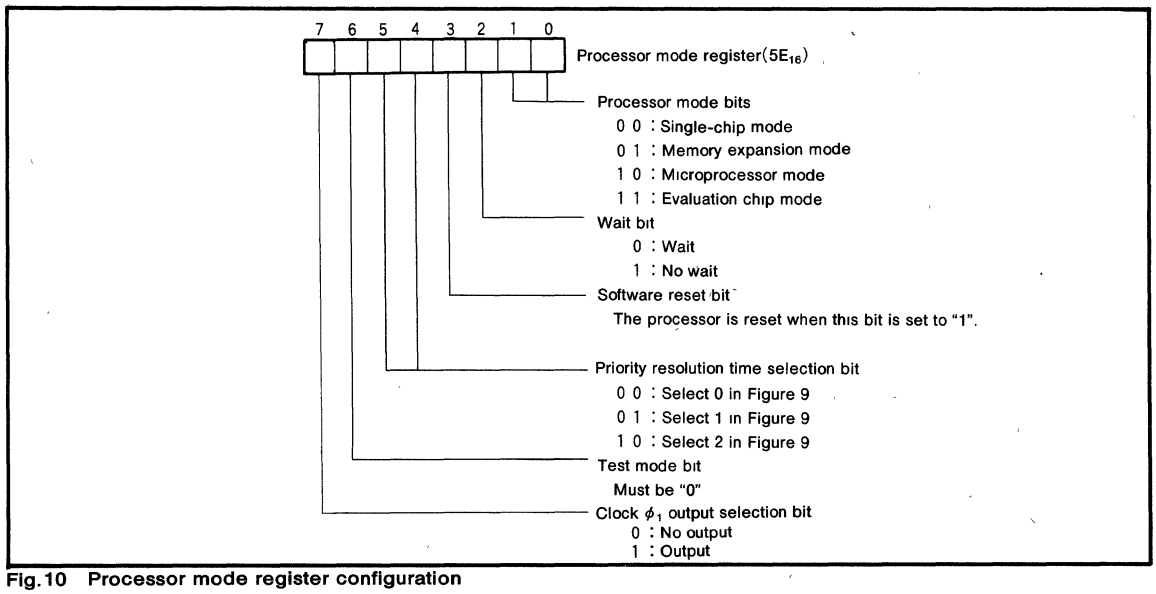


Fig.10 Processor mode register configuration

When bit 2 of the timer Ai mode register is "1", the output is generated from TAi_{OUT} pin. The output is toggled each time the contents of the counter reaches to 0000_{16}. When the contents of the count start flag is "0", "L" is output from TAi_{OUT} pin.

When bit 2 is "0", TAi_{OUT} can be used as a normal port pin.

When bit 4 is "0", TAi_{IN} can be used as a normal port pin.

When bit 4 is "1", counting is performed only while the input signal from the TAi_{IN} pin is "H" or "L" as shown in Figure 14. Therefore, this can be used to measure the pulse width of the TAi_{IN} input signal. Whether to count while the input signal is "H" or while it is "L" is determined by bit 3. If bit 3 is "1", counting is performed while the TAi_{IN} pin input signal is "H" and if bit 3 is "0", counting is performed while it is "L".

Note that the duration of "H" or "L" on the TAi_{IN} pin must be two or more cycles of the timer count source.

When data is written to timer Ai register, the same data is also written to the reload register and the counter. The count start flag corresponding to the written timer is cleared to "0" and count is stopped. The contents of the counter can be read at any time.

When the value set in the timer Ai register is n, the timer frequency dividing ratio is $1/(n+1)$.

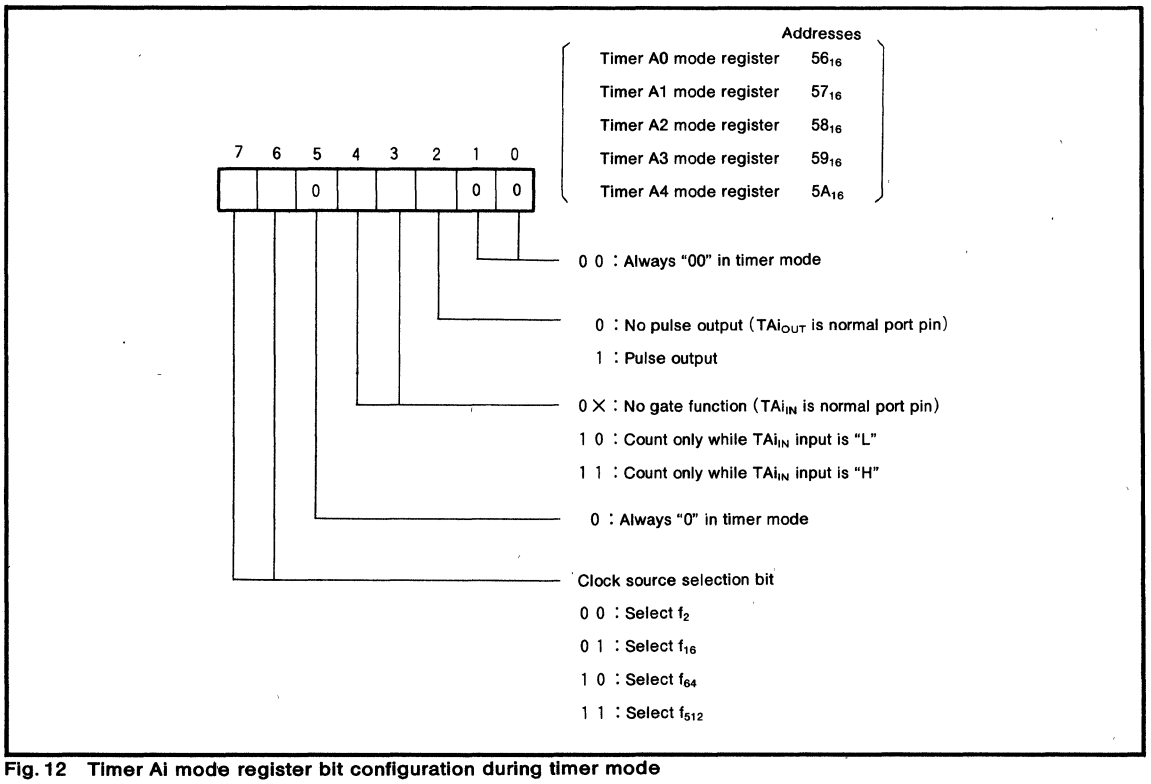


Fig. 12 Timer Ai mode register bit configuration during timer mode

(2) Event counter mode [01]

Figure 15 shows the bit configuration of the timer Ai mode register during event counter mode. In event counter mode, the bit 0 of the timer Ai mode register must be "1" and bit 1 and 5 must be "0".

The input signal from the TAi_{IN} pin is counted when the count start flag shown in Figure 13 is "1" and counting is stopped when it is "0".

Count is performed at the falling edge of the input signal when bit 3 is "0" and at the rising edge of the signal when it is "1".

In event counter mode, whether to increment or decrement the count can be selected with the up-down flag or the input signal from the TAi_{OUT} pin.

When bit 4 of the timer Ai mode register is "0", the up-down flag is used to determine whether to increment or decrement the count (decrement when the flag is "0" and increment when it is "1"). Figure 16 shows the bit configuration of the up-down flag.

When bit 4 of the timer Ai mode register is "1", the input signal from the TAi_{OUT} pin is used to determine whether to increment or decrement the count. However, note that bit 2 must be "0" if bit 4 is "1" because if bit 2 is "1", TAi_{OUT} pin becomes an output pin with pulse output.

Also, the increment or decrement count interval must be at least two cycles of the timer count source.

The count is decremented when the input signal from the TAi_{OUT} pin is "L" and incremented when it is "H".

An interrupt request signal is generated and the interrupt request bit in the timer Ai interrupt control register is set when the counter reaches 0000_{16} (decrement count) or $FFFF_{16}$ (increment count). At the same time, the contents of the reload register is transferred to the counter and the count is continued.

At decrement count, if bit 2 is "1", the output is generated from the TAi_{OUT} pin. The output is toggled each time the counter reaches 0000_{16}. At increment count, If bit 2 is "1", the output is generated from the TAi_{OUT} pin. The output is toggled each time the counter reaches $FFFF_{16}$.

If bit 2 is "0", TAi_{OUT} pin can be used as a normal port pin. However, if bit 4 is "1" and the TAi_{OUT} pin is used as an output pin, the output from the pin changes the count direction. Therefore, bit 4 should be "0" unless the output from the TAi_{OUT} pin is to be used to select the count direction.

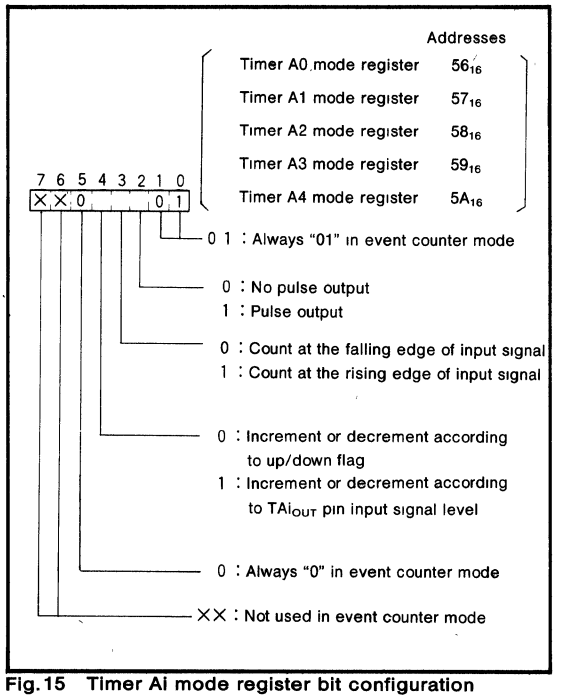


Fig. 15 Timer Ai mode register bit configuration during event counter mode

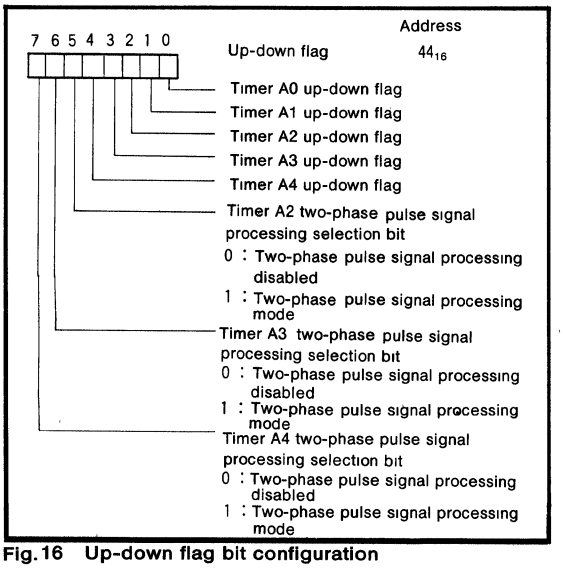


Fig. 16 Up-down flag bit configuration

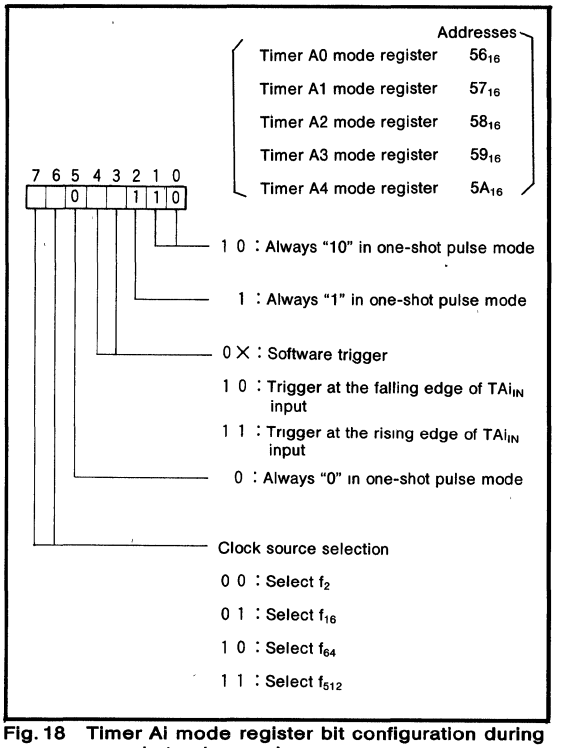


Fig. 18 Timer Ai mode register bit configuration during one-shot pulse mode

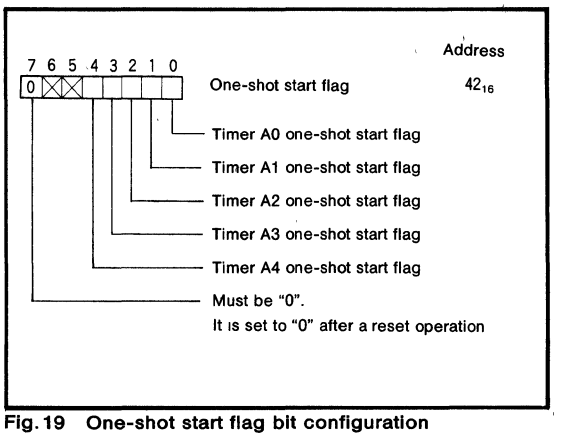


Fig. 19 One-shot start flag bit configuration

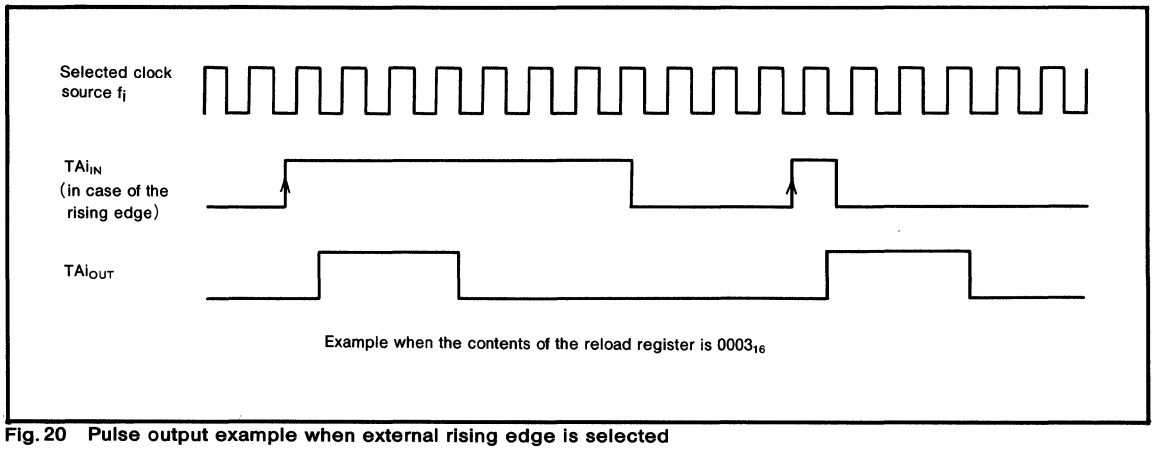


Fig. 20 Pulse output example when external rising edge is selected

(4) Pulse width modulation mode (11)

Figure 22 shows the bit configuration of the timer Ai mode register during pulse width modulation mode. In pulse width modulation mode, bits 0, 1, and 2 must be set to "1".

Bit 5 is used to determine whether to perform 16-bit length pulse width modulator or 8-bit length pulse width modulator. 16-bit length pulse width modulator is performed when bit 5 is "0" and 8-bit length pulse width modulator is performed when it is "1". The 16-bit length pulse width modulator is described first.

The pulse width modulator can be started with a software trigger or with an input signal from a TAi_{IN} pin (external trigger).

The software trigger mode is selected when bit 4 is "0". Pulse width modulator is started and pulse is output from TAi_{OUT} when the timer Ai start flag is set to "1".

The external trigger mode is selected when bit 4 is "1". Pulse width modulator starts when a trigger signal is input from the TAi_{IN} pin when the timer Ai start flag is "1". Whether to trigger at the fall or rise of the trigger signal is determined by bit 3. The trigger is at the fall of the trigger signal when bit 3 is "0" and at the rise when it is "1".

If data is written in timer Ai when the timer Ai start flag is "0" (that is when pulse width modulator is halted), the data is written in the reload register and the counter. Then when the time Ai start flag is set to "1" and a software trigger or an external trigger is issued to start modulation, the waveform shown in Figure 23 is output continuously. Once modulation is started, triggers are not accepted. If the value in the reload register is m, the duration "H" of pulse is

$$\frac{1}{\text{selected clock frequency}} \times m$$

and the output pulse period is

$$\frac{1}{\text{selected clock frequency}} \times (2^{16}-1).$$

An interrupt request signal is generated and the interrupt request bit in the timer Ai interrupt control register is set at each fall of the output pulse.

To change the pulse width, write the data during the time after the pulse falls but before the rise of the next pulse. In contrast to the timer mode, data is written in the reload register only and not in the counter. Furthermore, the timer Ai start flag is unaffected and the counter is not stopped.

The contents of the reload register are transferred to the counter just before the rise of the next pulse so that the pulse width is changed from the next output pulse.

To read the timer Ai is performed from the reload register instead of the counter.

The 8-bit length pulse width modulator is described next.

The 8-bit length pulse width modulator is selected when the timer Ai mode register bit 5 is "1".

The reload register and the counter are both divided into 8-bit halves.

The low order 8 bits function as a prescaler and the high order 8 bits function as the 8-bit length pulse width modulator. The prescaler counts the clock selected by bits 6 and 7. A pulse is generated when the counter reaches 0000_{16} as shown in Figure 24. At the same time, the contents of the reload register is transferred to the counter and count is continued.

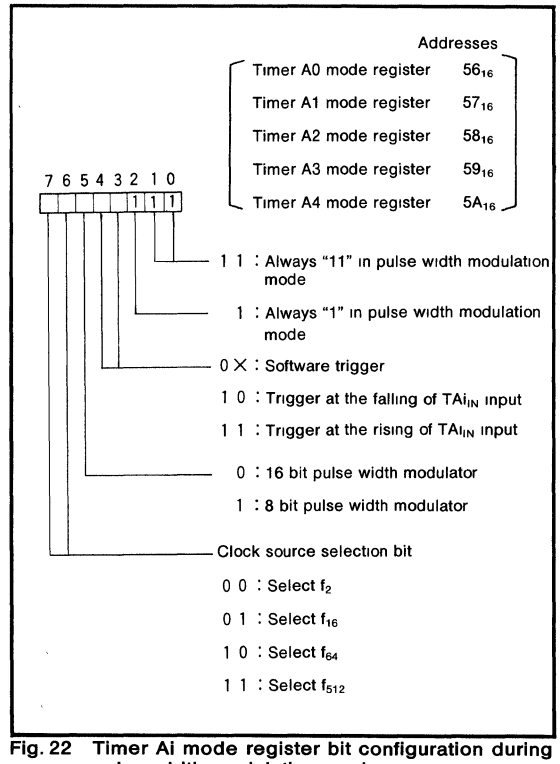


Fig. 22 Timer Ai mode register bit configuration during pulse width modulation mode

TIMER B

Figure 25 shows a block diagram of timer B.

Timer B has three modes; timer mode, event counter mode, and pulse period measurement/pulse width measurement mode. The mode is selected with bits 0 and 1 of the timer Bi mode register (i = 0 to 2). Each of these modes is described below.

(1) Timer mode [00]

Figure 26 shows the bit configuration of the timer Bi mode register during timer mode. Bits 0, and 1 of the timer Bi mode register must always be "0" in timer mode.

Bits 6 and 7 are used to select the clock source. The counting of the selected clock starts when the count start flag "1" and stops when "0".

As shown in Figure 13, the timer Bi count start flag is at the same address as the timer Ai count start flag. The count is decremented, an interrupt occurs, and the interrupt request bit in the timer Bi interrupt control register is set when the contents becomes 0000_{16}. At the same time, the contents of the reload register is stored in the counter and count is continued.

Timer Bi does not have a pulse output function or a gate function like timer A.

When data is written to timer Bi register, the same data is written to the reload register and the counter. The count start flag of the corresponding to the written timer is cleared to "0" and count is stopped. The contents of the counter can be read at any time.

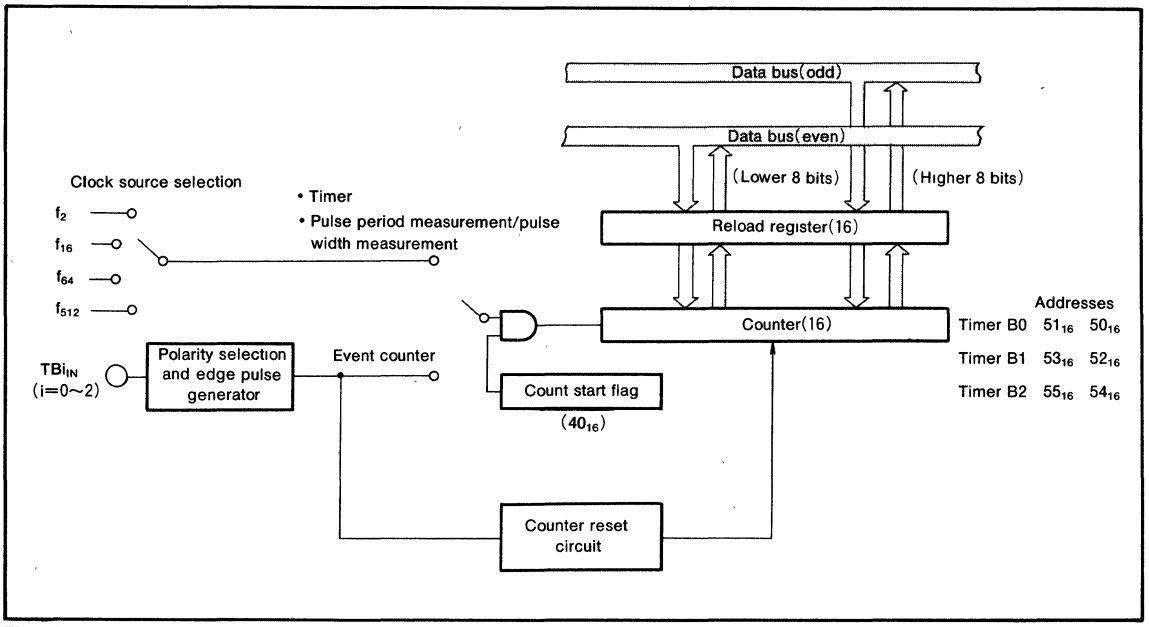


Fig. 25 Timer B block diagram

After the contents of the counter is transferred to the reload register, an interrupt request signal is generated and the interrupt request bit in the timer Bi interrupt control register is set. However, no interrupt request signal is generated when the contents of the counter is transferred first time to the reload register after the count start flag is set to "1".

When bit 3 is "1", the pulse width measurement mode is selected. Pulse width measurement mode is similar to pulse period measurement mode except that the clock is counted from the fall of the TBi_{IN} pin input signal to the next rise or from the rise of the input signal to the next fall as shown in Figure 30.

When timer Bi is read, the contents of the reload register is read.

Note that in this mode, the interval between the fall of the TBi_{IN} pin input signal to the next rise or from the rise to the next fall must be at least two cycles of the timer count source.

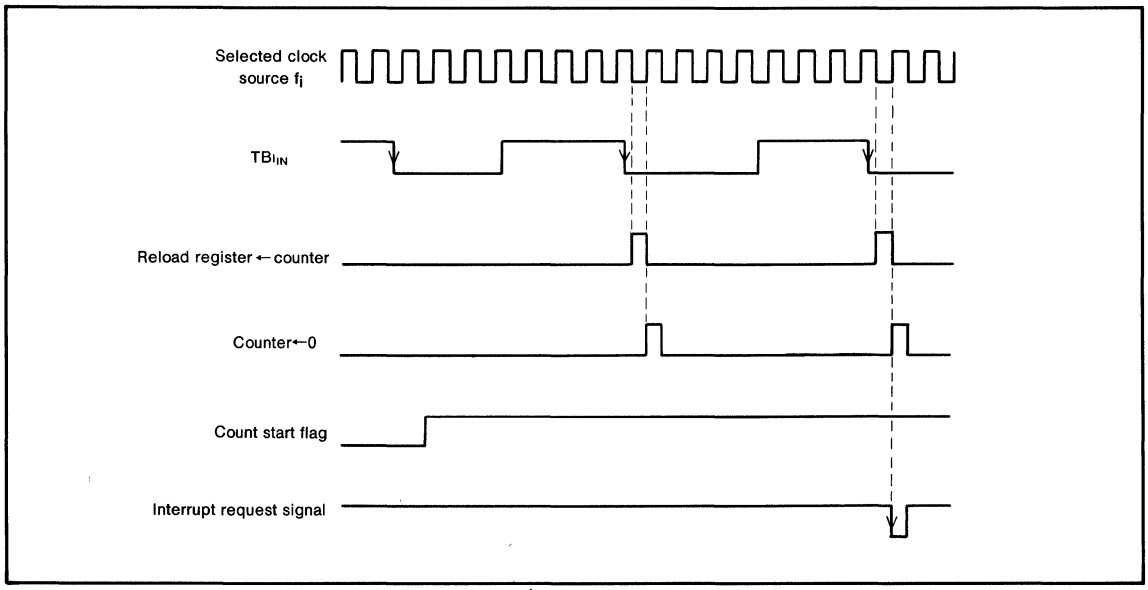


Fig. 29 Pulse period measurement mode operation (example of measuring the interval between the falling edge to next falling one)

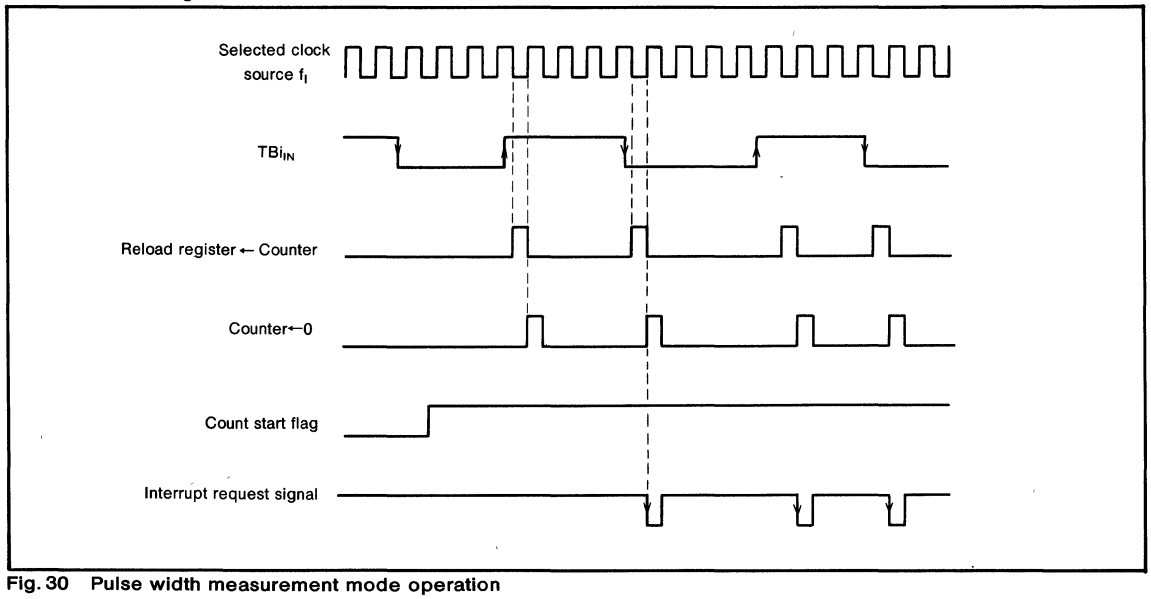


Fig. 30 Pulse width measurement mode operation

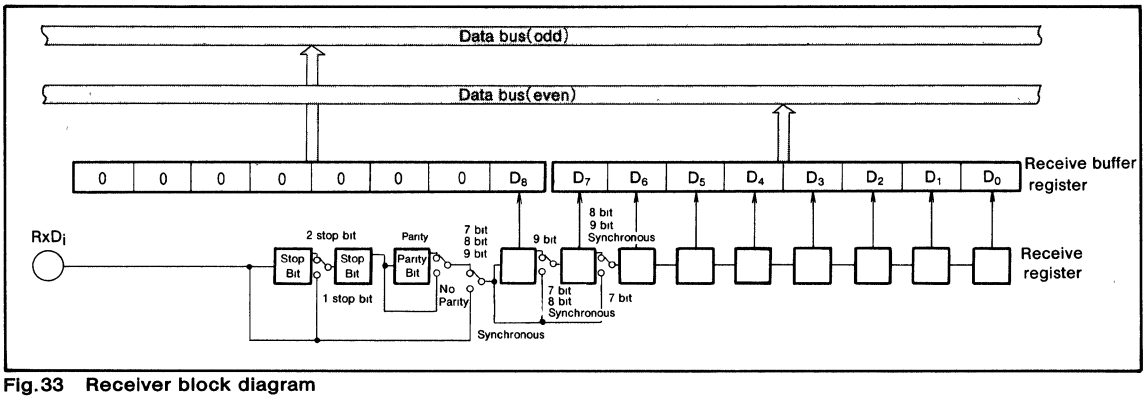


Fig.33 Receiver block diagram

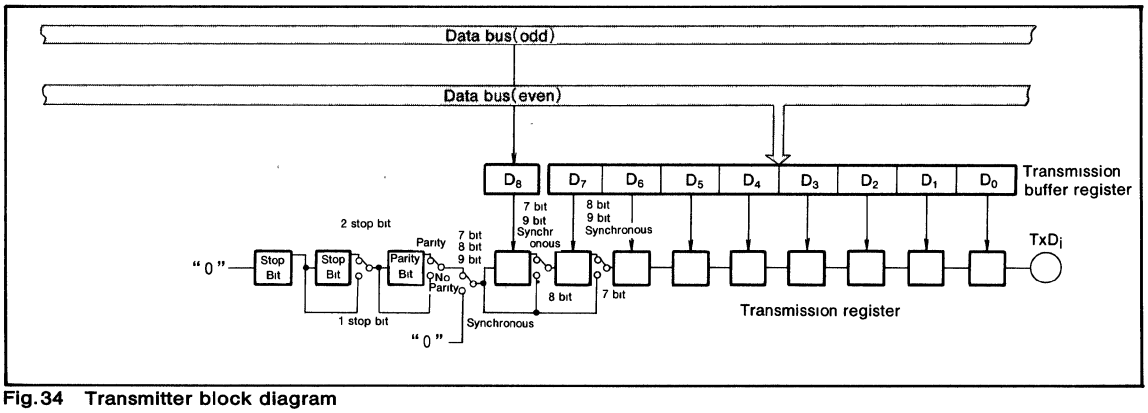


Fig.34 Transmitter block diagram

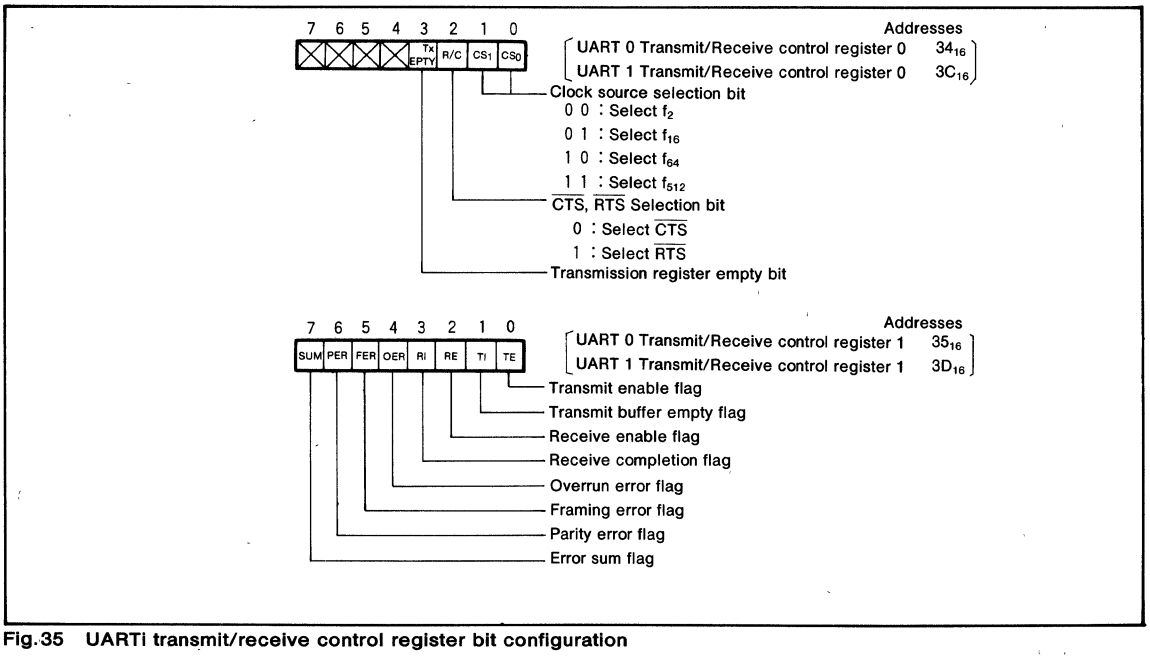


Fig.35 UARTi transmit/receive control register bit configuration

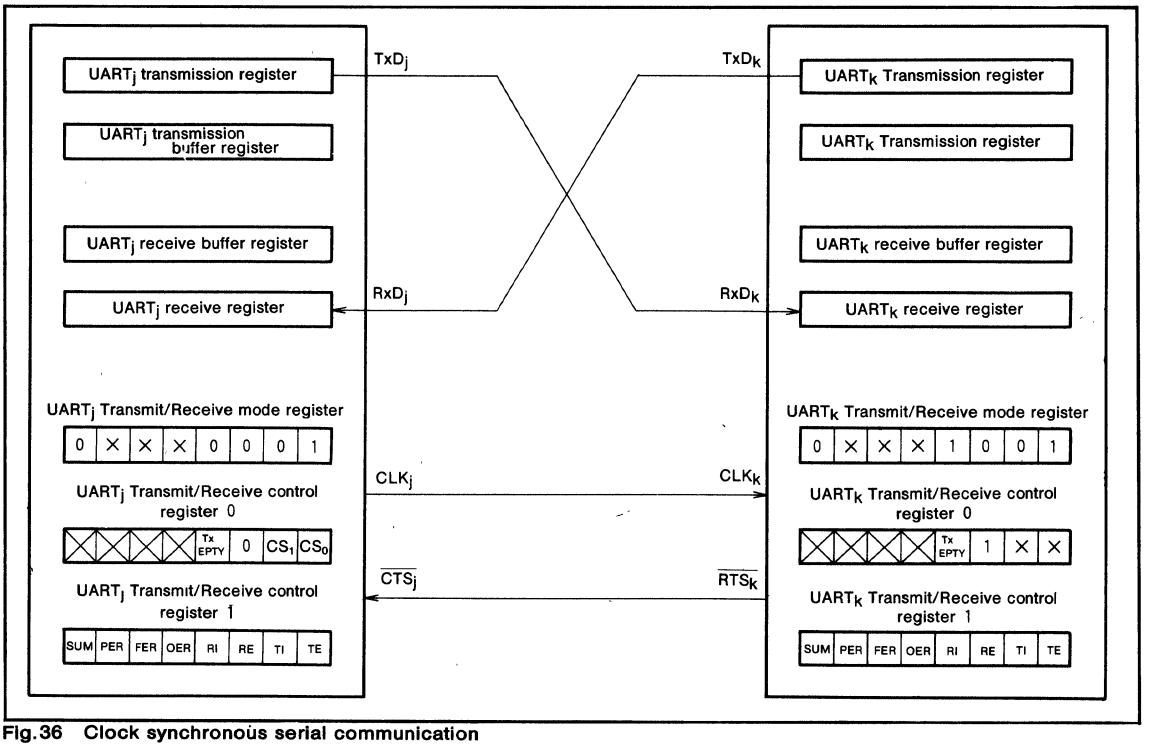


Fig.36 Clock synchronous serial communication

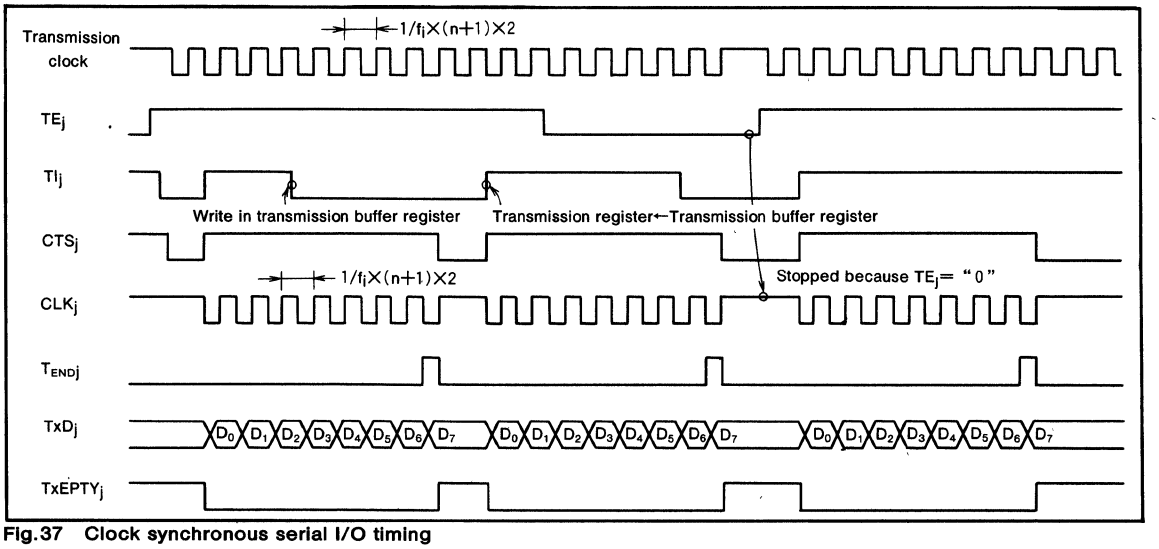


Fig.37 Clock synchronous serial I/O timing

Bit 6 is the parity bit selection bit which indicates whether to add parity bit or not.
Bits 4 to 6 should be set or reset according to the data format of the communicating devices.
Bit 7 is the sleep selection bit. The sleep mode is described later.
The UART$_i$ transmit/receive control regisger 0 bit 2 is used to determine whether to use $\overline{CTS_i}$ input or $\overline{RTS_i}$ output. $\overline{CTS_i}$ input used if bit 2 is "0" and $\overline{RTS_i}$ output is used if bit 2 is "1".
If $\overline{CTS_i}$ input is selected, the user can control whether to stop or start transmission by external $\overline{CTS_i}$ input. $\overline{RTS_i}$ will be described later.

Transmission

Transmission is started when the bit 0 (TEi flag) of UART$_i$ transmit/receive control register 1 is "1", the bit 1 (TI$_i$ flag) is "0", and $\overline{CTS_i}$ input is "L" if $\overline{CTS_i}$ input is selected. As shown in Figure 38 and 39, data is output from the TxD$_i$ pin with the stop bit and parity bit specified by the bits 4 to 6 of UART$_i$ transmit/receive mode register bits. The data is output from the least significant bit.
The TI$_i$ flag indicates whether the transmission butter is empty or not. It is cleared to "0" when data is written in the transmission buffer and set to "1" when the contents of the transmission buffer register is transferred to the transmission register.
When the transmission register becomes empty after the contents has been transmitted, data is transferred automatically form the transmission buffer register to the transmission register if the next transmission start condition is satisfied.
Once transmission has started, the TE$_i$ flag, TI$_i$ flag, and $\overline{CTS_i}$ signal (if $\overline{CTS_i}$ input is selected) are ignored until data transmission is completed.
Therefore, transmission does not stop until it completes even if the TE$_i$ flag is cleared during transmission.
The transmission start condition indicated by TE$_i$ flag, TI$_i$ flag, and $\overline{CTS_i}$ is checked while the T_{ENDi} signal shown in Figure 38 is "H". Therefore, data can be transmitted continuously if the next transmission data is written in the transmission butter register and TI$_i$ flag is cleared to 0 before the T_{ENDi} signal goes "H".
The bit 3 (TxEPTY$_i$ flag) of UART$_i$ transmit/receive control register 0 changes to "1" at the next cycle after the T_{ENDi} signal goes "H" and changes to "0" when transmission starts. Therefore, this flag can be used to determine whether data transmision is completed.
When the TI$_i$ flag changes from "0" to "1", the interrupt request bit in the UART$_i$ transmissoin interrupt control register is set to "1".

Receive

Receive is enabled when the bit 2 (RE$_i$ flag) of UART$_i$ transmit/receive control register 1 is set. As shown in Figure 40, the frequency divider circuit at the receiving end begin to work when a start bit is arrived and the data is received.

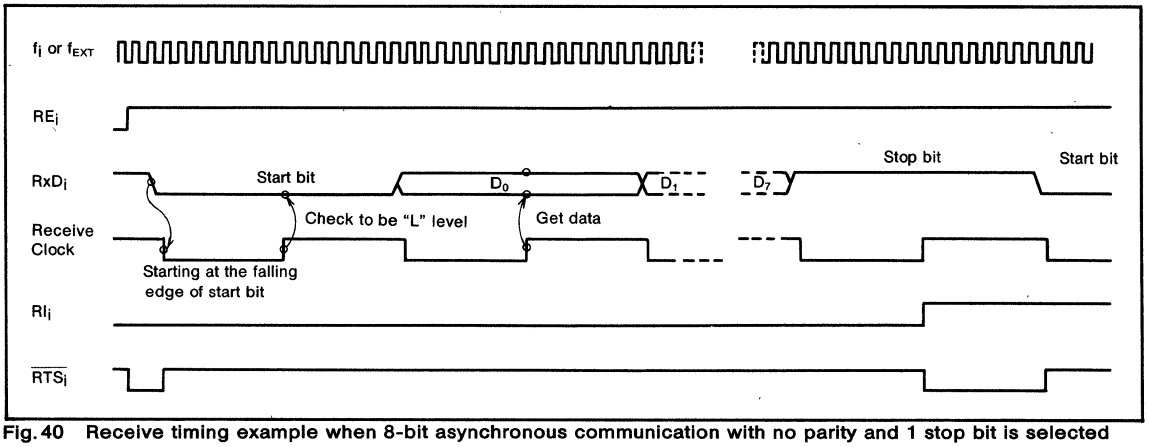


Fig. 40 Receive timing example when 8-bit asynchronous communication with no parity and 1 stop bit is selected

A-D CONVERTER

The A-D converter is an 8-bit successive approximation converter.

Figure 41 shows a block diagram of the A-D converter and Figure 42 shows the bit configuration of the A-D control register. The frequency of the A-D converter operating clock ϕ_{AD} is selected by the bit 7 of the A-D control register. When bit 7 is "0", ϕ_{AD} is the clock frequency divided by 8. That is, $\phi_{AD}=f(X_{IN})/8$. When bit 7 is "1", ϕ_{AD} is the clock frequency divided by 4 and ϕ_{AD} is $=f(X_{IN})/4$. The ϕ_{AD} during A-D conversion must be 250KHz minimum because the comparator consists of a capacity coupling amplifier.

The operating mode is selected by the bits 3 and 4 of A-D control register. The available operating modes are one-shot, repeat, single sweep, and repeat sweep.

The bit of data direction register bit corresponding to the A-D converter pin must be "0" (input mode) because the analog input port is shared with port P7.

The operation of each mode is described below.

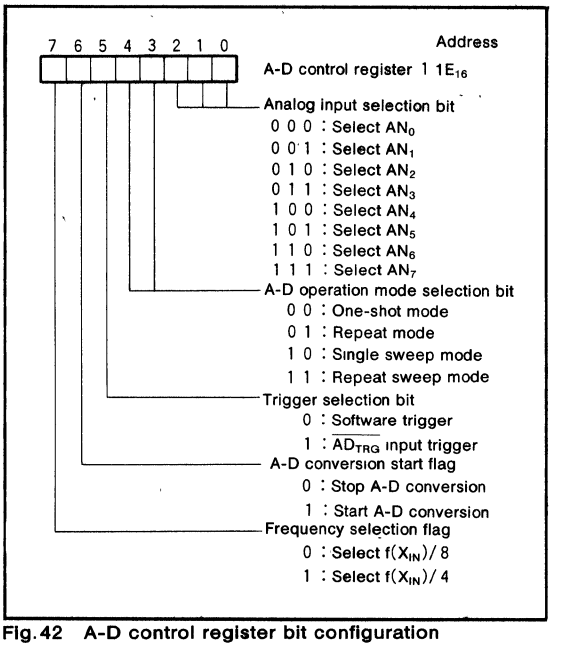


Fig. 42 A-D control register bit configuration

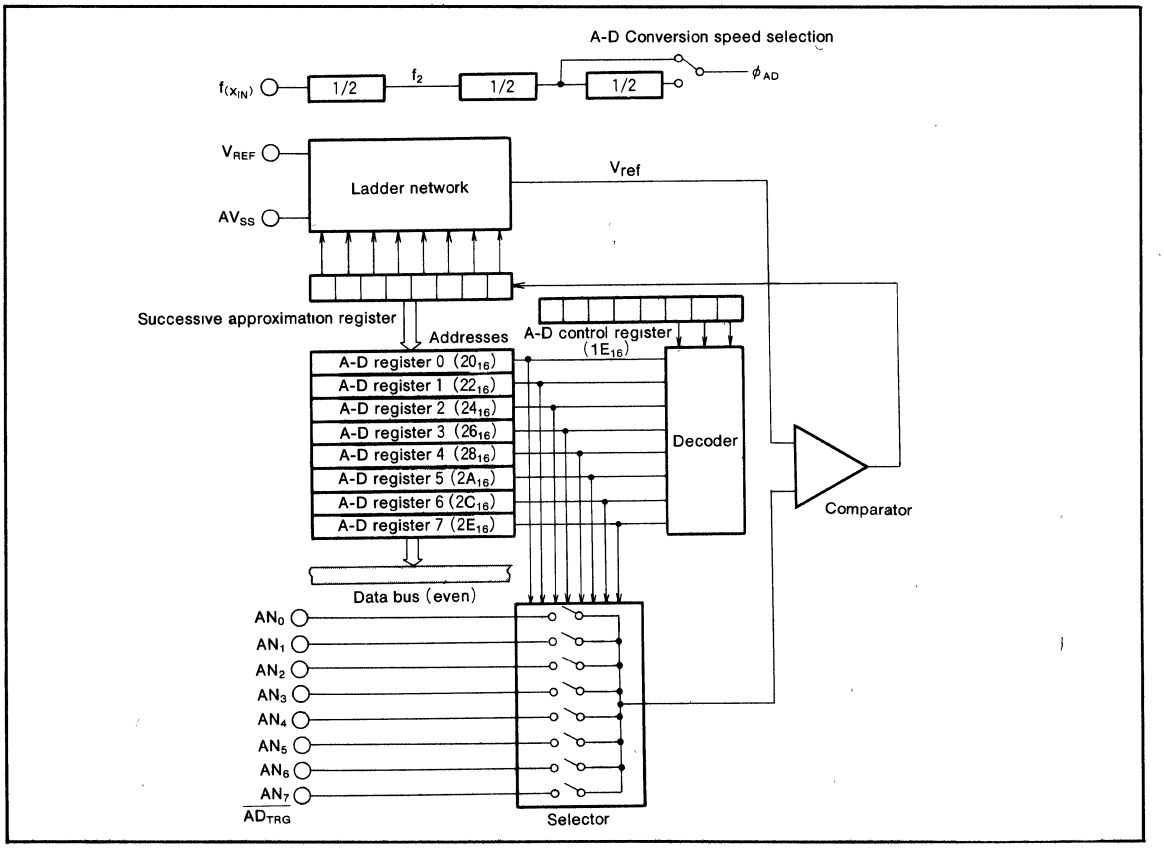


Fig. 41 A-D converter block diagram

WATCHDOG TIMER

The watchdog timer is used to detect unexpected execution sequence caused by software run-away.

Figure 43 shows a block diagram of the watchdog timer. The watchdog timer consists of a 12-bit binary counter.

The watchdog timer counts the clock frequency divided by 32 (f_{32}) or by 512 (f_{512}). Whether to count f_{32} or f_{512} is determined by the watchdog timer frequency selection flag shown in Figure 44. f_{512} is selected when the flag is "0" and f_{32} is selected when it is "1". The flag is cleared after reset.

FFF_{16} is set in the watchdog timer when "L" or $2V_{CC}$ is applied to the $\overline{RESET}$ pin, STP instruction is executed, data is written to the watchdog timer, or the most significant bit of the watchdog timer become "0".

After FFF_{16} is set in the watchdog timer, the contents of watchdog timer is decremented by one at every cyle of selected frequency f_{32} or f_{512}, and after 2048 counts, the most significant bit of watchdog timer become "0", and a watchdog timer interrupt request bit is set, and FFF_{16} is preset in the watchdog timer.

Normally, a program is witten so that data is written in the watchdog timer before the most significant bit of the watchdog timer become "0". If this routine is not executed due to unexpected program execution, the most significant bit of the watchdog timer become eventually "0" and an interrupt is generated.

The processor can be reset by setting the bit 3 (software reset bit) of processor mode register described in Figure 10 in the interrupt section and generating a reset pulse.

The watchdog timer stops its function when the $\overline{RESET}$ pin voltage is raised to double the V_{CC} voltage.

The watchdog timer can also be used to recover from when the clock is stopped by the STP instruction. Refer to the section on clock generation circuit for more details.

The watchdog timer hold the contents during a hold state and the frequency is stopped to input.

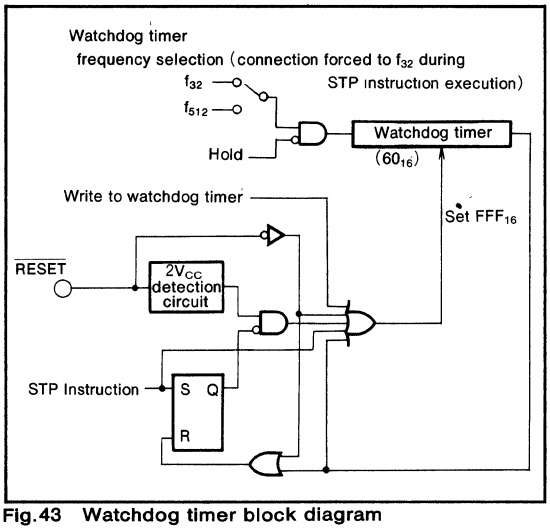


Fig.43 Watchdog timer block diagram

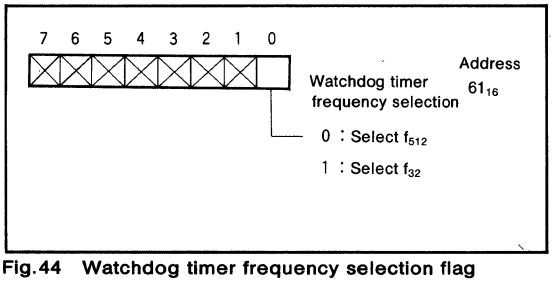


Fig.44 Watchdog timer frequency selection flag

INPUT/OUTPUT PINS

Ports P8 to P0 all have a data direction register and each bit can be programmed for input or output. A pin becomes an output pin when the corresponding data direction register is set and an input pin when it is cleared.

When pin programmed for output, the data is written to the port latch and it is output to the output pin. When a pin is programmed for output, the contents of the port latch is read instead of the value of the pin. Therefore, a previously output value can be read correctly even when the output "L" voltage is raised due to reasons such as directly driving an LED.

A pin programmed for input is floating and the value input to the pin can be read. When a pin is programmed for input, the data is written only in the port latch and the pin stays floating.

If an input/output pin is not used as an output port, clear the bit of the corresponding data direction register so that the pin become input mode.

Figure 47 shows a block diagram of ports P8 to P0 in single-chip mode and the $\overline{E}$ pin output.

In memory expansion mode, microprocessor mode, and evaluation chip mode, ports P4 to P0 are also used as address, data, and control signal pins.

Refer to the section on processor modes for more details.

PROCESSOR MODE

The bits 0 and 1 of processor mode register as shown in Figure 48 are used to select any mode of single-chip mode, memory expansion mode, microprocessor mode, and evaluation chip mode.

Ports P3 to P0 and a part of port P4 are used as address, data, and control signal I/O pins except in single-chip mode.

Figure 49 shows the functions of ports P4 to P0 in each mode.

The external memory area changes when the mode changes.

Figure 50 shows the memory map for each mode.

Refer to Figure 1 for the memory map of the single-chip mode. The external memory area can be accessed except in single-chip mode. The accessing of the external memory is affected by the BYTE pin and the bit 2 (wait bit) of processor mode register . These will be described next.

· BYTE pin

When accessing the external memory, the level of the BYTE pin is used to determine whether to use the data bus as 8-bit width or 16-bit width.

The data bus width is 8 bits when the level of the BYTE pin is "H" and port P2 becomes the data I/O pin.

The data bus width is 16 bits when the level of the BYTE pin is "L" and ports P1 and P2 become the data I/O pins.

When accessing the internal memory, the data bus width is always 16 bits regardless of the BYTE pin level.

An exclusive mode in the evaluation chip mode allows the BYTE pin level to be set to $2 \cdot V_{CC}$. In this case, the operation is slightly different from the above. This is described in the evaluation chip mode section.

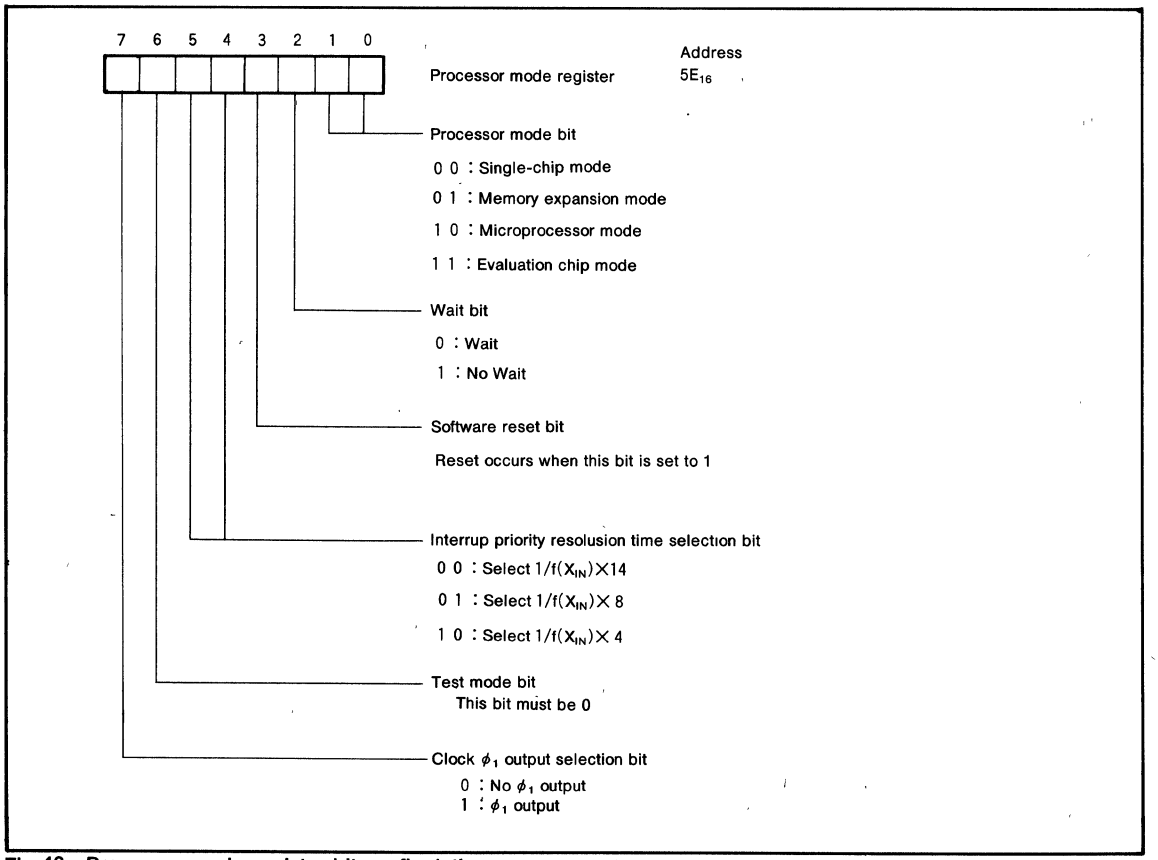


Fig. 48 Processor mode register bit configulation

· Wait bit

As shown in Figure 51, when the external memory area is accessed with the processor mode register bit 2 (wait bit) cleared to "0", the access time becomes twice the access time than the wait bit is "1" (no wait). The wait bit is cleared during reset.

The accessing of internal memory area is performed in no wait mode regardless of the wait bit.

The processor modes are described below.

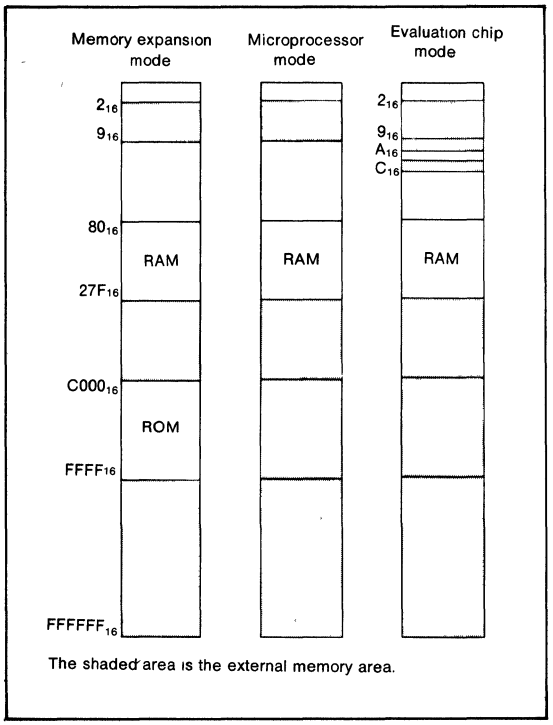


Fig.50 External memory area for each processor mode

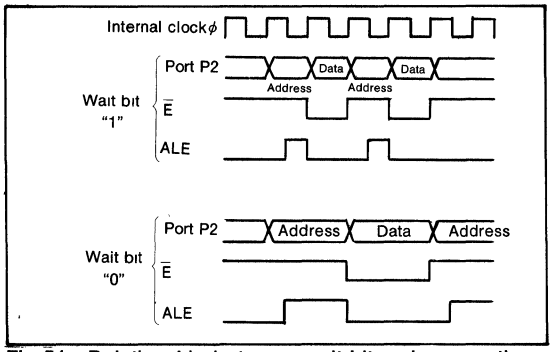


Fig.51 Relationship between wait bit and access time

(1) Single-chip mode [00]

single-chip mode is entered by connecting the CNV_{SS} pin to V_{SS} and starting from reset. Ports P4 to P0 all function as normal I/O ports.

(2) Memory expansion mode [01]

Memory expansion mode is entered by setting the processor mode bits to "01" after connecting the CNV_{SS} pin to V_{SS} and starting from reset.

Port P0 becomes an address output pin and loses its I/O port function.

Port P1 has two functions depending on the level of the BYTE pin.

When the BYTE pin level is "L", port P1 functions as an address output pin while $\overline{E}$ is "H" and as an odd address data I/O pin while $\overline{E}$ is "L". However, if an internal memory is read, external data is ignored while $\overline{E}$ is "L". In this case the I/O port function is lost.

When the BYTE pin level "H", port P1 functions as an address output pin and loses its I/O port function.

Port P2 has two functions depending on the level of the BYTE pin.

When the BYTE pin level is "L", port P2 functions as an address output pin while $\overline{E}$ is "H" and as an even address data I/O pin while $\overline{E}$ is "L". However, if an internal memory is read, external data is ignored while $\overline{E}$ is "L".

When the BYTE pin level is "H", port P2 functions as an address output pin while $\overline{E}$ is "H" and as an even and odd address data I/O pin while $\overline{E}$ is "L". However, if an internal memory is read, external data is ignored while $\overline{E}$ is "L". In this case the I/O port function is lost.

Ports $P3_0$, $P3_1$, $P3_2$, and $P3_3$ become R/$\overline{W}$, $\overline{BHE}$, ALE, and $\overline{HLDA}$ output pin respectively and lose their I/O port functions.

R/$\overline{W}$ is a read/write signal which indicates a read when it is "H" and a write when it is "L".

$\overline{BHE}$ is a byte high enable signal which indicates that an odd address is accessed when it is "L".

Therefore, two bytes at even and odd addresses are accessed simultaneously if address A_0 is "L" and $\overline{BHE}$ is "L".

CLOCK GENERATING CIRCUIT

Figure 52 shows a block diagram of the clock generator.

When an STP instruction is executed, the internal clock ϕ stops oscillating at "L" level. At the same time, FFF_{16} is written to watchdog timer and the watchdog timer input connection is forced to f_{32}. This connection is broken and connected to the input determined by the watchdog timer frequency selection flag when the most significant bit of the watchdog timer is cleared or reset.

Oscillation resumes when an interrupt is received, but the internal clock ϕ remains at "L" level until the most significant bit of the watchdog timer is cleared. This is to avoid the unstable interval at the start of oscillation when using a ceramic resonator.

When a WIT instruction is executed, the internal clock ϕ stops at "L" level, but the oscillator does not stop. The clock is restarted when an interrupt is received. Instructions can be executed immediately because the oscillator is not stopped.

The stop or wait state is released when an interrupt is received or when reset is issued. Therefore, interrupts must be enabled before executing a STP or WIT instruction.

Figure 53 shows a circuit example using a ceramic (or quartz crystal) resonator. Use the manufacturer's recommended values for constants such as capacitance which differ for each resonator. Figure 54 shows an example of using an external clock signal.

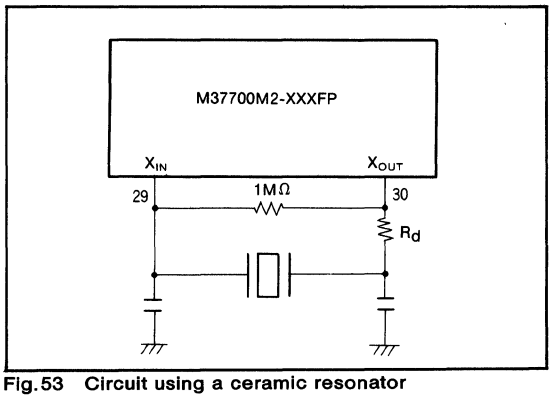


Fig.53 Circuit using a ceramic resonator

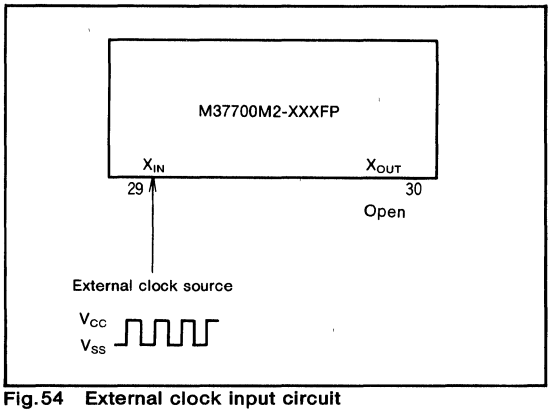


Fig.54 External clock input circuit

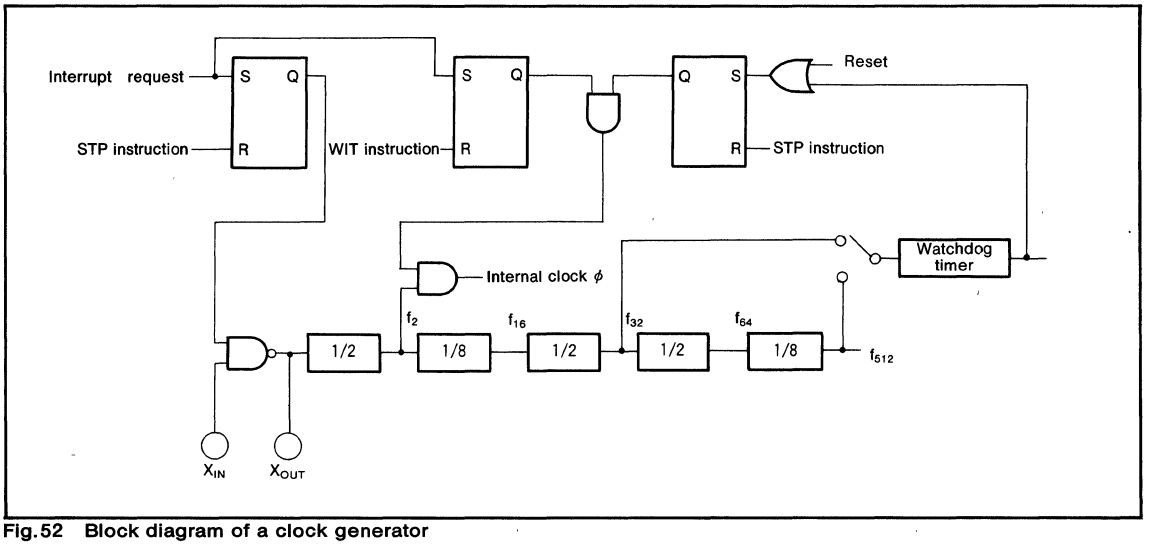


Fig.52 Block diagram of a clock generator

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Supply voltage		−0.3∼7	V
AV_{CC}	Analog supply voltage		−0.3∼7	V
V_I	Input voltage $\overline{RESET}$, CNV_{SS}, BYTE		−0.3∼12	V
V_I	Input voltage $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$, V_{REF}, X_{IN}		−0.3∼V_{CC}+0.3	V
V_O	Output voltage $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$, X_{OUT}, $\overline{E}$		−0.3∼V_{CC}+0.3	V
P_d	Power dissipation	T_a=25℃	300	mW
T_{opr}	Operating temperature		−10∼70	℃
T_{stg}	Storage temperature		−40∼125	℃

RECOMMENDED OPERATING CONDITIONS (V_{CC}=5V±10%, T_a=−10∼70℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max	
V_{CC}	Supply voltage	4.5	5.0	5.5	V
AV_{CC}	Analog supply voltage		V_{CC}		V
V_{SS}	Supply voltage		0		V
AV_{SS}	Analog supply voltage		0		V
V_{IH}	High-level input voltage $P0_0$∼$P0_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	0.8V_{CC}		V_{CC}	V
V_{IH}	High-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in single-chip mode)	0.8V_{CC}		V_{CC}	V
V_{IH}	High-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, (in memory expansion mode and microprocessor mode)	0.5V_{CC}		V_{CC}	V
V_{IL}	Low-level input voltage $P0_0$∼$P0_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	0		0.2V_{CC}	V
V_{IL}	Low-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, (in single-chip mode)	0		0.2V_{CC}	V
V_{IL}	Low-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in memory expansion mode and microprocessor mode)	0		0.16V_{CC}	V
$I_{OH(peak)}$	High-level peak output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$			−10	mA
$I_{OH(avg)}$	High-level average output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$			−5	mA
$I_{OL(peak)}$	Low-level peak output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$			10	mA
$I_{OL(avg)}$	Low-level average output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$			5	mA
$f(X_{IN})$	External clock frequency input — M37700M2-XXXFP, M37700SFP			8	MHz
	External clock frequency input — M37700M2AXXXFP, M37700SAFP			16	MHz

Note 1. Average output current is the average value of a 100ms interval.
2. The sum of $I_{OL(peak)}$ for ports P0, P1, P2, P3 and P8 must be 80mA or less, the sum of $I_{OH(peak)}$ for ports P0, P1, P2, P3 and P8 must be 80mA or less, the sum of $I_{OL(peak)}$ for ports P4, P5, P6 and P7 must be 80mA or less, and the sum of $I_{OH(peak)}$ for ports P4, P5, P6 and P7 must be 80mA or less.

TIMING REQUIREMENTS (V_{CC}=5V±10%, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

External clock input

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
t_C	External clock input cycle time		125			ns
$t_{W(H)}$	External clock input high-level pulse width		50			ns
$t_{W(L)}$	External clock input low-level pulse width		50			ns
t_r	External clock rise time				20	ns
t_f	External clock fall time				20	ns

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
$t_{SU(P0D-E)}$	Port P0 input setup time		200			ns
$t_{SU(P1D-E)}$	Port P1 input setup time		200			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		200			ns
$t_{SU(P3D-E)}$	Port P3 input setup time		200			ns
$t_{SU(P4D-E)}$	Port P4 input setup time		200			ns
$t_{SU(P5D-E)}$	Port P5 input setup time		200			ns
$t_{SU(P6D-E)}$	Port P6 input setup time		200			ns
$t_{SU(P7D-E)}$	Port P7 input setup time		200			ns
$t_{SU(P8D-E)}$	Port P8 input setup time		200			ns
$t_{h(E-P0D)}$	Port P0 input hold time		0			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(E-P3D)}$	Port P3 input hold time		0			ns
$t_{h(E-P4D)}$	Port P4 input hold time		0			ns
$t_{h(E-P5D)}$	Port P5 input hold time		0			ns
$t_{h(E-P6D)}$	Port P6 input hold time		0			ns
$t_{h(E-P7D)}$	Port P7 input hold time		0			ns
$t_{h(E-P8D)}$	Port P8 input hold time		0			ns

Memory expansion mode and microprocessor mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{SU(P1D-E)}$	Port P1 input setup time		60			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		60			ns
$t_{SU(RDY-\phi_1)}$ *1	$\overline{RDY}$ input setup time		70			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(\phi_1-RDY)}$ *1	$\overline{RDY}$ input hold time		0			ns

*1 : New standard attending on the change in product specification.

Timer B input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		500			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		250			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		250			ns

Timer B input (Pulse period measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		1000*2			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		500*2			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		500*2			ns

Timer B input (Pulse width measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{C(TB)}$	TBi_{IN} input cycle time		1000*2			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		500*2			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		500*2			ns

A-D trigger input

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(AD)}$	$\overline{AD_{TRG}}$ input cycle time (minimum allowable trigger)		2000			ns
$t_{W(ADL)}$	$\overline{AD_{TRG}}$ input low-level pulse width		250			ns

Serial I/O

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{C(CK)}$	CLKi input cycle time		500			ns
$t_{W(CKH)}$	CLKi input high-level pulse width		250			ns
$t_{W(CKL)}$	CLKi input low-level pulse width		250			ns
$t_{d(C-Q)}$ *1	TxDi output delay time				150	ns
$t_{h(C-Q)}$ *1	TxDi hold time		30			ns
$t_{su(D-C)}$ *1	RxDi input setup time		60			ns
$t_{h(C-D)}$ *1	RxDi input hold time		90			ns

External interrupt $\overline{INTi}$ input

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{W(INH)}$	$\overline{INTi}$ input high-level pulse width		250			ns
$t_{W(INL)}$	$\overline{INTi}$ input low-level pulse width		250			ns

*1 : New standard attending on the change in product specification.
*2 : Changed standard attending on the change in product specification.

Memory expansion mode and microprocessor mode (when wait bit = "0", and external memory area is accessed)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 55	350			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				120	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				5	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		350			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				120	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				5	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		350			ns
$t_{d(E-HLDA)}$	$\overline{HLDA}$ output delay time				100	ns
$t_{d(ALE-E)}$	ALE output delay time		4*2			ns
$t_{W(ALE)}$	ALE pulse width		350			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		350			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		350			ns
$t_{d(E-\phi_1)}$*1	ϕ_1 output delay time		0		30	ns
$t_{h(E-P0A)}$	Port P0 address hold time		50*2			ns
$t_{h(ALE-P1A)}$*1	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		50*2			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		50*2			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		50*2			ns
$t_{h(ALE-P2A)}$*1	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		50*2			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		50*2			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$*1	$\overline{E}$ pulse width		470			ns

*1 : New standard attending on the change in product specification
*2 : Changed standard attending on the change in product specification.

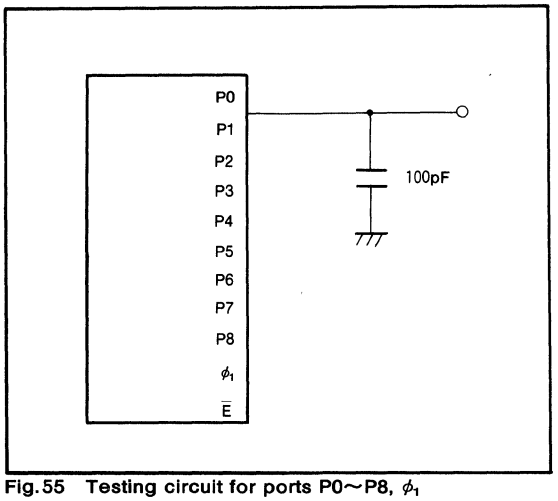


Fig. 55 Testing circuit for ports P0~P8, ϕ_1

TIMING REQUIREMENTS (V_{CC}=5V±10%, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

External clock input

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
t_C	External clock input cycle time		62			ns
$t_{W(H)}$	External clock input high-level pulse width		25			ns
$t_{W(L)}$	External clock input low-level pulse width		25			ns
t_r	External clock rise time				10	ns
t_f	External clock fall time				10	ns

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{SU(P0D-E)}$	Port P0 input setup time		100			ns
$t_{SU(P1D-E)}$	Port P1 input setup time		100			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		100			ns
$t_{SU(P3D-E)}$	Port P3 input setup time		100			ns
$t_{SU(P4D-E)}$	Port P4 input setup time		100			ns
$t_{SU(P5D-E)}$	Port P5 input setup time		100			ns
$t_{SU(P6D-E)}$	Port P6 input setup time		100			ns
$t_{SU(P7D-E)}$	Port P7 input setup time		100			ns
$t_{SU(P8D-E)}$	Port P8 input setup time		100			ns
$t_{h(E-P0D)}$	Port P0 input hold time		0			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(E-P3D)}$	Port P3 input hold time		0			ns
$t_{h(E-P4D)}$	Port P4 input hold time		0			ns
$t_{h(E-P5D)}$	Port P5 input hold time		0			ns
$t_{h(E-P6D)}$	Port P6 input hold time		0			ns
$t_{h(E-P7D)}$	Port P7 input hold time		0			ns
$t_{h(E-P8D)}$	Port P8 input hold time		0			ns

Memory expansion mode and microprocessor mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{SU(P1D-E)}$	Port P1 input setup time		45			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		45			ns
$t_{SU(RDY-\phi_1)}$ *1	$\overline{RDY}$ input setup time		60			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(\phi_1-RDY)}$ *1	$\overline{RDY}$ input hold time		0			ns

*1 : New standard attending on the change in product specification.

Timer B input (Cont input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		250			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		125			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		125			ns

Timer B input (Pulse period measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		500*2			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		250*2			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		250*2			ns

Timer B input (Pulse width measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		500*2			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		250*2			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		250*2			ns

A-D trigger input

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max	
$t_{C(AD)}$	$\overline{AD_{TRG}}$ input cycle time (minimum allowable trigger)		1000			ns
$t_{W(ADL)}$	$\overline{AD_{TRG}}$ input low-level pulse width		125			ns

Serial I/O

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{C(CK)}$	CLKi input cycle time		250			ns
$t_{W(CKH)}$	CLKi input high-level pulse width		125			ns
$t_{W(CKL)}$	CLKi input low-level pulse width		125			ns
$t_{d(C-Q)}$*1	TxDi output delay time				90	ns
$t_{h(C-Q)}$*1	TxDi hold time		30			ns
$t_{su(D-C)}$*1	RxDi input setup time		30			ns
$t_{h(C-D)}$*1	RxDi input hold time		90			ns

External interrupt $\overline{INTi}$ input

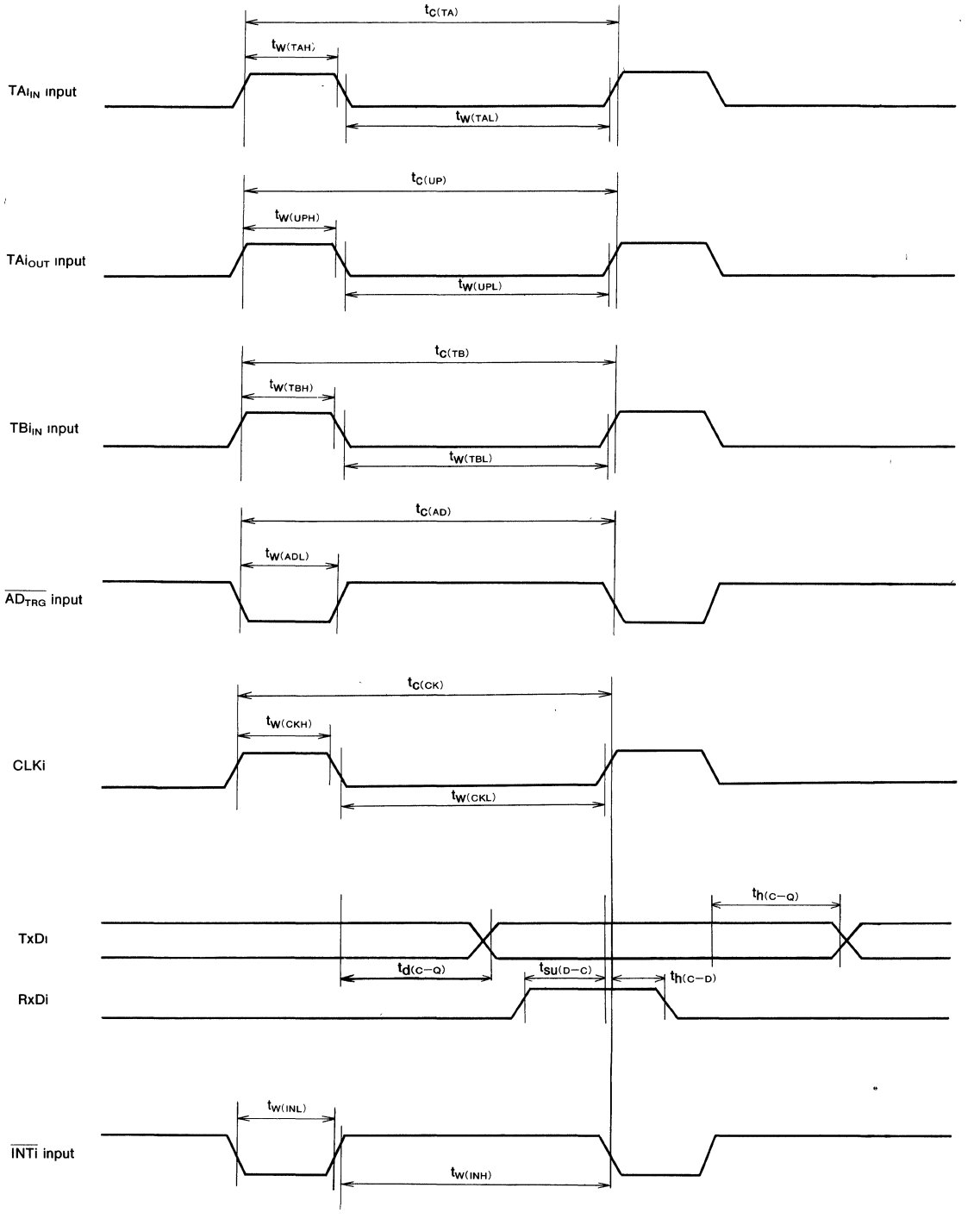
Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{W(INH)}$	$\overline{INTi}$ input high-level pulse width		250			ns
$t_{W(INL)}$	$\overline{INTi}$ input low-level pulse width		250			ns

*1 : New standard attending on the change in product specification.
*2 : Changed standard attending on the change in product specification

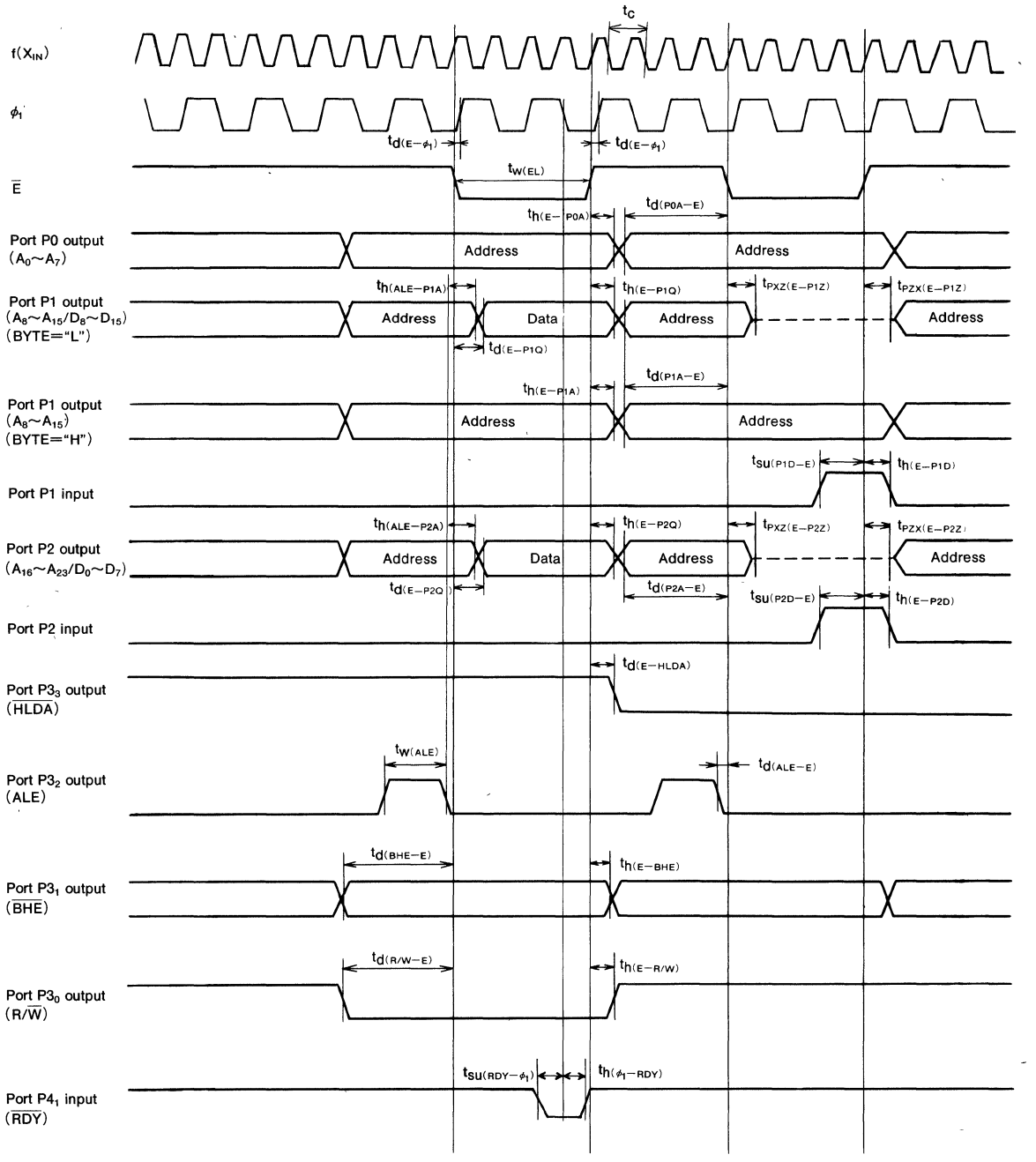
Memory expansion mode and microprocessor mode (when wait bit = "0", and external memory area is accessed)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 55	155			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				80	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				5	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		155			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				80	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				5	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		155			ns
$t_{d(E-HLDA)}$	$\overline{HLDA}$ output delay time				50	ns
$t_{d(ALE-E)}$	ALE output delay time		4*2			ns
$t_{W(ALE)}$	ALE pulse width		165			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		155			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		155			ns
$t_{d(E-\phi_1)}$*1	ϕ_1 output delay time		0		20	ns
$t_{h(E-P0A)}$	Port P0 address hold time		25*2			ns
$t_{h(ALE-P1A)}$*1	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		25*2			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		25*2			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		25*2			ns
$t_{h(ALE-P2A)}$*1	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		25*2			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		25*2			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$*1	$\overline{E}$ pulse width		220			ns

*1 : New standard attending on the change in product specification.
*2 : Changed standard attending on the change in product specification.


$t_{C(TA)}$
$t_{W(TAH)}$
TAi_{IN} input
$t_{W(TAL)}$
$t_{C(UP)}$
$t_{W(UPH)}$
TAi_{OUT} input
$t_{W(UPL)}$
$t_{C(TB)}$
$t_{W(TBH)}$
TBi_{IN} input
$t_{W(TBL)}$
$t_{C(AD)}$
$t_{W(ADL)}$
$\overline{AD_{TRG}}$ input
$t_{C(CK)}$
$t_{W(CKH)}$
CLKi
$t_{W(CKL)}$
$t_{h(C-Q)}$
TxDi
$t_{d(C-Q)}$
$t_{su(D-C)}$
$t_{h(C-D)}$
RxDi
$t_{W(INL)}$
$\overline{INTi}$ input
$t_{W(INH)}$

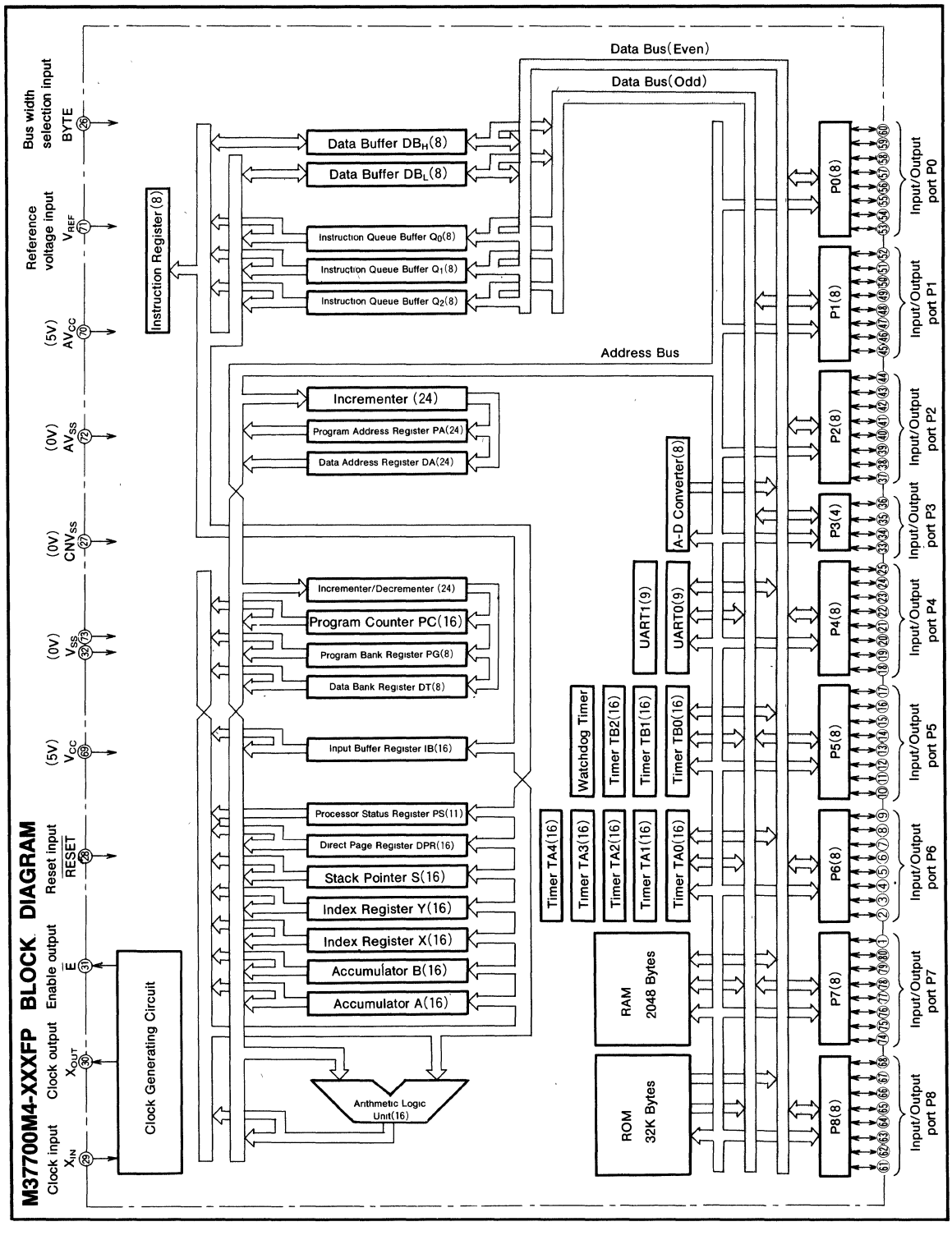
Memory expansion mode and microprocessor mode (when wait bit="0", and external memory area is accessed)



Test conditions

- V_{CC}= 5 V±10%
- Output timing voltage : V_{OL}=0.8V, V_{OH}=2.0V
- Ports P1,P2 input : V_{IL}=0.8V, V_{IH}=2.5V
- Port $P4_1$ input : V_{IL}=1.0V, V_{IH}=4.0V

M37700M4-XXXFP BLOCK DIAGRAM



MITSUBISHI ELECTRIC

M37701M2-XXXSP, M37701M2AXXXSP
M37701SSP, M37701SASP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER

DESCRIPTION

The M37701M2-XXXSP, M37701M2AXXXSP, M37701SSP and M37701SASP are single-chip microcomputers designed with high-performance CMOS silicon gate technology. These are housed in a 64-pin shrink plastic molded DIP. These single-chip microcomputers have a large 16M bytes address space, three instruction queue buffers, and two data buffers for high-speed instruction execution. The CPU is a 16-bit parallel processor that can also be switched to perform 8-bit parallel processing. These microcomputers are suitable for office, business and industrial equipment controller that require high-speed processing of large data.

