K-SERIES® MICROCONTROLLERS

DATA BOOK

1994

8- and 16-Bit Microcontrollers

Development Tools

NEC

K-SERIES® MICROCONTROLLERS DATA BOOK 1994

NEC

NEC Electronics Inc.

NEC

1994
K-SERIES MICROCONTROLLERS
DATA BOOK

Document No. 50053-1

NEC

Contents

NEC

Section 7
Soldering

Section 8
Package Drawings

Section 8 (cont)
Package Drawings

NEC

Numerical Index

NEC

Section 1
Reliability and Quality Control

NECEL Electronics Inc. is dedicated to the QCD principle of providing the highest *quality* product at the lowest possible *cost* with on-time *delivery* to our customers.

As large-scale integrated (LSI) circuits increase in density, the reliability of individual devices imposes a more profound impact on system reliability. As a result, great emphasis has been placed by LSI circuit manufacturers on assuring device reliability.

Conventionally, performing reliability tests and using feedback from the field have been the only methods of monitoring and measuring reliability. As LSI density increases, however, internal device circuit elements have become more difficult to activate from external terminals and to detect their degradation. Testing and feedback alone cannot provide enough information to ensure today's demanding reliability requirements.

To guarantee and improve the reliability of LSI circuits, a new philosophy and methodology are needed for reliability assurance. Quality and reliability must not only be monitored and measured but, most importantly, must be built into the product.

BUILT-IN TQC

NECEL introduced the concept of total quality control (TQC) across its entire semiconductor product line to implement this philosophy. Quality control is now an integral part of each process step and requires production, engineering, quality control staffs, and all management personnel to participate in TQC activities. Figure 1 is a flowchart that shows how these activities form a comprehensive quality control system at NECEL.

In addition to TQC, NECEL introduced a pre-screening method into the production line that eliminates potentially defective units. This combination of building in quality and screening out projected early failures has resulted in superior quality and reliability.

Most LSI circuits use high-density MOS technology. Their state-of-the-art high performance improved fine-line generation techniques. When physical parameters are reduced, circuit density and performance increase and active circuit power dissipation decreases.The information presented here will show that this advanced technology combined with the practice of TQC yields products as reliable as those from previous technologies.

APPROACHES TO TQC

TQC activities are geared toward total customer satisfaction. The success of these activities depends on management's commitment to enhancing employee development, maintaining a customer-first attitude, and fulfilling community responsibilities.

TQC is implemented in the following steps. First, quality control is embedded into each process, allowing early detection of possible failure mechanisms and immediate feedback. Second, the reliability and quality assurance policy is upheld through company-wide quality control activities. Third, emphasis is placed on research and development efforts to achieve even higher standards of device quality and reliability. Fourth, extensive failure analysis is performed periodically, and appropriate corrective actions are taken as preventative measures.

Process control limits are based on statistical data gathered from this analysis and used to determine the effectiveness of the in-process quality control steps.

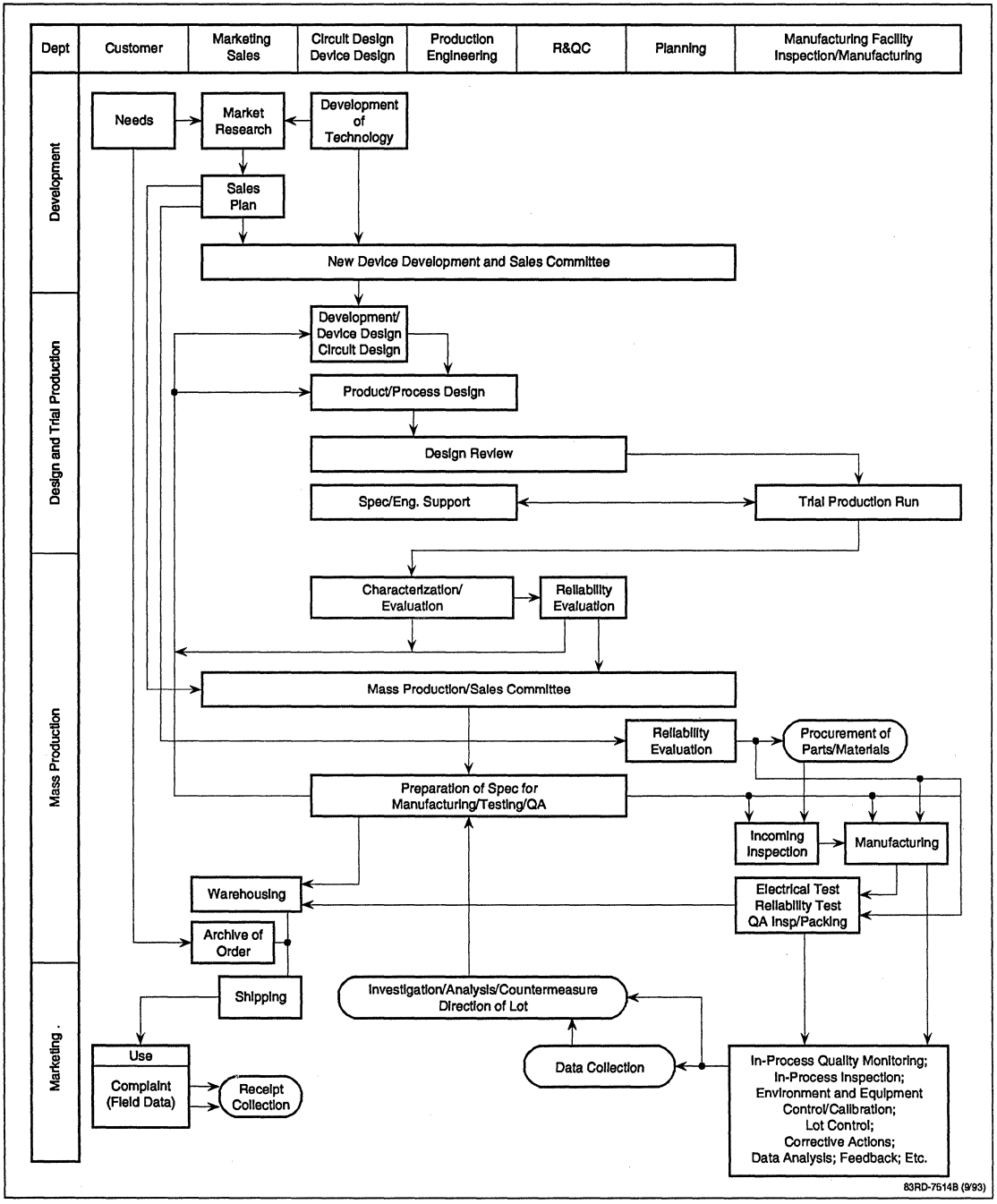
New standards are continuously upgraded, and the iterative process continues. The goal is to maintain the superior product quality and reliability that has become synonymous with the NEC name.

Zero Defects Program

One of the quality control activities that involves every staff level is the "Zero Defects" (ZD) program. The purpose of the ZD program is to minimize, if not prevent, defects due to controllable causes. These activities are organized by groups of workers around these four premises.

- A group must have a target or purpose to pursue.
- Several groups can be organized to pursue a common target.
- Each group must have a responsible leader.
- Each group is well supported by management.

Figure 1. NECEL's Quality Control System



The group's target is selected from items relating to specifications, inspections, operation standards, etc. When past data is available, a Pareto diagram is created and reviewed to select an item most in need of quality improvement. Target defects related to this item are clearly defined. Records are analyzed to compute numerical equivalents of the defects. Then, action is taken to control these defects.

Statistical Approach

Another approach to quality control is statistical analysis. NECEL uses statistical analysis at each stage of LSI product development, trial runs, and mass production. These are some implementations of this statistical approach:

- Process comparisons
- Control charts
- Data analysis
  — Correlation, regression, multivariance, etc.
- Cp/Cpk studies
  — Variables and attributes data (performed monthly)

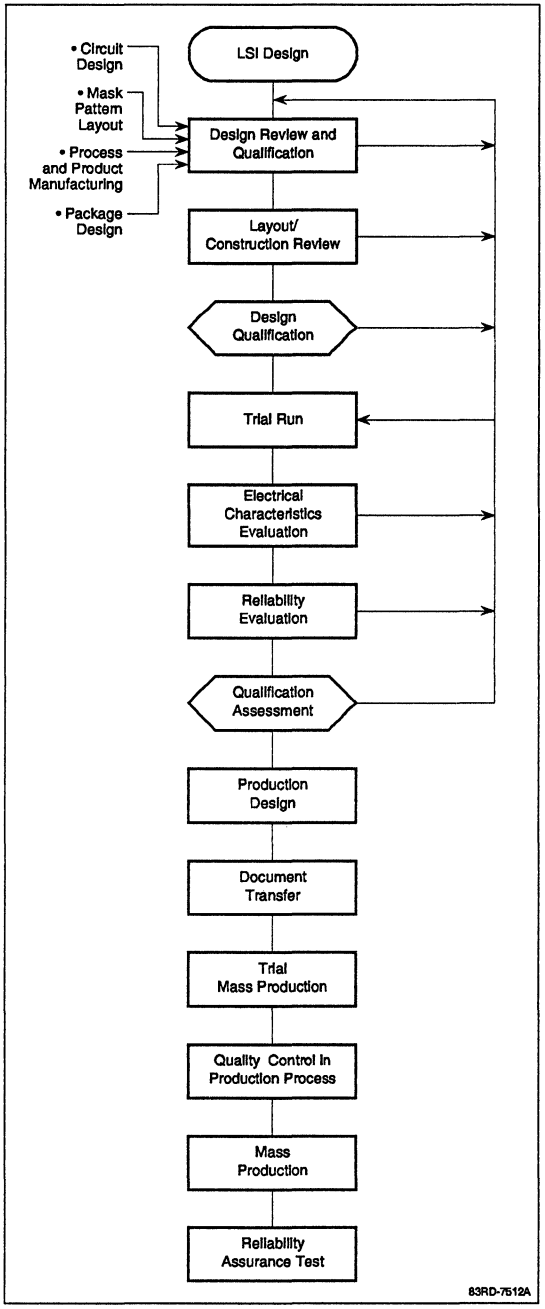
Process control sheets and other QC tools are used to monitor important parameters such as Cp, Cpk, X, X-R, electrical parameters, pattern dimensions, bond strength, test percentage defects, etc. The results of these studies are monitored by the production staff, QC engineers, and other associated engineers. If any out-of-control or out-of-specification limit is observed, corrective procedures are quickly taken.

IMPLEMENTATION OF QUALITY CONTROL

Building quality into a product requires early detection of possible failure mechanisms and immediate feedback to remove such problems. A fixed quality inspection station often cannot provide prompt and accurate feedback about the process steps prior to the inspection. Quality control functions have therefore been distributed into each process step including the conceptual stage. These are the most significant areas where quality control has been placed:

- Product development
- Incoming material inspection
- Wafer processing
- Chip mounting and packaging
- Electrical testing and infant mortality screening
- Outgoing material inspection
- Reliability assurance tests
- Process/product changes

Figure 2. New Product Development



Product Development

New product development includes the product concept, device proposal review, physical element design and organization, engineering evaluation, and, finally, product transfer to manufacturing. Quality and reliability are considered at every step. The new product development flow at NECEL is shown in figure 2.