The differences between M37701M2-XXXSP, M37701M2AXXXSP, M37701SSP and M37701SASP are the ROM size and the external clock input frequency as shown below. Therefore, the following descriptions will be for the M37701M2-XXXSP unless otherwise noted.

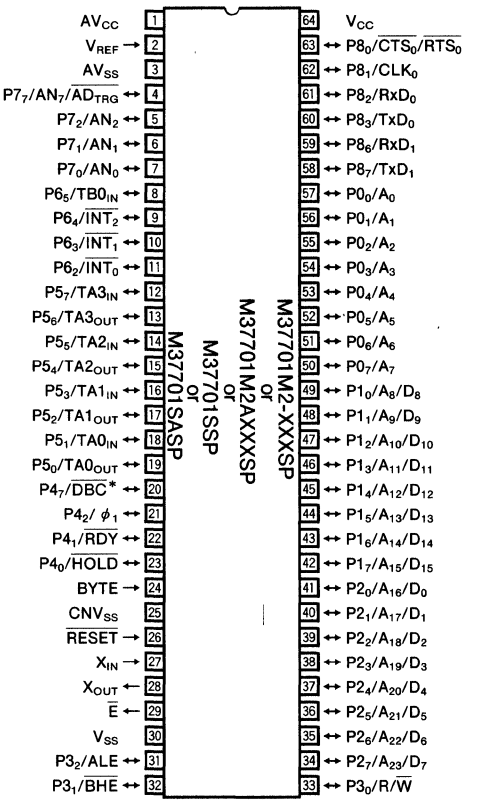
Type name	ROM size	External clock input frequency
M37701M2-XXXSP	16K bytes	8 MHz
M37701M2AXXXSP	16K bytes	16MHz
M37701SSP	External	8 MHz
M37701SASP	External	16MHz

The M37701M2-XXXSP cuts down the pins of M37700M2-XXXFP. Refer to the section on M37700M2-XXXFP for the functional differences.

DISTINCTIVE FEATURES

- Number of basic instructions 103
- Memory size ROM ..... 16K bytes
  RAM ..... 512 bytes
- Instruction execution time
  M37701M2-XXXSP, M37701SSP
  (The fastest instruction at 8 MHz frequency) 500ns
  M37701M2AXXXSP, M37701SASP
  (The fastest instruction at 16 MHz frequency) 250ns
- Single power supply ..... 5V±10%
- Low power dissipation (at 8 MHz frequency)
  ..... 30mW (Typ.)
- Interrupts ..... 19 types 7 levels
- Multiple function 16-bit timer ..... 5+3
- UART (may also be synchronous) 2
- 8-bit A-D converter ..... 4-channel inputs
- 12-bit watchdog timer
- Programmable input/output
  (ports P0, P1, P2, P3, P4, P5, P6, P7, P8) 53

PIN CONFIGURATION (TOP VIEW)



Outline 64P4B

* Used in the evaluation chip mode only

APPLICATION

Control devices for office equipment such as copiers, printers, typewriters, facsimiles, word processors, and personal computers

Control devices for industrial equipment such as ME, NC, communication, and measuring instruments

FUNCTIONS OF M37701M2-XXXSP

Parameter		Functions
Number of basic instructions		103
Instruction execution time	M37701M2-XXXSP, M37701SSP	500ns (the fastest instructions, at 8MHz frequency)
	M37701M2AXXXSP, M37701SASP	250ns (the fastest instructions, at 16MHz frequency)
Memory size	ROM	16K bytes
	RAM	512 bytes
Input/Output ports	P0, P1, P2, P5	8-bit× 4
	P8	6-bit× 1
	P4, P6, P7	4-bit× 3
	P3	3-bit× 1
Multi-function timers	TA0, TA1, TA2, TA3, TA4	16-bit× 5 (4 Input/Output functions)
	TB0, TB1, TB2	16-bit× 3 (1 Input function)
Serial I/O		UART×2(One can be set clock synchronous serial I/O)
A-D converter		8-bit× 1 (4 channels)
Watchdog timer		12-bit× 1
Interrupts		3 external types, 16 internal types (Each interrupt can be set the priority levels to 0 ~ 7)
Clock generating circuit		Built-in(externally connected to a ceramic resonator or quartz crystal resonator)
Supply voltage		5 V±10%
Power dissipation		30mW(at external 8 MHz frequency)
Input/Output characteristic	Input/Output voltage	5 V
	Output current	5 mA
Memory expansion		Maximum 16M bytes
Operating temperature range		−10~70℃
Device structure		CMOS high-performance silicon gate process
Package		64-pin shrink plastic molded DIP

BASIC FUNCTION BLOCKS

The functional differences between the M7701M2-XXXSP and M37700M2-XXXFP are described below. The M37701M2-XXXSP has the same functions as the M37700M2-XXXFP, except these points. Refer to the section on the M37700M2-XXXFP.

A-D CONVERTER

Analog signals are input through four channels, AN_0, AN_1, AN_2 and AN_7 In one-shot mode and repeat mode, select one on AN_0, AN_1, AN_2, and AN_7 as analog input by the analog input selection bits (bits 2, 1 and 0) of A-D control register. Set the bits of the directional registers for ports corresponding to analog input channels AN_3, AN_4, AN_5 and AN_6 not having pins to "1" (output mode) and output "0" to the ports.

In the single sweep mode and repeat sweep mode, the M37701M2-XXXSP operates the same as the M37700M2-XXXFP. Set the directional register bits of ports corresponding to AN0, AN1, AN2, and AN7 to "0" (input mode), and the bits of the directional registers for ports corresponding to AN_3, AN_4, AN_5 and AN_6 not having pins to "1" (output mode), and output "0" to the ports. In the single sweep mode and repeat sweep mode, the contents of A-D register bits corresponding to analog input channels AN_3, AN_4, AN_5, and AN_6 not having pins are undefined.

TIMER

Since timer A4 has no input/output function and timer B1, B2 have no input function, timers A4, B1 and B2 operate only in timer mode. Therefore, only clock source can be selected by the bits 7 and 6 of timer mode register for each of timers A4, B1 and B2. The bits of timer mode register must be "0" except for the clock source selection bits. Other timers A0, A1, A2, A3 and B0 have the same functions as the M37700M2-XXXFP.

SERIAL I/O

UART1 has only the asynchronous serial communication function and no clock synchronous serial communication function. Therefore, do not select the clock synchronous serial communication function ("001") by the serial communication method selection bits (bits 2, 1 and 0) of UART1 transmit/receive mode register. Since UART1 does not have the functions of $\overline{CTS}$ and $\overline{RTS}$, the $\overline{CTS}$ and $\overline{RTS}$ selection bit (bit 2) of UART1 transmit/receive control register 0 must always be "1". UART0 has the same function as the M37700M2-XXXFP.

INPUT/OUTPUT PINS

The port registers and directional registers for ports P4, P6, P7 and P8 have eigtht bits, the directional register bits having no pins must always be set to the output mode. Since port $P3_3$ is not available as a pin although it has port register and directional register, port $P3_3$ must be set to the output mode.

ADDRESSING MODES

The M37701M2-XXXSP has 28 powerful addressing modes. Refer to the Series MELPS 7700 addressing mode description for the details of each addressing mode.

MACHINE INSTRUCTION LIST

The M37701M2-XXXSP has 103 machine instructions. Refer to the Series MELPS 7700 machine instruction list for details.

DATA REQUIRED FOR MASK ORDERING

Please send the following data for mask orders.

(1) M37701M2-XXXSP mask ROM order confirmation form
(2) Mark specification form for 64P4B
(3) ROM data (EPROM 3 sets)

Table 1. The functional differences between the M37701M2-XXXSP and M37700M2-XXXFP

Parameter	M37701M2-XXXSP	M37700M2-XXXFP
Input/Output ports	P0~P2, P5 ····· 8-bit× 4 P8 ····· 6-bit× 1 P4, P6, P7 ····· 4-bit× 3 P3 ····· 3-bit× 1 (without $\overline{HLDA}$)	P0~P2, P4~P8 ····· 8-bit× 8 P3 ····· 4-bit× 1 (with $\overline{HLDA}$)
Timer	Timer A ····· with Input/Output ports 16-bit× 4 only timer mode 16-bit× 1	Timer A ····· with Input/Output ports 16-bit× 5
	Timer B ····· with Input ports 16-bit× 1 only timer mode 16-bit× 2	Timer B ····· with Input ports 16-bit× 3
Serial I/O	(UART or clock synchronous serial I/O)× 1 UART× 1	(UART or clock synchronous serial I/O)× 2
A-D converter	8-bit× 1 (4 channels)	8-bit× 1 (8 channels)

M37701M2-XXXSP

ELECTRICAL CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min	Limits Typ	Limits Max.	Unit
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_7$, $P6_2$~$P6_5$, $P7_0$~$P7_2$, $P7_7$, $P8_0$~$P8_3$, $P8_6$, $P8_7$	I_{OH}=−10mA	3			V
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$	I_{OH}=−400μA	4.7			V
V_{OH}	High-level output voltage $P3_2$	I_{OH}=−10mA	3.1			V
		I_{OH}=−400μA	4.8			
V_{OH}	High-level output voltage $\overline{E}$	I_{OH}=−10mA	3.4			V
		I_{OH}=−400μA	4.8			
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_7$, $P6_2$~$P6_5$, $P7_0$~$P7_2$, $P7_7$, $P8_0$~$P8_3$, $P8_6$, $P8_7$	I_{OL}=10mA			2	V
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$	I_{OL}=2mA			0.45	V
V_{OL}	Low-level output voltage $P3_2$	I_{OL}=10mA			1.9	V
		I_{OL}=2mA			0.43	
V_{OL}	Low-level output voltage $\overline{E}$	I_{OL}=10mA			1.6	V
		I_{OL}=2mA			0.4	
$V_{T+}-V_{T-}$	Hysteresis $\overline{HOLD}$, $\overline{RDY}$, $TA0_{IN}$~$TA3_{IN}$, $TB0_{IN}$, $\overline{INT_0}$~$\overline{INT_2}$, $\overline{AD_{TRG}}$, $\overline{CTS_0}$, CLK_0		0.4		1	V
$V_{T+}-V_{T-}$	Hysteresis $\overline{RESET}$		0.2		0.5	V
$V_{T+}-V_{T-}$	Hysteresis X_{IN}		0.1		0.3	V
I_{IH}	High-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_2$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_7$, $P6_2$~$P6_5$, $P7_0$~$P7_2$, $P7_7$, $P8_0$~$P8_3$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=5V			5	μA
I_{IL}	Low-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_2$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_7$, $P6_2$~$P6_5$, $P7_0$~$P7_2$, $P7_7$, $P8_0$~$P8_3$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=0V			−5	μA
V_{RAM}	RAM hold voltage	When clock is stopped	2			V
I_{CC}	Power supply current	In single-chip mode output only pin is open and other pins are V_{SS} during reset — f(X_{IN})=8MHz, square waveform		6	12	mA
		In single-chip mode output only pin is open and other pins are V_{SS} during reset — T_a=25℃ when clock is stopped			1	μA
		In single-chip mode output only pin is open and other pins are V_{SS} during reset — T_a=70℃ when clock is stopped			10	

A-D CONVERTER CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min.	Limits Typ	Limits Max	Unit
—	Resolution	V_{REF}=V_{CC}			8	Bits
—	Absolute accuracy	V_{REF}=V_{CC}			±3	LSB
R_{LADDER}	Ladder resistance	V_{REF}=V_{CC}	2		10	kΩ
t_{CONV}	Conversion time		28.5			μs
V_{REF}	Reference voltage		2		V_{CC}	V
V_{IA}	Analog input voltage		0		V_{REF}	V

Timer A input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		250			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		125			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		125			ns

Timer A input (Gating input in timer mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{C(TA)}$	TAi_{IN} input cycle time		1000			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		500			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		500			ns

Timer A input (External trigger input in one-shot pulse mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		500			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		250			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		250			ns

Timer A input (External trigger input in pulse width modulation mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		250			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		250			ns

Timer A input (Up-down input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
$t_{C(UP)}$	TAi_{OUT} input cycle time		1000			ns
$t_{W(UPH)}$	TAi_{OUT} input high-level pulse width		500			ns
$t_{W(UPL)}$	TAi_{OUT} input low-level pulse width		500			ns

SWITCHING CHARACTERISTICS (V_{CC}=5V±10%, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{d(E-P0Q)}$	Port P0 data output delay time	Figure 1			200	ns
$t_{d(E-P1Q)}$	Port P1 data output delay time				200	ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				200	ns
$t_{d(E-P3Q)}$	Port P3 data output delay time				200	ns
$t_{d(E-P4Q)}$	Port P4 data output delay time				200	ns
$t_{d(E-P5Q)}$	Port P5 data output delay time				200	ns
$t_{d(E-P6Q)}$	Port P6 data output delay time				200	ns
$t_{d(E-P7Q)}$	Port P7 data output delay time				200	ns
$t_{d(E-P8Q)}$	Port P8 data output delay time				200	ns

Memory expansion mode and microprocessor mode (when wait bit = "1")

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 1	100			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				120	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				40	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		100			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				120	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				40	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		100			ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		100			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		100			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		100			ns
$t_{d(E-\phi 1)}$	ϕ_1 output delay time		0		30	ns
$t_{h(E-P0A)}$	Port P0 address hold time		50			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		50			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		50			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		50			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		50			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		50			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		220			ns

M37701M2AXXXSP

ELECTRICAL CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_7$, $P6_2$~$P6_5$, $P7_0$~$P7_2$, $P7_7$, $P8_0$~$P8_3$, $P8_6$, $P8_7$	I_{OH}=−10mA	.3			V
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$	I_{OH}=−400μA	4.7			V
V_{OH}	High-level output voltage $P3_2$	I_{OH}=−10mA	3.1			V
		I_{OH}=−400μA	4.8			
V_{OH}	High-level output voltage $\overline{E}$	I_{OH}=−10mA	3.4			V
		I_{OH}=−400μA	4.8			
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_7$, $P6_2$~$P6_5$, $P7_0$~$P7_2$, $P7_7$, $P8_0$~$P8_3$, $P8_6$, $P8_7$	I_{OL}=10mA			2	V
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$	I_{OL}=2mA			0.45	V
V_{OL}	Low-level output voltage $P3_2$	I_{OL}=10mA			1.9	V
		I_{OL}=2mA			0.43	
V_{OL}	Low-level output voltage $\overline{E}$	I_{OL}=10mA			1.6	V
		I_{OL}=2mA			0.4	
V_{T+}−V_{T-}	Hysteresis $\overline{HOLD}$, $\overline{RDY}$, $TA0_{IN}$~$TA3_{IN}$, $TB0_{IN}$, $\overline{INT_0}$~$\overline{INT_2}$, $\overline{AD_{TRG}}$, $\overline{CTS_0}$, CLK_0		0.4		1	V
V_{T+}−V_{T-}	Hysteresis $\overline{RESET}$		0.2		0.5	V
V_{T+}−V_{T-}	Hysteresis X_{IN}		0.1		0.3	V
I_{IH}	High-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_2$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_7$, $P6_2$~$P6_5$, $P7_0$~$P7_2$, $P7_7$, $P8_0$~$P8_3$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=5V			5	μA
I_{IL}	Low-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_2$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_7$, $P6_2$~$P6_5$, $P7_0$~$P7_2$, $P7_7$, $P8_0$~$P8_3$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=0V			−5	μA
V_{RAM}	RAM hold voltage	When clock is stopped	2			V
I_{CC}	Power supply current	In single-chip mode output only pin is open and other pins are V_{SS} during reset. — f(X_{IN})=16MHz, square waveform		12	24	mA
		T_a=25℃ when clock is stopped			1	μA
		T_a=70℃ when clock is stopped.			10	

A-D CONVERTER CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
—	Resolution	V_{REF}=V_{CC}			8	Bits
—	Absolute accuracy	V_{REF}=V_{CC}			±3	LSB
R_{LADDER}	Ladder resistance	V_{REF}=V_{CC}	2		10	kΩ
t_{CONV}	Conversion time		14.25			μs
V_{REF}	Reference voltage		2		V_{CC}	V
V_{IA}	Analog input voltage		0		V_{REF}	V

Timer A input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{C(TA)}$	TAi_{IN} input cycle time		125			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		62			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		62			ns

Timer A input (Gating input in timer mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{C(TA)}$	TAi_{IN} input cycle time		500			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		250			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		250			ns

Timer A input (External trigger input in one-shot pulse mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{C(TA)}$	TAi_{IN} input cycle time		250			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		125			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		125			ns

Timer A input (External trigger input in pulse width modulation mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		125			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		125			ns

Timer A input (Up-down input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{C(UP)}$	TAi_{OUT} input cycle time		500			ns
$t_{W(UPH)}$	TAi_{OUT} input high-level pulse width		250			ns
$t_{W(UPL)}$	TAi_{OUT} input low-level pulse width		250			ns

SWITCHING CHARACTERISTICS (V_{CC}=5V±10%, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

Single-chip mode

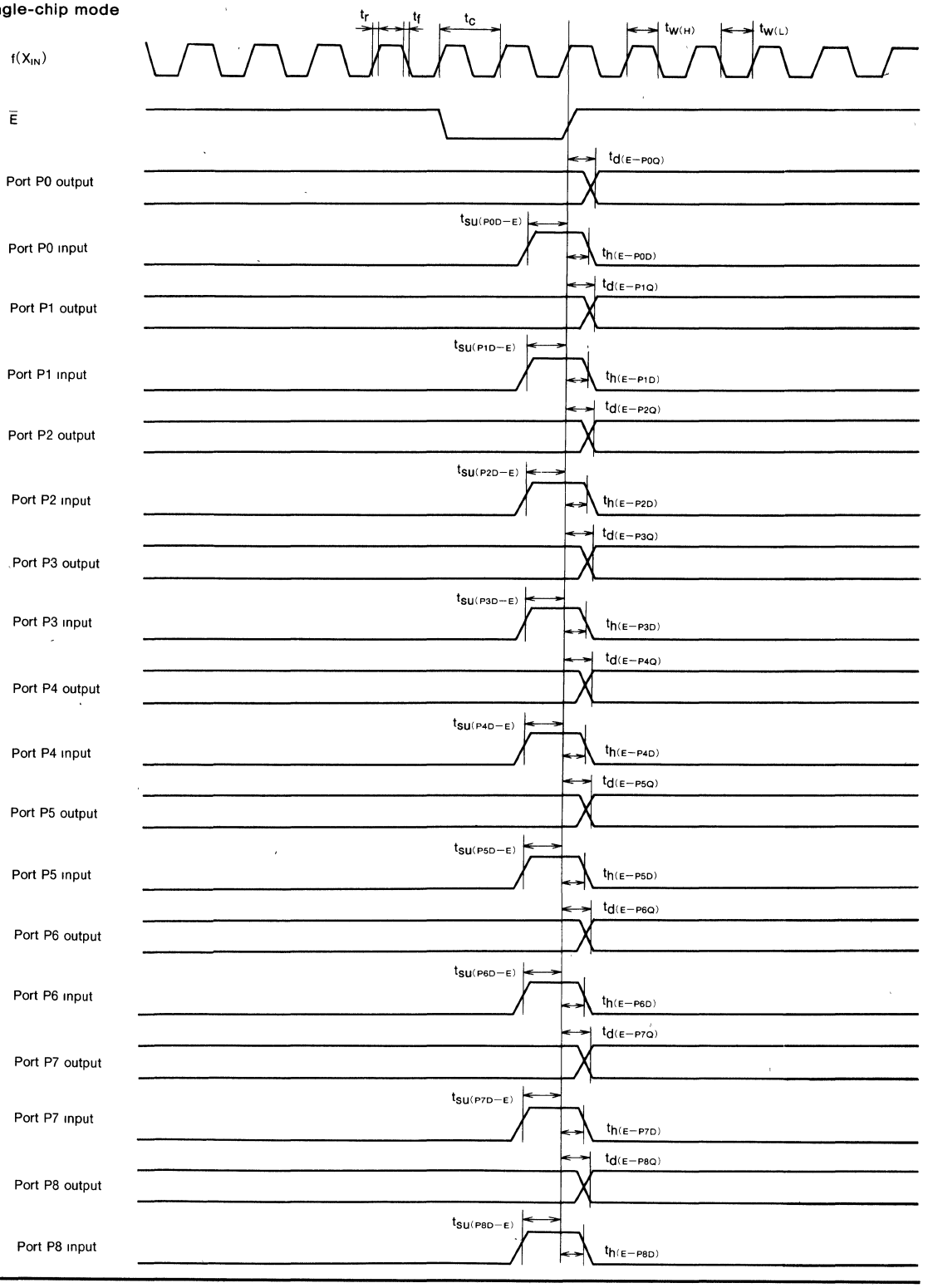
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{d(E-P0Q)}$	Port P0 data output delay time	Figure 1			100	ns
$t_{d(E-P1Q)}$	Port P1 data output delay time				100	ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				100	ns
$t_{d(E-P3Q)}$	Port P3 data output delay time				100	ns
$t_{d(E-P4Q)}$	Port P4 data output delay time				100	ns
$t_{d(E-P5Q)}$	Port P5 data output delay time				100	ns
$t_{d(E-P6Q)}$	Port P6 data output delay time				100	ns
$t_{d(E-P7Q)}$	Port P7 data output delay time				100	ns
$t_{d(E-P8Q)}$	Port P8 data output delay time				100	ns

Memory expansion mode and microprocessor mode (when wait bit = "1")

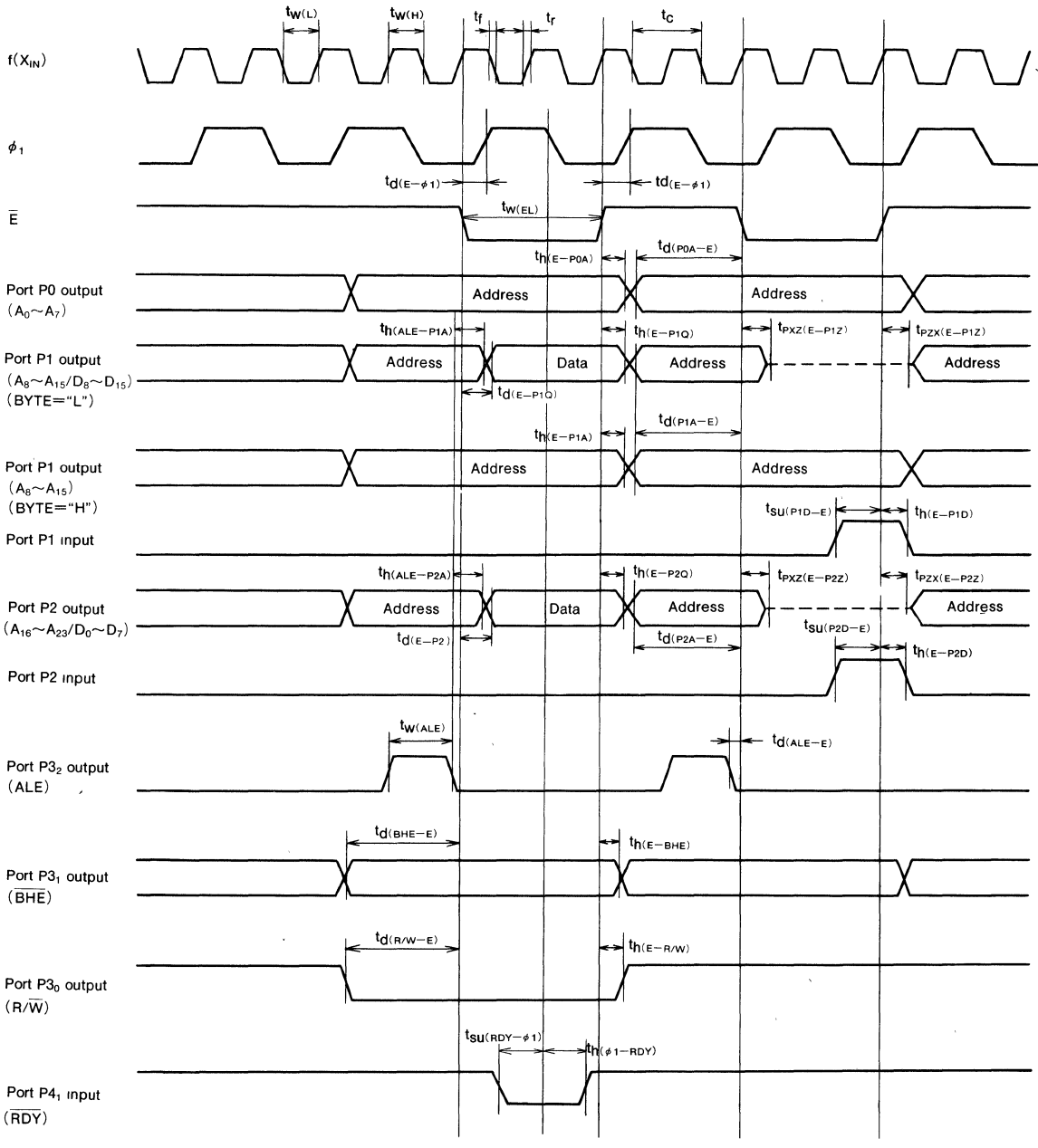
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max.	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 1	30			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				80	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				40	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		30			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				80	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				40	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		30			ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		40			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		30			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		30			ns
$t_{d(E-\phi 1)}$	ϕ_1 output delay time		0		20	ns
$t_{h(E-P0A)}$	Port P0 address hold time		25			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		25			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		25			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		25			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		25			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		25			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		95			ns

TIMING DIAGRAM

Single-chip mode



Memory expansion mode and microprocessor mode (When wait bit="1")



Test conditions

- V_{CC}= 5 V±10%
- Output timing voltage : V_{OL}=0.8V, V_{OH}=2.0V
- Ports P1,P2 input : V_{IL}=0.8V, V_{IH}=2.5V
- Port $P4_1$ input : V_{IL}=1.0V, V_{IH}=4.0V

M37701M4-XXXSP, M37701M4AXXXSP
M37701S4SP, M37701S4ASP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER

DESCRIPTION

The M37701M4-XXXSP, M37701M4AXXXSP, M37701S4SP and M37701S4ASP are single-chip microcomputers designed with high-performance CMOS silicon gate technology. These are housed in a 64-pin shrink plastic molded DIP. These single-chip microcomputers have a large 16M bytes address space, three instruction queue buffers, and two data buffers for high-speed instruction execution. The CPU is a 16-bit parallel processor that can also be switched to perform 8-bit parallel processing. These microcomputers are suitable for office, business and industrial equipment controller that require high-speed processing of large data.

The differences between M37701M4-XXXSP, M37701M4AXXXSP, M37701S4SP and M37701S4ASP are the ROM size and the external clock input frequency as shown below. Therefore, the following descriptions will be for the M37701M4-XXXSP unless otherwise noted.

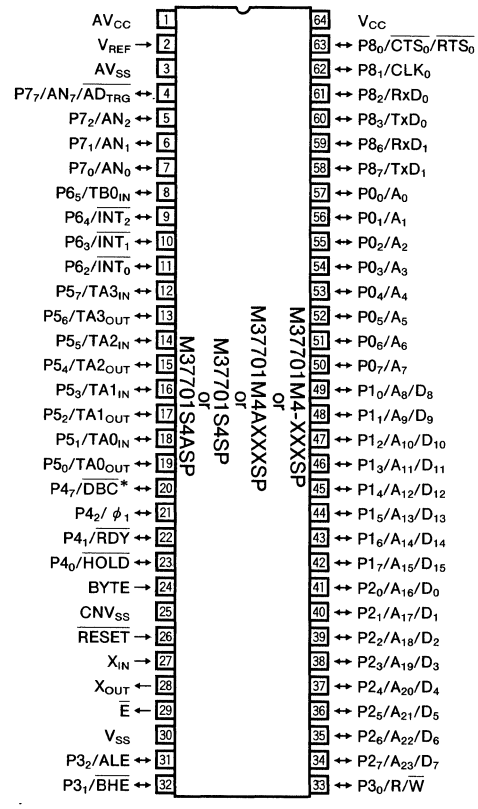
Type name	ROM size	External clock input frequency
M37701M4-XXXSP	32K bytes	8 MHz
M37701M4AXXXSP	32K bytes	16MHz
M37701S4SP	External	8 MHz
M37701S4ASP	External	16MHz

The M37701M4-XXXSP has the same functions as the M37701M2-XXXSP except for the memory size.

DISTINCTIVE FEATURES

- Number of basic instructions 103
- Memory size ROM ..... 32K bytes
  RAM ..... 2048 bytes
- Instruction execution time
  M37701M4-XXXSP, M37701S4SP
  (The fastest instruction at 8 MHz frequency) 500ns
  M37701M4AXXXSP, M37701S4ASP
  (The fastest instruction at 16 MHz frequency) 250ns
- Single power supply ..... 5V±10%
- Low power dissipation (at 8 MHz frequency)
  ..... 30mW (Typ.)
- Interrupts ..... 19 types 7 levels
- Multiple function 16-bit timer ..... 5+3
- UART (may also be synchronous) 2
- 8-bit A-D converter ..... 4-channel inputs
- 12-bit watchdog timer
- Programmable input/output
  (ports P0, P1, P2, P3, P4, P5, P6, P7, P8) 53

PIN CONFIGURATION (TOP VIEW)



Outline 64P4B

* Used in the evaluation chip mode only

APPLICATION

Control devices for office equipment such as copiers, printers, typewriters, facsimiles, word processors, and personal computers

Control devices for industrial equipment such as ME, NC, communication, and measuring instruments

THE FUNCTIONS AND CHARACTERISTICS

The M37701M4-XXXSP has the same functions and characteristics as the M37701M2-XXXSP except for the ROM and RAM size. Refer to the section on the M37701M2-XXXSP.

DATA REQUIRED FOR MASK ORDERING

Please send the following data for mask orders.

(1) M37701M4-XXXSP mask ROM order confirmation form
(2) Mark specification form for 64P4B
(3) ROM data (EPROM 3 sets)

M37704M2-XXXFP, M37704M2AXXXFP
M37704S1FP, M37704S1AFP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER

DESCRIPTION

The M37704M2-XXXFP, M37704M2AXXXFP, M37704S1FP and M37704S1AFP are single-chip microcomputers designed with high-performance CMOS silicon gate technology. These are housed in a 80-pin plastic molded QFP. These single-chip microcomputers have a large 16M bytes address space, three instruction queue buffers, and two data buffers for high-speed instruction execution. The CPU is a 16-bit parallel processor that can also be switched to perform 8-bit parallel processing. These microcomputers are suitable for office, business, and industrial equipment controller that require high-speed processing of large data. Also, the incorporated motor control circuit makes these microcomputers suitable for control of equipment that requires motor control.

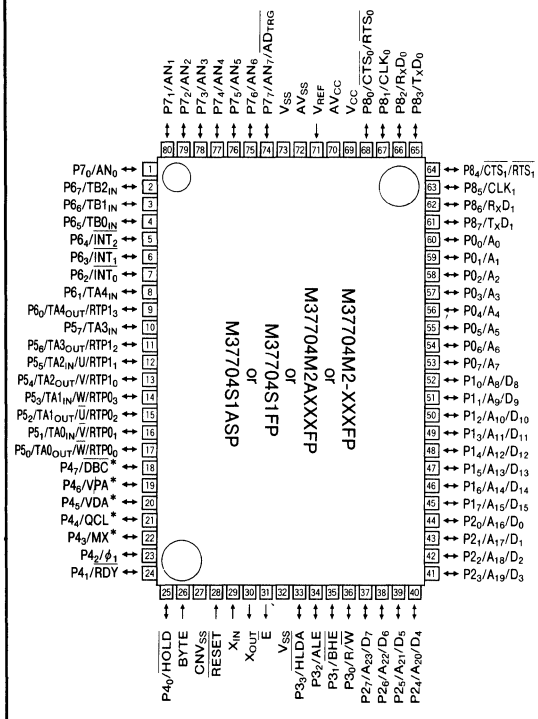
The differences between M37704M2-XXXFP, M37704M2AXXXFP, M37704S1FP and M37704S1AFP are the ROM size and the external clock input frequency as shown below. Therefore, the following descriptions will be for the M37704M2-XXXFP unless otherwise noted.

Type name	ROM size	External clock input frequency
M37704M2-XXXFP	16K bytes	8 MHz
M37704M2AXXXFP	16K bytes	16MHz
M37704S1FP	External	8 MHz
M37704S1AFP	External	16MHz

DISTINCTIVE FEATURES

- Number of basic instructions 103
- Memory size ROM ..... 16K bytes
  RAM ..... 512 bytes
- Instruction execution time
  M37704M2-XXXFP, M37704S1FP
  (The fastest instruction at 8 MHz frequency) 500ns
  M37704M2AXXXFP, M37704S1AFP
  (The fastest instruction at 16 MHz frequency) 250ns
- Single power supply ..... 5V±10%
- Low power dissipation (at 8 MHz frequency)
  ..... 30mW (Typ.)
- Interrupts ..... 19 types 7 levels
- Multiple function 16-bit timer ..... 5+3
  (Three-phase motor drive waveform or pulse motor drive waveform can be output.)
- UART (may also be synchronous) 2
- 8-bit A-D converter ..... 8-channel inputs
- 12-bit watchdog timer
- Programmable input/output
  (ports P0, P1, P2, P3, P4, P5, P6, P7, P8) 68

PIN CONFIGURATION (TOP VIEW)



Outline 80P6N

* Used in the evaluation chip mode only

APPLICATION

Control devices for office equipment such as copiers, printers, typewriters, facsimiles, word processors, and personal computers

Control devices for industrial equipment such as ME, NC, general purpose inverter and measuring instruments.

FUNCTIONS OF M37704M2-XXXFP

Parameter		Functions
Number of basic instructions		103
Instruction execution time	M37704M2-XXXFP, M37704S1FP	500ns (the fastest instructions, at 8MHz frequency)
	M37704M2AXXXFP, M37704S1AFP	250ns (the fastest instructions, at 16MHz frequency)
Memory size	ROM	16K bytes
	RAM	512 bytes
Input/Output ports	P0～P2, P4～P8	8-bit×8
	P3	4-bit×1
Multi-function timers	TA0, TA1, TA2, TA3, TA4	16-bit×5
	TB0, TB1, TB2	16-bit×3
Serial I/O		(UART or clock synchronous serial I/O)×2
A-D converter		8-bit×1 (8 channels)
Watchdog timer		12-bit×1
Dead-time timer		8-bit×3
Interrupts		3 external types, 16 internal types (Each interrupt can be set the priority levels to 0～7)
Clock generating circuit		Built-in(externally connected to a ceramic resonator or quartz crystal resonator)
Supply voltage		5 V±10%
Power dissipation		30mW(at external 8 MHz frequency)
Input/Output characteristic	Input/Output voltage	5 V
	Output current	5 mA
Memory expansion		Maximum 16M bytes
Operating temperature range		−10～70℃
Device structure		CMOS high-performance silicon gate process
Package		80-pin plastic molded QFP

BASIC FUNCTION BLOCKS

The M37704M2-XXXFP contains the following devices on a single chip: ROM and RAM for storing instructions and data, CPU for processing, bus interface unit (which controls instruction prefetch and data read/write between CPU and memory), timers, UART, A-D converter, and other peripheral devices such as I/O ports. Each of these devices are described below.

MEMORY

The memory map is shown in Figure 1. The address space is 16M bytes from addresses 0_{16} to $FFFFFF_{16}$. The address space is divided into 64K bytes units called banks. The banks are numbered from 0_{16} to FF_{16}.

Built-in ROM, RAM and control registers for built-in peripheral devices are assigned to bank 0.

The 16K bytes area from addresses $C000_{16}$ to $FFFF_{16}$ is the built-in ROM. Addresses $FFD6_{16}$ to $FFFF_{16}$ are the RESET and interrupt vector addresses and contain the interrupt vectors. Refer to the section on interrupts for details.

The 512 bytes area from addresses 80_{16} to $27F_{16}$ contains the built-in RAM. In addition to storing data, the RAM is used as stack during a subroutine call, or interrupts.

Assigned to addresses 0_{16} to $7F_{16}$ are peripheral devices such as I/O ports, A-D converter, UART, timer, and interrupt control registers.

A 256 bytes direct page area can be allocated anywhere in bank 0 using the direct page register DPR. In direct page addressing mode, the memory in the direct page area can be accessed with two words thus reducing program steps.

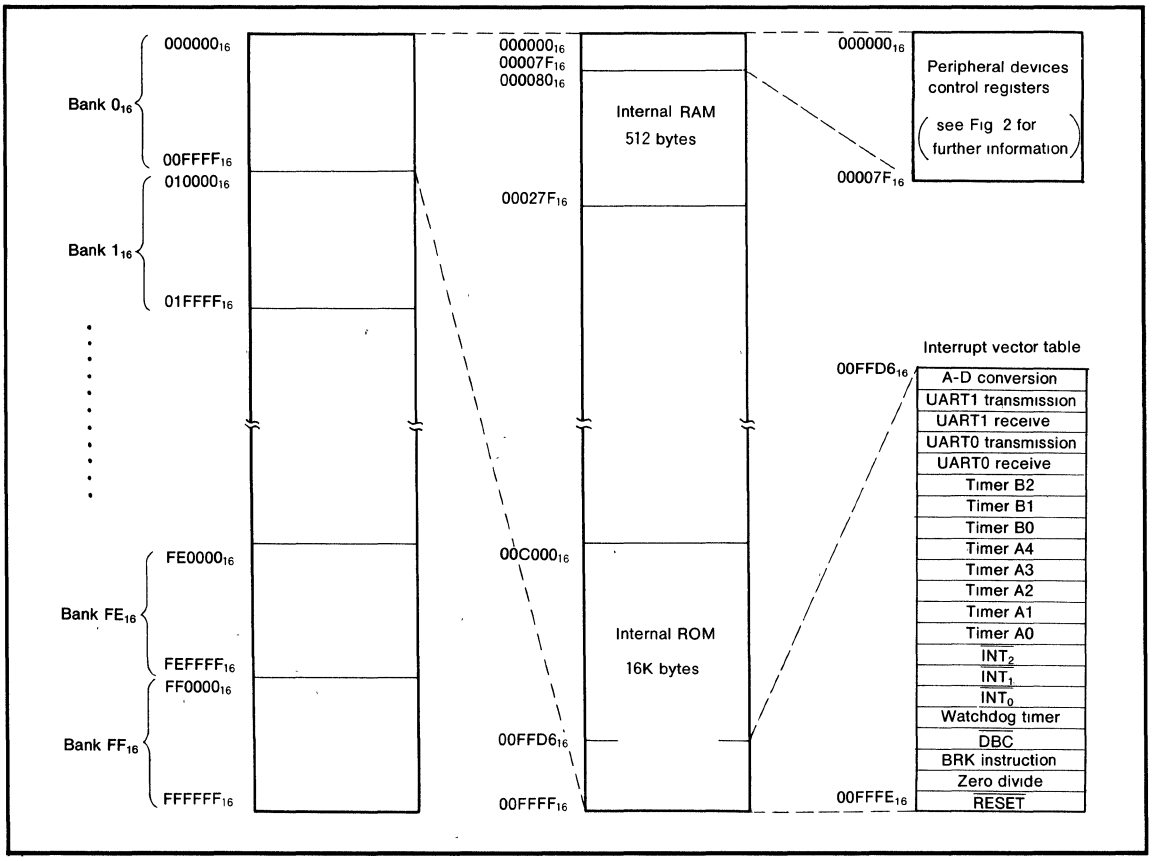


Fig. 1 Memory map

CENTRAL PROCESSING UNIT (CPU)

The CPU has ten registers and is shown in Figure 3. Each of these registers is described below.

ACCUMULATOR A (A)

Accumulator A is the main register of the microcomputer. It consists of 16 bits and the lower 8 bits can be used separately. The data length flag m determines whether the register is used as 16-bit register or as 8-bit register. It is used as a 16-bit register when flag m is "0" and as an 8-bit register when flag m is "1". Flag m is a part of the processor status register (PS) which is described later.

Data operations such as calculations, data transfer, input/output, etc., is executed mainly through the accumulator.

ACCUMULATOR B (B)

Accumulator B has the same functions as accumulator A, but the use of accumulator B requires more instruction bytes and execution cycles than accumulator A.

INDEX REGISTER X (X)

Index register X consists of 16 bits and the lower 8 bits can be used separately. The index register length flag x determines whether the register is used as 16-bit register or as 8-bit register. It is used as a 16-bit register when flag x is "0" and as an 8-bit register when flag x is "1". Flag x is a part of the processor status register (PS) which is described later.

In index addressing mode, register X is used as the index register and the contents of this address is added to obtain the real address.

Also, when executing a block transfer instruction MVP or MVN, the contents of index register X indicates the low-order 16 bits of the source data address. The third byte of the MVP and MVN is the high-order 8 bits of the source data address.

INDEX REGISTER Y (Y)

Index register Y consists of 16 bits and the lower 8 bits can be used separately. The index register length flag x determines whether the register is used as 16-bit register or as 8-bit register. It is used as a 16-bit register when flag x "0" and as an 8-bit register when flag x is "1". Flag x is a part of the processor status register (PS) which is described later.

In index addressing mode, register Y is used as the index register and the contents of this address is added to obtain the real address.

Also, when executing a block transfer instruction MVP or MVN, the content of index register Y indicates the low-order 16 bits of the destination address. The second byte of the MVP and MVN is the high-order 8 bits of the destination data address.

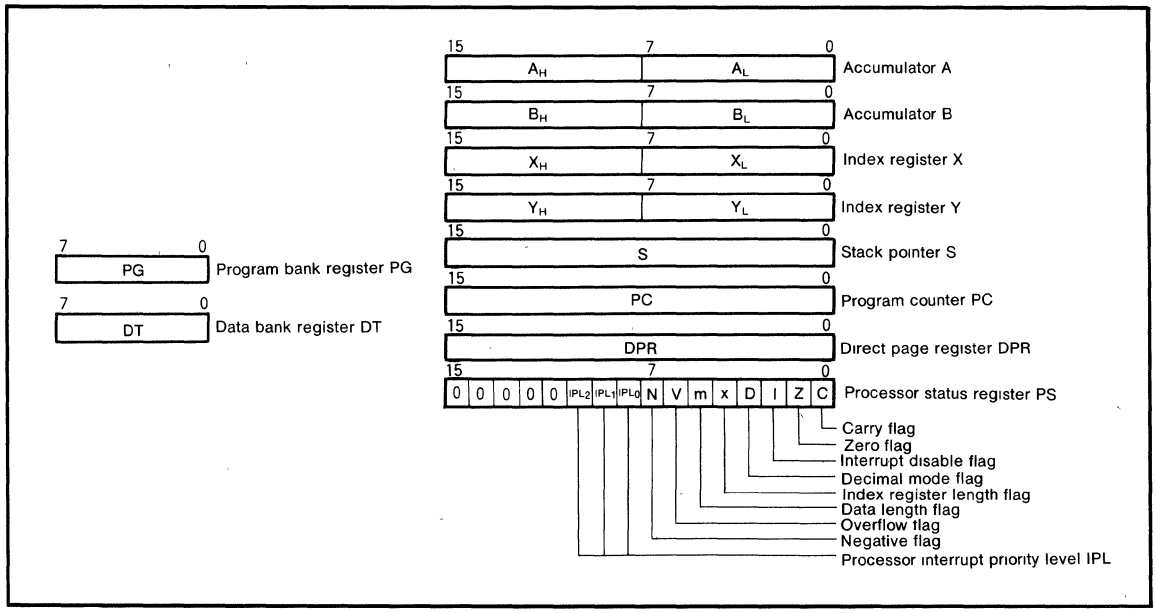


Fig. 3 Register structure

MITSUBISHI ELECTRIC

5. Index register length flag (x)

The index register length flag determines whether index register X and index register Y are used as 16-bit registers or as 8-bit registers. The registers are used as 16-bit registers when flag x is "0" and as 8-bit registers when it is "1". This flag can be set and reset with the SEP and CLP instructions.

6. Data length flag (m)

The data length flag determines whether the data length is 16-bit or 8-bit. The data length is 16-bit when flag m is "0" and 8-bit when it is "1". This flag can be set and reset with the SEM and CLM instructions or with the SEP and CLP instructions.

7. Overflow flag (V)

The overflow flag has meaning when addition or subtraction is performed a word as signed binary number. When the data length flag m is "0", the overflow flag is set when the result of addition or subtraction is outside the range between −32768 and +32767. When the data length flag m is "1", the overflow flag is set when the result of addition or subtraction is outside the range between −128 and +127. It is reset in all other cases. The overflow flag can also be set and reset directly with the SEP, and CLV or CLP instructions.

8. Negative flag (N)

The negative flag is set when the result of arithmetic operation or data transfer is negative (If data length flag m is "0", when data bit 15 is "1". If data length flag m is "1", when data bit 7 is "1".) It is reset in all other cases. It can also be set and reset with the SEP and CLP instructions.

9. Processor interrupt priority level (IPL)

The processor interrupt priority level (IPL) consists of 3 bits and determines the priority of processor interrupts from level 0 to level 7. Interrupt is enabled when the interrupt priority of the device requesting interrupt (set using the interrupt control register) is higher than the processor interrupt priority. When interrupt is enabled, the current processor interrupt priority level is saved in a stack and the processor interrupt priority level is replaced by the interrupt priority level of the device requesting the interrupt. Refer to the section on interrupts for more details.

BUS INTERFACE UNIT

The CPU operates on an internal clock frequency which is obtained by dividing the external clock frequency $f_{(X_{IN})}$ by two. This frequency is twice the bus cycle frequency. In order to speed-up processing, a bus interface unit is used to pre-fetch instructions when the data bus is idle. The bus interface unit synchronizes the CPU and the bus and prefetches instructions. Figure 4 shows the relationship between the CPU and the bus interface unit. The bus interface unit has a program address register, a 3-byte instruction queue buffer, a data address register, and a 2-byte data buffer.

The bus interface unit obtains an instruction code from memory and stores it in the instruction queue buffer, obtains data from memory and stores it in the data buffer, or writes the data from the data buffer to the memory.

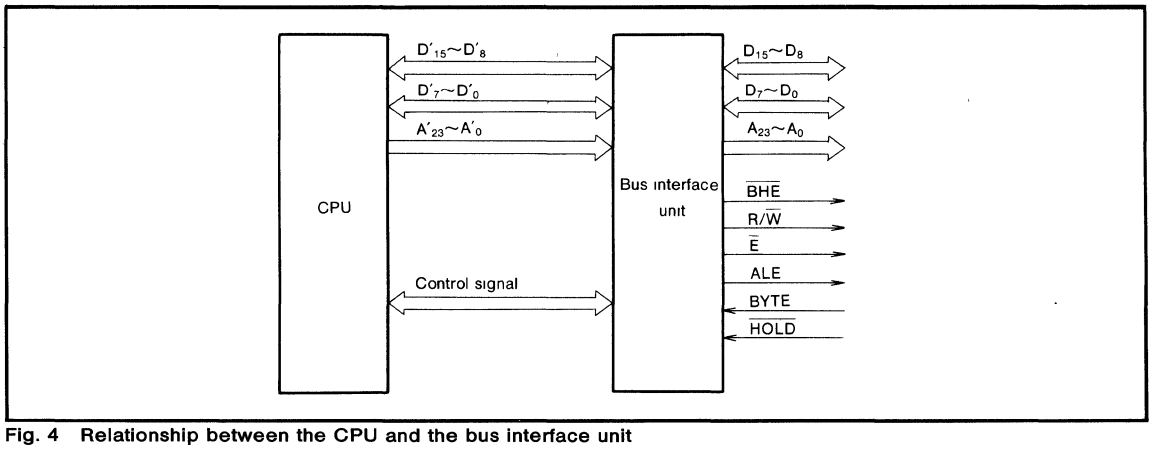


Fig. 4 Relationship between the CPU and the bus interface unit

Instruction code read will be described first.
The CPU obtains instruction codes from the instruction queue buffer and executes them. The CPU notifies the bus interface unit that it is requesting an instruction code during an instruction code request cycle. If the requested instruction code is not yet stored in the instruction queue buffer, the bus interface unit halts the CPU until it can store more instructions than requested in the instruction queue buffer.
Even if there is no instruction code request from the CPU, the bus interface unit reads instruction codes from memory and stores them in the instruction queue buffer when the instruction queue buffer is empty or when only one instruction code is stored and the bus is idle on the next cycle.
This is referred to as instruction pre-fetching.
Normally, when reading an instruction code from memory, if the accessed address is even the next odd address is read together with the instruction code and stored in the instruction queue buffer.
However, in memory expansion mode or microprocessor mode, if the bus width switching pin BYTE is "H", external data bus width is 8 bits and the address to be read is in external memory area is odd, only one byte is read and stored in the instruction queue buffer. Therefore, waveform (1) or (3) in Figure 5 is used for instruction code read.
Data read and write are described below.
The CPU notifies the bus interface unit when performing data read or write. At this time, the bus interface unit halts the CPU if the bus interface unit is already using the bus or if there is a request with higher priority. When data read or write is enabled, the bus interface unit uses one of the waveforms from (1) to (6) in Figure 5 to perform the operation.
During data read, the CPU waits until the entire data is stored in the data buffer. The bus interface unit sends the address received from the CPU to the address bus. Then it reads the memory when the $\overline{E}$ signal is "L" and stores the result in the data buffer.
During data write, the CPU writes the data in the data buffer and the bus interface unit writes it to memory. Therefore, the CPU can proceed to the next step without waiting for write to complete. The bus interface unit sends the address received from the CPU to the address bus. Then when the $\overline{E}$ signal is "L", the bus interface unit sends the data in the data buffer to the data bus and writes it to memory.

Table 2. Addresses of interrupt control registers

Interrupt control registers	Addresses
A-D conversion interrupt control register	000070_{16}
UART0 transmit interrupt control register	000071_{16}
UART0 receive interrupt control register	000072_{16}
UART1 transmit interrupt control register	000073_{16}
UART1 receive interrupt control register	000074_{16}
Timer A0 interrupt control register	000075_{16}
Timer A1 interrupt control register	000076_{16}
Timer A2 interrupt control register	000077_{16}
Timer A3 interrupt control register	000078_{16}
Timer A4 interrupt control register	000079_{16}
Timer B0 interrupt control register	$00007A_{16}$
Timer B1 interrupt control register	$00007B_{16}$
Timer B2 interrupt control register	$00007C_{16}$
$\overline{INT_0}$ interrupt control register	$00007D_{16}$
$\overline{INT_1}$ interrupt control register	$00007E_{16}$
$\overline{INT_2}$ interrupt control register	$00007F_{16}$

Interrupts caused by a BRK instruction and when dividing by zero are software interrupts and are not included in this list.

Other interrupts previously mentioned are A-D converter, UART, Timer, INT interrupts. The priority of these interrupts can be changed by changing the priority level in the corresponding interrupt control register by software.

Figure 8 shows a diagram of the interrupt priority resolution circuit. When an interrupt is caused, the each interrupt device compares its own priority with the priority from above and if its own priority is higher, then it sends the priority below and requests the interrupt. If the priorities are the same, the one above has priority.

This comparison is repeated to select the interrupt with the highest priority among the interrupts that are being requested. Finally the selected interrupt is compared with the processor interrupt priority level (IPL) contained in the processor status register (PS) and the request is accepted if it is higher than IPL and the interrupt disable flag I is "0". The request is not accepted if flag I is "1". The reset, $\overline{DBC}$, and watchdog timer interrupts are not affected by the interrupt disable flag I.

When an interrupt is accepted, the contents of the processor status register (PS) is saved to the stack and the interrupt disable flag I is set to "1".

Furthermore, the interrupt request bit of the accepted interrupt is cleared to "0" and the processor interrupt priority level (IPL) in the processor status register (PS) is replaced by the priority level of the accepted interrupt.

Therefore, multi-level priority interrupts are possible by resetting the interrupt disable flag I to "0" and enable further interrupts.

For reset, $\overline{DBC}$, watchdog timer, zero divide, and BRK instruction interrupts, which do not have an interrupt control register, the processor interrupt level (IPL) is set as shown in Table 3.

Priority resolution is performed by latching the interrupt request bit and interrupt priority level so that they do not change. They are sampled at the first half and latched at the last half of the operation code fetch cycle.

Because priority resolution takes some time, no sampling pulse is generated for a certain interval even if it is the next operation code fetch cycle.

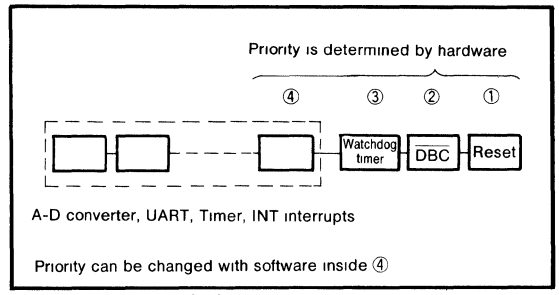


Fig. 7 Interrupt priority

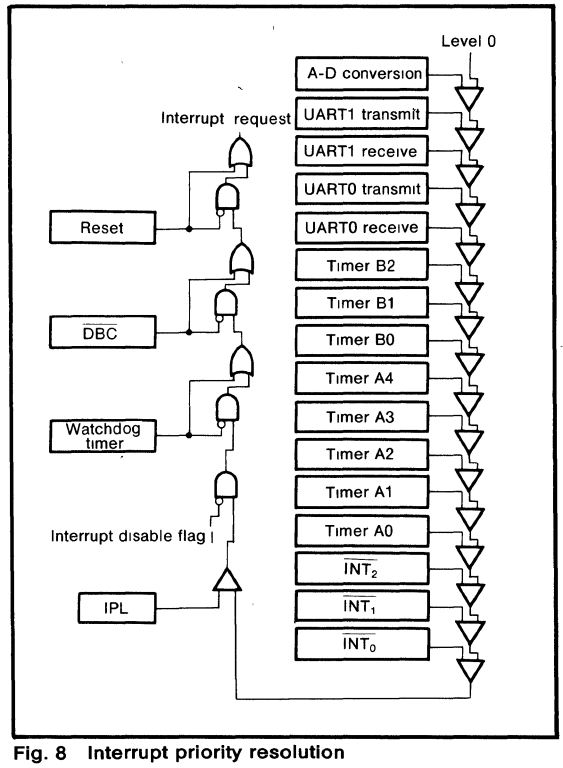


Fig. 8 Interrupt priority resolution

TIMER

There are eight 16-bit timers. They are divided by type into timer A(5) and timer B(3).

The timer I/O pins are shared with I/O pins for port P5 and P6. To use these pins as timer input pins, the data direction register bit corresponding to the pin must be cleared to "0" to specify input mode.

Using this timer, confirm the function as this timer is different a little from M37700M2-XXXFP's.

TIMER A

Figure 11 shows a block diagram of timer A.

Timer A has four modes; timer mode, event counter mode, one-shot pulse mode, and pulse width modulation mode. The mode is selected with bits 0 and 1 of the timer Ai mode register (i = 0 to 4). Each of these modes is described below.

(1) Timer mode (00)

Figure 12 shows the bit configuration of the timer Ai mode register during timer mode. Bits 0, 1, and 5 of the timer Ai mode register must always be "0" in timer mode.

Bit 3 is ignored if bit 4 is "0".

Bits 6 and 7 are used to select the timer counter source. The counting of the selected clock starts when the count start flag is "1" and stops when it is "0".

Figure 13 shows the bit configuration of the count start flag. The counter is decremented, an interrupt is caused and the interrupt request bit in the timer Ai interrupt control register is set when the contents becomes 0000_{16}. At the same time, the contents of the reload register is transferred to the counter and count is continued.

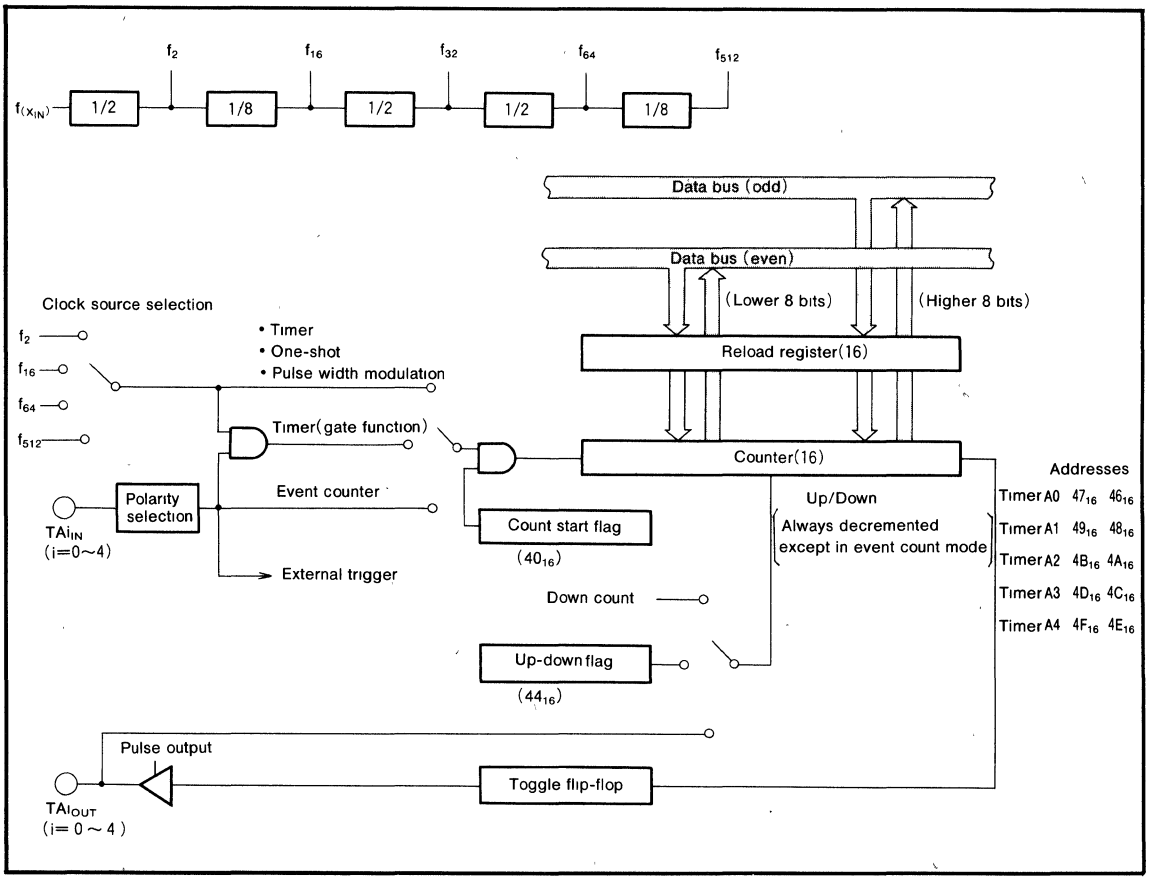


Fig. 11 Block diagram of timer A

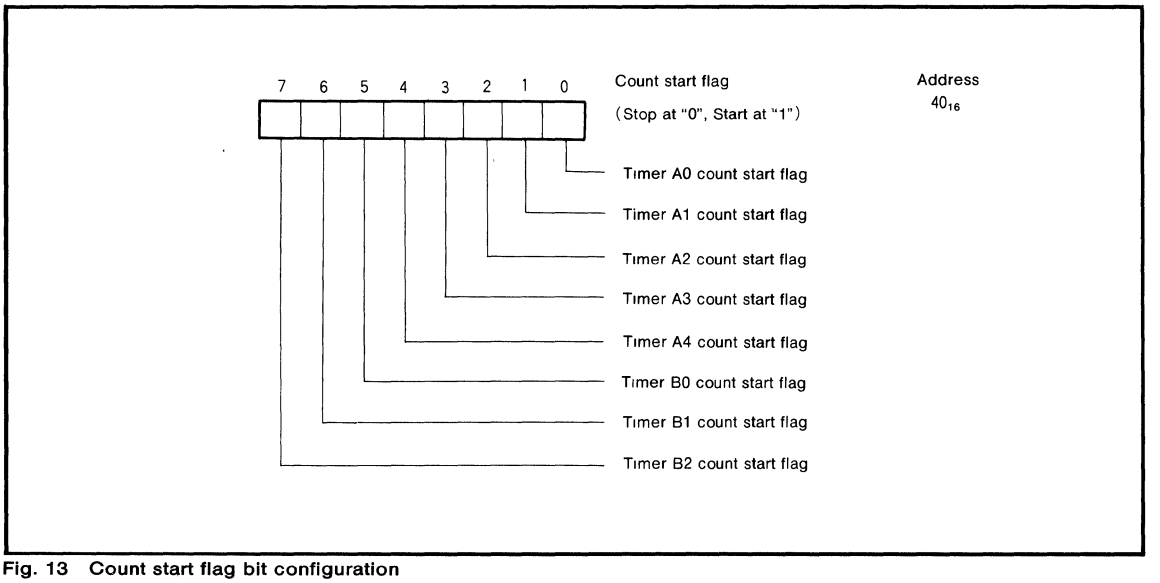


Fig. 13 Count start flag bit configuration

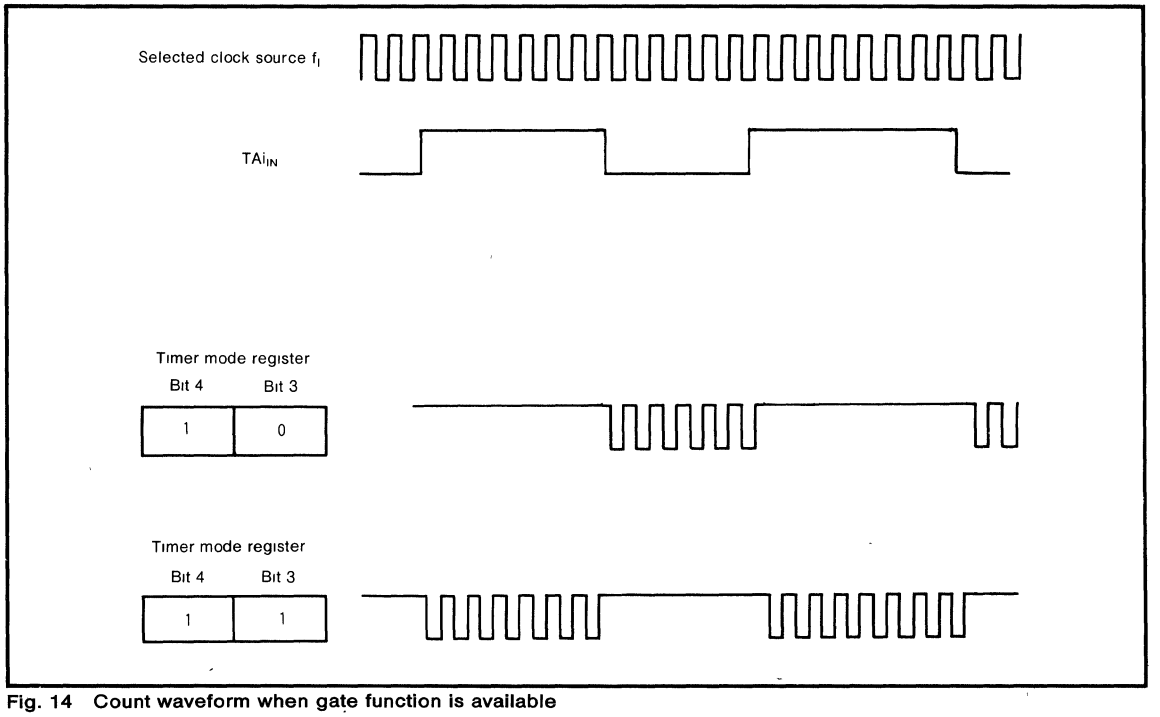


Fig. 14 Count waveform when gate function is available

Data write and data read are performed in the same way as for timer mode. That is, when data is written to timer Ai halted, it is also written to the reload register and the counter. When data is written to timer Ai which is busy, the data is written to the reload register, but not to the counter. The counter is reloaded with new data from the reload register at the next reload time. The counter can be read at any time.

In event counter mode, whether to increment or decrement the counter can also be determined by supplying two-phase pulse input with phase shifted by 90° to timer A2, A3, or A4. There are two types of two-phase pulse processing operations. One uses timers A2 and A3, and the other uses timer A4. In either processing operation, two-phase pulse is input in the same way, that is, pulses out of phase by 90° are input at the TAj_{OUT} (j=2 to 4) pin and TAj_{IN} pin.

When timers A2 and A3 are used, as shown in Figure 17, the count is incremented when a rising edge is input to the TAk_{IN} pin after the level of TAk_{OUT} (k=2, 3) pin changes from "L" to "H", and when the falling edge is inserted, the count is decremented.

For timer A4, as shown in Figure 18, when a phase related pulse with a rising edge input to the $TA4_{IN}$ pin is input after the level of $TA4_{OUT}$ pin changes from "L" to "H", the count is incremented at the respective rising edge and falling edge of the $TA4_{OUT}$ pin and $TA4_{IN}$ pin.

When a phase related pulse with a falling edge input to the $TA4_{OUT}$ pin is input after the level of $TA4_{IN}$ pin changes from "H" to "L", the count is decremented at the respective rising edge and falling edge of the $TA4_{IN}$ pin and $TA4_{OUT}$ pin. When performing this two-phase pulse signal processing, timer Aj mode register bit 0 and bit 4 must be set to "1" and bits 1, 2, 3, and 5 must be "0". Bits 6 and 7 are ignored. Note that bits 5, 6, and 7 of the up-down flag register (44_{16}) are the two-phase pulse signal processing selection bit for timer A2, A3, and A4 respectively. Each timer operates in normal event counter mode when the corresponding bit is "0" and performs two-phase pulse signal processing when it is "1".

Count is started by setting the count start flag to "1". Data write and read are performed in the same way as for normal event counter mode. Note that the direction register of the input port must be set to input mode because two-phase pulse signal is input. Also, there can be no pulse output in this mode.

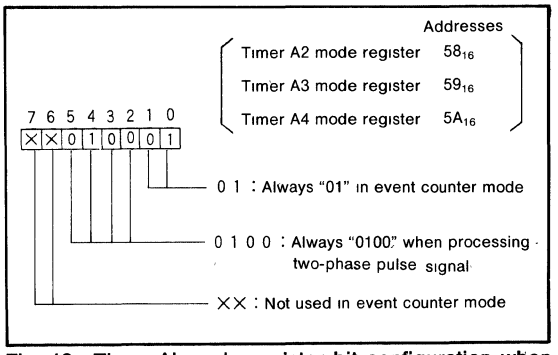


Fig. 19 Timer Aj mode register bit configuration when performing two-phase pulse signal processing in event counter mode

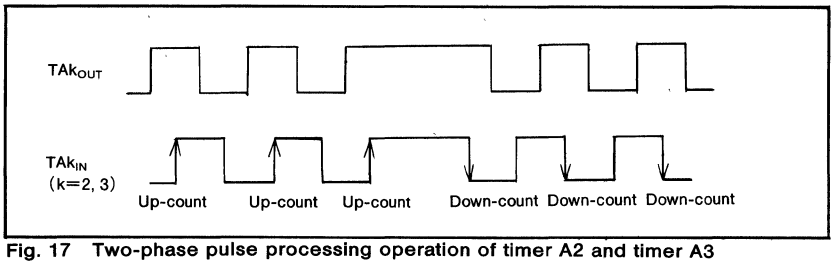


Fig. 17 Two-phase pulse processing operation of timer A2 and timer A3

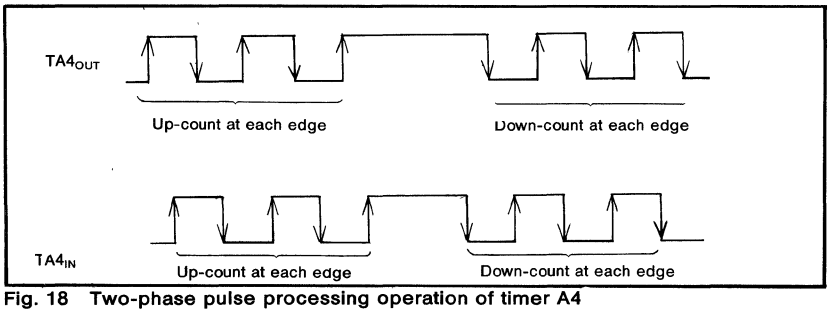


Fig. 18 Two-phase pulse processing operation of timer A4

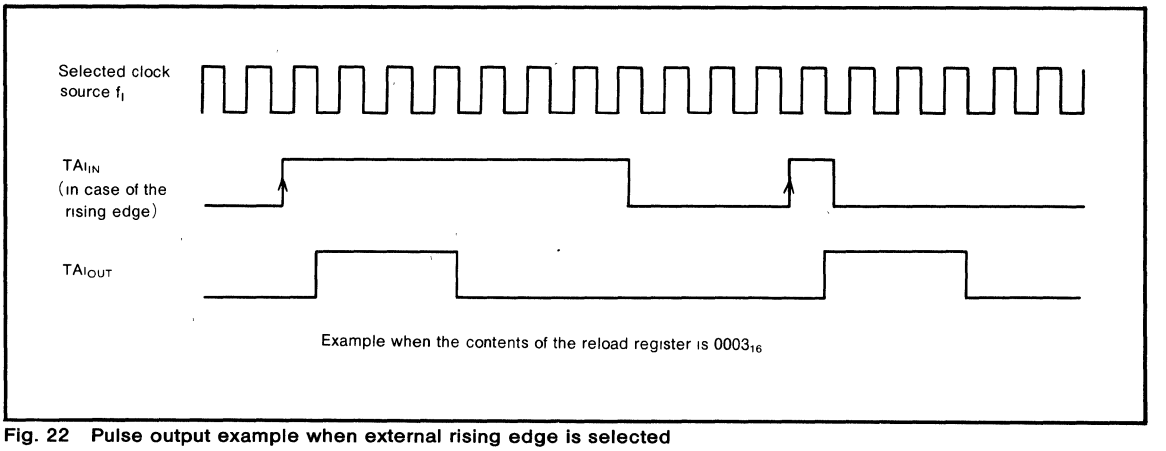


Fig. 22 Pulse output example when external rising edge is selected

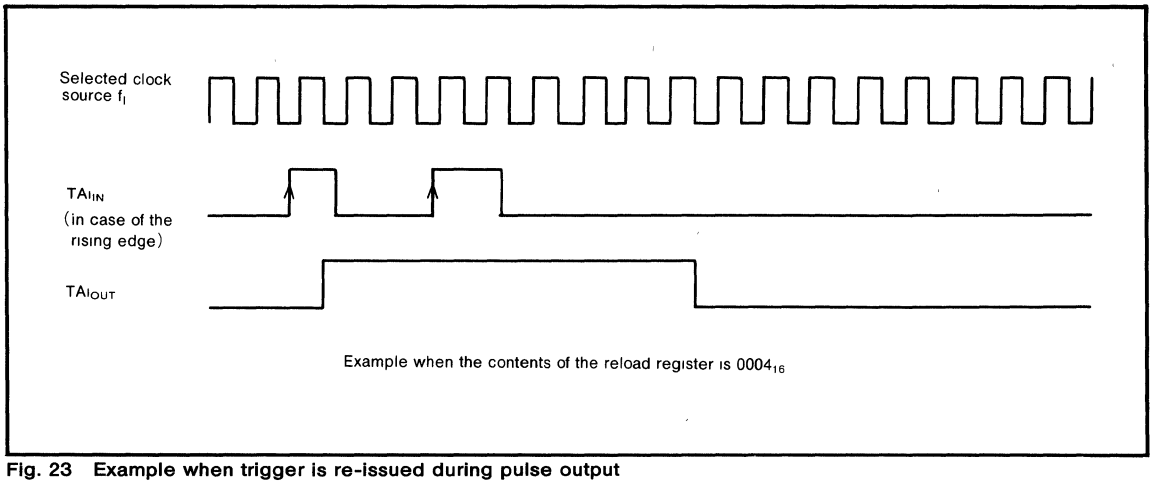


Fig. 23 Example when trigger is re-issued during pulse output

Therefore, if the low order 8-bit of the reload register is n, the period of the generated pulse is

$$\frac{1}{\text{selected clock frequency}} \times (n+1).$$

The high order 8-bit function as an 8-bit length pulse width modulator using this pulse as input. The operation is the same as for 16-bit length pulse width modulator except that the length is 8 bits. If the high order 8-bit of the reload register is m, the duration "H" of pulse is

$$\frac{1}{\text{selected clock frequency}} \times (n+1) \times m.$$

And the output pulse period is

$$\frac{1}{\text{selected clock frequency}} \times (n+1) \times (2^8-1).$$

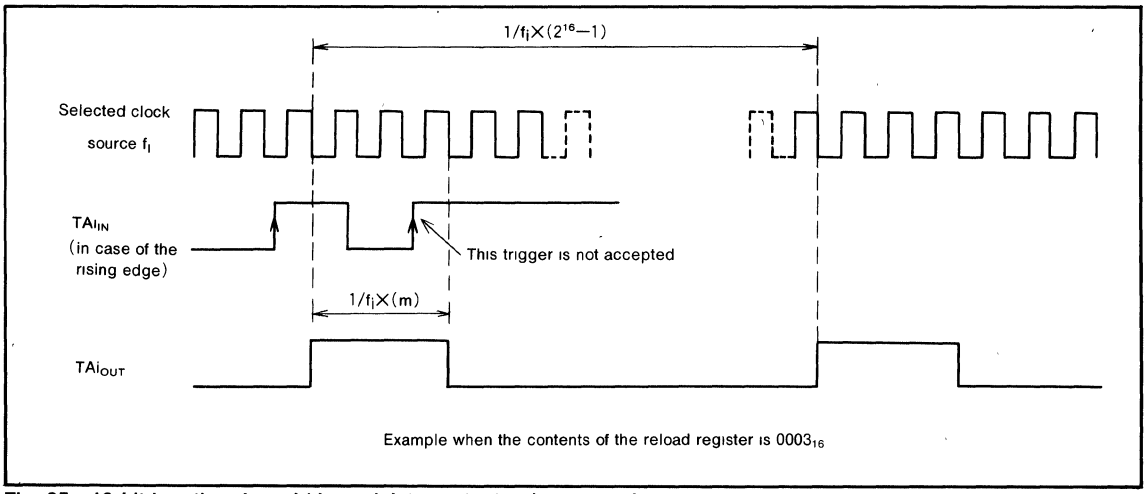


Fig. 25 16-bit length pulse width modulator output pulse example

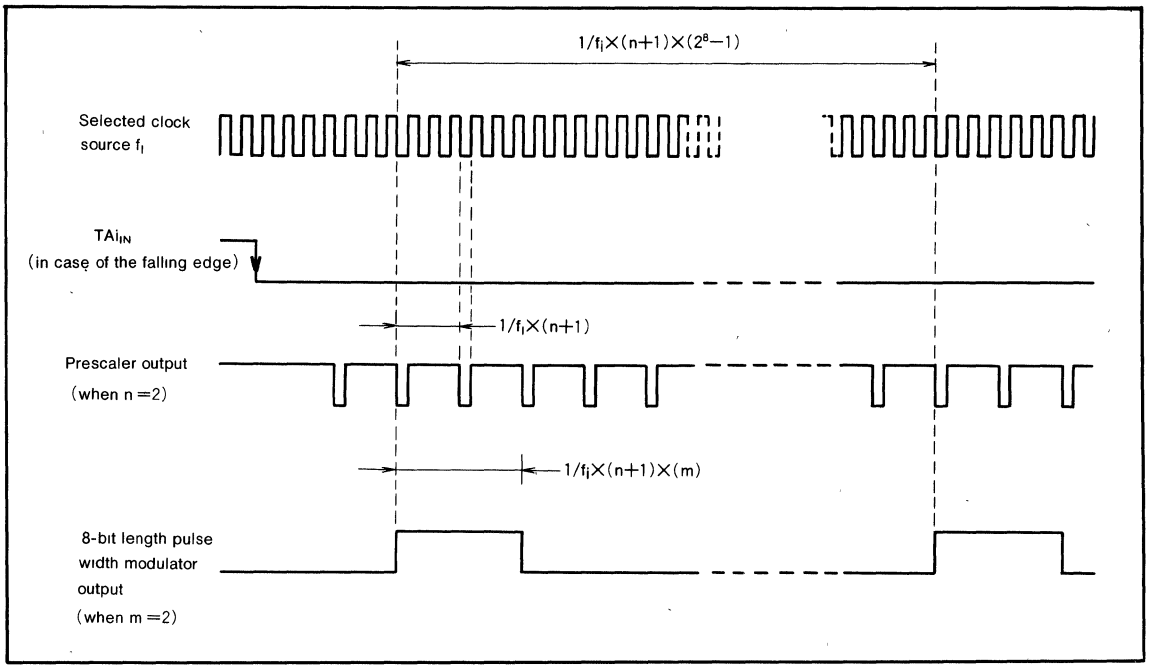


Fig. 26 8-bit length pulse width modulator output pulse example

(2) Event counter mode (01)

Figure 29 shows the bit configuration of the timer Bi mode register during event counter mode. In event counter mode, the bit 0 in the timer Bi mode register must be "1" and bit 1 must be "0".

The input signal from the TBi_{IN} pin is counted when the count start flag is "1" and counting is stopped when it is "0". Count is performed at the fall of the input signal when bits 2, and 3 are "0" and at the rise of the input signal when bit 3 is "0" and bit 2 is "1".

When bit 3 is "1" and bit 2 is "0", count is performed at the rise and fall of the input signal.

Data write, data read and timer interrupt are performed in the same way as for timer mode.

(3) Pulse period measurement/pulse width measurement mode (10)

Figure 30 shows the bit configuration of the timer Bi mode register during pulse period measurement/pulse width measurement mode.

In pulse period measurement/pulse width measurement mode, bit 0 must be "0" and bit 1 must be "1". Bits 6 and 7 are used to select the clock source. The selected clock is counted when the count start flag is "1" and counting stops when it is "0".

The pulse period measurement mode is selected when bit 3 is "0". In pulse period measurement mode, the selected clock is counted during the interval starting at the fall of the input signal from the TBi_{IN} pin to the next fall or at the rise of the input signal to the next rise and the result is stored in the reload register. In this case, the reload register acts as a buffer register.

When bit 2 is "0", the clock is counted from the fall of the input signal to the next fall. When bit 2 is "1", the clock is counted from the rise of the input signal to the next rise.

In the case of counting from the fall of the input signal to the next fall, counting is performed as follows. As shown in Figure 31, when the fall of the input signal from TBi_{IN} pin is detected, the contents of the counter is transferred to the reload register. Next the counter is cleared and count is started from the next clock. When the fall of the next input signal is detected, the contents of the counter is transferred to the reload register once more, the counter is cleared, and the count is started. The period from the fall of the input signal to the next fall is measured in this way.

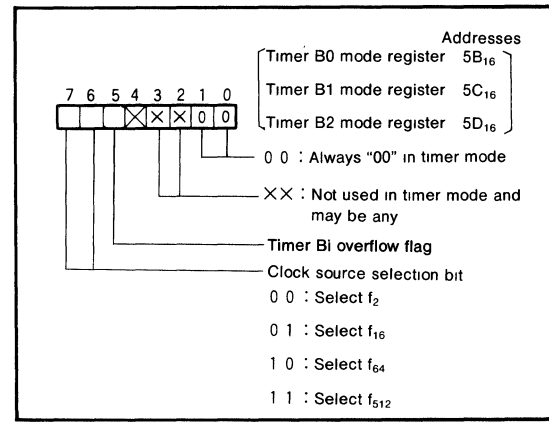


Fig. 28 Timer Bi mode register bit configuration during timer mode

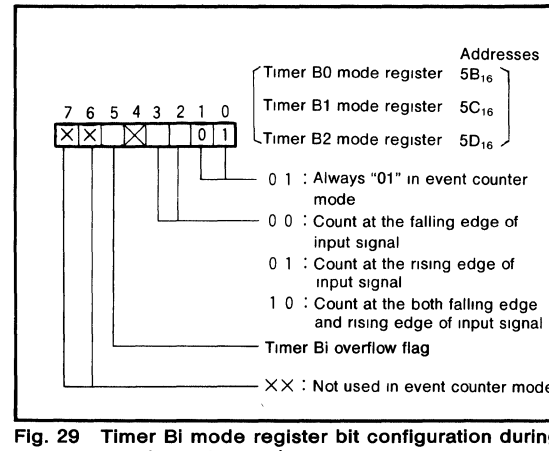


Fig. 29 Timer Bi mode register bit configuration during event counter mode

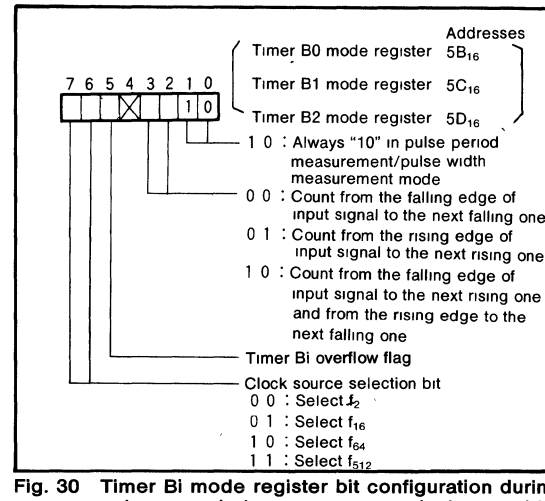


Fig. 30 Timer Bi mode register bit configuration during pulse period measurement/pulse width measurement mode

Timer B can detect phase difference by using timer B1 and timer B0. The phase detection mode is explained below. Figure 33 shows a block diagram of the phase detection mode. In the phase detection mode, timer B1 and timer B0 are used. Set timer B1 to the timer mode and timer B0 to the pulse period measurement/pulse width measurement mode. For selection of the phase detection mode, set bit 4 of the timer B0 mode register ($5B_{16}$ address) to "1". Bit 5 of timer B0 mode register functions as phase detection flag by setting bit 4 to "1". Figure 34 shows the bit configuration of the timer B0 mode register and the timer B1 mode register in the phase detection mode.

Figure 35 shows an example of operation in the phase detection mode. First, each time the counter of timer B1 is set to 0000_{16} in the timer mode of timer B1, a signal reversing polarity is generated as a reference signal. Next an external signal is input from the $TB0_{IN}$ pin. By setting bit 4 of the timer B0 mode register to "1", timer B0 measures the pulse width of the logical sum (AND) signal of the reference signal generated from timer B1 and the external input signal from the $TB0_{IN}$ pin. The bit 5 (phase detection flag) of the timer B0 mode register indicates whether the phase of the signal from the $TB0_{IN}$ pin is ahead or behind with respect to the reference signal. The phase detection flag gets the input level from the $TB0_{IN}$ pin at the rising from "L" to "H" of the reference signal. "0" of phase detection flag indicates that the phase of the input signal from the $TB0_{IN}$ pin is behind with respect to the reference signal. "1" indicates that the phase of the input signal from the $TB0_{IN}$ pin is ahead with respect to the reference signal.

To detect the phase difference between the reference signal and the input signal of the $TB0_{IN}$ pin, advance the phase of the input signal from the $TB0_{IN}$ pin with respect to the reference signal. In this state, the phase difference between the reference signal and the input signal from the $TB0_{IN}$ pin can be detected by measuring the pulse width by timer B0.

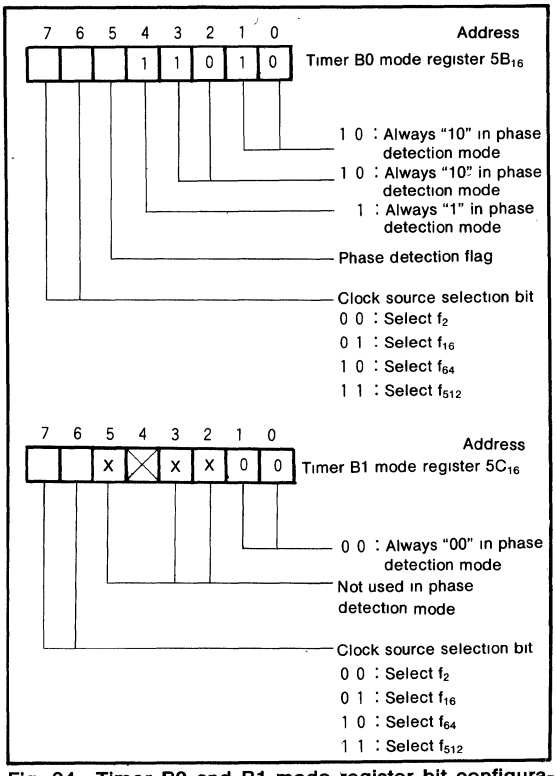


Fig. 34 Timer B0 and B1 mode register bit configuration during phase detection mode

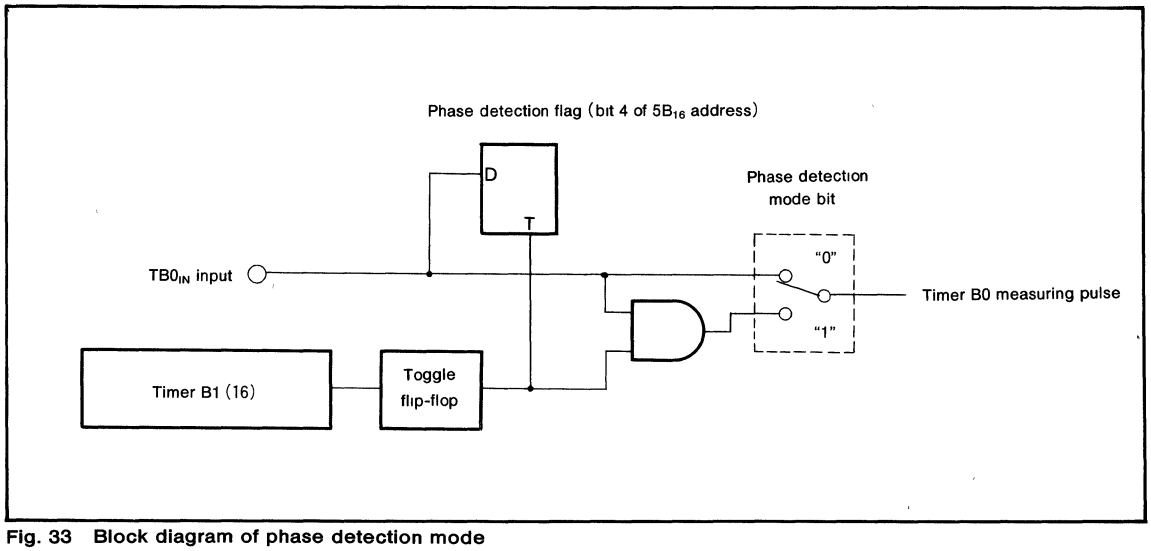


Fig. 33 Block diagram of phase detection mode

The dead-time timer operates as a one-shot pulse timer. When a trigger arrives, the dead-time timer starts pulse output, and when the value of the timer reaches 00_{16}, it terminates pulse output, stops operation, and waits for the next trigger.

The output polarity of three-phase waveform depends on the output polarity setting toggle flip-flops. When the contents of the output polarity toggle flip-flops is "0", the positive phase waveform is output at "H" level, and when "1", it is output at "L" level (three-phase waveform is output using negative logic).

The output polarity setting toggle flip-flops each have output polarity setting buffers shown in Figure 39. When the contents of timer B2 counter reaches 0000_{16}, the contents of output polarity setting buffers is set in the output polarity setting toggle flip-flops. After this, the output polarity setting toggle flip-flops have polarity reversed for each termination of one-shot pulse of timer (timer A2, A1 or A0) corresponding to each phase.

An example of U phase waveform is shown in Figure 40 to explain waveform output operation. Writing "0" to the U phase waveform start level setting bit (bit 1 of 64_{16} address) and actuating timer B2 makes the three-phase waveform mode effective. When the contents of timer B2 counter reaches 0000_{16}, timer A2 starts one-shot pulse output. At this time, the contents ("0" in this case) of U phase output polarity setting buffer is set in the output polarity setting toggle flip-flop 2. At termination of one-shot pulse output of timer A2, the contents of the output polarity setting toggle flip-flop 2 changes from "0" to "1" and a one-shot pulse of the eight-bit dead-time timer is output to set a time so that the "L" level of U phase waveform and $\overline{U}$ phase waveform with the negative phase of U phase waveform does not overlap.

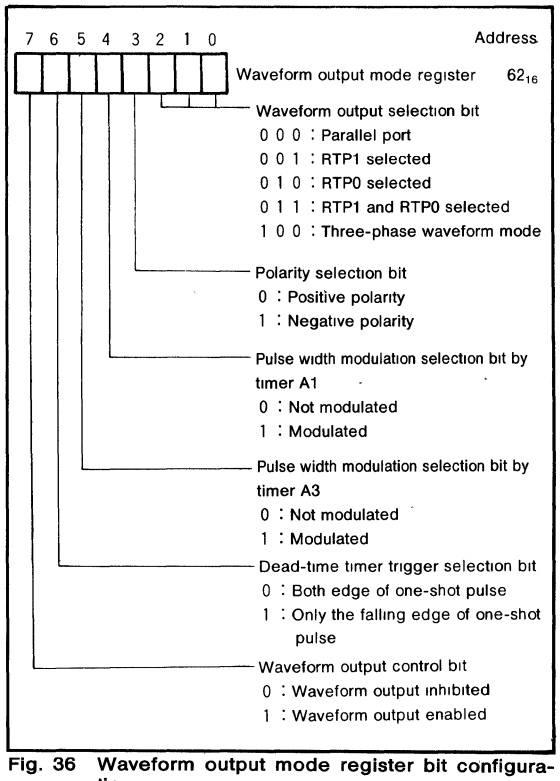


Fig. 36 Waveform output mode register bit configuration

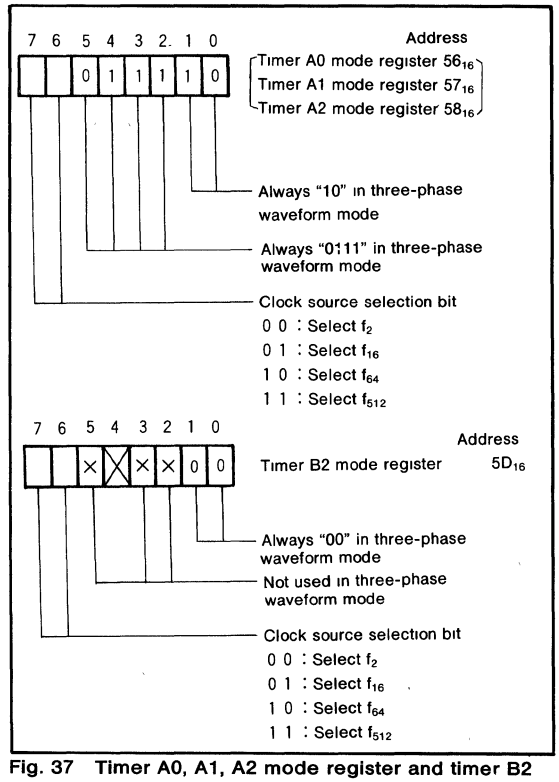


Fig. 37 Timer A0, A1, A2 mode register and timer B2 mode register bit configuration

The U-phase waveform output that began at "H" level remains at "H" level until termination of the one-shot pulse output of the dead-time timer, even when the contents of the output polarity setting toggle flip-flop 2 changes from "0" to "1" due to one-shot pulse output of timer A2. At termination of the one-shot pulse output of the dead-time timer, "1" of the output polarity setting toggle flip-flop 2 already reversed becomes effective and the U-phase waveform changes to "L" level. Next write "1" again to the U-phase output polarity setting buffer (bit 1 of 64_{16} address) before the counter of timer B2 reaches 0000_{16}. When the counter of timer B2 reaches 0000_{16}, the one-shot pulse output of timer B2 begins to operate. At the same time, "1" written to the U-phase output polarity setting buffer is set in the output polarity setting toggle flip-flop 2 and the U-phase waveform output remains at "L" level. At termination of the one-shot pulse output of timer A2, the contents of the output polarity setting toggle flip-flop 2 changes from "1" to "0" and the one-shot pulse output of the dead-time timer begins to operate. The U-phase waveform output, when the contents of the output polarity setting toggle flip-flop changes from "1" to "0", changes form "L" to "H" without waiting for termination of the one-shot pulse output of the dead-time timer.

U-phase waveform is generated by repeating the above operation. $\overline{U}$-phase waveform with the negative phase of U-phase waveform is generated in the same way as U-phase waveform, except that the signal contents of the output polarity setting toggle flip-flop 2 is the very reverse of that in U-phase waveform. In this way, U-phase waveform and $\overline{U}$-phase waveform with the negative phase are generated from the pins so that the "L" level does not overlap.

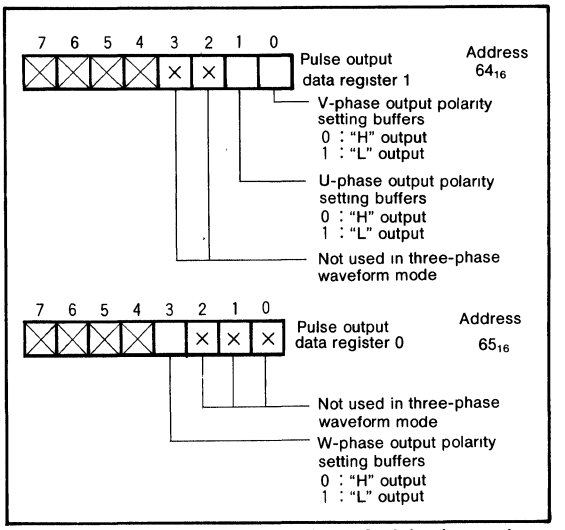


Fig. 39 Pulse output data register 0, 1 in three-phase waveform mode

The width of "L" level can be changed by changing the value of timer B2 and the value of timer A2. This technique for generating waveforms with "L" level not overlapping is also applicable to V phase, W phase, and their negative phases, $\overline{V}$ phase and $\overline{W}$ phase, by using corresponding timers.

The above explanation is for an example of generating three-phase waveform by the triangular wave modulation (called double edge modulation), but three-phase waveform by the saw-tooth-wave modulation (called signal edge modulation) can also be generated by fixing the start level of each phase.

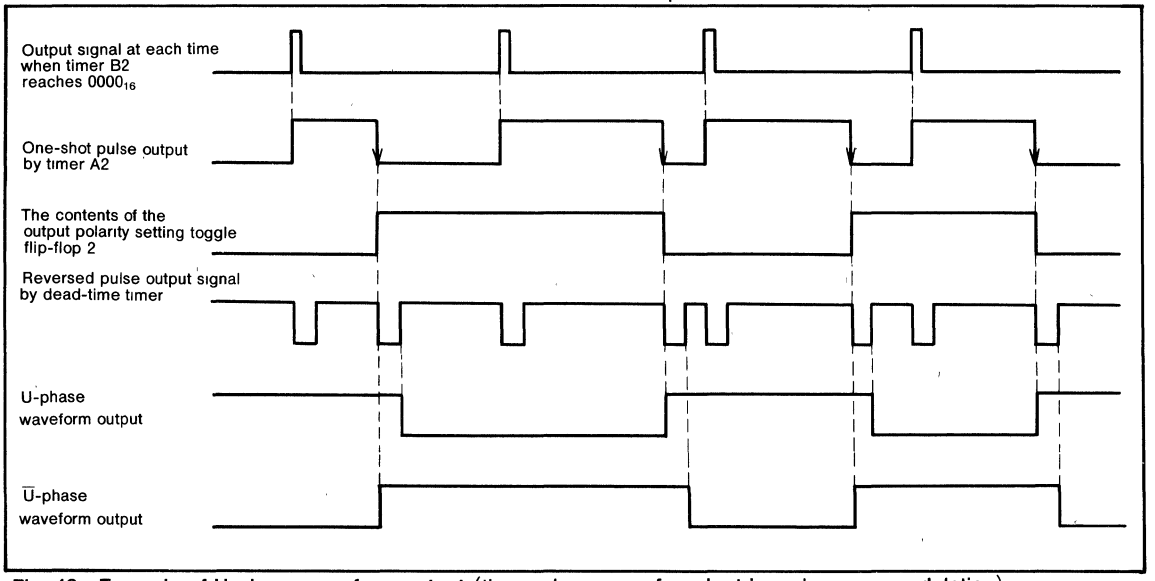


Fig. 40 Example of U-phase waveform output (three-phase waveform by triangular wave modulation)

When "0" is written to a specified bit of the pulse output data register, "L" level is output to the corresponding pulse output port when the counter of corresponding timer becomes 0000_{16}, and when "1" is written, "H" level is output to the pulse output port.

Pulse width modulation can be applied to each pulse output port. Since pulse width modulation involves the use of timers A3 and A1, activate these timers in pulse width modulation mode. When a certain bit of the pulse output register is "1", pulse width modulation is output from the pulse output port when the counter of the corresponding timer becomes 0000_{16}.

Ports $P6_0$, $P5_6$, $P5_5$ and $P5_4$ are applied pulse width modulation by timer A3 by setting the pulse width modulation selection bit by timer A3 (bit 5) of the waveform output mode register to "1".

Ports $P5_3$, $P5_2$, $P5_1$ and $P5_0$ are applied pulse width modulation by timer A1 by setting the pulse width modulation selection bit by timer A1 (bit 4) of the waveform output mode register to "1".

The contents of the pulse output data register 0 can be reversed and output to pulse output ports $P5_3$, $P5_2$, $P5_1$ and $P5_0$ by the polarity selection bit (bit 3) of the waveform output mode register. When the polarity selection bit is "0", the contents of the pulse output data register 0 is output unchangeably, and when "1", the contents of the pulse output data register 0 is reversed and output. When pulse width modulation is applied, likewise the polarity reverse to pulse width modulation can be selected by the polarity selection bit.

Figure 44 shows example of waveforms in pulse output port mode.

Ports selecting the pulse output port mode can control output as in the three-phase wavefrom mode by the waveform output control bit (bit 7) of the waveform output mode register (62_{16} address).

When the waveform output control bit is set to "1", a waveform is output from the port. When this bit is set to "0", waveform output from the port is stopped and the port is placed in floating state.

This bit can be set to "0" by instructions, by inputting a falling edge to the $\overline{INT_0}$ pin, or reset.

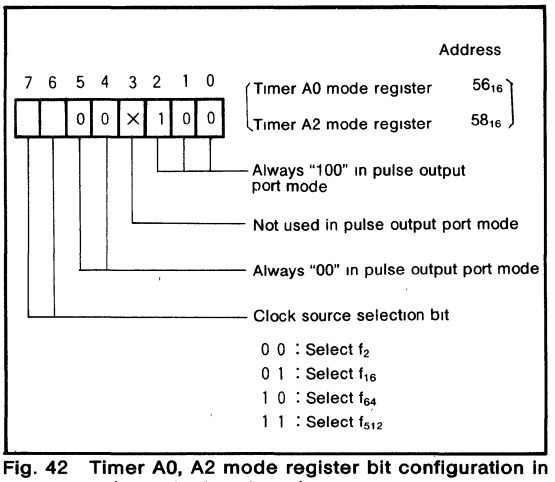


Fig. 42 Timer A0, A2 mode register bit configuration in pulse output port mode

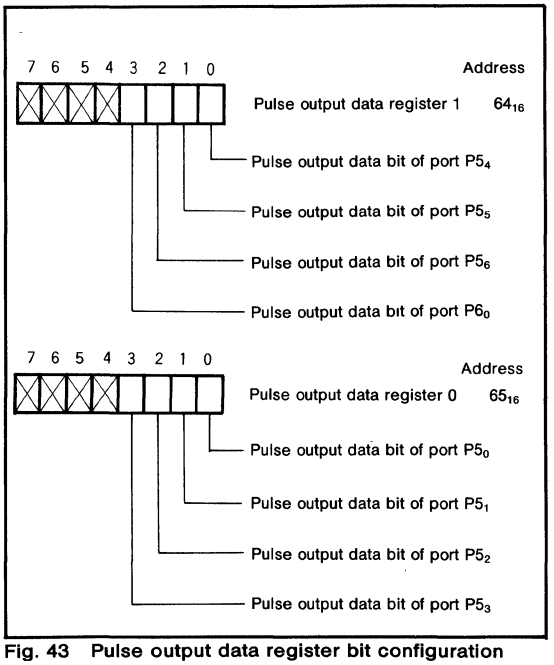


Fig. 43 Pulse output data register bit configuration

SERIAL I/O PORTS

Two independent serial I/O ports are provided. Figure 45 shows a block diagram of the serial I/O ports.

Bits 0, 1, and 2 of the UARTi (i = 0, 1) Transmit/Receive mode register shown in Figure 46 are used to determine whether to use port P8 as parallel port, clock synchronous serial I/O port, or asynchronous (UART) serial I/O port using start and stop bits.

Figures 47 and 48 show the connections of receiver/transmitter according to the mode.

Figure 49 shows the bit configuration of the UARTi transmit/receive control register.

Each communication method is described below.

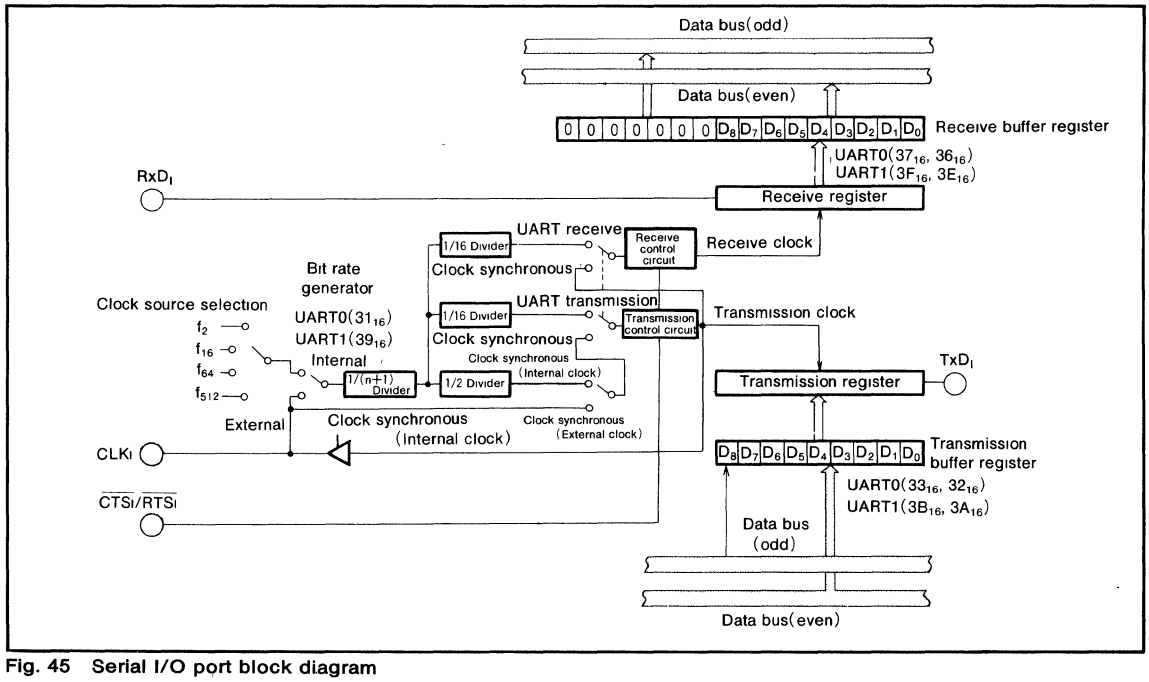


Fig. 45 Serial I/O port block diagram

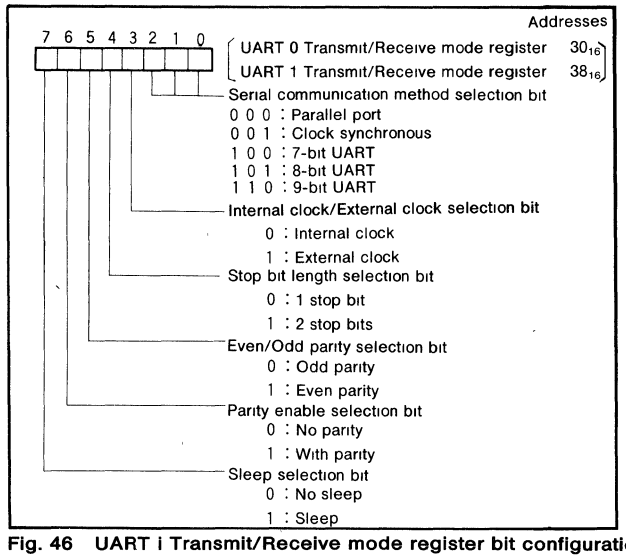


Fig. 46 UART i Transmit/Receive mode register bit configuration

CLOCK SYNCHRONOUS SERIAL COMMUNICATION

A case where communication is performed between two clock synchronous serial I/O ports as shown in Figure 50 will be described. (The transmission side will be denoted by subscript j and the receiving side will be denoted by subscript k.)

Bit 0 of the UARTj transmit/receive mode register and UARTk transmit/receive mode register must be set to "1" and bits 1 and 2 must be "0". The length of the transmission data is fixed at 8 bits.

Bit 3 of the UARTj transmit/receive mode register of the clock sending side is cleared to "0" to select the internal clock. Bit 3 of the UARTk transmit/receive mode register of the clock receiving side is set to "1" to select the external clock. Bits 4, 5 and 6 are ignored in clock synchronous mode. Bit 7 must always be "0".

The clock source is selected by bit 0 (CS_0) and bit 1 (CS_1) of the clock sending side UARTj transmit/receive control register 0. As shown in Figure 45, the selected clock is divided by (n + 1), then by 2, passed through a transmisson control circuit, and output as transmisson clock CLKj. Therefore, when the selected clock is fi,

$$\text{Bit Rate} = fi / \{(n+1) \times 2\}$$

On the clock receiving side, the CS_0 and CS_1 bits of the UARTk transmit/receive control register are ignored because an external clock is selected.

The bit 2 of the clock sending side UARTj transmit/receive control register is clear to "0"to select $\overline{CTS}j$ input. The bit 2 of the clock receiving side is set to "1" to select $\overline{RTS}k$ output. $\overline{CTS}$, and $\overline{RTS}$ signals are described later.

Transmission

Transmission is started when the bit 0 (TEj flag) of UARTj transmit/receive control register 1 is "1", bit 1 is (Tlj flag) of one is "0", and $\overline{CTS}j$ input is "L". As shown in Figure 51, data is output from TxDj pin when transmission clock CLKj changes from "H" to "L". The data is output from the least significant bit.

The Tlj flag indicates whether the transmission buffer register is empty or not. It is cleared to "0" when data is written in the transmission buffer register and set to "1" when the contents of the transmission buffer register is transferred to the transmission register.

When the transmission register becomes empty after the contents has been transmitted, data is transferred automatically from the transmission buffer register to the transmission register if the next transmission start condition is satisfied. If the bit 2 of UARTj transmit/receive control register 0 is "1", $\overline{CTS}j$ input is ignored and transmission start is controlled only by the TEj flag and Tlj flag. Once transmission has started, the TEj flag, Tlj flag, and $\overline{CTS}j$ signals are ignored until data transmission completes. Therefore, transmission is not interrupt when $\overline{CTS}j$ input is changed to "H" during transmission.

The transmission start condition indicated by TEj flag, Tlj flag, and $\overline{CTS}j$ is checked while the T_{ENDj} signal shown in Figure 51 is "H". Therefore, data can be transmitted continuously if the next transmission data is written in the transmission buffer register and Tlj flag is cleared to "0" before the T_{ENDj} signal goes "H".

The bit 3 (TxEPTYj flag) of UARTj transmit/receive control register 0 changes to "1" at the next cycle after the T_{ENDj} signal goes "H" and changes to "0" when transmission starts. Therefore, this flag can be used to determine whether data transmission has completed.

When the Tlj flag changes from "0" to "1", the interrupt request bit in the UARTj transmission interrupt control register is set to "1".

Receive

Receive starts when the bit 2 (RE_k flag) of $UART_k$ transmit/receive control register 1 is set to "1".

The $\overline{RTS}_k$ output is "H" when the RE_k flag is "0" and goes "L" when the RE_k flag changed to "1". It goes back to "H" when receive starts. Therefore, the $\overline{RTS}_k$ output can be used to determine whether the receive register is ready to receive. It is ready when $\overline{RTS}_k$ output is "L".

The data from the RxD_k pin is retrieved and the contents of the receive register is shifted by 1 bit each time the transmission clock CLKj changes from "L" to"H". When an 8-bit data is received, the contents of the receive register is transferred to the receive buffer register and the bit 3 (RI_k flag) of $UART_k$ transmit/receive control register 1 is set to "1". In other words, the setting of the RI_k flag indicates that the receive buffer register contains the received data. At this point, $\overline{RTS}j$ output goes "L" to indicate that the next data can be received. When the RI_k flag changes from "0" to "1", the interrupt request bit in the $UART_k$ receive interrupt control register is set to "1". Bit 4 (OER_k flag) of $UART_k$ transmit/receive control register is set to "1" when the next data is transferred from the receive register to the receive buffer register while RI_k flag is "1", and indicates that the next data was transferred to the receive register before the contents of the receive buffer register was read. RI_k and OER_k flags are cleared automatically to "0" when the low-order byte of the receive buffer register is read. The OER_k flag is also cleared when the RE_k flag is cleared. Bit 5 (FER_k flag), bit 6 (PER_k flag), and bit 7 (SUM_k flag) are ignored in clock synchronous mode.

As shown in Figure 50, with clock synchronous serial communication, data cannot be received unless the transmitter is operating because the receive clock is created from the transmission clock. Therefore, the transmitter must be operating even when there is no data to be sent from $UART_k$ to UARTj.

ASYNCHRONOUS SERIAL COMMUNICATION

Asynchronous serial communication can be performed using 7-, 8-, or 9-bit length data. The operation is the same for all data lengths. The following is the description for 8-bit asynchronous communication.

With 8-bit asynchronous communication, the bit 0 of UARTi transmit/receive mode register is "1", the bit 1 is "0", and the bit 2 is "1".

Bit 3 is used to select an internal clock or an external clock. If bit 3 is "0", an internal clock is selected and if bit 3 is "1", then external clock is selected. If an internal clock is selected, the bit 0 (CS_0) and bit 1 (CS_1) of UARTi transmit/receive control register 0 are used to select the clock source. When an internal clock is selected for asynchronous serial communication, the CLK pin can be used as a normal I/O pin.

The selected internal or external clock is divided by (n+1), then by 16, and passed through a control circuit to create the UART transmission clock or UART receive clock.

Therefore, the transmission speed can be changed by changing the contents n of the bit rate generator. If the selected clock is an internal clock fi or an external clock f_{EXT},

$$\text{Bit Rate} = (f_i \text{ or } f_{EXT}) / \{ (n+1) \times 16 \}$$

Bit 4 is the stop bit length selection bit to select 1 stop bit or 2 stop bits.

The bit 5 is a selection bit of odd parity or even parity.

In the odd parity mode, the parity bit is adjusted so that the sum of the 1's in the data and parity bit is always odd.

In the even parity mode, the parity bit is adjusted so that the sum of the 1's in the data and parity bit is always even.

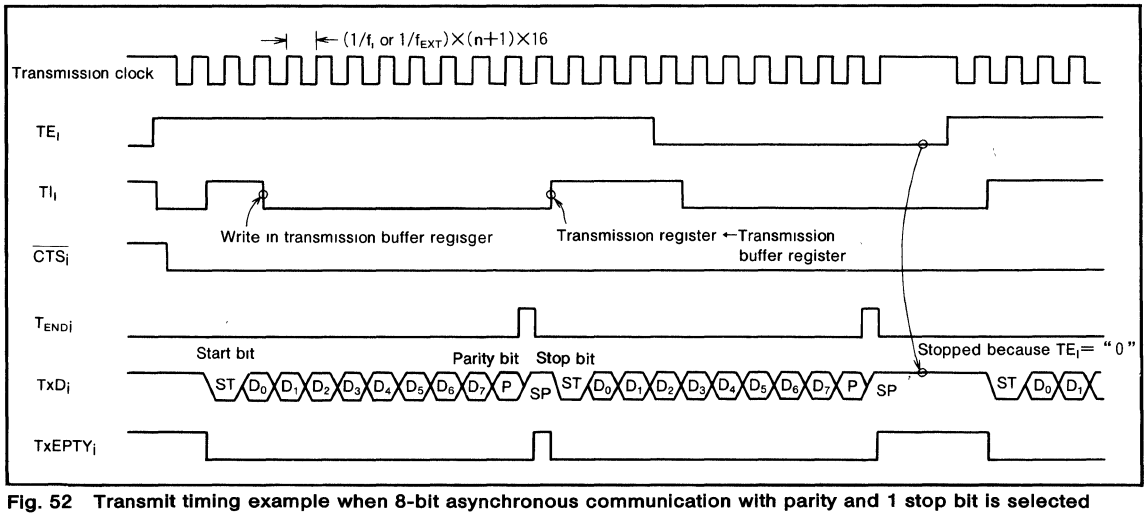


Fig. 52 Transmit timing example when 8-bit asynchronous communication with parity and 1 stop bit is selected

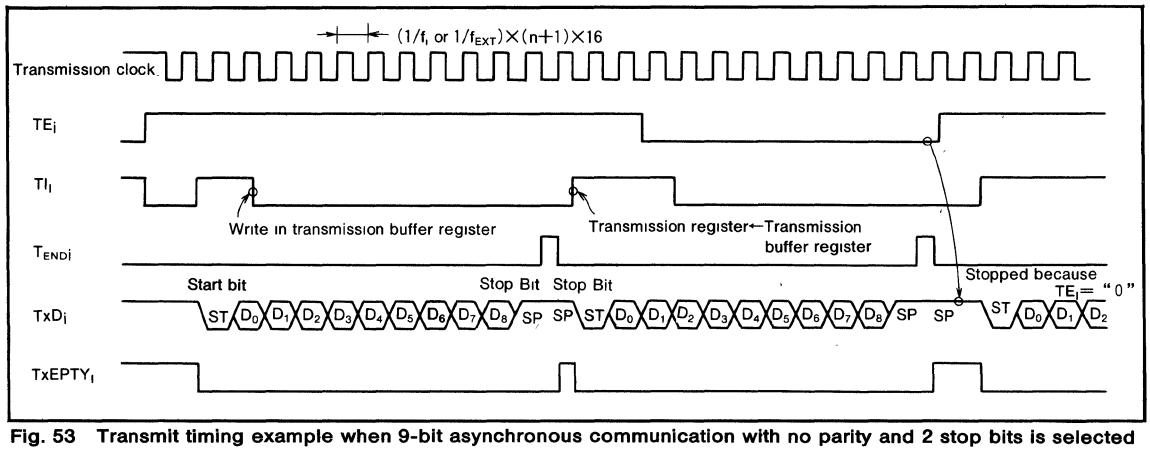


Fig. 53 Transmit timing example when 9-bit asynchronous communication with no parity and 2 stop bits is selected

MITSUBISHI ELECTRIC

If $\overline{RTS_i}$ output is selected by setting the bit 2 of $UART_i$ transmit/receive control register 0 to "1", the $\overline{RTSi}$ output is "H" when the RE_i flag is "0". When the RE_i flag changes to "1", the $\overline{RTSi}$ output goes "L" to indicate receive ready and returns to "H" once receive has started. In other words, $\overline{RTS_i}$ output can be used to determine externally whether the receive register is ready to receive.

The entire transmission data bits are received when the start bit passes the final bit of the receive block shown in Figure 33. At this point, the contents of the receive register is transferred to the receive buffer register and the bit 3 of $UART_i$ transmit/receive control register 1 is set. In other words, the RI_i flag indicates that the receive buffer register contains data when it is set. If $\overline{RTS_i}$ output is selected, $\overline{RTS_i}$ output goes "L" to indicate that the register is ready to receive the next data.

The interrupt request bit in the $UART_i$ receive interrupt control register is set when the RI_i flag changes from "0" to "1".

The bit 4 (OER_i flag) of $UART_i$ transmission control register 1 is set when the next data is transferred from the receive register to the receive buffer register while the RI_i flag is "1". In other words when an overrun error occurs. If the OER_i flag is "1", it indicates that the next data has been transferred to the receive buffer register before the contents of the receive butter register has been read.

Bit 5 (FER_i flag) is set when the number of stop bits is less than required (framing error).

Bit 6 (PER_i flag) is set when a parity error occurs.

Bit 7 (SUM_i flag) is set when either the OER_i flag, FER_i flag, or the PER_i flag is set. Therefore, the SUM_i flag can be used to determine whether there is an error.

The setting of the RI_i flag, OER_i flag, FER_i flag, and the PERi flag is performed while transferring the contents of the receive register to the receive buffer register. The RI_i OER_i, FER_i, PER_i, and SUM_i flags are cleared when the low order byte of the receive buffer register is read or when the RE_i flag is cleared.

Sleep mode

The sleep mode is used to communicate only between certain microcomputers when multiple microcomputers are connected through serial I/O.

The sleep mode is entered when the bit 7 of $UART_i$ transmit/receive mode register is set.

The operation of the sleep mode for an 8-bit asynchronous communication is described below.

When sleep mode is selected, the contents of the receive register is not transferred to the receive buffer register if bit 7 (bit 6 if 7-bit asynchronous communication and bit 8 if 9-bit asychronous communication) of the received data is "0". Also the RI_i, OER_i, FER_i, PER_i, and the SUM_i flag are unchanged. Therefore, the interrupt request bit of the $UART_i$ receive interrupt control register is also unchanged.

Normal receive operation takes place when bit 7 of the received data is "1".

The following is an example of how the sleep mode can be used.

The main microcomputer first sends data with bit 7 set to "1" and bits 0 to 6 set to the address of the subordinate microcomputer which wants to communicate with. Then all subordinate microcomputers receive the same data. Each subordinate microcomputer checks the received data, clears the sleep bit if bits 0 to 6 are its own address and sets the sleep bit if not. Next the main microcomputer sends data with bit 7 cleared. Then the microcomputer with the sleep bit cleared will receive the data, but the microcomputer with the sleep bit set will not. In this way, the main microcomputer is able to communicate with only the designated microcomputer.

(1) One-shot mode [00]

The A-D conversion pins are selected with the bit 0 to 2 of A-D control register. A-D conversion can be started by a software trigger or by an external trigger.

A software trigger is selected when the bit 5 of A-D control register is "0" and an external trigger is selected when it is "1".

When a software trigger is selected, A-D conversion is started when bit 6 (A-D conversion start flag) is set. A-D conversion ends after 57 ϕ_{AD} cycles and an interrupt request bit is set in the A-D conversion interrupt control register. At the same time, A-D control register bit 6 (A-D conversion start flag) is cleared and A-D conversion stops. The result of A-D conversion is stored in the A-D register corresponding to the selected pin.

If an external trigger is selected, A-D conversion starts when the A-D conversion start flag is "1" and the $\overline{AD_{TRG}}$ input changes from "H" to "L". In this case, the pins that can be used for A-D conversion are AN_0 to AN_6 because the $\overline{AD_{TRG}}$ pin is shared with the analog voltage input pin AN_7.

The operation is the same as with software trigger except that the A-D conversion start flag is not cleared after A-D conversion and a retrigger can be available during A-D conversion.

(2) Repeat mode [01]

The operation of this mode is the same as the operation of one-shot mode except that when A-D conversion of the selected pin is complete and the result is stored in the A-D register, conversion does not stop, but is repeated. Also, no interrupt request is issued in this mode. Furthermore, if software trigger is selected, the A-D conversion start flag is not cleared. The contents of the A-D register can be read at any time.

(3) Single sweep mode [10]

In the sweep mode, the number of analog input pins to be swept can be selected. Analog input pins are selected by bits 1 and 0 of the A-D sweep pin selection register ($1F_{16}$ address) shown in Figure 57. Two pins, four pins, six pins, or eight pins can be selected as analog input pins, depending on the contents of these bits.

A-D conversion is performed only for selected input pins. After A-D conversion is performed for input of AN_0 pin, the conversion result is stored in A-D register 0, and in the same way, A-D conversion is performed for selected pins one after another. After A-D conversion is performed for all selected pins, the sweep is stopped.

A-D conversion can be started with a software trigger or with an external trigger input. A software trigger is selected when bit 5 is "0" and an external trigger is selected when it is "1".

When a software trigger is selected, A-D conversion is started when A-D control register bit 6 (A-D conversion start flag) is set. When A-D conversion of all selected pins end, an interrupt request bit is set in the A-D conversion interrupt control register. At the same time, A-D control register bit 6 (A-D conversion start flag) is cleared and A-D conversion stops.

When an external trigger is selected, A-D conversion starts when the A-D conversion start flag is "1" and the $\overline{AD_{TRG}}$ input changes from "H" to "L". In this case, the A-D conversion result of the trigger input itself is stored in the A-D register 7 because the $\overline{AD_{TRG}}$ pin is shared with AN_7 pin.

The operation is the same as done by software trigger except that the A-D conversion start flag is not cleared after A-D conversion and a retrigger can be available during A-D conversion.

(4) Repeat sweep mode [11]

The difference with the single sweep mode is that A-D conversion does not stop after converting from the AN_0 pin to the selected pins, but repeats again from the AN_0 pin. The repeat is performed among the selected pins. Also, no interrupt request is generated. Furthermore, if software trigger is selected, the A-D conversion start flag is not cleared. The A-D register can be read at any time.

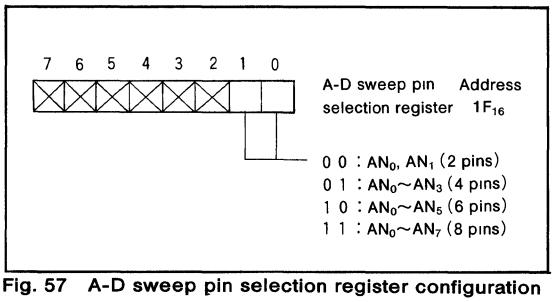


Fig. 57 A-D sweep pin selection register configuration

RESET CIRCUIT

Reset occurs when the $\overline{\text{RESET}}$ pin is returned to "H" level after holding it at "L" level when the power voltage is at 5V ±10%. Program execution starts at the address formed by setting the address pins A_{23} ~ A_{16} to 00_{16}, A_{15} ~ A_8 to the contents of address $FFFF_{16}$, and A_7 ~ A_0 to the contents of address $FFFE_{16}$.

Figure 60 shows the status of the internal registers when a reset occurs.

Figure 61 shows an example of a reset circuit. The reset input voltage must be held 0.9V or lower when the power voltage reaches 4.5V.

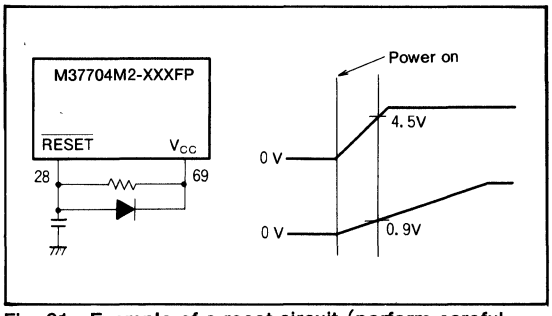


Fig. 61 Example of a reset circuit (perform careful evaluation at the system design level before using)

		Address	
(1)	Port P0 data directional register	(04_{16})···	00_{16}
(2)	Port P1 data directional register	(05_{16})···	00_{16}
(3)	Port P2 data directional register	(08_{16})···	00_{16}
(4)	Port P3 data directional register	(09_{16})···	××××0000
(5)	Port P4 data directional register	($0C_{16}$)···	00_{16}
(6)	Port P5 data directional register	($0D_{16}$)···	00_{16}
(7)	Port P6 data directional register	(10_{16})···	00_{16}
(8)	Port P7 data directional register	(11_{16})···	00_{16}
(9)	Port P8 data directional register	(14_{16})···	00_{16}
(10)	A-D control register	($1E_{16}$)···	00000???
(11)	A-D sweep pin selection register	($1F_{16}$)···	××××××11
(12)	UART 0 Transmit/Receive mode register	(30_{16})···	00_{16}
(13)	UART 1 Transmit/Receive mode register	(38_{16})···	00_{16}
(14)	UART 0 Transmit/Receive control register 0	(34_{16})···	××××1000
(15)	UART 1 Transmit/Receive control register 0	($3C_{16}$)···	××××1000
(16)	UART 0 Transmit/Receive control register 1	(35_{16})···	00000010
(17)	UART 1 Transmit/Receive control register 1	($3D_{16}$)···	00000010
(18)	Count start flag	(40_{16})···	00_{16}
(19)	One-shot start flag	(42_{16})···	0××00000
(20)	Up-down flag	(44_{16})···	00_{16}
(21)	Timer A0 mode register	(56_{16})···	00_{16}
(22)	Timer A1 mode register	(57_{16})···	00_{16}
(23)	Timer A2 mode register	(58_{16})···	00_{16}
(24)	Timer A3 mode register	(59_{16})···	00_{16}
(25)	Timer A4 mode register	($5A_{16}$)···	00_{16}
(26)	Timer B0 mode register	($5B_{16}$)···	00_{16}
(27)	Timer B1 mode register	($5C_{16}$)···	001×0000
(28)	Timer B2 mode register	($5D_{16}$)···	001×0000

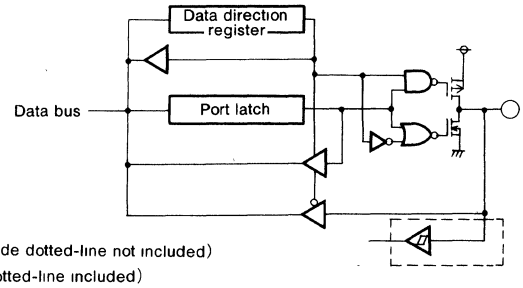
		Address	
(29)	Processor mode register	($5E_{16}$)···	00_{16}
(30)	Watchdog timer	(60_{16})···	FFF_{16}
(31)	Watchdog timer frequency selection flag	(61_{16})···	0
(32)	Waveform output mode register	(62_{16})···	00_{16}
(33)	A-D conversion interrupt control register	(70_{16})···	××××0000
(34)	UART 0 transmission interrupt control register	(71_{16})···	××××0000
(35)	UART 0 receive interrupt control register	(72_{16})···	××××0000
(36)	UART 1 transmission interrupt control register	(73_{16})···	××××0000
(37)	UART 1 receive interrupt control register	(74_{16})···	××××0000
(38)	Timer A0 interrupt control register	(75_{16})···	××××0000
(39)	Timer A1 interrupt control register	(76_{16})···	××××0000
(40)	Timer A2 interrupt control register	(77_{16})···	××××0000
(41)	Timer A3 interrupt control register	(78_{16})···	××××0000
(42)	Timer A4 interrupt control register	(79_{16})···	××××0000
(43)	Timer B0 interrupt control register	($7A_{16}$)···	××××0000
(44)	Timer B1 interrupt control register	($7B_{16}$)···	××××0000
(45)	Timer B2 interrupt control register	($7C_{16}$)···	××××0000
(46)	$\overline{\text{INT 0}}$ interrupt control register	($7D_{16}$)···	××00000
(47)	$\overline{\text{INT 1}}$ interrupt control register	($7E_{16}$)···	××00000
(48)	$\overline{\text{INT 2}}$ interrupt control register	($7F_{16}$)···	××00000
(49)	Processor status register PS		000??000??1??
(50)	Program bank register PG		00_{16}
(51)	Program counter PC_H		Content of $FFFF_{16}$
(52)	Program counter PC_L		Content of $FFFE_{16}$
(53)	Direct page register DPR		0000_{16}
(54)	Data bank register DT		00_{16}

Contents of other registers and RAM are not initialized and should be initialized by software.

Fig. 60 Microcomputer internal status during reset

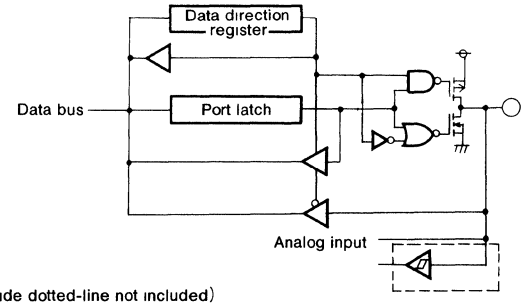
• Port $P0_7$~$P0_0$, $P1_7$~$P1_0$, $P2_7$~$P2_0$, $P3_3$~$P3_0$, $P4_6$~$P4_2$ (Inside dotted-line not included)

Port $P4_0$, $P4_1$, $P4_7$, $P5_7$, $P6_7$~$P6_1$, $P8_2$, $P8_6$ (Inside dotted-line included, but $P8_2$, $P8_6$ are without hysterisis)



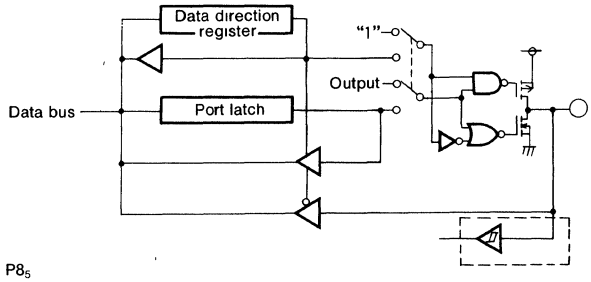
• Port $P7_6$~$P7_0$ (Inside dotted-line not included)

• Port $P7_7$ (Inside dotted-line included)

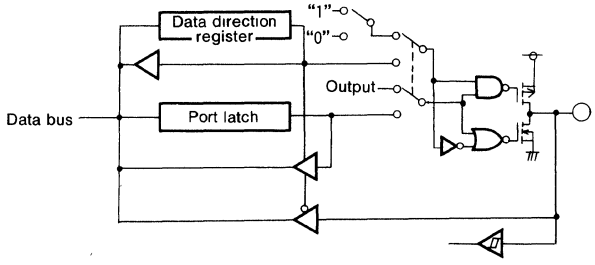


• Port $P8_3$, $P8_7$ (Inside dotted-line not included)

Port $P5_0$~$P5_6$, $P6_0$ (Inside dotted-line included)



• Port $P8_0$, $P8_1$, $P8_4$, $P8_5$



• $\overline{E}$

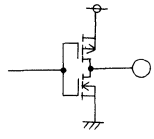


Fig. 62 Block diagram for ports P8 to P0 in single-chip mode and the $\overline{E}$ pin output

Port \ Mode	CM_1=0, CM_0=0 Single-chip Mode	CM_1=0, CM_0=1 Memory Expansion Mode	CM_1=1, CM_0=0 Microprocessor Mode	CM_1=1, CM_0=1 Evaluation Chip Mode
Port P0	$\overline{E}$; $P0_7$~$P0_0$: I/O Port	$\overline{E}$; $P0_7$~$P0_0$: Address A_7~A_0	Same as left	Same as left
Port P1 BYTE= "L"	$\overline{E}$; $P1_7$~$P1_0$: I/O Port	$\overline{E}$; A_{15}~A_8; $P1_7$~$P1_0$: Address / Data (odd)	Same as left	Same as left
Port P1 BYTE= "H" or $2 \cdot V_{CC}$ (Evaluation chip mode only)	$P1_7$~$P1_0$: I/O Port	$\overline{E}$; $P1_7$~$P1_0$: Address A_{15}~A_8	Same as left	$\overline{E}$; A_{15}~A_8: Address / Data (odd) Port P4, P5 and their direction registers are treated as 16-bit wide bus If BYTE=$2 \cdot V_{CC}$, the internal ROM area is also treated as 16-bit wide bus
Port P2 BYTE= "L"	$\overline{E}$; $P2_7$~$P2_0$: I/O Port	$\overline{E}$; A_{23}~A_{16}; $P2_7$~$P2_0$: Address / Data (even)	Same as left	Same as left
Port P2 BYTE= "H" or $2 \cdot V_{CC}$ (Evaluation chip mode only)		$\overline{E}$; A_{23}~A_{16}; $P2_7$~$P2_0$: Address / Data (even, odd)	Same as left	$\overline{E}$; A_{23}~A_{16}; $P2_7$~$P2_0$: Address / Data (even, odd) Same as for Port P1
Port P3	$\overline{E}$; $P3_3$~$P3_0$: I/O Port	$\overline{E}$; $P3_3$: $\overline{HLDA}$; $P3_2$: ALE; $P3_1$: $\overline{BHE}$; $P3_0$: R/$\overline{W}$	Same as left	Same as left
Port P4	$\overline{E}$; $P4_7$~$P4_0$: I/O Port * When processor mode register bit 7 = "0"	$\overline{E}$; $P4_7$~$P4_2$: I/O Port; $P4_1$: $\overline{RDY}$; $P4_0$: $\overline{HOLD}$ * When processor mode register bit 7 = "0"	—	$\overline{E}$; $P4_7$: $\overline{DBC}$; $P4_6$: VPA; $P4_5$: VDA; $P4_4$: QCL; $P4_3$: MX; $P4_2$: ϕ_1; $P4_1$: $\overline{RDY}$; $P4_0$: $\overline{HOLD}$
	$P4_2$: ϕ_1 Same as above except $P4_2$ * When processor mode register bit 7 = "1"	$P4_2$: ϕ_1 Same as above except $P4_2$ * When processor mode register bit 7 = "1"	Same as left in spite of processor mode register bit 7	

Fig. 64 Processor mode and ports P4 to P0 functions

ALE is an address latch enable signal used to latch the address signal from a multiplexed signal of address and data. The latch is transparent while ALE is "H" to let the address signal pass through and held while ALE is "L".

$\overline{HLDA}$ is a hold acknowledge signal and is used to notify externally when the microcomputer receives $\overline{HOLD}$ input and enters into hold state.

Ports $P4_0$ and $P4_1$ become $\overline{HOLD}$ and $\overline{RDY}$ input pin respectively and lose their output pin function, but the input pin function remains.

$\overline{HOLD}$ is a hold request signal. It is an input signal used to put the microcomputer in hold state. Ports P0, P1, P2, $P3_0$, and $P3_1$ are floating while the microcomputer stays in hold state.

$\overline{RDY}$ is a ready signal. If this signal goes "L", the internal clock ϕ stops at "L". When ϕ_1 output from port $P4_2$ is selected by setting bit 7 of processor mode register to "1", ϕ_1 output keeps on. $\overline{RDY}$ is used when slow external memory is attached.

(3) Microprocessor mode [10]

Microprocessor mode is entered by connecting the CNV_{SS} pin to V_{CC} and starting from reset. It can also be entered by programming the processor mode bits to "10" after connecting the CNV_{SS} pin to V_{SS} and starting from reset. This mode is similar to memory expansion mode except that internal ROM is disabled and an external memory is required, and ϕ_1 from port $P4_2$ is always output in spite of bit 7 of processor mode register.

(4) Evaluation chip mode [11]

Evaluation chip mode is entered by applying voltage twice the V_{CC} voltage to the CNV_{SS} pin. This mode is normally used for evaluation tools.

The functions of ports P0 and P3 are the same as in memory expansion mode.

Port P1 functions as an address output pin while $\overline{E}$ is "H" and as data I/O pin of odd addresses while $\overline{E}$ is "L" regardless of the BYTE pin level. However, if an internal memory is read, external data is ignored while $\overline{E}$ is "L".

Port P2 function as an address output pin while $\overline{E}$ is "H" and as data I/O pin of even addresses while $\overline{E}$ is "L" when the BYTE pin level is "L". However, if an internal memory is read, external data is ignored while $\overline{E}$ is "L".

When the BYTE pin level is "H" or $2 \cdot V_{CC}$, port P2 functions as an address output pin while $\overline{E}$ is "H" and as data I/O pin of even and odd addresses while $\overline{E}$ is "L". However, if an internal memory is read, external data is ignored while $\overline{E}$ is "L".

Port P4 and its data direction register which are located at address $0A_{16}$ and $0C_{16}$ are treated differently in evaluation chip mode. When these addresses are accessed, the data bus width is treated as 16 bits regardless of the BYTE pin level, and the access cycle is treated as internal memory regardless of the wait bit.

When a voltage twice the V_{CC} voltage is applied to the BYTE pin, the addresses corresponding to the internal ROM area are also treated as 16-bit data bus.

The functions of ports $P4_0$ and $P4_1$ are the same as in memory expansion mode.

Ports $P4_2$ to $P4_6$ become ϕ_1, MX, QCL, VDA, and VPA output pins respectively. Port $P4_7$ becomes the $\overline{DBC}$ input pin.

ϕ_1 from port $P4_2$ divided the clock to X_{IN} pin by 2 is always output in spite of bit 7 of processor mode register.

The MX signal normally contains the contents of flag m, but the contents of flag x is output if the CPU is using flag x.

QCL is the queue buffer clear signal. It becomes "H" when the instruction queue buffer is cleared, for example, when a jump instruction is executed.

VDA is the valid data address signal. It becomes "H" while the CPU is reading data from data buffer or writing data to data buffer. It also becomes "H" when the first byte of the instruction (operation code) is read from the instruction queue buffer.

VPA is the valid program address signal. It becomes "H" while the CPU is reading an instruction code from the instruction queue buffer.

$\overline{DBC}$ is the debug control signal and is used for debugging.

Table 5 shows the relationship between the CNV_{SS} pin input levels and processor modes.

Table 5. Relationship between the CNV_{SS} pin input levels and processor modes

CNV_{SS}	Mode	Description
V_{SS}	• Single-chip • Memory expansion • Microprocessor • Evaluation chip	Single-chip mode upon starting after reset. Other modes can be selected by changing the processor mode bit by software
V_{CC}	• Microprocessor • Evaluation chip	Microprocessor mode upon starting after reset Evaluation chip mode can be selected by changing the processor mode bit by software.
$2 \cdot V_{CC}$	• Evaluation chip	• Evaluation chip mode only.

ADDRESSING MODES

The M37704M2-XXXFP has 28 powerful addressing modes. Refer to the Series MELPS 7700 addressing mode description for the details of each addressing mode.

MACHINE INSTRUCTION LIST

The M37704M2-XXXFP has 103 machine instructions. Refer to the Series MELPS 7700 machine instruction list for details.

DATA REQUIRED FOR MASK ORDERING

Please send the following data for mask orders.

(1) M37704M2-XXXFP mask ROM order confirmation form
(2) Mark specification form for 80P6N
(3) ROM data (EPROM 3 sets)

M37704M2-XXXFP

ELECTRICAL CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P3_3$, $P4_0$~$P4_7$, $P5_0$~$P5_7$, $P6_0$~$P6_7$, $P7_0$~$P7_7$, $P8_0$~$P8_7$	I_{OH}=−10mA	3			V
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P3_3$	I_{OH}=−400μA	4.7			V
V_{OH}	High-level output voltage $P3_2$	I_{OH}=−10mA	3.1			V
		I_{OH}=−400μA	4.8			
V_{OH}	High-level output voltage $\overline{E}$	I_{OH}=−10mA	3.4			V
		I_{OH}=−400μA	4.8			
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P3_3$, $P4_0$~$P4_7$, $P5_6$, $P5_7$, $P6_0$~$P6_7$, $P7_0$~$P7_7$, $P8_0$~$P8_7$	I_{OL}=10mA			2	V
V_{OL}	Low-level output voltage $P5_0$~$P5_5$	I_{OL}=20mA			2	V
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P3_3$	I_{OL}=2mA			0.45	V
V_{OL}	Low-level output voltage $P3_2$	I_{OL}=10mA			1.9	V
		I_{OL}=2mA			0.43	
V_{OL}	Low-level output voltage $\overline{E}$	I_{OL}=10mA			1.6	V
		I_{OL}=2mA			0.4	
V_{T+}−V_{T-}	Hysteresis $\overline{HOLD}$, $\overline{RDY}$, $TA0_{IN}$~$TA4_{IN}$, $TB0_{IN}$~$TB2_{IN}$, $\overline{INT_0}$~$\overline{INT_2}$, $\overline{AD_{TRG}}$, $\overline{CTS_0}$, $\overline{CTS_1}$, CLK_0, CLK_1		0.4		1	V
V_{T+}−V_{T-}	Hysteresis $\overline{RESET}$		0.2		0.5	V
V_{T+}−V_{T-}	Hysteresis X_{IN}		0.1		0.3	V
I_{IH}	High-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_3$, $P4_0$~$P4_7$, $P5_0$~$P5_7$, $P6_0$~$P6_7$, $P7_0$~$P7_7$, $P8_0$~$P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=5V			5	μA
I_{IL}	Low-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_3$, $P4_0$~$P4_7$, $P5_0$~$P5_7$, $P6_0$~$P6_7$, $P7_0$~$P7_7$, $P8_0$~$P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=0V			−5	μA
V_{RAM}	RAM hold voltage	When clock is stopped	2			V
I_{CC}	Power supply current	In single-chip mode output only pin is open and other pins are V_{SS} during reset — f(X_{IN})=8MHz, square waveform		6	12	mA
		T_a=25℃ when clock is stopped.			1	μA
		T_a=70℃ when clock is stopped.			10	

A-D CONVERTER CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
—	Resolution	V_{REF}=V_{CC}			8	Bits
—	Absolute accuracy	V_{REF}=V_{CC}			±3	LSB
R_{LADDER}	Ladder resistance	V_{REF}=V_{CC}	2		10	kΩ
t_{CONV}	Conversion time		28.5			μs
V_{REF}	Reference voltage		2		V_{CC}	V
V_{IA}	Analog input voltage		0		V_{REF}	V

Timer A input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		250			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		125			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		125			ns

Timer A input (Gating input in timer mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		1000			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		500			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		500			ns

Timer A input (External trigger input in one-shot pulse mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		500			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		250			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		250			ns

Timer A input (External trigger input in pulse width modulation mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		250			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		250			ns

Timer A input (Up-down input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(UP)}$	TAi_{OUT} input cycle time		5000			ns
$t_{W(UPH)}$	TAi_{OUT} input high-level pulse width		2500			ns
$t_{W(UPL)}$	TAi_{OUT} input low-level pulse width		2500			ns
$t_{SU(UP-T_{IN})}$	TAi_{OUT} input setup time		1000			ns
$t_{h(T_{IN}-UP)}$	TAi_{OUT} input hold time		1000			ns

SWITCHING CHARACTERISTICS (V_{CC}=5V±10%, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{d(E-P0Q)}$	Port P0 data output delay time	Figure 70			200	ns
$t_{d(E-P1Q)}$	Port P1 data output delay time				200	ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				200	ns
$t_{d(E-P3Q)}$	Port P3 data output delay time				200	ns
$t_{d(E-P4Q)}$	Port P4 data output delay time				200	ns
$t_{d(E-P5Q)}$	Port P5 data output delay time				200	ns
$t_{d(E-P6Q)}$	Port P6 data output delay time				200	ns
$t_{d(E-P7Q)}$	Port P7 data output delay time				200	ns
$t_{d(E-P8Q)}$	Port P8 data output delay time				200	ns

Memory expansion mode and microprocessor mode (when wait bit = "1")

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 70	100			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				120	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				5	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		100			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				120	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				5	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		100			ns
$t_{d(E-HLDA)}$	$\overline{HLDA}$ output delay time				100	ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		100			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		100			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		100			ns
$t_{d(E-\phi_1)}$	ϕ_1 output delay time		0		30	ns
$t_{h(E-P0A)}$	Port P0 address hold time		50			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		50			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		50			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		50			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		50			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		50			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		220			ns

M37704M2AXXXFP

ELECTRICAL CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min.	Limits Typ.	Limits Max.	Unit
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P3_3$, $P4_0$~$P4_7$, $P5_0$~$P5_7$, $P6_0$~$P6_7$, $P7_0$~$P7_7$, $P8_0$~$P8_7$	I_{OH}=−10mA	3			V
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P3_3$	I_{OH}=−400μA	4.7			V
V_{OH}	High-level output voltage $P3_2$	I_{OH}=−10mA	3.1			V
		I_{OH}=−400μA	4.8			
V_{OH}	High-level output voltage $\overline{E}$	I_{OH}=−10mA	3.4			V
		I_{OH}=−400μA	4.8			
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P3_3$, $P4_0$~$P4_7$, $P5_6$, $P5_7$, $P6_0$~$P6_7$, $P7_0$~$P7_7$, $P8_0$~$P8_7$	I_{OL}=10mA			2	V
V_{OL}	Low-level output voltage $P5_0$~$P5_5$	I_{OL}=20mA			2	V
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P3_3$	I_{OL}=2mA			0.45	V
V_{OL}	Low-level output voltage $P3_2$	I_{OL}=10mA			1.9	V
		I_{OL}=2mA			0.43	
V_{OL}	Low-level output voltage $\overline{E}$	I_{OL}=10mA			1.6	V
		I_{OL}=2mA			0.4	
$V_{T+}-V_{T-}$	Hysteresis $\overline{HOLD}$, $\overline{RDY}$, $TA0_{IN}$~$TA4_{IN}$, $TB0_{IN}$~$TB2_{IN}$, $\overline{INT_0}$~$\overline{INT_2}$, $\overline{AD_{TRG}}$, $\overline{CTS_0}$, $\overline{CTS_1}$, CLK_0, CLK_1		0.4		1	V
$V_{T+}-V_{T-}$	Hysteresis $\overline{RESET}$		0.2		0.5	V
$V_{T+}-V_{T-}$	Hysteresis X_{IN}		0.1		0.3	V
I_{IH}	High-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_3$, $P4_0$~$P4_7$, $P5_0$~$P5_7$, $P6_0$~$P6_7$, $P7_0$~$P7_7$, $P8_0$~$P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=5V			5	μA
I_{IL}	Low-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_3$, $P4_0$~$P4_7$, $P5_0$~$P5_7$, $P6_0$~$P6_7$, $P7_0$~$P7_7$, $P8_0$~$P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=0V			−5	μA
V_{RAM}	RAM hold voltage	When clock is stopped.	2			V
I_{CC}	Power supply current	In single-chip mode output only pin is open and other pins are V_{SS} during reset. f(X_{IN})=16MHz, square waveform		12	24	mA
		T_a=25℃ when clock is stopped.			1	μA
		T_a=70℃ when clock is stopped.			10	

A-D CONVERTER CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min.	Limits Typ.	Limits Max.	Unit
—	Resolution	V_{REF}=V_{CC}			8	Bits
—	Absolute accuracy	V_{REF}=V_{CC}			±3	LSB
R_{LADDER}	Ladder resistance	V_{REF}=V_{CC}	2		10	kΩ
t_{CONV}	Conversion time		14.25			μs
V_{REF}	Reference voltage		2		V_{CC}	V
V_{IA}	Analog input voltage		0		V_{REF}	V

Timer A input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		125			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		62			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		62			ns

Timer A input (Gating input in timer mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		500			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		250			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		250			ns

Timer A input (External trigger input in one-shot pulse mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		250			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		125			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		125			ns

Timer A input (External trigger input in pulse width modulation mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		125			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		125			ns

Timer A input (Up-down input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(UP)}$	TAi_{OUT} input cycle time		2500			ns
$t_{W(UPH)}$	TAi_{OUT} input high-level pulse width		1250			ns
$t_{W(UPL)}$	TAi_{OUT} input low-level pulse width		1250			ns
$t_{SU(UP-T_{IN})}$	TAi_{OUT} input setup time		500			ns
$t_{h(T_{IN}-UP)}$	TAi_{OUT} input hold time		500			ns

SWITCHING CHARACTERISTICS (V_{CC}=5V±10%, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

Single-chip mode

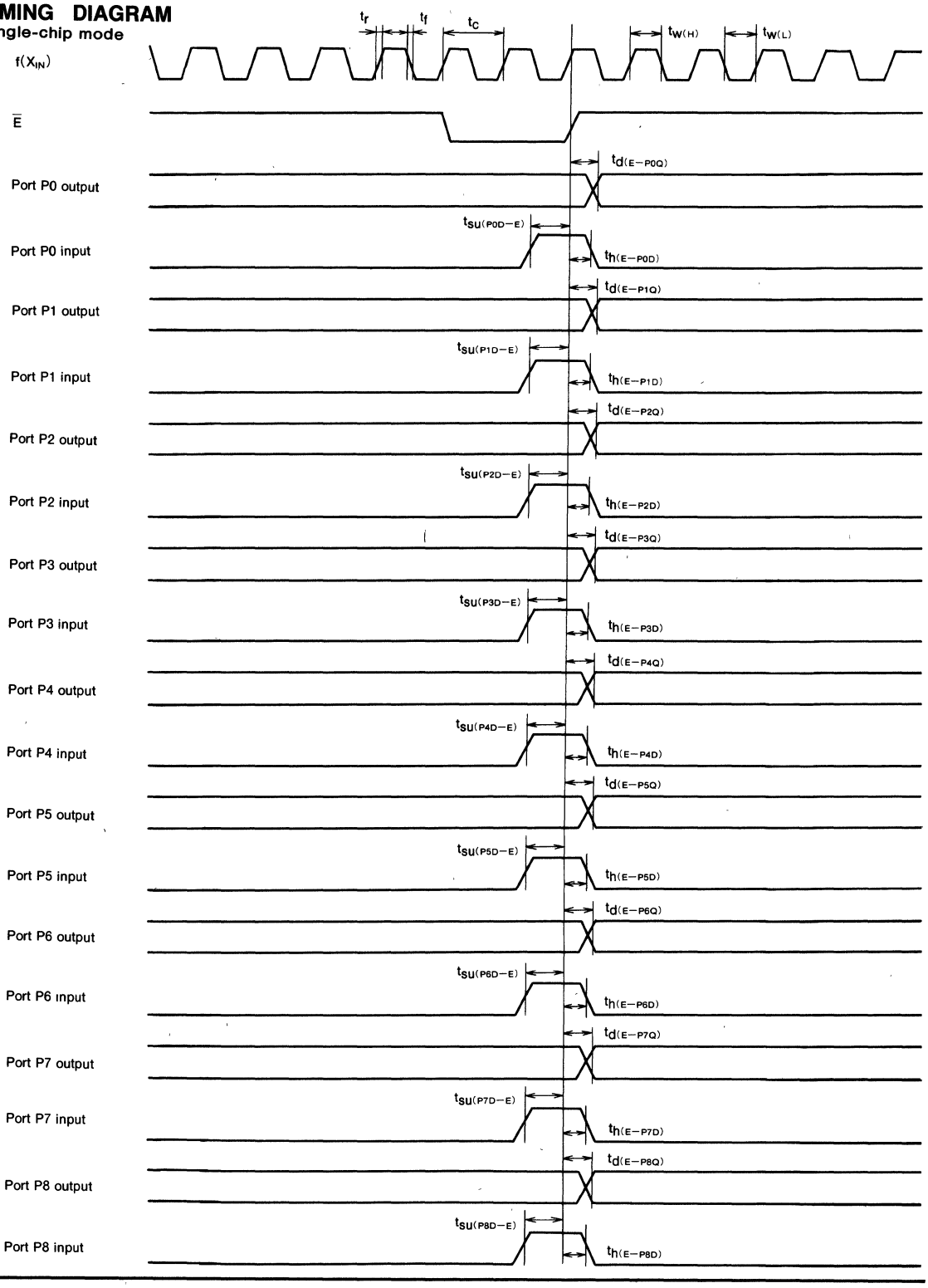
Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{d(E-P0Q)}$	Port P0 data output delay time	Figure 70			100	ns
$t_{d(E-P1Q)}$	Port P1 data output delay time				100	ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				100	ns
$t_{d(E-P3Q)}$	Port P3 data output delay time				100	ns
$t_{d(E-P4Q)}$	Port P4 data output delay time				100	ns
$t_{d(E-P5Q)}$	Port P5 data output delay time				100	ns
$t_{d(E-P6Q)}$	Port P6 data output delay time				100	ns
$t_{d(E-P7Q)}$	Port P7 data output delay time				100	ns
$t_{d(E-P8Q)}$	Port P8 data output delay time				100	ns

Memory expansion mode and microprocessor mode (when wait bit = "1")

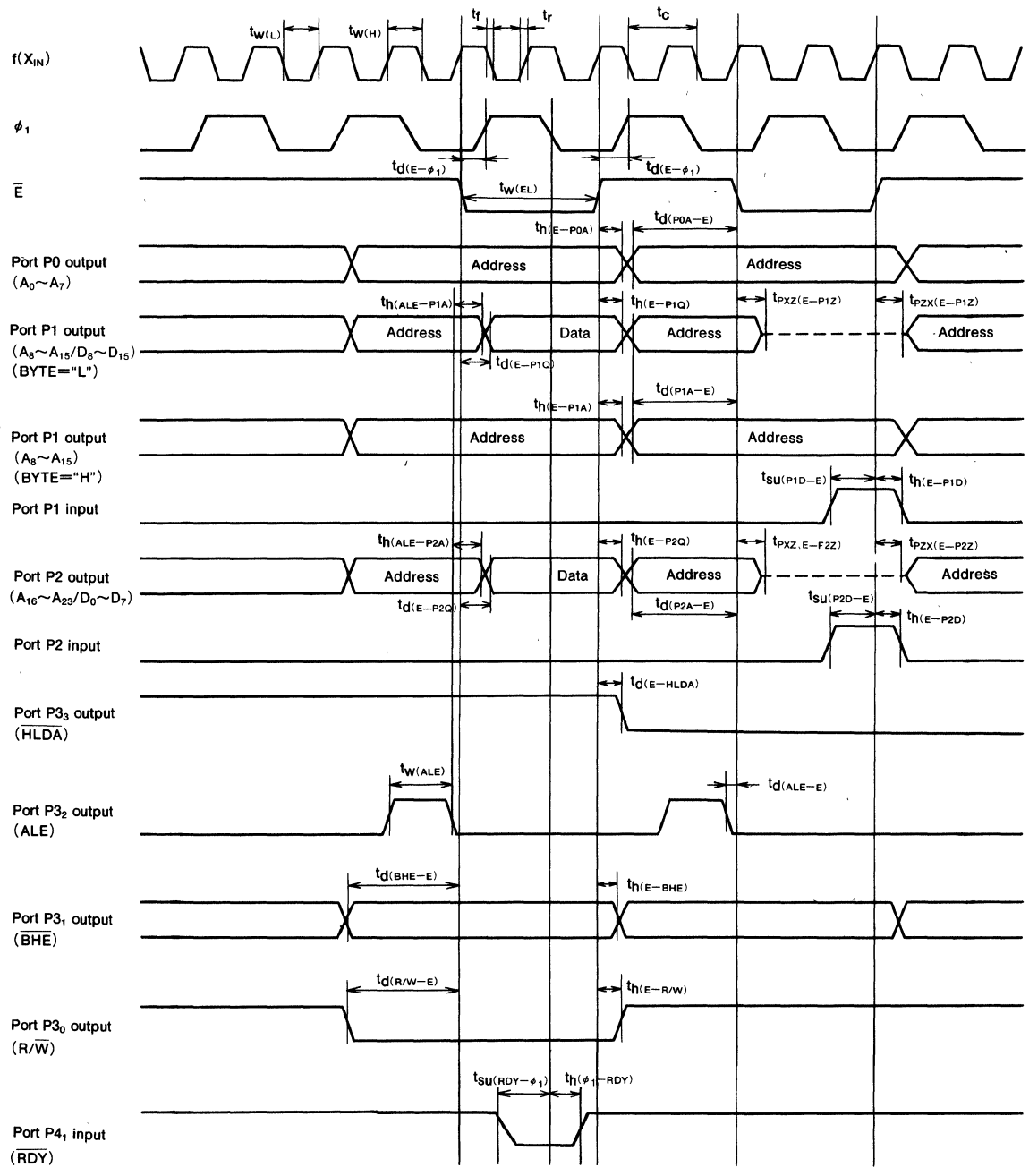
Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 70	30			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				80	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				5	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		30			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				80	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				5	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		30			ns
$t_{d(E-HLDA)}$	$\overline{HLDA}$ output delay time				50	ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		40			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		30			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		30			ns
$t_{d(E-\phi_1)}$	ϕ_1 output delay time		0		20	ns
$t_{h(E-P0A)}$	Port P0 address hold time		25			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		25			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		25			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		25			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		25			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		25			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		95			ns

TIMING DIAGRAM

Single-chip mode



Memory expansion mode and microprocessor mode (When wait bit="1")



Test conditions

- V_{CC}= 5 V±10%
- Output timing voltage : V_{OL}=0.8V, V_{OH}=2.0V
- Ports P1,P2 input : V_{IL}=0.8V, V_{IH}=2.5V
- Port $P4_1$ input : V_{IL}=1.0V, V_{IH}=4.0V

M37705M2-XXXSP, M37705M2AXXXSP
M37705S1SP, M37705S1ASP

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER

DESCRIPTION

The M37705M2-XXXSP, M37705M2AXXXSP, M37705S1SP, and M37705S1ASP are single-chip microcomputers designed with high-performance CMOS silicon gate technology. These are housed in a 64-pin shrink plastic molded DIP. These single-chip microcomputers have a large 16M bytes address space, three instruction queue buffers, and two data buffers for high-speed instruction execution. The CPU is a 16-bit parallel processor that can also be switched to perform 8-bit parallel processing. These microcomputers are suitable for office, business and industrial equipment controller that require high-speed processing of large data.

Also, the incorporated motor control circuit makes these microcomputers suitable for control of equipment that requires motor control.

The differences between M37705M2-XXXSP, M37705M2AXXXSP, M37705S1SP and M37705S1ASP are the ROM size and the external clock input frequency as shown below. Therefore, the following descriptions will be for the M37705M2-XXXFP unless otherwise noted.

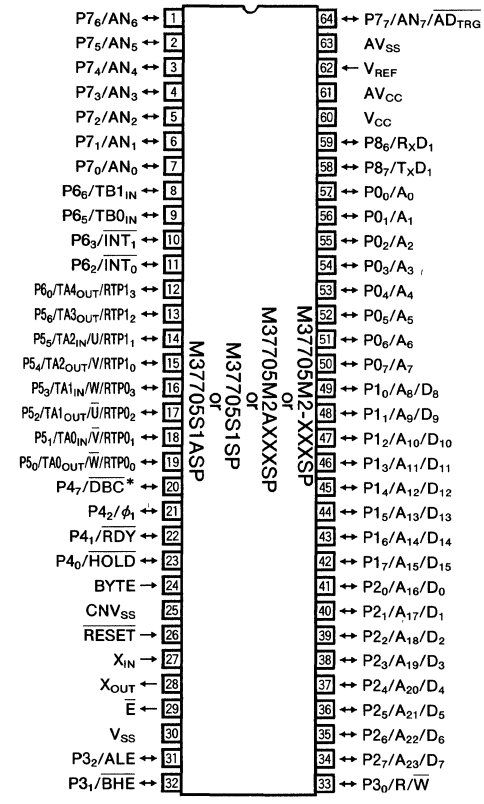
Type name	ROM size	External clock input frequency
M37705M2-XXXSP	16K bytes	8 MHz
M37705M2AXXXSP	16K bytes	16MHz
M37705S1SP	External	8 MHz
M37705S1ASP	External	16MHz

The M37705M2-XXXSP cuts down the pins of M37704M2-XXXFP. Refer to the section on M37704M2-XXXFP for the functional differences.

DISTINCTIVE FEATURES

- Number of basic instructions 103
- Memory size ROM ..... 16K bytes
  RAM ..... 512 bytes
- Instruction execution time
  M37705M2-XXXSP, M37705S1SP
  (The fastest instruction at 8 MHz frequency) 500ns
  M37705M2AXXXSP, M37705S1ASP
  (The fastest instruction at 16 MHz frequency) 250ns
- Single power supply ..... 5V±10%
- Low power dissipation (at 8 MHz frequency)
  ..... 30mW (Typ.)
- Interrupts ..... 16 types 7 levels
- Multiple function 16-bit timer ..... 5+3
  (Three-phase motor drive waveform or pulse motor drive waveform can be output.)
- UART 1
- 8-bit A-D converter ..... 8-channel inputs
- 12-bit watchdog timer
- Programmable input/output
  (ports P0, P1, P2, P3, P4, P5, P6, P7, P8) 53

PIN CONFIGURATION (TOP VIEW)



Outline 64P4B

*: Used in the evaluation chip mode only

APPLICATION

Control devices for office equipment such as copiers, printers, typewriters, facsimiles, word processors, and personal computers

Control devices for industrial equipment such as ME, NC, general purpose inverter, and measuring instruments

FUNCTIONS OF M37705M2-XXXSP

Parameter		Functions
Number of basic instructions		103
Instruction execution time	M37705M2-XXXSP, M37705S1SP	500ns (the fastest instructions, at 8MHz frequency)
	M37705M2AXXXSP, M37705S1ASP	250ns (the fastest instructions, at 16MHz frequency)
Memory size	ROM	16K bytes
	RAM	512 bytes
Input/Output ports	P0, P1, P2, P7	8-bit× 4
	P5	7-bit× 1
	P6	5-bit× 1
	P4	4-bit× 1
	P3	3-bit× 1
	P8	2-bit× 1
Multi-function timers	TA0, TA1, TA2, TA3, TA4	16-bit× 5 (3 input/output and 2 output functions)
	TB0, TB1, TB2	16-bit× 3 (2 input functions)
Serial I/O		UART ×1
A-D converter		8-bit× 1 (8 channels)
Watchdog timer		12-bit× 1
Dead-time timer		8-bit× 3
Interrupts		2 external types, 14 internal types (Each interrupt can be set the priority levels to 0 ~ 7)
Clock generating circuit		Built-in(externally connected to a ceramic resonator or quartz crystal resonator)
Supply voltage		5 V±10%
Power dissipation		30mW(at external 8 MHz frequency)
Input/Output characteristic	Input/Output voltage	5 V
	Output current	5 mA
Memory expansion		Maximum 16M bytes
Operating temperature range		−10~70℃
Device structure		CMOS high-performance silicon gate process
Package		64-pin shrink plastic molded DIP

BASIC FUNCTION BLOCKS

The functional differences between the M37705M2-XXXSP and M37704M2-XXXFP are described below. The M37705M2-XXXSP has the same functions as the M37704M2-XXXFP, except these points. Refer to the section on the M37704M2-XXXFP.

TIMER

Since timers A3 and A4 have no input pin, timers A3 and A4 operate only in the modes except for event counter mode and select only no input function by timer A3 and timer A4 mode register.

Since timer B2 has no input pin, timer B2 operates only in timer mode. Therefore, only clock source can be selected by the bits 7 and 6 of timer B2 mode register. The bits of timer mode register must be "0" except for the clock source selection bits. Other timers A0, A1, A2, B0 and B1 have the same functions as the M37704M2-XXXFP.

SERIAL I/O

Serial I/O is only UART1. UART1 has only the asynchronous serial communication function and no clock synchronous serial communication function. Therefore, do not select the clock synchronous serial communication function ("001") by the serial communication method selection bits (bits 2, 1 and 0) of UART1 transmit/receive mode register. Since UART1 does not have the functions of $\overline{CTS}$ and $\overline{RTS}$, the $\overline{CTS}$, $\overline{RTS}$ selection bit (bit 2) of UART1 transmit/receive control register must always be "1".

Since UART0 has no function as serial I/O, set all the serial communication method selection bits (bits 2, 1 and 0) of UART0 transmit/receive mode register to "0".

INPUT/OUTPUT PINS

Though the port registers and directional registers for ports P4, P5, P6 and P8 have eight bits, the directional register bits having no pins must always be set to the output mode. Since port $P3_3$ is not available as a pin although it has port register and directional register, port $P3_3$ must be set to the output mode.

ADDRESSING MODES

The M37705M2-XXXSP has 28 powerful addressing modes. Refer to the Series MELPS 7700 addressing mode description for the details of each addressing mode.

MACHINE INSTRUCTION LIST

The M37705M2-XXXSP has 103 machine instructions. Refer to the Series MELPS 7700 machine instruction list for details.

DATA REQUIRED FOR MASK ORDERING

Please send the following data for mask orders.