Design is the first and most important step in new product development. NECEL believes that the foundation of device quality is determined at the design stage. The four steps involved are circuit design, mask pattern layout, package design, and the setting of process and product manufacturing conditions. Design standards have been established at NECEL to maximize quality and reliability.

After completion of the design, a design review is performed to check for conformity to design standards and to consider other factors influencing reliability and quality. At this stage, modification or re-design may be necessary. NECEL believes that design reviews are essential for product modifications as well as for newly designed products.

Once a design successfully passes its review, a trial run takes place in which the product's electrical and mechanical characteristics, quality, and reliability are evaluated.

Additional runs are performed in which process conditions are varied deliberately, causing characteristic factors to change in mass production. These samples are evaluated to determine the best combination of process conditions. Reliability tests are then conducted to check the new product's electrical and mechanical stress resistance. If no problems are found at this stage, the product is approved for mass production.

Mass production begins after the product design department prepares a schedule that includes reliability and quality control steps. The standards for production and control steps are continuously re-examined for possible improvement, even after mass production has started.

Incoming Material Inspection

NECEL has the following programs to control incoming materials:

- Vendor/material qualification system
- Purchasing specifications for materials
- Incoming materials inspection
- Inspection data feedback
- Meetings with vendors concerning quality
- Vendor audits

If any parts or materials are rejected at incoming inspection, they are returned to the vendor with a rejection notification form specifying the failure items and modes. The results of these inspections are used to rate the vendors for future purchasing.

In-Process Quality Inspection

Typical in-process quality inspections performed at wafer fabrication, chip mounting and packaging, and device testing stages are listed in appendix 1A and appendix 1B.

Electrical Testing and Screening

At the first electrical test, dc parameters are tested according to electrical specifications on 100% of each lot. This is a prescreening prior to any infant mortality test. At the second electrical test, ac functional tests as well as dc parameter tests are performed on 100% of each lot. If the percentage of defective units in a lot is unacceptably high in this test, the lot is subjected to an infant mortality rescreen. During this time, any defective units undergo extensive failure analysis. The results of these analyses are fed back into the process through corrective actions.

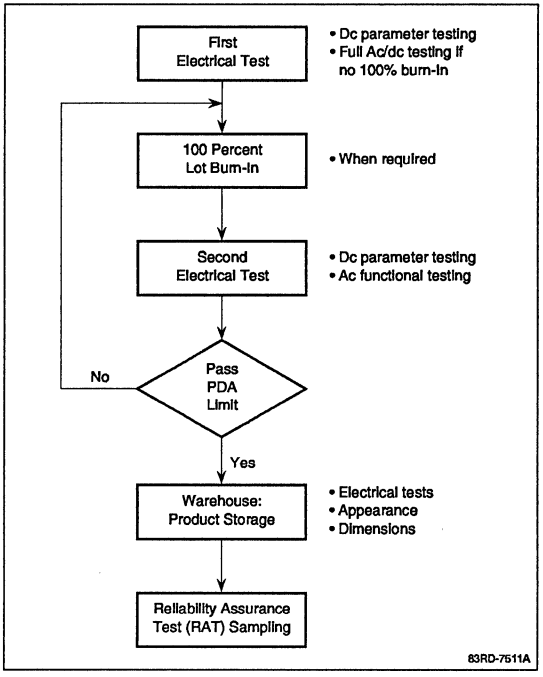
Figure 3 is a flowchart of the typical infant mortality screening and electrical testing.

Outgoing Inspection

Prior to warehouse storage or shipment, lots are subjected to an outgoing inspection according to the following sampling plan:

- Electrical
  - — Dc parameters, lot tolerance parts defective (LTPD) 3%
  - — Ac functional LTPD 3%
- Appearance
  - — Major LTPD 3%
  - — Minor LTPD 7%

Figure 3. Electrical Testing and Screening



Reliability Assurance Tests

Prior to shipment, representative samples from each process family are taken on a regular basis and subjected to monitoring reliability tests. This testing is performed to confirm that NECEL's products continually meet their field reliability targets.

Process/Product Changes

As mentioned previously, a design review occurs for product changes as well as for new products. Once a design is approved and processes are altered for maximum quality, qualification testing is performed to check reliability. If the test results are acceptable, the product is internally qualified for mass production.

The typical reliability qualification tests performed at NECEL are listed in appendix 3.

RELIABILITY THEORY

Reliability is defined as a characteristic of an item expressed by the probability that it will perform a required function, under specific conditions, for a certain period of time. The concept of probability, the definition of required function, and the knowledge of how time affects the item of concern are therefore necessary tools for the study of reliability.

Definition of a required function, by implication, treats the definition of a failure. Failure of a device is defined as the termination of a device's ability to perform its required function. A device has failed if it is unable to meet guaranteed values given in its electrical specifications.

Failures are categorized by the period of time in which they occur. The critical times used in the discussion of device reliability and failure are the periods of early, random, and wearout failures. Probability is used to quantitatively estimate reliability levels during these periods as well as overall reliability. The relevant theories and methods of calculation will be discussed later.

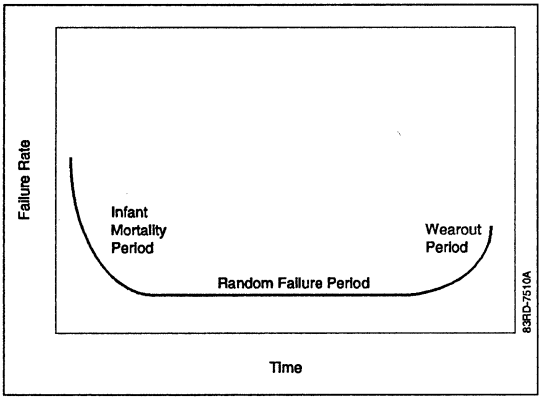
Regarding individual devices, specific failure mechanisms seen in life tests and in infant mortality screening tests are the parameters of concern in the determination of overall device failure rates, thus reliability levels.

Regarding systems, the sum of individual device failure rates is the expected failure rate of the system hardware.

Life Distribution

The fundamental principles of reliability engineering predict that the failure rate of a group of devices will follow the well-known bathtub curve in figure 4.

Figure 4. Reliability Life (Bathtub) Curve



The curve is divided into three regions: infant mortality, random failures, and wearout failures.

The infant mortality section of the curve, where the failure rate is declining rapidly, represents the early-life device failures. These failures are usually associated with one or more manufacturing defects.

After a period of time, the failure rate reaches a low value. This random failure area of the curve represents the useful portion of a device's life. During this random failure period, a slight decline is observed due to the depletion of potential random failures from the general population.

Wearout failures occur at the end of useful device life. These failures are observed in the rapidly rising failure rate portion of the curve; devices are wearing out both physically and electrically.

Therefore, for a device that has a very long life expectancy compared to the system that contains it, the areas of concern will be the infant mortality and random failure portions of the bathtub curve.

Failure Distribution at NECEL

To eliminate infant mortality failures, NECEL subjects its products to production burn-in whenever necessary. This burn-in is performed at an elevated temperature on 100% of the devices involved and is designed to remove potentially defective units.

After elimination of early device failures, a system will be left to the random failures of its components. To make proper projections of the failure rate of a system in the operating environment, random failure rates must be predicted for the system's components.

To qualitatively study random failures, integrated circuits returned from the field, as well as in-house life testing failures, undergo extensive failure analyses at respective NEC manufacturing divisions. Failure mechanisms are identified and resulting data is fed back to appropriate production and engineering groups. Long-term failure rates are determined from this data to quantitatively study this random failure population.

Infant Mortality Failure Screening

Establishing infant mortality screening requires knowledge of likely failure mechanisms and their associated activation energies.

Typical problems associated with infant mortality failures are manufacturing defects and process anomalies, which consist of contamination, cracked chips, wire bond shorts, or bad wire bonds. Since these problems can result from a number of possible failure mechanisms, the activation energy for infant mortality can vary considerably. Correspondingly, the effectiveness of an infant mortality screening condition (preferably at some stress level to shorten the screening time) varies greatly with the failure mechanism.

For example, failures due to ionic contamination have an activation energy of approximately 1.0 eV. Therefore, a 15-hour stress at 125°C junction temperature would be the equivalent of approximately 314 days of operation at a junction temperature of 55°C. On the other hand, failures due to oxide defects have an activation energy of approximately 0.3 eV. A 15-hour stress at 125°C junction temperature in this case would be the equivalent of approximately 4 days of operation at 55°C junction temperature. The condition and duration of infant mortality screening is determined by the economic factors involved in the screening and by the allowable rate of component failure. A component failure causes a system failure.

Empirical data gathered at NECEL indicates that any early failures generally occur after less than 4 hours of stress at 125°C ambient temperature. This fact is supported by the bathtub curve created from actual life test results. The failure rate after 4 hours of such stress testing shows random distribution as opposed to the rapidly decreasing failure rate observed in the early life portion of the curve.

Whenever necessary, NECEL has adopted this infant mortality burn-in at 125°C as a standard production screening procedure. NECEL believes it is imperative that failure modes associated with such infant mortality screens be understood and fixed at the manufacturing level. Failure analysis is performed on all infant mortality failures for this purpose. This in-line data coupled with data accumulated from the field is used to introduce corrective actions and quality improvement measures. If the early-life failures of a device can be minimized or eliminated and countermeasures appropriately monitored, then such screens can be eliminated. The result of such practices is that field reliability of NECEL devices is an order of magnitude higher than NECEL's long-term failure rate goals.

Table 1. Typical Reliability Test Results

Name	Type	HTB (1000H)	T/H (1000H)	PCT (192H)	T/C (300)
Micro (Note 1)	NMOS	9/26169 (13 FIT)	3/15977	0/16928	0/3542
	CMOS	7/29829 (4.3 FIT)	7/23123	0/23275	0/12238
Memory (HTOL)	1 Meg DRAM (Note 2)	44/38217 (43 FIT)	0/18210	0/6320	0/11300
	4 Meg DRAM (Note 3)	12/8085 (2.2 FIT)	1/2866	0/2100	0/2020
	256K SRAM (Note 4)	1/2812 (22 FIT)	1/2562	0/1900	0/3232
	1 Meg SRAM (Note 4)	0/2136 (1.25 FIT)	2/1959	0/1080	0/1375
ASIC (Note 5)	CMOS	7/8787 (21 FIT)	0/3577	5/13971	6/9693
	BiCMOS	3/2801 (29 FIT)	0/3601	0/4535	0/5825

Note:

Information in the table above has been extracted from NECEL report numbers:

(1) IRQ-3Q-24163
(2) TRQ-93-01-0142
(3) TRQ-93-01-0141
(4) TRQ-93-07-0165
(5) TRQ-93-07-0163

Accelerated Reliability Testing

NECEL performs extensive reliability testing at both pre-production and post-production levels to ensure that all products meet NECEL's minimum expectations and those of the field.

Assume an electronic system contains 1000 integrated circuits and that 1% system failures per month can be tolerated by this system. The allowable failure rate per component is then calculated as follows:

$$\frac{1\% \text{ failures}}{720 \text{ hours} \times 1000 \text{ pieces}} = (0.0014)\frac{\%\text{ failures}}{1000 \text{ hours}}$$

$$= 14 \text{ FITs}$$

The rate of 14 FITs corresponds to one failure in 85 devices during an operating test of approximately 10,000 hours. To demonstrate this reliability level in a reasonable amount of time, a test condition is apparently required to accelerate the time-to-failure in a predictable and understandable way.