(1) M37705M2-XXXSP mask ROM order confirmation form
(2) Mark specification form for 64P4B
(3) ROM data (EPROM 3 sets)

The functional differences between the M37705M2-XXXSP and M37704M2-XXXFP

Parameter	M37705M2-XXXSP	M37704M2-XXXFP
I/O port	P0, P1, P2, P7 ····· 8-bit× 4 P5 ····· 7-bit× 1 P6 ····· 5-bit× 1 P4 ····· 4-bit× 1 P3 ····· 3-bit× 1 P8 ····· 2-bit× 1 (without $\overline{HLDA}$ pin)	P0~P2, P4~P8 ····· 8-bit× 8 P3 ····· 4-bit× 1 (with $\overline{HLDA}$ pin)
Timer	Timer A ····· with I/O pins 16-bit× 3 with output pins 16-bit× 2	Timer A ····· with I/O pins 16-bit× 5
	Timer B ····· with input pins 16-bit× 2 only timer mode 16-bit× 1	Timer B ····· with input pins 16-bit× 3
Serial I/O	UART (no clock synchronous serial I/O)× 1	(UART or clock synchronous serial I/O)× 2
Package	64-pin shrink plastic molded DIP	80-pin plastic molded QFP

M37705M2-XXXSP

ELECTRICAL CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min	Limits Typ	Limits Max.	Unit
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$~$P7_7$, $P8_6$, $P8_7$	I_{OH}=−10mA	3			V
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$	I_{OH}=−400μA	4.7			V
V_{OH}	High-level output voltage $P3_2$	I_{OH}=−10mA	3.1			V
		I_{OH}=−400μA	4.8			
V_{OH}	High-level output voltage $\overline{E}$	I_{OH}=−10mA	3.4			V
		I_{OH}=−400μA	4.8			
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P4_0$~$P4_2$, $P4_7$, $P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$~$P7_7$, $P8_6$, $P8_7$	I_{OL}=10mA			2	V
V_{OL}	Low-level output voltage $P5_0$~$P5_5$	I_{OL}=20mA			2	V
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$	I_{OL}=2mA			0.45	V
V_{OL}	Low-level output voltage $P3_2$	I_{OL}=10mA			1.9	V
		I_{OL}=2mA			0.43	
V_{OL}	Low-level output voltage $\overline{E}$	I_{OL}=10mA			1.6	V
		I_{OL}=2mA			0.4	
$V_{T+}-V_{T-}$	Hysteresis $\overline{HOLD}$, $\overline{RDY}$, $TA0_{IN}$~$TA2_{IN}$, $TB0_{IN}$, $TB1_{IN}$, $\overline{INT_0}$, $\overline{INT_1}$, AD_{TRG}		0.4		1	V
$V_{T+}-V_{T-}$	Hysteresis $\overline{RESET}$		0.2		0.5	V
$V_{T+}-V_{T-}$	Hysteresis X_{IN}		0.1		0.3	V
I_{IH}	High-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_2$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$~$P7_7$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=5V			5	μA
I_{IL}	Low-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_2$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$~$P7_7$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=0V			−5	μA
V_{RAM}	RAM hold voltage	When clock is stopped	2			V
I_{CC}	Power supply current	In single-chip mode output only pin is open and other pins are V_{SS} during reset; f(X_{IN})=8MHz, square waveform		6	12	mA
		T_a=25℃ when clock is stopped.			1	μA
		T_a=70℃ when clock is stopped.			10	

A-D CONVERTER CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min.	Limits Typ	Limits Max.	Unit
—	Resolution	V_{REF}=V_{CC}			8	Bits
—	Absolute accuracy	V_{REF}=V_{CC}			±3	LSB
R_{LADDER}	Ladder resistance	V_{REF}=V_{CC}	2		10	kΩ
t_{CONV}	Conversion time		28.5			μs
V_{REF}	Reference voltage		2		V_{CC}	V
V_{IA}	Analog input voltage		0		V_{REF}	V

Timer A input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		250			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		125			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		125			ns

Timer A input (Gating input in timer mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{C(TA)}$	TAi_{IN} input cycle time		1000			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		500			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		500			ns

Timer A input (External trigger input in one-shot pulse mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max	
$t_{C(TA)}$	TAi_{IN} input cycle time		500			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		250			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		250			ns

Timer A input (External trigger input in pulse width modulation mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		250			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		250			ns

Timer A input (Up-down input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max	
$t_{C(UP)}$	TAi_{OUT} input cycle time		5000			ns
$t_{W(UPH)}$	TAi_{OUT} input high-level pulse width		2500			ns
$t_{W(UPL)}$	TAi_{OUT} input low-level pulse width		2500			ns
$t_{SU(UP-T_{IN})}$	TAi_{OUT} input setup time		1000			ns
$t_{h(T_{IN}-UP)}$	TAi_{OUT} input hold time		1000			ns

SWITCHING CHARACTERISTICS (V_{CC}=5V±10%, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
$t_{d(E-P0Q)}$	Port P0 data output delay time	Figure 1			200	ns
$t_{d(E-P1Q)}$	Port P1 data output delay time				200	ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				200	ns
$t_{d(E-P3Q)}$	Port P3 data output delay time				200	ns
$t_{d(E-P4Q)}$	Port P4 data output delay time				200	ns
$t_{d(E-P5Q)}$	Port P5 data output delay time				200	ns
$t_{d(E-P6Q)}$	Port P6 data output delay time				200	ns
$t_{d(E-P7Q)}$	Port P7 data output delay time				200	ns
$t_{d(E-P8Q)}$	Port P8 data output delay time				200	ns

Memory expansion mode and microprocessor mode (when wait bit = "1")

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 1	100			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				120	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				5	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		100			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				120	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				5	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		100			ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		100			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		100			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		100			ns
$t_{d(E-\phi_1)}$	ϕ_1 output delay time		0		30	ns
$t_{h(E-P0A)}$	Port P0 address hold time		50			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		50			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		50			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		50			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		50			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		50			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		220			ns

M37705M2AXXXSP

ELECTRICAL CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min.	Limits Typ.	Limits Max.	Unit
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$~$P7_7$, $P8_6$, $P8_7$	I_{OH}=−10mA	3			V
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$	I_{OH}=−400μA	4.7			V
V_{OH}	High-level output voltage $P3_2$	I_{OH}=−10mA	3.1			V
		I_{OH}=−400μA	4.8			
V_{OH}	High-level output voltage $\overline{E}$	I_{OH}=−10mA	3.4			V
		I_{OH}=−400μA	4.8			
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P4_0$~$P4_2$, $P4_7$, $P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$~$P7_7$, $P8_6$, $P8_7$	I_{OL}=10mA			2	V
V_{OL}	Low-level output voltage $P5_0$~$P5_5$	I_{OL}=20mA			2	V
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$	I_{OL}=2mA			0.45	V
V_{OL}	Low-level output voltage $P3_2$	I_{OL}=10mA			1.9	V
		I_{OL}=2mA			0.43	
V_{OL}	Low-level output voltage $\overline{E}$	I_{OL}=10mA			1.6	V
		I_{OL}=2mA			0.4	
V_{T+}−V_{T-}	Hysteresis $\overline{HOLD}$, $\overline{RDY}$, $TA0_{IN}$~$TA2_{IN}$, $TB0_{IN}$, $TB1_{IN}$, $\overline{INT_0}$, $\overline{INT_1}$, $\overline{AD_{TRG}}$		0.4		1	V
V_{T+}−V_{T-}	Hysteresis $\overline{RESET}$		0.2		0.5	V
V_{T+}−V_{T-}	Hysteresis X_{IN}		0.1		0.3	V
I_{IH}	High-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_2$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$~$P7_7$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=5V			5	μA
I_{IL}	Low-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_2$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$~$P7_7$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=0V			−5	μA
V_{RAM}	RAM hold voltage	When clock is stopped	2			V
I_{CC}	Power supply current	In single-chip mode output only pin is open and other pins are V_{SS} during reset — f(X_{IN})=16MHz, square waveform		12	24	mA
		In single-chip mode output only pin is open and other pins are V_{SS} during reset — T_a=25℃ when clock is stopped			1	μA
		In single-chip mode output only pin is open and other pins are V_{SS} during reset — T_a=70℃ when clock is stopped.			10	

A-D CONVERTER CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min	Limits Typ	Limits Max	Unit
—	Resolution	V_{REF}=V_{CC}			8	Bits
—	Absolute accuracy	V_{REF}=V_{CC}			±3	LSB
R_{LADDER}	Ladder resistance	V_{REF}=V_{CC}	2		10	kΩ
t_{CONV}	Conversion time		14.25			μs
V_{REF}	Reference voltage		2		V_{CC}	V
V_{IA}	Analog input voltage		0		V_{REF}	V

Timer A input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		125			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		62			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		62			ns

Timer A input (Gating input in timer mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		500			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		250			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		250			ns

Timer A input (External trigger input in one-shot pulse mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		250			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		125			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		125			ns

Timer A input (External trigger input in pulse width modulation mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max	
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		125			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		125			ns

Timer A input (Up-down input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
$t_{C(UP)}$	TAi_{OUT} input cycle time		2500			ns
$t_{W(UPH)}$	TAi_{OUT} input high-level pulse width		1250			ns
$t_{W(UPL)}$	TAi_{OUT} input low-level pulse width		1250			ns
$t_{SU(UP-T_{IN})}$	TAi_{OUT} input setup time		500			ns
$t_{h(T_{IN}-UP)}$	TAi_{OUT} input hold time		500			ns

SWITCHING CHARACTERISTICS (V_{CC}=5V±10%, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

Single-chip mode

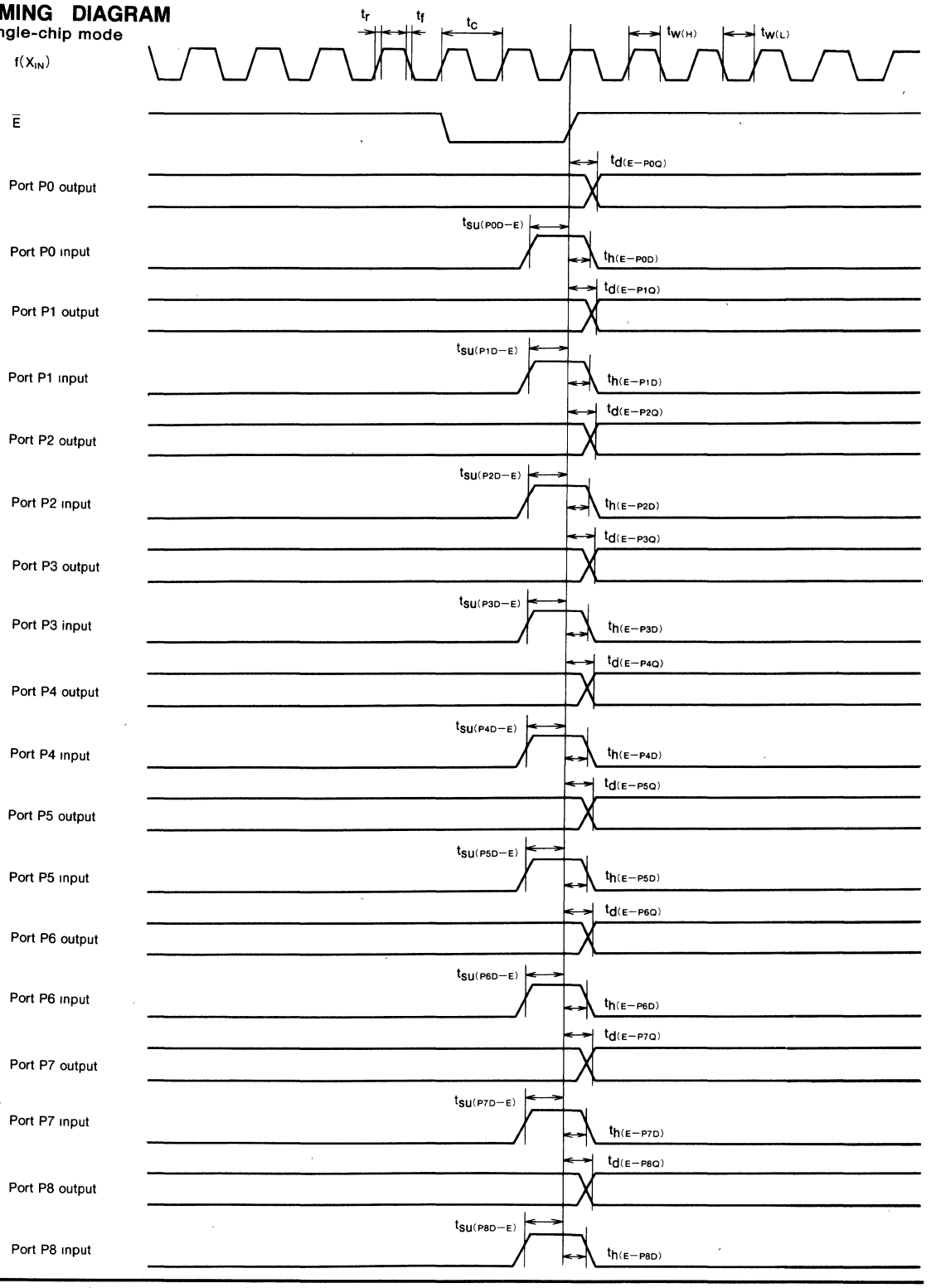
Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{d(E-P0Q)}$	Port P0 data output delay time	Figure 1			100	ns
$t_{d(E-P1Q)}$	Port P1 data output delay time				100	ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				100	ns
$t_{d(E-P3Q)}$	Port P3 data output delay time				100	ns
$t_{d(E-P4Q)}$	Port P4 data output delay time				100	ns
$t_{d(E-P5Q)}$	Port P5 data output delay time				100	ns
$t_{d(E-P6Q)}$	Port P6 data output delay time				100	ns
$t_{d(E-P7Q)}$	Port P7 data output delay time				100	ns
$t_{d(E-P8Q)}$	Port P8 data output delay time				100	ns

Memory expansion mode and microprocessor mode (when wait bit = "1")

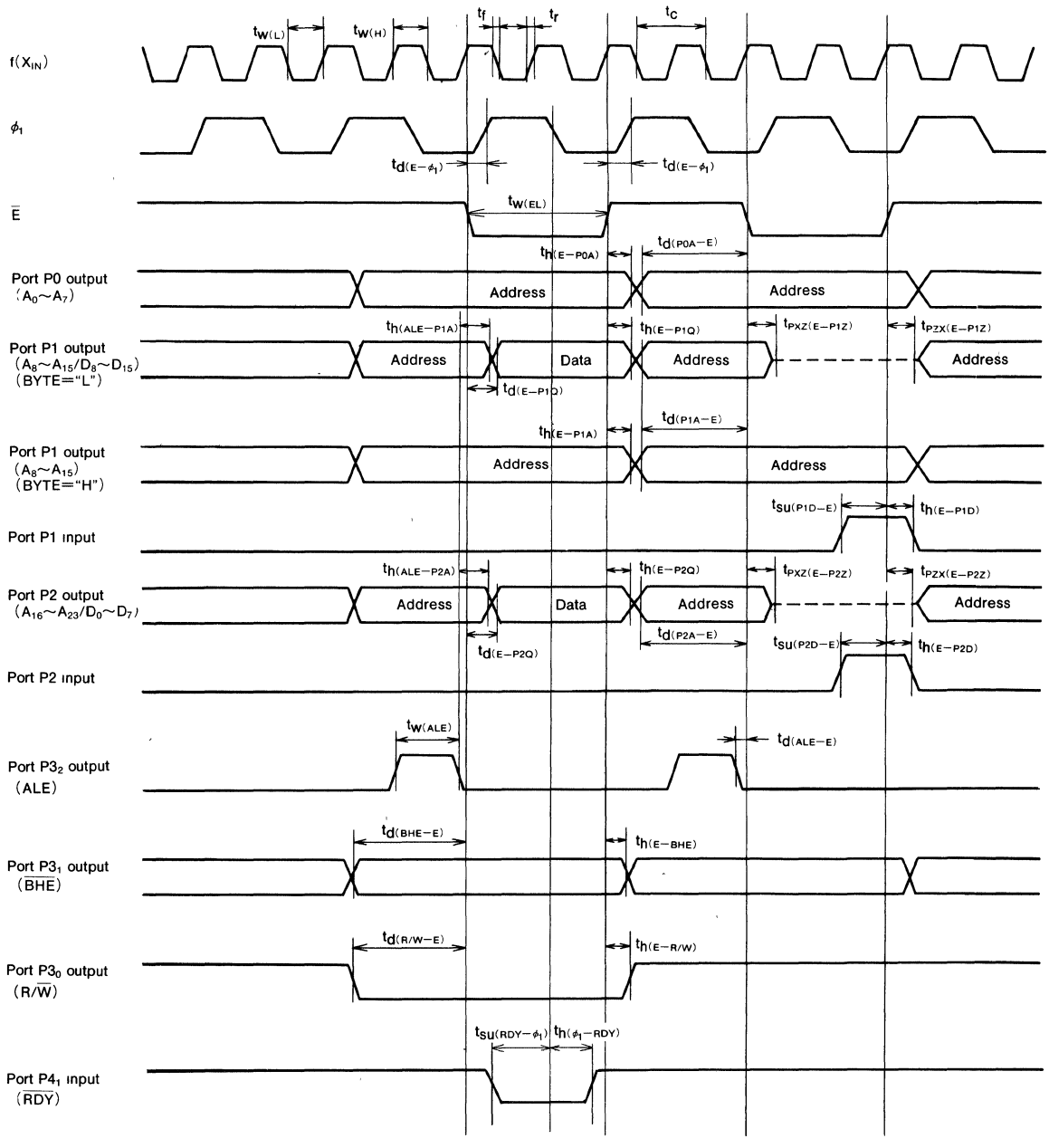
Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 1	30			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				80	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				5	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		30			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				80	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				5	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		30			ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		40			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		30			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		30			ns
$t_{d(E-\phi_1)}$	ϕ_1 output delay time		0		20	ns
$t_{h(E-P0A)}$	Port P0 address hold time		25			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		25			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		25			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		25			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		25			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		25			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		95			ns

TIMING DIAGRAM

Single-chip mode



Memory expansion mode and microprocessor mode(When wait bit = "1")



Test conditions

- V_{CC}= 5 V±10%
- Output timing voltage : V_{OL}=0.8V, V_{OH}=2.0V
- Port P1, P2 input : V_{IL}=0.8V, V_{IH}=2.5V
- Port $P4_1$ input : V_{IL}=1.0V, V_{IH}=4.0V

M37795SJ,M37795STJ

16-BIT CMOS MICROCOMPUTER

DESCRIPTION

The M37795SJ, and the M37795STJ are 16-bit microcomputers designed with high-performance CMOS silicon gate technology. These are housed in a 84-pin PLCC. These microcomputers have a large 16M bytes address space, the instruction queue buffers, and the data buffers for high-speed instruction execution. The CPU is a 16-bit parallel processor that can also be switched to perform 8-bit parallel processing. Utilizing its built-in peripheral functions such as timer system, 10-bit A-D converter and pulse width modulator (PWM), the M37795SJ and the M37795STJ are especially suited for industrial machinery applications that require real time control capability.

The M37795STJ is a version of the M37795SJ that has been upgraded for use in automobile vehicles. Its function and performance are same as for the M37795SJ, but it has a different operating temperature range which is shown below.

Therefore, the following descriptions will be for the M37795SJ unless other wise noted.

Type name	Operating temperature range
M37795SJ	−20∼75℃
M37795STJ	−40∼85℃

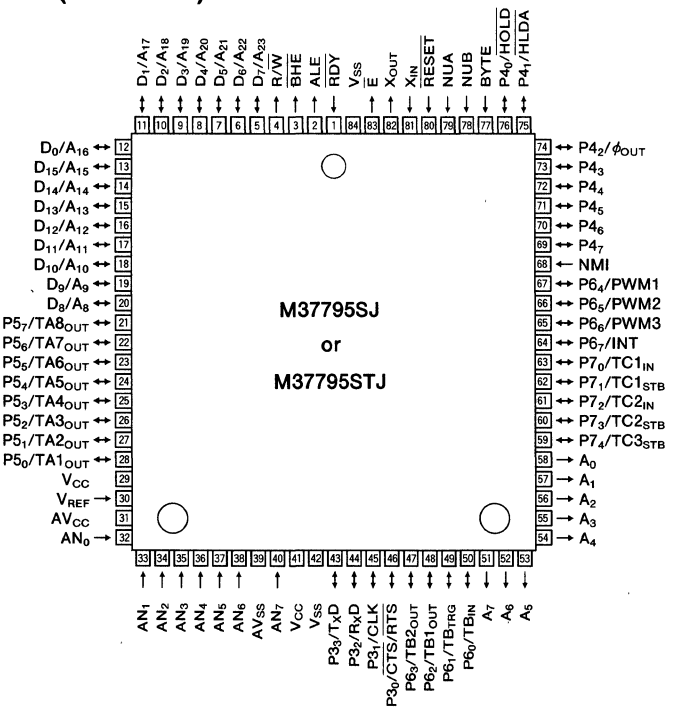
DISTINCTIVE FEATURES

- Number of basic instructions 103
- Instruction execution time
  (The fastest instruction at 8 MHz frequency) ..... 500 ns
- Single power supply ..... 5V±10%
- Low power dissipation (at 8 MHz frequency) ..... 50mW (Typ.)
- Memory ..... ROM/RAM external
- Interrupt ..... 20 types, 7 level
- 16-bit timer 16
- PWM 3
- UART (may also be synchronous) 1
- 10-bit A-D converter ..... 8 channel input
- Watchdog timer 1
- Programmable input/output
  (ports P3, P4, P5, P6, P7,) 33

APPLICATION

For factory automation system, automobiles and other industrial machinery control.

PIN CONFIGURATION (TOP VIEW)



Outline 84P0

MITSUBISHI ELECTRIC

FUNCTIONS OF M37795SJ

Parameter		Functions
Number of basic instructions		103
Instruction execution time		500ns (the fastest instructions, at 8MHz frequency)
Memory		ROM/RAM external
Input/Output port	P4~P6	8-bit×3
	P3	4-bit×1
	P7	5-bit×1
Analog input port	AN_0~AN_7	8-bit×1
Multi-function 16-bit Timer	TA1~TA8	16-bit×8
	TB1~TB4	16-bit×4
	TC1~TC3	16-bit×3
	TD	16-bit×1
PWM		8-bit×3
A-D converter		10-bit×1 (8 channels)
Serial I/O		(UART or Clock Synchronous)×1
Watchdog timer		12-bit×1
Interrupts		4 external types, 16 internal types Maskable interrupt factor is capable of setting the Interrupt Priority Level (IPL) of 0 to 7 for each factor.
Clock generating circuit		Built-in (externally connected ceramic or quarts crystal oscillator)
Supply voltage		5V±10%
Power dissipation		50mW (at external 8MHz frequency)
Operating temperature range	M37795SJ	−20~75℃
	M37795STJ	−40~85℃
Device structure		CMOS high-performance silicon gate process
Package		84-pin PLCC

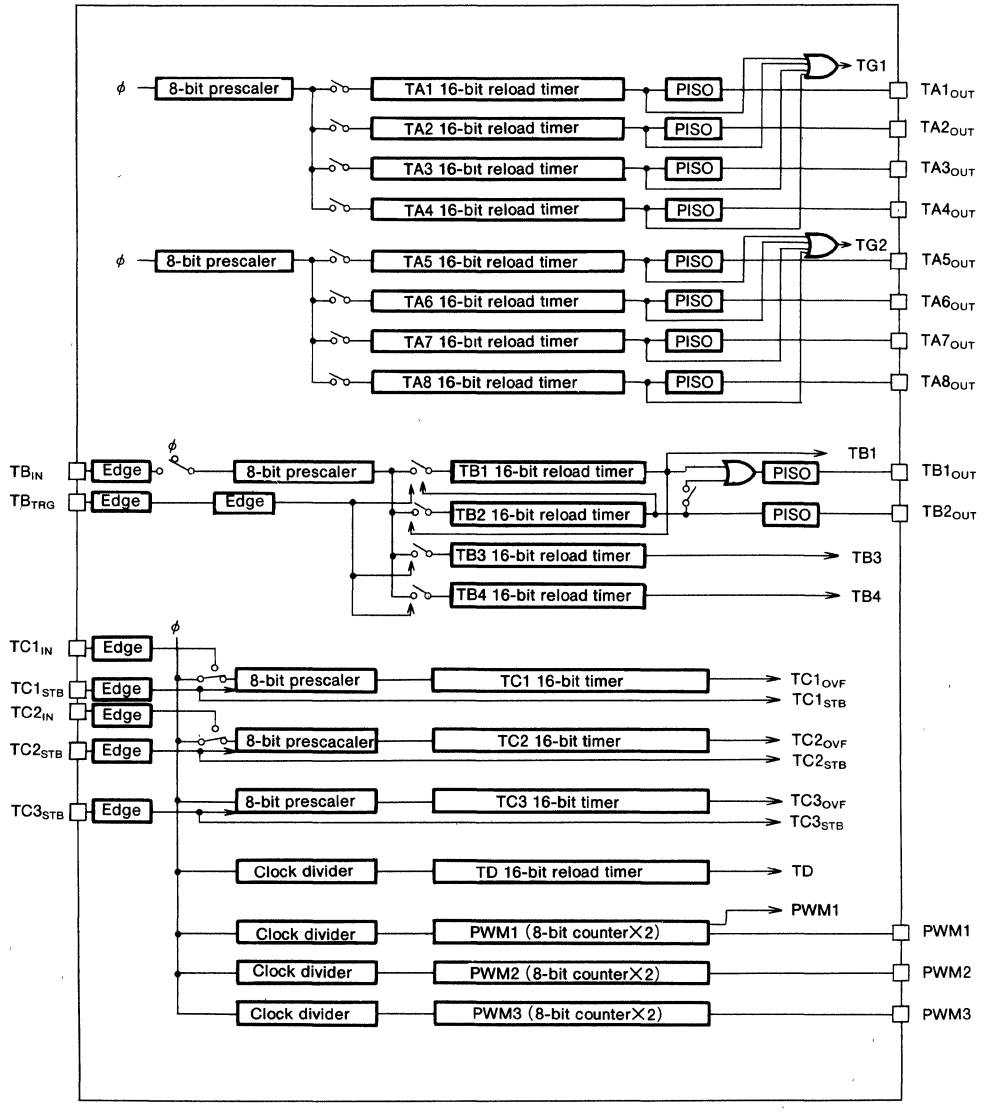
Pin	Name	Input/ Output	Functions
$P4_0$~$P4_7$	I/O port P4	I/O	This is a CMOS input/output port $P4_0$, $P4_1$ and $P4_2$ have double functions, which can be selected by software
$\overline{HOLD}$ ($P4_0$)	Hold request signal	Input	This is the hold request input to the CPU Input of "L" level sets the CPU in the hold status when the currently executing bus cycle is finished. Input of "H" level releases the hold and the CPU resumes execution
$\overline{HLDA}$ ($P4_1$)	Hold acknowledge signal	Output	When CPU is in the hold status, "L" level signal is output
ϕ_{OUT}($P4_2$)	System clock output	Output	This is the external output of CPU system clock (ϕ)
$P5_0$~$P5_7$	I/O port P5	I/O	This is a CMOS input/output port All 8 bits have double functions, which can be selected by softwave
$TA1_{OUT}$ ~$TA8_{OUT}$ ($P5_0$~$P5_7$)	PISO data output	Output	PISO data update by synchronizing with the timer underflow, etc of the timers, TA1~TA8
$P6_0$~$P6_7$	I/O port P6	I/O	This is a CMOS input/output port All 8 bits have double functions, which can be selected by softwere.
TB_{IN}($P6_0$)	Timer B clock	Input	This signal is used as the clock input for the timers, TB1~TB4
TB_{TRG}($P6_1$)	Timer B trigger	Input	This is the trigger signal input for the timers, TB1, TB3 and TB4
$TB1_{OUT}$($P6_2$) $TB2_{OUT}$($P6_3$)	PISO data output	Output	PISO data update by synchronizing with the timer underflow, etc of the timers, TB1 and TB2
PWM1 ~PWM3 ($P6_4$~$P6_6$)	PWM output	Output	Output pulse of this pin is generated by PWM
INT($P6_7$)	External interrupt input	Input	This is the interrupt input pin Edge sense or level sense can be specified
$P7_0$~$P7_4$	I/O port P7	I/O	This is a CMOS input/output port All 5 bits have double functions, which can be selected by software.
$TC1_{IN}$($P7_0$)	Timer C1 clock	Input	This is the timer TC1's clock input
$TC1_{STB}$($P7_1$)	Timer C1 strobe	Input	This is the strobe signal input for timer TC1's measurement function
$TC2_{IN}$($P7_2$)	Timer C2 clock	Input	This is the timer TC2's clock input.
$TC2_{STB}$($P7_3$)	Timer C2 strobe	Input	This is the strobe signal input for timer TC2's meapurement function.
$TC3_{STB}$($P7_4$)	Timer C3 strobe	Input	This is the strobe signal input for timer TC3's measurement function
AN_0~AN_7	Analog input port	Input	This is an 8-bit analog input pin to the A-D converter

Address (Hexadecimal notaion)	
000000	
000001	
000002	
000003	
000004	
000005	
000006	
000007	
000008	
000009	
00000A	
00000B	Port P3 data register
00000C	Port P4 data register
00000D	Port P5 data register
00000E	Port P6 data register
00000F	Port P7 data register
000010	
000011	
000012	
000013	Port P3 direction register
000014	Port P4 direction register
000015	Port P5 direction register
000016	Port P6 direction register
000017	Port P7 direction register
000018	
000019	
00001A	
00001B	Port P3 operation mode register
00001C	Port P4 operation mode register
00001D	Port P5 operation mode register
00001E	Port P6 operation mode register
00001F	Port P7 operation mode register
000020	A-D control register
000021	
000022	A-D succesive approximation register
000023	
000024	
000025	
000026	
000027	
000028	
000029	
00002A	
00002B	
00002C	
00002D	
00002E	
00002F	
000030	UART transmit/receive mode register
000031	UART baud rate register
000032	UART transmission buffer register
000033	
000034	UART transmission/receive control register 0
000035	UART transmission/receive control register 1
000036	UART receive buffer register
000037	
000038	
000039	
00003A	
00003B	
00003C	
00003D	
00003E	
00003F	

Address (Hexadecimal notation)	
000040	Timer A1 counter
000041	
000042	Timer A1 reload register
000043	
000044	Timer A2 counter
000045	
000046	Timer A2 reload register
000047	
000048	Timer A3 counter
000049	
00004A	Timer A3 reload register
00004B	
00004C	Timer A4 counter
00004D	
00004E	Timer A4 reload register
00004F	
000050	Timer A5 counter
000051	
000052	Timer A5 reload register
000053	
000054	Timer A6 counter
000055	
000056	Timer A6 reload register
000057	
000058	Timer A7 counter
000059	
00005A	Timer A7 reload register
00005B	
00005C	Timer A8 counter
00005D	
00005E	Timer A8 reload register
00005F	
000060	TG1 prescaler
000061	TG2 prescaler
000062	Timer A control register-enable
000063	Timer A protect register
000064	Timer A control register-CW
000065	Timer A control register-P/N
000066	Timer A interrupt mask register
000067	Timer A interrupt status register
000068	Timer A1 PISO register
000069	Timer A2 PISO register
00006A	Timer A3 PISO register
00006B	Timer A4 PISO register
00006C	Timer A5 PISO register
00006D	Timer A6 PISO register
00006E	Timer A7 PISO register
00006F	Timer A8 PISO register
000070	Timer D counter
000071	
000072	Timer D reload register
000073	
000074	Timer D operation control register
000075	
000076	
000077	
000078	
000079	
00007A	
00007B	
00007C	
00007D	
00007E	
00007F	

Fig. 2 Location of peripheral devices control registers (1)

M37795SJ TIMER SYSTEM BLOCK DIAGRAM



MITSUBISHI ELECTRIC

M37795SJ

ELECTRICAL CHARACTERISTICS (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
V_{OH}	High-level output voltage	I_{OH}=−2mA	V_{CC}−1			V
V_{OL}	Low-level output voltage	I_{OL}=2mA			0.45	V
I_{IH}	High-level input current	V_I=V_{CC}	−5		5	μA
I_{IL}	Low-level input current	V_I=0V	−5		5	μA
I_{CC}	Power supply current (Note 1)	f(X_{IN})=8MHz square waveform		8(Note 2)	20	mA
		T_a=25℃ when clock is stopped			1	μA
		T_a=75℃ when clock is stopped			15	
$V_{T+}-V_{T-}$	Hysteresis TB_{IN}, TB_{TRG}, $TC1_{IN}$, $TC2_{IN}$, $TC1_{STB}$, $TC2_{STB}$, $TC3_{STB}$, $\overline{RESET}$ (Note3)	V_{CC}=5V	0.8			V
$V_{T+}-V_{T-}$	Hysteresis $\overline{HOLD}$, INT, CLK, CTS/$\overline{RTS}$, NMI, $\overline{RDY}$ (Note 4)	V_{CC}=5V	0.2			V

Note 1. When reset with output pins open and connecting input pins to V_{SS}.
2. V_{CC}=5V, T_a=25℃
3. Double function except for $\overline{RESET}$ pin
4. Double function except for NMI and $\overline{RDY}$ pins.

A-D CONVERTER CHARACTERISTICS (AV_{CC}=V_{REF}=5.12V, T_a=25℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
—	Resolution			10	Bits
—	Absolute accuracy			±3	LSB
t_{CONV}	Conversion time	40.5			μs
I_{IAN}	Analog input leak current (Note 1)	−200		200	nA

Note 1. Input leak current of AN_0~AN_7 with A-D converter halted
Input voltage is 0≦V_I≦AV_{CC}.

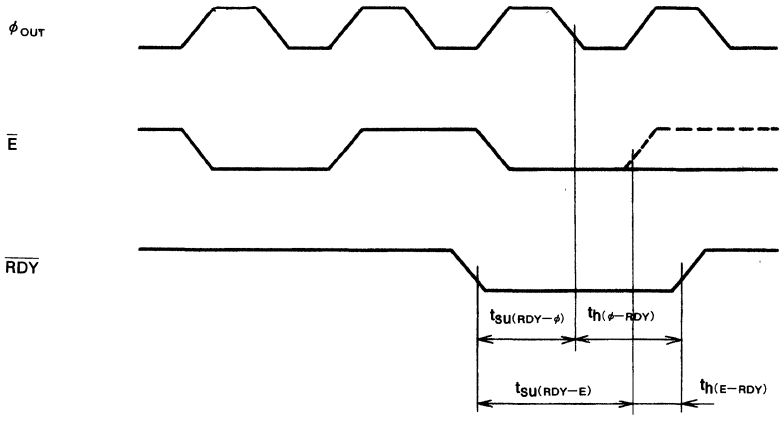
TIMING REQUIREMENTS (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max	
t_C	Cycle time	125			ns
$t_{SU(D-E)}$	Data input setup time	80			ns
$t_{h(E-D)}$	Data input hold time	0			ns

$\overline{RDY}$ characteristics (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

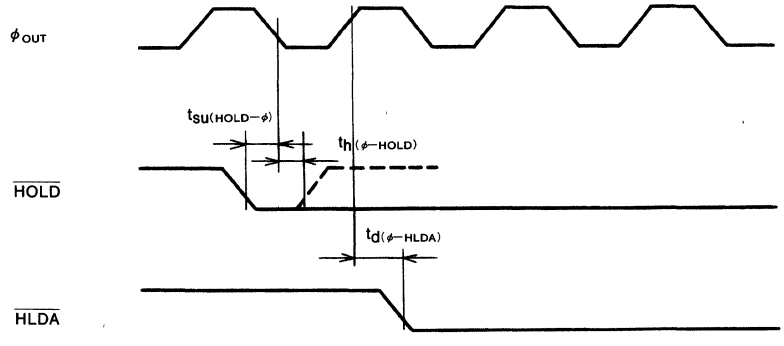
Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max	
$t_{su(RDY-\phi)}$	$\overline{RDY}$ input setup time	100		220	ns
$t_{h(\phi-RDY)}$	$\overline{RDY}$ input hold time	60			ns
$t_{su(RDY-E)}$	$\overline{RDY}$ input setup time	220		300	ns
$t_{h(E-RDY)}$	$\overline{RDY}$ input hold time	−20			ns

Note. Follow the characteristics of ϕ or $\overline{E}$.



$\overline{HOLD}$ characteristics (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

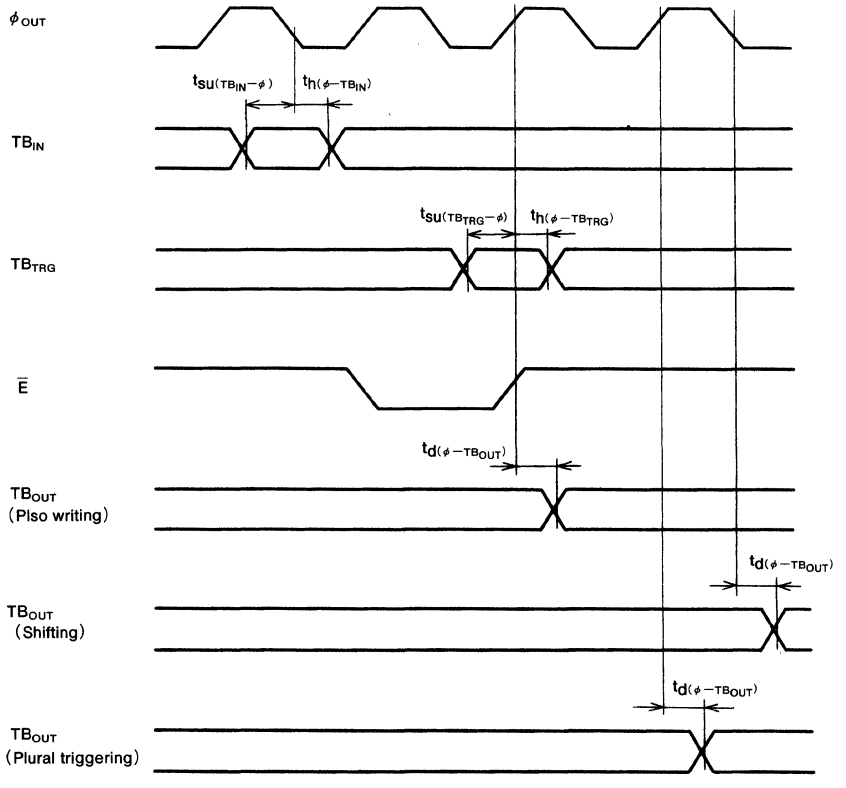
Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{su(HOLD-\phi)}$	$\overline{HOLD}$ input setup time	100			ns
$t_{h(\phi-HOLD)}$	$\overline{HOLD}$ input hold time	30			ns
$t_{d(\phi-HLDA)}$	$\overline{HLDA}$ output delay time			80	ns



Timer B characteristics (V_{CC}=5V±10%, T_a=−20~75℃, $f(X_{IN})$=8MHz)

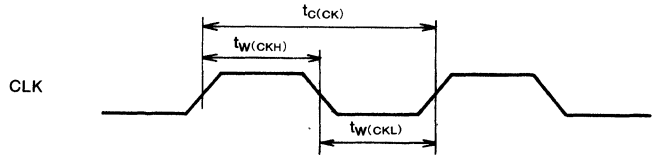
Symbol	Parameter		Limits			Unit
			Min	Typ.	Max.	
$t_{su(TB_{IN}-\phi)}$	TB_{IN} input setup time		200			ns
$t_{h(\phi-TB_{IN})}$	TB_{IN} input hold time		0			ns
$t_{su(TB_{STB}-\phi)}$	TB_{STB} input setup time		200			ns
$t_{h(\phi-TB_{STB})}$	TB_{STB} input hold time		0			ns
$t_{d(\phi-TB_{OUT})}$	TB_{OUT} output delay time	PISO writing			200	ns
		Shifting			200	
		Plural triggering (Note 1)			200	

Note 1. Shift by TB2 underflow with plural trigger.



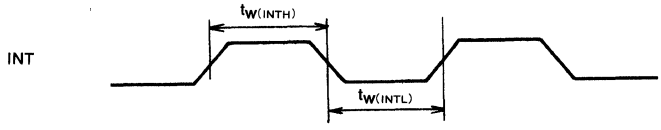
UART characteristics (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ	Max.	
$t_{C(CK)}$	CLK input cycle time	500			ns
$t_{W(CKH)}$	CLK input high-level pulse width	250			ns
$t_{W(CKL)}$	CLK input low-level pulse width	250			ns



INT characteristics (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{W(INTH)}$	INT input high-level pulse width	250			ns
$t_{W(INTL)}$	INT input low-level pulse width	250			ns



SWITCHING CHARACTERISTICS (V_{CC}=5V±10%, T_a=−40~85℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{W(ALE)}$	ALE pulse width	80			ns
$t_{C(ALE-E)}$	$\overline{E}$ end time after ALE	300			ns
$t_{d(ALE-E)}$	$\overline{E}$ start time after ALE	0			ns
$t_{W(E)}$	$\overline{E}$ pulse width	180			ns
$t_{pzv(E-DZ)}$	Floating release delay time	60			ns
$t_{d(A-E)}$	Address output delay time	100			ns
$t_{pvz(E-DZ)}$	Floating start delay time			40	ns
$t_{d(E-D)}$	Data output delay time			100	ns
$t_{v(ALE-A)}$	Address delay time after ALE	10			ns
$t_{v(E-D)}$	Data valid time after $\overline{E}$	10			ns
$t_{v(E-A)}$	Address valid time after $\overline{E}$	10			ns
$t_{d(CONT-E)}$	Control signal delay time	100			ns
$t_{v(E-CONT)}$	Control signal valid time	10			ns

Note. Limits have guaranties under load capacity of the test pin=100pF including test tool's capacity. If capacities of pins are much different from each other, limits may not be followed. So pay attention to the design of the board.

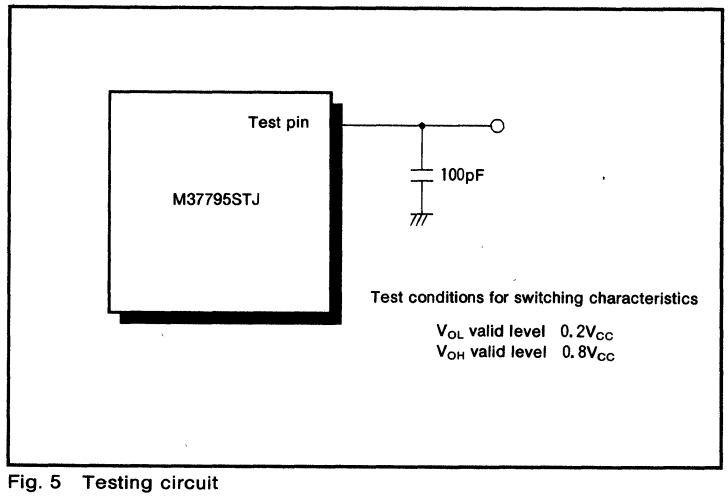
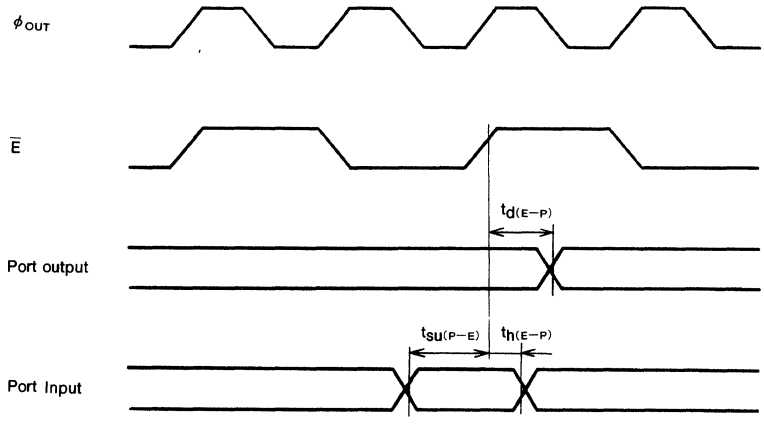


Fig. 5 Testing circuit

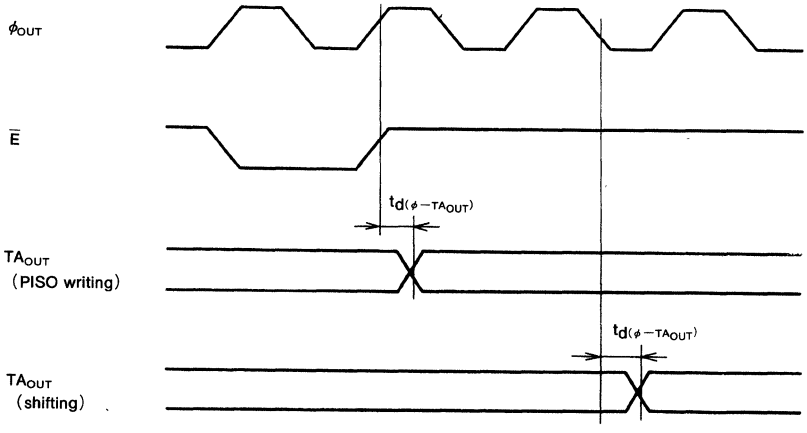
Port characteristics (V_{CC}=5V±10%, T_a=−40~85℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min	Typ.	Max.	
$t_{su(P-E)}$	Port input setup time	200			ns
$t_{h(E-P)}$	Port input hold time	20			ns
$t_{d(E-P)}$	Port data output delay time			200	ns



Timer A characteristics (V_{CC}=5V±10%, T_a=−40~85℃, f(X_{IN})=8MHz)

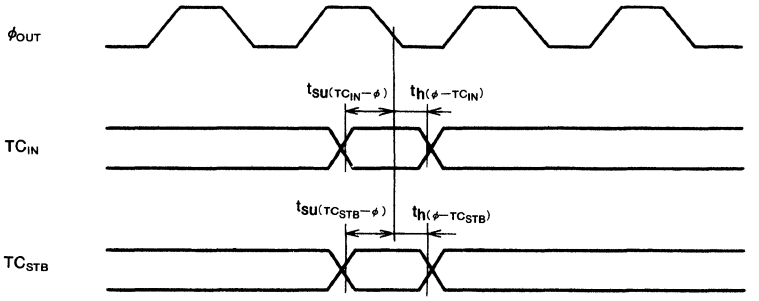
Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max.	
$t_{d(\phi-TA_{OUT})}$	TA_{OUT} output delay time	PISO writing			200	ns
		Shifting			200	ns



MITSUBISHI ELECTRIC

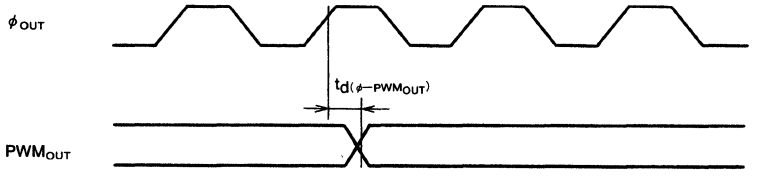
Timer C characteristics (V_{CC}=5V±10%, T_a=−40~85℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{su(TC_{IN}-\phi)}$	TC_{IN} input setup time	200			ns
$t_{h(\phi-TC_{IN})}$	TC_{IN} input hold time	0			ns
$t_{su(TC_{STB}-\phi)}$	TC_{STB} input setup time	200			ns
$t_{h(\phi-TC_{STB})}$	TC_{STB} input hold time	0			ns

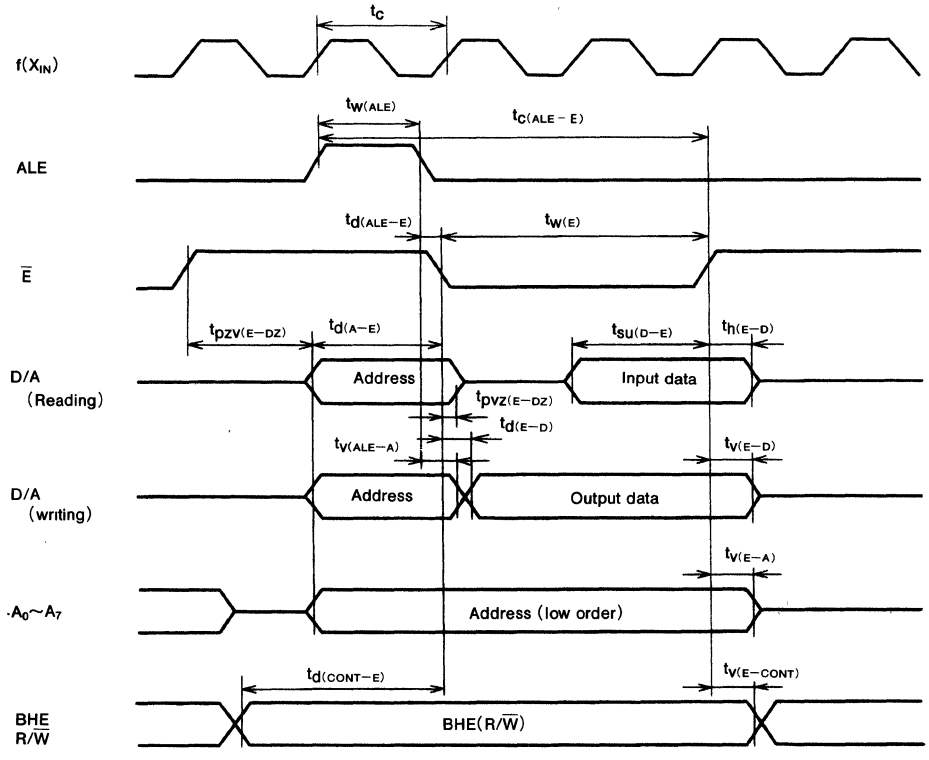


PWM characteristics (V_{CC}=5V±10%, T_a=−40~85℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{d(\phi-PWM_{OUT})}$	PWM_{OUT} output delay time			200	ns



TIMING DIAGRAM



MITSUBISHI ELECTRIC

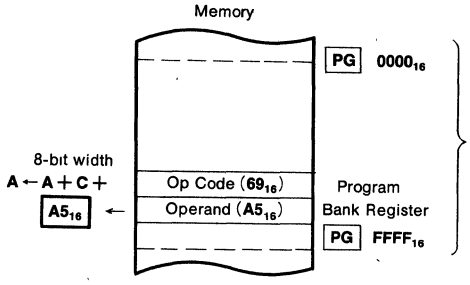
SERIES MELPS 7700 ADDRESSING MODES

Mode : Immediate addressing mode

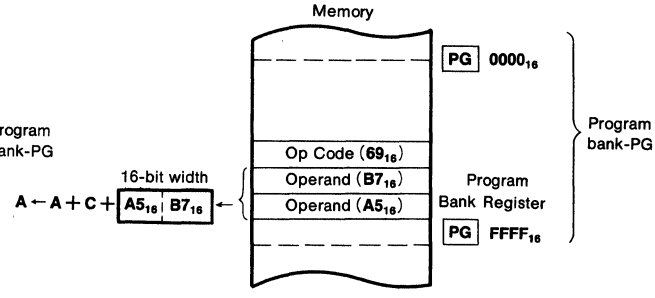
Function : A portion of the instruction is the actual data. Such instruction code may cross over the bank boundary.

Instruction : **ADC, AND, CLP, CMP, CPX, CPY, DIV, EOR, LDA, LDT, LDX, LDY, MPY, ORA, RLA, SBC, SEP**

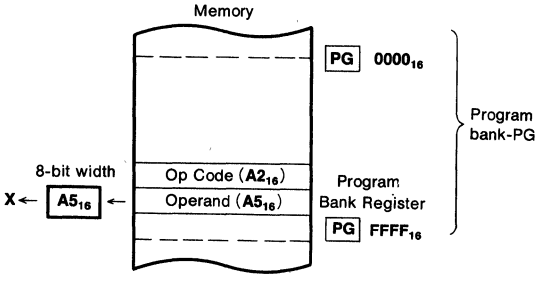
ex. : Mnemonic **ADC A,#0A5H** (**m = 1**) Machine code 69_{16} $A5_{16}$



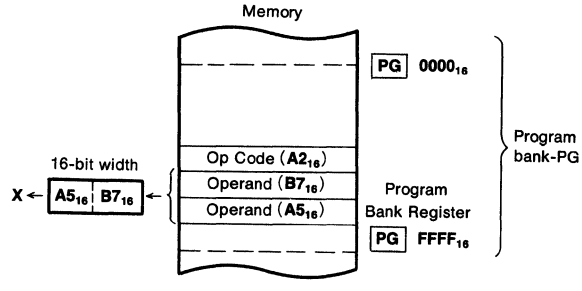
ex. : Mnemonic **ADC A,#0A5B7H** (**m = 0**) Machine code 69_{16} $B7_{16}$ $A5_{16}$



ex. : Mnemonic **LDX #0A5H** (**x = 1**) Machine code $A2_{16}$ $A5_{16}$



ex. : Mnemonic **LDX #0A5B7H** (**x = 0**) Machine code $A2_{16}$ $B7_{16}$ $A5_{16}$

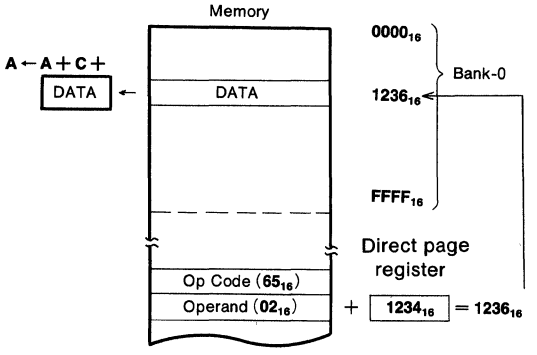


Mode : Direct addressing mode

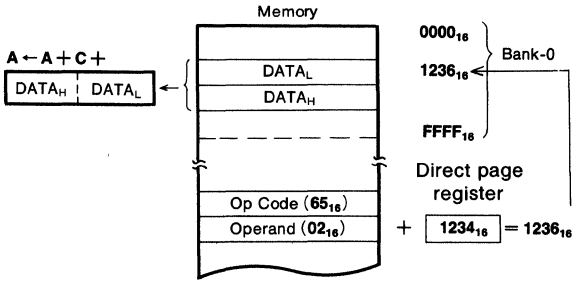
Function : The contents of the bank-0 memory location specified by the result of adding the second byte of the instruction to the contents of the direct page register become the actual data. If, however, addition of the instruction's second byte to the direct page register's contents result in a value that exceeds the bank-0 range, the specified location will be in bank-1

Instruction : **ADC, AND, ASL, CMP, CPX, CPY, DEC, DIV, EOR, INC, LDA, LDM, LDX, LDY, LSR, MPY, ORA, ROL, ROR, SBC, STA, STX, STY**

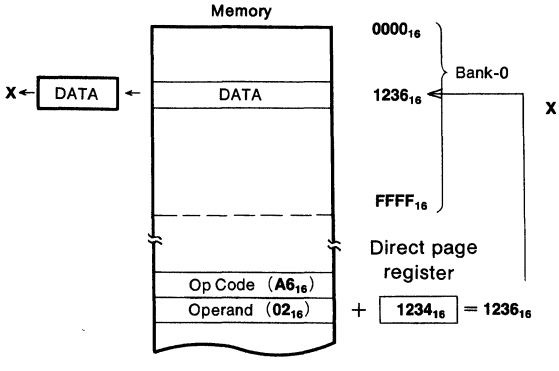
ex. : Mnemonic **ADC A,02H** (**m = 1**) Machine code 65_{16} 02_{16}



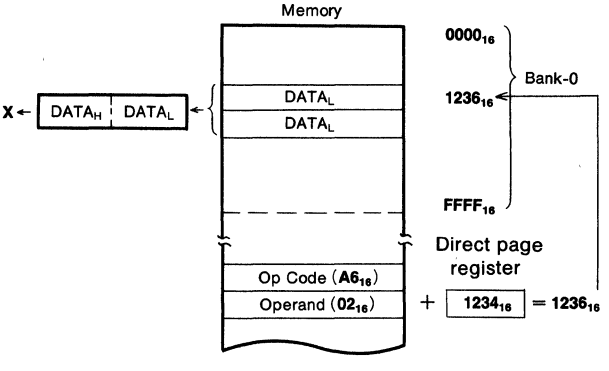
ex. : Mnemonic **ADC A,02H** (**m = 0**) Machine code 65_{16} 02_{16}



ex. : Mnemonic **LDX 02H** (**x = 1**) Machine code $A6_{16}$ 02_{16}



ex. : Mnemonic **LDX 02H** (**x = 0**) Machine code $A6_{16}$ 02_{16}

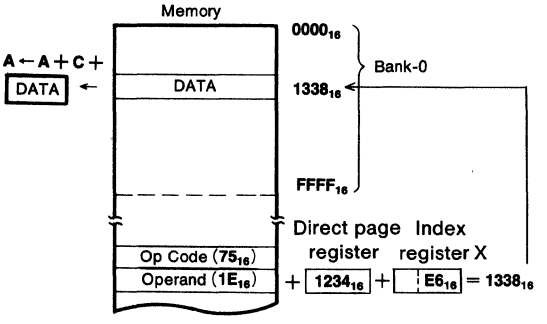


Mode : Direct indexed X addressing mode

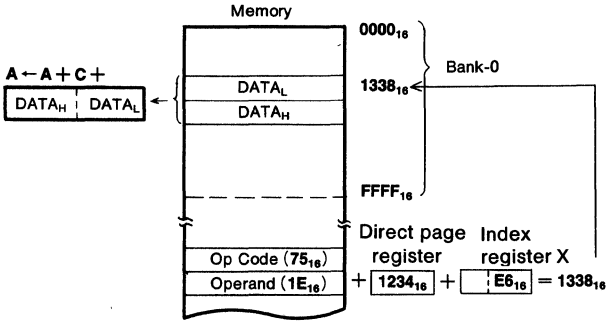
Function : The contents of the bank-0 memory location specified by the result of adding the second byte of the instruction, the contents of the direct page register and the contents of the index register X become the actual data. If, however, addition of the instruction's second byte, the direct page register's contents and the index register X's contents results in a value that exceeds the bank-0 or bank-1 range, the specified location will be in bank-1 or bank-2.

Instruction : **ADC, AND, ASL, CMP, DEC, DIV, EOR, INC, LDA, LDM, LDY, LSR, MPY, ORA, ROL, ROR, SBC, STA, STY**

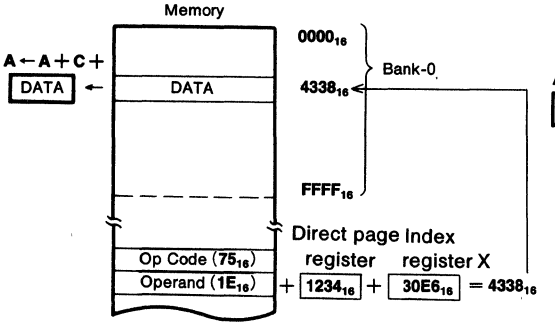
ex. : Mnemonic **ADC A,1EH,X** (**m = 1, x = 1**) Machine code 75_{16} $1E_{16}$



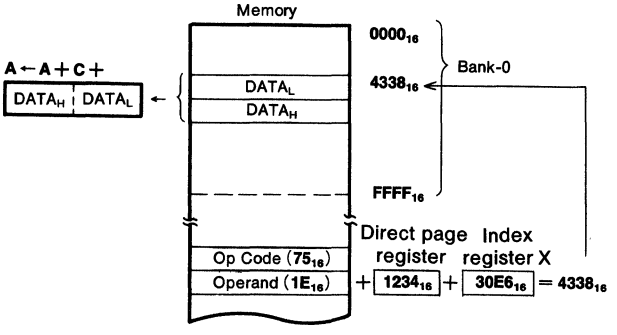
ex. : Mnemonic **ADC A,1EH,X** (**m = 0, x = 1**) Machine code 75_{16} $1E_{16}$



ex. : Mnemonic **ADC A,1EH,X** (**m = 1, x = 0**) Machine code 75_{16} $1E_{16}$



ex. : Mnemonic **ADC A,1EH,X** (**m = 0, x = 0**) Machine code 75_{16} $1E_{16}$



MITSUBISHI ELECTRIC

SERIES MELPS 7700 ADDRESSING MODES

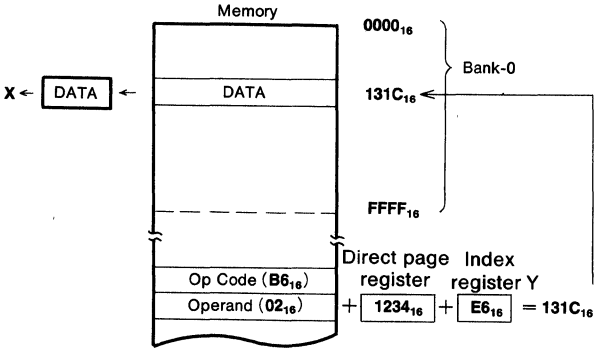
Mode : Direct indexed Y addressing mode

Function : The contents of the bank-0 memory location specified by the result of adding the second byte of the instruction, the contents of the direct page register and the contents of the index register Y become the actual data. If, however, addition of the instruction's second byte, the direct page register's contents and the index register Y's contents results in a value that exceeds the bank-0 or bank-1 range, the specified location will be in bank-1 or bank-2.

Instruction : **LDX, STX**

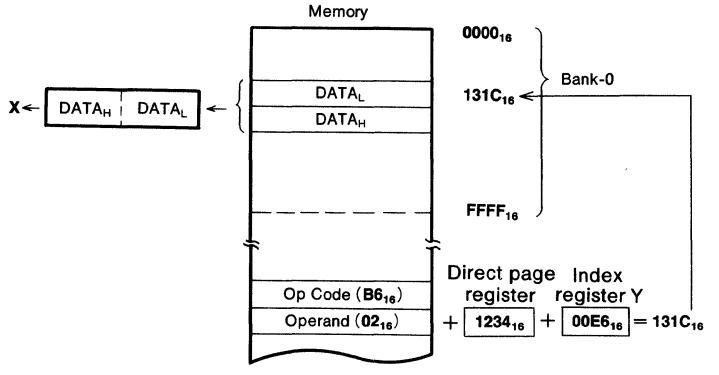
ex. : Mnemonic **LDX 02H,Y** (**x = 1**)

Machine code **$B6_{16}$ 02_{16}**



ex. : Mnemonic **LDX 02H,Y** (**x = 0**)

Machine code **$B6_{16}$ 02_{16}**



SERIES MELPS 7700 ADDRESSING MODES

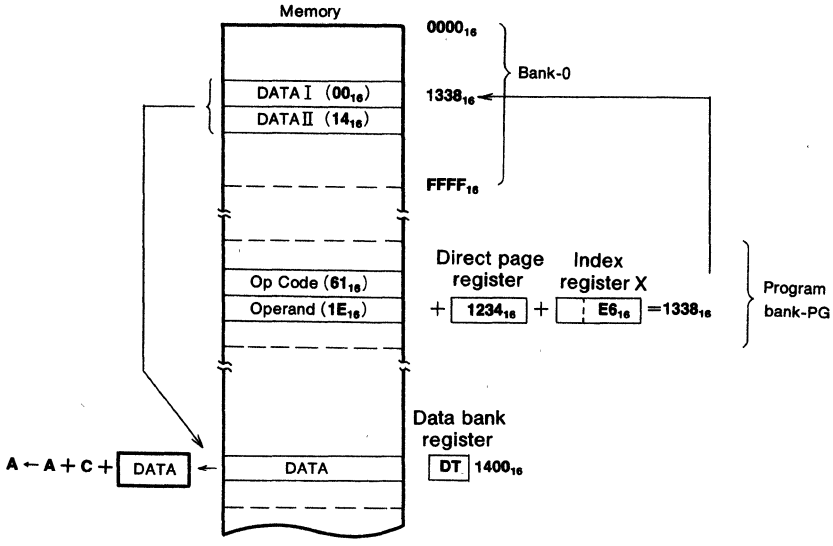
Mode : **Direct indexed X indirect addressing mode**

Function : The value obtained by adding the instruction's second byte, the contents of the direct page register and the contents of the index register X specifies 2 adjacent bytes in memory bank-0, and the contents of these bytes in memory bank-0, and the contents of these bytes in memory bank-DT (DT is contents of data bank register) become the actual data. If, however, the value obtained by adding the instruction's second byte, the direct page register's contents and the index register X's contents exceeds the bank-0 or bank-1 range, the specified location will be in bank-1 or bank-2

Instruction : **ADC, AND, CMP, DIV, EOR, LDA, MPY, ORA, SBC, STA**

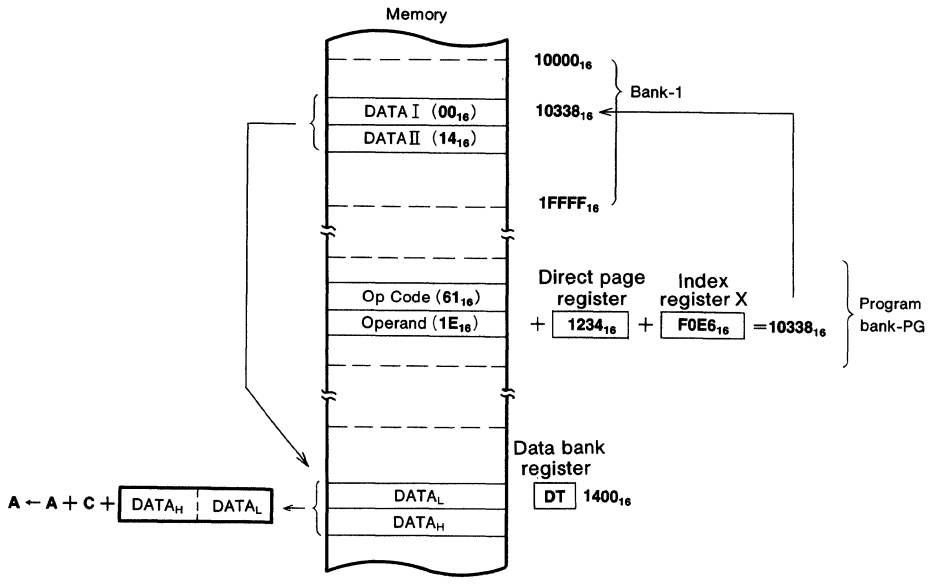
ex. : Mnemonic — **ADC A, (1EH, X)** **(m = 1, x = 1)**

Machine code — 61_{16} $1E_{16}$



ex. : Mnemonic
ADC A, (1EH, X)
(m = 0, x = 0)

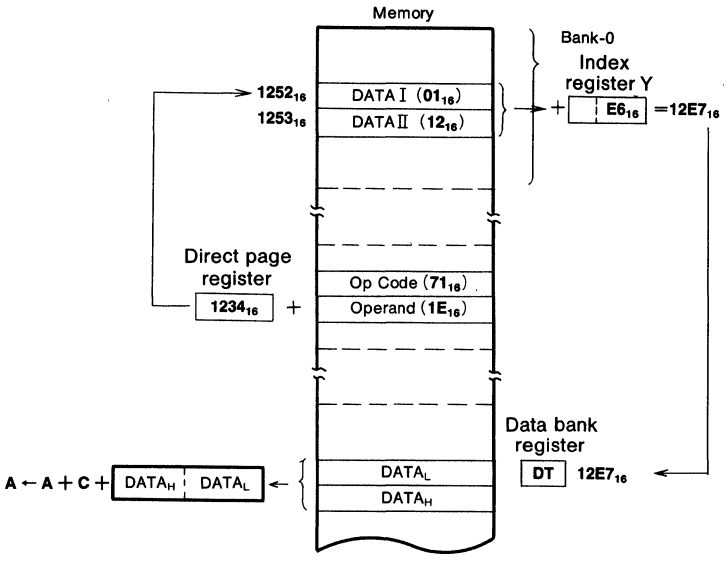
Machine code
61_{16} $1E_{16}$



MITSUBISHI ELECTRIC

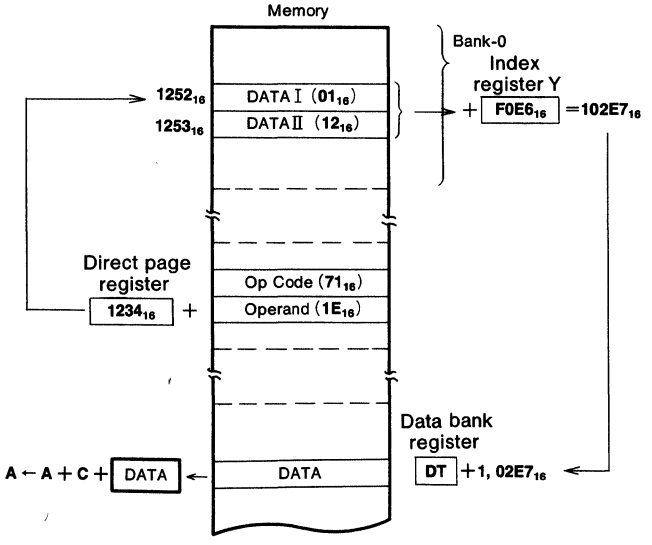
ex. : Mnemonic
ADC A, (1EH), Y
(m = 0, x = 1)

Machine code
71_{16} $1E_{16}$



ex. : Mnemonic
ADC A, (1EH), Y
(m = 1, x = 0)

Machine code
71_{16} $1E_{16}$



MITSUBISHI ELECTRIC

SERIES MELPS 7700 ADDRESSING MODES

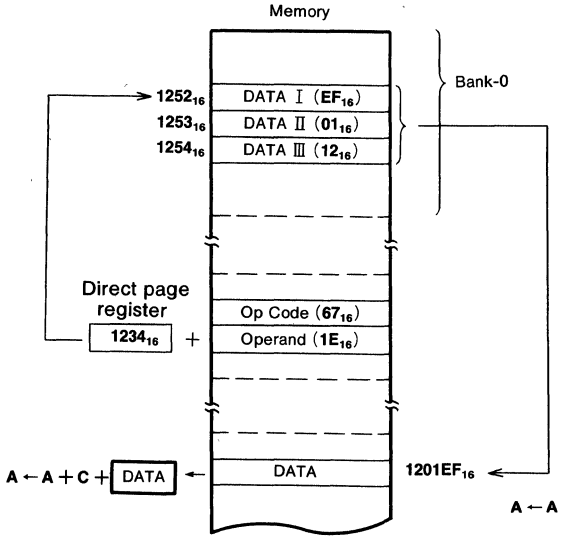
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER

Mode : Direct indirect long addressing mode

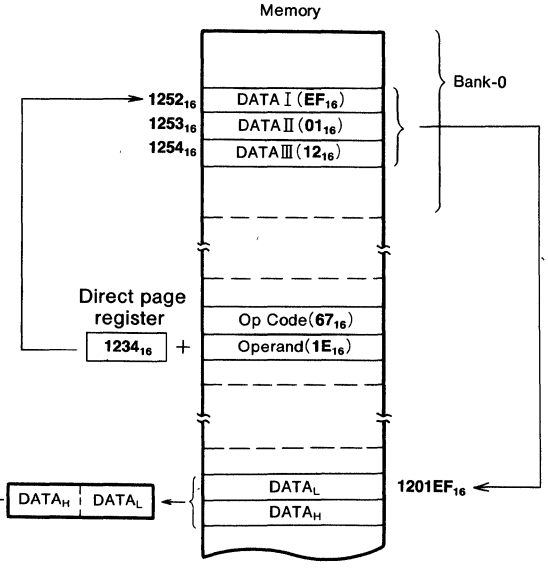
Function : The value obtained by adding the instruction's second byte and the contents of the direct page register specifies 3 adjacent bytes in memory bank-0, and the contents of these bytes specify the address of the memory location that contains the actual data. If, however, the value obtained by adding the contents of the instruction's second byte and the direct page register exceeds the bank-0 range, the specified location will be in bank-1. The 3 adjacent bytes memory location may be spread over two different banks.

Instruction : **ADC, AND, CMP, DIV, EOR, LDA, MPY, ORA, SBC, STA**

ex. : Mnemonic **ADCL A, (1EH) (m=1)** Machine code 67_{16} $1E_{16}$

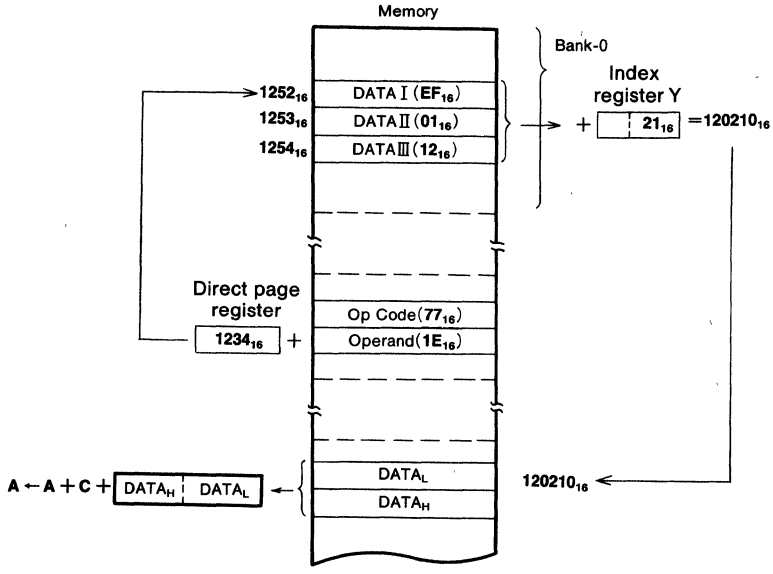


ex. : Mnemonic **ADCL A, (1EH) (m=0)** Machine code 67_{16} $1E_{16}$



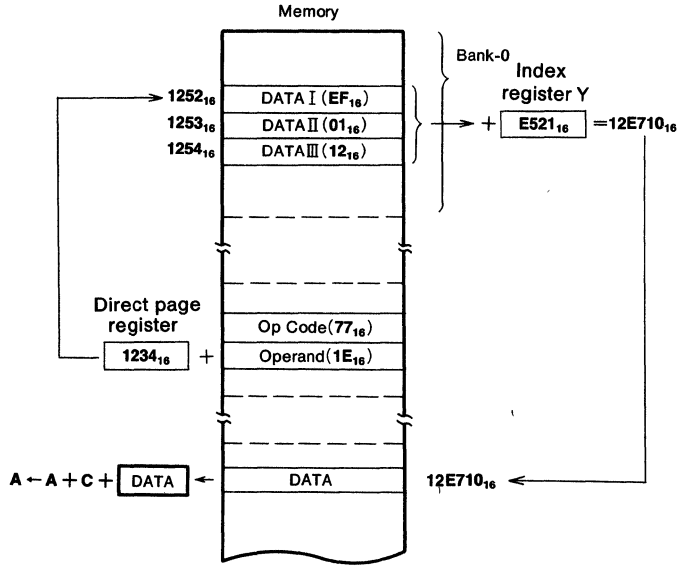
ex. : Mnemonic
ADCL A,(1EH), Y
(m=0, x=1)

Machine code
77_{16} $1E_{16}$



ex. : Mnemonic
ADCL A,(1EH), Y
(m=1, x=0)

Machine code
77_{16} $1E_{16}$

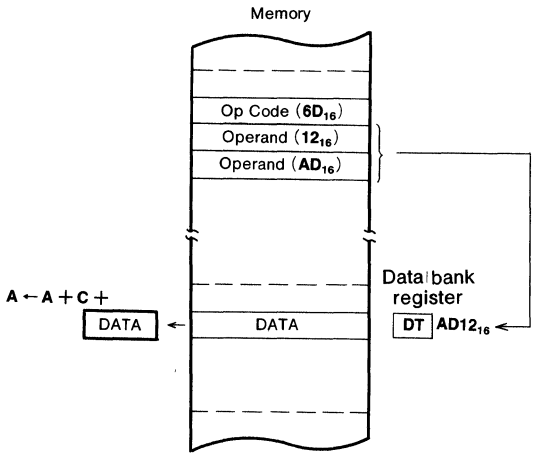


Mode : Absolute addressing mode

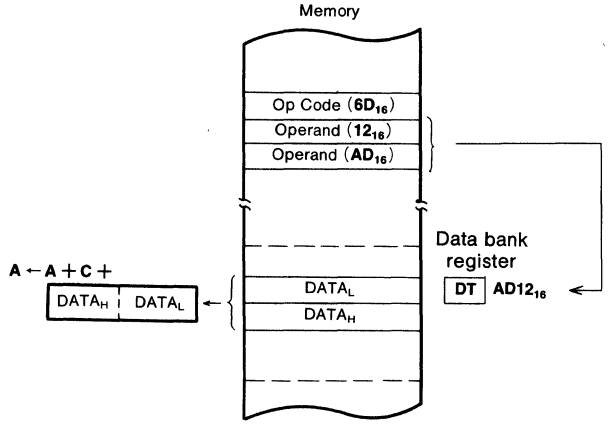
Function : The contents of the memory locations specified by the instruction's second and third bytes and the contents of the data bank register are the actual data. Note that, in the cases of the JMP and JSR instructions, the instructions' second and third byte contents are transferred to the program counter.

Instruction : **ADC, AND, ASL, CMP, CPX, CPY, DEC, DIV, EOR, INC, JMP, JSR, LDA, LDM, LDX, LDY, LSR, MPY, ORA, ROL, ROR, SBC, STA, STX, STY**

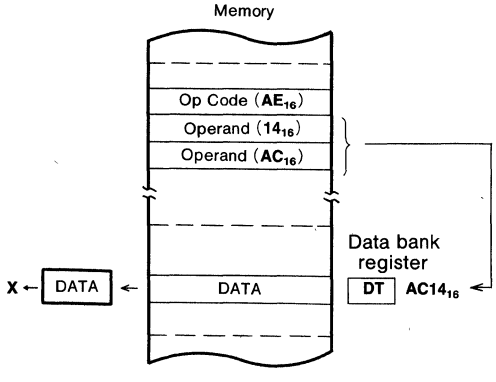
ex. : Mnemonic **ADC A, 0AD12H** (**m=1**) Machine code $6D_{16}$ 12_{16} AD_{16}



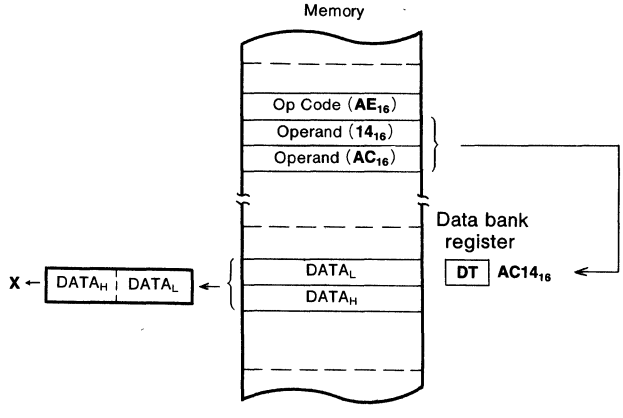
ex. : Mnemonic **ADC A, 0AD12H** (**m=0**) Machine code $6D_{16}$ 12_{16} AD_{16}



ex. : Mnemonic **LDX 0AC14H** (**x=1**) Machine code AE_{16} 14_{16} AC_{16}



ex. : Mnemonic **LDX 0AC14H** (**x=0**) Machine code AE_{16} 14_{16} AC_{16}



MITSUBISHI ELECTRIC

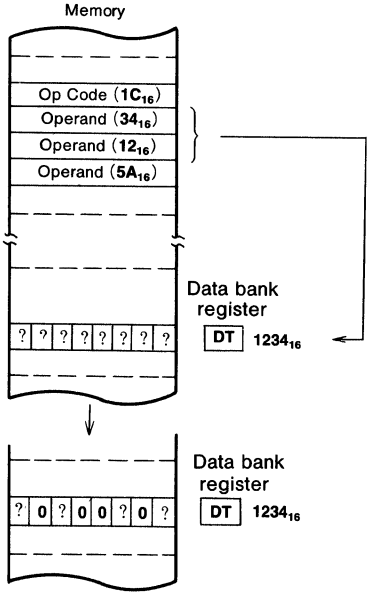
Mode : Absolute bit addressing mode

Function : The contents of the instruction's second and third bytes and the contents of the data bank register specify the memory locations, and data for multiple bit positions in the memory locations are specified by a bit pattern specified in the instruction's fourth and fifth bytes (the fourth byte only if the m flag is set to 1).

Instruction : **CLB, SEB**

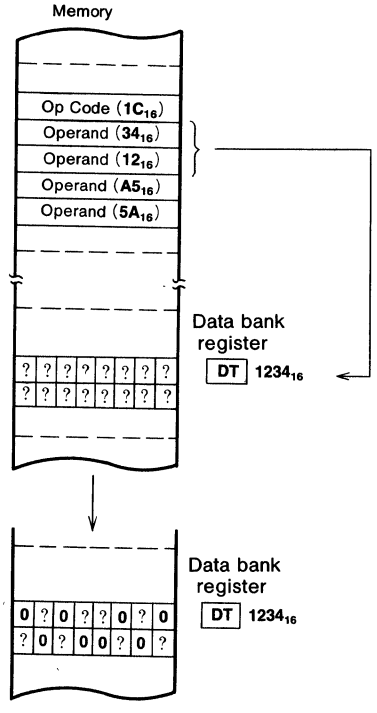
ex. : Mnemonic
CLB #5AH, 1234H
(m=1)

Machine code
$1C_{16}$ 34_{16} 12_{16} $5A_{16}$

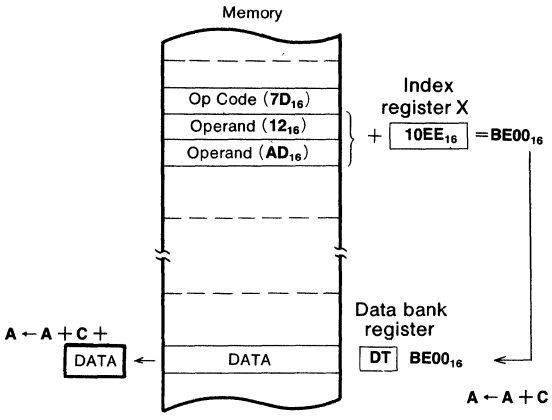


ex. : Mnemonic
CLB #5AA5H, 1234H
(m=0)

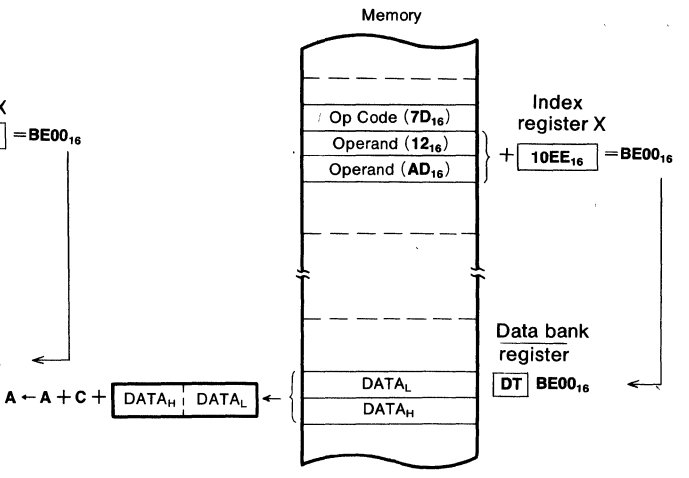
Machine code
$1C_{16}$ 34_{16} 12_{16} $A5_{16}$ $5A_{16}$



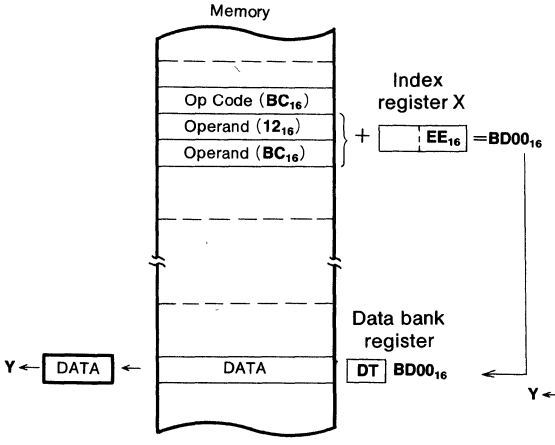
ex. : Mnemonic **ADC A, 0AD12H, X** **(m=1, x=0)** Machine code $7D_{16}\ 12_{16}\ AD_{16}$



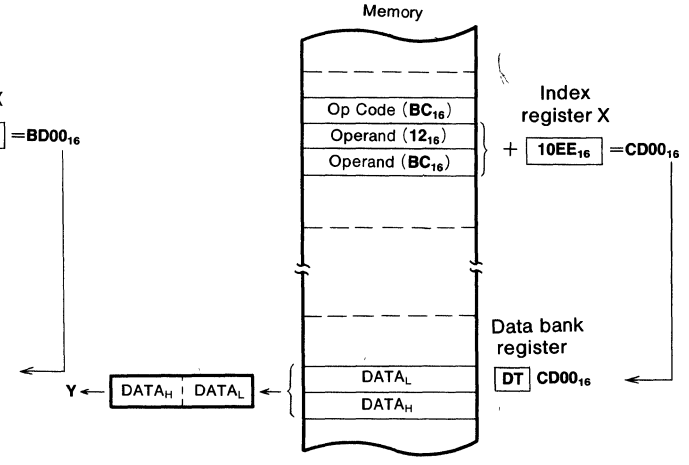
ex. : Mnemonic **ADC A, 0AD12H, X** **(m=0, x=0)** Machine code $7D_{16}\ 12_{16}\ AD_{16}$



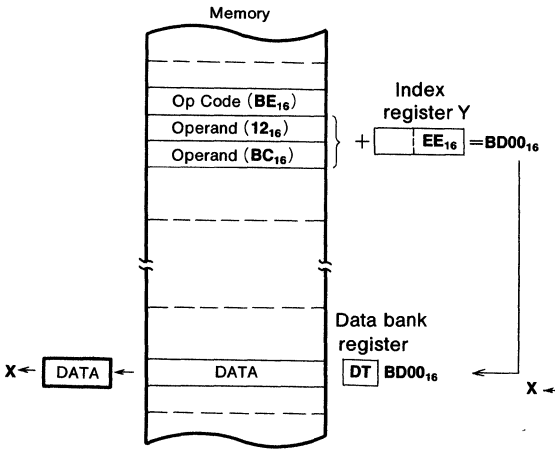
ex. : Mnemonic **LDY 0BC12H, X** **(x=1)** Machine code $BC_{16}\ 12_{16}\ BC_{16}$



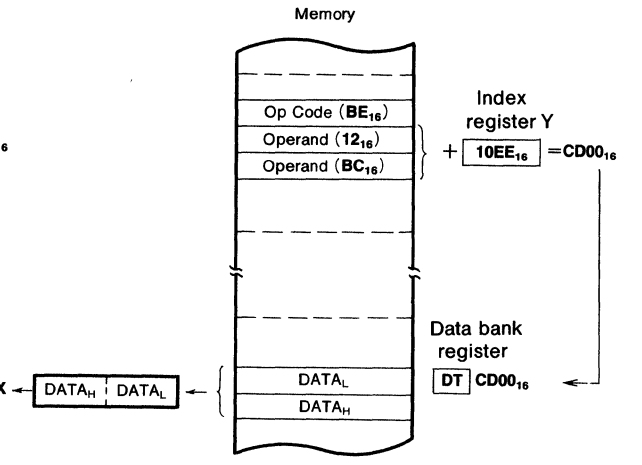
ex. : Mnemonic **LDY 0BC12H, X** **(x=0)** Machine code $BC_{16}\ 12_{16}\ BC_{16}$



ex. : Mnemonic **LDX 0BC12H, Y** (**x=1**) Machine code BE_{16} 12_{16} BC_{16}



ex. : Mnemonic **LDX 0BC12H, Y** (**x=0**) Machine code BE_{16} 12_{16} BC_{16}

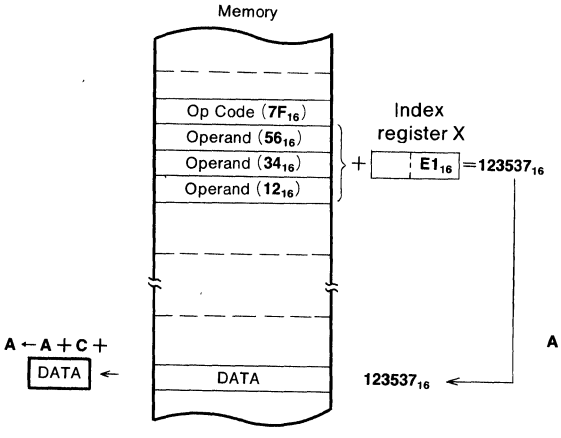


Mode : Absolute long indexed X addressing mode

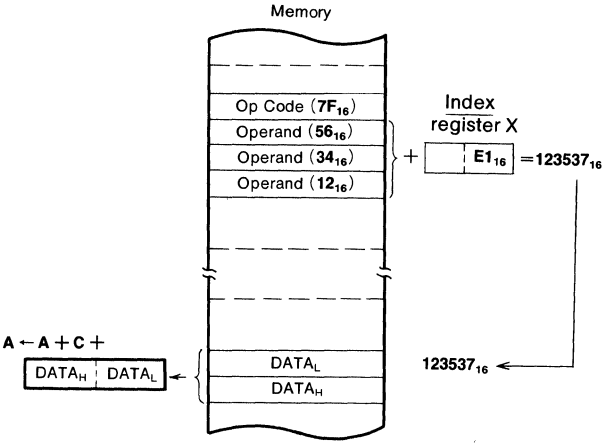
Function : The contents of the memory location specified by adding the numeric value expressed by the instruction's second, third and fourth bytes with the contents of the index register X are the actual data.

Instruction : **ADC, AND, CMP, DIV, EOR, LDA, MPY, ORA, SBC, STA**

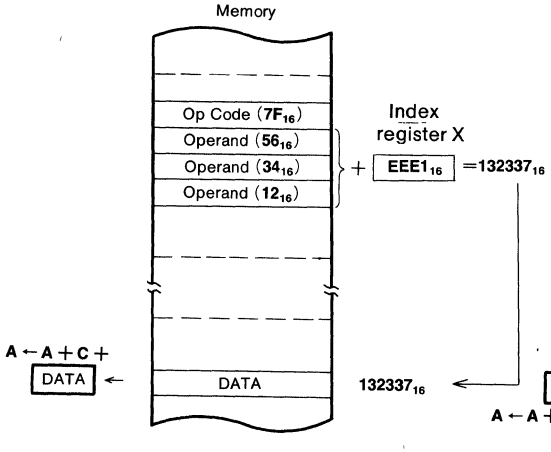
ex. : Mnemonic **ADC A, 123456H, X** (**m=1, x=1**) Machine code $7F_{16}$ 56_{16} 34_{16} 12_{16}



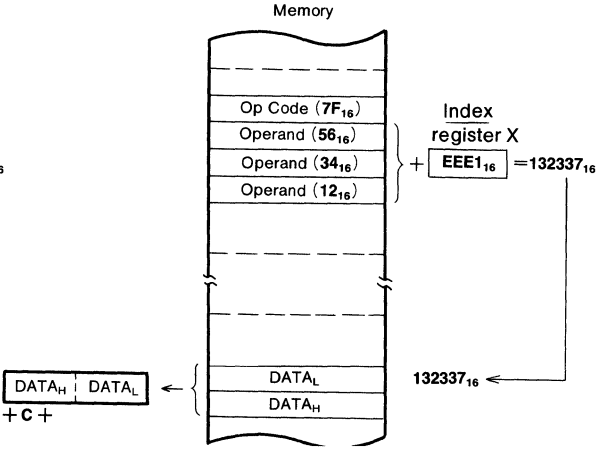
ex. : Mnemonic **ADC A, 123456H, X** (**m=0, x=1**) Machine code $7F_{16}$ 56_{16} 34_{16} 12_{16}



ex. : Mnemonic **ADC A, 123456H, X** (**m=0, x=1**) Machine code $7F_{16}$ 56_{16} 34_{16} 12_{16}



ex. : Mnemonic **ADC A, 123456H, X** (**m=0, x=0**) Machine code $7F_{16}$ 56_{16} 34_{16} 12_{16}



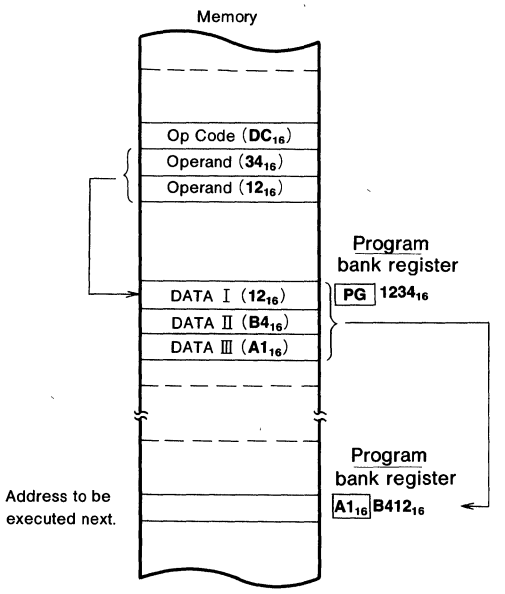
MITSUBISHI ELECTRIC

Mode : **Absolute indirect long addressing mode**

Function : The instruction's second and third bytes specify 3 adjacent bytes in memory, and the contents of these bytes specify the address to which a jump is to be made.

instruction : **JMP**

ex. : Mnemonic **JMPL(1234H)** Machine code **DC_{16} 34_{16} 12_{16}**



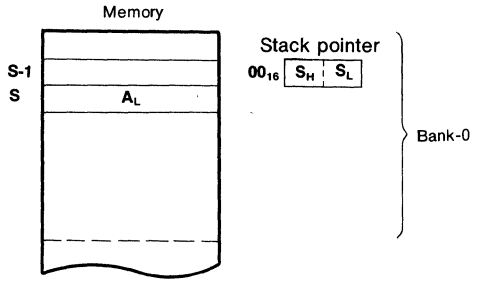
DATA III is loaded in the program bank register

Mode : Stack addressing mode

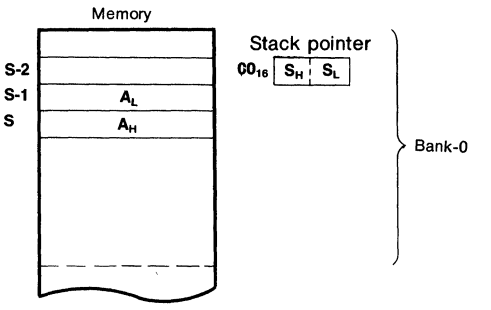
Function : Register contents are saved to or restored from the memory location specified by the stack pointer. The stack pointer is set in bank-0.

Instruction : **PEA, PEI, PER, PHA, PHB, PHD, PHG, PHP, PHT, PHX, PHY, PLA, PLB, PLD, PLP, PLT, PLX, PLY, PSH, PUL**

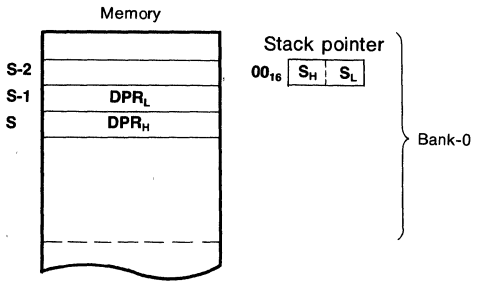
ex. : Mnemonic **PHA** (**m=1**) Machine code 48_{16}



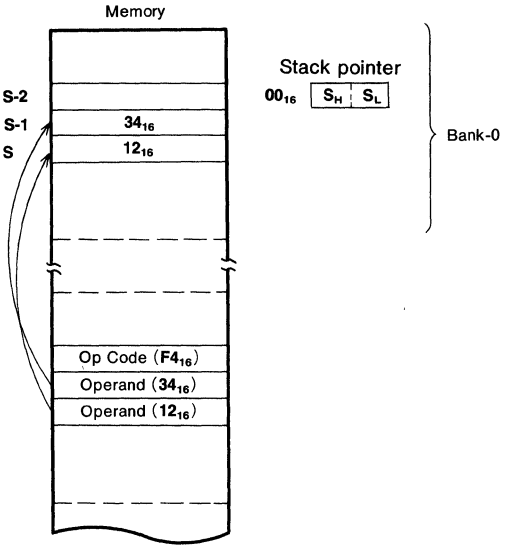
ex. : Mnemonic **PHA** (**m=0**) Machine code 48_{16}



ex. : Mnemonic **PHD** Machine code $0B_{16}$



ex. : Mnemonic **PEA # 1234H** Machine code $F4_{16}$ 34_{16} 12_{16}



Mode : **Relative addressing mode**

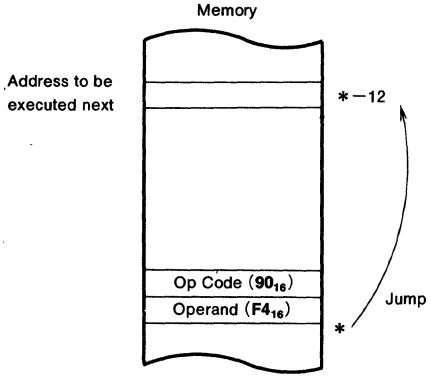
Function : Branching occurs to the address specified by the value resulting from addition of the contents of the program counter and the instruction's second byte. In the case of a long branch by the BRA instruction, a 15-bit signed numeric value formed by the contents of the instruction's second and third bytes is added to the program counter contents. If the addition generates a carry or borrow, 1 is added to or subtracted from the program bank register.

Instruction : **BCC, BCS, BEQ, BMI, BNE, BPL, BRA, BVC, BVS**

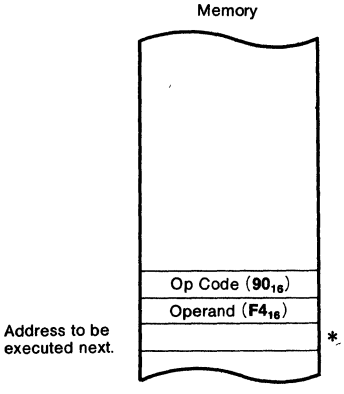
ex. : Mnemonic — Machine code

BCC ＊−12 — **90_{16} $F4_{16}$**

Branches to the address ＊−12 if the carry flag (C) has been cleared

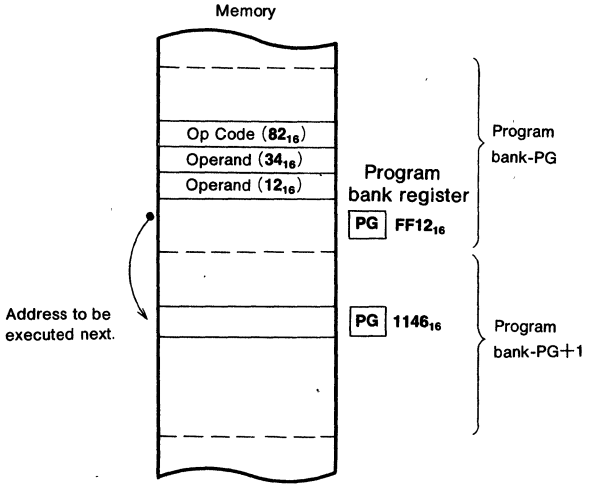


Advances to the address ＊ if the carry flag (C) has been set.



ex. : Mnemonic — Machine code

BRA 1234H — **82_{16} 34_{16} 12_{16}**

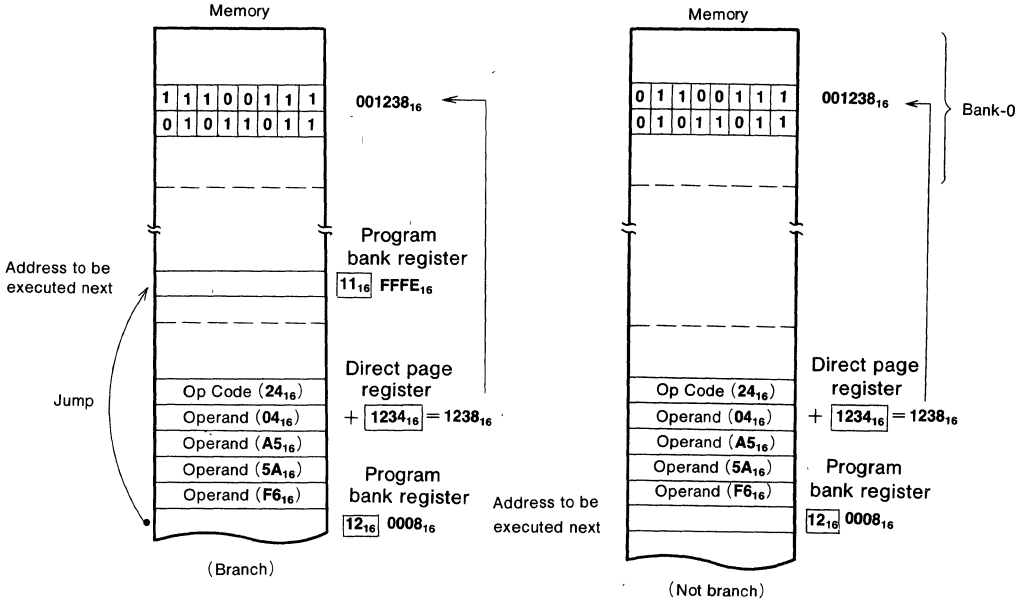


MITSUBISHI ELECTRIC

ex. : Mnemonic — Machine code

BBS #5AA5H, 04H, 0F6H — **24_{16} 04_{16} $A5_{16}$ $5A_{16}$ $F6_{16}$**

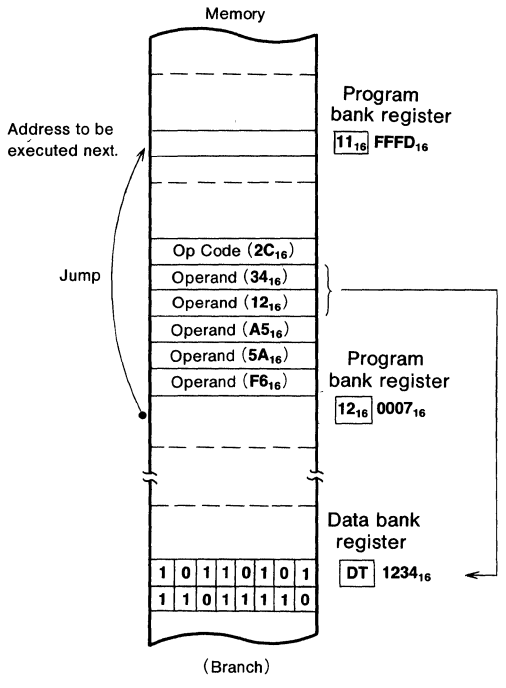
(m=0)

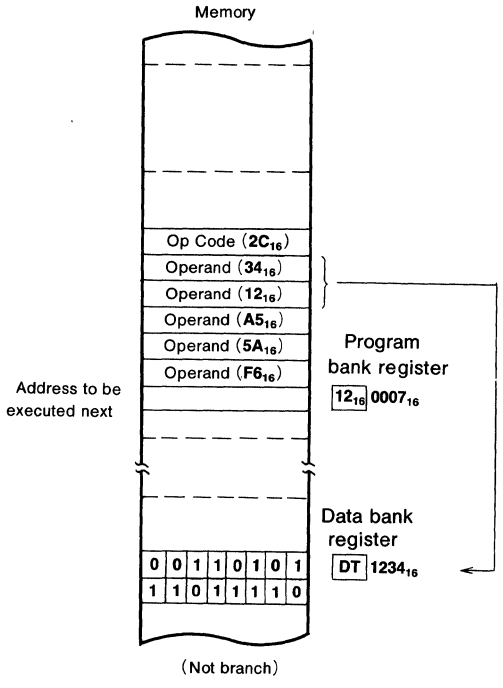


ex. : Mnemonic — Machine code

BBS #5AA5H, 1234H, 0F6H — $2C_{16}$ 34_{16} 12_{16} $A5_{16}$ $5A_{16}$ $F6_{16}$

(m=0)





SERIES MELPS 7700 ADDRESSING MODES

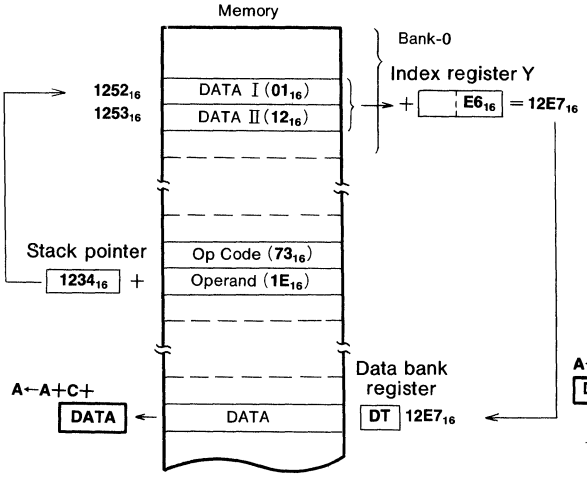
SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER

Mode : Stack pointer relative indirect indexed Y addressing mode

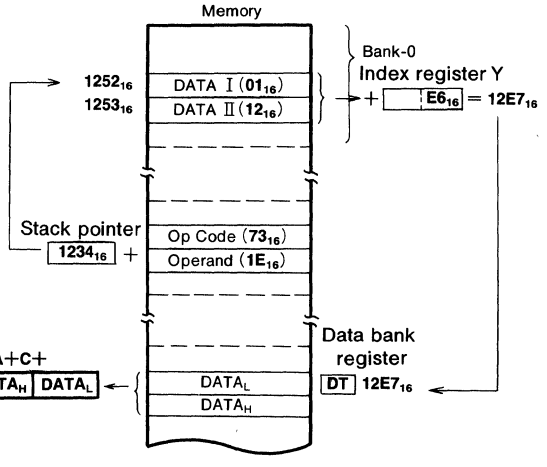
Function : The value obtained by adding the instruction's second byte and the contents of the stack pointer specifies 2 adjacent bytes in memory. The value obtained by adding the contents of these bytes and the contents of the index register Y specifies address of the actual data in memory bank-DT (DT is contents of data bank register). If addition of the 2 bytes in memory with the contents of the index register Y generate a carry, the bank number will be 1 larger than the contents of the data bank register.

Instruction : **ADC, AND, CMP, DIV, EOR, LDA, MPY, ORA, SBC, STA**

ex. : Mnemonic **ADC A,(1EH, S), Y** (**m=1, x=1**) Machine code 73_{16} $1E_{16}$



ex. : Mnemonic **ADC A,(1EH, S), Y** (**m=0, x=1**) Machine code 73_{16} $1E_{16}$



MITSUBISHI ELECTRIC

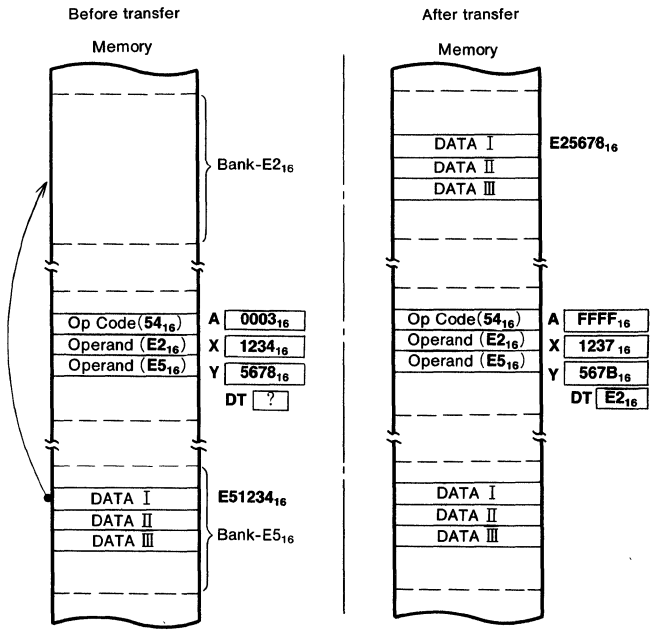
Mode : **Block transfer addressing mode**

Function : The instruction's second byte specifies the transfer-to data bank, and the contents of the index register Y specify the transfer - to address within the data bank. The instruction's third byte specifies the transfer-from data bank, and the contents of the index register X specify the address in the data bank where the data to be transferred is stored. The contents of the accumulator A constitute the number of bytes to be transfeered. Upon termination of transfer, the contents of the data bank register will specify the transfer-to data bank. The MVN instruction is used for transfer to lower address location. In this case, the contents of the index registers X and Y are incremented each time data is transferred. The MVP instruction is used for transfer to higher address location. In this case, the contents of the index registers X and Y are decremented each time data is transferred. The block of data to be transferred may cross over the bank boundary.

Instruction : **MVN, MVP**

ex. : Mnemonic — Machine code

MVN 0E2H, 0E5H — **54_{16} $E2_{16}$ $E5_{16}$**



SERIES MELPS 7700 INSTRUCTION CODE TABLE

SINGLE-CHIP 16-BIT CMOS MICROCOMPUTER

INSTRUCTION CODE TABLE-1

D_7~D_4 \ D_3~D_0 (Hexadecimal notation)		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	BRK	ORA A,(DIR,X)		ORA A,SR	SEB DIR,b	ORA A,DIR	ASL DIR	ORA A,L(DIR)	PHP	ORA A,IMM	ASL A	PHD	SEB ABS,b	ORA A,ABS	ASL ABS	ORA A,ABL
0001	1	BPL	ORA A,(DIR),Y	ORA A,(DIR)	ORA A,(SR),Y	CLB DIR,b	ORA A,DIR,X	ASL DIR,X	ORA A,L(DIR),Y	CLC	ORA A,ABS,Y	DEC A	TAS	CLB ABS,b	ORA A,ABS,X	ASL ABS,X	ORA A,ABL,X
0010	2	JSR ABS	AND A,(DIR,X)	JSR ABL	AND A,SR	BBS DIR,b,R	AND A,DIR	ROL DIR	AND A,L(DIR)	PLP	AND A,IMM	ROL A	PLD	BBS ABS,b,R	AND A,ABS	ROL ABS	AND A,ABL
0011	3	BMI	AND A,(DIR),Y	AND A,(DIR)	AND A,(SR),Y	BBC DIR,b,R	AND A,DIR,X	ROL DIR,X	AND A,L(DIR),Y	SEC	AND A,ABS,Y	INC A	TSA	BBC ABS,b,R	AND A,ABS,X	ROL ABS,X	AND A,ABL,X
0100	4	RTI	EOR A,(DIR,X)	Note 1	EOR A,SR	MVP	EOR A,DIR	LSR DIR	EOR A,L(DIR)	PHA	EOR A,IMM	LSR A	PHG	JMP ABS	EOR A,ABS	LSR ABS	EOR A,ABL
0101	5	BVC	EOR A,(DIR),Y	EOR A,(DIR)	EOR A,(SR),Y	MVN	EOR A,DIR,X	LSR DIR,X	EOR A,L(DIR),Y	CLI	EOR A,ABS,Y	PHY	TAD	JMP ABL	EOR A,ABS,X	LSR ABS,X	EOR A,ABL,X
0110	6	RTS	ADC A,(DIR,X)	PER	ADC A,SR	LDM DIR	ADC A,DIR	ROR DIR	ADC A,L(DIR)	PLA	ADC A,IMM	ROR A	RTL	JMP (ABS)	ADC A,ABS	ROR ABS	ADC A,ABL
0111	7	BVS	ADC A,(DIR),Y	ADC A,(DIR)	ADC A,(SR),Y	LDM DIR,X	ADC A,DIR,X	ROR DIR,X	ADC A,L(DIR),Y	SEI	ADC A,ABS,Y	PLY	TDA	JMP (ABS,X)	ADC A,ABS,X	ROR ABS,X	ADC A,ABL,X
1000	8	BRA REL	STA A,(DIR,X)	BRA REL	STA A,SR	STY DIR	STA A,DIR	STX DIR	STA A,L(DIR)	DEY	Note 2	TXA	PHT	STY ABS	STA A,ABS	STX ABS	STA A,ABL
1001	9	BCC	STA A,(DIR),Y	STA A,(DIR)	STA A,(SR),Y	STY DIR,X	STA A,DIR,X	STX DIR,Y	STA A,L(DIR),Y	TYA	STA A,ABS,Y	TXS	TXY	LDM ABS	STA A,ABS,X	LDM ABS,X	STA A,ABL,X
1010	A	LDY IMM	LDA A,(DIR,X)	LDX IMM	LDA A,SR	LDY DIR	LDA A,DIR	LDX DIR	LDA A,L(DIR)	TAY	LDA A,IMM	TAX	PLT	LDY ABS	LDA A,ABS	LDX ABS	LDA A,ABL
1011	B	BCS	LDA A,(DIR),Y	LDA A,(DIR)	LDA A,(SR),Y	LDY DIR,X	LDA A,DIR,X	LDX DIR,Y	LDA A,L(DIR),Y	CLV	LDA A,ABS,Y	TSX	TYX	LDY ABS,X	LDA A,ABS,X	LDX ABS,Y	LDA A,ABL,X
1100	C	CPY IMM	CMP A,(DIR,X)	CLP IMM	CMP A,SR	CPY DIR	CMP A,DIR	DEC DIR	CMP A,L(DIR)	INY	CMP A,IMM	DEX	WIT	CPY ABS	CMP A,ABS	DEC ABS	CMP A,ABL
1101	D	BNE	CMP A,(DIR),Y	CMP A,(DIR)	CMP A,(SR),Y	PEI	CMP A,DIR,X	DEC DIR,X	CMP A,L(DIR),Y	CLM	CMP A,ABS,Y	PHX	STP	JMP L(ABS)	CMP A,ABS,X	DEC ABS,X	CMP A,ABL,X
1110	E	CPX IMM	SBC A,(DIR,X)	SEP IMM	SBC A,SR	CPX DIR	SBC A,DIR	INC DIR	SBC A,L(DIR)	INX	SBC A,IMM	NOP	PSH	CPX ABS	SBC A,ABS	INC ABS	SBC A,ABL
1111	F	BEQ	SBC A,(DIR),Y	SBC A,(DIR)	SBC A,(SR),Y	PEA	SBC A,DIR,X	INC DIR,X	SBC A,L(DIR),Y	SEM	SBC A,ABS,Y	PLX	PUL	JSR (ABS,X)	SBC A,ABS,X	INC ABS,X	SBC A,ABL,X

Note 1 : 42_{16} specifies the contents of the INSTRUCTION CODE TABLE-2
About the second word's codes, refer to the INSTRUCTION CODE TABLE-2.

2 : 89_{16} specifies the contents of the INSTRUCTION CODE TABLE-3
About the third word's codes, refer to the INSTRUCTION CODE TABLE-2

INSTRUCTION CODE TABLE-3 (The first word's code of each instruction is 89_{16})

D_7~D_4 \ D_3~D_0	Hexadecimal notation	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0		MPY (DIR,X)		MPY SR		MPY DIR		MPY L(DIR)		MPY IMM				MPY ABS		MPY ABL
0001	1		MPY (DIR),Y	MPY (DIR)	MPY (SR),Y		MPY DIR,X		MPY L(DIR),Y		MPY ABS,Y				MPY ABS,X		MPY ABL,X
0010	2		DIV (DIR,X)		DIV SR		DIV DIR		DIV L(DIR)	XAB	DIV IMM				DIV ABS		DIV ABL
0011	3		DIV (DIR),Y	DIV (DIR)	DIV (SR),Y		DIV DIR,X		DIV L(DIR),Y		DIV ABS,Y				DIV ABS,X		DIV ABL,X
0100	4										RLA IMM						
0101	5																
0110	6																
0111	7																
1000	8																
1001	9																
1010	A																
1011	B																
1100	C			LDT IMM													
1101	D																
1110	E																
1111	F																

MACHINE INSTRUCTIONS

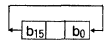
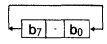
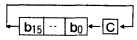
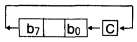
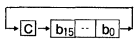
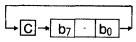
Symbol	Function	Details	Addressing mode																														
			IMP			IMM			A			DIR			DIR,b			DIR,X			DIR,Y			(DIR)			(DIR,X)			(DIR),Y		
			op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#
ADC (Note 1,2)	$A_{CC},C \leftarrow A_{CC}+M+C$	Adds the carry, the accumulator and the memory contents The result is entered into the accumulator When the D flag is "0", binary additions is done, and when the D flag is "1", decimal addition is done				69	2	2				65	4	2				75	5	2				72	6	2	61	7	2	71	8	2
						42 69	4	3				42 65	6	3				42 75	7	3				42 72	8	3	42 61	9	3	42 71	10	3
AND (Note 1,2)	$A_{CC} \leftarrow A_{CC} \wedge M$	Obtains the logical product of the contents of the accumulator and the contents of the memory The result is entered into the accumulator				29	2	2				25	4	2				35	5	2				32	6	2	21	7	2	31	8	2
						42 29	4	3				42 25	6	3				42 35	7	3				42 32	8	3	42 21	9	3	42 31	10	3
ASL (Note 1)	m=0 C ← b_{15} ... b_0 ←0 m=1 C ← b_7 ... b_0 ←0	Shifts the accumulator or the memory contents one bit to the left "0" is entered into bit 0 of the accumulator or the memory The contents of bit 15 (bit 7 when the m flag is "1") of the accumulator or memory before shift is entered into the C flag							0A	2	1	06	7	2				16	7	2												
									42 0A	4	2																					
BBC (Note 3,5)	Mb=0 ?	Tests the specified bit of the memory Branches when all the contents of the specified bit is "0"																															
BBS (Note 3,5)	Mb=1 ?	Tests the specified bit of the memory Branches when all the contents of the specified bit is "1"																															
BCC (Note 3)	C=0 ?	Branches when the contents of the C flag is "0"																															
BCS (Note 3)	C=1 ?	Branches when the contents of the C flag is "1"																															
BEQ (Note 3)	Z=1 ?	Branches when the contents of the Z flag is "1"																															
BMI (Note 3)	N=1 ?	Branches when the contents of the N flag is "1"																															
BNE (Note 3)	Z=0 ?	Branches when the contents of the Z flag is "0"																															
BPL (Note 3)	N=0 ?	Branches when the contents of the N flag is "0"																															
BRA (Note 4)	PC←PC±offset PG←PG+1 (carry occured) PG←PG−1 (borrow occured)	Jumps to the address indicated by the program counter plus the offset value																															
BRK	PC←PC+2 M(S)←PG S←S−1 M(S)←PC_H S←S−1 M(S)←PC_L S←S−1 M(S)←PS_H S←S−1 M(S)←PS_L S←S−1 I←1 PC_L←AD_L PC_H←AD_H PG←00_{16}	Executes software interruption	00	15	2																												
BVC (Note 3)	V=0 ?	Branches when the contents of the V flag is "0"																															
BVS (Note 3)	V=1 ?	Branches when the contents of the V flag is "1"																															
CLB (Note 5)	Mb←0	Makes the contents of the specified bit in the memory "0"													14	8	3																
CLC	C←0	Makes the contents of the C flag "0"	18	2	1																												
CLI	I←0	Makes the contents of the I flag "0"	58	2	1																												
CLM	m←0	Makes the contents of the m flag "0"	D8	2	1																												
CLP	PSb←0	Specifies the bit position in the processor status register by the bit pattern of the second byte in the instruction, and sets "0" in that bit				C2	4	2																									
CLV	V←0	Makes the contents of the V flag "0"	B8	2	1																												
CMP (Note 1,2)	A_{CC}−M	Compares the contents of the accumulator with the contents of the memory				C9	2	2				C5	4	2				D5	5	2				D2	6	2	C1	7	2	D1	8	2
						42 C9	4	3				42 C5	6	3				42 D5	7	3				42 D2	8	3	42 C1	9	3	42 D1	10	3

MITSUBISHI ELECTRIC

Symbol	Function	Details	Addressing mode																																
			IMP			IMM			A			DIR			DIR,b			DIR,X			DIR,Y			(DIR)			(DIR,X)			(DIR),Y					
			op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#			
CPX (Note 2)	X−M	Compares the contents of the index register X with the contents of the memory				E0	2	2				E4	4	2																					
CPY (Note 2)	Y−M	Compares the contents of the index register Y with the contents of the memory				C0	2	2				C4	4	2																					
DEC (Note 1)	A_{CC}←A_{CC}−1 or M←M−1	Decrements the contents of the accumulator or memory by 1							1A	2	1	C6	7	2				D6	7	2															
									42 1A	4	2																								
DEX	X←X−1	Decrements the contents of the index register X by 1	CA	2	1																														
DEY	Y←Y−1	Decrements the contents of the index register Y by 1	88	2	1																														
DIV (Note 2,10)	A(quotient)←B,A/M B(remainder)	The numeral that places the contents of accumulator B to the higher order and the contents of accumulator A to the lower order is divided by the contents of the memory The quotient is entered into accumulator A and the remainder into accumulator B				89 29	27	3				89 25	29	3				89 35	30	3				89 32	31	3	89 21	32	3	89 31	33	3			
EOR (Note 1,2)	A_{CC}←A_{CC}¥M	Logical exclusive sum is obtained of the contents of the accumulator and the contents of the memory The result is placed into the accumulator				49	2	2				45	4	2				55	5	2				52	6	2	41	7	2	51	8	2			
						42 49	4	3				42 45	6	3				42 55	7	3				42 52	8	3	42 41	9	3	42 51	10	3			
INC (Note 1)	A_{CC}←A_{CC}+1 or M ←M+1	Increments the contents of the accumulator or memory by 1							3A	2	1	E6	7	2				F6	7	2															
									42 3A	4	2																								
INX	X←X+1	Increments the contents of the index register X by 1	E8	2	1																														
INY	Y←Y+1	Increments the contents of the index register Y by 1	C8	2	1																														
JMP	ABS PC_L ← AD_L PC_H ← AD_H ABL PC_L ← AD_L PC_H ← AD_H PG←AD_G (ABS) PC_L ←(AD_H, AD_L) PC_H ←(AD_H, AD_L+1) L(ABS) PC_L ←(AD_H, AD_L) PC_H ←(AD_H, AD_L+1) PG ←(AD_H, AD_L+2) (ABS, X) PC_L ←(AD_H, AD_L+X) PC_H ←(AD_H, AD_L+X +1)	Places a new address into the program counter and jumps to that new address																																	
JSR	ABS M(S)← PC_H S←S−1 M(S)← PC_L S←S−1 PC_L ← AD_L PC_H ← AD_H ABL M(S)← PG S←S−1 M(S)← PC_H S←S−1 M(S)←PC_L S ← S−1 PC_L ← AD_L PC_H ← AD_H PG ← AD_G (ABS, X) M(S)← PC_H S←S−1 M(S)← PC_L S←S−1 PC_L ←(AD_H, AD_L+X) PC_H ←(AD_H, AD_L+X +1)	Saves the contents of the program counter (also the contents of the program bank register for ABL) into the stack, and jumps to the new address.																																	

Symbol	Function	Details	Addressing mode																																	
			IMP			IMM			A			DIR			DIR,b			DIR,X			DIR,Y			(DIR)			(DIR,X)			(DIR),Y					
			op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#
LDA (Note 1,2)	$A_{CC} \leftarrow M$	Enters the contents of the memory into the accumulator				A9	2	2				A5	4	2				B5	5	2				B2	6	2	A1	7	2	B1	8	2
						42 A9	4	3				42 A5	6	3				42 B5	7	3				42 B2	8	3	42 A1	9	3	42 B1	10	3
LDM (Note 5)	$M \leftarrow IMM$	Enters the immediate value into the memory.										64	4	3				74	5	3												
LDT	$DT \leftarrow IMM$	Enters the immediate value into the data bank register				89 C2	5	3																								
LDX (Note 2)	$X \leftarrow M$	Enters the contents of the memory into index register X				A2	2	2				A6	4	2							B6	5	2									
LDY (Note 2)	$Y \leftarrow M$	Enters the contents of the memory into index register Y				A0	2	2				A4	4	2				B4	5	2												
LSR (Note 1)	m=0 0→[b_{15} ··· b_0]→C m=1 0→[b_7 b_0]→C	Shifts the contents of the accumulator or the contents of the memory one bit to the right The bit 0 of the accumulator or the memory is entered into the C flag "0" is entered into bit 15 (bit 7 when the m flag is "1")							4A	2	1	46	7	2				56	7	2												
									42 4A	4	2																					
MPY (Note 2,11)	B, A←A*M	Multiplies the contents of accumulator A and the contents of the memory The higher order of the result of operation are entered into accumulator B, and the lower order into accumulator A				89 09	16	3				89 05	18	3				89 15	19	3				89 12	20	3	89 01	21	3	89 11	22	3
MVN (Note 8)	$Mn+1 \leftarrow Mm+1$	Transmits the data block The transmission is done from the lower order address of the block																														
MVP (Note 9)	$Mn-1 \leftarrow Mm-1$	Transmits the data block Transmission is done form the higher order address of the data block																														
NOP	PC←PC+1	Advances the program counter, but performs nothing else	EA	2	1																											
ORA (Note 1,2)	$A_{CC} \leftarrow A_{CC} \vee M$	Logical sum per bit of the contents of the accumulator and the contents of the memory is obtained The result is entered into the accumulator				09	2	2				05	4	2				15	5	2				12	6	2	01	7	2	11	8	2
						42 09	4	3				42 05	6	3				42 15	7	3				42 12	8	3	42 01	9	3	42 11	10	3
PEA	$M(S) \leftarrow IMM_2$ $S \leftarrow S-1$ $M(S) \leftarrow IMM_1$ $S \leftarrow S-1$	The 3rd and the 2nd bytes of the instruction are saved into the stack, in this order																														
PEI	$M(S) \leftarrow M((DPR)+IMM+1)$ $S \leftarrow S-1$ $M(S) \leftarrow M((DPR)+IMM)$ $S \leftarrow S-1$	Specifies 2 sequential bytes in the direct page in the 2nd byte of the instruction, and saves the contents into the stack																														
PER	$EAR \leftarrow PC+IMM_2.IMM_1$ $M(S) \leftarrow EAR_H$ $S \leftarrow S-1$ $M(S) \leftarrow EAR_L$ $S \leftarrow S-1$	Regards the 2nd and 3rd bytes of the instruction as 16-bit numerals, adds them to the program counter, and saves the result into the stack																														
PHA	m=0 $M(S) \leftarrow A_H$ $S \leftarrow S-1$ $M(S) \leftarrow A_L$ $S \leftarrow S-1$ m=1 $M(S) \leftarrow A_L$ $S \leftarrow S-1$	Saves the contents of accumulator A into the stack																														
PHB	m=0 $M(S) \leftarrow B_H$ $S \leftarrow S-1$ $M(S) \leftarrow B_L$ $S \leftarrow S-1$ m=1 $M(S) \leftarrow B_L$ $S \leftarrow S-1$	Saves the contents of accumulator B into the stack																														

Symbol	Function	Details	Addressing mode																																		
			IMP			IMM			A			DIR			DIR,b			DIR,X			DIR,Y			(DIR)			(DIR,X)			(DIR),Y							
			op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#					
PHD	M(S)←DPR_H S←S−1 M(S)←DPR_L S←S−1	Saves the contents of the direct page register into the stack																																			
PHG	M(S)←PG S←S−1	Saves the contents of the program bank register into the stack																																			
PHP	M(S)←PS_H S←S−1 M(S)←PS_L S←S−1	Saves the contents of the program status register into the stack																																			
PHT	M(S)←DT S←S−1	Saves the contents of the data bank register into the stack																																			
PHX	x=0 M(S)←X_H S←S−1 M(S)←X_L S←S−1 x=1 M(S)←X_L S←S−1	Saves the contents of the index register X into the stack																																			
PHY	x=0 M(S)←Y_H S←S−1 M(S)←Y_L S←S−1 x=1 M(S)←Y_L S←S−1	Saves the contents of the index register Y into the stack																																			
PLA	m=0 S←S+1 A_L←M(S) S←S+1 A_H←M(S) m=1 S←S+1 A_L←M(S)	Restores the contents of the stack on the accumulator A																																			
PLB	m=0 S←S+1 B_L←M(S) S←S+1 B_H←M(S) m=1 S←S+1 B_L←M(S)	Restores the contents of the stack on the accumulator B																																			
PLD	S←S+1 DPR_L←M(S) S←S+1 DPR_H←M(S)	Restores the contents of the stack on the direct page register																																			
PLP	S←S+1 PS_L←M(S) S←S+1 PS_H←M(S)	Restores the contents of the stack on the processor status register																																			
PLT	S←S+1 DT←M(S)	Restores the contents of the stack on the data bank register																																			
PLX	x=0 S←S+1 X_L←M(S) S←S+1 X_H←M(S) x=1 S←S+1 X_L←M(S)	Restores the contents of the stack on the index register X																																			

Symbol	Function	Details	Addressing mode																																		
			IMP			IMM			A			DIR			DIR,b			DIR,X			DIR,Y			(DIR)			(DIR,X)			(DIR),Y								
			op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#						
PLY	x=0 S←S+1 Y_L←M(S) S←S+1 Y_H←M(S) x=1 S←S+1 Y_L←M(S)	Restores the contents of the stack on the index register Y																																			
PSH (Note 6)	M(S)←A, B, X···	Saves the registers among accumulator, index register, direct page register, data bank register, program bank register, or processor status register, specified by the bit pattern of the second byte of the instruction into the stack																																			
PUL (Note 7)	A, B, X·· ←M(S)	Restores the contents of the stack to the registers among accumulator, index register, direct page register, data bank register, or processor status register, specified by the bit pattern of the second byte of the instruction																																			
RLA (Note 13)	m=0 n bit rotate left  m=1 n bit rotate left 	Rotates the contents of the accumulator A, n bits to the left				89 49	6 + 1	3																															
ROL (Note 1)	m=0  m=1 	Links the accumulator or the memory to C flag, and rotates result to the left by 1 bit							2A 42 2A	2 4	1 2	26	7	2				36	7	2																			
ROR (Note 1)	m=0  m=1 	Links the accumulator or the memory to C flag, and rotates result to the right by 1 bit							6A 42 6A	2 4	1 2	66	7	2				76	7	2																			
RTI	S←S+1 PS_L←M(S) S←S+1 PS_H←M(S) S←S+1 PC_L←M(S) S←S+1 PC_H←M(S) S←S+1 PG←M(S)	Returns from the interruption routine	40	11	1																																
RTL	S←S+1 PC_L←M(S) S←S+1 PC_H←M(S) S←S+1 PG←M(S)	Returns from the subroutine The contents of the program bank register are also restored	6B	8	1																																
RTS	S←S+1 PC_L←M(S) S←S+1 PC_H←M(S)	Returns from the subroutine The contents of the program bank register are not restored	60	5	1																																
SBC (Note 1,2)	A_{CC}, C←A_{CC}−M−$\overline{C}$	Subtracts the contents of the memory and the borrow from the contents of the accumulator				E9 42 E9	2 4	2 3				E5 42 E5	4 6	2 3				F5 42 F5	5 7	2 3				F2 42 F2	6 8	2 3	E1 42 E1	7 9	2 3	F1 42 F1	8 10	2 3						

Symbol	Function	Details	Addressing mode																																
			IMP			IMM			A			DIR			DIR,b			DIR,X			DIR,Y			(DIR)			(DIR,X)			(DIR),Y					
			op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#	op	n	#			
SEB (Note 5)	Mb←1	Makes the contents of the specified bit in the memory "1"													04	8	3																		
SEC	C←1	Makes the contents of the C flag "1"	38	2	1																														
SEI	I←1	Makes the contents of the I flag "1".	78	2	1																														
SEM	m←1	Makes the contents of the m flag "1"	F8	2	1																														
SEP	PSb←1	Set the specified bit of the processor status register's lower byte (PS_L) to "1"				E2	3	2																											
STA (Note 1)	M←A_{CC}	Stores the contents of the accumulator into the memory										85	4	2				95	5	2				92	7	2	81	7	2	91	7	2			
												42 85	6	3				42 95	7	3				42 92	9	3	42 81	9	3	42 91	9	3			
STP		Stops the oscillation of the oscillator.	DB	3	1																														
STX	M←X	Stores the contents of the index register X into the memory.										86	4	2							96	5	2												
STY	M←Y	Stores the contents of the index register Y into the memory.										84	4	2				94	5	2															
TAD	DPR←A	Transmits the contents of the accumulator A to the direct page register	5B	2	1																														
TAS	S←A	Transmits the contents of the accumulator A to the stack pointer	1B	2	1																														
TAX	X←A	Transmits the contents of the accumulator A to the index register X	AA	2	1																														
TAY	Y←A	Transmits the contents of the accumulator A to the index register Y.	A8	2	1																														
TBD	DPR←B	Transmits the contents of the accumulator B to the direct page register	42 5B	4	2																														
TBS	S←B	Transmits the contents of the accumulator B to the stack pointer.	42 1B	4	2																														
TBX	X←B	Transmits the contents of the accumulator B to the index register X.	42 AA	4	2																														
TBY	Y←B	Transmits the contents of the accumulator B to the index register Y	42 A8	4	2																														
TDA	A←DPR	Transmits the contents of the direct page register to the accumulator A.	7B	2	1																														
TDB	B←DPR	Transmits the contents of the direct page register to the accumulator B.	42 7B	4	2																														
TSA	A←S	Transmits the contents of the stack pointer to the accumulator A	3B	2	1																														
TSB	B←S	Transmits the contents of the stack pointer to the accumulator B.	42 3B	4	2																														
TSX	X←S	Transmits the contents of the stack pointer to the index register X	BA	2	1																														
TXA	A←X	Transmits the contents of the index register X to the accumulator A	8A	2	1																														
TXB	B←X	Transmits the contents of the index register X to the accumulator B	42 8A	4	2																														
TXS	S←X	Transmits the contents of the index register X to the stack pointer.	9A	2	1																														
TXY	Y←X	Transmits the contents of the index register X to the index register Y	9B	2	1																														
TYA	A←Y	Transmits the contents of the index register Y to the accumulator A.	98	2	1																														
TYB	B←Y	Transmits the contents of the index register Y to the accumulator B	42 98	4	2																														
TYX	X←Y	Transmits the contents of the index register Y to the index register X.	BB	2	1																														
WIT		Stops the internal clock.	CB	3	1																														
XAB	A⇄B	Exchanges the contents of the accumulator A and the contents of the accumulator B	89 28	6	2																														

The number of cycles shown in the table is described in case of the fastest mode for each instruction. The number of cycles shown in the table is calculated for $DPR_L=0$. The number of cycles in the addressing mode concerning the DPR when $DPR_L \neq 0$ must be incremented by 1
The number of cycles shown in the table differs according to the bytes fetched into the instruction queue buffer, or according to whether the memory read/write address is odd or even. It also differs when the external region memory is accessed by BYTE="H"

Note 1. The operation code at the upper row is used for accumulator A, and the operation at the lower row is used for accumulator B.

Note 2. When setting flag m=0 to handle the data as 16-bit data in the immediate addressing mode, the number of bytes increments by 1

Note 3. The number of cycles increments by 2 when branching

Note 4. The operation code on the upper row is used for branching in the range of −128~+127, and the operation code on the lower row is used for branching in the range of −32768~+32767

Note 5. When handling 16-bit data with flag m=0, the byte in the table is incremented by 1

Note 6.

Type of register	A	B	X	Y	DPR	DT	PG	PS
Number of cycles	2	2	2	2	2	1	1	2

The number of cycles corresponding to the register to be pushed are added The number of cycles when no pushing is done is 12 i_1 indicates the number of registers among A, B, X, Y, DPR, and PS to be saved, while i_2 indicates the number of registers among DT and PG to be saved.

Note 7.

Type of register	A	B	X	Y	DPR	DT	PS
Number of cycles	3	3	3	3	4	3	3

The number of cycles corresponding to the register to be pulled are added The number of cycles when no pulling is done is 14 i_1 indicates the number of registers among A, B, X, Y, DT, and PS to be restored, while $i_2=1$ when DPR is to be restored

Note 8. The number of cycles is the case when the number of bytes to be transfered is even
When the number of bytes to be transfered is odd, the number is calculated as,

$$7+(i/2)\times 7+4$$

Note that, (i/2) shows the integer part when i is divided by 2.

Note 9. The number of cycles is the case when the number of bytes to be transfered is even
When the number of bytes to be transfered is odd, the number is calculated as,

$$9+(i/2)\times 7+5$$

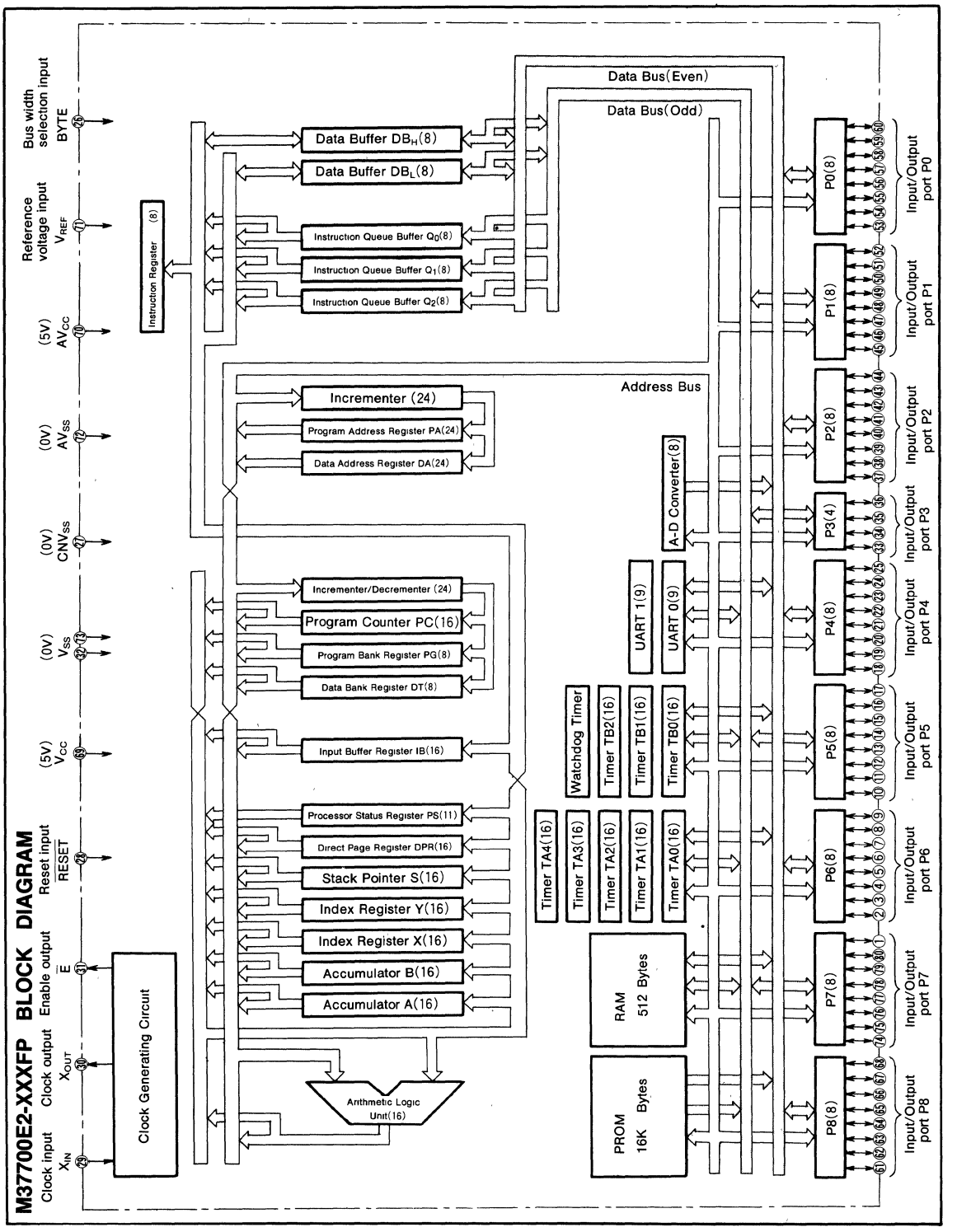
Note that, (i/2) shows the integer part when i is divided by 2

Note 10. The number of cycles is the case in the 16-bit÷8-bit operation. The number of cycles is incremented by 16 for 32-bit÷16-bit operation.

Note 11. The number of cycles is the case in the 8-bit×8-bit operation The number of cycles is incremented by 8 for 16-bit ×16-bit operation

Note 12. When setting flag x=0 to handle the data as 16-bit data in the immediate addressing mode, the number of bytes increments by 1

Note 13. When flag m is 0, the byte in the table is incremented by 1.



MITSUBISHI ELECTRIC

PIN DESCRIPTION (NORMAL MODE)

Pin	Name	Input/Output	Functions
V_{CC}, V_{SS}	Power supply		Supply 5 V±5 % to V_{CC} and 0 V to V_{SS}
CNV_{SS}	CNV_{SS} input	Input	This pin controls the processor mode. Connect to V_{SS} for single-chip mode.
$\overline{RESET}$	Reset input	Input	To enter the reset state, this pin must be kept at a "L" condition which should be maintained for the required time
X_{IN}	Clock input	Input	These are I/O pins of internal clock generating circuit Connect a ceramic or quartz crystal resonator between X_{IN} and X_{OUT} When an external clock is used, the clock source should be connected to the X_{IN} pin and the X_{OUT} pin should be left open.
X_{OUT}	Clock output	Output	
$\overline{E}$	Enable output	Output	Data or instruction read and data write are performed when output from this pin is "L".
BYTE	Bus width selection input	Input	In memory expansion mode or microprocessor mode, this pin determines whether the external data bus is 8-bit width or 16-bit width The width is 16 bits when "L" signal inputs and 8 bits when "H" signal inputs.
AV_{CC}, AV_{SS}	Analog supply input		Power supply for the A-D converter Connect AV_{CC} to V_{CC} and AV_{SS} to V_{SS} externally.
V_{REF}	Reference voltage input	Input	This is reference voltage input pin for the A-D converter.
$P0_0$~$P0_7$	I/O port P0	I/O	In single-chip mode, port P0 becomes an 8-bit I/O port An I/O directional register is available so that each pin can be programmed for input or output. These ports are in input mode when reset . Address(A_7~A_0)is output in memory expansion mode or microprocessor mode
$P1_0$~$P1_7$	I/O port P1	I/O	In single-chip mode, these pins have the same functions as port P0. When the BYTE pin is set to "L" in memory expansion mode or microprocessor mode and external data bus is 16-bit width, high-order data (D_{15}~D_8)is input or output when $\overline{E}$ output is "L" and an address (A_{15}~A_8) is output when $\overline{E}$ output is "H" If the BYTE pin is "H" that is an external data bus is 8-bit width, only address(A_{15}~A_8)is output.
$P2_0$~$P2_7$	I/O port P2	I/O	In single-chip mode, these pins have the same functions as port P0 In memory expansion mode or microprocessor mode low-order data(D_7~D_0) is input or output when $\overline{E}$ output is "L" and an address(A_{23}~A_{16})is output when $\overline{E}$ output is "H".
$P3_0$~$P3_3$	I/O port P3	I/O	In single-chip mode , these pins have the same functions as port P0 In memory expansion mode or microprocessor mode, R/$\overline{W}$, $\overline{BHE}$, ALE, and $\overline{HLDA}$ signals are output.
$P4_0$~$P4_7$	I/O port P4	I/O	In single-chip mode, these pins have the same functions as port P0 In memory expansion mode or microprocessor mode, $P4_0$ and $P4_1$ become $\overline{HOLD}$ and $\overline{RDY}$ input pin respectively. Functions of other pins are the same as in single-chip mode Port $P4_2$ can be programed for ϕ_1 output pin divided the clock to X_{IN} pin by 2
$P5_0$~$P5_7$	I/O port P5	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as I/O pins for timer A0, timer A1, timer A2, and timer A3
$P6_0$~$P6_7$	I/O port P6	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as I/O pins for timer A4, external interrupt input $\overline{INT_0}$, $\overline{INT_1}$, and $\overline{INT_2}$ pins, and input pins for timer B0, timer B1, and timer B2
$P7_0$~$P7_7$	I/O port P7	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as analog input AN_0~AN_7 input pins $P7_7$ also has an A-D conversion trigger input function.
$P8_0$~$P8_7$	I/O port P8	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as R_XD, T_XD, CLK, $\overline{CTS}$/$\overline{RTS}$ pins for UART 0 and UART 1

EPROM MODE

The M37700E2-XXXFP features an EPROM mode in addition to its normal modes. When the $\overline{RESET}$ signal level is "L", the chip automatically enters the EPROM mode. Table 1 list the correspondence between pins and Fig. 2 gives the pin connections in the EPROM mode.

When in the EPROM mode, ports P0, P1, P2, $P5_1$, $P5_2$, CNV_{SS} and BYTE are used for the EPROM (equivalent to the M5M27C256K). When in this mode, the built-in PROM can be written to or read from using these pins in the same way as with the M5M27C256K.

This chip does not have Device Identifer Mode, so that set the corresponding program algorithm. The program area should specify address 4000_{16}~$7FFF_{16}$ for the M37700E2-XXXFP.

Connect the clock which is either ceramic resonator or external clock to X_{IN} pin and X_{OUT} pin.

Table 1 Pin function in EPROM programming mode

	M37700E2-XXXFP	M5M27C256K
V_{CC}	V_{CC}	V_{CC}
V_{PP}	CNV_{SS}, BYTE	V_{PP}
V_{SS}	V_{SS}	V_{SS}
Address input	Ports P0, $P1_0$~$P1_6$	A_0~A_{14}
Data I/O	Port P2	D_0~D_7
$\overline{CE}$	$P5_2$	$\overline{CE}$
$\overline{OE}$	$P5_1$	$\overline{OE}$

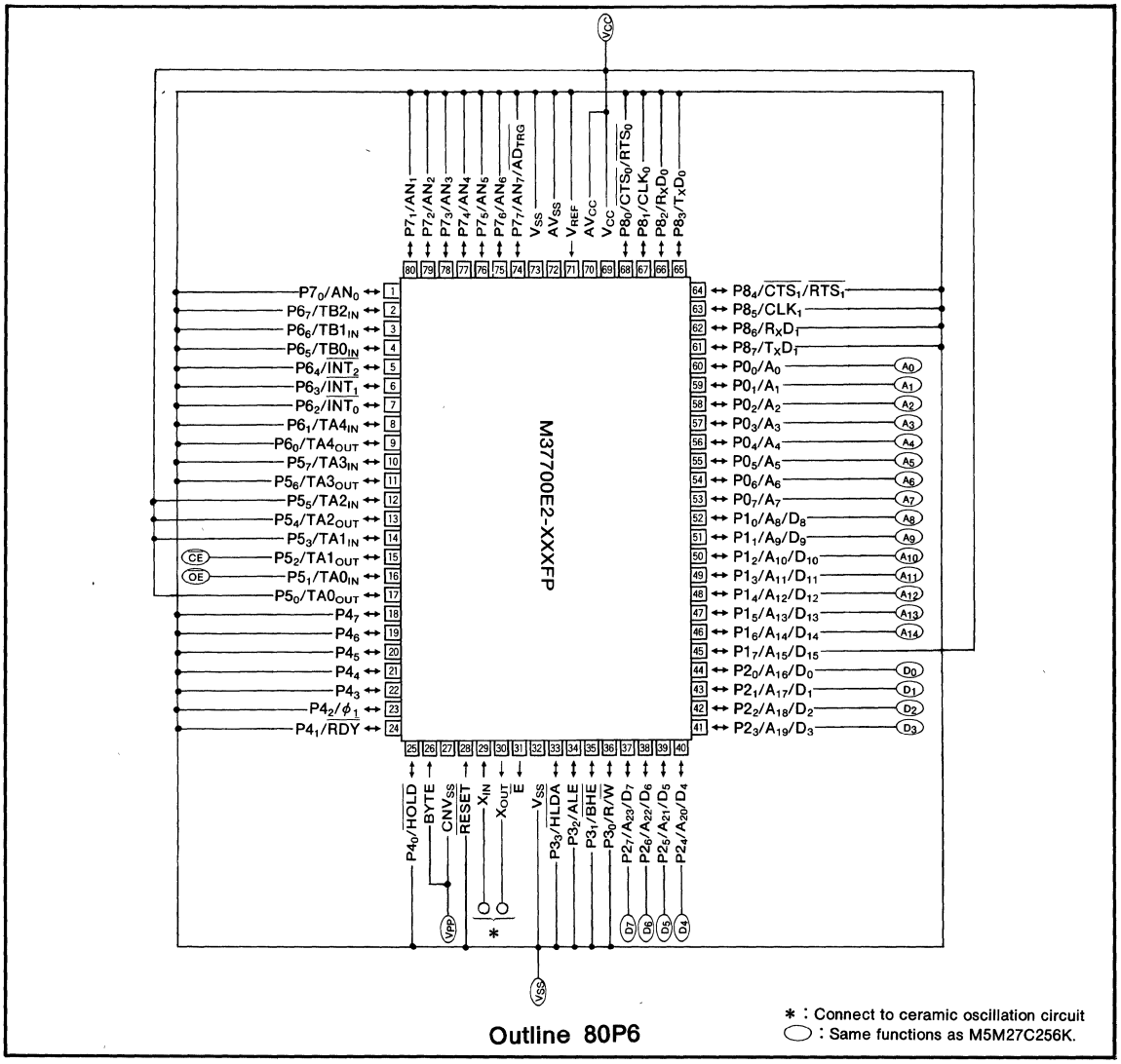
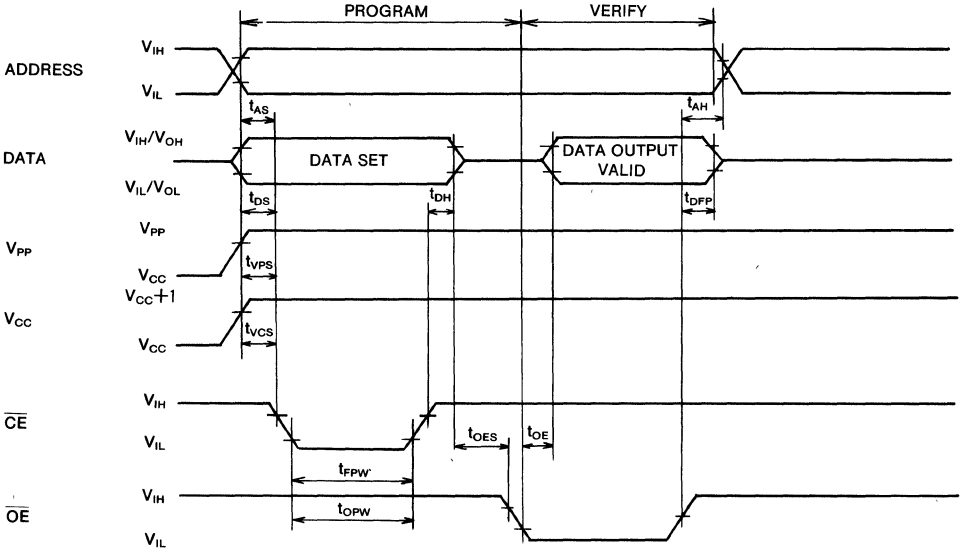
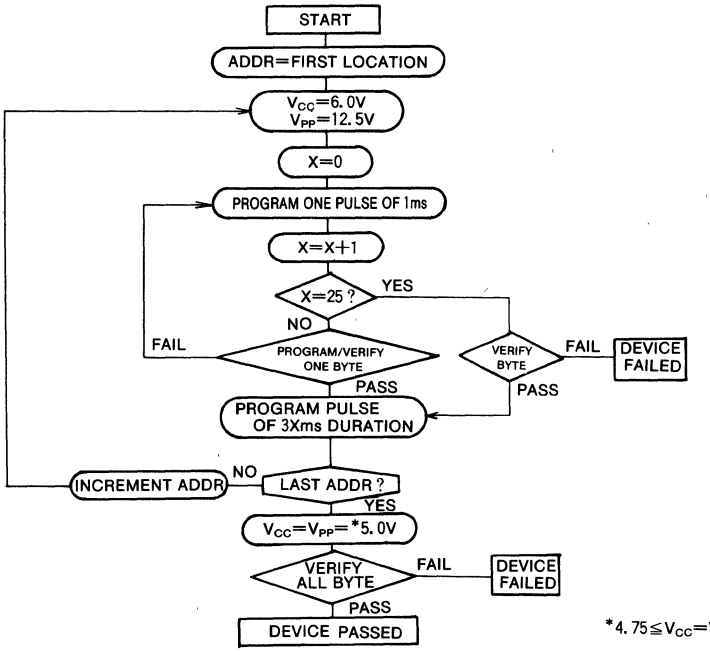


Fig. 1 Pin connection in EPROM programming mode

AC waveforms



Fast programming algorithm flow chart



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Supply voltage		−0.3∼7	V
AV_{CC}	Analog supply voltage		−0.3∼7	V
V_I	Input voltage $\overline{RESET}$, CNV_{SS}, BYTE		−0.3∼12	V
V_I	Input voltage $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$, V_{REF}, X_{IN}		−0.3∼V_{CC}+0.3	V
V_O	Output voltage $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$, X_{OUT}, $\overline{E}$		−0.3∼V_{CC}+0.3	V
P_d	Power dissipation	T_a=25℃	300	mW
Topr	Operating temperature		−10∼70	℃
Tstg	Storage temperature		−40∼125	℃

Note 1. Input voltage for CNV_{SS} and BYTE pins is 13V in writing to PROM

RECOMMENDED OPERATING CONDITIONS (V_{CC}=5V±5%, T_a=−10∼70℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Typ	Max	
V_{CC}	Supply voltage	4.75	5.0	5.25	V
AV_{CC}	Analog supply voltage		V_{CC}		V
V_{SS}	Supply voltage		0		V
AV_{SS}	Analog supply voltage		0		V
V_{IH}	High-level input voltage $P0_0$∼$P0_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	0.8V_{CC}		V_{CC}	V
V_{IH}	High-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in single-chip mode)	0.8V_{CC}		V_{CC}	V
V_{IH}	High-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in memory expansion mode and microprocessor mode)	0.5V_{CC}		V_{CC}	V
V_{IL}	Low-level input voltage $P0_0$∼$P0_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	0		0.2V_{CC}	V
V_{IL}	Low-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in single-chip mode)	0		0.2V_{CC}	V
V_{IL}	Low-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in memory expansion mode and microprocessor mode)	0		0.16V_{CC}	V
$I_{OH(peak)}$	High-level peak output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$			−10	mA
$I_{OH(avg)}$	High-level average output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$			−5	mA
$I_{OL(peak)}$	Low-level peak output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$			10	mA
$I_{OL(avg)}$	Low-level average output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$			5	mA
$f(X_{IN})$	External clock frequency input — M37700M2-XXXFP, M37700SFP			8	MHz
	External clock frequency input — M37700M2AXXXFP, M37700SAFP			16	MHz

Note 1. Average output current is the average value of a100ms interval
2. The sum of $I_{OL(peak)}$ for ports P0, P1, P2, P3 and P8 must be 80mA or less, the sum of $I_{OH(peak)}$ for ports P0, P1, P2, P3 and P8 must be 80mA or less, the sum of $I_{OL(peak)}$ for ports P4, P5, P6 and P7 must be 80mA or less, and the sum of $I_{OH(peak)}$ for ports P4, P5, P6 and P7 must be 80mA or less

TIMING REQUIREMENTS (V_{CC}=5V±5%, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

External clock input

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
t_C	External clock input cycle time		125			ns
$t_{W(H)}$	External clock input high-level pulse width		50			ns
$t_{W(L)}$	External clock input low-level pulse width		50			ns
t_r	External clock rise time				20	ns
t_f	External clock fall time				20	ns

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{SU(P0D-E)}$	Port P0 input setup time		200			ns
$t_{SU(P1D-E)}$	Port P1 input setup time		200			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		200			ns
$t_{SU(P3D-E)}$	Port P3 input setup time		200			ns
$t_{SU(P4D-E)}$	Port P4 input setup time		200			ns
$t_{SU(P5D-E)}$	Port P5 input setup time		200			ns
$t_{SU(P6D-E)}$	Port P6 input setup time		200			ns
$t_{SU(P7D-E)}$	Port P7 input setup time		200			ns
$t_{SU(P8D-E)}$	Port P8 input setup time		200			ns
$t_{h(E-P0D)}$	Port P0 input hold time		0			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(E-P3D)}$	Port P3 input hold time		0			ns
$t_{h(E-P4D)}$	Port P4 input hold time		0			ns
$t_{h(E-P5D)}$	Port P5 input hold time		0			ns
$t_{h(E-P6D)}$	Port P6 input hold time		0			ns
$t_{h(E-P7D)}$	Port P7 input hold time		0			ns
$t_{h(E-P8D)}$	Port P8 input hold time		0			ns

Memory expansion mode and microprocessor mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{SU(P1D-E)}$	Port P1 input setup time		60			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		60			ns
$t_{SU(RDY-\phi_1)}$	$\overline{RDY}$ input setup time		70			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(\phi_1-RDY)}$	$\overline{RDY}$ input hold time		0			ns

Timer B input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		500			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		250			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		250			ns

Timer B input (Pulse period measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{C(TB)}$	TBi_{IN} input cycle time		1000			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		500			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		500			ns

Timer B input (Pulse width measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
$t_{C(TB)}$	TBi_{IN} input cycle time		1000			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		500			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		500			ns

A-D trigger input

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
$t_{C(AD)}$	$\overline{AD_{TRG}}$ input cycle time (minimum allowable trigger)		2000			ns
$t_{W(ADL)}$	$\overline{AD_{TRG}}$ input low-level pulse width		250			ns

Serial I/O

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max.	
$t_{C(CK)}$	CLKi input cycle time		500			ns
$t_{W(CKH)}$	CLKi input high-level pulse width		250			ns
$t_{W(CKL)}$	CLKi input low-level pulse width		250			ns
$t_{d(C-Q)}$	TxDi output delay time				150	ns
$t_{h(C-Q)}$	TxDi hold time		30			ns
$t_{su(D-C)}$	RxDi input setup time		60			ns
$t_{h(C-D)}$	RxDi input hold time		90			ns

External interrupt $\overline{INTi}$ input

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{W(INH)}$	$\overline{INTi}$ input high-level pulse width		250			ns
$t_{W(INL)}$	$\overline{INTi}$ input low-level pulse width		250			ns

SWITCHING CHARACTERISTICS (V_{CC}=5V±5%, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{d(E-P0Q)}$	Port P0 data output delay time	Figure 2			200	ns
$t_{d(E-P1Q)}$	Port P1 data output delay time				200	ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				200	ns
$t_{d(E-P3Q)}$	Port P3 data output delay time				200	ns
$t_{d(E-P4Q)}$	Port P4 data output delay time				200	ns
$t_{d(E-P5Q)}$	Port P5 data output delay time				200	ns
$t_{d(E-P6Q)}$	Port P6 data output delay time				200	ns
$t_{d(E-P7Q)}$	Port P7 data output delay time				200	ns
$t_{d(E-P8Q)}$	Port P8 data output delay time				200	ns

Memory expansion mode and microprocessor mode (when wait bit = "1")

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 2	100			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				120	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				5	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		100			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				120	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				5	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		100			ns
$t_{d(E-HLDA)}$	$\overline{HLDA}$ output delay time				100	ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		100			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		100			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		100			ns
$t_{d(E-\phi_1)}$	ϕ_1 output delay time		0		30	ns
$t_{h(E-P0A)}$	Port P0 address hold time		50			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		50			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		50			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		50			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		50			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		50			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		220			ns

TIMING REQUIREMENTS (V_{CC}=5V±5%, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

External clock input

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
t_C	External clock input cycle time		62			ns
$t_{W(H)}$	External clock input high-level pulse width		25			ns
$t_{W(L)}$	External clock input low-level pulse width		25			ns
t_r	External clock rise time				10	ns
t_f	External clock fall time				10	ns

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{su(P0D-E)}$	Port P0 input setup time		100			ns
$t_{su(P1D-E)}$	Port P1 input setup time		100			ns
$t_{su(P2D-E)}$	Port P2 input setup time		100			ns
$t_{su(P3D-E)}$	Port P3 input setup time		100			ns
$t_{su(P4D-E)}$	Port P4 input setup time		100			ns
$t_{su(P5D-E)}$	Port P5 input setup time		100			ns
$t_{su(P6D-E)}$	Port P6 input setup time		100			ns
$t_{su(P7D-E)}$	Port P7 input setup time		100			ns
$t_{su(P8D-E)}$	Port P8 input setup time		100			ns
$t_{h(E-P0D)}$	Port P0 input hold time		0			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(E-P3D)}$	Port P3 input hold time		0			ns
$t_{h(E-P4D)}$	Port P4 input hold time		0			ns
$t_{h(E-P5D)}$	Port P5 input hold time		0			ns
$t_{h(E-P6D)}$	Port P6 input hold time		0			ns
$t_{h(E-P7D)}$	Port P7 input hold time		0			ns
$t_{h(E-P8D)}$	Port P8 input hold time		0			ns

Memory expansion mode and microprocessor mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{su(P1D-E)}$	Port P1 input setup time		45			ns
$t_{su(P2D-E)}$	Port P2 input setup time		45			ns
$t_{su(RDY-\phi_1)}$	$\overline{RDY}$ input setup time		60			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(\phi_1-RDY)}$	$\overline{RDY}$ input hold time		0			ns

Timer B input (Cont input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		250			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		125			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		125			ns

Timer B input (Pulse period measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		500			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		250			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		250			ns

Timer B input (Pulse width measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		500			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		250			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		250			ns

A-D trigger input

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{C(AD)}$	$\overline{AD_{TRG}}$ input cycle time (minimum allowable trigger)		1000			ns
$t_{W(ADL)}$	$\overline{AD_{TRG}}$ input low-level pulse width		125			ns

Serial I/O

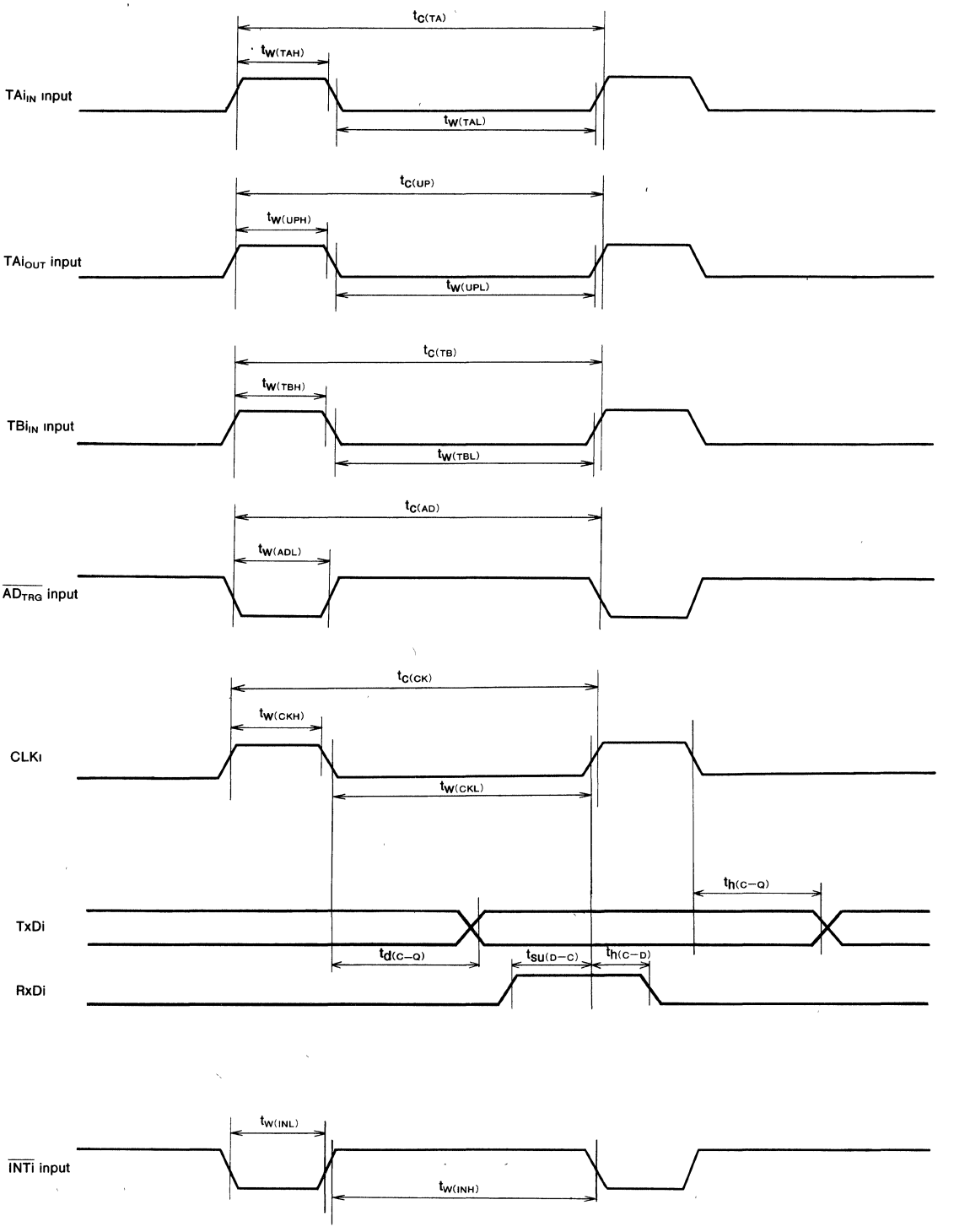
Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{C(CK)}$	CLKi input cycle time		250			ns
$t_{W(CKH)}$	CLKi input high-level pulse width		125			ns
$t_{W(CKL)}$	CLKi input low-level pulse width		125			ns
$t_{d(C-Q)}$	TxDi output delay time				90	ns
$t_{h(C-Q)}$	TxDi hold time		30			ns
$t_{su(D-C)}$	RxDi input setup time		30			ns
$t_{h(C-D)}$	RxDi input hold time		90			ns

External interrupt $\overline{INTi}$ input

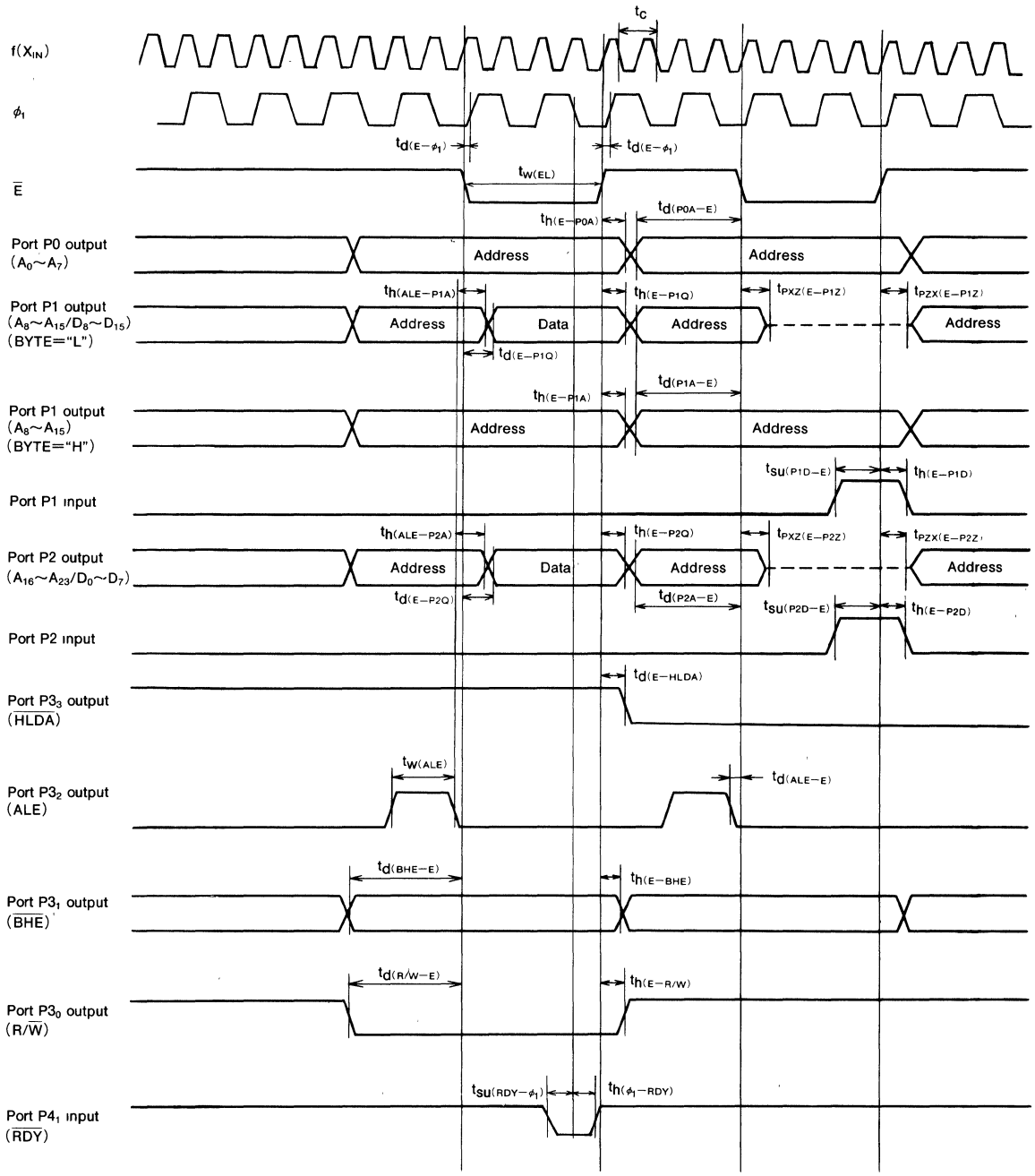
Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{W(INH)}$	$\overline{INTi}$ input high-level pulse width		250			ns
$t_{W(INL)}$	$\overline{INTi}$ input low-level pulse width		250			ns

Memory expansion mode and microprocessor mode (when wait bit = "0", and external memory area is accessed)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 2	155			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				80	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				5	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		155			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				80	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				5	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		155			ns
$t_{d(E-HLDA)}$	$\overline{HLDA}$ output delay time				50	ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		165			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		155			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		155			ns
$t_{d(E-\phi_1)}$	ϕ_1 output delay time		0		20	ns
$t_{h(E-P0A)}$	Port P0 address hold time		25			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		25			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		25			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		25			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		25			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		25			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		220			ns

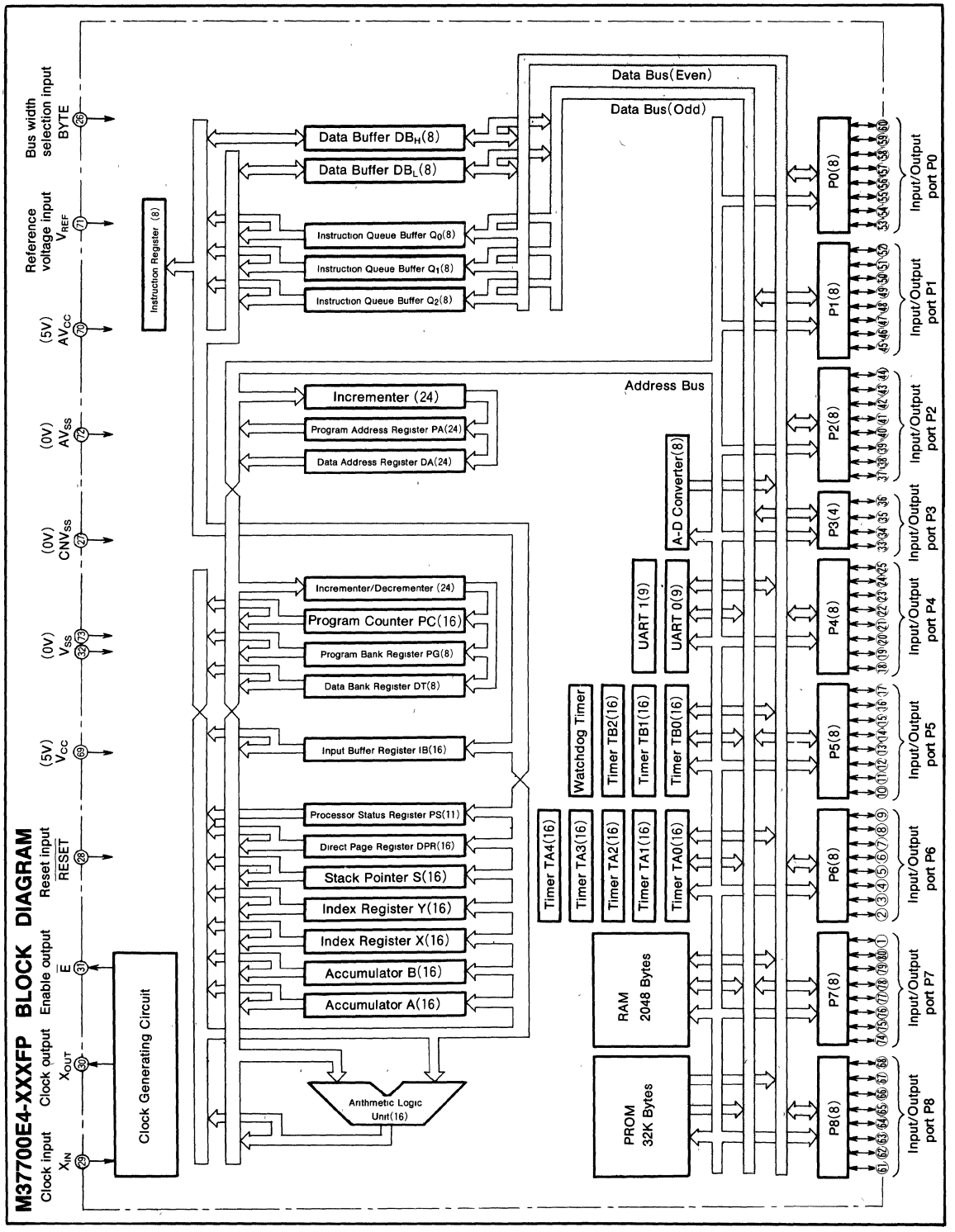

$t_{C(TA)}$
$t_{W(TAH)}$
TAi_{IN} input
$t_{W(TAL)}$
$t_{C(UP)}$
$t_{W(UPH)}$
TAi_{OUT} input
$t_{W(UPL)}$
$t_{C(TB)}$
$t_{W(TBH)}$
TBi_{IN} input
$t_{W(TBL)}$
$t_{C(AD)}$
$t_{W(ADL)}$
$\overline{AD_{TRG}}$ input
$t_{C(CK)}$
$t_{W(CKH)}$
CLKi
$t_{W(CKL)}$
$t_{h(C-Q)}$
TxDi
$t_{d(C-Q)}$
$t_{su(D-C)}$
$t_{h(C-D)}$
RxDi
$t_{W(INL)}$
$\overline{INTi}$ input
$t_{W(INH)}$

Memory expansion mode and microprocessor mode (when wait bit="0", and external memory area is accessed)

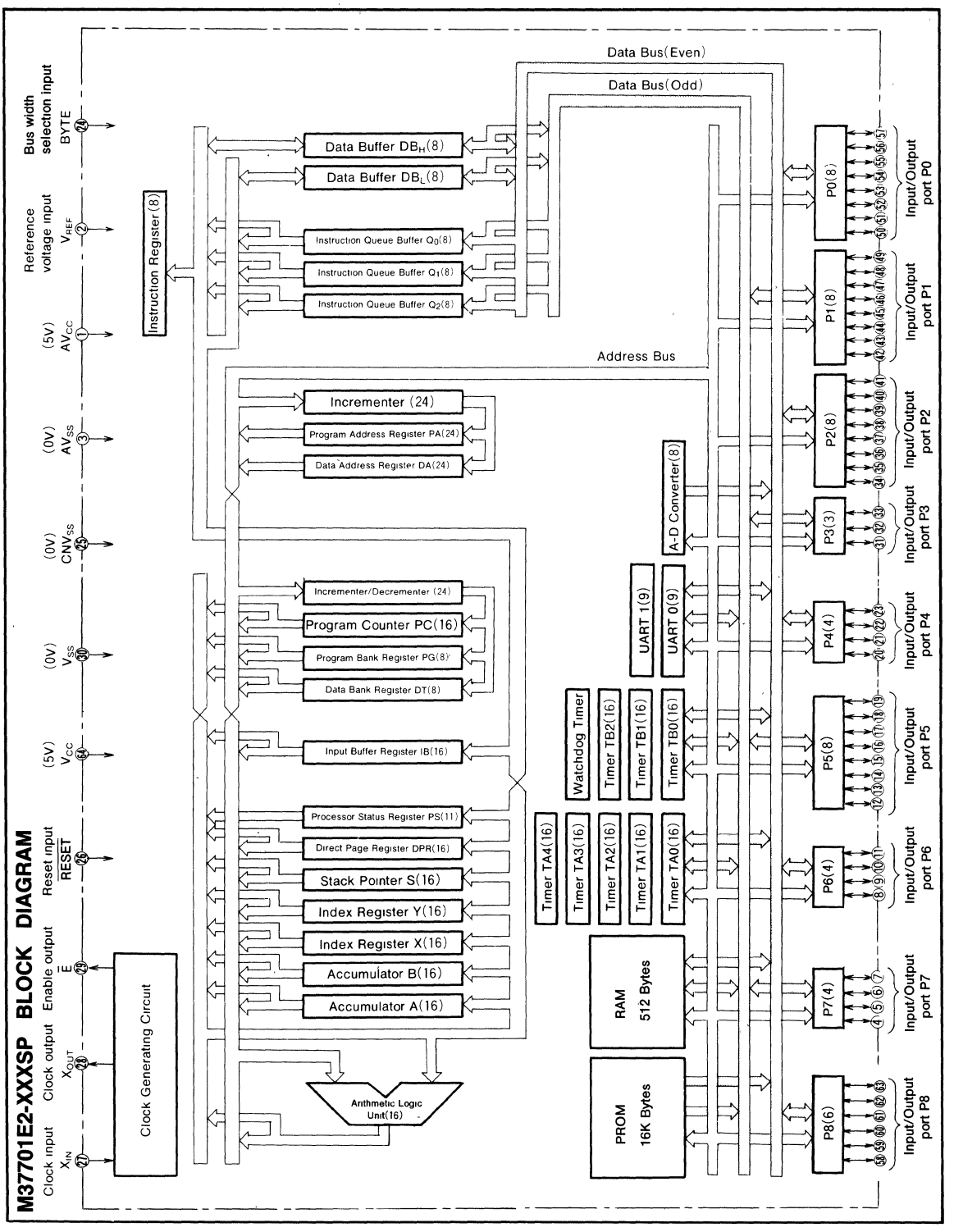


Test conditions

- V_{CC}= 5 V± 5 %
- Output timing voltage : V_{OL}=0.8V, V_{OH}=2.0V
- Ports P1,P2 input : V_{IL}=0.8V, V_{IH}=2.5V
- Port $P4_1$ input : V_{IL}=1.0V, V_{IH}=4.0V



MITSUBISHI ELECTRIC



MITSUBISHI ELECTRIC

PIN DESCRIPTION

Pin	Name	Input/Output	Functions
V_{CC}, V_{SS}	Power supply		Supply 5 V± 5 % to V_{CC} and 0 V to V_{SS}.
CNV_{SS}	CNV_{SS} input	Input	This pin controls the processor mode. Connect to V_{SS} for single-chip mode.
$\overline{RESET}$	Reset input	Input	To enter the reset state, this pin must be kept at a "L" condition should be maintained for the required time.
X_{IN}	Clock input	Input	These are I/O pins of internal clock generating circuit. Connect a ceramic or quartz crystal resonator between X_{IN} and X_{OUT}. When an external clock is used, the clock source should be connected to the X_{IN} pin and the X_{OUT} pin should be left open.
X_{OUT}	Clock output	Output	
$\overline{E}$	Enable output	Output	Data or instruction read and data write are performed when output from this pin is "L".
BYTE	Bus width selection input	Input	In memory expansion mode or microprocessor mode, this pin determines whether the external data bus is 8-bit width or 16-bit width The width is 16 bits when "L" signal inputs and 8 bits when "H" signal inputs.
AV_{CC}, AV_{SS}	Analog supply input		Power supply for the A-D converter Connect AV_{CC} to V_{CC} and AV_{SS} to V_{SS} externally
V_{REF}	Reference voltage input	Input	This is reference voltage input pin for the A-D converter.
$P0_0$~$P0_7$	I/O port P0	I/O	In single-chip mode, port P0 becomes an 8-bit I/O port. An I/O directional register is available so that each pin can be programmed for input or output. These ports are in input mode when reset . Address(A_7~A_0)is output in memory expansion mode or microprocessor mode.
$P1_0$~$P1_7$	I/O port P1	I/O	In single-chip mode, these pins have the same functions as port P0. When the BYTE pin is set to "L" in memory expansion mode or microprocessor mode and external data bus is 16-bit width, high-order data (D_{15}~D_8)is input or output when $\overline{E}$ output is "L" and an address (A_{15}~A_8) is output when $\overline{E}$ output is "H" If the BYTE pin is "H" that is an external data bus is 8-bit width, only address(A_{15}~A_8)is output.
$P2_0$~$P2_7$	I/O port P2	I/O	In single-chip mode, these pins have the same functions as port P0. In memory expansion mode or microprocessor mode low-order data(D_7~D_0) is input or output when $\overline{E}$ output is "L" and an address(A_{23}~A_{16})is output when $\overline{E}$ output is "H"
$P3_0$~$P3_2$	I/O port P3	I/O	In single-chip mode , these pins have the same functions as port P0. In memory expansion mode or microprocessor mode, R/$\overline{W}$, $\overline{BHE}$, and ALE signals are output.
$P4_0$~$P4_2$, $P4_7$	I/O port P4	I/O	In single-chip mode, these pins have the same functions as port P0. In memory expansion mode or microprocessor mode, $P4_0$ and $P4_1$ become $\overline{HOLD}$ and $\overline{RDY}$ input pin respectively Functions of other pins are the same as in single-chip mode Port $P4_2$ can be programed for ϕ_1 output pin divided the clock to X_{IN} pin by 2.
$P5_0$~$P5_7$	I/O port P5	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as I/O pins for timer A0, timer A1, timer A2, and timer A3
$P6_2$~$P6_5$	I/O port P6	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as external interrupt input $\overline{INT_0}$, $\overline{INT_1}$, and $\overline{INT_2}$ pins, and input pins for timer B0
$P7_0$~$P7_2$, $P7_7$	I/O port P7	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as analog input AN_0~AN_2 and AN_7 input pins. $P7_7$ also has an A-D conversion trigger input function
$P8_0$~$P8_3$, $P8_6$, $P8_7$	I/O port P8	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as R_XD, T_XD, CLK, $\overline{CTS}$/$\overline{RTS}$ pins for UART 0, and as RxD, TxD pins for UART 1.