The most common method for decreasing time-to-failure is the use of high temperature to accelerate physiochemical reactions that can lead to device failure. Other stressful environmental conditions are voltage, current, humidity, vibration, or some combination of these. Appendix 2 lists typical accelerated reliability assurance tests performed at NECEL on molded integrated circuits. Table 1 shows the results of some of these tests for various process types.

Reliability Assurance Tests

NECEL's life tests consist of the high-temperature operating/bias life (HTOL/HTB), the high-humidity storage life (HHSL), the high-temperature, high-humidity (T/H = HHSL + bias), and the high-temperature storage life (HTSL). Additionally, NECEL performs various environmental and mechanical tests.

HTOL/HTB Test. These tests are used to accelerate failure mechanisms by operating devices in a dynamic (operating life) or static (bias) condition at an elevated temperature of 125°C. The data obtained is translated to a lower temperature to estimate device life expectancy using the Arrhenius relationship explained later.

HHSL and T/H Tests. Integrated circuits are extremely sensitive to the effects of humidity such as electrolytic corrosion between biased lines. The high-temperature and high-humidity tests are performed to detect failure mechanisms accelerated by temperature and humidity, such as leakage related problems and drifts in device parameters due to process instability.

HTSL Test. Another common test is the high-temperature storage life test in which devices are subjected to elevated temperatures with no applied bias. This test is used to detect process instability and stress migration problems.

Environmental Tests. Other environmental tests such as the pressure cooker test (PCT) or the temperature cycling test (T/C) detect problems related to the package and/or interactions between materials as well as the degradation of environmentally sensitive device characteristics.

Failure Rate Calculation/Prediction

To predict the device failure rate from accelerated life test data, the activation energies of the failure mechanisms involved should be considered. In some cases, an average activation energy is assumed to accomplish a quick first-order approximation. NECEL assumes an average activation energy of 0.7 eV or 0.45 eV for most products (0.3 eV for high-density memory devices). These values have been assessed from extensive reliability test results and yield conservative failure rates.

Since most semiconductor failures are temperature dependent, the Arrhenius relationship is used to normalize failure rate predictions at a system operation temperature of 55°C. It assumes that temperature dependence is an exponential function that defines the probability of failure occurrence, and that degradation of a performance parameter is linear with time. The Arrhenius model includes the effects of temperature and activation energies of the failure mechanisms in the following Arrhenius equation:

$$A = \exp \frac{-E_A(T_{J1}-T_{J2})}{k(T_{J1})(T_{J2})}$$

Where:
A(T) = Acceleration factor
E_A = Activation energy
T_{J1} = Junction temperature (in K) at T_{A1} = 55°C
T_{J2} = Junction temperature (in K) at T_{A2} = 125°C
k = Boltzmann's constant = 8.62×10^{-5} eV/K

Because the thermal resistance and power dissipation of a particular device type cannot be ignored, junction temperatures (T_{J1} and T_{J2}) are used instead of ambient temperatures (T_{A1} and T_{A2}). We calculate junction temperatures using the following formula:
$T_J = T_A$ + (thermal resistance)(power diss. at T_A)

With this information, a temperature acceleration factor can be calculated.

In some cases, the effect of voltage acceleration on failure rate must also be considered. Voltage acceleration can be characterized by the following equation:
$A(V) = \exp [-\beta(Vd - Vs)]$

Where:
Vd = Operating voltage (5.5 V)
Vs = Life test stress voltage (7 V)
β = Empirically determined constant (dependent on electric field constant and oxide thickness)

The constant β has been given the value $\approx$ 1, which is a conservative figure. Therefore, the overall acceleration factor will be determined as the product:
$A(T,V) = A(T) * A(V)$

To estimate long-term failure rate, the acceleration factor must be multiplied by the actual time to determine the simulated test time. From the high-temperature operating or bias life test results, failure rates can then be predicted at a 60% confidence level using the following equation:

$$L = \frac{(X^2)10^5}{2T}$$

Where:
L = Failure rate in %/1000 hours
X^2 = The tabular value of chi-squared distribution at a given confidence level and calculated degrees of freedom (2f + 2, where f = number of failures) See note below.
T = # of equivalent device hours = (# of devices) x (# of test hours) x (acceleration factor)

Note: Since the failures of concern here are the long-term failures, not the infant mortality failures (that is, the end of the downward slope and the middle constant section of the bathtub curve in figure 4), X^2 is determined by assuming a one-sided, fixed time test.

Another method of expressing failures is in FITs (failures in time). One FIT is equal to one failure in 10^9 hours. Since L is already expressed as %/1000 hours (10^{-5} failure/hr), an easy conversion from %/1000 hours to FIT would be to multiply the value of L by 10^4.

To accurately determine this failure rate, a statistically large sample size must be accumulated. Depending on the accuracy needed, the following conditions should be imposed:

- A minimum of 1.2 million device hours (equal to sample size multiplied by test period) at 125°C should be accumulated to accurately predict a failure rate of 0.02% per 1000 hours at 55°C, with a 60% confidence level.
- A minimum of 3 million device hours at 125°C should be accumulated to accurately predict a failure rate of 0.01% per 1000 hours at 55°C, with a 60% confidence level.

Failure Rate Calculation Example. As an example of how this failure rate is calculated, assume a sample of 960 pieces was subjected to 1000 hours at 125°C burn-in. One reject was observed. Given that the acceleration factor was calculated to be 34.6 using the Arrhenius equation, what is the failure rate normalized to 55°C using a confidence level of 60%. Express the failure rate in FITs.

Solution:

For $n = 2f + 2 = 2(1) + 2 = 4$, $X^2 = 4.046$

$$\text{Then } L = \frac{(X^2)10^5}{2T} \text{ (\%/1000 hours)}$$

$$= \frac{(X^2)10^5 \text{ (\%/1000 hours)}}{2(\#\text{ devices})(\#\text{ test hours})(\text{accel. factor})}$$

$$= \frac{(4.046)10^5}{2(960)(1000)(34.6)} = 0.0061 \text{ (\%/1000 hours)}$$

Therefore, $FIT = (0.0061)(10^4) = 61$

Failure Rate Goals

Outgoing electrical and mechanical quality levels, as well as mortality and long-term failure rates, are monitored and checked against quality and reliability targets. Long-term failure rate goals are based on mask and process designs. NECEL's quality and reliability targets are listed in tables 2 and 3.

Table 2. NECEL Quality Targets

	Outgoing Electrical (PPM)							Outgoing Mechanical (PPM)						
					BiPolar							BiPolar		
Year	Memory	Micro	System Micro	CMOS ASIC	RAM	ECL G/A	BiCMOS G/A	Memory	Micro	System Micro	CMOS ASIC	RAM	ECL G/A	BiCMOS G/A
1993	10	60	50	50	80	300	80	10	60	50	50	80	300	80
1994	3.4	40	40	10	80	300	80	3.4	40	40	10	80	300	80
1995	3.4	40	30	5	80	150	50	3.4	40	30	5	80	150	50

Table 3. NECEL Reliability Targets

	Infant Mortality (FIT)							Long-Term Reliability (FIT)						
					BiPolar							BiPolar		
Year	Memory	Micro	System Micro	CMOS ASIC	RAM	ECL G/A	BiCMOS G/A	Memory	Micro	System Micro	CMOS ASIC	RAM	ECL G/A	BiCMOS G/A
1993	10	40	50	100	50	300	80	10	30	100	50	30	300	80
1994	3.4	30	40	50	50	300	80	3.4	20	100	10	30	300	80
1995	3.4	30	30	10	50	150	50	3.4	20	100	5	30	150	50

FAILURE ANALYSIS

At NECEL, failure analysis is performed not only on reliability testing and field failures, but also on products that exhibit defects during production. This data is closely checked for correlation process quality information, inspection results, and reliability test data. Information derived from these failure analyses is fed back into the process.

Since many failure mechanisms can be exhibited by LSI devices, highly advanced analytical tools and methodologies are required to investigate such LSI failures in detail. The standard failure analysis flowchart relating to the returned products from customers is shown in appendix 4.

Special Grade Devices

Some applications require a wider temperature range and/or higher reliability than most, such as medical or safety equipment, transportation control systems, etc. For these requirements, NEC offers special grade devices based on a mutual quality agreement. The typical differences between special and standard grade devices are shown in table 4. NEC's quality and reliability targets for grade (A) microprocessor/controller products are shown in table 5.

SUMMARY

Building quality and reliability into products by forming a total quality control system is the most efficient way to ensure product success.

The combination of building quality into products, effective prescreening of potential failures, and monitoring of reliability through extensive testing has established a singularly high standard for NECEL's large-scale integrated circuits.

The company's quality control program supports research and development activities, failure analyses, and process improvements. With this extensive program, NECEL continuously sets and maintains higher standards of quality and reliability.

Table 4. Standard Grade and Special Grade Differences

Item	Standard Grade	Special Grade
Reliability Evaluation		
HTB, T/H HTS	> 1000 hours	> 2000 hours
T/C	> 100 cy	> 300 cy
PCT	> 96 hours	> 192 hous
Quality	Standard quality control steps	Special control when necessary
Screening	Standard burn-in	Increased burn-in time
Electrical testing	Standard	Addition of high temperature testing (if not performed already)
Storage life	3 years	5 years

Table 5. Grade (A) Micro Reliability and Quality Targets

Quality/Rel. Item	1993	1994	1995
Outgoing electrical (PPM)	10	3.4	3.4
Outgoing mechanical (PPM)	10	3.4	3.4
Infant mortality (FIT)	10	5	5
Long term rel. (FIT)	10	5	5

Appendix 1A. Typical QC Flow for CMOS Fabrication

Wafer Fabrication Process QC Flow (CMOS)

Flow	Process Material	In-process Inspection/quality Monitor
	Silicon Wafer	
	Incoming Inspection	Resistivity (sampling by lot) Dimension (sampling by lot) Visual (sampling by lot)
	Well Formation	
	Oxidation Photo Lithography	Oxide thickness (sampling by lot) Alignment and etching accuracy (sampling by lot)
	Ion Implantation	Layer resistance (sampling by lot)
	Field Formation	
	Deposition Photo Lithography	Deposit thickness (sampling by lot) Alignment and etching accuracy (sampling by lot)
	Oxidation	Oxide thickness (sampling by lot)
	Channel Stopper Formation	
	Photo Lithography	Alignment and etching accuracy (sampling by lot)
	Ion Implantation Oxidation	Layer resistance (sampling by lot) Oxide thickness (sampling by lot)
	Gate Formation	
	Deposition Doping Photo Lithography	Deposit thickness (sampling by lot) Layer resistance (sampling by lot) Alignment and etching accuracy (sampling by lot) Gate electrode width (sampling by lot)
	p/n SD Formation	
	Photo Lithography	Alignment and etching accuracy (sampling by lot)
	Ion Implantation Anneal	Layer resistance (sampling by lot)
	Contact Hole	
	Deposition Photo Lithography	Deposit thickness (sampling by lot) Alignment and etching accuracy (sampling by lot)
	Metallization	
	Metal Deposition Photo Lithography	Metal thickness (sampling by lot) Alignment and etching accuracy (sampling by lot)
	Alloy	Parametric test (sampling by lot)
	Passivation	
	Deposition Photo Lithography	Deposit thickness (sampling by lot) Alignment and etching accuracy (sampling by lot)
	Wafer Sort	Electrical test

Contact hole and metallization steps are repeated twice.