EPROM MODE

The M37701E2-XXXSP features an EPROM mode in addition to its normal modes. When the $\overline{RESET}$ signal level is "L", the chip automatically enters the EPROM mode. Table 1 list the correspondence between pins and Fig. 1 gives the pin connections in the EPROM mode.

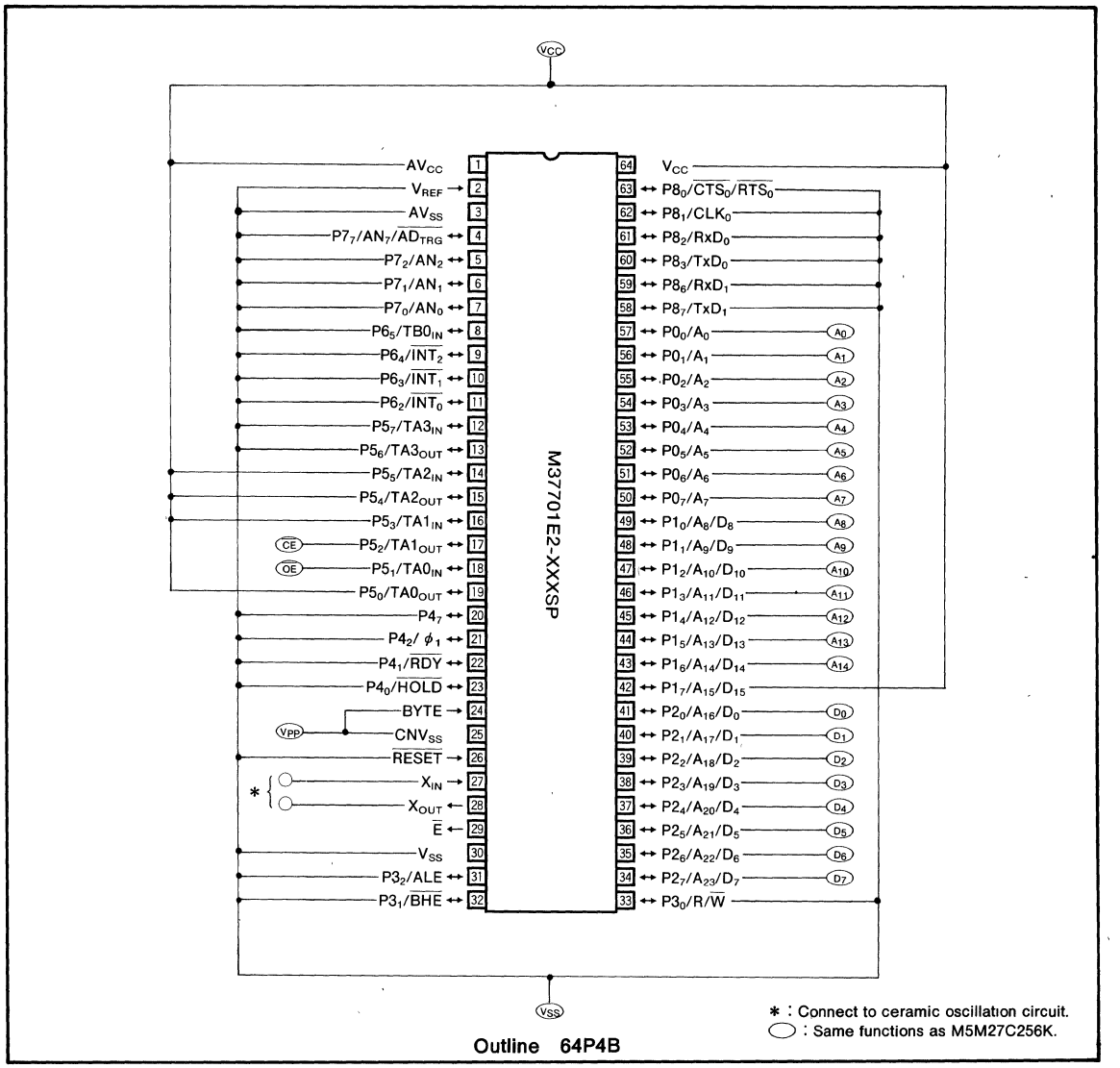
When in the EPROM mode, ports P0, P1, P2, $P5_1$, $P5_2$, CNV_{SS} and BYTE are used for the EPROM (equivalent to the M5M27C256K). When in this mode, the built-in PROM can be written to or read from using these pins in the same way as with the M5M27C256K.

This chip does not have Device Identifer Mode, so that set the corresponding program algorithm. The program area should specify address 4000_{16}~$7FFF_{16}$ for the M37701E2-XXXSP.

Connect the clock which is either ceramic resonator or external clock to X_{IN} pin and X_{OUT} pin.

Table 1 Pin function in EPROM programming mode

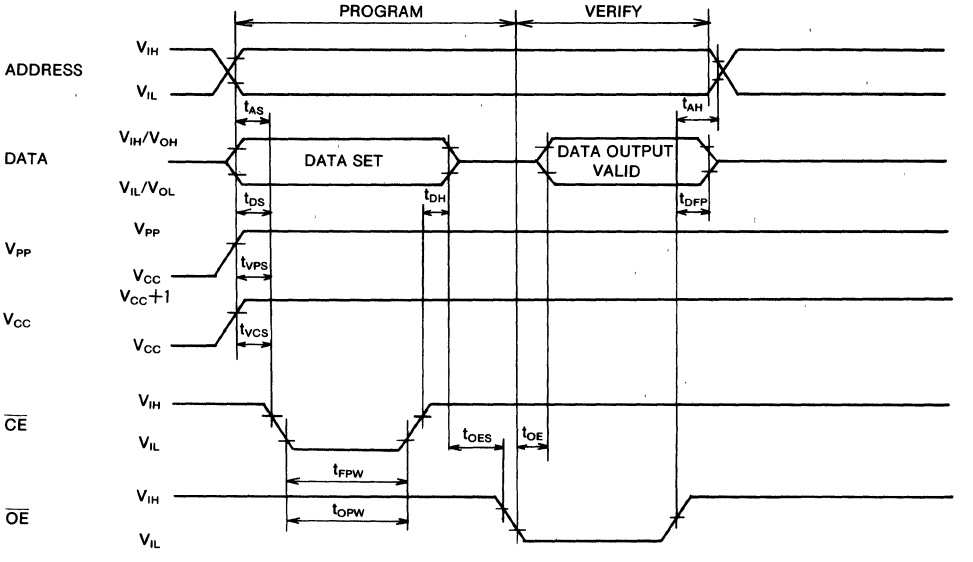
	M37701E2-XXXSP	M5M27C256K
V_{CC}	V_{CC}	V_{CC}
V_{PP}	CNV_{SS}, BYTE	V_{PP}
V_{SS}	V_{SS}	V_{SS}
Address input	Ports P0, $P1_0$~$P1_6$	A_0~A_{14}
Data I/O	Port P2	D_0~D_7
$\overline{CE}$	$P5_2$	$\overline{CE}$
$\overline{OE}$	$P5_1$	$\overline{OE}$



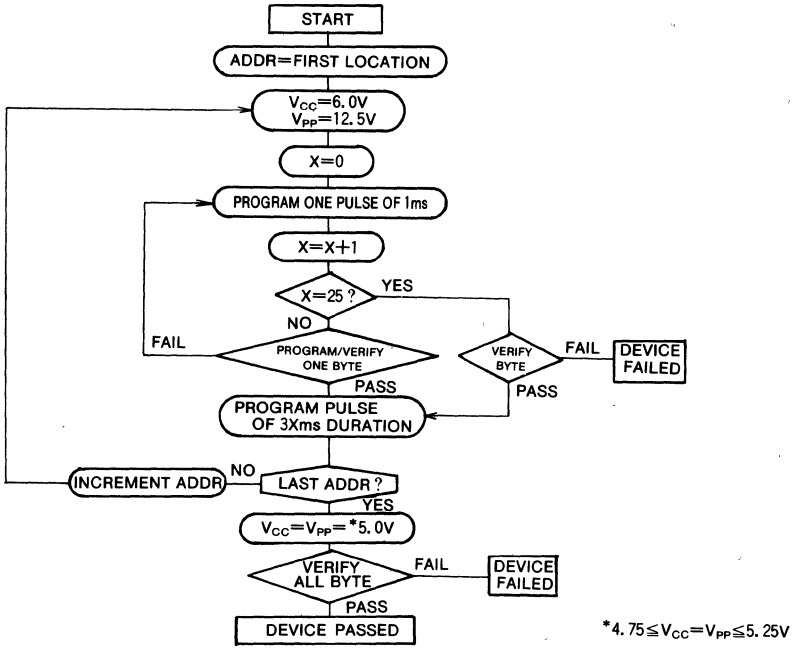
Outline 64P4B

Fig. 1 Pin connection in EPROM programming mode

AC waveforms



Fast programming algorithm flow chart



*4.75≦$V_{CC}=V_{PP}$≦5.25V

MITSUBISHI ELECTRIC

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Supply voltage		−0.3∼7	V
AV_{CC}	Analog supply voltage		−0.3∼7	V
V_I	Input voltage $\overline{RESET}$, CNV_{SS}, BYTE		−0.3∼12	V
V_I	Input voltage $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_7$, $P6_2$∼$P6_5$, $P7_0$∼$P7_2$, $P7_7$, $P8_0$∼$P8_3$, $P8_6$, $P8_7$, V_{REF}, X_{IN}		−0.3∼V_{CC}+0.3	V
V_O	Output voltage $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_7$, $P6_2$∼$P6_5$, $P7_0$∼$P7_2$, $P7_7$, $P8_0$∼$P8_3$, $P8_6$, $P8_7$, X_{OUT}, $\overline{E}$		−0.3∼V_{CC}+0.3	V
Pd	Power dissipation	T_a=25℃	1000	mW
Topr	Operating temperature		−10∼70	℃
Tstg	Storage temperature		−40∼125	℃

Note 1. Input voltage for CNV_{SS} and BYTE pins is 13V in writing to PROM.

RECOMMENDED OPERATING CONDITIONS (V_{CC}=5V±5%, T_a=−10∼70℃, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min	Typ	Max	
V_{CC}	Supply voltage		4.75	5.0	5.25	V
AV_{CC}	Analog supply voltage			V_{CC}		V
V_{SS}	Supply voltage			0		V
AV_{SS}	Analog supply voltage			0		V
V_{IH}	High-level input voltage $P0_0$∼$P0_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_7$, $P6_2$∼$P6_5$, $P7_0$∼$P7_2$, $P7_7$, $P8_0$∼$P8_3$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE		0.8V_{CC}		V_{CC}	V
V_{IH}	High-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in single-chip mode)		0.8V_{CC}		V_{CC}	V
V_{IH}	High-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in memory expansion mode and microprocessor mode)		0.5V_{CC}		V_{CC}	V
V_{IL}	Low-level input voltage $P0_0$∼$P0_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_7$, $P6_2$∼$P6_5$, $P7_0$∼$P7_2$, $P7_7$, $P8_0$∼$P8_3$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE		0		0.2V_{CC}	V
V_{IL}	Low-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in single-chip mode)		0		0.2V_{CC}	V
V_{IL}	Low-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in memory expansion mode and microprocessor mode)		0		0.16V_{CC}	V
$I_{OH(peak)}$	High-level peak output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_7$, $P6_2$∼$P6_5$, $P7_0$∼$P7_2$, $P7_7$, $P8_0$∼$P8_3$, $P8_6$, $P8_7$				−10	mA
$I_{OH(avg)}$	High-level average output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_7$, $P6_2$∼$P6_5$, $P7_0$∼$P7_2$, $P7_7$, $P8_0$∼$P8_3$, $P8_6$, $P8_7$				−5	mA
$I_{OL(peak)}$	Low-level peak output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_7$, $P6_2$∼$P6_6$, $P7_0$∼$P7_2$, $P7_7$, $P8_0$∼$P8_3$, $P8_6$, $P8_7$				10	mA
$I_{OL(avg)}$	Low-level average output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_7$, $P6_2$∼$P6_5$, $P7_0$∼$P7_2$, $P7_7$, $P8_0$∼$P8_3$, $P8_6$, $P8_7$				5	mA
$f(X_{IN})$	External clock frequency input	M37701E2-XXXSP			8	MHz
		M37701E2AXXXSP			16	

Note 1. Average output current is the average value of a100ms interval.
2. The sum of $I_{OL(peak)}$ for ports P0, P1, P2, P3 and P8 must be 80mA or less, the sum of $I_{OH(peak)}$ for ports P0, P1, P2, P3 and P8 must be 80mA or less, the sum of $I_{OL(peak)}$ for ports P4, P5, P6 and P7 must be 80mA or less, and the sum of $I_{OH(peak)}$ for ports P4, P5, P6 and P7 must be 80mA or less

TIMING REQUIREMENTS (V_{CC}=5V±5%, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

External clock input

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max	
t_C	External clock input cycle time		125			ns
$t_{W(H)}$	External clock input high-level pulse width		50			ns
$t_{W(L)}$	External clock input low-level pulse width		50			ns
t_r	External clock rise time				20	ns
t_f	External clock fall time				20	ns

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{SU(P0D-E)}$	Port P0 input setup time		200			ns
$t_{SU(P1D-E)}$	Port P1 input setup time		200			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		200			ns
$t_{SU(P3D-E)}$	Port P3 input setup time		200			ns
$t_{SU(P4D-E)}$	Port P4 input setup time		200			ns
$t_{SU(P5D-E)}$	Port P5 input setup time		200			ns
$t_{SU(P6D-E)}$	Port P6 input setup time		200			ns
$t_{SU(P7D-E)}$	Port P7 input setup time		200			ns
$t_{SU(P8D-E)}$	Port P8 input setup time		200			ns
$t_{h(E-P0D)}$	Port P0 input hold time		0			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(E-P3D)}$	Port P3 input hold time		0			ns
$t_{h(E-P4D)}$	Port P4 input hold time		0			ns
$t_{h(E-P5D)}$	Port P5 input hold time		0			ns
$t_{h(E-P6D)}$	Port P6 input hold time		0			ns
$t_{h(E-P7D)}$	Port P7 input hold time		0			ns
$t_{h(E-P8D)}$	Port P8 input hold time		0			ns

Memory expansion mode and microprocessor mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{SU(P1D-E)}$	Port P1 input setup time		60			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		60			ns
$t_{SU(RDY-\phi_1)}$	$\overline{RDY}$ input setup time		70			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(\phi 1-RDY)}$	$\overline{RDY}$ input hold time		0			ns

Timer B input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		500			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		250			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		250			ns

Timer B input (Pulse period measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{C(TB)}$	TBi_{IN} input cycle time		1000			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		500			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		500			ns

Timer B input (Pulse width measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{C(TB)}$	TBi_{IN} input cycle time		1000			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		500			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		500			ns

A-D trigger input

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{C(AD)}$	$\overline{AD_{TRG}}$ input cycle time (minimum allowable trigger)		2000			ns
$t_{W(ADL)}$	$\overline{AD_{TRG}}$ input low-level pulse width		250			ns

Serial I/O

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{C(CK)}$	CLKi input cycle time		500			ns
$t_{W(CKH)}$	CLKi input high-level pulse width		250			ns
$t_{W(CKL)}$	CLKi input low-level pulse width		250			ns
$t_{d(C-Q)}$	TxDi output delay time				150	ns
$t_{h(C-Q)}$	TxDi hold time		30			ns
$t_{SU(D-C)}$	RxDi input setup time		60			ns
$t_{h(C-D)}$	RxDi input hold time		90			ns

External interrupt $\overline{INTi}$ input

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max.	
$t_{W(INH)}$	$\overline{INTi}$ input high-level pulse width		250			ns
$t_{W(INL)}$	$\overline{INTi}$ input low-level pulse width		250			ns

Memory expansion mode and microprocessor mode (when wait bit = "0", and external memory area is accessed)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 2	350			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				120	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				40	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		350			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				120	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				40	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		350			ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		350			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		350			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		350			ns
$t_{d(E-\phi 1)}$	ϕ_1 output delay time		0		30	ns
$t_{h(E-P0A)}$	Port P0 address hold time		50			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		50			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		50			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		50			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		50			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		50			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		470			ns

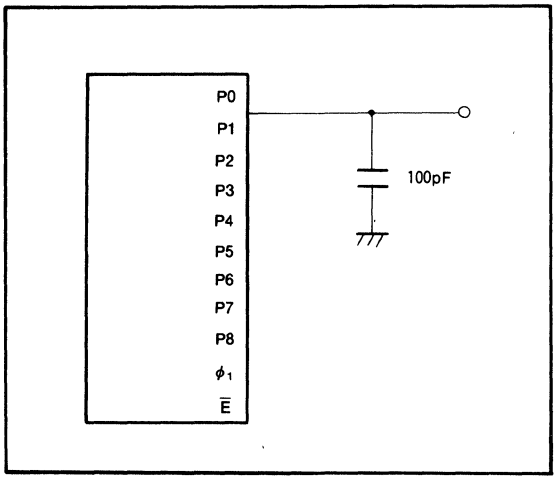


Fig. 2 Testing circuit for ports P0~P8, ϕ_1

TIMING REQUIREMENTS (V_{CC}=5V±5%, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

External clock input

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
t_C	External clock input cycle time		62			ns
$t_{W(H)}$	External clock input high-level pulse width		25			ns
$t_{W(L)}$	External clock input low-level pulse width		25			ns
t_r	External clock rise time				10	ns
t_f	External clock fall time				10	ns

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{SU(P0D-E)}$	Port P0 input setup time		100			ns
$t_{SU(P1D-E)}$	Port P1 input setup time		100			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		100			ns
$t_{SU(P3D-E)}$	Port P3 input setup time		100			ns
$t_{SU(P4D-E)}$	Port P4 input setup time		100			ns
$t_{SU(P5D-E)}$	Port P5 input setup time		100			ns
$t_{SU(P6D-E)}$	Port P6 input setup time		100			ns
$t_{SU(P7D-E)}$	Port P7 input setup time		100			ns
$t_{SU(P8D-E)}$	Port P8 input setup time		100			ns
$t_{h(E-P0D)}$	Port P0 input hold time		0			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(E-P3D)}$	Port P3 input hold time		0			ns
$t_{h(E-P4D)}$	Port P4 input hold time		0			ns
$t_{h(E-P5D)}$	Port P5 input hold time		0			ns
$t_{h(E-P6D)}$	Port P6 input hold time		0			ns
$t_{h(E-P7D)}$	Port P7 input hold time		0			ns
$t_{h(E-P8D)}$	Port P8 input hold time		0			ns

Memory expansion mode and microprocessor mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{SU(P1D-E)}$	Port P1 input setup time		45			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		45			ns
$t_{SU(RDY-\phi1)}$	$\overline{RDY}$ input setup time		60			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(\phi1-RDY)}$	$\overline{RDY}$ input hold time		0			ns

Timer B input (Cont input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max	
$t_{C(TB)}$	TBi_{IN} input cycle time		250			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		125			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		125			ns

Timer B input (Pulse period measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		500			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		250			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		250			ns

Timer B input (Pulse width measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		500			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		250			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		250			ns

A-D trigger input

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(AD)}$	$\overline{AD_{TRG}}$ input cycle time (minimum allowable trigger)		1000			ns
$t_{W(ADL)}$	$\overline{AD_{TRG}}$ input low-level pulse width		125			ns

Serial I/O

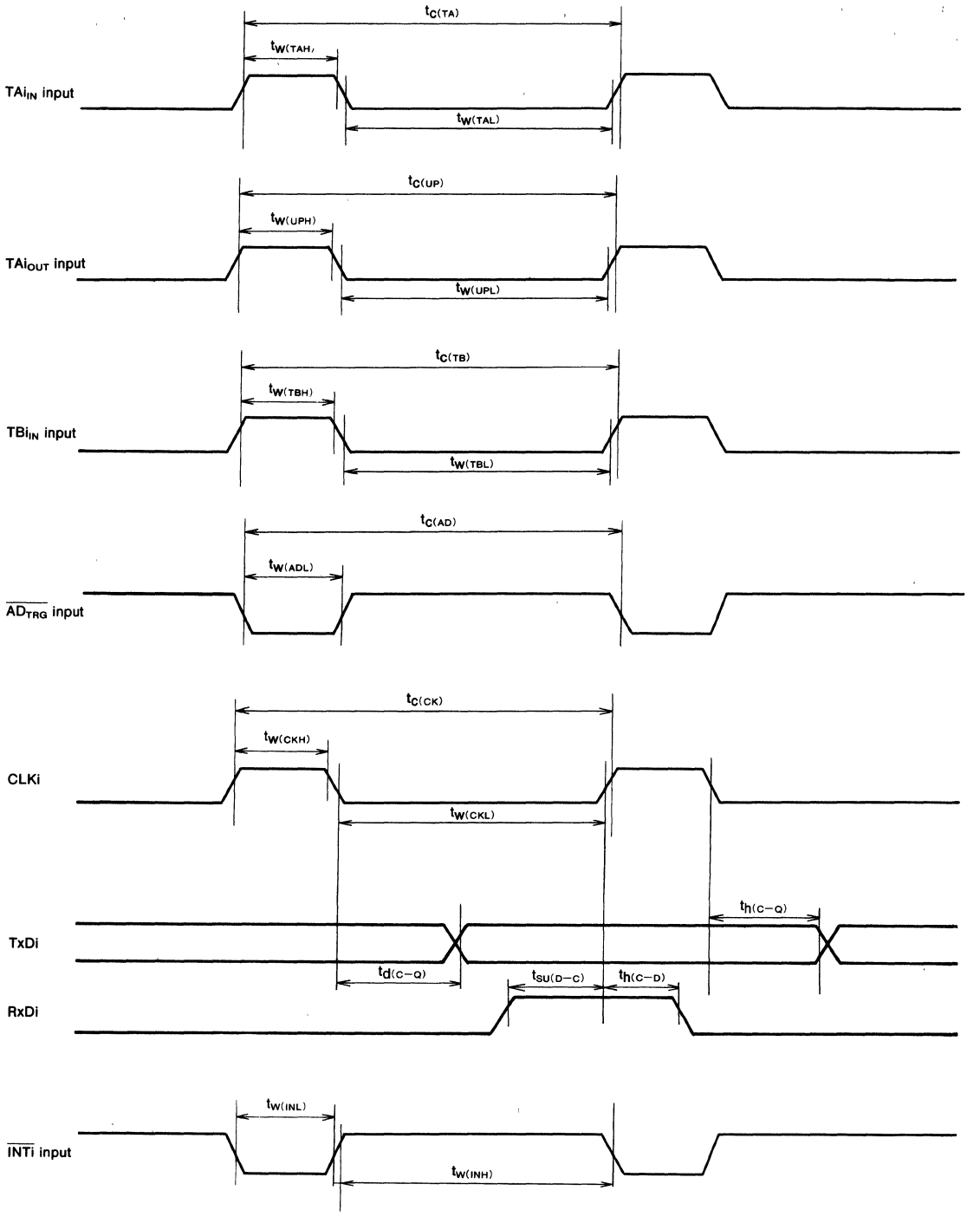
Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(CK)}$	CLKi input cycle time		250			ns
$t_{W(CKH)}$	CLKi input high-level pulse width		125			ns
$t_{W(CKL)}$	CLKi input low-level pulse width		125			ns
$t_{d(C-Q)}$	TxDi output delay time				90	ns
$t_{h(C-Q)}$	TxDi hold time		30			ns
$t_{SU(D-C)}$	RxDi input setup time		30			ns
$t_{h(C-D)}$	RxDi input hold time		90			ns

External interrupt $\overline{INTi}$ input

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{W(INH)}$	$\overline{INTi}$ input high-level pulse width		250			ns
$t_{W(INL)}$	$\overline{INTi}$ input low-level pulse width		250			ns

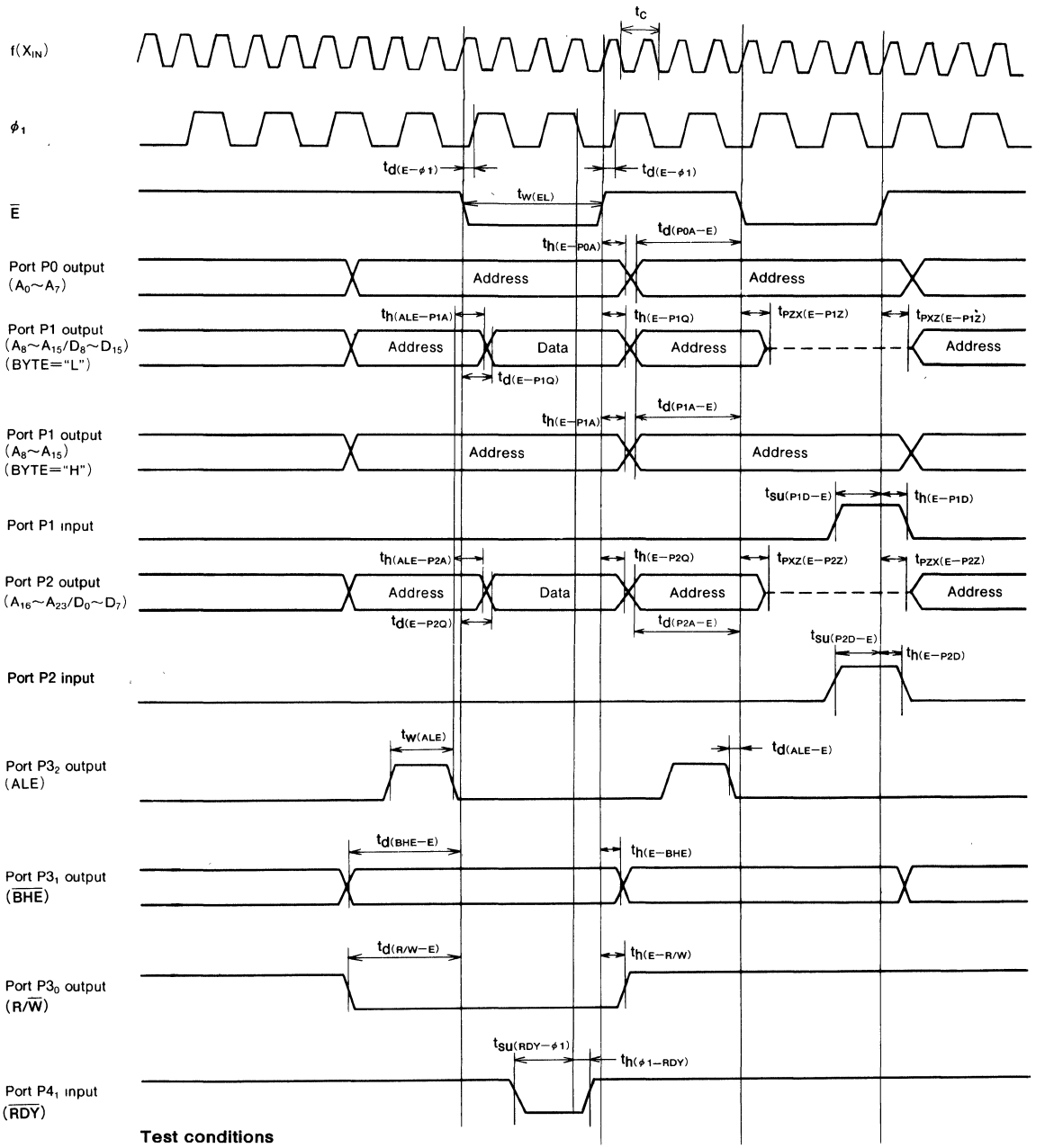
Memory expansion mode and microprocessor mode (when wait bit = "0", and external memory area is accessed)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 2	155			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				80	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				40	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		155			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				80	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				40	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		155			ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		165			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		155			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		155			ns
$t_{d(E-\phi 1)}$	ϕ_1 output delay time		0		20	ns
$t_{h(E-P0A)}$	Port P0 address hold time		25			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		25			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		25			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		25			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		25			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		25			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		220			ns



MITSUBISHI ELECTRIC

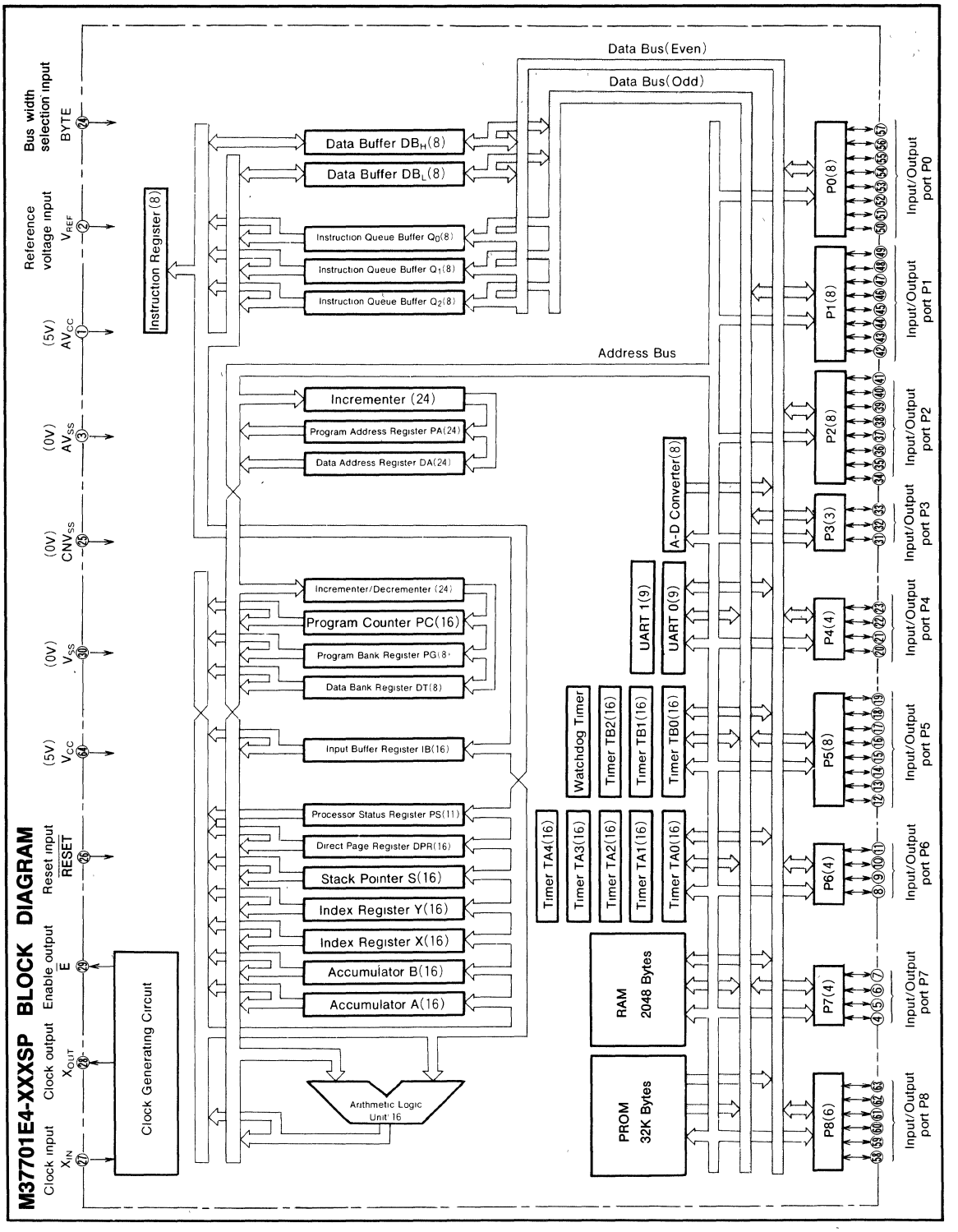
Memory expansion mode and microprocessor mode (when wait bit="0", and external memory area is accessed)



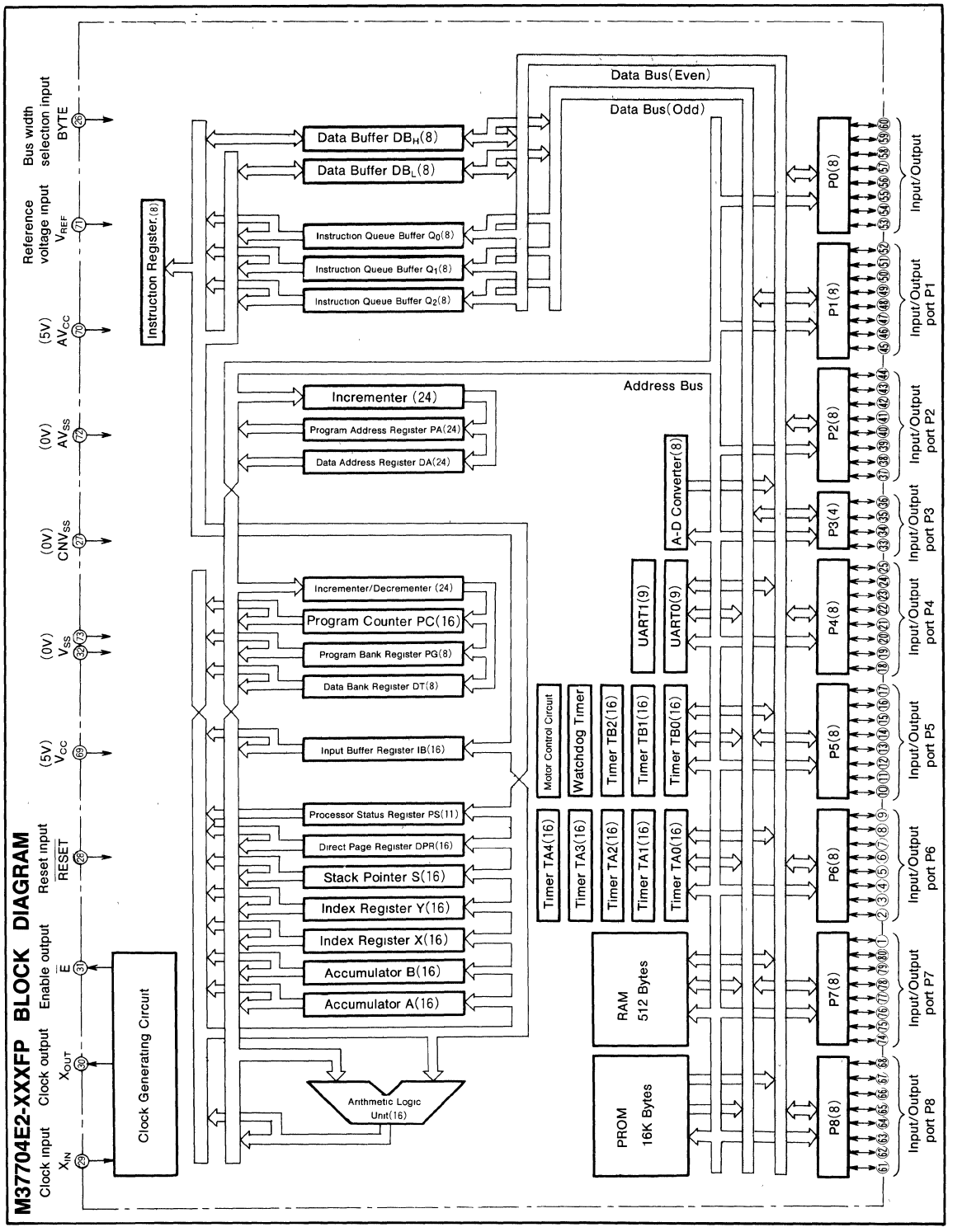
Test conditions

- V_{CC} = 5 V ± 5 %
- Output timing voltage : V_{OL} = 0.8V, V_{OH} = 2.0V
- Ports P1,P2 input : V_{IL} = 0.8V, V_{IH} = 2.5V
- Port $P4_1$ input : V_{IL} = 1.0V, V_{IH} = 4.0V

PROM VERSION of M37701M4-XXXSP,M37701M4AXXXSP



MITSUBISHI ELECTRIC



MITSUBISHI ELECTRIC

PIN DESCRIPTION (NORMAL MODE)

Pin	Name	Input/Output	Functions
V_{CC}, V_{SS}	Power supply		Supply 5 V± 5 % to V_{CC} and 0 V to V_{SS}
CNV_{SS}	CNV_{SS} input	Input	This pin controls the processor mode Connect to V_{SS} for single-chip mode
$\overline{RESET}$	Reset input	Input	To enter the reset state, this pin must be kept at a "L" condition which should be maintained for the required time.
X_{IN}	Clock input	Input	These are I/O pins of internal clock generating circuit Connect a ceramic or quartz crystal resonator between X_{IN} and X_{OUT} When an external clock is used, the clock source should be connected to the X_{IN} pin and the X_{OUT} pin should be left open
X_{OUT}	Clock output	Output	
$\overline{E}$	Enable output	Output	Data or instruction read and data write are performed when output from this pin is "L"
BYTE	Bus width selection input	Input	In memory expansion mode or microprocessor mode, this pin determines whether the external data bus is 8-bit width or 16-bit width. The width is 16 bits when "L" signal inputs and 8 bits when "H" signal inputs.
AV_{CC}, AV_{SS}	Analog supply input		Power supply for the A-D converter Connect AV_{CC} to V_{CC} and AV_{SS} to V_{SS} externally.
V_{REF}	Reference voltage input	Input	This is reference voltage input pin for the A-D converter
$P0_0$~$P0_7$	I/O port P0	I/O	In single-chip mode, port P0 becomes an 8-bit I/O port An I/O directional register is available so that each pin can be programmed for input or output These ports are in input mode when reset . Address(A_7~A_0)is output in memory expansion mode or microprocessor mode.
$P1_0$~$P1_7$	I/O port P1	I/O	In single-chip mode, these pins have the same functions as port P0. When the BYTE pin is set to "L" in memory expansion mode or microprocessor mode and external data bus is 16-bit width, high-order data (D_{15}~D_8)is input or output when $\overline{E}$ output is "L" and an address (A_{15}~A_8) is output when $\overline{E}$ output is "H" If the BYTE pin is "H" that is an external data bus is 8-bit width, only address(A_{15}~A_8)is output
$P2_0$~$P2_7$	I/O port P2	I/O	In single-chip mode, these pins have the same functions as port P0. In memory expansion mode or microprocessor mode low-order data(D_7~D_0) is input or output when $\overline{E}$ output is "L" and an address(A_{23}~A_{16})is output when $\overline{E}$ output is "H"
$P3_0$~$P3_3$	I/O port P3	I/O	In single-chip mode , these pins have the same functions as port P0 In memory expansion mode or microprocessor mode, R/$\overline{W}$, $\overline{BHE}$, ALE, and $\overline{HLDA}$ signals are output
$P4_0$~$P4_7$	I/O port P4	I/O	In single-chip mode, these pins have the same functions as port P0 In memory expansion mode or microprocessor mode, $P4_0$ and $P4_1$ become $\overline{HOLD}$ and $\overline{RDY}$ input pin respectively Functions of other pins are the same as in single-chip mode In single-chip mode or memory expansion mode, port $P4_2$ can be programed for ϕ_1 output pin divided the clock to X_{IN} pin by 2 In microprocessor mode, $P4_2$ always has the fanction as ϕ_1 output pin.
$P5_0$~$P5_7$	I/O port P5	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as I/O pins for timer A0, timer A1, timer A2, and timer A3 $P5_0$ to $P5_6$ also have the function as motor control output pins
$P6_0$~$P6_7$	I/O port P6	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as I/O pins for timer A4, external interrupt input $\overline{INT_0}$, $\overline{INT_1}$, and $\overline{INT_2}$ pins, and input pins for timer B0, timer B1, and timer B2 $P6_0$ and $P6_2$ also have the function as motor control output pins.
$P7_0$~$P7_7$	I/O port P7	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as analog input AN_0~AN_7 input pins. $P7_7$ also has an A-D conversion trigger input function
$P8_0$~$P8_7$	I/O port P8	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as R_XD, T_XD, CLK, $\overline{CTS}$/$\overline{RTS}$ pins for UART 0 and UART 1

M37704E2-XXXFP,M37704E2AXXXFP
M37704E2FS,M37704E2AFS

PROM VERSION of M37704M2-XXXFP,M37704M2AXXXFP

EPROM MODE

The M37704E2-XXXFP features an EPROM mode in addition to its normal modes. When the $\overline{RESET}$ signal level is "L", the chip automatically enters the EPROM mode. Table 1 list the correspondence between pins and Fig. 1 gives the pin connections in the EPROM mode.

When in the EPROM mode, ports P0, P1, P2, $P5_1$, $P5_2$, CNV_{SS} and BYTE are used for the EPROM (equivalent to the M5M27C256K). When in this mode, the built-in PROM can be written to or read from using these pins in the same way as with the M5M27C256K.

This chip does not have Device Identifer Mode, so that set the corresponding program algorithm. The program area should specify address 4000_{16}~$7FFF_{16}$ for the M37704E2-XXXFP.

Set the clock which is either ceramic resonator or external clock to X_{IN} pin and X_{OUT} pin.

Table 1 Pin function in EPROM programming mode

	M37704E2-XXXFP	M5M27C256K
V_{CC}	V_{CC}	V_{CC}
V_{PP}	CNV_{SS}, BYTE	V_{PP}
V_{SS}	V_{SS}	V_{SS}
Address input	Ports P0, $P1_0$~$P1_6$	A_0~A_{14}
Data I/O	Port P2	D_0~D_7
$\overline{CE}$	$P5_2$	$\overline{CE}$
$\overline{OE}$	$P5_1$	$\overline{OE}$

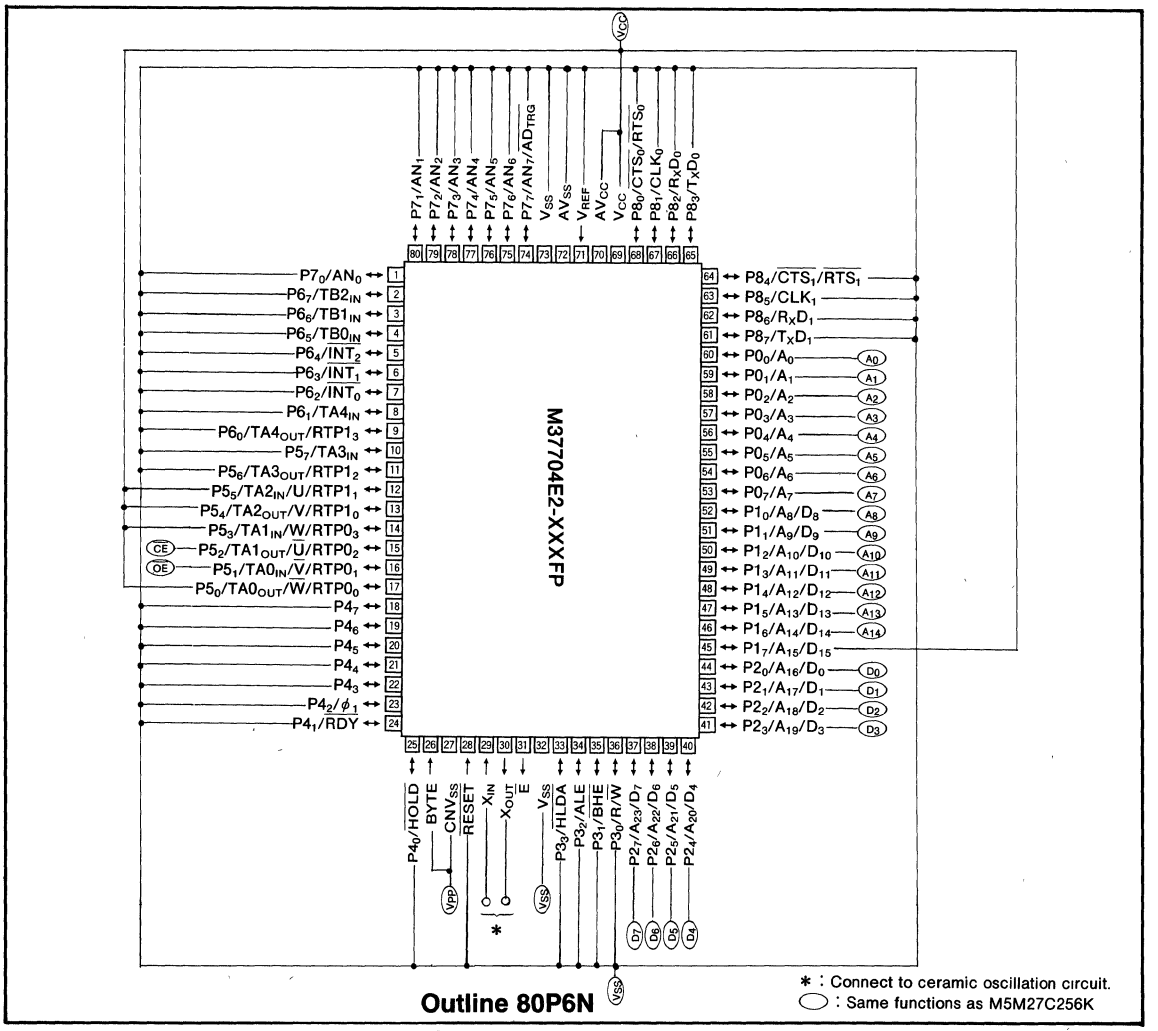
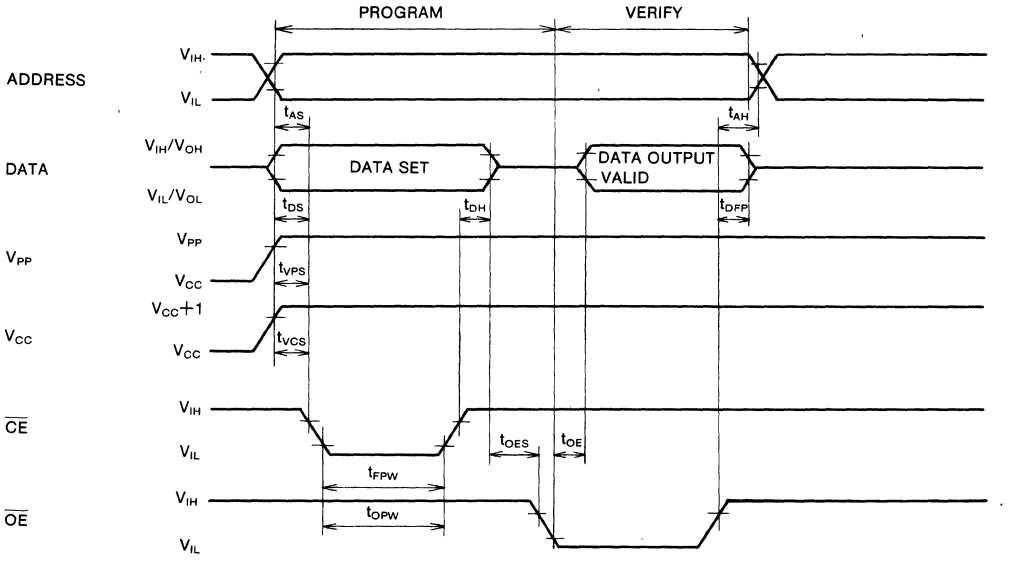
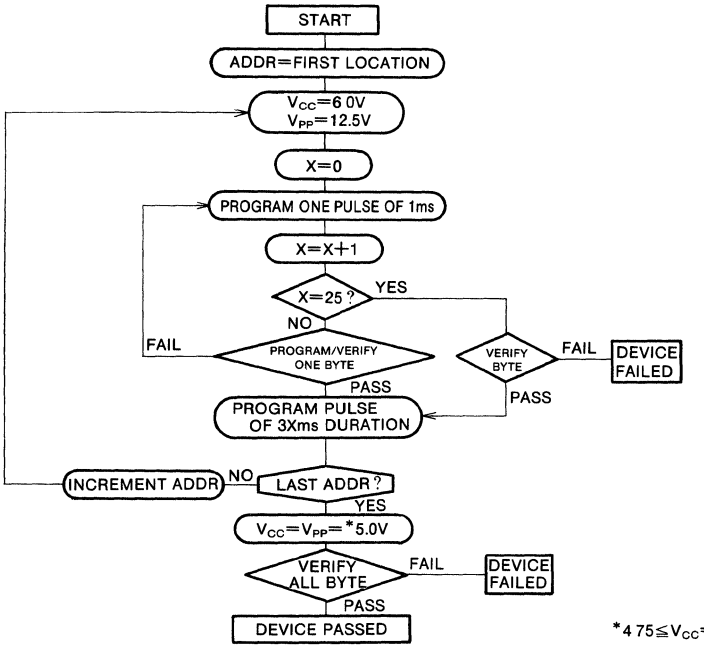


Fig. 1 Pin connection in EPROM programming mode

AC waveforms



Fast programming algorithm flow chart



*4 75≦V_{CC}=V_{PP}≦5 25V

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Supply voltage		−0.3∼7	V
AV_{CC}	Analog supply voltage		−0.3∼7	V
V_I	Input voltage $\overline{RESET}$, CNV_{SS}, BYTE		−0.3∼12 (Note 1)	V
V_I	Input voltage $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$, V_{REF}, X_{IN}		−0.3∼V_{CC}+0.3	V
V_O	Output voltage $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$, X_{OUT}, $\overline{E}$		−0.3∼V_{CC}+0.3	V
P_d	Power dissipation	T_a=25℃	300	mW
T_{opr}	Operating temperature		−10∼70	℃
T_{stg}	Storage temperature		−40∼125	℃

Note 1. Input voltage for CNV_{SS} and BYTE pins is 13V in writing to PROM

RECOMMENDED OPERATING CONDITIONS (V_{CC}=5V±5%, T_a=−10∼70℃, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min	Typ	Max	
V_{CC}	Supply voltage		4.75	5.0	5.25	V
AV_{CC}	Analog supply voltage			V_{CC}		V
V_{SS}	Supply voltage			0		V
AV_{SS}	Analog supply voltage			0		V
V_{IH}	High-level input voltage $P0_0$∼$P0_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE		0.8V_{CC}		V_{CC}	V
V_{IH}	High-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in single-chip mode)		0.8V_{CC}		V_{CC}	V
V_{IH}	High-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in memory expansion mode and microprocessor mode)		0.5V_{CC}		V_{CC}	V
V_{IL}	Low-level input voltage $P0_0$∼$P0_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE		0		0.2V_{CC}	V
V_{IL}	Low-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in single-chip mode)		0		0.2V_{CC}	V
V_{IL}	Low-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in memory expansion mode and microprocessor mode)		0		0.16V_{CC}	V
$I_{OH(peak)}$	High-level peak output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$				−10	mA
$I_{OH(avg)}$	High-level average output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_0$∼$P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$				−5	mA
$I_{OL(peak)}$	Low-level peak output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_6$, $P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$				10	mA
$I_{OL(peak)}$	Low-level peak output current $P5_0$∼$P5_5$				20	mA
$I_{OL(avg)}$	Low-level average output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_3$, $P4_0$∼$P4_7$, $P5_6$, $P5_7$, $P6_0$∼$P6_7$, $P7_0$∼$P7_7$, $P8_0$∼$P8_7$				5	mA
$I_{OL(avg)}$	Low-level average output current $P5_0$∼$P5_5$				15	mA
$f(X_{IN})$	External clock frequency input	M37704E2-XXXFP, M37704E2FS			8	MHz
		M37704E2AXXXFP, M37704E2AFS			16	

Note 1. Average output current is the average value of a100ms interval
2. The sum of $I_{OL(peak)}$ for ports P0, P1, P2, P3, and P8 must be 80mA or less, the sum of $I_{OH(peak)}$ for ports P0, P1, P2, P3, and P8 must be 80mA or less, the sum of $I_{OL(peak)}$ for ports P4, P5, P6, and P7 must be 110mA or less, and the sum of $I_{OH(peak)}$ for ports P4, P5, P6, and P7 must be 80mA or less

12·31·2099

TIMING REQUIREMENTS (V_{CC}=5V±5%, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

External clock input

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
t_C	External clock input cycle time		125			ns
$t_{W(H)}$	External clock input high-level pulse width		50			ns
$t_{W(L)}$	External clock input low-level pulse width		50			ns
t_r	External clock rise time				20	ns
t_f	External clock fall time				20	ns

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{SU(P0D-E)}$	Port P0 input setup time		200			ns
$t_{SU(P1D-E)}$	Port P1 input setup time		200			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		200			ns
$t_{SU(P3D-E)}$	Port P3 input setup time		200			ns
$t_{SU(P4D-E)}$	Port P4 input setup time		200			ns
$t_{SU(P5D-E)}$	Port P5 input setup time		200			ns
$t_{SU(P6D-E)}$	Port P6 input setup time		200			ns
$t_{SU(P7D-E)}$	Port P7 input setup time		200			ns
$t_{SU(P8D-E)}$	Port P8 input setup time		200			ns
$t_{h(E-P0D)}$	Port P0 input hold time		0			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(E-P3D)}$	Port P3 input hold time		0			ns
$t_{h(E-P4D)}$	Port P4 input hold time		0			ns
$t_{h(E-P5D)}$	Port P5 input hold time		0			ns
$t_{h(E-P6D)}$	Port P6 input hold time		0			ns
$t_{h(E-P7D)}$	Port P7 input hold time		0			ns
$t_{h(E-P8D)}$	Port P8 input hold time		0			ns

Memory expansion mode and microprocessor mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{SU(P1D-E)}$	Port P1 input setup time		60			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		60			ns
$t_{SU(RDY-\phi_1)}$	$\overline{RDY}$ input setup time		70			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(\phi_1-RDY)}$	$\overline{RDY}$ input hold time		0			ns

Timer B input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time (one edge count)		250			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width (one edge count)		125			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width (one edge count)		125			ns
$t_{C(TB)}$	TBi_{IN} input cycle time (both edges count)		500			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width (both edges count)		250			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width (both edges count)		250			ns

Timer B input (Pulse period measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{C(TB)}$	TBi_{IN} input cycle time		1000			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		500			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		500			ns

Timer B input (Pulse width measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{C(TB)}$	TBi_{IN} input cycle time		1000			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		500			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		500			ns

A-D trigger input

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{C(AD)}$	$\overline{AD_{TRG}}$ input cycle time (minimum allowable trigger)		2000			ns
$t_{W(ADL)}$	$\overline{AD_{TRG}}$ input low-level pulse width		250			ns

Serial I/O

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{C(CK)}$	CLKi input cycle time		500			ns
$t_{W(CKH)}$	CLKi input high-level pulse width		250			ns
$t_{W(CKL)}$	CLKi input low-level pulse width		250			ns
$t_{d(C-Q)}$	TxDi output delay time				150	ns
$t_{h(C-Q)}$	TxDi hold time		30			ns
$t_{su(D-C)}$	RxDi input setup time		60			ns
$t_{h(C-D)}$	RxDi input hold time		90			ns

External interrupt $\overline{INTi}$ input

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{W(INH)}$	$\overline{INTi}$ input high-level pulse width		250			ns
$t_{W(INL)}$	$\overline{INTi}$ input low-level pulse width		250			ns

Memory expansion mode and microprocessor mode (when wait bit = "0", and external memory area is accessed)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max.	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 2	350			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				120	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				5	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		350			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				120	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				5	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		350			ns
$t_{d(E-HLDA)}$	$\overline{HLDA}$ output delay time				100	ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		350			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		350			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		350			ns
$t_{d(E-\phi_1)}$	ϕ_1 output delay time		0		30	ns
$t_{h(E-P0A)}$	Port P0 address hold time		50			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		50			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		50			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		50			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		50			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		50			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		220			ns

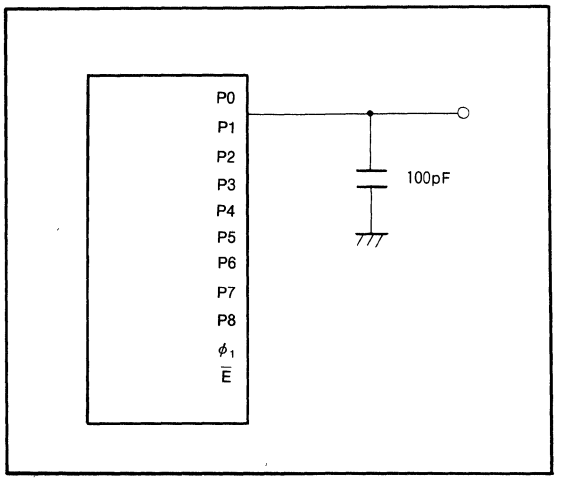


Fig. 2 Testing circuit for ports P0~P8, ϕ_1

TIMING REQUIREMENTS (V_{CC}=5V±5%, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

External clock input

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max	
t_C	External clock input cycle time		62			ns
$t_{W(H)}$	External clock input high-level pulse width		25			ns
$t_{W(L)}$	External clock input low-level pulse width		25			ns
t_r	External clock rise time				10	ns
t_f	External clock fall time				10	ns

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{SU(P0D-E)}$	Port P0 input setup time		100			ns
$t_{SU(P1D-E)}$	Port P1 input setup time		100			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		100			ns
$t_{SU(P3D-E)}$	Port P3 input setup time		100			ns
$t_{SU(P4D-E)}$	Port P4 input setup time		100			ns
$t_{SU(P5D-E)}$	Port P5 input setup time		100			ns
$t_{SU(P6D-E)}$	Port P6 input setup time		100			ns
$t_{SU(P7D-E)}$	Port P7 input setup time		100			ns
$t_{SU(P8D-E)}$	Port P8 input setup time		100			ns
$t_{h(E-P0D)}$	Port P0 input hold time		0			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(E-P3D)}$	Port P3 input hold time		0			ns
$t_{h(E-P4D)}$	Port P4 input hold time		0			ns
$t_{h(E-P5D)}$	Port P5 input hold time		0			ns
$t_{h(E-P6D)}$	Port P6 input hold time		0			ns
$t_{h(E-P7D)}$	Port P7 input hold time		0			ns
$t_{h(E-P8D)}$	Port P8 input hold time		0			ns

Memory expansion mode and microprocessor mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
$t_{SU(P1D-E)}$	Port P1 input setup time		45			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		45			ns
$t_{SU(RDY-\phi_1)}$	$\overline{RDY}$ input setup time		60			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(\phi_1-RDY)}$	$\overline{RDY}$ input hold time		0			ns

Timer B input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{C(TB)}$	TBi_{IN} input cycle time (one edge count)		125			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width (one edge count)		62			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width (one edge count)		62			ns
$t_{C(TB)}$	TBi_{IN} input cycle time (both edges count)		250			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width (both edges count)		125			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width (both edges count)		125			ns

Timer B input (Pulse period measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		500			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		250			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		250			ns

Timer B input (Pulse width measurement mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TB)}$	TBi_{IN} input cycle time		500			ns
$t_{W(TBH)}$	TBi_{IN} input high-level pulse width		250			ns
$t_{W(TBL)}$	TBi_{IN} input low-level pulse width		250			ns

A-D trigger input

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max.	
$t_{C(AD)}$	$\overline{AD_{TRG}}$ input cycle time (minimum allowable trigger)		1000			ns
$t_{W(ADL)}$	$\overline{AD_{TRG}}$ input low-level pulse width		125			ns

Serial I/O

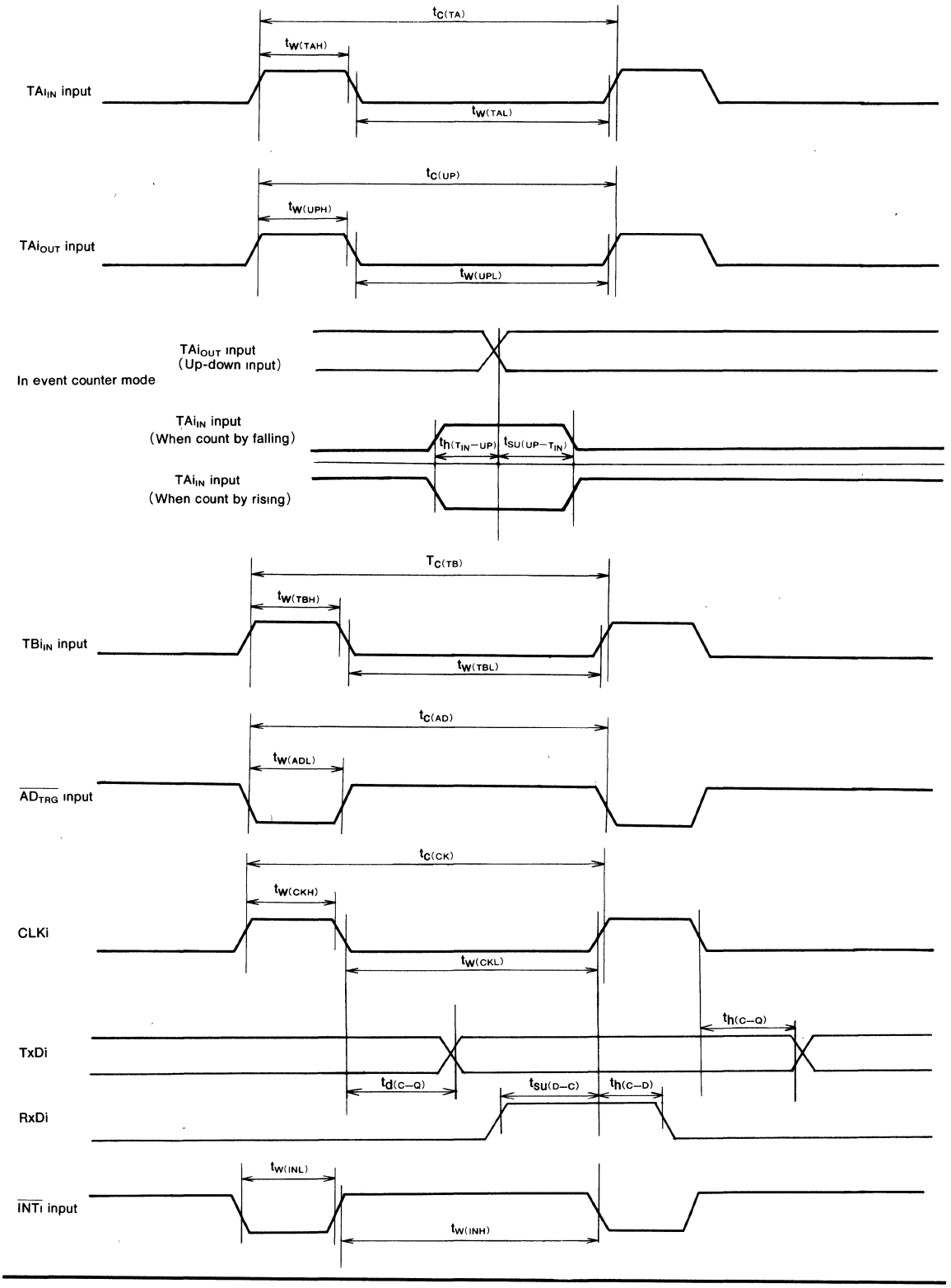
Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{C(CK)}$	CLKi input cycle time		250			ns
$t_{W(CKH)}$	CLKi input high-level pulse width		125			ns
$t_{W(CKL)}$	CLKi input low-level pulse width		125			ns
$t_{d(C-Q)}$	TxDi output delay time				90	ns
$t_{h(C-Q)}$	TxDi hold time		30			ns
$t_{su(D-C)}$	RxDi input setup time		30			ns
$t_{h(C-D)}$	RxDi input hold time		90			ns

External interrupt $\overline{INTi}$ input

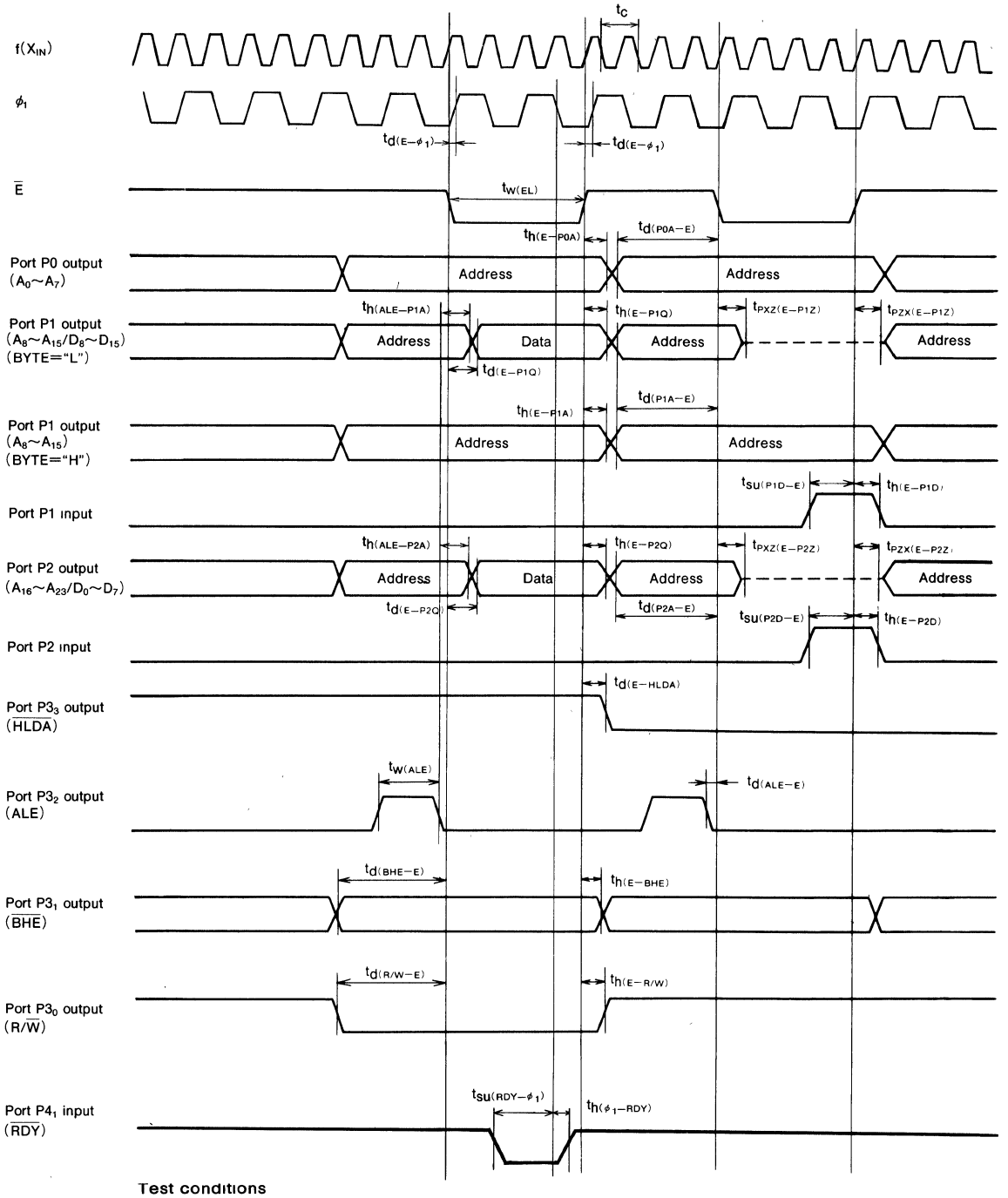
Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{W(INH)}$	$\overline{INTi}$ input high-level pulse width		250			ns
$t_{W(INL)}$	$\overline{INTi}$ input low-level pulse width		250			ns

Memory expansion mode and microprocessor mode (when wait bit="0", and external memory area is accessed)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 2	155			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				80	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				5	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		155			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				80	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				5	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		155			ns
$t_{d(E-HLDA)}$	$\overline{HLDA}$ output delay time				50	ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		165			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		155			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		155			ns
$t_{d(E-\phi_1)}$	ϕ_1 output delay time		0		20	ns
$t_{h(E-P0A)}$	Port P0 address hold time		25			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		25			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		25			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		25			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		25			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		25			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		220			ns

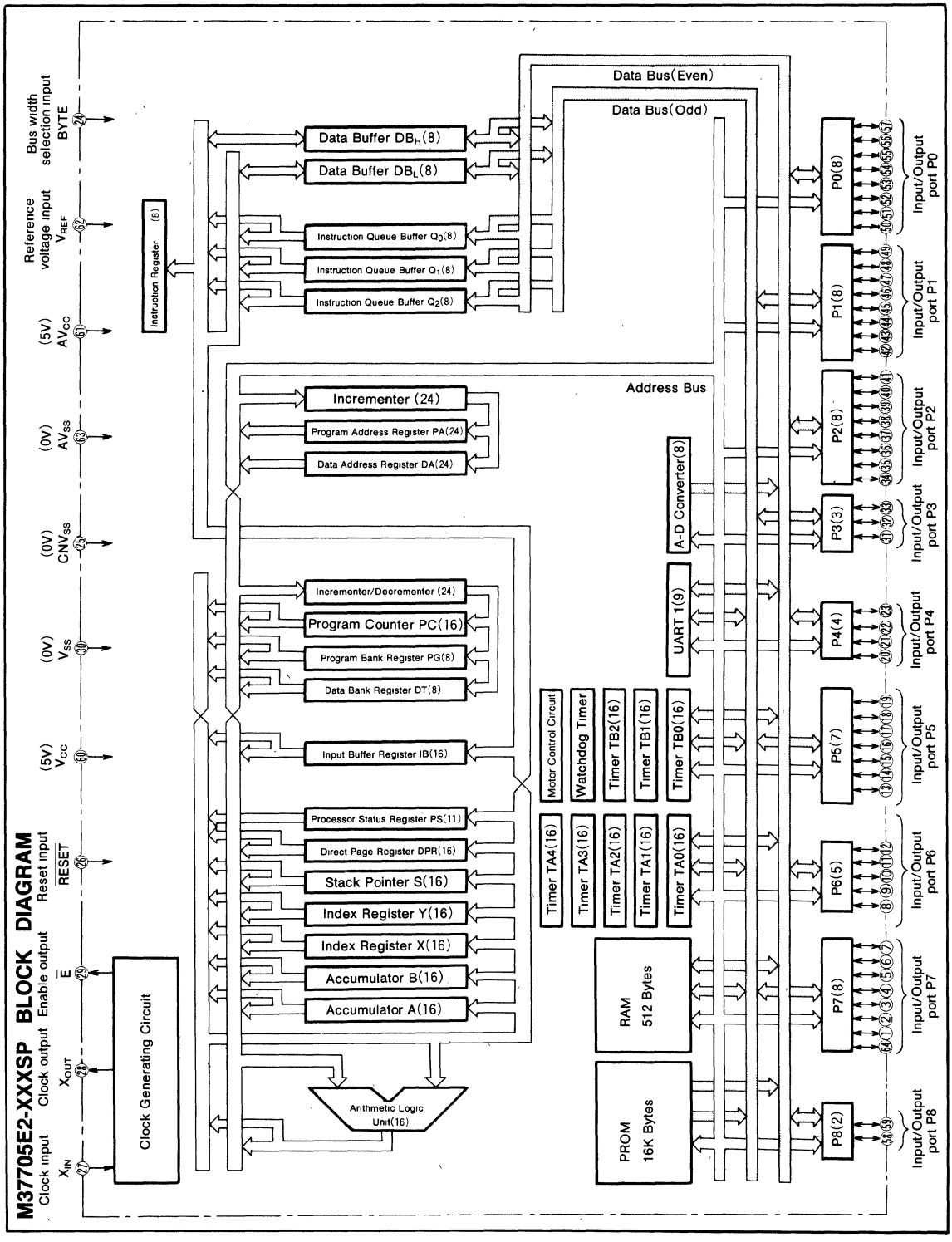

$t_{C(TA)}$
$t_{W(TAH)}$
TAi_{IN} input
$t_{W(TAL)}$
$t_{C(UP)}$
$t_{W(UPH)}$
TAi_{OUT} input
$t_{W(UPL)}$
TAi_{OUT} input
(Up-down input)
In event counter mode
TAi_{IN} input
(When count by falling)
$t_{h(T_{IN}-UP)}$
$t_{SU(UP-T_{IN})}$
TAi_{IN} input
(When count by rising)
$T_{C(TB)}$
$t_{W(TBH)}$
TBi_{IN} input
$t_{W(TBL)}$
$t_{C(AD)}$
$t_{W(ADL)}$
$\overline{AD_{TRG}}$ input
$t_{C(CK)}$
$t_{W(CKH)}$
CLKi
$t_{W(CKL)}$
$t_{h(C-Q)}$
TxDi
$t_{d(C-Q)}$
$t_{SU(D-C)}$
$t_{h(C-D)}$
RxDi
$t_{W(INL)}$
$\overline{INTi}$ input
$t_{W(INH)}$

Memory expansion mode and microprocessor mode (when wait bit="0", and external memory area is accessed)



Test conditions

- V_{CC} = 5 V ± 5 %
- Output timing voltage : V_{OL} = 0.8V, V_{OH} = 2.0V
- Ports P1,P2 input : V_{IL} = 0.8V, V_{IH} = 2.5V
- Port $P4_1$ input : V_{IL} = 1.0V, V_{IH} = 4.0V



MITSUBISHI ELECTRIC

PIN DESCRIPTION

Pin	Name	Input/Output	Functions
V_{CC}, V_{SS}	Power supply		Supply 5 V±5 % to V_{CC} and 0 V to V_{SS}
CNV_{SS}	CNV_{SS} input	Input	This pin controls the processor mode Connect to V_{SS} for single-chip mode.
$\overline{RESET}$	Reset input	Input	To enter the reset state, this pin must be kept at a "L" condition should be maintained for the required time
X_{IN}	Clock input	Input	These are I/O pins of internal clock generating circuit Connect a ceramic or quartz crystal resonator between X_{IN} and X_{OUT}. When an external clock is used, the clock source should be connected to the X_{IN} pin and the X_{OUT} pin should be left open.
X_{OUT}	Clock output	Output	
$\overline{E}$	Enable output	Output	Data or instruction read and data write are performed when output from this pin is "L".
BYTE	Bus width selection input	Input	In memory expansion mode or microprocessor mode, this pin determines whether the external data bus is 8-bit width or 16-bit width The width is 16 bits when "L" signal inputs and 8 bits when "H" signal inputs.
AV_{CC}, AV_{SS}	Analog supply input		Power supply for the A-D converter Connect AV_{CC} to V_{CC} and AV_{SS} to V_{SS} externally.
V_{REF}	Reference voltage input	Input	This is reference voltage input pin for the A-D converter.
$P0_0$~$P0_7$	I/O port P0	I/O	In single-chip mode, port P0 becomes an 8-bit I/O port. An I/O directional register is available so that each pin can be programmed for input or output These ports are in input mode when reset Address(A_7~A_0)is output in memory expansion mode or microprocessor mode.
$P1_0$~$P1_7$	I/O port P1	I/O	In single-chip mode, these pins have the same functions as port P0. When the BYTE pin is set to "L" in memory expansion mode or microprocessor mode and external data bus is 16-bit width, high-order data (D_{15}~D_8)is input or output when $\overline{E}$ output is "L" and an address (A_{15}~A_8) is output when $\overline{E}$ output is "H". If the BYTE pin is "H" that is an external data bus is 8-bit width, only address(A_{15}~A_8)is output.
$P2_0$~$P2_7$	I/O port P2	I/O	In single-chip mode, these pins have the same functions as port P0 In memory expansion mode or microprocessor mode low-order data(D_7~D_0) is input or output when $\overline{E}$ output is "L" and an address(A_{23}~A_{16})is output when $\overline{E}$ output is "H".
$P3_0$~$P3_2$	I/O port P3	I/O	In single-chip mode , these pins have the same functions as port P0 In memory expansion mode or microprocessor mode, R/$\overline{W}$, $\overline{BHE}$, and ALE signals are output
$P4_0$~$P4_2$, $P4_7$	I/O port P4	I/O	In single-chip mode, these pins have the same functions as port P0 In memory expansion mode or microprocessor mode, $P4_0$ and $P4_1$ become $\overline{HOLD}$ and $\overline{RDY}$ input pin respectively Functions of other pins are the same as in single-chip mode. In single-chip mode or memory expansion mode, port $P4_2$ can be programed for ϕ_1 output pin divided the clock to X_{IN} pin by 2. In microprocessor mode, $P4_2$ always has the fanction as ϕ_1 output pin
$P5_0$~$P5_6$	I/O port P5	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as I/O pins for timer A0, timer A1, timer A2, and output pin for timer A3. These pins also have the function as motor control output pin
$P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$	I/O port P6	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as output pins for timer A4, and input pins for external interrupt input $\overline{INT_0}$ and $\overline{INT_1}$ pins, and for timer B0 and timer B1, $P6_0$ and $P6_2$ also have the function as motor control output pins
$P7_0$~$P7_7$	I/O port P7	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as analog input AN_0~AN_7 input pins. $P7_7$ also has an A-D conversion trigger input function.
$P8_6$, $P8_7$	I/O port P8	I/O	In addition to having the same functions as port P0 in single-chip mode, these pins also function as R_XD and T_XD pins for UART 1.

EPROM MODE

The M37705E2-XXXSP features an EPROM mode in addition to its normal modes. When the $\overline{RESET}$ signal level is "L", the chip automatically enters the EPROM mode. Table 1 list the correspondence between pins and Fig. 1 gives the pin connections in the EPROM mode.

When in the EPROM mode, ports P0, P1, P2, $P5_1$, $P5_2$, CNV_{SS} and BYTE are used for the EPROM (equivalent to the M5M27C256K). When in this mode, the built-in PROM can be written to or read from using these pins in the same way as with the M5M27C256K.

This chip does not have Device Identifer Mode, so that set the corresponding program algorithm. The program area should specify address 4000_{16}~$7FFF_{16}$ for the M37705E2-XXXSP.

Connect the clock which is either ceramic resonator or external clock to X_{IN} pin and X_{OUT} pin.

Table 1 Pin function in EPROM programming mode

	M37705E2-XXXSP	M5M27C256K
V_{CC}	V_{CC}	V_{CC}
V_{PP}	CNV_{SS}, BYTE	V_{PP}
V_{SS}	V_{SS}	V_{SS}
Address input	Ports P0, $P1_0$~$P1_6$	A_0~A_{14}
Data I/O	Port P2	D_0~D_7
$\overline{CE}$	$P5_2$	$\overline{CE}$
$\overline{OE}$	$P5_1$	$\overline{OE}$

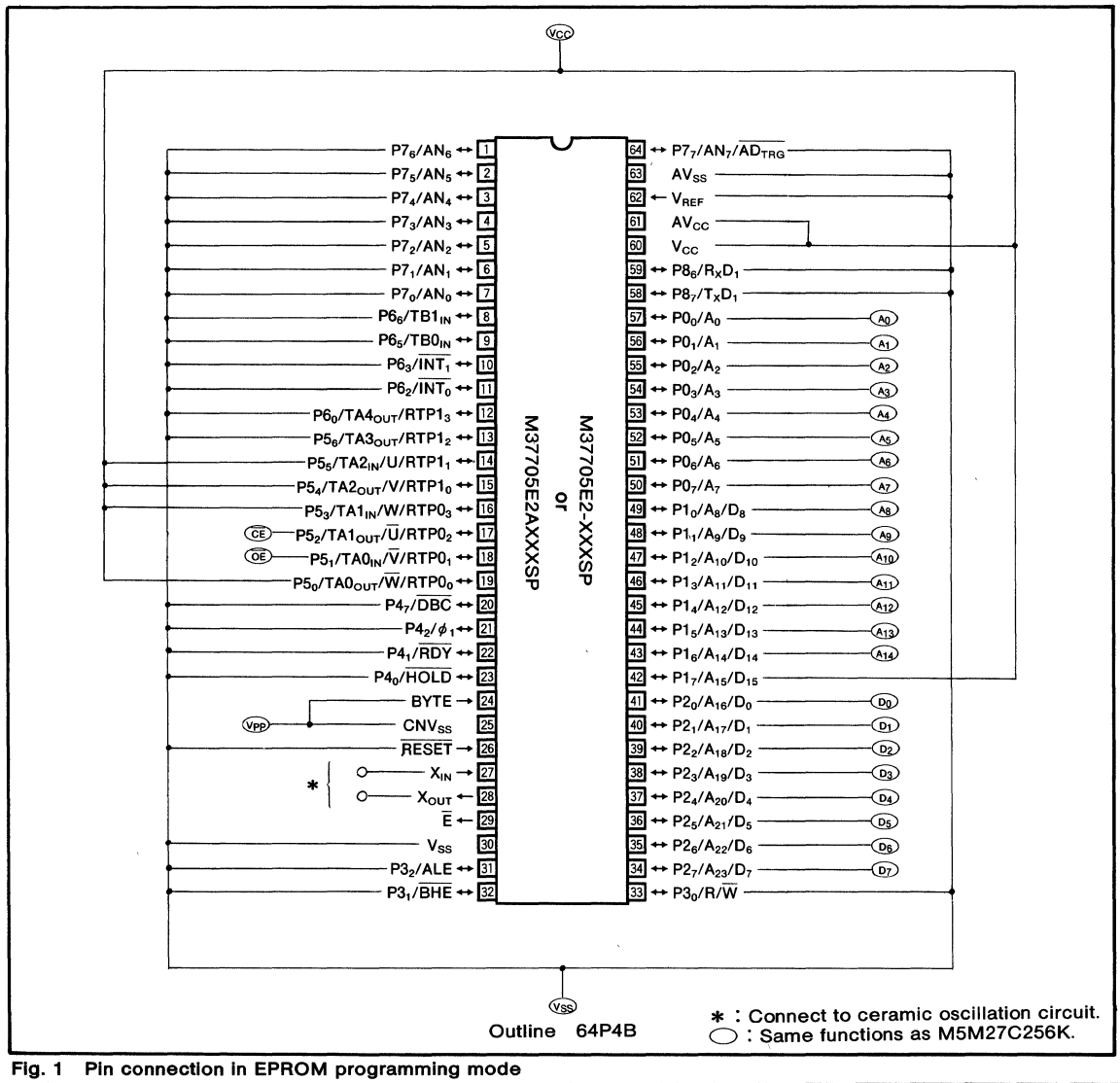
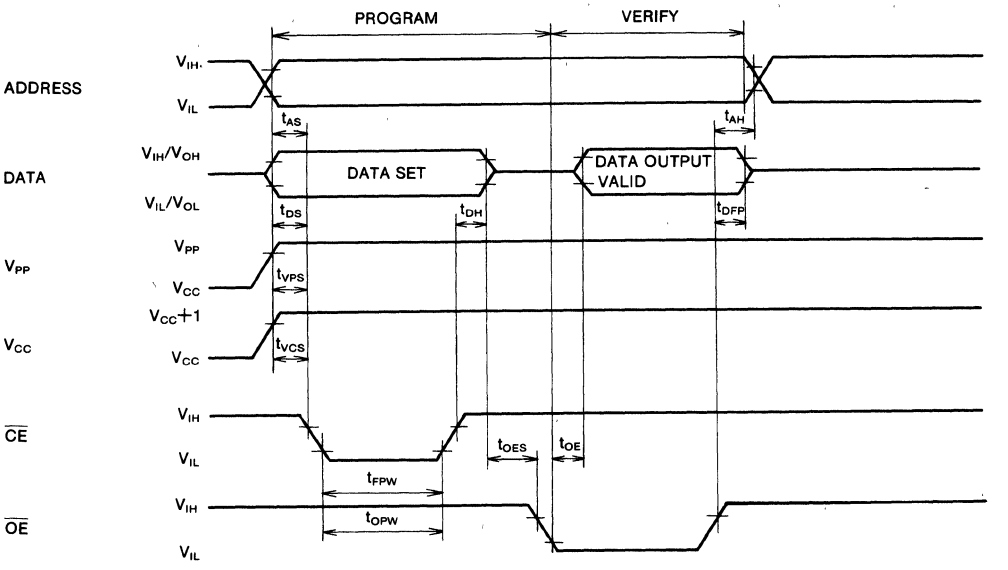
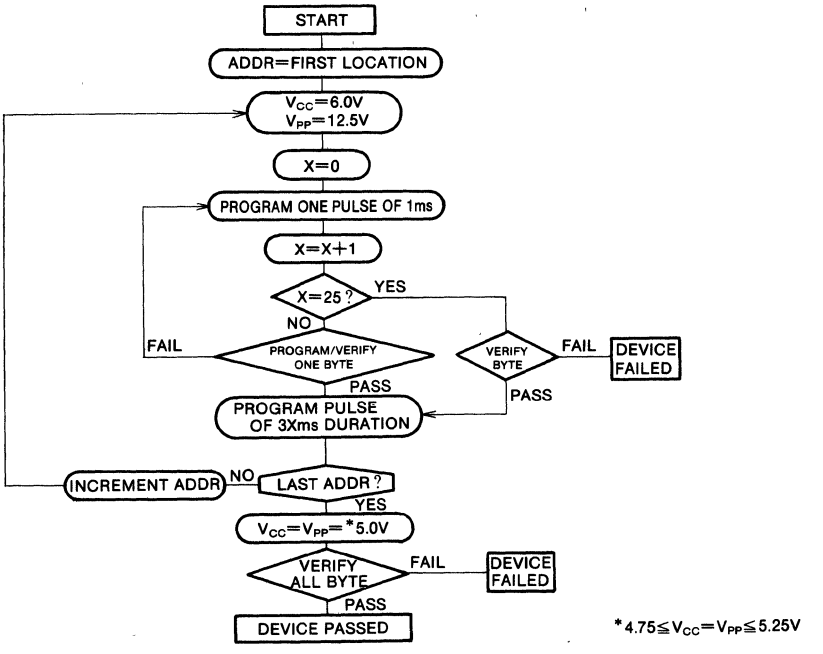


Fig. 1 Pin connection in EPROM programming mode

AC waveforms



Fast programming algorithm flow chart



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Supply voltage		−0.3∼7	V
AV_{CC}	Analog supply voltage		−0.3∼7	V
V_I	Input voltage $\overline{RESET}$, CNV_{SS}, BYTE		−0.3∼12(Note 1)	V
V_I	Input voltage $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$∼$P7_7$, $P8_6$, $P8_7$, V_{REF}, X_{IN}		−0.3∼V_{CC}+0.3	V
V_O	Output voltage $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$∼$P7_7$, $P8_6$, $P8_7$, X_{OUT}, $\overline{E}$		−0.3∼V_{CC}+0.3	V
P_d	Power dissipation	T_a=25℃	1000	mW
T_{opr}	Operating temperature		−10∼70	℃
T_{stg}	Storage temperature		−40∼125	℃

Note 1. Input voltage for CNV_{SS} and BYTE pins is 13V in writing to PROM.