83RD-7509B

Appendix 1B. Typical QC Flow for PLCC Assembly/Test

Process/Materials	Inspection of Manufacturing Conditions				Inspection of Manufacturing Qualities			
	Inspection Item	Frequency	Instrument	Inspected by	Inspection Item	Frequency	Instrument	Inspected by
1 Sorted Wafers								
2 Wafer Visual					Wafer Visual	100%	Naked Eye	Operator
3 Dicing	Table Speed DI Water Blade Height	Every Shift	Indicators Gauges	P.C.	Sawing Dimensions	Before Running	Microscope With Filter Eyepiece	Operator
4 Break and Expand	Wafer Break Conditions Wafer Expand Conditions	Every Shift	Indicators Gauges	P.C.	Wafer Visual	100%	Naked Eye	Operator
5 Die Visual Inspection					Die Visual	Every Lot Sampling (Or 100%)	Microscope	Operator
6 Lead Frames	Die Attached Conditions	Every Shift	Indicators Thermocouple, Potentiometer	P.C.	Die Visual Epoxy Coverage	Every Magazine	Naked Eye	Operator
7 Die Attached	Temperature					Every Shift	Microscope	
8 Epoxy Cure (Not Done for Gold Die Attached product)	Heat Temperature N_2 Flow	Every Shift	Indicators Gauges	P.C.	Shear Strength	Every Shift	Dynamometer	Operator
9 Fine Wire	Bonding Conditions	Every Shift	Indicators	P.C.	Visual	Every Magazine	Microscope	Operator
10 Wire Bonding	Temperature	Every Week	Thermocouple and Potentiometer	P.C.	Wire Pull Test	Every Shift	Tension Gauge	Operator
11 Pre-Seal Visual Inspection					Die Visual	Every Lot Sampling (Or 100%)	Microscope	Inspector
12 Molding Compound	Temperature of Pellet, Expiration Date	Every Shift	Thermocouple	P.C.				
13 Molding	Temperature Profile of Die Set Preheat Temperature Pressure Cure Time	Every Shift	Thermocouple, Potentiometer	P.C.	Visual	100%	Naked Eye	Operator
14 Mold Aging	Temperature	Every Shift	Indicator	P.C.				
15 Deflashing	Deflashing Conditions	Every Shift	Indicators	P.C.	Visual	Every Lot	Naked Eye	Operator
	Concentration	Every Week	Titration	Tech.				
	Density	Every Week	Density Meter	Tech.				
	Water Jet Pressure	Every Day	Gauge	Tech.				
16 Plating	Plating Conditions	Every Day	Indicators	P.C.				
	Concentration	Every Week	Titration	Tech.				

83RD-7515B

Appendix 1B. Typical QC Flow for PLCC Assembly/Test (cont)

Process/Materials	Inspection of Manufacturing Conditions				Inspection of Manufacturing Qualities			
	Inspection Item	Frequency	Instrument	Inspected by	Inspection Item	Frequency	Instrument	Inspected by
17 Plating Inspection					Visual	Every Lot	Naked Eye	Technician
					Plating Thickness	Every Lot	X-ray	Technician
					Composition	Every Lot	X-ray	Technician
					Solderability	Once/Day	Naked Eye	Technician
18 Marking Ink	Marking Conditions	Every Shift	Indicators	P.C.	Visual	Every Lot	Naked Eye	Operator
19 Marking								
20 Mark Cure	Temperature	Every Shift	Thermocouple	P.C.	Marking Permanency	Twice/Shift	Automatic Tester	Operator
21 Lead Forming	Dimensions	Every Shift (Before Running)	Test Jig. Caliper	Operator	Visual	Every Lot	Naked Eye	Operator
22 Final Assembly Inspection					Visual	Every Lot	Magnifying Lamp	Operator
23 First Electrical Sorting	P.M. Check	Every Day	P.M. Jig.	Operator				
	Sample Check	Before Testing	Test Samples	Operator	Electrical Characteristics	100%	IC Tester	Operator
24 Burn-In (When Necessary)	Burn-In Conditions	Every Batch	Indicator	P.C.				
25 First Electrical Sorting		Every Day	P.M. Jig.	Operator				
		Before Testing	Test Samples	Operator	Electrical Characteristics	100%	IC Tester	Operator
26 Reliability Assurance Test		Every Month						
27 In-Warehouse Inspection		Every Day	P.M. Jig.		Electrical Characteristics	Every Lot	IC Tester	Inspector
		Before Testing	Test Samples		Visual (Major)	Every Lot	Naked Eye and Microscope	Inspector
					Visual (Minor)	Every Lot	Naked Eye	Inspector
28 Warehousing								

83RD-7516B

Appendix 2. Typical Reliability Assurance Tests

Test	Symbol	MIL-STD-883C Method	Test Conditions
High-temperature operating/bias life (Note 1)	HTOL/HTB	1005	T_A = 125°C; V_{DD} specified per device type
High-temperature storage life (Note 1)	HTSL	1008	T_A = 150°C (175° or 200°C in some cases)
High-temperature/high-humidity (Note 1)	T/H		T_A = 85°C; RH = 85%; V_{DD} = 5.5 V
High-humidity storage life (Note 1)	HHSL		T_A = 85°C; RH = 85%
Pressure cooker (Note 1)	PCT		T_A = 125°C; P = 2.3 atm; RH = 100%
Temperature cycling (Note 1)	T/C	1010	−65°C to +150°C; 1 hour/cycle
Lead fatigue (Note 2)	C3	2004	90-degree bends; 3 bends without breaking
Solderability (Note 3)	C4	2003	230°C; 5 sec; rosin base flux
Soldering heat/temperature cycle/ thermal shock (Note 1)	C6	1010 1011 (Note 4)	10 sec @ 230°C; rosin base flux Ten 1-hour cycles @ −65°C to +150°C Fifteen 10-minute cycles @ 0°C to +100°C

Notes:

(1) Electrical test per data sheet is performed. Devices that exceed the data sheet limits are considered rejects.

(2) Broken lead is considered a reject.

(3) Less than 95% coverage is considered a reject.

(4) MIL-STD-750A, method 2031.

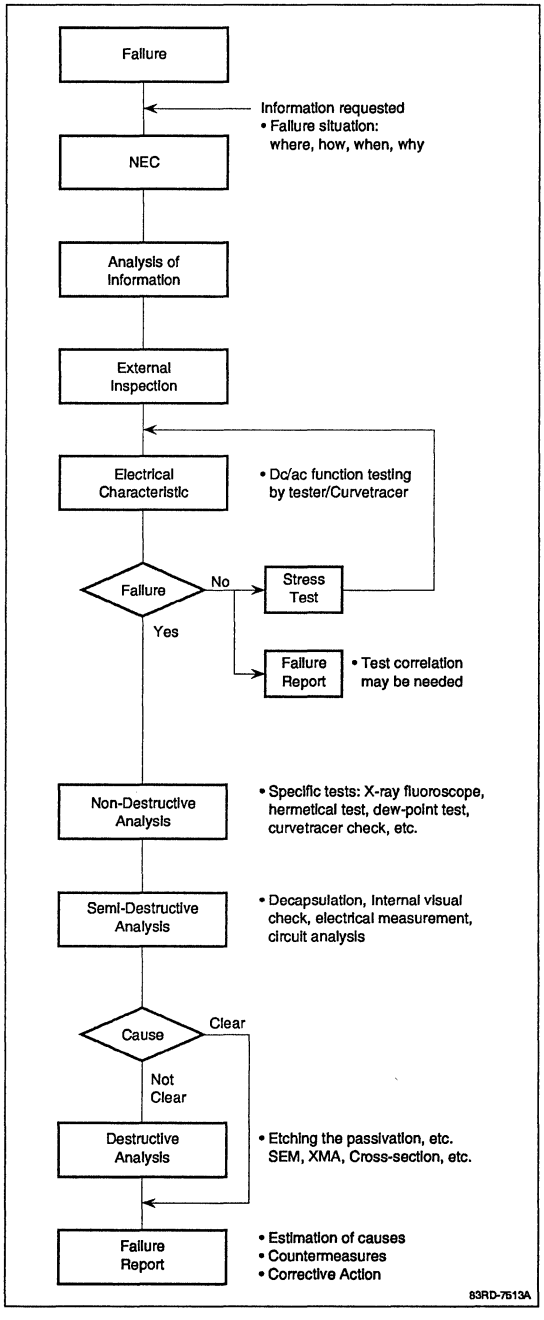
Appendix 3. New Product/Process Change Tests

Test	Sample Size	Newly Developed Product	Shrink Die	New Package	Wafer	Assembly	Test Conditions
High-temperature operating/bias life	20 - 50 pieces; 1 - lots	0	0	0	0	0	See appendix 2; 1000H
High-temperature storage life	10 - 20 pieces; 1 - 3 lots	0	0	0	0	0	T = 150°C (plastic); T = 175°C (ceramic); 1000H
High-temperature/ high-humidity bias life (plastic package)	20 - 50 pieces; 1 - 3 lots	0	0	0	0	0	See appendix 2; 1000H
Pressure cooker (plastic package)	10 - 20 pieces; 1 - 3 lots	0	0	0	0	0	See appendix 2; 288H
Thermal environmental	10 - 20 pieces; 1 - 3 lots	0	X	0	X	0	See appendix 2
Mechanical environmental (ceramic package)	10 - 20 pieces; 1 - 3 lots	0	X	0	X	0	20G, 10 - 2000Hz; 1500G, 0.5 ms; 20000G, 1 min
Lead fatigue	5 pieces; 1 - 3 lots	X	–	X	–	X	See appendix 2
Solderability	5 pieces; 1 - 3 lots	X	–	X	–	X	See appendix 2
ESD	20 pieces; 1 - 3 lots	0	0		0	X	(1) C = 200 pF, R = 0 (2) C = 100 pF, R = 1.5 k
Long term T/C	10 - 50 pieces; 1 - 3 lots	0	0	0	0	0	See appendix 2; 1000 cy

Notes:

0: Performed. X: Perform if necessary. –: Not performed.

Appendix 4. Failure Analysis Flowchart



NEC

Section 2
μPD78C00 Product Line
8-Bit, Single-Chip Microcontrollers

NEC
NEC Electronics Inc.

μPD78C14 Family (μPD78C10A/C11A/C12A/C14/C14A/CP14) 8-Bit, Single-Chip Microcontrollers With A/D Converter

September 1993

Description

This family of microcontrollers integrates sophisticated on-chip peripheral functions normally provided by external components. Their internal 16-bit ALU and data paths, combined with a powerful instruction set and addressing, make the devices appropriate in data processing as well as control applications.