RECOMMENDED OPERATING CONDITIONS (V_{CC}=5V±10%, T_a=−10∼70℃, unless otherwise noted)

Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max.	
V_{CC}	Supply voltage		4.75	5.0	5.25	V
AV_{CC}	Analog supply voltage			V_{CC}		V
V_{SS}	Supply voltage			0		V
AV_{SS}	Analog supply voltage			0		V
V_{IH}	High-level input voltage $P0_0$∼$P0_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$∼$P7_7$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE		0.8V_{CC}		V_{CC}	V
V_{IH}	High-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in single-chip mode)		0.8V_{CC}		V_{CC}	V
V_{IH}	High-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in memory expansion mode and microprocessor mode)		0.5V_{CC}		V_{CC}	V
V_{IL}	Low-level input voltage $P0_0$∼$P0_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$∼$P7_7$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE		0		0.2V_{CC}	V
V_{IL}	Low-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in single-chip mode)		0		0.2V_{CC}	V
V_{IL}	Low-level input voltage $P1_0$∼$P1_7$, $P2_0$∼$P2_7$ (in memory expansion mode and microprocessor mode)		0		0.16V_{CC}	V
$I_{OH(peak)}$	High-level peak output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$∼$P7_7$, $P8_6$, $P8_7$				−10	mA
$I_{OH(avg)}$	High-level average output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_0$∼$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$∼$P7_7$, $P8_6$, $P8_7$				−5	mA
$I_{OL(peak)}$	Low-level peak output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$∼$P7_7$, $P8_6$, $P8_7$				10	mA
$I_{OL(peak)}$	Low-level peak output current $P5_0$∼$P5_5$				20	mA
$I_{OL(avg)}$	Low-level average output current $P0_0$∼$P0_7$, $P1_0$∼$P1_7$, $P2_0$∼$P2_7$, $P3_0$∼$P3_2$, $P4_0$∼$P4_2$, $P4_7$, $P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$∼$P7_7$, $P8_6$, $P8_7$				5	mA
$I_{OL(avg)}$	Low-level average output current $P5_0$∼$P5_5$				15	mA
$f(X_{IN})$	External clock frequency input	M37705E2-XXXSP, M37705E2SS			8	MHz
		M37705E2AXXXSP, M37705E2ASS			16	

Note 1. Average output current is the average value of a100ms interval.
2. The sum of $I_{OL(peak)}$ for ports P0, P1, P2, P3, and P8 must be 80mA or less,
the sum of $I_{OH(peak)}$ for ports P0, P1, P2, P3, and P8 must be 80mA or less,
the sum of $I_{OL(peak)}$ for ports P4, P5, P6, and P7 must be 110mA or less, and
the sum of $I_{OH(peak)}$ for ports P4, P5, P6, and P7 must be 80mA or less.

TIMING REQUIREMENTS (V_{CC}=5V±5%, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

External clock input

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max	
t_C	External clock input cycle time		125			ns
$t_{W(H)}$	External clock input high-level pulse width		50			ns
$t_{W(L)}$	External clock input low-level pulse width		50			ns
t_r	External clock rise time				20	ns
t_f	External clock fall time				20	ns

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{SU(P0D-E)}$	Port P0 input setup time		200			ns
$t_{SU(P1D-E)}$	Port P1 input setup time		200			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		200			ns
$t_{SU(P3D-E)}$	Port P3 input setup time		200			ns
$t_{SU(P4D-E)}$	Port P4 input setup time		200			ns
$t_{SU(P5D-E)}$	Port P5 input setup time		200			ns
$t_{SU(P6D-E)}$	Port P6 input setup time		200			ns
$t_{SU(P7D-E)}$	Port P7 input setup time		200			ns
$t_{SU(P8D-E)}$	Port P8 input setup time		200			ns
$t_{h(E-P0D)}$	Port P0 input hold time		0			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(E-P3D)}$	Port P3 input hold time		0			ns
$t_{h(E-P4D)}$	Port P4 input hold time		0			ns
$t_{h(E-P5D)}$	Port P5 input hold time		0			ns
$t_{h(E-P6D)}$	Port P6 input hold time		0			ns
$t_{h(E-P7D)}$	Port P7 input hold time		0			ns
$t_{h(E-P8D)}$	Port P8 input hold time		0			ns

Memory expansion mode and microprocessor mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{SU(P1D-E)}$	Port P1 input setup time		60			ns
$t_{SU(P2D-E)}$	Port P2 input setup time		60			ns
$t_{SU(RDY-\phi1)}$	$\overline{RDY}$ input setup time		70			ns
$t_{h(E-P1D)}$	Port P1 input hold time		0			ns
$t_{h(E-P2D)}$	Port P2 input hold time		0			ns
$t_{h(\phi1-RDY)}$	$\overline{RDY}$ input hold time		0			ns

Timer A input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
$t_{C(TA)}$	TAi_{IN} input cycle time		250			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		125			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		125			ns

Timer A input (Gating input in timer mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		1000			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		500			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		500			ns

Timer A input (External trigger input in one-shot pulse mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
$t_{C(TA)}$	TAi_{IN} input cycle time		500			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		250			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		250			ns

Timer A input (External trigger input in pulse width modulation mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		250			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		250			ns

Timer A input (Up-down input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(UP)}$	TAi_{OUT} input cycle time		5000			ns
$t_{W(UPH)}$	TAi_{OUT} input high-level pulse width		2500			ns
$t_{W(UPL)}$	TAi_{OUT} input low-level pulse width		2500			ns
$t_{SU(UP-T_{IN})}$	TAi_{OUT} input setup time		1000			ns
$t_{h(T_{IN}-UP)}$	TAi_{OUT} input hold time		1000			ns

SWITCHING CHARACTERISTICS (V_{CC}=5V±5%, V_{SS}=0V, T_a=25℃, f(X_{IN})=8MHz, unless otherwise noted)

Single-chip mode

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max.	
$t_{d(E-P0Q)}$	Port P0 data output delay time	Figure 2			200	ns
$t_{d(E-P1Q)}$	Port P1 data output delay time				200	ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				200	ns
$t_{d(E-P3Q)}$	Port P3 data output delay time				200	ns
$t_{d(E-P4Q)}$	Port P4 data output delay time				200	ns
$t_{d(E-P5Q)}$	Port P5 data output delay time				200	ns
$t_{d(E-P6Q)}$	Port P6 data output delay time				200	ns
$t_{d(E-P7Q)}$	Port P7 data output delay time				200	ns
$t_{d(E-P8Q)}$	Port P8 data output delay time				200	ns

Memory expansion mode and microprocessor mode (when wait bit = "1")

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max.	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 2	100			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				120	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				5	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		100			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				120	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				5	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		100			ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		100			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		100			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		100			ns
$t_{d(E-\phi 1)}$	ϕ_1 output delay time		0		30	ns
$t_{h(E-P0A)}$	Port P0 address hold time		50			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		50			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		50			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		50			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		50			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		50			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		220			ns

M37705E2AXXXSP

ELECTRICAL CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min.	Limits Typ.	Limits Max	Unit
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_5$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$~$P7_7$, $P8_6$, $P8_7$	I_{OH}=−10mA	3			V
V_{OH}	High-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$	I_{OH}=−400μA	4.7			V
V_{OH}	High-level output voltage $P3_2$	I_{OH}=−10mA	3.1			V
		I_{OH}=−400μA	4.8			
V_{OH}	High-level output voltage $\overline{E}$	I_{OH}=−10mA	3.4			V
		I_{OH}=−400μA	4.8			
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$, $P4_0$~$P4_2$, $P4_7$, $P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$~$P7_7$, $P8_6$, $P8_7$	I_{OL}=10mA			2	V
V_{OL}	Low-level output voltage $P5_0$~$P5_5$	I_{OL}=20mA			2	V
V_{OL}	Low-level output voltage $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$, $P3_1$	I_{OL}=2mA			0.45	V
V_{OL}	Low-level output voltage $P3_2$	I_{OL}=10mA			1.9	V
		I_{OL}=2mA			0.43	
V_{OL}	Low-level output voltage $\overline{E}$	I_{OL}=10mA			1.6	V
		I_{OL}=2mA			0.4	
$V_{T+}-V_{T-}$	Hysteresis $\overline{HOLD}$, $\overline{RDY}$, $TA0_{IN}$~$TA2_{IN}$, $TB0_{IN}$, $TB1_{IN}$, $\overline{INT_0}$, $\overline{INT_1}$, $\overline{AD_{TRG}}$		0.4		1	V
$V_{T+}-V_{T-}$	Hysteresis $\overline{RESET}$		0.2		0.5	V
$V_{T+}-V_{T-}$	Hysteresis X_{IN}		0.1		0.3	V
I_{IH}	High-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_2$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$~$P7_7$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=5V			5	μA
I_{IL}	Low-level input current $P0_0$~$P0_7$, $P1_0$~$P1_7$, $P2_0$~$P2_7$, $P3_0$~$P3_2$, $P4_0$~$P4_2$, $P4_7$, $P5_0$~$P5_6$, $P6_0$, $P6_2$, $P6_3$, $P6_5$, $P6_6$, $P7_0$~$P7_7$, $P8_6$, $P8_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	V_I=0V			−5	μA
V_{RAM}	RAM hold voltage	When clock is stopped.	2			V
I_{CC}	Power supply current	In single-chip mode output only pin is open and other pins are V_{SS} during reset. f(X_{IN})=16MHz, square waveform		12	24	mA
		T_a=25℃ when clock is stopped.			1	μA
		T_a=70℃ when clock is stopped			10	

A-D CONVERTER CHARACTERISTICS (V_{CC}=5V, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits Min.	Limits Typ	Limits Max.	Unit
—	Resolution	V_{REF}=V_{CC}			8	Bits
—	Absolute accuracy	V_{REF}=V_{CC}			±3	LSB
R_{LADDER}	Ladder resistance	V_{REF}=V_{CC}	2		10	kΩ
t_{CONV}	Conversion time		14.25			μs
V_{REF}	Reference voltage		2		V_{CC}	V
V_{IA}	Analog input voltage		0		V_{REF}	V

Timer A input (Count input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ.	Max	
$t_{C(TA)}$	TAi_{IN} input cycle time		125			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		62			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		62			ns

Timer A input (Gating input in timer mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
$t_{C(TA)}$	TAi_{IN} input cycle time		500			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		250			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		250			ns

Timer A input (External trigger input in one-shot pulse mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(TA)}$	TAi_{IN} input cycle time		250			ns
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		125			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		125			ns

Timer A input (External trigger input in pulse width modulation mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{W(TAH)}$	TAi_{IN} input high-level pulse width		125			ns
$t_{W(TAL)}$	TAi_{IN} input low-level pulse width		125			ns

Timer A input (Up-down input in event counter mode)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
$t_{C(UP)}$	TAi_{OUT} input cycle time		2500			ns
$t_{W(UPH)}$	TAi_{OUT} input high-level pulse width		1250			ns
$t_{W(UPL)}$	TAi_{OUT} input low-level pulse width		1250			ns
$t_{SU(UP-T_{IN})}$	TAi_{OUT} input setup time		500			ns
$t_{h(T_{IN}-UP)}$	TAi_{OUT} input hold time		500			ns

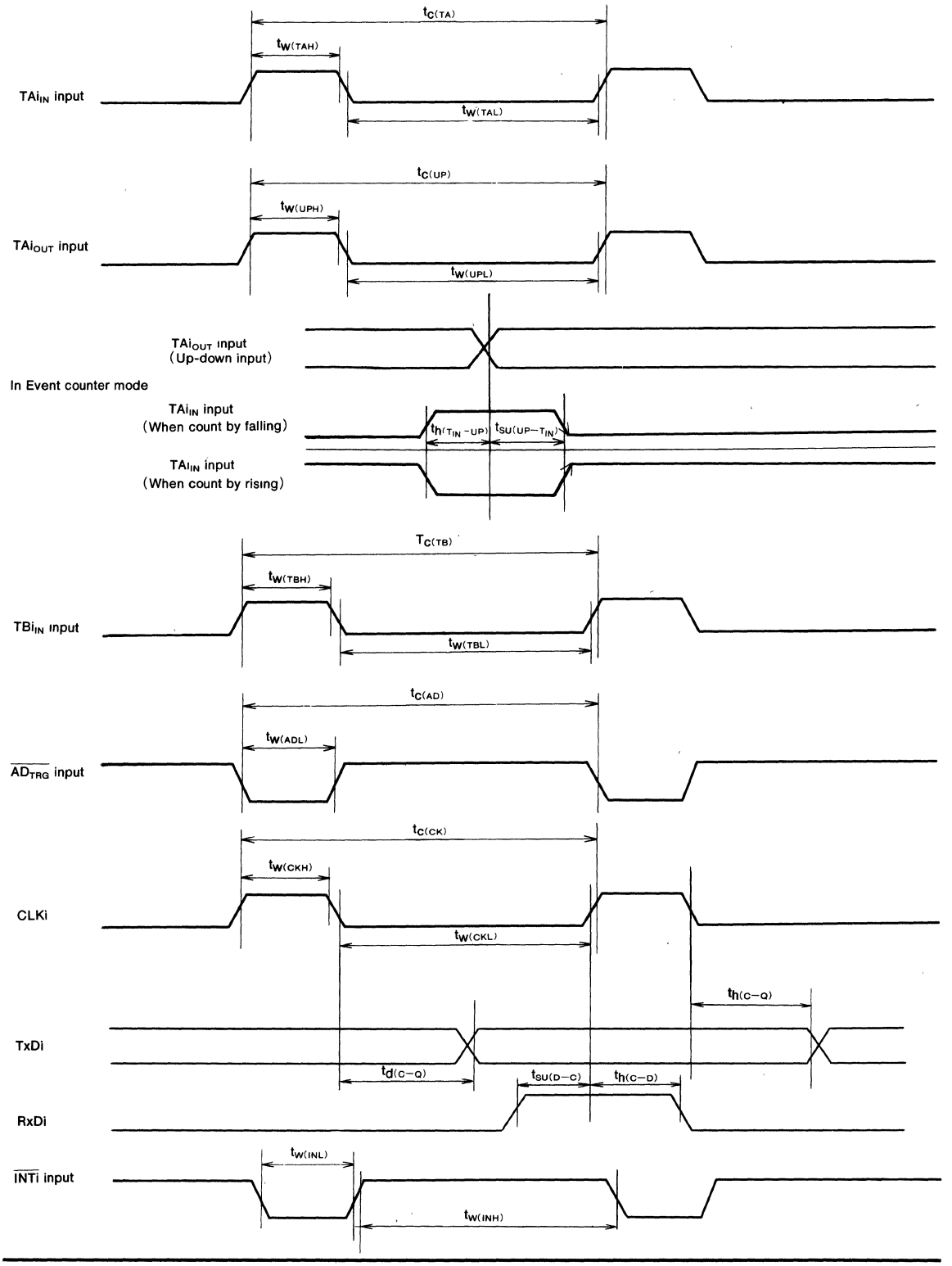
SWITCHING CHARACTERISTICS (V_{CC}=5V±5%, V_{SS}=0V, T_a=25℃, f(X_{IN})=16MHz, unless otherwise noted)

Single-chip mode

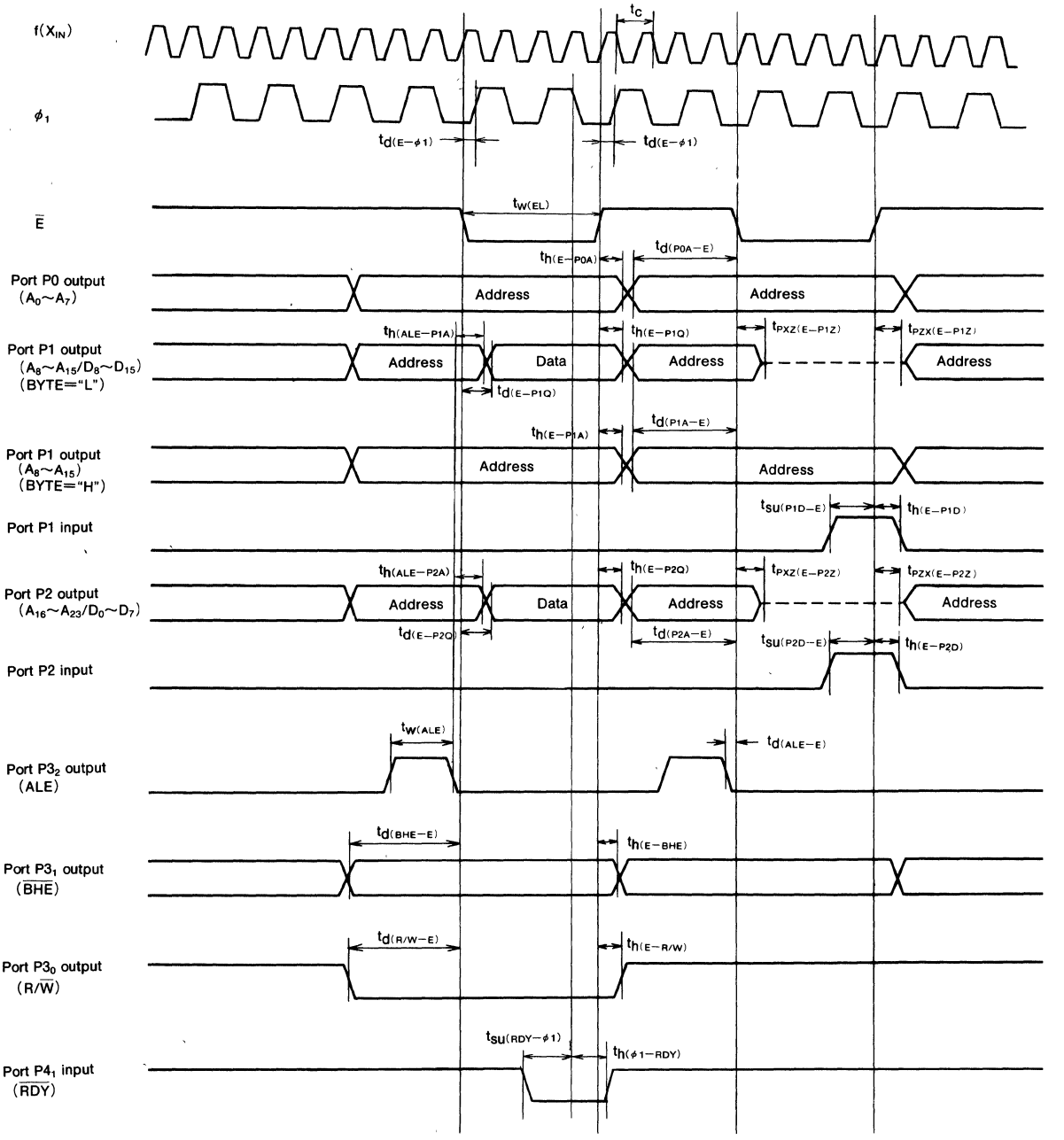
Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
$t_{d(E-P0Q)}$	Port P0 data output delay time	Figure 2			100	ns
$t_{d(E-P1Q)}$	Port P1 data output delay time				100	ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				100	ns
$t_{d(E-P3Q)}$	Port P3 data output delay time				100	ns
$t_{d(E-P4Q)}$	Port P4 data output delay time				100	ns
$t_{d(E-P5Q)}$	Port P5 data output delay time				100	ns
$t_{d(E-P6Q)}$	Port P6 data output delay time				100	ns
$t_{d(E-P7Q)}$	Port P7 data output delay time				100	ns
$t_{d(E-P8Q)}$	Port P8 data output delay time				100	ns

Memory expansion mode and microprocessor mode (when wait bit = "1")

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ	Max	
$t_{d(P0A-E)}$	Port P0 address output delay time	Figure 2	30			ns
$t_{d(E-P1Q)}$	Port P1 data output delay time (BYTE="L")				80	ns
$t_{PXZ(E-P1Z)}$	Port P1 floating start delay time (BYTE="L")				5	ns
$t_{d(P1A-E)}$	Port P1 address output delay time		30			ns
$t_{d(E-P2Q)}$	Port P2 data output delay time				80	ns
$t_{PXZ(E-P2Z)}$	Port P2 floating start delay time				5	ns
$t_{d(P2A-E)}$	Port P2 address output delay time		30			ns
$t_{d(ALE-E)}$	ALE output delay time		4			ns
$t_{W(ALE)}$	ALE pulse width		40			ns
$t_{d(BHE-E)}$	$\overline{BHE}$ output delay time		30			ns
$t_{d(R/W-E)}$	R/$\overline{W}$ output delay time		30			ns
$t_{d(E-\phi 1)}$	ϕ_1 output delay time		0		20	ns
$t_{h(E-P0A)}$	Port P0 address hold time		25			ns
$t_{h(ALE-P1A)}$	Port P1 address hold time (BYTE="L")		9			ns
$t_{h(E-P1Q)}$	Port P1 data hold time (BYTE="L")		25			ns
$t_{PZX(E-P1Z)}$	Port P1 floating release delay time (BYTE="L")		25			ns
$t_{h(E-P1A)}$	Port P1 address hold time (BYTE="H")		25			ns
$t_{h(ALE-P2A)}$	Port P2 address hold time		9			ns
$t_{h(E-P2Q)}$	Port P2 data hold time		25			ns
$t_{PZX(E-P2Z)}$	Port P2 floating release delay time		25			ns
$t_{h(E-BHE)}$	$\overline{BHE}$ hold time		20			ns
$t_{h(E-R/W)}$	R/$\overline{W}$ hold time		20			ns
$t_{W(EL)}$	$\overline{E}$ pulse width		95			ns

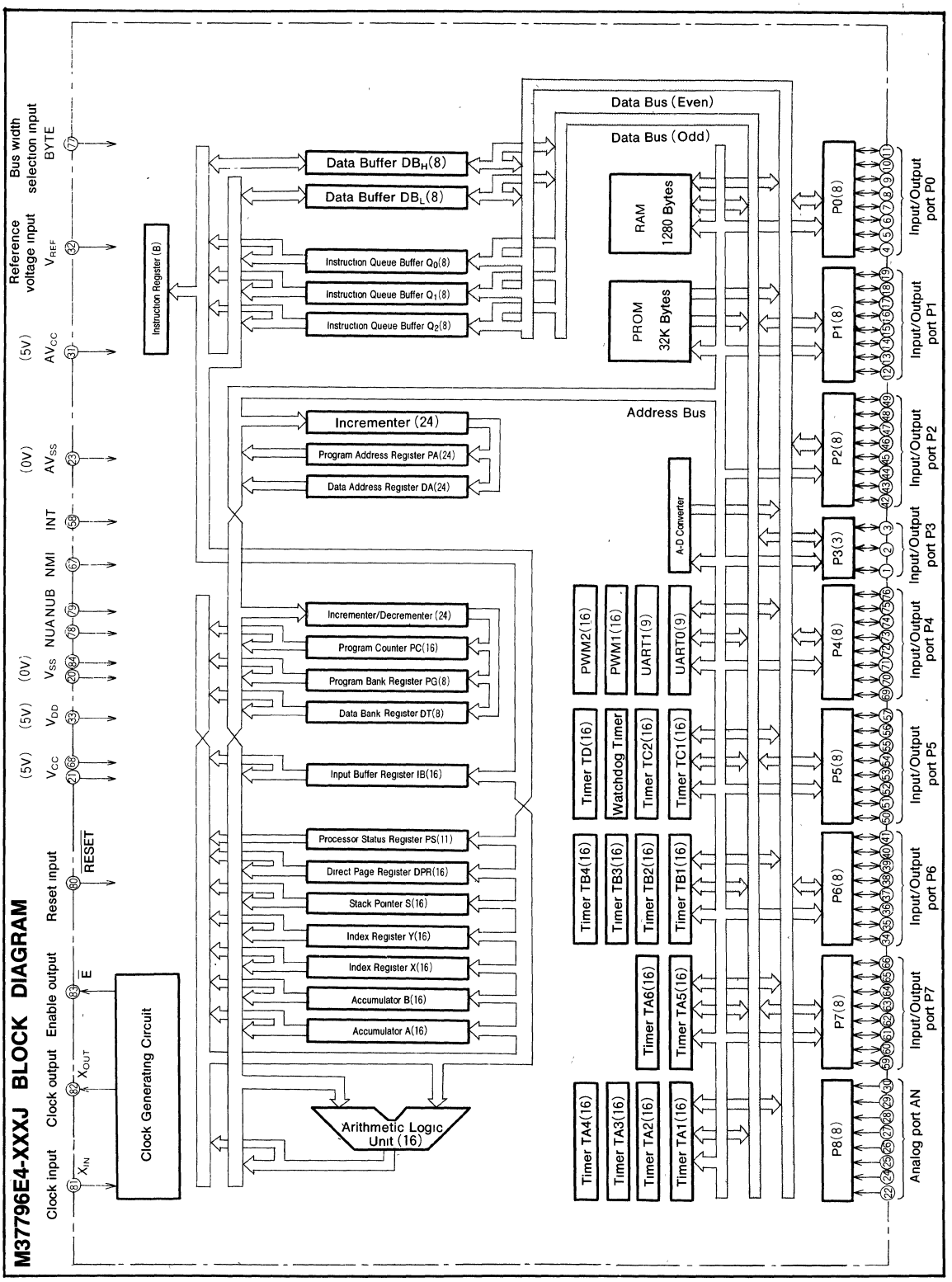

$t_{C(TA)}$
$t_{W(TAH)}$
TAi_{IN} input
$t_{W(TAL)}$
$t_{C(UP)}$
$t_{W(UPH)}$
TAi_{OUT} input
$t_{W(UPL)}$
TAi_{OUT} input
(Up-down input)
In Event counter mode
TAi_{IN} input
(When count by falling)
$t_{h(T_{IN}-UP)}$
$t_{SU(UP-T_{IN})}$
TAi_{IN} input
(When count by rising)
$T_{C(TB)}$
$t_{W(TBH)}$
TBi_{IN} input
$t_{W(TBL)}$
$t_{C(AD)}$
$t_{W(ADL)}$
$\overline{AD_{TRG}}$ input
$t_{C(CK)}$
$t_{W(CKH)}$
CLKi
$t_{W(CKL)}$
$t_{h(C-Q)}$
TxDi
$t_{d(C-Q)}$
$t_{SU(D-C)}$
$t_{h(C-D)}$
RxDi
$t_{W(INL)}$
$\overline{INTi}$ input
$t_{W(INH)}$

Memory expansion mode and microprocessor mode (When wait bit = "0", and external memory area is accessed)



Test conditions

- V_{CC} = 5 V ± 5 %
- Output timing voltage : V_{OL} = 0.8V, V_{OH} = 2.0V
- Port P1, P2 input : V_{IL} = 0.8V, V_{IH} = 2.5V
- Port $P4_1$ input : V_{IL} = 1.0V, V_{IH} = 4.0V



MITSUBISHI ELECTRIC

PIN DESCRIPTION (1)

Pin	Name	Input/ Output	Functions
V_{CC}, V_{DD}, V_{SS}	Power supply		Supply 5V ±10% to V_{CC}, V_{DD} and 0V to V_{SS}.
$\overline{RESET}$	Reset input	Input	To enter the reset state, this pin must be kept at a "L" condition should be maintained for the required time.
X_{IN}	Clock input	Input	These are I/O pins of internal clock generating circuit. Connect a ceramic or quartz crystal resonator between X_{IN} and X_{OUT}. When an external clock is used, the clock source should be connected to the X_{IN} pin and the X_{OUT} pin should be left open.
X_{OUT}	Clock output	Output	
NUA	NUA input	Input	These pins change processor mode Connect both pins to V_{SS} in single-chip mode.
NUB	NUB input	Input	
$\overline{E}$	Enable output	Output	Data or instruction read and data write are performed when output from this pin is "L".
BYTE	Bus width selection input	Input	This pin determines whether the external data bus is 8-bit width or 16-bit width. The width is 16-bit width when "L" signal inputs and 8-bit when "H" signal inputs.
NMI	Non-maskable interrupt input pin	Input	This is the non-maskable interrupt input pin. Connect with GND as this pin is not usually used
INT	Interrupt input	Input	This pin is interrupt input pin which can use edge or level.
AV_{CC} AV_{SS}	A-D power supply		Power supply for the A-D converter.
V_{REF}	Reference voltage input	Input	This is reference voltage input pin for A-D converter.
$P0_0$~$P0_7$	I/O port P0	I/O	In single-chip mode, port P0 is 8-bit CMOS input/output port Each pin can be programmed for input or output pin by the direction register.
D_0/A_0~D_7/A_7 ($P0_0$~$P0_7$)	Data (low-order) / Address bus (low-order)	I/O	In the other mode except for single chip-mode, Multiplex signal consisting of low-order 8-bit of data bus and address bus is input/output.
$P1_0$~$P1_7$	I/O port P1	I/O	In single-chip mode, port P1 is 8-bit CMOS input/output same as port P0.
D_8/A_8~D_{15}/A_{15} ($P1_0$~$P1_7$)	Data (high-order) /Address bus (middle-order)	I/O	In the other mode except for single chip-mode, Multiplex signal consisting of high-order 8-bit of data bus and middle-order 8-bit address bus is input/output.
$P2_0$~$P2_7$	I/O port P2	I/O	In single-chip mode, port P2 is 8-bit CMOS input/output same as port P0.
A_{16}~A_{23} ($P2_0$~$P2_7$)	Address bus (high-order)	Output	In memory expansion mode and microprocessor mode, 8-bit high-order address are output.
WEMP0($P2_0$) REMP0($P2_1$) WEMP1($P2_2$) REMP1($P2_3$) WEMP2($P2_4$) REMP2($P2_5$) WEMP3($P2_6$) REMP3($P2_7$)	Emulation strobe signal	Output	In port emulation mode, these pins are strobe signal output pins which indicates writing or reading to port P0~P3 with "H" level

PIN DESCRIPTION (PROM MODE)

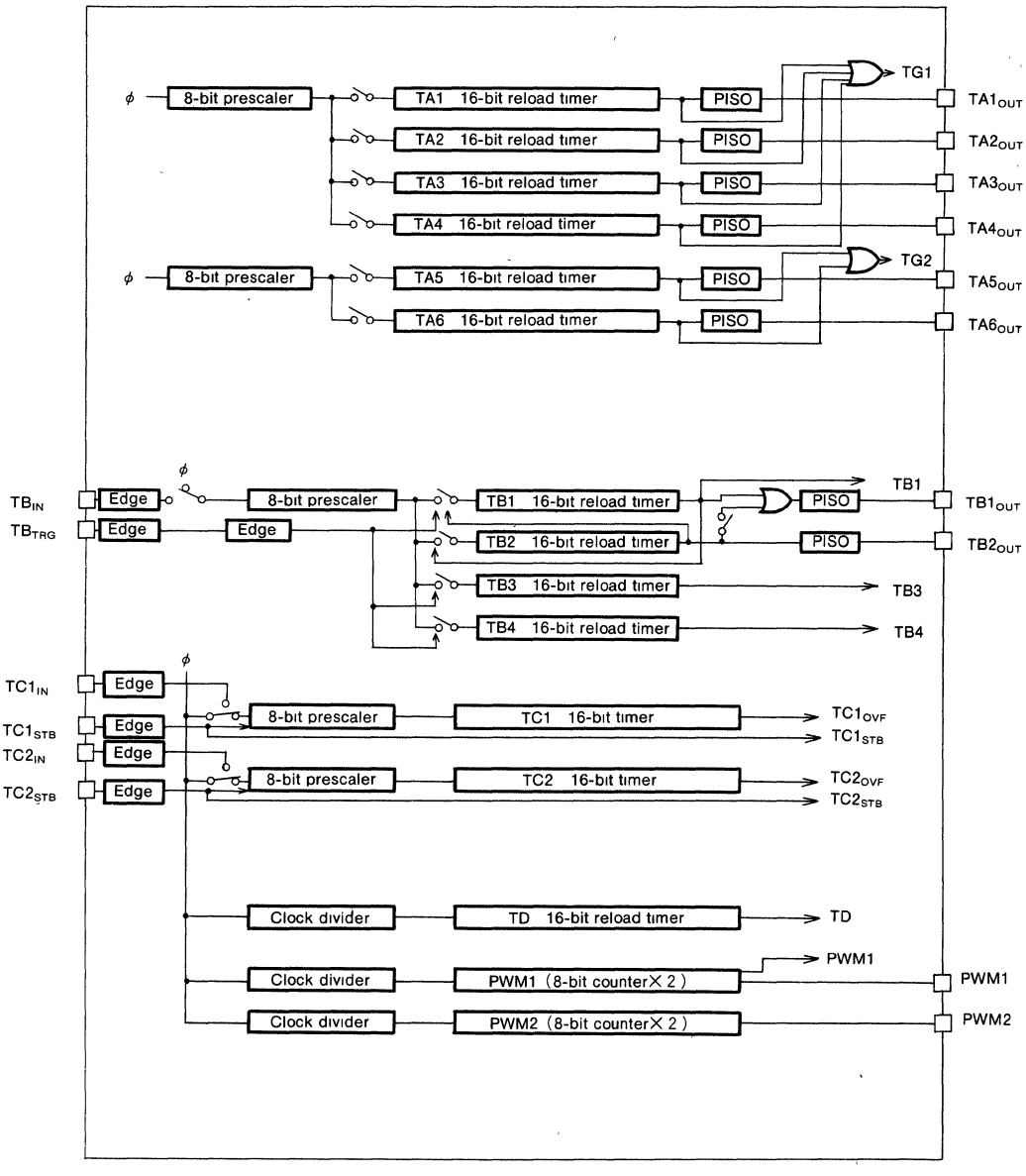
Pin	Name	Input/Output	Functions
V_{CC}, V_{DD}, V_{SS}	Power supply		Supply 5 V± 5 % to V_{CC}, V_{DD} and 0 V to V_{SS}.
INT	V_{PP} input	Input	Connect to V_{PP} when programming or verifing
BYTE	V_{PP} input	Input	Connect to V_{PP} when programming or verifing.
$\overline{RESET}$	Reset input	Input	Connect to V_{SS}.
X_{IN}	Clock input	Input	Connect a ceramic resonator between X_{IN} and X_{OUT}.
X_{OUT}	Clock output	Output	
$\overline{E}$	Enable output	Output	Keep open.
AV_{CC}, AV_{SS}	A-D power supply	Input	Connect AV_{CC} to V_{CC}, and AV_{SS} to V_{SS}.
V_{REF}	Reference voltage input	Input	Connect to V_{SS}
$P0_0$~$P0_7$	Data I/O (D_0~D_7)	I/O	Port P0 functions as the 8 bits data bus (D_0~D_7)
$P1_0$~$P1_7$	Address input (A_8~A_{14})	Input	Port $P1_0$~$P1_6$ functions as the higher 7 bits address input (A_8~A_{14}). Connect $P1_7$ to V_{CC}.
$P2_0$~$P2_7$	Address input (A_0~A_7)	Input	Port P2 functions as the lower 8 bits address input (A_0~A_7).
$P3_0$~$P3_3$	Input port P3	Input	Connect to V_{SS}.
$P4_0$~$P4_7$	Input port P4	Input	Connect $P4_3$ and $P4_5$ to V_{CC}, and $P4_0$~$P4_2$, $P4_4$ and $P4_6$ to V_{SS} Keep $P4_7$ open.
$P5_0$~$P5_7$	Control signal input	Input	$P5_1$ and $P5_2$ function as $\overline{OE}$ and $\overline{CE}$ input pin Connect $P5_0$, $P5_3$, $P5_4$, $P5_5$ and $P5_7$ to V_{CC}. Connect $P5_6$ to V_{SS}.
$P6_0$~$P6_7$	Input port P6	Input	Connect to V_{SS}.
$P7_0$~$P7_7$	Input port P7	Input	Connect to V_{SS}.
AN_0~AN_7	Input port P8	Input	Connect to V_{SS}.
NMI		Input	Connect to V_{SS}.
NUA, NUB		Input	Connect to V_{SS}.

Address (Hexadecimal notation)	
000000	
000001	
000002	
000003	
000004	
000005	
000006	
000007	
000008	Port P0 data register
000009	Port P1 data register
00000A	Port P2 data register
00000B	Port P3 data register
00000C	Port P4 data register
00000D	Port P5 data register
00000E	Port P6 data register
00000F	Port P7 data register
000010	Port P0 direction register
000011	Port P1 direction register
000012	Port P2 direction register
000013	Port P3 direction register
000014	Port P4 direction register
000015	Port P5 direction register
000016	Port P6 direction register
000017	Port P7 direction register
000018	
000019	
00001A	
00001B	
00001C	Port P4 operation mode register
00001D	Port P5 operation mode register
00001E	Port P6 operation mode register
00001F	Port P7 operation mode register
000020	A-D control register
000021	
000022	A-D succesive approximation register
000023	
000024	
000025	
000026	
000027	
000028	
000029	
00002A	
00002B	
00002C	
00002D	
00002E	
00002F	
000030	UART0 transmit/receive mode register
000031	UART0 baud rate register
000032	UART0 transmission buffer register
000033	
000034	UART0 transmission/receive control register 0
000035	UART0 transmission/receive control register 1
000036	UART0 receive buffer register
000037	
000038	UART1 transmit/receive mode register
000039	UART1 baud rate register
00003A	UART1 transmission buffer register
00003B	
00003C	UART1 transmission/receive control register 0
00003D	UART1 transmission/receive control register 1
00003E	UART1 receive buffer register
00003F	

Address (Hexadecimal notation)	
000040	Timer A1 counter
000041	
000042	Timer A1 reload register
000043	
000044	Timer A2 counter
000045	
000046	Timer A2 reload register
000047	
000048	Timer A3 counter
000049	
00004A	Timer A3 reload register
00004B	
00004C	
00004D	
00004E	
00004F	
000050	Timer A4 counter
000051	
000052	Timer A4 reload register
000053	
000054	Timer A5 counter
000055	
000056	Timer A5 reload register
000057	
000058	Timer A6 counter
000059	
00005A	Timer A6 reload register
00005B	
00005C	
00005D	
00005E	
00005F	
000060	TG1 prescaler
000061	TG2 prescaler
000062	Timer A control register-enable
000063	Timer A protect register
000064	Timer A control register-CW
000065	Timer A control register-P/N
000066	Timer A interrupt mask register
000067	Timer A interrupt status register
000068	Timer A1 PISO register
000069	Timer A2 PISO register
00006A	Timer A3 PISO register
00006B	Timer A4 PISO register
00006C	Timer A5 PISO register
00006D	Timer A6 PISO register
00006E	
00006F	
000070	Timer D counter
000071	
000072	Timer D reload register
000073	
000074	Timer D operation control register
000075	
000076	
000077	
000078	
000079	
00007A	
00007B	
00007C	
00007D	
00007E	
00007F	

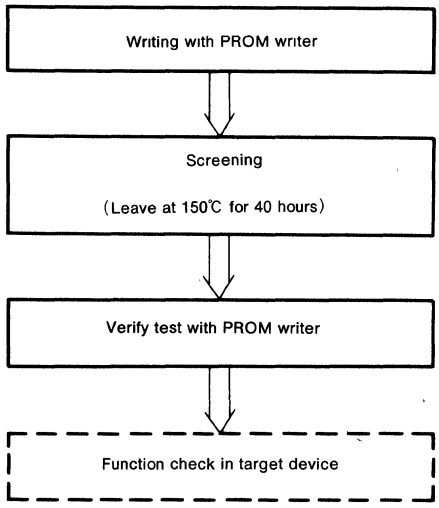
Fig. 2 Location of peripheral devices control registers (1)

M37796E4-XXXJ TIMER SYSTEM BLOCK DIAGRAM



SAFETY INSTRUCTIONS

The programmable M37796E4J and M37796E4TJ that are shipped in blank are also provided. For the M37796E4J and M37796E4TJ, Mitsubishi Electric corp. does not perform PROM write test and screening in the assembly process and following processes. To improve reliability after write, performing write and test according to the flow below before use is recommended.



ADDRESSING MODES

The M37796E4-XXXJ has 28 powerful addressing modes. Refer to the Series MELPS 7700 addressing mode description for the details of each addressing mode.

MACHINE INSTRUCTION LIST

The M37796E4-XXXJ has 103 machine instructions. Refer to the Series MELPS 7700 machine instruction list for details.

DATA REQUIRED FOR PROM ORDERING

Please send the following data for writing to PROM.

(1) M37796E4-XXXJ writing to PROM order confirmation form
(2) Mark specification form for 84P0
(3) ROM data (EPROM 3 sets)

M37796E4-XXXJ

ELECTRICAL CHARACTERISTICS (V_{CC}=V_{DD}=5V±10%, V_{SS}=0V, T_a=−20~75℃, f(X_{IN})=8MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
V_{OH}	High-level output voltage	I_{OH}=−2mA	V_{CC}−1			V
V_{OL}	Low-level output voltage	I_{OL}=2mA			0.45	V
I_{IH}	High-level input current	V_I=V_{CC}	−5		5	μA
I_{IL}	Low-level input current	V_I=0V	−5		5	μA
V_{DD}	RAM hold voltage		2.5		5.5	V
I_{DD}	RAM hold current	T_a=25℃, when clock is stopped			1	μA
		T_a=75℃, when clock is stopped			7.5	
I_{CC}	Power supply current (Note 1)	f(X_{IN})=8MHz, square waveform		15(Note 2)	30	mA
		T_a=25℃, when clock is stopped			1	μA
		T_a=75℃ when clock is stopped			15	
V_{T+}−V_{T-}	Hysteresis TB_{IN}, TB_{TRG}, $TC1_{IN}$, $TC2_{IN}$, $TC1_{STB}$, $TC2_{STB}$, $\overline{RESET}$ (Note3)	V_{CC}=5V	0.8			V
V_{T+}−V_{T-}	Hysteresis $\overline{HOLD}$, CLK0, CLK1, $\overline{CTS0/RTS0}$, $\overline{CTS1/RTS1}$, INT, NMI (Note 4)	V_{CC}=5V	0.2			V

Note 1. When reset with output pins open and connecting input pins to V_{SS}
2. V_{CC}=5V, T_a=25℃
3. Double function except for $\overline{RESET}$ pin
4. Double function except for INT and NMI pins

A-D CONVERTER CHARACTERISTICS (AV_{CC}=V_{REF}=5.12V, T_a=25℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ	Max.	
—	Resolution			10	Bits
—	Absolute accuracy			±3	LSB
—	Offset error			±2	LSB
—	Full-scale error (Note 1)			±2	LSB
t_{CONV}	Conversion time	40.5			μs
I_{IAN}	Analog input leak current (Note 2)	−200		200	nA

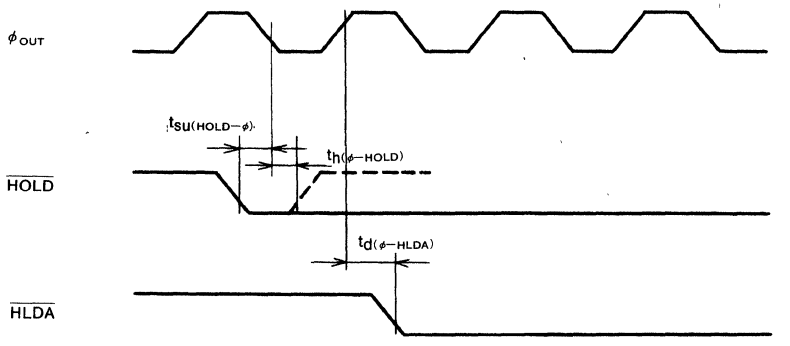
Note 1. The difference from ideal 10-bit A-D converter characteristics when offset error is not corrected.
2. Input leak current of AN_0~AN_7 with A-D converter halted. Input voltage is 0≦V_I≦AV_{CC}.

TIMING REQUIREMENTS (V_{CC}=V_{DD}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
t_C	Cycle time	125			ns
$t_{SU(D-E)}$	Data input setup time	80			ns
$t_{H(E-D)}$	Data input hold time	0			ns

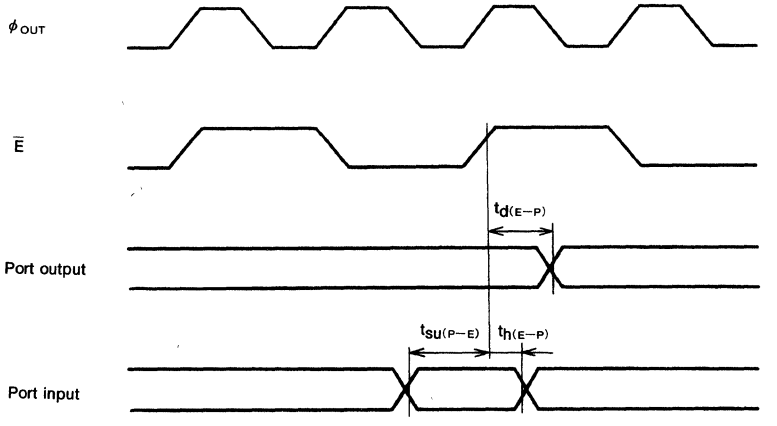
$\overline{\text{HOLD}}$ characteristics (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{su(HOLD-\phi)}$	$\overline{\text{HOLD}}$ input setup time	100			ns
$t_{h(\phi-HOLD)}$	$\overline{\text{HOLD}}$ input hold time	30			ns
$t_{d(\phi-HLDA)}$	$\overline{\text{HLDA}}$ output delay time			80	ns



Port characteristics (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

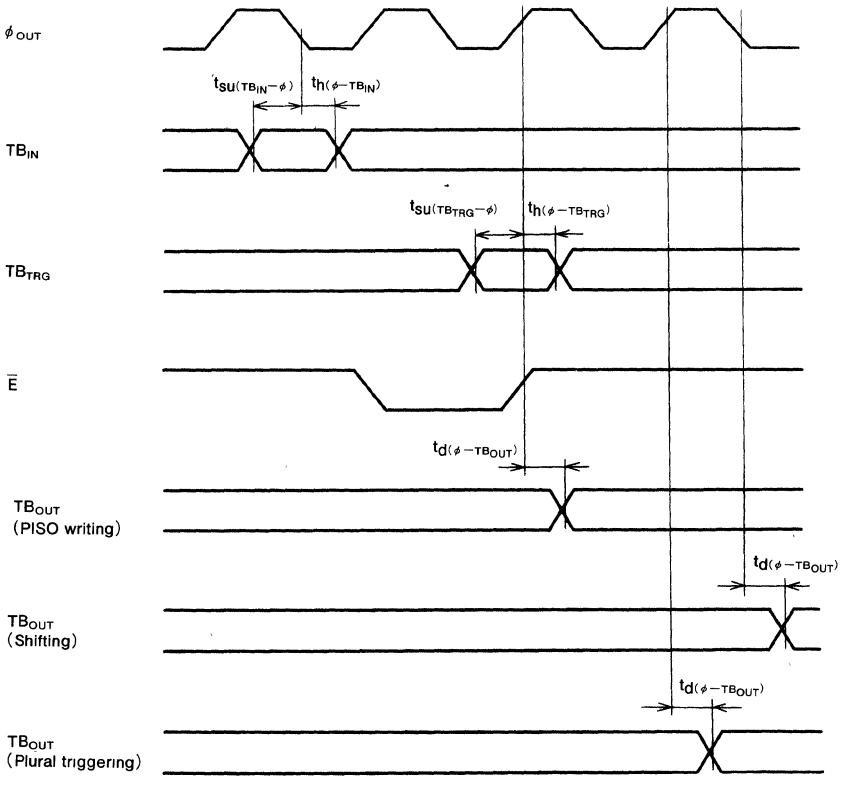
Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{su(P-E)}$	Port input setup time	200			ns
$t_{h(E-P)}$	Port input hold time	20			ns
$t_{d(E-P)}$	Port data output delay time			200	ns



Timer B characteristics (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

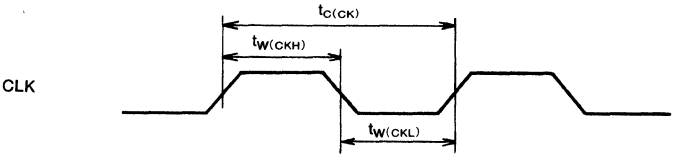
Symbol	Parameter		Limits			Unit
			Min.	Typ	Max	
$t_{su(TB_{IN}-\phi)}$	TB_{IN} input setup time		200			ns
$t_{h(\phi-TB_{IN})}$	TB_{IN} input hold time		0			ns
$t_{su(TB_{STB}-\phi)}$	TB_{STB} input setup time		200			ns
$t_{h(\phi-TB_{STB})}$	TB_{STB} input hold time		0			ns
$t_{d(\phi-TB_{OUT})}$	TB_{OUT} output delay time	PISO writing			200	ns
		Shifting			200	
		Plural triggering (Note 1)			200	

Note 1. Shift by TB2 underflow with plural trigger.



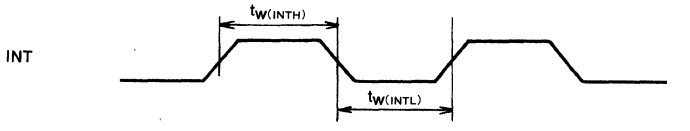
UART characteristics (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ	Max	
$t_{C(CK)}$	CLK input cycle time	500			ns
$t_{W(CKH)}$	CLK input high-level pulse width	250			ns
$t_{W(CKL)}$	CLK input low-level pulse width	250			ns



INT characteristics (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min	Typ	Max.	
$t_{W(INTH)}$	INT input high-level pulse width	250			ns
$t_{W(INTL)}$	INT input low-level pulse width	250			ns



SWITCHING CHARACTERISTICS (V_{CC}=V_{DD}=5V±10%, T_a=−40~85℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{W(ALE)}$	ALE pulse width	80			ns
$t_{C(ALE-E)}$	$\overline{E}$ end time after ALE	300			ns
$t_{d(ALE-E)}$	$\overline{E}$ start time after ALE	0			ns
$t_{W(E)}$	$\overline{E}$ pulse width	180			ns
$t_{pzv(E-DZ)}$	Floating release delay time	60			ns
$t_{d(A-E)}$	Address output delay time	100			ns
$t_{pvz(E-DZ)}$	Floating start delay time			40	ns
$t_{d(E-D)}$	Data output delay time			100	ns
$t_{v(ALE-A)}$	Address valid time after ALE	10			ns
$t_{v(E-D)}$	Data valid time after $\overline{E}$	10			ns
$t_{v(E-A)}$	Address valid time after $\overline{E}$	10			ns
$t_{d(CONT-E)}$	Control signal delay time	100			ns
$t_{v(E-CONT)}$	Control signal valid time	10			ns

Note. Limits have guaranties under load capacity of the test pin=100pF including test tool's capacity. If capacities of pins are much different from each other, limits may not be followed. So pay attention to the design of the board.

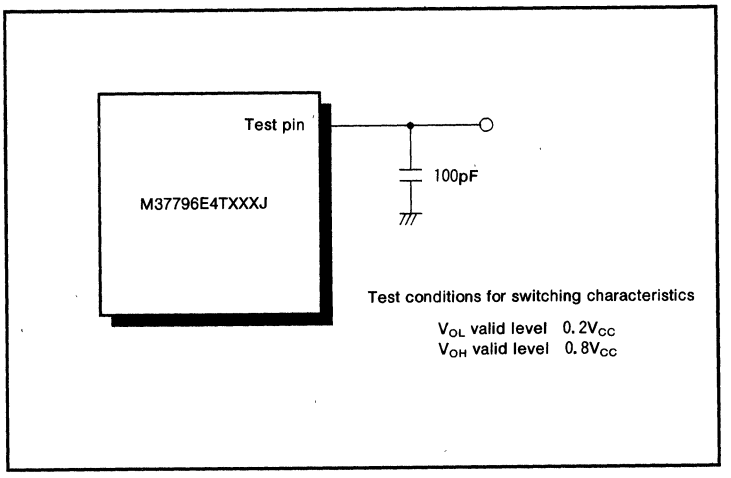
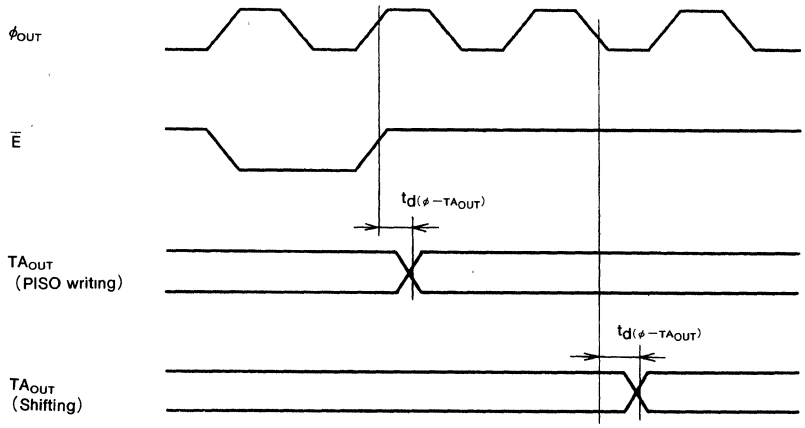


Fig. 5 Testing circuit

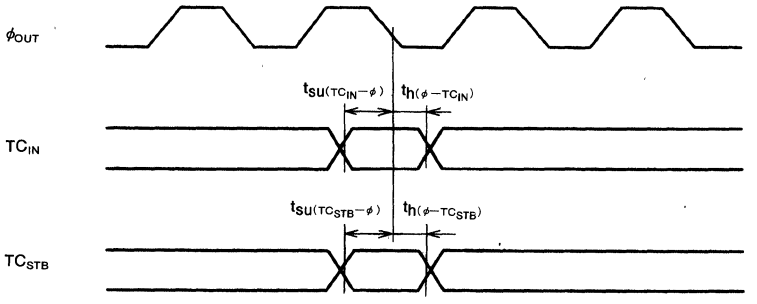
Timer A characteristics (V_{CC}=5V±10%, T_a=−40~85℃, f(X_{IN})=8MHz)

Symbol	Parameter		Limits			Unit
			Min.	Typ.	Max	
$t_{d(\phi-TA_{OUT})}$	TA_{OUT} output delay time	PISO writing			200	ns
		Shifting			200	



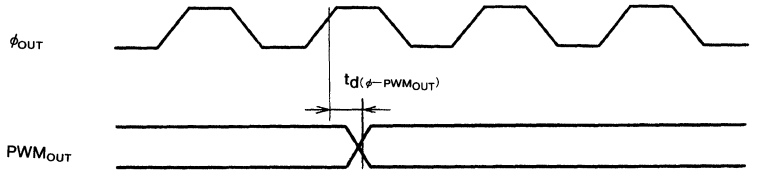
Timer C characteristics (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max.	
$t_{su(TC_{IN}-\phi)}$	TC_{IN} input setup time	200			ns
$t_{h(\phi-TC_{IN})}$	TC_{IN} input hold time	0			ns
$t_{su(TC_{STB}-\phi)}$	TC_{STB} input setup time	200			ns
$t_{h(\phi-TC_{STB})}$	TC_{STB} input hold time	0			ns

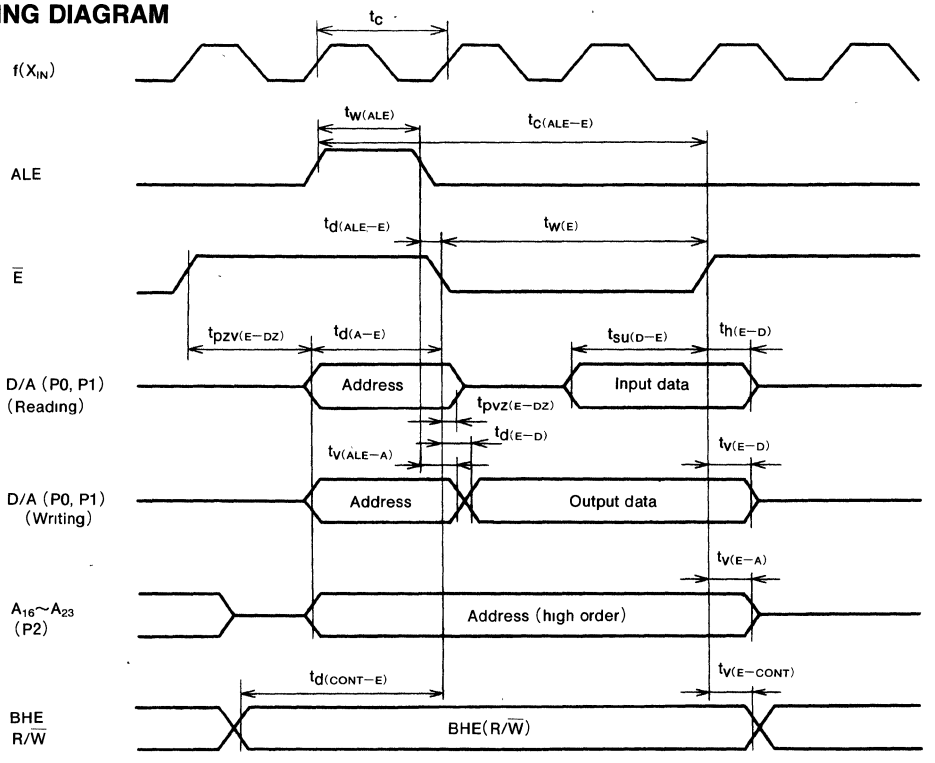


PWM characteristics (V_{CC}=5V±10%, T_a=−20~75℃, f(X_{IN})=8MHz)

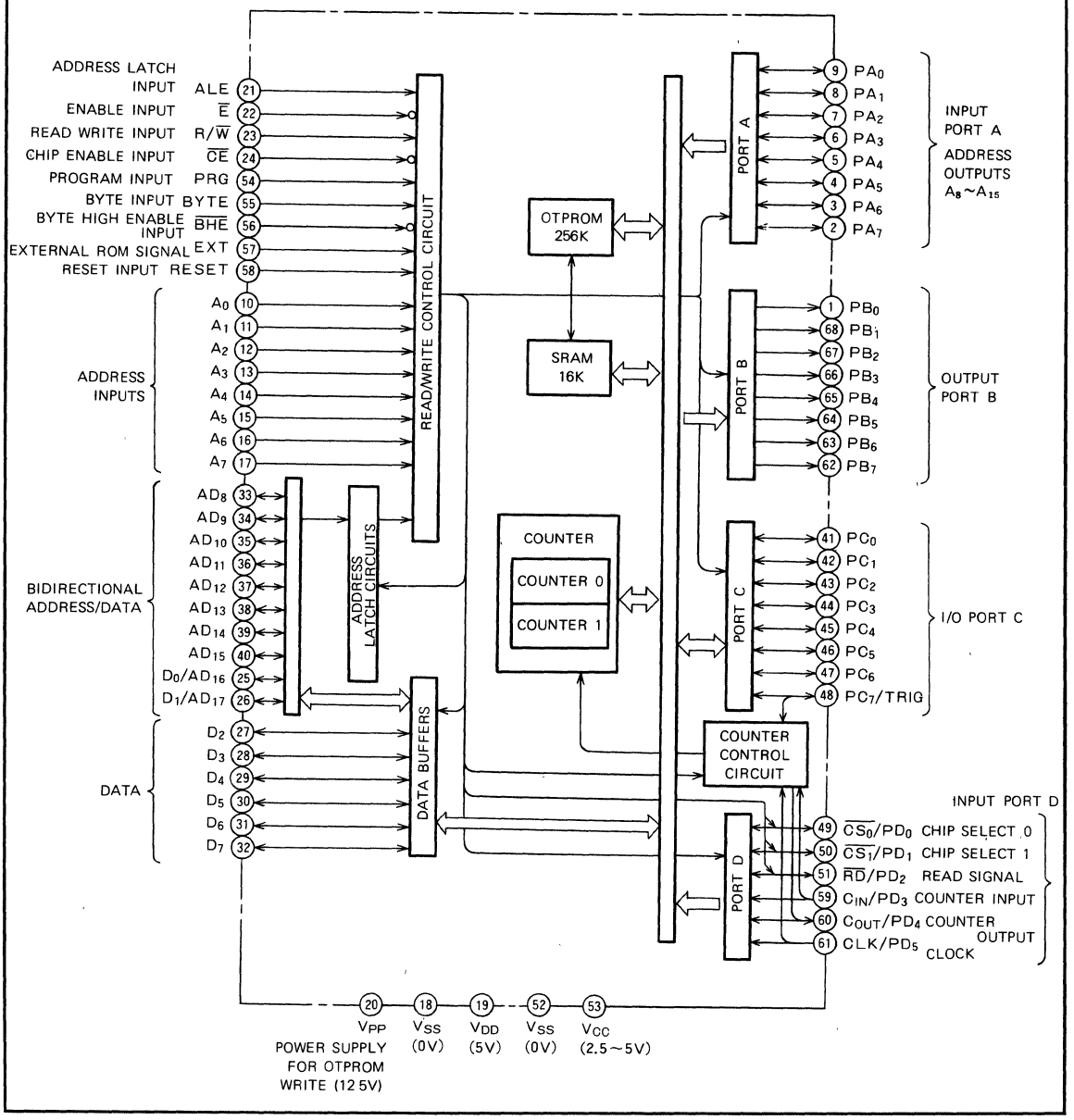
Symbol	Parameter	Limits			Unit
		Min.	Typ.	Max	
$t_{d(\phi-PWM_{OUT})}$	PWM_{OUT} output delay time			200	ns



TIMING DIAGRAM



BLOCK DIAGRAM



MITSUBISHI ELECTRIC

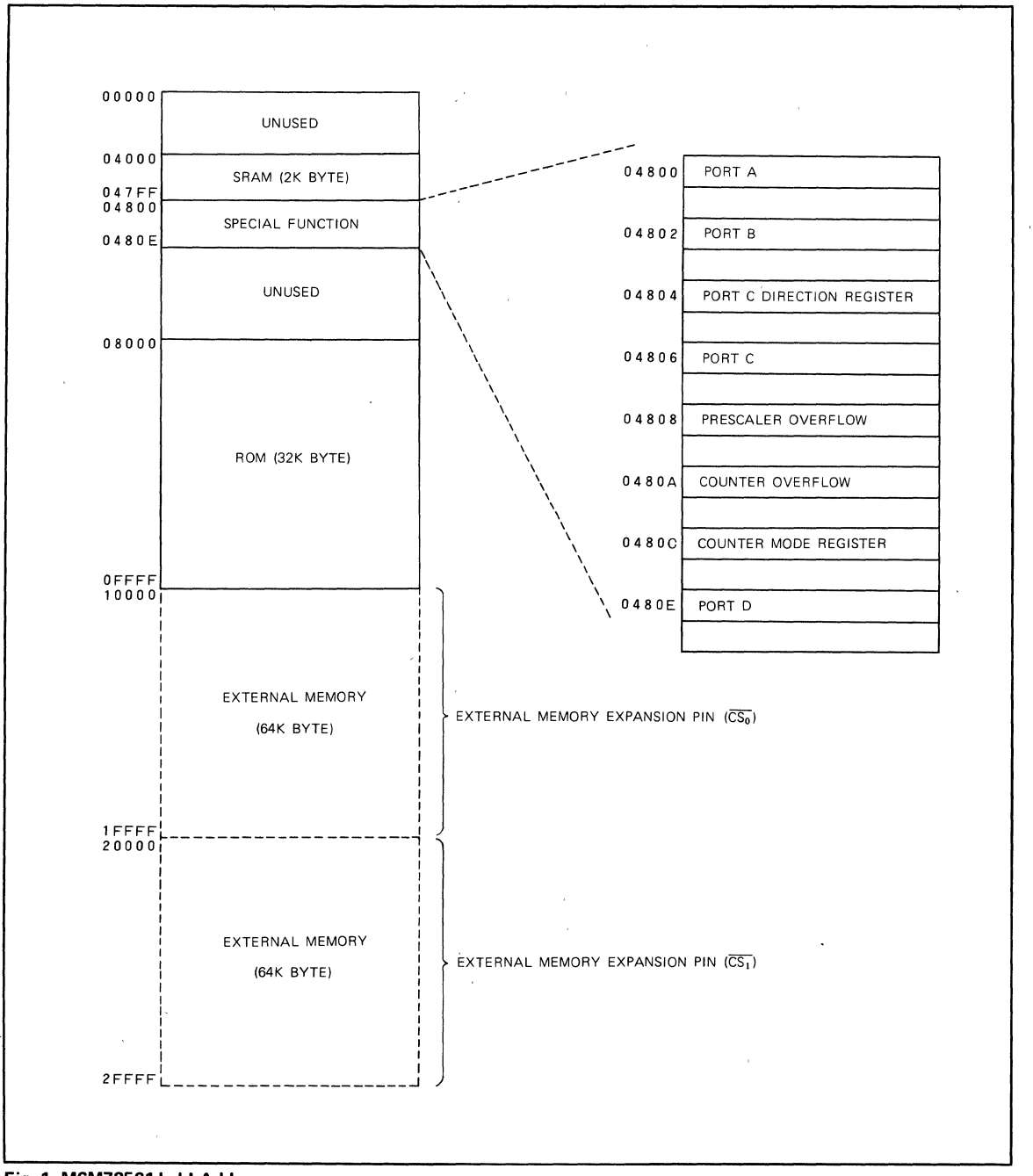


Fig. 1 M6M72561J, J-I Address area

OTP ROM Mode Table

(x is irrelevant)

Mode \ Pin name		ALE (21)	$\overline{E}$ (22)	R/$\overline{W}$ (23)	$\overline{CE}$ (24)	PRG (54)	BYTE (55)	$\overline{BHE}$ (56)	EXT (57)	RESET (58)	V_{DD} (19)	V_{PP} (20)	V_{CC} (53)	Data I/O		
														$D_0/A_{16}, D_1/A_{17}$	$D_2 \sim D_7$	$AD_8 \sim AD_{15}$
Normal read	8-bit mode	V_{IH} pulse	V_{IL}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	× Note 1	V_{IL}	×	5V	5V	5V	Address input Data output	Output	Address input Data floating
	16-bit mode Note 2	V_{IH} pulse	V_{IL}	V_{IH}	V_{IL}	V_{IL}	V_{IL}	V_{IL}	V_{IL}	×	5V	5V	5V	Address input Data output	Output	Address input Data output
Output disable		V_{IL}	V_{IH}	×	V_{IL}	V_{IL}	×	×	V_{IL}	×	5V	5V	5V	Data floating	Floating	Data floating
Power down mode		×	×	×	V_{IH}	×	×	×	×	V_{IH}	5V	5V	5V	Floating	Floating	Floating
Note 3 ROM Only mode	Read	V_{IH}	V_{IL}	V_{IH}	V_{IL}	V_{IH}	V_{IH}	×	V_{IL}	×	5V	5V	5V	Output	Output	Address input
	Output disable	V_{IH}	V_{IH}	V_{IH}	V_{IL}	V_{IH}	V_{IH}	×	V_{IL}	×	5V	5V	5V	Floating	Floating	Address input
	Power down	×	×	×	V_{IH}	×	×	×	×	V_{IH}	5V	5V	5V	Floating	Floating	Address input
	Program	V_{IH}	V_{IH}	V_{IH}	V_{IL} pulse	V_{IH}	V_{IH}	×	V_{IL}	×	6V	12.5V	6V	Data input	Input	Address input
	Program verify	V_{IH}	V_{IL}	V_{IH}	V_{IH}	V_{IH}	V_{IH}	×	V_{IL}	×	6V	12.5V	6V	Output	Output	Address input
External ROM mode		V_{IH} pulse	V_{IL}	V_{IH}	V_{IL}	V_{IL}	×	×	V_{IH}	×	5V	5V	5V	Address input Data floating	Floating	Address input Data floating

Note 1 If $\overline{BHE}$ is fixed to V_{IH}, Non-ALE is enabled, and if ALE is fixed to V_{IH}, address inputs of D_0/A_{16} and D_1/A_{17} are not needed

2 A_0 is fixed to V_{IL}.

3 If PRG is fixed to V_{IH}, ROM Only mode is enabled and address input is fixed to Non-ALE mode and address inputs of A_{15}, A_{16} and A_{17} are not needed.

SRAM Mode Table

(x is irrelevant)

Mode \ Pin name			ALE (21)	$\overline{E}$ (22)	R/$\overline{W}$ (23)	$\overline{CE}$ (24)	PRG (54)	BYTE (55)	$\overline{BHE}$ (56)	A_0 (10)	EXT (57)	RESET (58)	V_{DD} (19)	V_{PP} (20)	V_{CC} (53)	Data I/O		
																$D_0/A_{16}, D_1/A_{17}$	$D_2 \sim D_7$	$AD_8 \sim AD_{15}$
Read	8-bit mode		V_{IH} pulse	V_{IL}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	× Note 4	Address input	×	×	5V	5V	5V	Address input Data output	Output	Address input Data floating
	16-bit mode		V_{IH} pulse	V_{IL}	V_{IH}	V_{IL}	V_{IL}	V_{IL}	V_{IL}	V_{IL}	×	×	5V	5V	5V	Address input Data output	Output	Address input Data output
Write	8-bit mode		V_{IH} pulse	V_{IL}	V_{IL}	V_{IL}	V_{IL}	V_{IH}	× Note 4	Address input	×	×	5V	5V	5V	Address input Data output	Input	Address input Data floating
	16-bit mode	$D_0 \sim D_7$ access	V_{IH} pulse	V_{IL}	V_{IL}	V_{IL}	V_{IL}	V_{IL}	V_{IH}	V_{IL}	×	×	5V	5V	5V	Address input Data input	Input	Address input Data input
		$D_8 \sim D_{15}$ access	V_{IH} pulse	V_{IL}	V_{IL}	V_{IL}	V_{IL}	V_{IL}	×	V_{IH}	×	×	5V	5V	5V	Address input Data X	×	Address input Data input
		$D_0 \sim D_{15}$ access	V_{IH} pulse	V_{IL}	V_{IL}	V_{IL}	V_{IL}	V_{IL}	V_{IL}	V_{IL}	×	×	5V	5V	5V	Address input Data input	Input	Address input Data input
Power-down mode			×	×	×	V_{IH}	×	×	×	×	×	V_{IH}	5V	5V	5V	Floating	Floating	Floating

Note 4 If $\overline{BHE}$ is fixed to V_{IH}, Non-ALE mode is enabled, and if ALE is fixed to V_{IH}, address inputs of D_0/A_{16} and D_1/A_1 are not needed

(3) Counter

The input pulse from C_{IN} pin is counted by an 8-bit up-counter. The count value can be preset or output from data pins D_0 to D_7.

Table 3 shows the table for counter mode.

Initial setting for counter use

C_{IN}/PD_3, C_{OUT}/PD_4, CLK/PD_5 are common function pins related to a counter. Port PD should be specified to enable counter pin for counter use. This can be specified by writing "H" to data pin D_7 after the address $0480E_{16}$ is specified. Then clock pulse should be continuously input from CLK pin. The counter operation is synchronized with this clock pulse.

(Prescaler mode)

This mode is enabled by an 8-bit prescaler counter (C_0) to count the input pulse from C_{IN} pin and by an 8-bit counter (C_1) to count the overflow output pulse. Each counter returns to the preset count value by the overflow output pulse to count up again.

Table 4 Mode setting and counter preset

Parameter	Write address	Write data
Mode setting (counter mode and edge mode)	$0480C_{16}$	See Table 3.
C_0 counter preset	04808_{16}	Any preset value from 00 to FF
C_1 counter preset	$0480A_{16}$	Any preset value from 00 to FF

(Double count mode)

In this mode, both C_0 counter and C_1 counter count the input pulse from C_{IN} pin. C_0 counter is preset by the trigger input from PC_7/TRIG pin and C_1 counter is preset by the overflow output pulse of C_0 counter.

To enable the double count mode, PC_7/TRIG pin should be initialized as TRIG pin by specifying the address 04804_{16} and by writing "L" to D_7 pin.

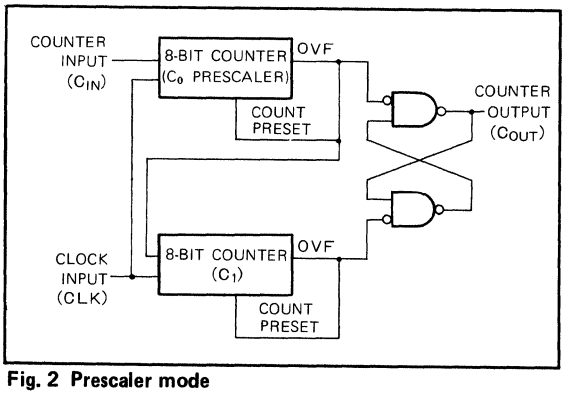


Fig. 2 Prescaler mode

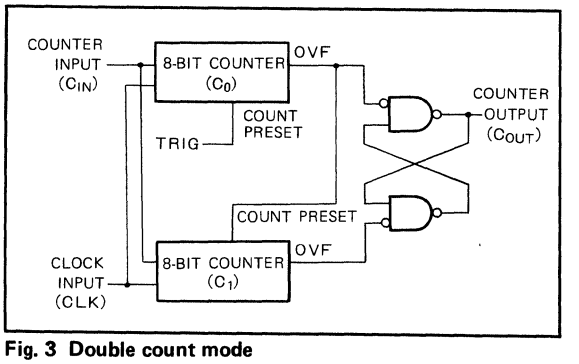


Fig. 3 Double count mode

Table 3 Counter mode table

Counter mode register ($0480C_{16}$)			Mode name		Counter configuration
D_7	D_5	D_4	Counter mode	Edge mode	
0	0	0	Prescaler mode	Count disable mode	See Fig 2. • Counter is preset by the OVF output timing
	0	1		Falling edge count mode	
	1	0		Rising edge count mode	
	1	1		Double edge count mode	
1	0	0	Double count mode	Count disable mode	See Fig 3 • Counter is preset by the external trigger timing
	0	1		Falling edge count mode	
	1	0		Rising edge count mode	
	1	1		Double edge count mode	

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Test conditions	Ratings	Unit
V_{I1}	All input or output voltage except V_{PP}	With respect to GND	−0.6～7	V
V_{I2}	V_{PP} supply voltage during programming		−0.6−14.0	V
V	Output voltage		−0.6～7	V
Topr	Operating temperature		−50～95	°C
Tstg	Storage temperature		−65～150	°C

CAPACITANCE

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
C_I	Input capacitance (Note 5)	Ta=25°C, f=1MHz, V_I=V_O=0V		4	8	pF
C_O	Output capacitance (Note 6)			4	12	pF

Note 5 A_0～A_7, ALE, $\overline{E}$, R/$\overline{W}$, $\overline{CE}$, PRG, BYTE, $\overline{BHE}$, EXT, RESET, C_{IN}/PD_3, CLK/PD_5

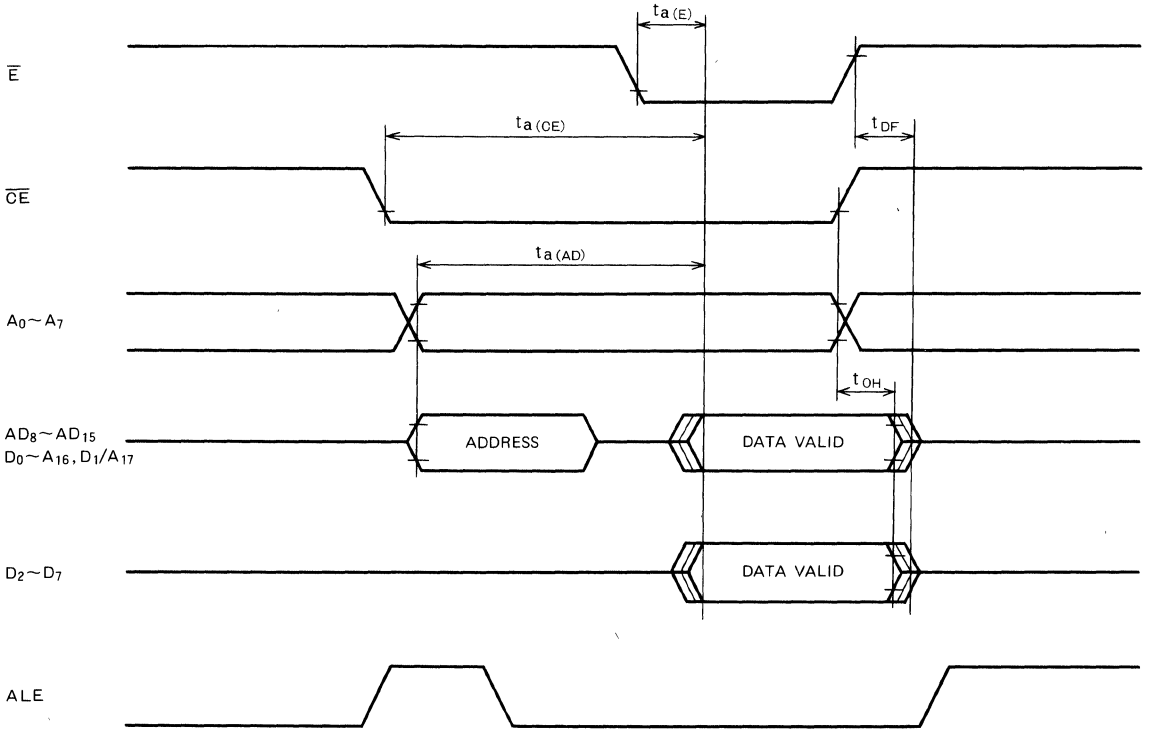
6 D_0/A_{16}, D_1/A_{17}, D_2～D_7, AD_8～AD_{15}, PA_0～PA_7, PB_0～PB_7, PC_0～PC_7
$\overline{CS_0}$/PD_0, $\overline{CS_1}$/PD_1, $\overline{RD}$/PD_2, C_{OUT}/PD_4

TIMING REQUIREMENTS ($Ta = 0 \sim 70°C^*$, $V_{DD} = V_{CC} = V_{PP} = 5V \pm 10\%$, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{su(A-L)}$	Address setup time before latch		30			ns
$t_{h(L-A)}$	Address hold time after latch		20			ns
$t_{w(L)}$	Latch pulse width		70			ns
$t_{w(W)}$	Write low-level pulse width		70			ns
$t_{su(A-W)}$	Address setup time before write		70			ns
$t_{su(D-W)}$	Data setup time before write		60			ns
$t_{w(E)}$	Write low level pulse width		70			ns
$t_{su(A-E)}$	Address setup time before write		60			ns
$t_{su(D-E)}$	Data setup time before write		60			ns
$t_{h(W-D)}$	Data hold time after write		10			ns
$t_{h(E-D)}$	Data hold time after write		10			ns
$t_{h(W-A)}$	Address hold time after write		20			ns
$t_{h(E-A)}$	Address hold time after write		20			ns
$t_{c(W-L)}$	Latch pulse rise after write		20			ns
$t_{c(E-L)}$	Latch pulse rise after write		20			ns
$t_{c(L-W)}$	Write pulse fall after latch		30			ns
$t_{c(L-E)}$	Write pulse fall after latch		30			ns
$t_{c(A-E)}$	Write pulse fall after address input		0			ns
$t_{dis(W)}$	Output disable after write				60	ns
$t_{c(E-A)}$	Address input after write		50			ns
$t_{c(CL-E)}$	E rise after CLK rise at counter-preset		15			ns
$t_{su(PT-E)}$	Port setup time before read		200			ns
$t_{h(E-PT)}$	Port hold time after read		20			ns
$t_{c(CLK)}$	Clock cycle time		250			ns
$t_{w(CLKH)}$	CLK input high-level pulse width		125			ns
$t_{w(CLKL)}$	CLK input low-level pulse width		125			ns
t_r	CLK input rise time				20	ns
t_f	CLK input fall time				20	ns
$t_{w(CINH)}$	Counter event input high-level pulse width		$4 \times t_{c(CLK)}$			ns
$t_{w(CINL)}$	Counter event input low-level pulse width		$4 \times t_{c(CLK)}$			ns
$t_{w(TRGH)}$	Trigger event input high-level pulse width		$4 \times t_{c(CLK)}$			ns
$t_{w(TRGL)}$	Trigger event input low-level pulse width		$4 \times t_{c(CLK)}$			ns

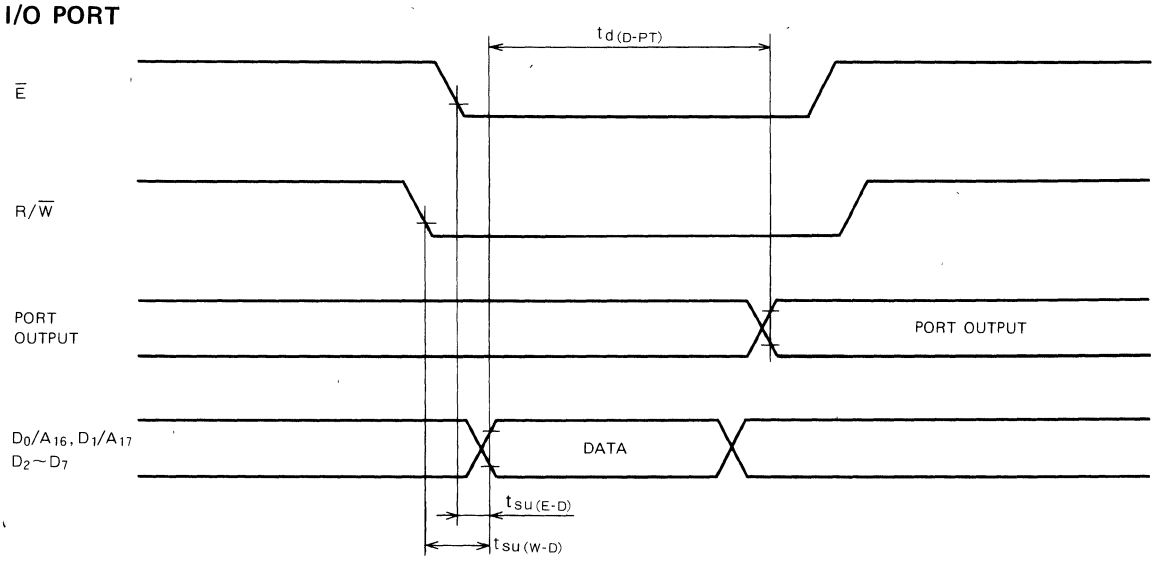
* $Ta = -40 \sim 85°C$ for M6M72561J-I

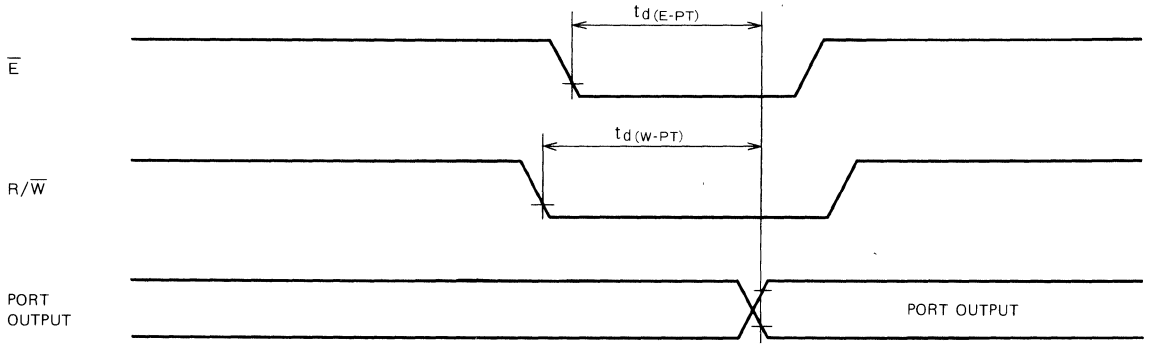
AC WAVEFORMS
READ OPERATION

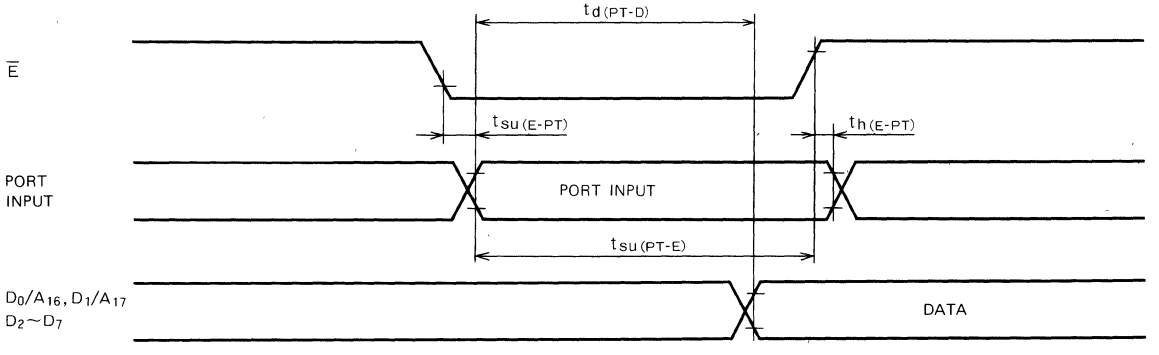


TIMING DIAGRAM

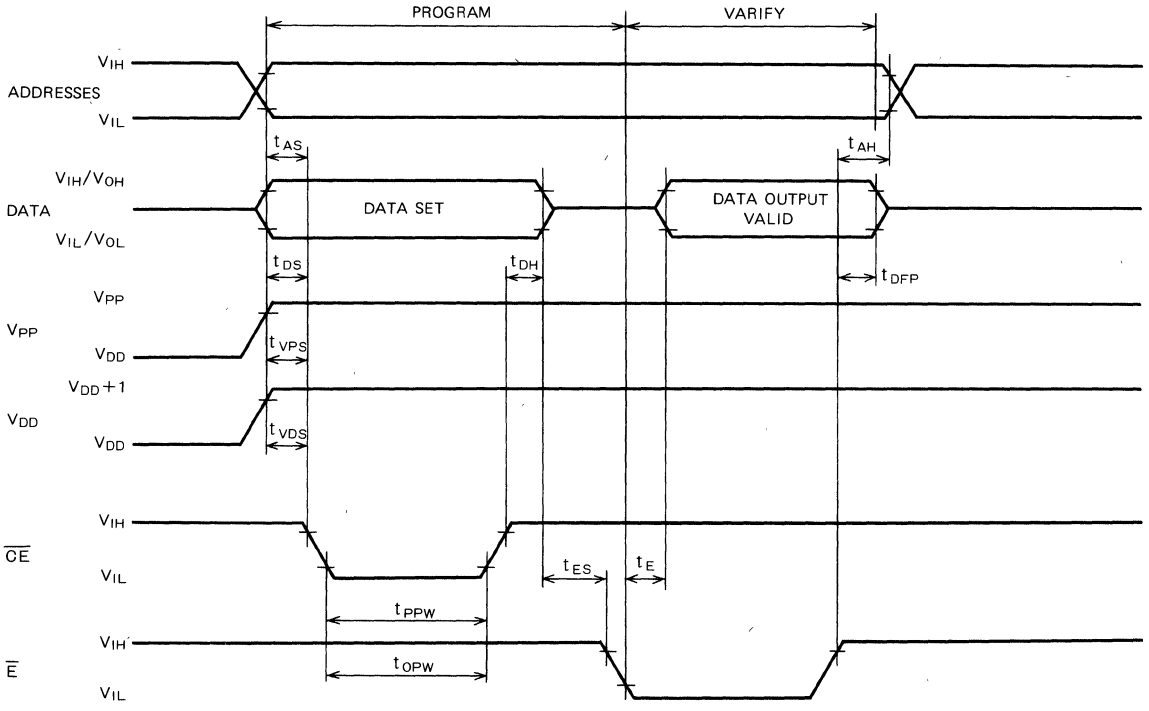
I/O PORT





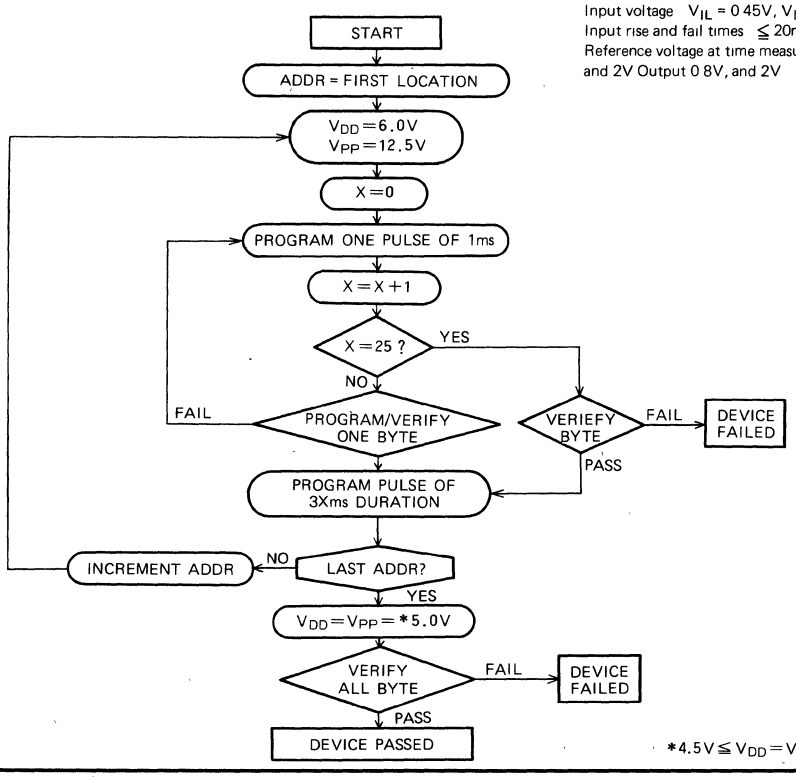


AC WAVEFORMS



FAST PROGRAMMING ALGORITHM FLOW CHART

Test conditions for A C. characteristics
Input voltage V_{IL} = 0 45V, V_{IH} = 2 4V
Input rise and fall times ≦ 20ns
Reference voltage at time measurement Input 0 8V and 2V Output 0 8V, and 2V



*4.5V≦ $V_{DD}=V_{PP}$ ≦5.5V

Table 1 Pseudo directives function list

Pseudo directives classification	Functions	Pseudo directives
Assembly control	• Declares data length	.INDEX .DATA
	• Declares data bank / direct page register value	.DT. DP
	• Conditional assemble	.IF (.ELSE) .ENDIF
	includes file	.INCLUDE
	• Equation	.EQU
	• Declares end of program	.END
	• Out puts messages	.ASSERT
	• Declares assemble error	ERROR
Address control	• Declares address	.ORG
	• Allocates RAM area	.BLKB .BLKW .BLKA .BLKD
	• Defines data	BYTE .WORD .ADDR .DWORD
	• Corrects location alignment	.EVEN
Linkage control	• Specifies section name	.SECTION
	• Specifies global lavel name	.DPEXT .DTEXT .EXT .PUB
	• Specifies linkage file name	.OBJ .LIB
	• Specifies version	VER
Listing control	• Specifies new page and title	.PAGE
	• Specifies listing format (columns and rows counts)	.COL .LINE
	• Specifies listing output / suppression	.LIST .NLIST
	• Specifies macrogeneration line listing output / suppression	.LISTM .NLISTM
C language source level debugging support	• Outputs begining of line	CLINE
	• Specifies function start / end	.FUNC .ENDFUNC
	• Outputs language type	.LANGUAGE
	• Setting pointer length	.POINTER
	• Specifies source file name	.SOURCE
Subspriction	• Declares program name	.PROGNAME
	• Declares area name	IO .ENDIO RAM ENDRAM
	• Declares module name	.PROCMAIN .PROCSUB .PROCINT .ENDPROC

Table 2 Macro instructions function list

Macro instructions classification	Functions	Macro instructions
System macro instructions	Repeats the processing operation in accordance with the number of arguments contained in the operand	.REPEATI ~.ENDM
	Repeats the processing operation in accordance with the number of characters presented by the operand as arguments	.REPEATC ~.ENDM
	Repeats the processing operation a specific number of times	.REPEAT ~.ENDM
User macro instructions	Defines the macro instructions.	.MACRO ~ ENDM
	Forces the termination of macro generation	.EXITM
	Changes the labels used in the macro definitions into intra-macro local labels	LOCAL

CONSTRUCTION

The following shows the software construction of RASM77.