The devices integrate a 16-bit ALU, 4K, 8K, or 16K-byte ROM, 256-byte RAM, an eight-channel A/D converter, a multifunction 16-bit timer/event counter, two 8-bit timers, a USART, and two zero-cross detect inputs on a single die, allowing their use in fast, high-end processing applications. This involves analog signal interface and processing.

The μPD78C14 family includes: 4K, 8K, and 16K-byte mask ROM devices, embedded with a custom customer program; ROMless devices for use with up to 64K-bytes of external memory; and 16K-byte EPROM or OTP ROM devices for prototyping and low-volume production. The μPD78C11A/C12A/C14A also have mask optional pullup resistors available on ports A, B, and C.

Features

- CMOS technology
  - 25 mA operating current (78C10A/C11A/C12A)
  - 30 mA operating current (78C14/C14A)
- Complete single-chip microcontroller
  - 16-bit ALU
  - 4K, 8K, or 16K x 8 ROM
  - 256-byte RAM
- 44 I/O lines
- Mask optional pullup resistors
  - Ports A, B, and C (μPD78C11A/C12A/C14A only)
- Two zero-cross detect inputs
- Two 8-bit timers
- Expansion capabilities
  - 8085A-like bus
  - 60K-byte external memory address range
- Eight-channel, 8-bit A/D converter
  - Autoscan mode
  - Channel select mode
- Full-duplex USART
  - Synchronous and asynchronous
- 159 instructions
  - 16-bit arithmetic, multiply, and divide
  - HALT and STOP instructions
- 0.8-μs instruction cycle time (15-MHz operation)
- Prioritized interrupt structure
  - Three external
  - Eight internal
- Standby function
- On-chip clock generator

Ordering Information

Part Number (Note 1)	Package	Package Drawing	Quality Grade (Note 3)
ROMless			
μPD78C10ACW	64-pin SDIP	P64C-70-750A, C	Standard
AGF-3BE	64-pin QFP (Note 2)	P64GF-100-3B8, 3BE-1	Standard
AGF(A)-3BE			Special
AGQ-36	64-pin QUIP	P64GQ-100-36	Standard
AGQ(A)-36			Special
AL	68-pin PLCC	P68L-50A1-1	Standard
AL(A)			Special
4K Mask ROM			
μPD78C11ACW-xxx	64-pin SDIP	P64C-70-750A, C	Standard
AGF-xxx-3BE	64-pin QFP (Note 2)	P64GF-100-3B8, 3BE-1	Standard
AGF(A)-xxx-3BE			Special
AGQ-xxx-36	64-pin QUIP	P64GQ-100-36	Standard
AGQ(A)-xxx-36			Special
AGQ-xxx-37	64-pin QUIP (straight)	P64GQ-100-37	Standard
AL-xxx	68-pin PLCC	P68L-50A1-1	Standard
AL(A)-xxx			Special
8K Mask ROM			
μPD78C12ACW-xxx	64-pin SDIP	P64C-70-750A, C	Standard
AG-xxx-37	64-pin QUIP (straight)	P64GQ-100-37	Standard
AG-xxx-36	64-pin QUIP	P64GQ-100-36	Standard
AG(A)-xxx-36			Special
AGF-xxx-3BE	64-pin QFP (Note 2)	P64GF-100-3B8, 3BE-1	Standard
AL-xxx	68-pin PLCC	P68L-50A1-1	Standard
AL(A)-xxx			Special
16K Mask ROM			
μPD78C14AG-xxx-AB8	64-pin QFP	P64GC-80-AB8-2	Standard
CW-xxx	64-pin SDIP	P64C-70-750A, C	
G-xxx-36	64-pin QUIP	P64GQ-100-36	
G-xxx-37	64-pin QUIP (straight)	P64GQ-100-37	
G-xxx-1B	64-pin QFP (Note 2)	P64G-100-12, 1B-1	
GF-xxx-3BE	64-pin QFP (Note 2)	P64GF-100-3B8, 3BE-1	
L-xxx	68-pin PLCC	P68L-50A1-1	
16K OTP ROM			
μPD78CP14CW	64-pin SDIP	P64C-70-750A, C	Standard
G-36	64-pin QUIP	P64GQ-100-36	
G-37	64-pin QUIP (straight)	P64GQ-100-37	
GF-3BE	64-pin QFP (Note 2)	P64GF-100-3B8, 3BE-1	
L	68-pin PLCC	P68L-50A1-1	

Ordering Information (cont)

Part Number (Note 1)	Package	Package Drawing	Quality Grade (Note 3)
16K UV EPROM			
μPD78CP14DW	64-pin CER SDIP w/window	P64DW-70-750A	Standard
R	64-pin CER QUIP w/window	P64RQ-100-A	
16K OTP ROM			
μPD78CP14G(A)-36	64-pin QUIP	P64GQ-100-36	Special

Notes:

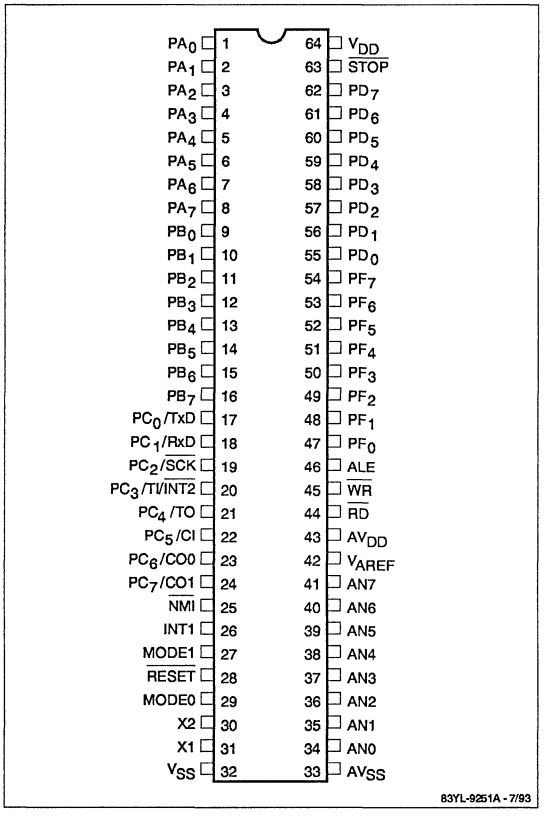
(1) xxx indicates ROM code suffix

(2) Engineering samples supplied in a ceramic QFP package

(3) Special grade devices have the symbol (A) embedded in the part number

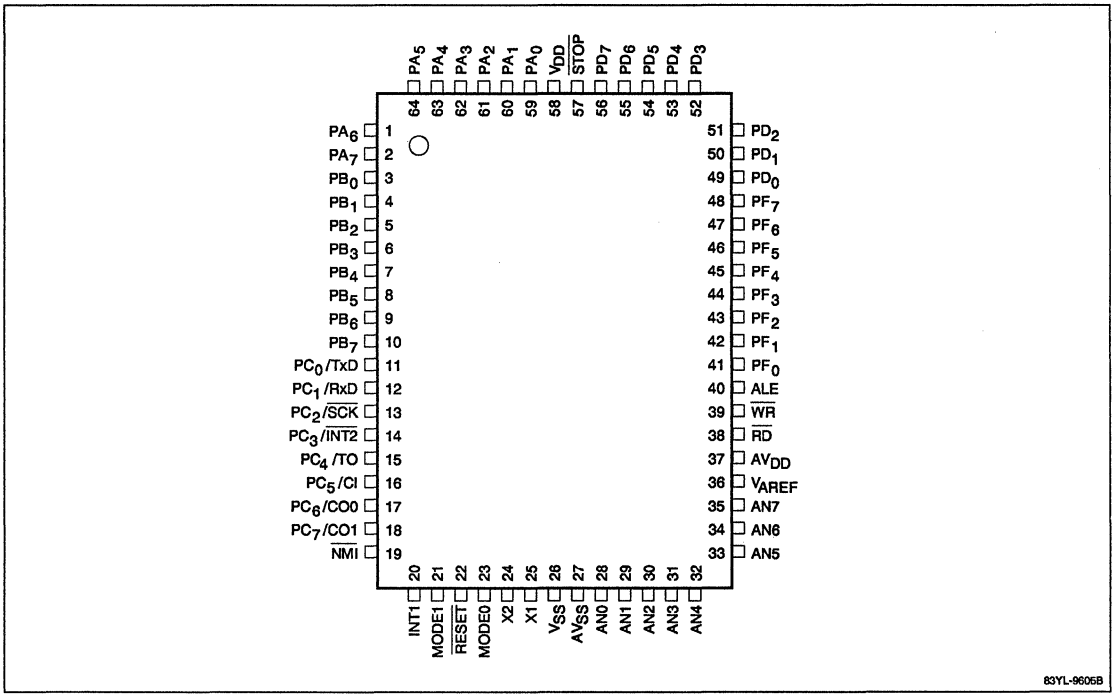
Pin Configurations

64-Pin QUIP or SDIP (Plastic or Ceramic)



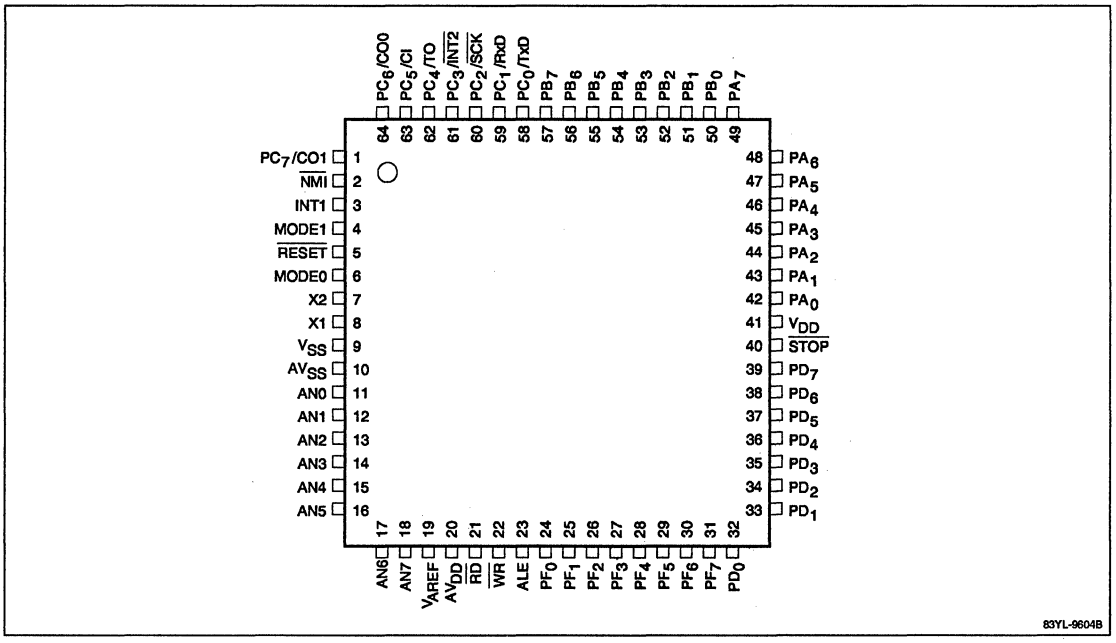
Pin Configurations (cont)

64-Pin QFP (20mm x 14mm)



Pin Configurations (cont)

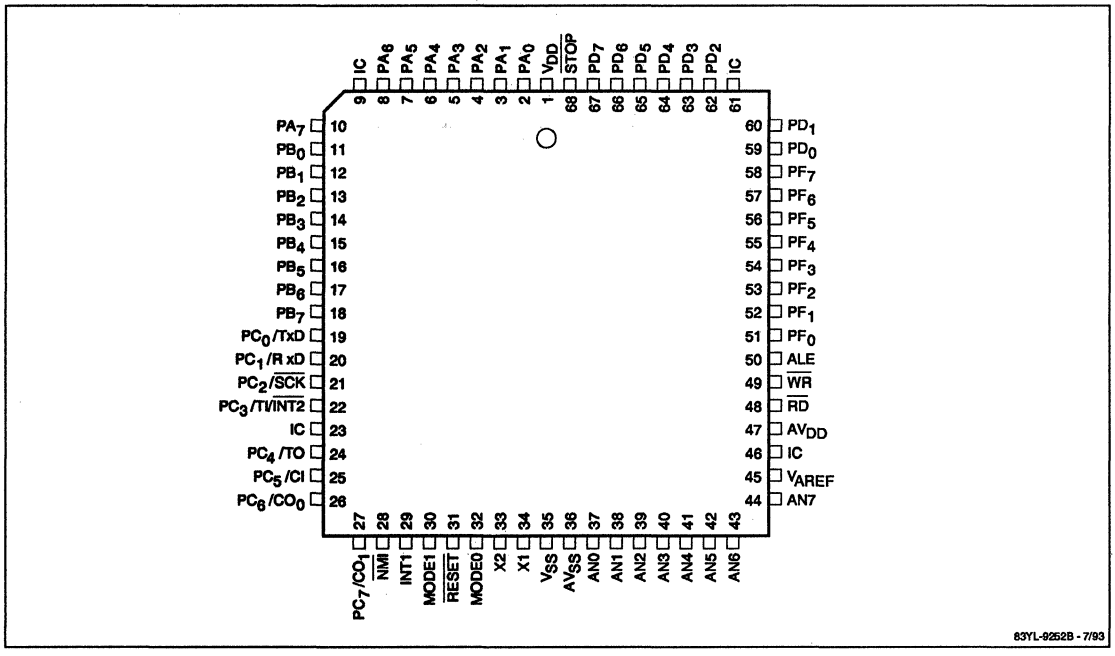
64-Pin QFP (14mm x 14mm)



2a

Pin Configurations (cont)

68-Pin PLCC



Pin Identification

Symbol	Function
ALE	Address latch enable output
AN0-AN7	A/D converter analog inputs 0-7
INT1	Interrupt request 1 input
MODE0	Mode 0 input; I/O memory output
MODE1	Mode 1 input
$\overline{NMI}$	Nonmaskable interrupt input
PA_0 - PA_7	Port A I/O
PB_0 - PB_7	Port B I/O
PC_0/TxD	Port C I/O line 0; transmit data output
PC_1/RxD	Port C I/O line 1; receive data input
$PC_2/\overline{SCK}$	Port C I/O line 2; serial clock I/O
PC_3/TI/$\overline{INT2}$	Port C I/O line 3; timer input; interrupt request 2 input
PC_4/TO	Port C I/O line 4; timer output
PC_5/CI	Port C I/O line 5; counter input
PC_6, PC_7/ CO_0,CO_1	Port C I/O lines 6, 7; counter outputs 0, 1
PD_0 - PD_7	Port D I/O; expansion memory address, data bus (bits AD_0 - AD_7)
PF_0 - PF_7	Port F I/O; expansion memory address, (bits AB_8 - AB_{15})
$\overline{RD}$	Read strobe output
$\overline{RESET}$	Reset input
$\overline{STOP}$	Stop mode control input
V_{AREF}	A/D converter reference voltage
$\overline{WR}$	Write strobe output
X1, X2	Crystal connections 1, 2
AV_{DD}	A/D converter power supply voltage
AV_{SS}	A/D converter power supply ground
V_{DD}	5 V power supply
V_{SS}	Ground
IC	Internal connection

PIN FUNCTIONS

ALE (Address Latch Enable). The ALE output is used to latch the address of PD_0 - PD_7 into an external latch.

AN0-AN7 (Analog Inputs). These are the eight analog inputs to the A/D converter. AN4-AN7 can also be used as a digital input for falling edge detection.

CI (Counter Input). External pulse input to timer/event counter.

CO_0, CO_1 (Counter Outputs). Programmable waveform outputs based on timer/event counter.

INT1 (Interrupt Request 1). INT1 is a rising edge triggered, maskable interrupt input. It is also an ac-input, zero-cross detection terminal.

If the optional pullup resistor is specified for this pin on the μPD78C11A/C12A/C14A, the zero-cross detection circuitry will not function.

$\overline{INT2}$ (Interrupt Request 2). $\overline{INT2}$ is a falling edge triggered, maskable interrupt input. It is also an ac-input, zero-cross detection terminal.

MODE0, MODE1 (Mode 0, 1). The MODE0 and MODE1 inputs select the amount of external memory. MODE0 outputs the $\overline{IO}$ signal and MODE1 outputs the $\overline{M1}$ signal. An external pullup resistor to V_{DD} is required if the input is to be a logic high.

The value of this pullup resistor, R, is dependent on t_{CYC} and is calculated as follows: R in KΩis $4 \leq R \leq 0.4$ t_{CYC} where t_{CYC} is in ns units.

$\overline{NMI}$ (Nonmaskable Interrupt). Falling edge, Schmitt-triggered nonmaskable interrupt input.

PA_0 - PA_7 (Port A). Port A is an 8-bit three-state port. Each bit is independently programmable as either input or output. Reset makes all lines of port A inputs. Mask optional pullup resistors are available on the μPD78C11A/C12A/C14A.

PB_0 - PB_7 (Port B). Port B is an 8-bit three-state port. Each bit is independently programmable as either input or output. Reset makes all lines of port B inputs. Mask optional pullup resistors are available on the μPD78C11A/C12A/C14A.

PC_0 - PC_7 (Port C). Port C is an 8-bit three-state port. Each bit is independently programmable as either input or output. Alternatively, the lines of port C can be used as control lines for the USART, interrupts, and timer. Reset makes all lines of port C inputs. Mask optional pullup resistors are available on the μPD78C11A/C12A/C14A.

PD_0 - PD_7 (Port D). Port D is an 8-bit three-state port. It can be programmed as either 8 bits of input or 8 bits of output. When external expansion memory is used, port D acts as the multiplexed address/data bus.

PF_0 - PF_7 (Port F). Port F is an 8-bit three-state port. Each bit is independently programmable as either input or output. When external expansion memory is used, port F outputs the high-order address bits.

$\overline{RD}$ (Read Strobe). The three-state $\overline{RD}$ output goes low to gate data from external devices onto the data bus. $\overline{RD}$ goes high during reset.

$\overline{RESET}$ (Reset). When the Schmitt-triggered $\overline{RESET}$ input is brought low, it initializes the device.

RxD (Receive Data). Serial data input terminal.

$\overline{SCK}$ (Serial Clock). Output for the serial clock when internal clock is used. Input for serial clock when external clock is used.

$\overline{STOP}$ (STOP Mode Control Input). A low-level input on $\overline{STOP}$ (Schmitt-triggered input) stops the system clock oscillator.

TI (Timer Input). Timer input terminal.

TO (Timer Output). The output of TO is a square wave with a frequency determined by the timer/counter.

TxD (Transmit Data). Serial data output terminal.

V_{AREF} (A/D Converter Reference). V_{AREF} sets the upper limit for the A/D conversion range.

$\overline{WR}$ (Write Strobe). The three-state $\overline{WR}$ output goes low to indicate that the data bus holds valid data. It is a strobe signal for external memory or I/O write operations. $\overline{WR}$ goes high during reset.

X1, X2 (Crystal Connections). X1 and X2 are the system clock crystal oscillator terminals. X1 is the input for an external clock.

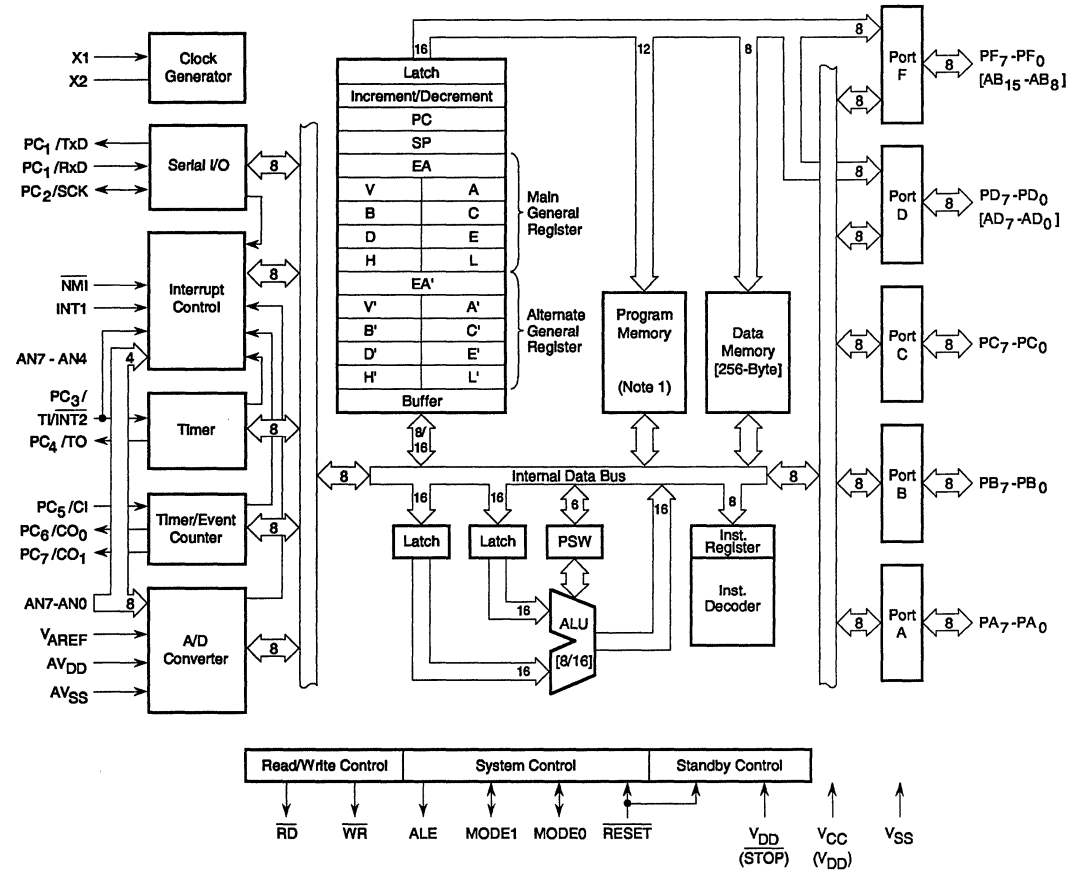
AV_{DD} (A/D Converter Power). This is the power supply voltage for the A/D converter.

AV_{SS} (A/D Converter Power Ground). AV_{SS} is the ground potential for the A/D converter power supply.