Relocatable macro assembler RASM77	Assembles the assembly language source files for series MELSP 7700, and generates the relocatable object files.
Linkage editor LINK77	Generates a machine language file in hexadecimal format from the plural relocatable files generated by RASM77.
Librarian LIB77	Generates a library file from the relocatable files generated by RASM77
Cross referencer CRF77	Generates a cross refrence file from the assembly language source files for series MELPS 7700 The cross refrence file shows the points defined, and refrenced the labels and symbols.
File converter HEXTOS2	Generates a S format machine language file from a hexadecimal format machine language file generated LINK77

ENVIRONMENT

RASM77 operates in the following OS environments.

- MS-DOS
- SUN-3/SUN OS
- VAX/ULTRIX
- HP9000/HP-UX

Note 1 : MS-DOS is a registered trademark of Microsoft Corp.

Note 2 : SUN-3 and SUN OS are registered trademarks of Sun Microsystems Corp.

Note 3 : VAX and ULTRIX are registered trademarks of Digital Equipment Corp.

Note 4 : HP9000 and HP-UX are registered trademarks of HEWLETT PACKARD Corp.

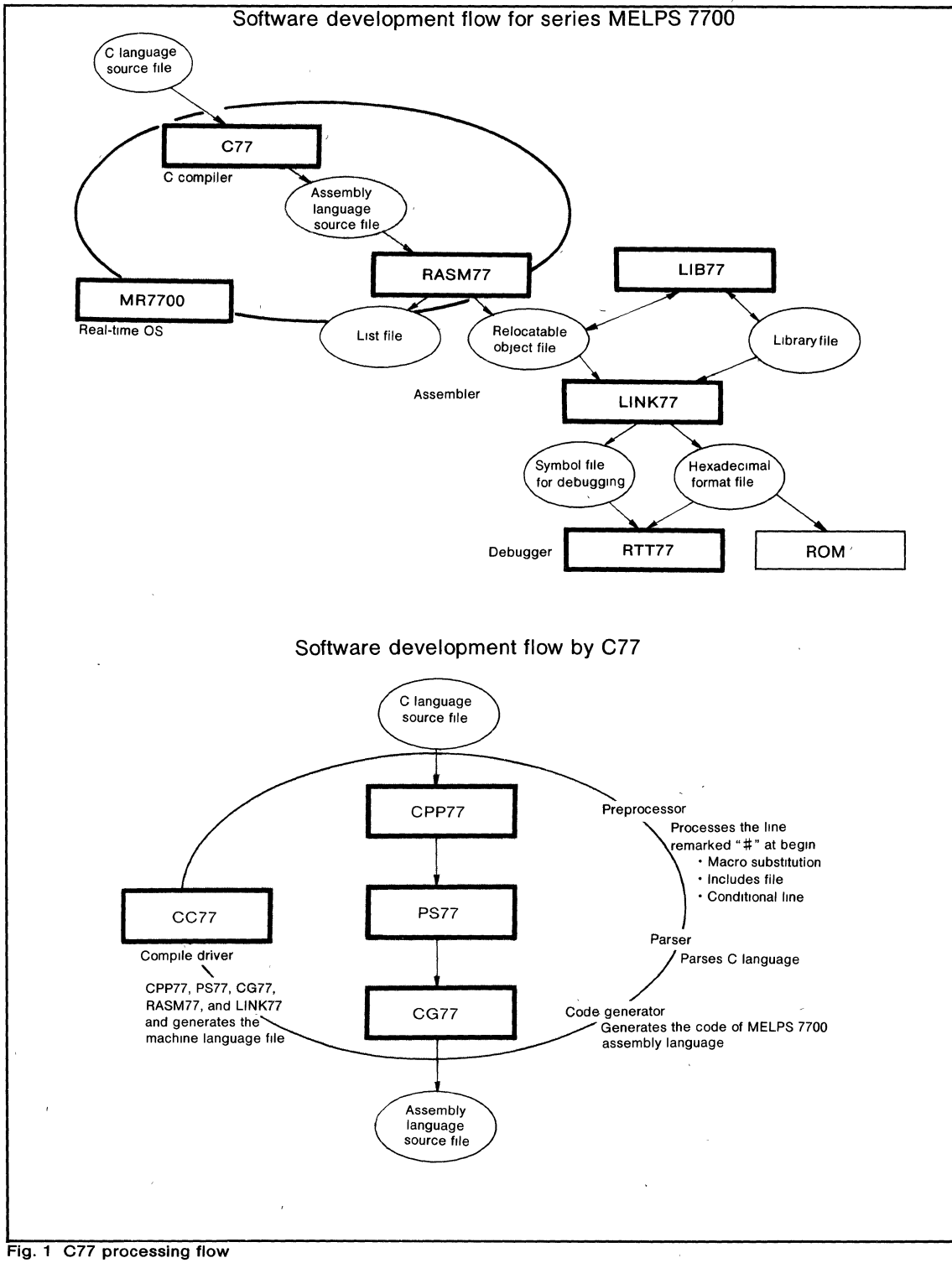


Fig. 1 C77 processing flow

DESCRIPTION

SDB77 is the series MELPS 7700 debugger control software prepared for the PC4816 system. SDB77 operates on a 16-bit personal computer that supports the MS-DOS environment, and controls the PC4816 system through the RS-232C serial interface.

DISTINCTIVE FEATURES

SDB77 supports the following features:

- Support of the runtime debugging function that refers memory during execution of a program
- Support of various break point setting functions
- Support of various real-time trace control functions
- Support of the execution time measurement function
- Support of the C language source line display function
- Apply to every MCU in the series MELPS 7700 with the use of data files that contain particular data of individual MCU
- The initialization file automatically sets debugging start conditions.
- The batch command enables continuors execution of debugging commands.
- Debugging history can be recorded in the log file.
- Support of the debugging information definition file that automatically sets break points and trace points.
- Support of the online manual

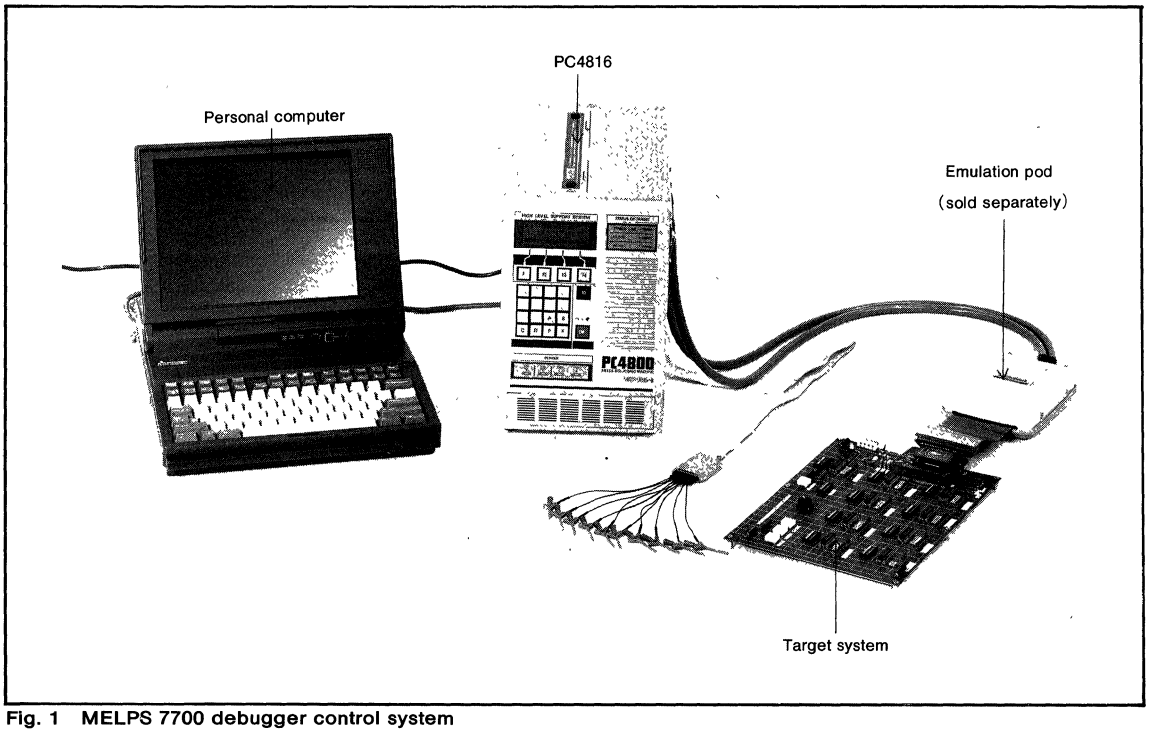


Fig. 1 MELPS 7700 debugger control system

DESCRIPTION

MR7700 is the real-time operating system developed for the 16-bit single-chip microcomputer MELPS 7700 series. MR7700KIT is the tool kit required to develop application software incorporating MR7700.

DISTINCTIVE FEATURES

- Based on with μITRON specifications
- Real-time operating system intended for industrial incorporation systems.
- High speed processing
- Minimam size of OS kernel by automatically selecting only necessary modules
- Application programs can be developed in C language by using C compiler C77.

FUNCTIONS

(1) Outline specifications of MR7700

Table 1 shows outline specifications of MR7700.

Table 1 Outline specifications of MR7700

Item	Specifications
Target MCU	Series MELPS 7700
Maximum number of tasks	124
Priority number	1~63
Number of system calls	51
OS nucleus code size	About 0 9K bytes to 5 8K bytes (standard model)
OS nucleus data size	Minimam 14 bytes and 9 bytes increased by each task
OS nucleus description language	Assembly
Task switching time	About 30μs max
Maximum interrupt inhibited time	About 11 μs to 24μs

The OS nucleus data size does not include stack size. Because the 16 bytes of task context (for standard model) is allocated on the stack, contexts of individual tasks are not included in the OS data size.

Time data in the table is measured by M37700M2 (16MHz) at 0 wait memory access and 16-bit width. The task switching time and maximum interrupt inhibited time are for a wup-tsk system call at priority number 1.

(2) Kernel configuration for MR7700

The MR7700 kernel consists of two module groups: One is the requiste module group that performs real-time processing and multi-tasking, the other is the expansion function module group from which modules can be selected according to user applications.

Figure 1 shows the module configuration of the MR7700 kernel. The application program is prepared by the user, and is composed of tasks, an interrupt processing handler, a cycle execution handler, and an alarm handler.

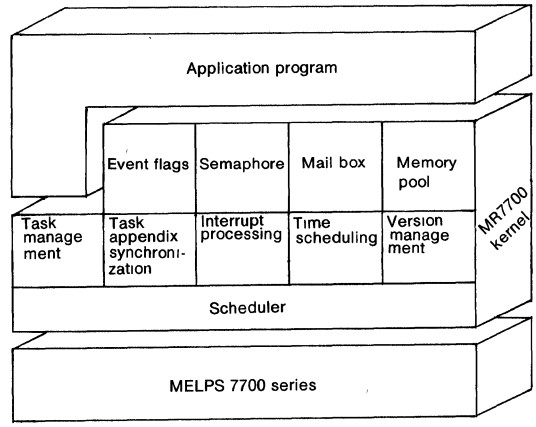


Fig. 1 MR7700 configuration

MR7700SRC

MELPS 7700 REAL-TIME OS SOURCE KIT

DESCRIPTION

MR7700 is the real-time opearating system developed for the 16-bit one-chip microcomputer MELPS 7700 series.
MR7700SRC is the mass production contract that authorries mass production of products incroporating MR7700.

CONFIGURATION

MR7700SRC consists of the followings elements:

- MR7700 source
- Mass production right (to incorporate MR7700 into an unlimited number of products of a single model)

CONTRACT OUTLIVE

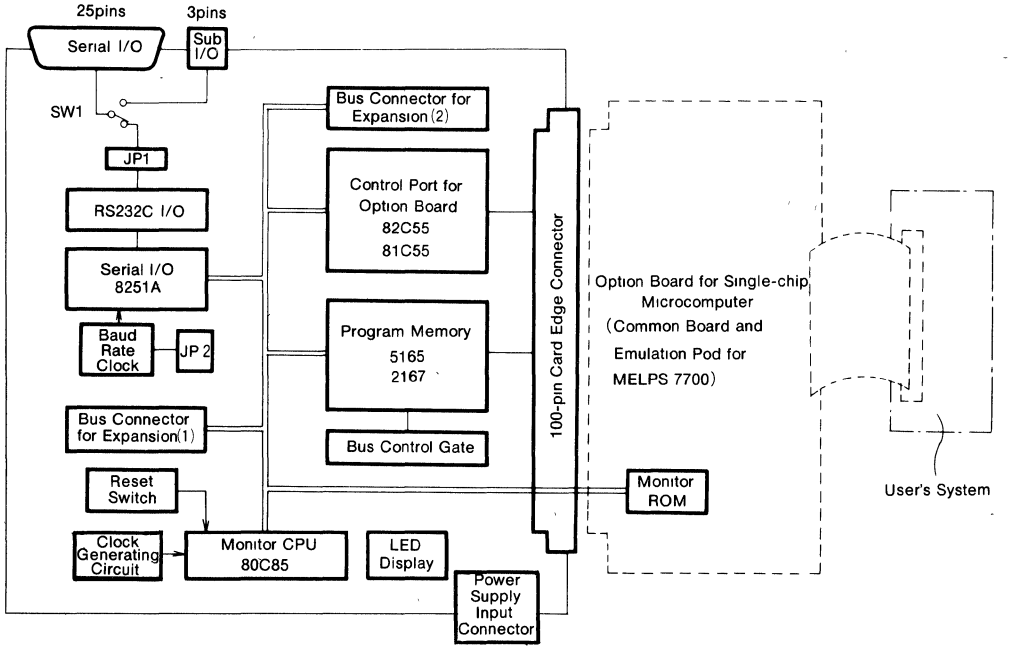
MR7700SRC is the license to manufacture a large quantity of products that incorporate MR7700. This license authorizes production of an unlimited number of products by five specified computer systems and five engineers in a single section. No royalties are charged nor report of output is required. To carry out development by employing more engineers or more systems than specified, the user must acquire the right of use.
MR7700SRC guarantees maintence (including software version up) of MR7700 for two years without payment.

NOTES

Purchase of MR7700KIT (development kit) is the prerequiste to close contract MR7700SRC.

- MELPS is a registered trademark of Mitsubishi Electric Corp.

PC4000E BLOCK DIAGRAM



MITSUBISHI ELECTRIC

PC4816

MELPS 7700 DEBUGGING MACHINE

DESCRIPTION

PC4816 is the high performance debugging machine for Mitsubishi original 16-bit microcomputers.

PC4816 executes debugging with a help of emulation pods connected to it. Various emulation pods are prepared for individual microcomputers and provided separately.

PC4816 can be connected to a host machine with a personal computer (hereafter, PC), and can be controlled by control software on the host machine.

Because PC4816 supports simple I/O functions, it can be used as a stand-alone unit after down-load the user's program.

DISTINCTIVE FEATURES

- Runtime debugging function is supported.
- Program evaluation function is supported.
- Usable as a stand-alone unit.
- Evaluation functions can be enhanced by adding boards.
- Compact design requires less space on a desk.

APPLICATIONS

For development of hardware and software for systems using MELPS 7700, and for program debugging.

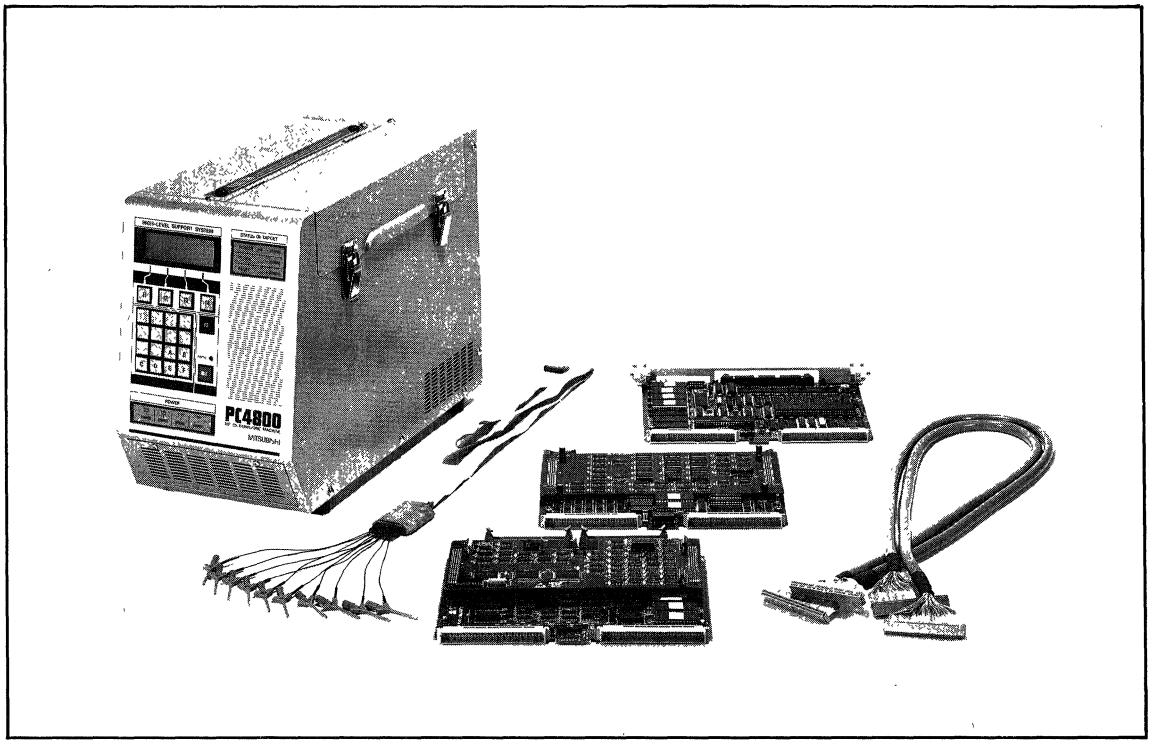


Fig. 1 The exterior of PC4816

FUNCTIONS

PC4816 consists of the following three kind products:

- PC4800

This is the debugger that incorporates the power unit, interface with the host machine, key entry unit, LCD display unit, monitor CPU board, emulation board, and container rack.

- M37700T-CNT

This is the emulation control board that consists of the MCU execution control circuit and program memory.

- PCA4800R

This is the debugging function board that consists of the real-time tracer and coverage function circuit.

PC4816 is used combined with emulation pods that are prepared for individual MCU and provided separately.

PC4800 is connected to the host machine via a serial circuit (RS-232C). PC4816 is connected to the emulation board and the debugging function board via two 96-pin connectors.

When PC4816 is turned on, the monitor CPU is activated and the monitor programs stored in the EPROM on various boards in the rack are started.

When the monitor programs are started, the start message is displayed on the LCD display unit on PC4816. Then the system waits for control instructions from debugger control software SDB77 on the host machine.

When the debugger control software is started, the host machine requests details (board configuration, monitor program version number, etc.) to be reported via the serial circuit. Acceptance of various debugging commands is enabled after PC4816 returns the details correctly.

- Serial circuits : Complying with RS-232C, two circuits
- Parallel circuit : Complying with GP-IB
- Power supply : Power units applicable to 115V and 220V systems
- Board rack : Five boards can be accommodated.

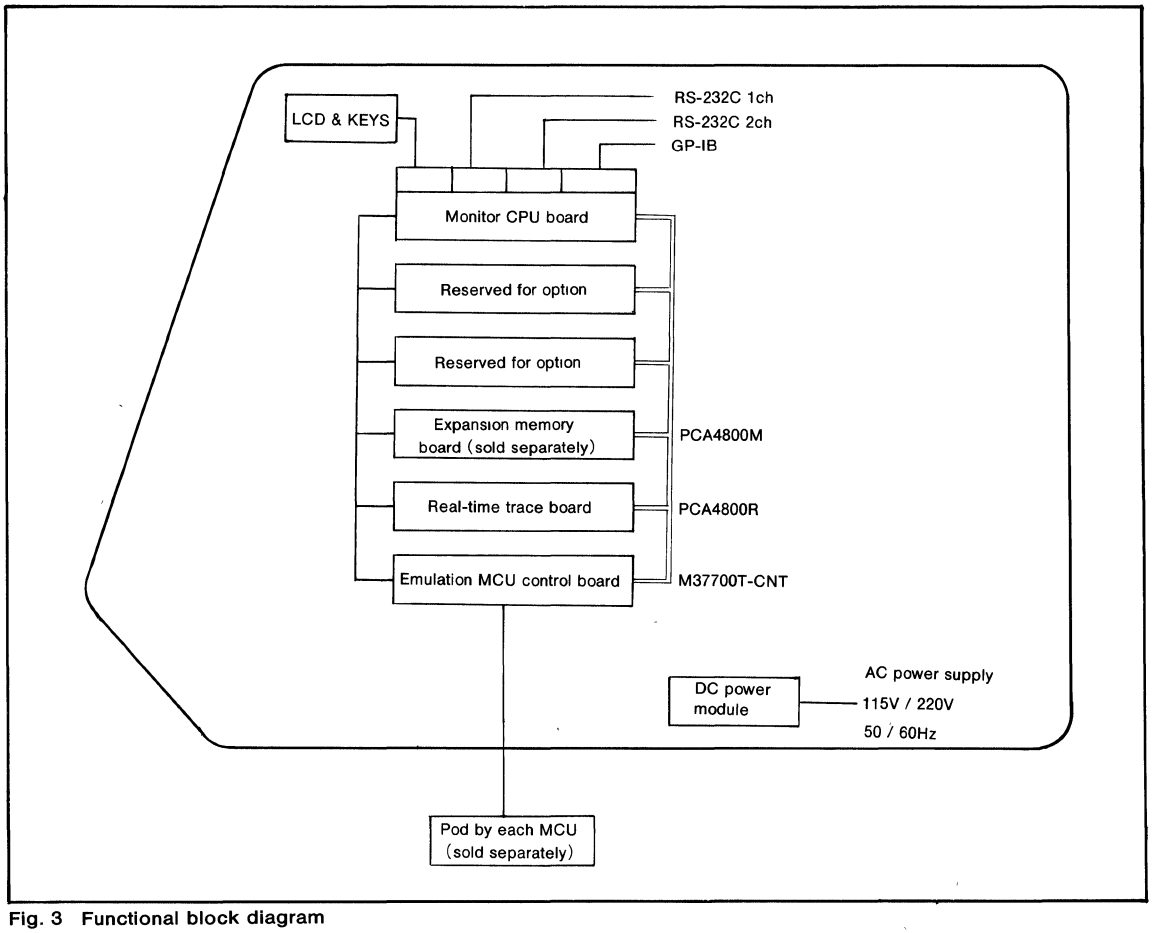


Fig. 3 Functional block diagram

PCA4800M

1M BYTE MEMORY EXPANSION BOARD for PC4816

DESCRIPTION

PCA4800M is the emulation memory expansion board that is used incorporated in PC4816, the debugging machine for MELPS 7700.

1M bytes of high speed SRAM, which can be distributed over areas of 16M bytes in units of 4K bytes in mounted.

DISTINCTIVE FEATURES

- Applicable to 8 or 16-bit wide memory buses.

SPECIFICATIONS

Item	Contents
Debugger	PC4816
Corresponding MCU	MELPS 7700 series
Corresponding control board	M37700T-CNT
Corresponding bus width	8, 16-bit
Memory capacity	IMB
Mapping units	4KB
Supply voltage	5 [V] ± 5 [%]

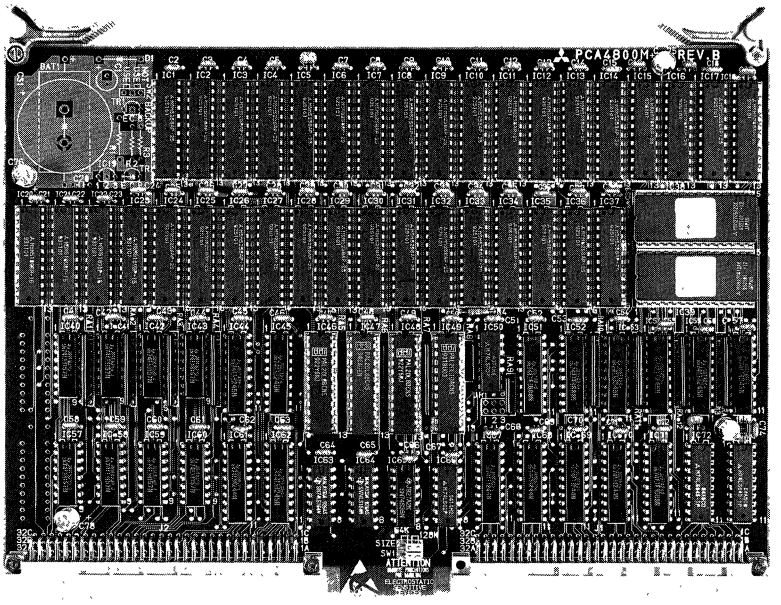


Fig. 1 The exterior of PCA4800M

M37700T-POD

MELPS 7700 EMULATION POD(SINGLE-CHIP MODE,CORRESPOND to PC4000E)

DESCRIPTION

The M37700T-POD is an emulation pod constructs the option board for single-chip mode of MELPS 7700 emulation common board M37700T-RTT.
M37700T-POD is used with M37700T-RTT and PC4000E.

DISTINCTIVE FEATURES

- Debugging functions can be used to connect with PC4000E, M37700T-RTT, and this pod.
- It can be corresponded to a system with 16MHz oscillator.
- The M37700T-POD can be connected by an attached flat cable with the users system.
- The M37700T-POD can be use an internal clock or an external oscillator.

STRUCTURE

The M37700T-POD consists of the following hardwares :

1. Evaluation MCU (M37700SAFP) and related circuits.
2. Gatearray for port emulation and related circuit.

FUNCTIONS

To connect the M37700T-RTT, the M37700T-POD constructs the opion board for single-chip mode of M37700M2-XXXFP or M37700M2AXXXFP.
The PC4000E debugging machine is controlled by the monitor program ROM on the M37700T-RTT, and the evaluation MCU (M37700SAFP) mounted on the M37700T-POD executes the program stored in the program RAM on the M37700T-RTT.
Single-step execution, execution with break points, and reading of internal status with the evaluation MCU halted are performed under the control of the monitor CPU on the PC4000E.

SPECIFICATIONS

Item	Specification
Corresponded microcomputer	M37700M2-XXXFP M37700M2AXXXFP M37700M4-XXXFP M37700M4AXXXFP M37701M2-XXXSP M37701M2AXXXSP M37701M4-XXXSP M37701M4AXXXSP M37704M2-XXXFP* M37704M2AXXXFP* M37705M2-XXXSP* M37705M2AXXXSP*
Corresponded mode	Single-chip mode
Clock frequency	16MHz (on board) External oscillator 16MHz (max)
Common emulation board	M37700T-RTT
Power supply	PC4000P usable
Connection with user system	Connected by the attached cable

*Be necessary to order exchanging the evaluation MCU

M37700T-HPD

MELPS 7700 HIGH LEVEL EMULATION POD(CORRESPOND to PC4816)

DESCRIPTION

The M37700T-HPD is a high level pod for MELPS 7700 is an emulation pod which enables debugging system PC4816. It can be used as an emulator corresponding to M37700 and M37701 by conection with M37700T-CNT built-in PC4816.

DISTINCTIVE FEATURES

- The electrical characteristics of the direct ports is excellent because of the distance between the emulation MCU and the users system is short.
- The whole processor mode that single-chip mode, microprocessor mode and memory expansion mode can be corresponded alone.
- Compact pod

STRUCTURE

M37700T-HPD consists of the following hardware.

(1) Evaluation MCU (M37700SAFP-A)
(2) Clock oscillation compact board
(3) Port emulation circuit
(4) I/O buffer

SPECIFICATIONS

Item	Contents
Debugger	PC4816
Corresponding MCU	M37700, M37701
Clock frequency	16MHz
Corresponding control board	M37700T-CNT
Corresponding MCU mode	(1) Single-chip mode (2) Memory expansion mode of 8 and 16-bit bus width (3) Microprosessor mode of 8 and 16-bit bus width
Supply voltage	5 [V] ± 5 [%]

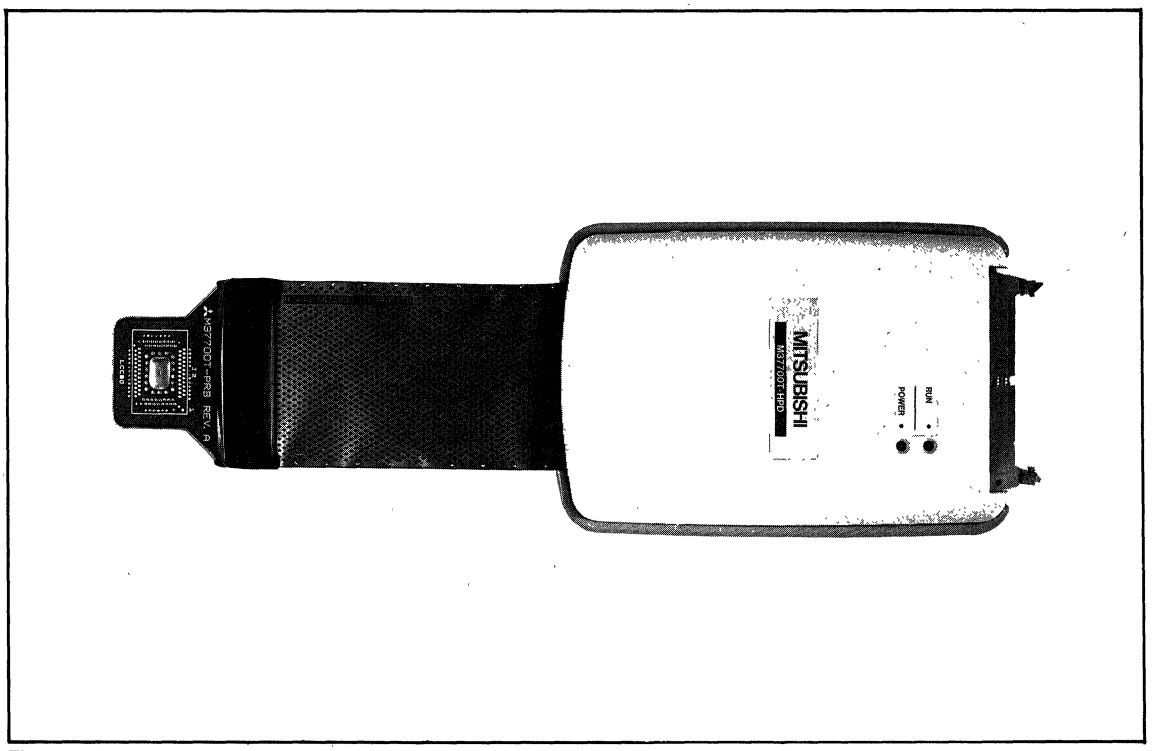


Fig. 1 The exterior of M37700T-HPD

M37704T-HPD

MELPS 7700 HIGH LEVEL EMULATION POD(CORRESPOND to PC4816)

DESCRIPTION

The M37704T-HPD is a high level pod for MELPS 7700 is an emulation pod which enables debugging system PC4816. It can be used as an emulator corresponding to M37704 and M37705 by conection with M37700T-CNT built-in PC4816.

DISTINCTIVE FEATURES

- The electrical characteristics of the direct ports is excellent because of the distance between the emulation MCU and the users system is short.
- The whole processor mode that single-chip mode, microprocessor mode and memory expansion mode can be corresponded alone.
- Compact pod

STRUCTURE

M37704T-HPD consists of the following hardware.

(1) Evaluation MCU (M37705ASP)
(2) Clock oscillation compact board
(3) Port emulation circuit
(4) I/O buffer

SPECIFICATIONS

Item	Contents
Debugger	PC4816
Corresponding MCU	M37704, M37705
Clock frequency	16MHz
Corresponding control board	M37700T-CNT
Corresponding MCU mode	(1) Single-chip mode (2) Memory expansion mode of 8 and 16-bit bus width (3) Microprosessor mode of 8 and 16-bit bus width
Supply voltage	5 [V] ± 5 [%]

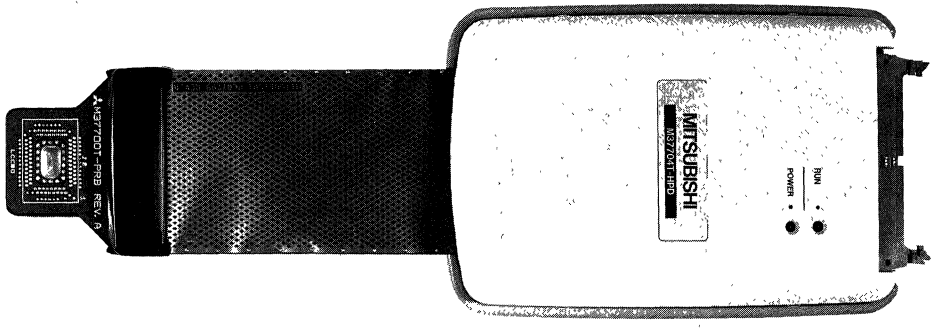


Fig. 1 The exterior of M37704T-HPD

MITSUBISHI ELECTRIC

M37795T-HPD

PRELIMINARY

MELPS 7700 HIGH LEVEL EMULATION POD(CORRESPOND to PC4816)

DESCRIPTION

The M37795T-HPD is a high level pod for MELPS 7700 is an emulation pod which enables debugging system PC4816. It can be used as an emulator corresponding to M37795SJ by conection with M37795T-CNT built-in PC4816.

DISTINCTIVE FEATURES

- The electrical characteristics of the direct ports is excellent because of the distance between the emulation MCU and the users system is short.
- Compact pod

STRUCTURE

M37795T-HPD consists of the following hardware.

(1) Evaluation MCU (M37795SJ)
(2) Clock oscillation compact board
(3) Port emulation circuit
(4) I/O buffer

SPECIFICATIONS

Item	Contents
Debugger	PC4816
Corresponding MCU	M37795SJ
Clock frequency	16MHz
Corresponding control board	M37700T-CNT
Corresponding MCU mode	Microprosessor mode of 8 and 16-bit bus width
Supply voltage	5 [V] ± 5 [%]

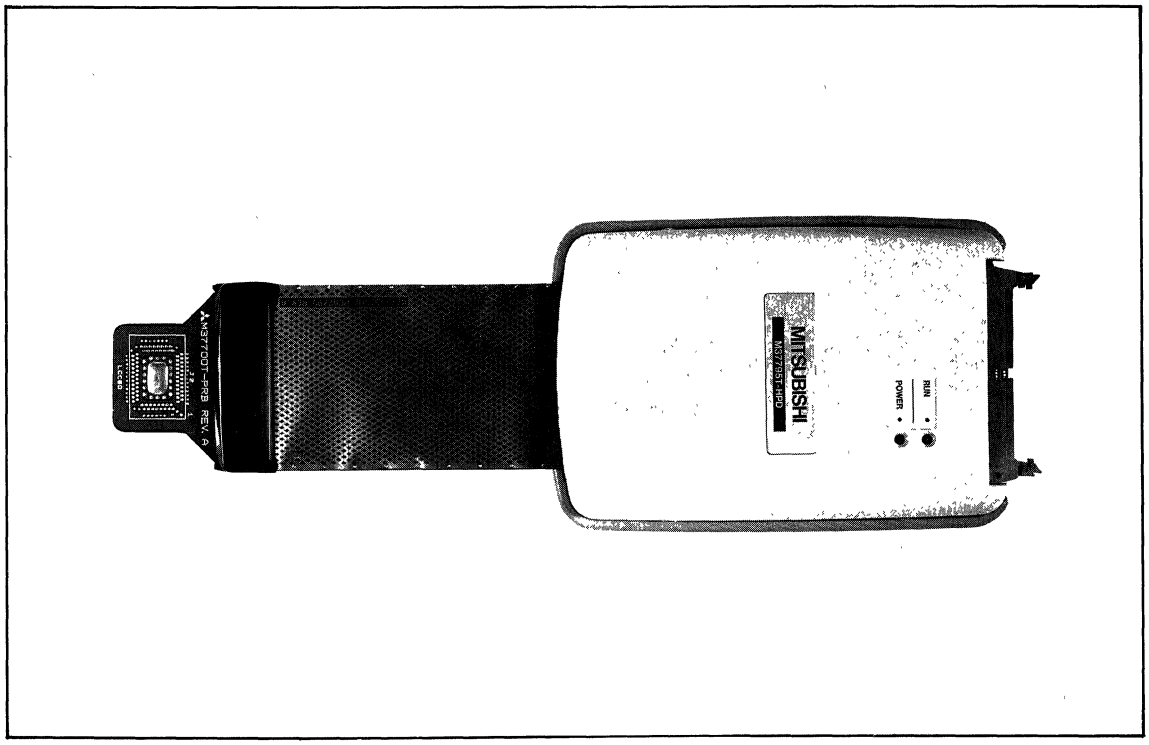


Fig. 1 The exterior of M37795T-HPD

PCA4707

PROGRAM WRITING ADAPTER for ONE TIME PROM VERSION MICROCOMPUTERS

DESCRIPTION

The PCA4707 is an adapter writing the program into the EPROM version microcomputer M37700E2-XXXFP, M37700E2AXXXFP, M37700E4-XXXFP, or M37700E4AXXXFP.
It can be used the PROM writer (correspond to M5M27C2-56K) on the market to write the program through the PCA4707.

STRUCTURE

The PCA4707 is consisted of two printed circuit boards named PCA4707 and PCA4701B.

PCA4707 The EPROM version microcomputer is mounted on this board.
The PCA4707 has a 80-pin IC socket to mount the EPROM version microcomputer, and a dual in line socket to connect with the PCA4701B.

PCA4701B The PCA4701B has two dual in line connecter. One of them is used connecting with the PROM writer, and anoter one is used connecting with the PCA4707.

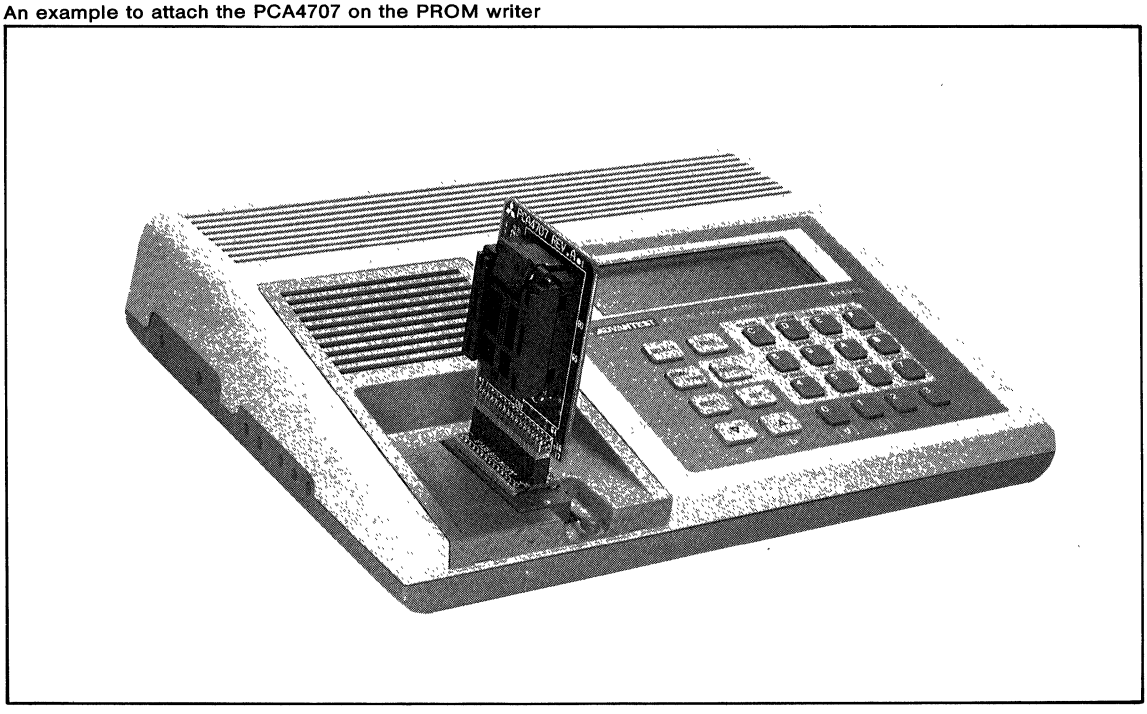
NOTE

Some kind of PROM writer cannot write the program on the EPROM version microcomputer through the PCA4707.

SPECIFICATIONS

Item	Specification
Correspond microcomputer	M37700E2-XXXFP M37700E2AXXXFP M37700E4-XXXFP M37700E4AXXXFP
Clock frequency	8MHz (ceramic oscillator on the PCA4707)
Power supply	Supplied from PROM writer
Specification of PROM writer	(1) Be able to write to M5M27C256K, or the other EPROM equal (2) Be able to write, to compare check, and erace check with designating the area (3) Be able to supply the current about 10mA from the source (5V line) to external.

An example to attach the PCA4707 on the PROM writer



MITSUBISHI ELECTRIC

PCA4709

PROGRAM WRITING ADAPTER for PROM VERSION MICROCOMPUTERS

DESCRIPTION

The PCA4709 is an adapter writing the program into the EPROM version microcomputer M37701E2-XXXSP, M37701E2AXXXSP, M37701E4-XXXSP, M37701E4AXXXSP, M37705E2-XXXSP, M37705E2AXXXSP, M37705E2SS, or M37705E2ASS.

It can be used the PROM writer (correspond to M5M27C2-56K) on the market to write the program through the PCA4709.

STRUCTURE

The PCA4709 is consisted of two printed circuit boards named PCA4709 and PCA4740B.

PCA4709 The EPROM version microcomputer is mounted on this board.
The PCA4709 has a 64-pin shrink DIP type IC socket to mount the EPROM version microcomputer, and a dual in line socket to connect with the PCA4740B.

PCA4740B The PCA4740B has two dual in line connecter. One of them is used connecting with the PROM writer, and anoter one is used connecting with the PCA4709.

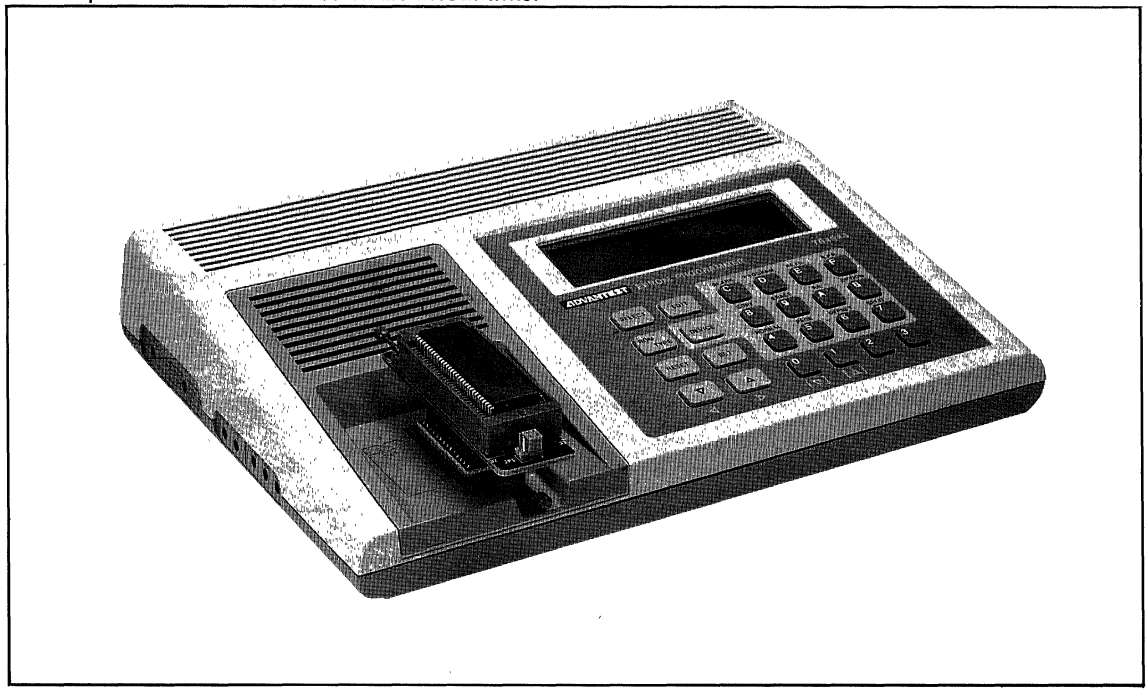
NOTE

Some kind of PROM writer cannot write the program on the EPROM version microcomputer through the PCA4709.

SPECIFICATIONS

Item	Specification
Correspond microcomputer	M37701E2-XXXSP, M37701E2AXXXSP M37701E4-XXXSP, M37701E4AXXXSP M37705E2-XXXSP, M37705E2AXXXSP M37705E2SS, M37705E2ASS
Clock frequency	8MHz (ceramic oscillator on the PCA4709)
Power supply	Supplied from PROM writer
Specification of PROM writer	(1) Be able to write to M5M27C256K, or the other EPROM equal (2) Be able to write, to compare check, and erace check with designating the area. (3) Be able to supply the current about 10mA from the source (5V line) to external.

An example to attach the PCA4709 on the PROM writer



MITSUBISHI ELECTRIC

PCA4780

PITCH CONVERTER for 80-PIN FLAT PACKAGE

DESCRIPTION

The PCA4780 is a pitch converter for 80-pin flat package connect the printed circuit pattern of the system mounted the 80-pin flat package single-chip microcomputer and that option board or evaluation board.

DISTINCTIVE FEATURES

- The PCA4780 can be used for an in circuit emulation probe to debug the system mounted the 80-pin flat package microcomputer.
- The PCA4780 can be converted the 40-pin, 1 inch dual in line pitch to 80-pin 0.8mm flat package pitch.

CONFIGURATION

The PCA4780 is structured following two parts

(1) FLAT80 : The printed circuit flexible board
The FLAT80 has the 80-pin (0.8mm pitch) flat package pattern to attach the CUBE80 and two 40-pin connecters to connect the attachment cables of the option board or evaluation board, and these are connected one-to-one.

(2) CUBE80 : An electrode for flat package pattern
The CUBE80 has two electrodes formed same as the 80-pin (0.8mm pitch) flat package microcomputer.
One of that is connected with the user system, and another is connected with the FLAT80.
It is necessary soldering to connect there.

ATTACHMENT

PCA4780 is attached the following process:

(1) One of the electrodes of the CUBE80 is soldered with the user system, and the another is soldered with the FLAT80.

(2) The FLAT80 is connected to the option board or the evaluation board by the attachment cables.

(3) The GND electrode of the FLAT80 is connected with the GND electrode of the option board or the evaluation board.

SPECIFICATIONS

Item	Specification
Correspond package	80-pin flat package (pins interval 0.8mm) TYPE 80P6
Components	CUBE80 : Electrodes for the flat package FLAT80 : The pitch converter
Connection	CUBE80 : The FLAT80 and the user system by solder FLAT80 : Two 40-pin dual in line connecters and cables

PCA7780G02

PITCH CONVERTOR for 80-PIN FLAT PACKAGE

DESCRIPTION

PCA7780G02 is a pitch convertor for 80-pin flat package to connect the mounting pattarn of the system mounted 80-pin flat package type single-chip microcomputer to the option boards, evaluation boards, and so on.

It can be connected to mounted evaluation microcomputer or the emulator by mounting the 80LCC IC socket to the foot pattern of 80-pin flat package on the terget system.

DISTINCTIVE FEATURES

- It can be used as incircuit emulation probe to debug the system mounted 80-pin flat package type microcomputer.
- It can be converted the pattern that has 80-pin two line half pitch dual-in-line to the pattern of 80-pin flat package (0.8mm pitch).

The following boards and IC socket are needed to use PCA7780G02.

(1) Using series MELPS 7700(M377XX)
PCA7781 : Pitch convertor

(2) Using M37450 family
PCA7782 : Pitch convertor

(3) 80LCC socket

- Surface mounting type IC61-0804-034
(Yamaichi electric industory corporation.)
- Manual solderable type 80LCC-046 (refer to figure 1)
(Mitsubishi electric corporation)

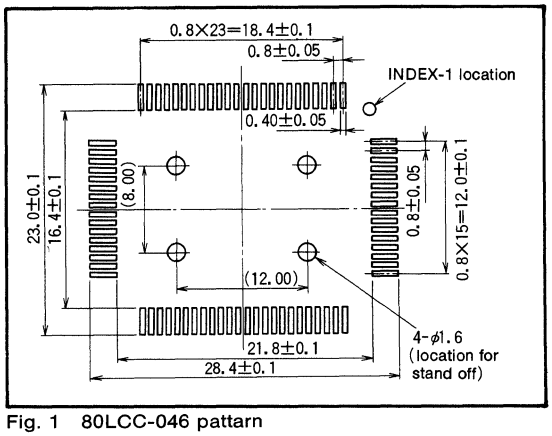


Fig. 1 80LCC-046 pattarn

FUNCTIONS

The process of PCA7780G02 connecting as following.

(1) Soldering the 80-pin LCC socket to the terget system.
(2) Connect the PCA7780G02 and PCA7781 or PCA7782 to the flat cable of the option board.
(3) Connect the PCA7780G02 to the 80-pin LCC socket .
(4) Connect the GND pin of PCA7781 or PCA7782 to the GND line of the option board.

SPECIFICATION

Item	Contents
Correspond MCU	80-pin flat package (0.8mm pitch) Type name is 80P6.

NOTE

The mounting pattern of 80QFP must be designed larger than normally to correspond to the packages 80P6, 80P6N, and two types 80LCC socket (refer Figure 2).

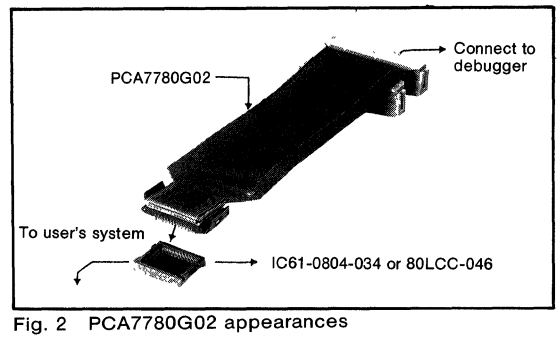


Fig. 2 PCA7780G02 appearances

SERIES MELPS 7700 MASK ROM ORDERING METHOD

GZZ—SH00—92A〈74B0〉

Mask ROM number	

SERIES MELPS 7700 MASK ROM ORDER CONFIRMATION FORM
SINGLE-CHIP 16-BIT MICROCOMPUTER M37700M2-XXXFP
MITSUBISHI ELECTRIC

Receipt	Date :	
	Section head signature	Supervisor signature

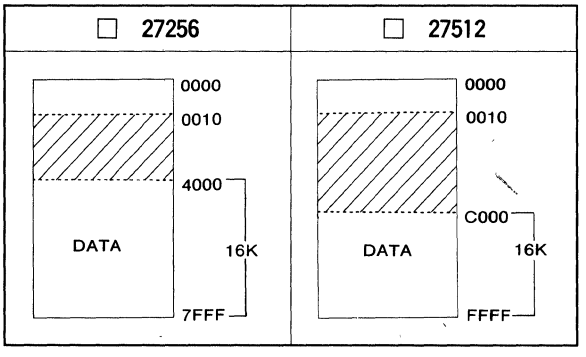
Note : Please fill in all items marked※

※	Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
		Date issued	Date :			

※1. Confirmation
Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM areas | | | | | (hexadecimal notation)

EPROM Type :



(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to 10_{16} are the area for storing the data on model designation and options. This area must be written with the data shown below.
Details for option data are given next in the section describing the STP instruction option.
Address and data are written in hexadecimal notation.

	Address		Address		Address
4D	0	2D	8	Option data	10
33	1	FF	9		
37	2	FF	A		
37	3	FF	B		
30	4	FF	C		
30	5	FF	D		
4D	6	FF	E		
32	7	FF	F		

※2. STP instruction option
One of the following sets of data should be written to the option data address (10_{16}) of the EPROM you have ordered. Check @ in the appropriate box.

☐ STP instruction enable | 01_{16} | Address 10_{16}
☐ STP instruction disable | 00_{16} | Address 10_{16}

※3. Mark specification
Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (80P6 for M37700M2-XXXFP) and attach to the Mask ROM Order Confirmation Form.

※4. Comments

SERIES MELPS 7700 MASK ROM ORDERING METHOD

GZZ—SH01—33A〈81A0〉

SERIES MELPS 7700 MASK ROM ORDER CONFIRMATION FORM
SINGLE-CHIP 16-BIT MICROCOMPUTER M37700M4-XXXFP
MITSUBISHI ELECTRIC

Mask ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please fill in all items marked※

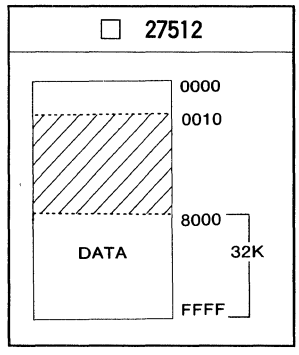
※ Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
	Date issued	Date :			

※1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM areas | | | | | (hexadecimal notation)

EPROM Type :



(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to 10_{16} are the area for storing the data on model designation and options. This area must be written with the data shown below.
Details for option data are given next in the section describing the STP instruction option.
Address and data are written in hexadecimal notation.

	Address		Address		Address
4D	0	2D	8	Option data	10
33	1	FF	9		
37	2	FF	A		
37	3	FF	B		
30	4	FF	C		
30	5	FF	D		
4D	6	FF	E		
34	7	FF	F		

※2. STP instruction option

One of the following sets of data should be written to the option data address (10_{16}) of the EPROM you have ordered. Check @ in the appropriate box.

☐ STP instruction enable | 01_{16} | Address 10_{16}
☐ STP instruction disable | 00_{16} | Address 10_{16}

※3. Mark specification

Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (80P6 for M37700M4-XXXFP) and attach to the Mask ROM Order Confirmation Form.

※4. Comments

SERIES MELPS 7700 MASK ROM ORDERING METHOD

GZZ—SH01—69A〈8ZA0〉

SERIES MELPS 7700 MASK ROM ORDER CONFIRMATION FORM
SINGLE-CHIP 16-BIT MICROCOMPUTER M37701M2-XXXSP
MITSUBISHI ELECTRIC

Mask ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please fill in all items marked※

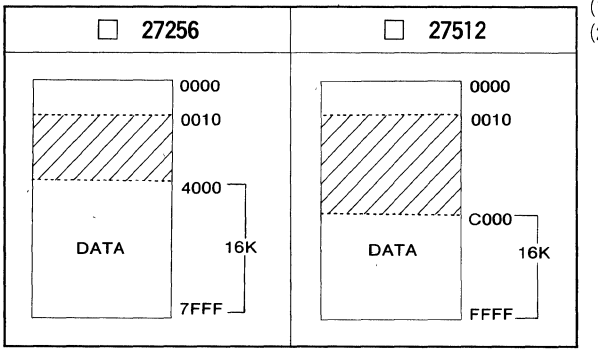
※ Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
	Date issued	Date :			

※1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM areas [ ][ ][ ][ ] (hexadecimal notation)

EPROM Type :



(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to 10_{16} are the area for storing the data on model designation and options. This area must be written with the data shown below.
Details for option data are given next in the section describing the STP instruction option.
Address and data are written in hexadecimal notation.

	Address		Address		Address
4D	0	2D	8	Option data	10
33	1	FF	9		
37	2	FF	A		
37	3	FF	B		
30	4	FF	C		
31	5	FF	D		
4D	6	FF	E		
32	7	FF	F		

※2. STP instruction option

One of the following sets of data should be written to the option data address (10_{16}) of the EPROM you have ordered. Check @ in the appropriate box.

☐ STP instruction enable [01_{16}] Address 10_{16}
☐ STP instruction disable [00_{16}] Address 10_{16}

※3. Mark specification

Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (64P4B for M37701M2-XXXSP) and attach to the Mask ROM Order Confirmation Form.

※4. Comments

MITSUBISHI ELECTRIC

SERIES MELPS 7700 MASK ROM ORDERING METHOD

GZZ—SH02—01A〈8ZA0〉

SERIES MELPS 7700 MASK ROM ORDER CONFIRMATION FORM
SINGLE-CHIP 16-BIT MICROCOMPUTER M37701M4-XXXSP
MITSUBISHI ELECTRIC

Mask ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

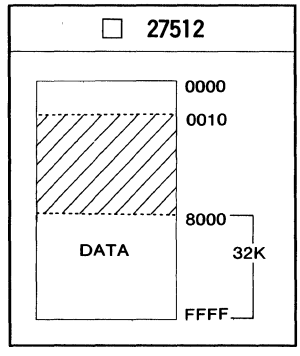
Note : Please fill in all items marked※

※ Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
	Date issued	Date :			

※1. Confirmation
Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM areas | | | | | (hexadecimal notation)

EPROM Type :



(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to 10_{16} are the area for storing the data on model designation and options. This area must be written with the data shown below.
Details for option data are given next in the section describing the STP instruction option.
Address and data are written in hexadecimal notation.

	Address		Address		Address
4D	0	2D	8	Option data	10
33	1	FF	9		
37	2	FF	A		
37	3	FF	B		
30	4	FF	C		
31	5	FF	D		
4D	6	FF	E		
34	7	FF	F		

※2. STP instruction option
One of the following sets of data should be written to the option data address (10_{16}) of the EPROM you have ordered. Check @ in the appropriate box.

☐ STP instruction enable | 01_{16} | Address 10_{16}
☐ STP instruction disable | 00_{16} | Address 10_{16}

※3. Mark specification
Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (64P4B for M37701M4-XXXSP) and attach to the Mask ROM Order Confirmation Form.

※4. Comments

SERIES MELPS 7700 MASK ROM ORDERING METHOD

GZZ—SH01—75A〈8ZA0〉

SERIES MELPS 7700 MASK ROM ORDER CONFIRMATION FORM
SINGLE-CHIP 16-BIT MICROCOMPUTER M37704M2-XXXFP
MITSUBISHI ELECTRIC

Mask ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please fill in all items marked※

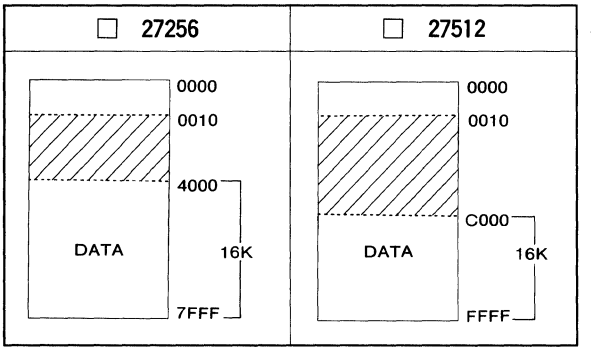
※ Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
	Date issued	Date :			

※1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM areas | | | | | (hexadecimal notation)

EPROM Type :



(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to 10_{16} are the area for storing the data on model designation and options. This area must be written with the data shown below.
Details for option data are given next in the section describing the STP instruction option.
Address and data are written in hexadecimal notation.

	Address		Address		Address
4D	0	2D	8	Option data	10
33	1	FF	9		
37	2	FF	A		
37	3	FF	B		
30	4	FF	C		
34	5	FF	D		
4D	6	FF	E		
32	7	FF	F		

※2. STP instruction option

One of the following sets of data should be written to the option data address (10_{16}) of the EPROM you have ordered. Check @ in the appropriate box.

☐ STP instruction enable 01_{16} Address 10_{16}
☐ STP instruction disable 00_{16} Address 10_{16}

※3. Mark specification

Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (80P6N for M37704M2-XXXFP) and attach to the Mask ROM Order Confirmation Form.

※4. Comments

SERIES MELPS 7700 MASK ROM ORDERING METHOD

GZZ—SH01—81A〈8ZA0〉

SERIES MELPS 7700 MASK ROM ORDER CONFIRMATION FORM
SINGLE-CHIP 16-BIT MICROCOMPUTER M37705M2-XXXSP
MITSUBISHI ELECTRIC

Mask ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please fill in all items marked※

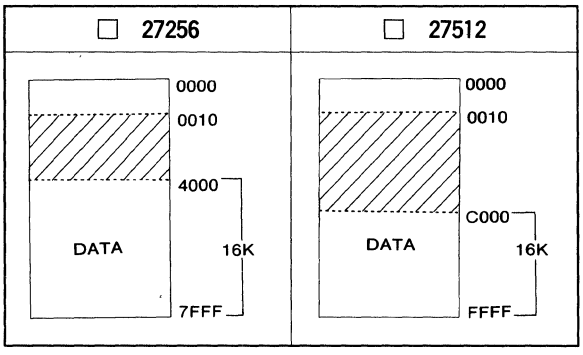
※ Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
	Date issued	Date :			

※1. Confirmation

Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce masks based on this data. We shall assume the responsibility for errors only if the mask ROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM areas [ ][ ][ ][ ] (hexadecimal notation)

EPROM Type :



(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to 10_{16} are the area for storing the data on model designation and options. This area must be written with the data shown below.
Details for option data are given next in the section describing the STP instruction option.
Address and data are written in hexadecimal notation.

	Address		Address		Address
4D	0	2D	8	Option data	10
33	1	FF	9		
37	2	FF	A		
37	3	FF	B		
30	4	FF	C		
35	5	FF	D		
4D	6	FF	E		
32	7	FF	F		

※2. STP instruction option

One of the following sets of data should be written to the option data address (10_{16}) of the EPROM you have ordered. Check @ in the appropriate box.

☐ STP instruction enable 01_{16} Address 10_{16}
☐ STP instruction disable 00_{16} Address 10_{16}

※3. Mark specification

Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (64P4B for M37705M2-XXXSP) and attach to the Mask ROM Order Confirmation Form.

※4. Comments

SERIES MELPS 7700 PROM ORDERING METHOD

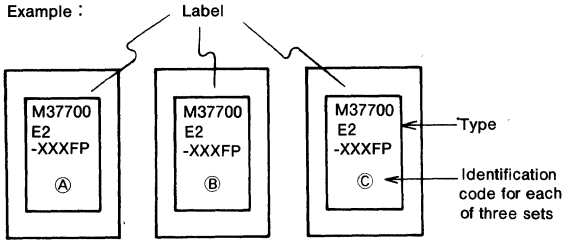
PROM ORDERING METHOD

Mitsubishi Electric corp. accepts order to transfer EPROM supplied program data into the one time PROMs in single-chip 16-bit microcomputers. When placing such order, please submit the information described below.

1. Writing to PROM Order Confirmation Form ····· 1 set
   (There is a specific form to be used for each model.)
2. Data to be written into PROM built in ····· EPROM
   (Please provide three sets containing the identical data.)
3. Mark Specification Form ····· 1 set

NOTES

(1) Acceptable EPROM type
Any EPROM made by Mitsubishi Electric corp. that is listed in the Writing to PROM Order Confirmation Form may be used.



(2) EPROM window labeling
Please write the model name and the identification code (A, B, C) on the label for each of the three sets of data EPROMs provided.

(3) Calculation and indication of check sum code
Please calculate the total number of data in words in the EPROM, and write the number in 4-digit hexadecimal form in the check sum code field of the Writing to PROM Order Confirmation Form.

(4) Marking specification method
The permissible marking specifications differ depending on the shape of package. Please fill out the Marking Specification Form and attach it to the Writing to PROM Order Confirmation Form.

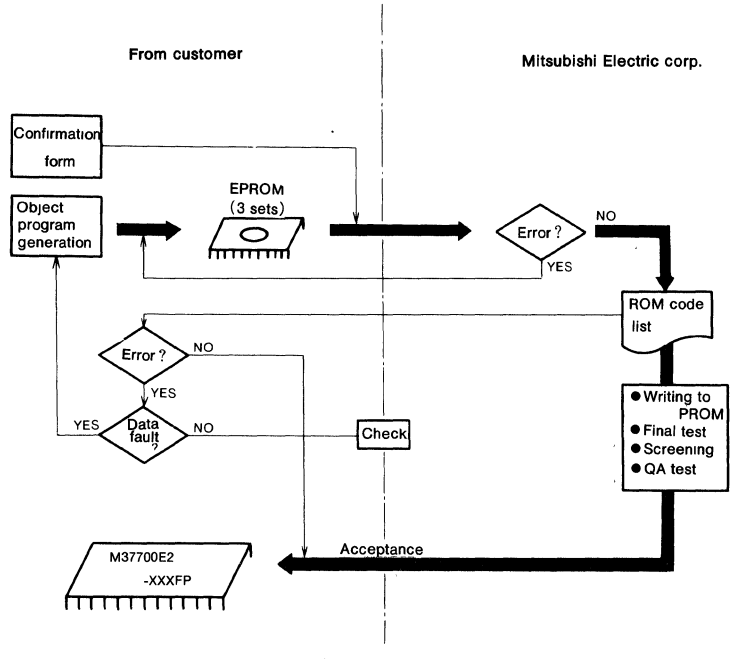
OUTLINE OF ORDER PROCESSING

Mitsubishi Electric corp. will produce Writing to PROM if at least two of the three EPROM sets submitted contain identical data.

If we find error when the submitted EPROMs are compared, we will contact your representative. Thus, we assume responsibility only when we produce Writing to PROMs that contain data other than the data correctly provided by the customer.

The chart below shows the flow of one time PROM production.

SERIES MELPS 7700 ONE TIME PROM DEVELOPMENT CAD SYSTEM



MITSUBISHI ELECTRIC

GZZ—SH01—57A〈84A0〉

SERIES MELPS 7700 WRITING TO PROM ORDER CONFIRMATION FORM
SINGLE-CHIP 16-BIT MICROCOMPUTER
M37700E2AXXXFP
MITSUBISHI ELECTRIC

ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

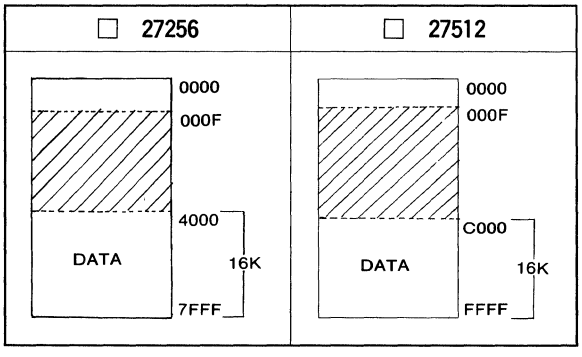
Note : Please fill in all items marked※

※	Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
		Date issued	Date :			

※1. Confirmation
Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce writing to PROM based on this data. We shall assume the responsibility for errors only if the written PROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM areas | | | | | (hexadecimal notation)

EPROM Type :



(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to $0F_{16}$ are the area for storing the data on model designation. This area must be written with the data shown below.
Address and data are written in hexadecimal notation.

	Address		Address
4D	0	41	8
33	1	FF	9
37	2	FF	A
37	3	FF	B
30	4	FF	C
30	5	FF	D
45	6	FF	E
32	7	FF	F

※2. Mark specification
Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (80P6 for M37700E2AXXXFP) and attach to the Writing to PROM Order Confirmation Form.

※3. Comments

SERIES MELPS 7700 PROM ORDERING METHOD

GZZ—SH01—36A〈81A0〉

SERIES MELPS 7700 WRITING TO PROM ORDER CONFIRMATION FORM

SINGLE-CHIP 16-BIT MICROCOMPUTER
M37700E4AXXXFP
MITSUBISHI ELECTRIC

ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

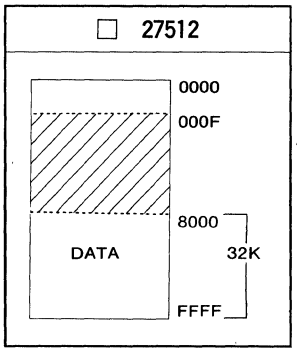
Note : Please fill in all items marked※

※	Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
		Date issued	Date :			

※1. Confirmation
Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce writing to PROM based on this data. We shall assume the responsibility for errors only if the written PROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM areas | | | | | (hexadecimal notation)

EPROM Type :



(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to $0F_{16}$ are the area for storing the data on model designation. This area must be written with the data shown below.
Address and data are written in hexadecimal notation.

	Address		Address
4D	0	41	8
33	1	FF	9
37	2	FF	A
37	3	FF	B
30	4	FF	C
30	5	FF	D
45	6	FF	E
34	7	FF	F

※2. Mark specification
Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (80P6 for M37700E4AXXXFP) and attach to the Writing to PROM Order Confirmation Form.

※3. Comments

GZZ—SH01—68A〈8ZA0〉

SERIES MELPS 7700 WRITING TO PROM ORDER CONFIRMATION FORM
SINGLE-CHIP 16-BIT MICROCOMPUTER
M37701E2AXXXSP
MITSUBISHI ELECTRIC

ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please fill in all items marked※

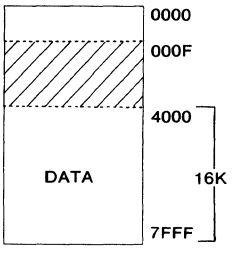
※	Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
		Date issued	Date :			

※1. Confirmation
Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce writing to PROM based on this data. We shall assume the responsibility for errors only if the written PROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

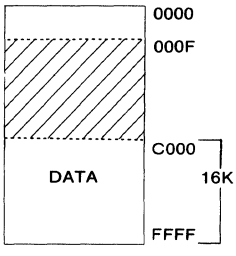
Checksum code for entire EPROM areas [ ][ ][ ][ ] (hexadecimal notation)

EPROM Type :

☐ 27256



☐ 27512



(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to $0F_{16}$ are the area for storing the data on model designation. This area must be written with the data shown below.
Address and data are written in hexadecimal notation.

	Address		Address
4D	0	41	8
33	1	FF	9
37	2	FF	A
37	3	FF	B
30	4	FF	C
31	5	FF	D
45	6	FF	E
32	7	FF	F

※2. Mark specification
Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (64P4B for M37701E2AXXXSP) and attach to the Writing to PROM Order Confirmation Form.

※3. Comments

MITSUBISHI ELECTRIC

GZZ—SH02—04A〈8ZA0〉

SERIES MELPS 7700 WRITING TO PROM ORDER CONFIRMATION FORM
SINGLE-CHIP 16-BIT MICROCOMPUTER
M37701E4AXXXSP
MITSUBISHI ELECTRIC

ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

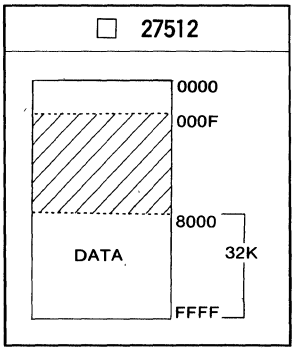
Note : Please fill in all items marked※

※ Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
	Date issued	Date :			

※1. Confirmation
Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce writing to PROM based on this data. We shall assume the responsibility for errors only if the written PROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM areas [ ][ ][ ][ ] (hexadecimal notation)

EPROM Type :



(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to $0F_{16}$ are the area for storing the data on model designation. This area must be written with the data shown below.
Address and data are written in hexadecimal notation.

	Address		Address
4D	0	41	8
33	1	FF	9
37	2	FF	A
37	3	FF	B
30	4	FF	C
31	5	FF	D
45	6	FF	E
34	7	FF	F

※2. Mark specification
Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (64P4B for M37701E4AXXXSP) and attach to the Writing to PROM Order Confirmation Form.

※3. Comments

GZZ—SH01—72A〈8ZA0〉

SERIES MELPS 7700 WRITING TO PROM ORDER CONFIRMATION FORM
SINGLE-CHIP 16-BIT MICROCOMPUTER
M37704E2AXXXFP
MITSUBISHI ELECTRIC

ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

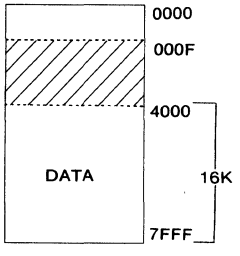
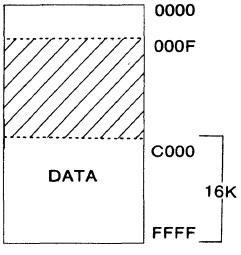
Note : Please fill in all items marked※

※ Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
	Date issued	Date :			

※1. Confirmation
Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce writing to PROM based on this data. We shall assume the responsibility for errors only if the written PROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM areas | | | | | (hexadecimal notation)

EPROM Type :

☐ 27256	☐ 27512
	

(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to $0F_{16}$ are the area for storing the data on model designation. This area must be written with the data shown below.
Address and data are written in hexadecimal notation.

	Address		Address
4D	0	41	8
33	1	FF	9
37	2	FF	A
37	3	FF	B
30	4	FF	C
34	5	FF	D
45	6	FF	E
32	7	FF	F

※2. Mark specification
Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (80P6N for M37704E2AXXXFP) and attach to the Writing to PROM Order Confirmation Form.

※3. Comments

SERIES MELPS 7700 PROM ORDERING METHOD

GZZ—SH01—80A〈8ZA0〉

SERIES MELPS 7700 WRITING TO PROM ORDER CONFIRMATION FORM
SINGLE-CHIP 16-BIT MICROCOMPUTER
M37705E2AXXXSP
MITSUBISHI ELECTRIC

ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

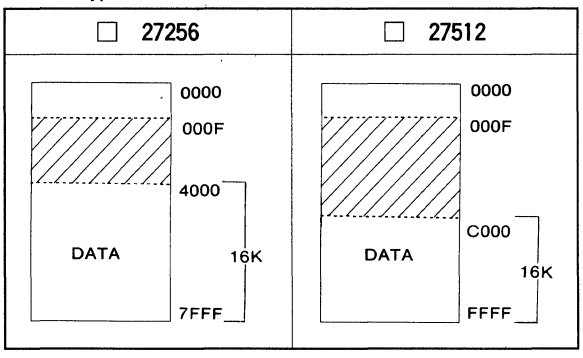
Note : Please fill in all items marked※

※	Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
		Date issued	Date :			

※1. Confirmation
Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce writing to PROM based on this data. We shall assume the responsibility for errors only if the written PROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM areas [ ][ ][ ][ ] (hexadecimal notation)

EPROM Type :



(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to $0F_{16}$ are the area for storing the data on model designation. This area must be written with the data shown below.
Address and data are written in hexadecimal notation.

	Address		Address
4D	0	41	8
33	1	FF	9
37	2	FF	A
37	3	FF	B
30	4	FF	C
35	5	FF	D
45	6	FF	E
32	7	FF	F

※2. Mark specification
Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (64P4B for M37705E2AXXXSP) and attach to the Writing to PROM Order Confirmation Form.

※3. Comments

GZZ—SH02—87A〈9XA0〉

SERIES MELPS 7700 WRITING TO PROM ORDER CONFIRMATION FORM
SINGLE-CHIP 16-BIT MICROCOMPUTER
M37796E4TXXXJ
MITSUBISHI ELECTRIC

ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

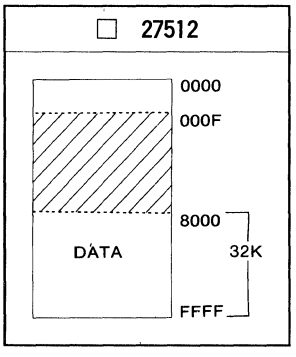
Note : Please fill in all items marked※

※	Customer	Company name	TEL ( )	Issuance signatures	Responsible officer	Supervisor
		Date issued	Date :			

※1. Confirmation
Specify the name of the product being ordered and the type of EPROMs submitted.
Three sets of EPROMs are required for each pattern (Check @ in the appropriate box).
If at least two of the three sets of EPROMs submitted contain the identical data, we will produce writing to PROM based on this data. We shall assume the responsibility for errors only if the written PROM data on the products we produce differ from this data. Thus, the customer must be especially careful in verifying the data contained in the EPROMs submitted.

Checksum code for entire EPROM areas | | | | | (hexadecimal notation)

EPROM Type :



(1) Set "FF_{16}" in the shaded area.
(2) Address 0_{16} to $0F_{16}$ are the area for storing the data on model designation. This area must be written with the data shown below.
Address and data are written in hexadecimal notation.

	Address		Address
4D	0	54	8
33	1	FF	9
37	2	FF	A
37	3	FF	B
39	4	FF	C
36	5	FF	D
45	6	FF	E
34	7	FF	F

※2. Mark specification
Mark specification must be submitted using the correct form for the type of package being ordered fill out the appropriate Mark Specification Form (84P0 for M37796E4TXXXJ) and attach to the Writing to PROM Order Confirmation Form.

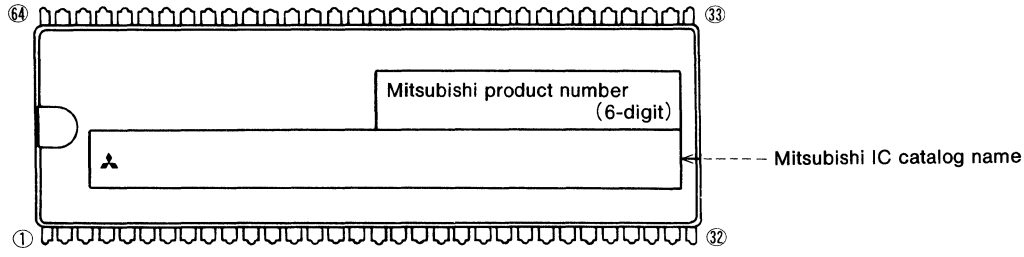
※3. Comments

64P4B (64-PIN SHRINK DIP) MARK SPECIFICATION FORM

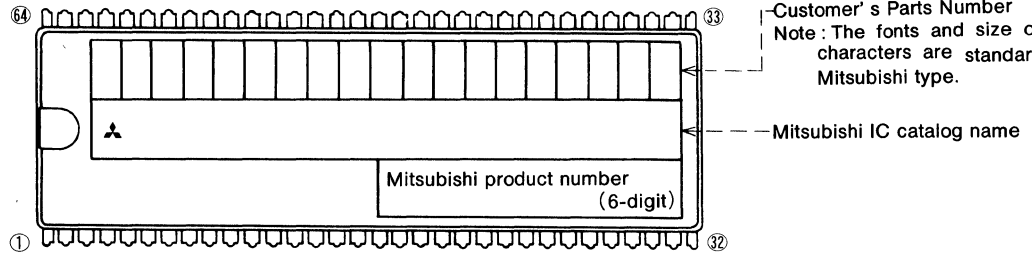
Mitsubishi IC catalog name

Please choose one of the marking types below (A, B, C), and enter the Mitsubishi IC catalog name and the special mark (if needed).

A. Standard Mitsubishi Mark



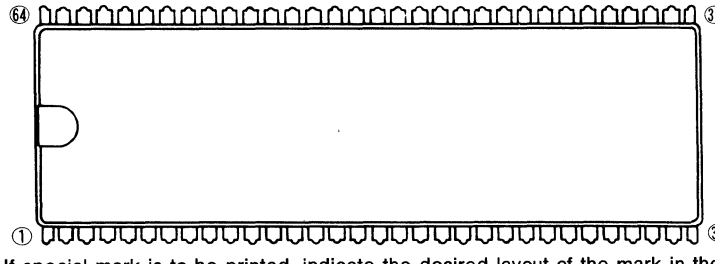
B. Customer' s Parts Number + Mitsubishi IC Catalog Name



Note1 : The mark field should be written right aligned.
2 : The fonts and size of characters are standard Mitsubishi type.
3 : Customer' s parts number can be up to 19 alphanumeric characters for capital letters, hyphens, commas, periods and so on.
4 : If the Mitsubishi logo ⅄ is not required, check the box on the right.

⅄ Mitsubishi logo is not required ☐

C. Special Mark Required



Note1 : If special mark is to be printed, indicate the desired layout of the mark in the upper figure. The layout will be duplicated technically as close as possible. Mitsubishi product number (6-digit) and Mask ROM number (3-digit) are always marked for sorting the products.
2 : If special character fonts (e. g., customer's trade mark logo) must be used in special mark, check the box on the right.
For the new special character fonts a clean font original (ideally logo drawing) must be submitted.

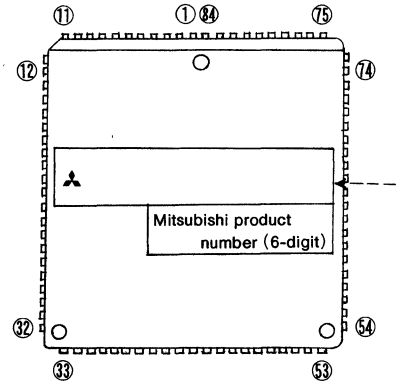
Special character fonts required ☐

84P0 (84-PIN PLCC) MARK SPECIFICATION FORM

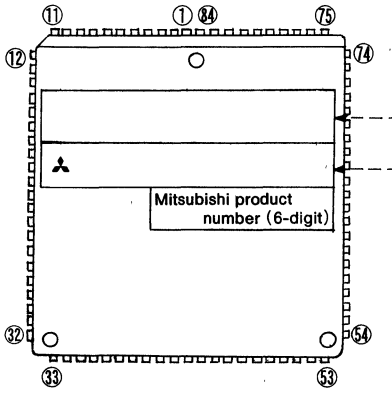
Mitsubishi IC catalog name

Please choose one of the marking types below (A, B, C), and enter the Mitsubishi IC catalog name and the special mark (if needed).

A. Standard Mitsubishi Mark



B. Customer's Parts Number + Mitsubishi IC Catalog Name

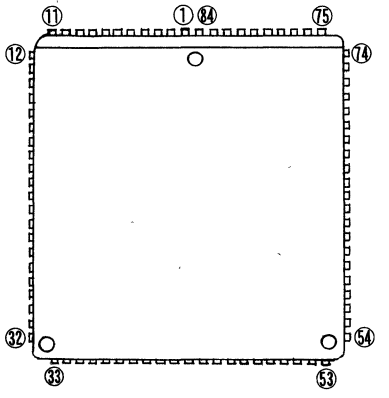


Note : The fonts and size of characters are standard Mitsubishi type.

Note1 : The mark field should be written right aligned.
2 : The fonts and size of characters are standard Mitsubishi type.
3 : Customer's parts number can be up to 16 alphanumeric characters for capital letters, hyphens, commas, periods and so on.
4 : If the Mitsubishi logo is not required, check the box below.

Mitsubishi logo is not required ☐

C. Special Mark Required



Note1 : If Special Mark is to be printed, indicate the desired layout of the mark in the left figure. The layout will be duplicated technically as close as possible. Mitsubishi product number (6-digit) and Mask ROM number (3-digit) are always marked.
2 : If special character fonts (e.g., customer's trade mark logo) must be used in special mark, check the box below.
For the new special character fonts a clean font original (ideally logo drawing) must be submitted.

Special character fonts required ☐

MITSUBISHI DATA BOOK
SINGLE-CHIP 16-BIT MICROCOMPUTERS

May. First Edition 1990

Editioned by
Committee of editing of Mitsubishi Semiconductor Data Book

Published by
Mitsubishi Electric Corp., Semiconductor Division

This book, or parts thereof, may not be reproduced in any form without permission of Mitsubishi Electric Corporation.
© 1990 MITSUBISHI ELECTRIC CORPORATION Printed in Japan

MITSUBISHI SEMICONDUCTORS
SINGLE-CHIP 16-BIT MICROCOMPUTERS

MITSUBISHI ELECTRIC CORPORATION
HEAD OFFICE: MITSUBISHI DENKI BLDG., MARUNOUCHI, TOKYO 100 TELEX: J24532 CABLE: MELCO TOKYO

These products or technologies are subject to Japanese and/or COCOM strategic restrictions, and diversion contrary thereto is prohibited.

H-D0341-A KI-9005 Printed in Japan (ROD)
© 1990 MITSUBISHI ELECTRIC CORPORATION

New publication, effective May. 1990.
Specifications subject to change without notice.