V_{DD} (Power Supply). V_{DD} is the +5-volt power supply.

V_{SS} (Ground). Ground potential.

Block Diagram



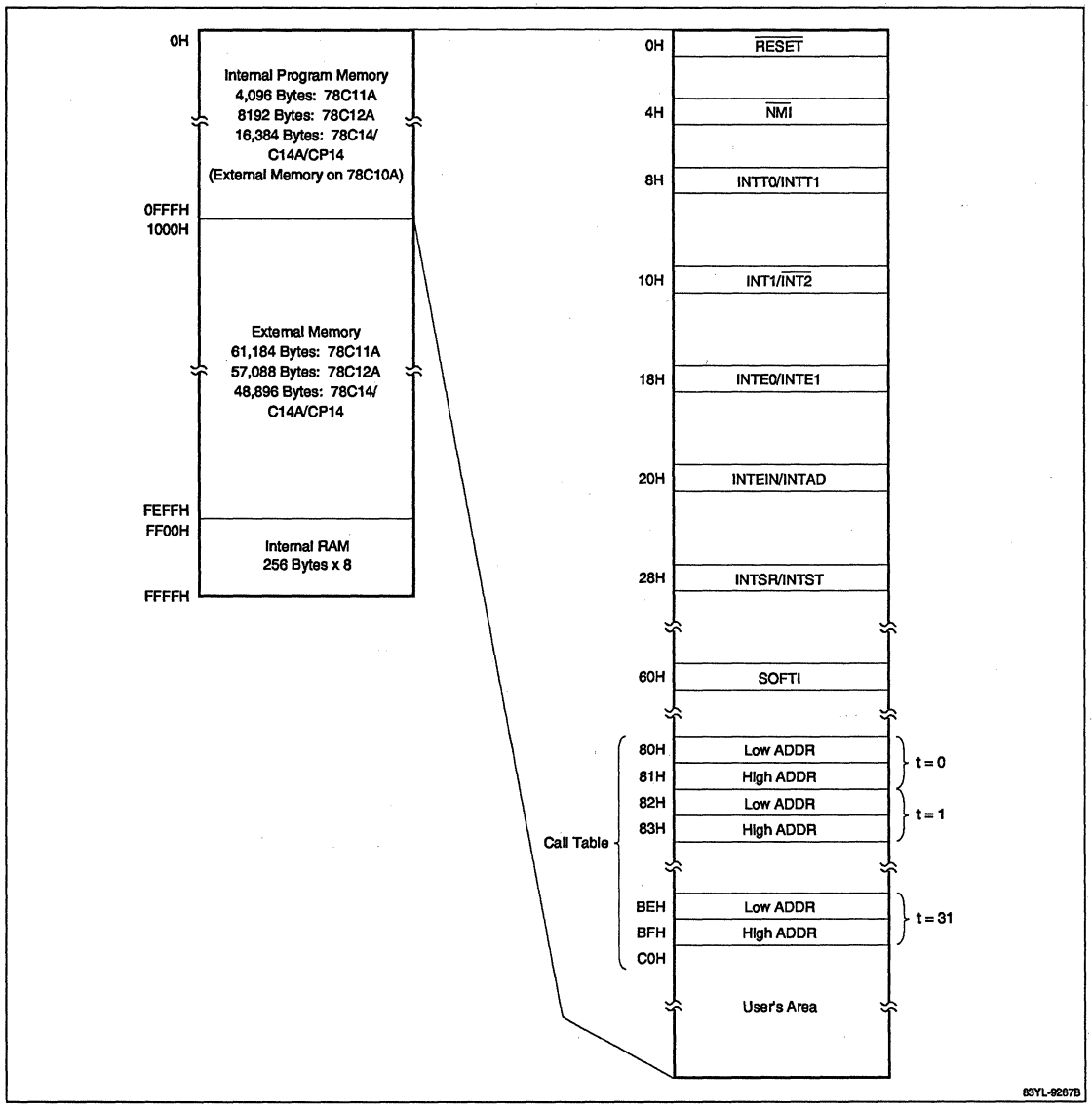
Note:

1. On-Chip ROM
   - 78C10A : 0
   - 78C11A : 4096 Bytes
   - 78C12A : 8192 Bytes
   - 78C14/C14A: 16384 Bytes
   - 78CP14 : 16384 Bytes EPROM/OTP ROM

83YL-9266B (9/93)

2a

Figure 1. Memory Map



FUNCTIONAL DESCRIPTION

Memory Map

The μPD78C14 family can directly address up to 64K bytes of memory. Except for the on-chip ROM (or PROM) and RAM (FF00H-FFFFH), any memory location can be used as ROM or RAM. The memory map, figure 1, defines the 0 to 64K-byte memory space for the μPD78C14 family.

The μPD78CP14 can be programmed in software to have 4K, 8K, or 16K-bytes of internal program memory. This programming is transparent to the ROM-based device, allowing easy transfer of code to a ROM-based device.

Input/Output

The μPD78C14 family has 44 digital I/O lines, five 8-bit ports (ports A, B, C, D, F), and four digital input lines (AN4-AN7).

Analog Input Lines. AN0-AN7 are configured as analog input lines for the on-chip A/D converter. Lines AN4-AN7 can be used as digital input lines for falling-edge detection.

Port A, Port B, Port C, Port F. Each line of these ports can be individually programmed as an input or output. When used as I/O ports, all have latched outputs and high-impedance inputs. On the μPD78C11A/C12A/C14A, mask optional pullup resistors are available for ports A, B, and C.

Port D. Port D can be programmed as a byte input or a byte output.

Control Lines. Under software control, each line of port C can be configured individually as a control line for the serial interface, timer, and timer/counter or as an I/O port.

Memory Expansion. In addition to the single-chip operation mode, the μPD78C14 family has four memory expansion modes. Under software control, port D can provide a multiplexed low-order address and data bus; port F can provide a high-order address bus. Table 1 shows the relation between memory expansion modes and the pin configurations of port D and port F.

Table 1. Memory Expansion Modes and Port Configurations

Memory Expansion	Port	Port Configuration
None	Port D	I/O port
	Port F	I/O port
256 bytes	Port D	Multiplexed address/data bus
	Port F	I/O port
4K bytes	Port D	Multiplexed address/data bus
	Port F (PF_0-PF_3)	Address bus
	Port F (PF_4-PF_7)	I/O port
16K bytes	Port D	Multiplexed address/data bus
	Port F (PF_0-PF_5)	Address bus
	Port F (PF_6-PF_7)	I/O port
60K bytes	Port D	Multiplexed address/data bus
	Port F	Address bus

Timers

The two 8-bit timers may be programmed independently or cascaded as a 16-bit timer. The timer can be software set to increment at intervals of four machine cycles (0.8 μs at 15-MHz operation) or 128 machine cycles (25.6 μs at 15 MHz), or to increment on receipt of a pulse at TI. Figure 2 is the block diagram for the timer.

Timer/Event Counter

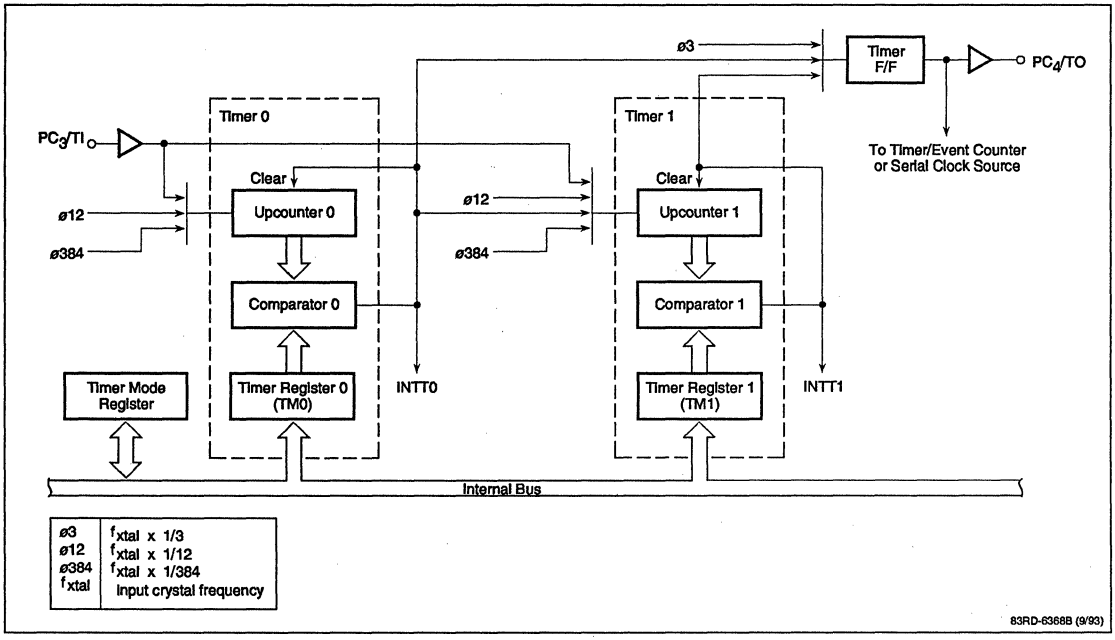
The 16-bit multifunctional timer/event counter (figure 3) can be used for the following operations:

- Interval timer
- External event counter
- Frequency measurement
- Pulse-width measurement
- Programmable frequency and duty cycle waveform output
- Single-pulse output

8-Bit A/D Converter

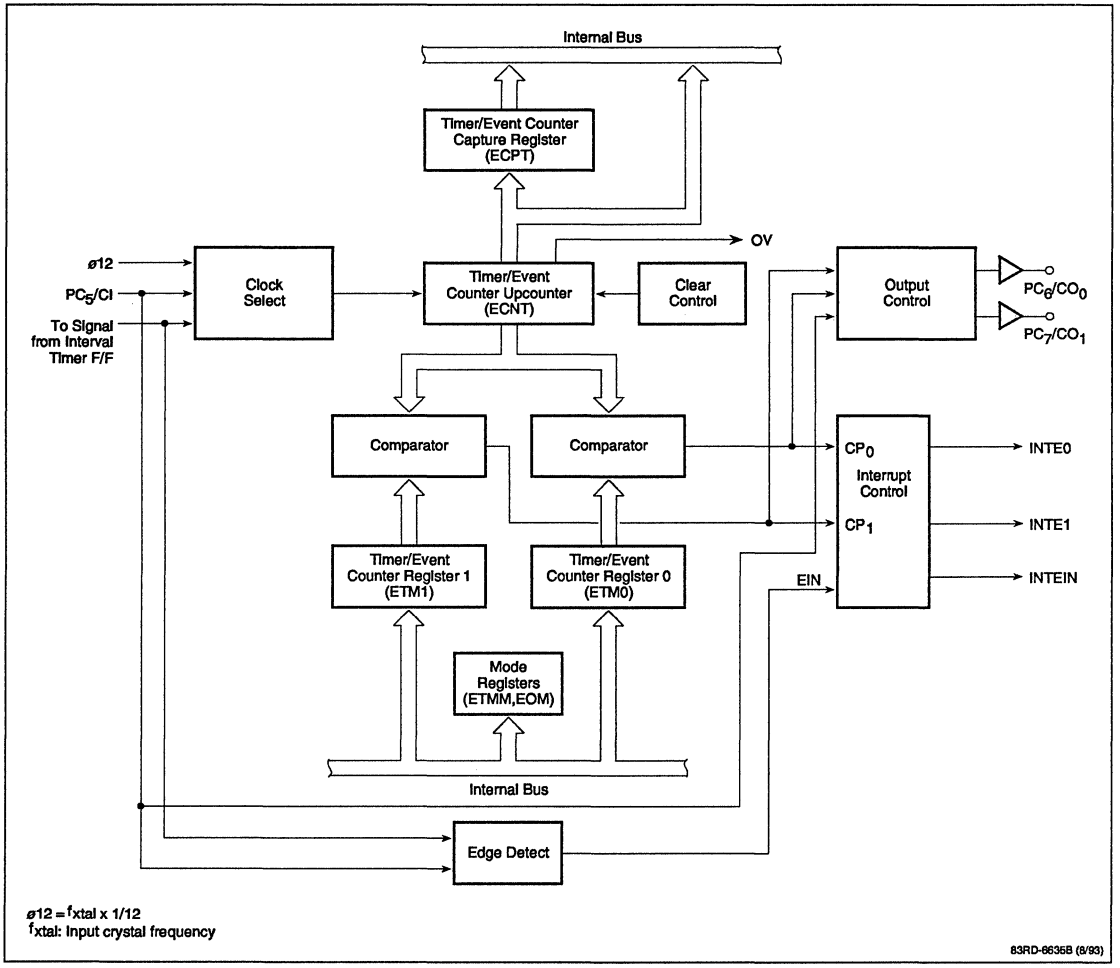
- Eight input channels
- Four conversion result registers
- Two powerful operation modes
  - — Autoscan mode
  - — Channel select mode
- Successive approximation technique
- Absolute accuracy: 0.6% FSR ± 1/2 LSB
- Conversion range: 0 to 5 V
- Conversion time: 38.4 μs
- Interrupt generation

Figure 2. Timer Block Diagram



2a

Figure 3. Block Diagram for the Timer/Event Counter



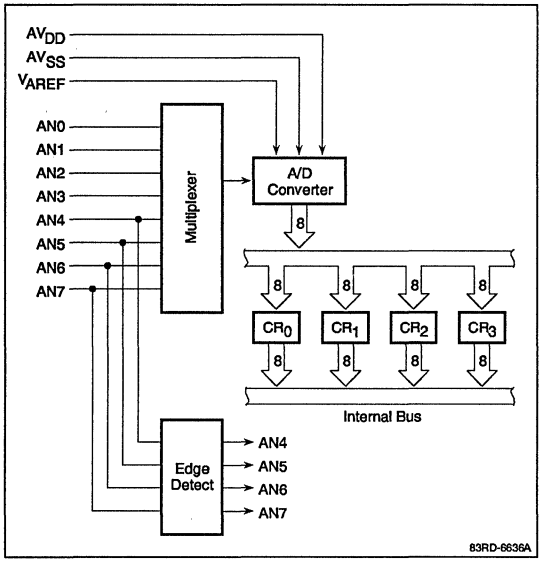
Analog/Digital Converter

The μPD78C14 family features an 8-bit, high-speed, high accuracy A/D converter. The A/D converter is made up of a 256-resistor ladder and a successive approximation register (SAR). There are four conversion result registers (CR0-CR3).

The eight-channel analog input may be operated in either of two modes. In the select mode, the conversion value of one analog input is sequentially stored in CR0-CR3. In the scan mode, either the upper four channels or the lower four channels may be specified. Then those four channels will be consecutively selected and the conversion results stored sequentially in the four conversion result registers.

Figure 4 is the block diagram for the A/D converter. To stop the operation of the A/D converter and thus reduce power consumption, set V_{AREF} = 0 V.

Figure 4. A/D Converter Block Diagram



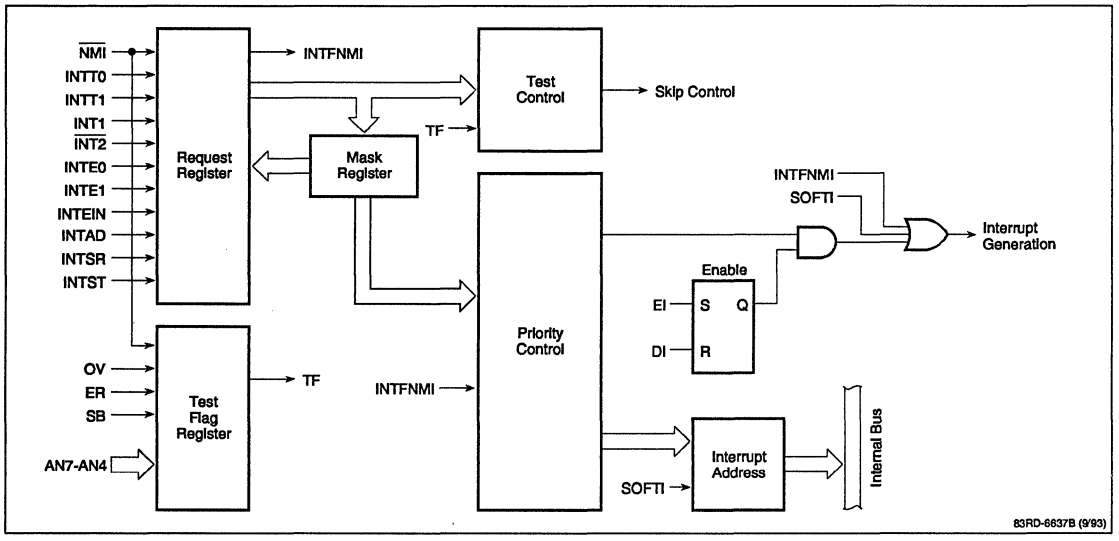
Interrupt Structure

There are 12 interrupt sources in the μPD78C14 family of chips. Three are external interrupts and nine are internal. Table 2 shows 11 interrupt sources divided into seven priority levels where IRQ0 is the highest and IRQ6 is the lowest. See figure 5.

Table 2. Interrupt Sources

Interrupt Request	Interrupt Address	Type of Interrupt	Internal/External
IRQ0	4	$\overline{NMI}$ (Nonmaskable interrupt)	External
IRQ1	8	INTT0, INTT1 (Coincidence signals from timers 0, 1)	Internal
IRQ2	16	INT1, $\overline{INT2}$ (Maskable interrupts)	External
IRQ3	24	INTE0, INTE1 (Coincidence signals from timer/event counter)	Internal
IRQ4	32	INTEIN (Falling signal of CI or TO into the timer/event counter)	Internal or External
		INTAD (A/D converter interrupt)	Internal
IRQ5	40	INTSR (Serial receive interrupt)	Internal
		INST (Serial send interrupt)	
IRQ6	96	SOFTI instruction	Internal

Figure 5. Interrupt Structure Block Diagram



Standby Functions

The μPD78C14 family has two standby modes: HALT and STOP. The HALT mode reduces power consumption to 50% of normal operating requirements, while maintaining the contents of on-chip registers, RAM, and control status. The system clock and on-board peripherals continue to operate, but the CPU stops executing instructions. The HALT mode is initiated by executing the HLT instruction. The HALT mode can be released by any nonmasked interrupt or by $\overline{\text{RESET}}$.

The STOP mode reduces power consumption to less than 0.1% of normal operating requirements. There are two STOP modes: type A and type B.

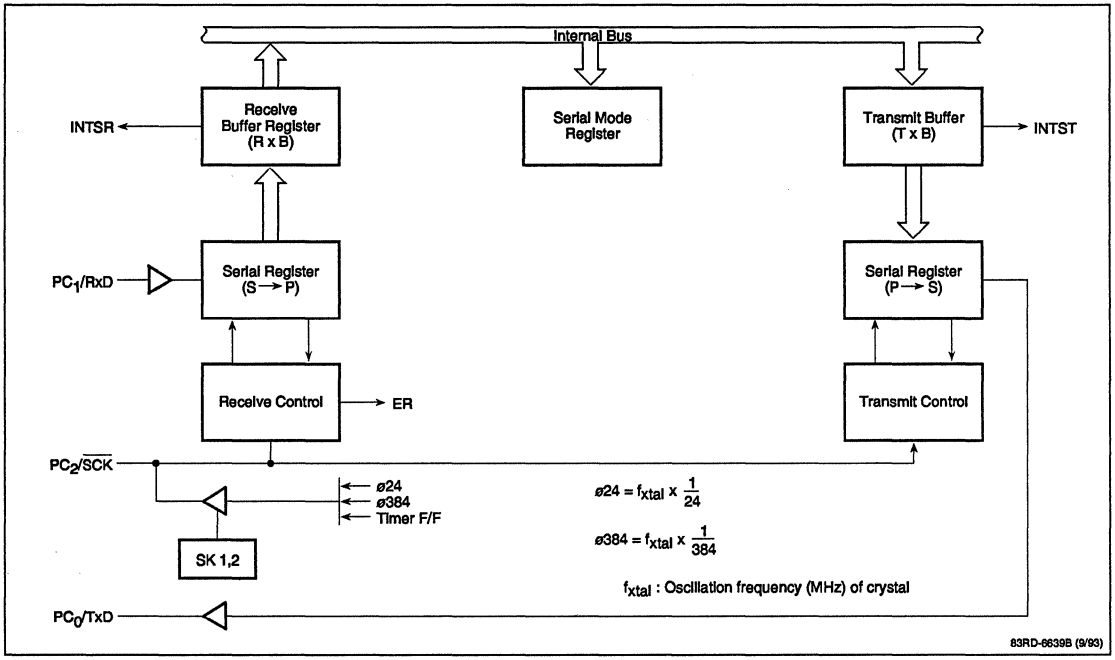
Type A is initiated by executing a STOP instruction. If V_{DD} is held above 2.5 V, the on-board RAM is saved. The oscillator is stopped. The STOP mode can be released by an input on $\overline{\text{NMI}}$ or $\overline{\text{RESET}}$. The user can program oscillator stabilization time up to 52.4 ms via timer 1. By checking the standby flag (SB), the user can determine whether the processor has been in the standby mode or has been powered up.

Type B is initiated by inputting a low level on the $\overline{\text{STOP}}$ input. The RAM contents are saved if V_{DD} is held above 2.5 V. The oscillator is stopped. The STOP mode is released by raising $\overline{\text{STOP}}$ to a high level. The oscillator stabilization time is fixed at 52.4 ms; 52.4 ms after $\overline{\text{STOP}}$ is raised, instruction execution will automatically begin at location 0. The stabilization time can be increased by holding $\overline{\text{RESET}}$ low for the required time period.

Universal Serial Interface

The serial interface can operate in one of three modes: synchronous, asynchronous, and I/O interface. The I/O interface mode transfers data MSB first, for easy interfacing to certain NEC peripheral devices. Synchronous and asynchronous modes transfer data LSB first. Synchronous operation offers two modes of data reception: search and nonsearch. In the search mode, data is transferred one bit at a time from the serial register to the receive buffer. This allows a software search for a sync character. In the nonsearch mode, data transfer from the serial register to the transmit buffer occurs eight bits at a time. Figure 6 shows the universal serial interface block diagram.

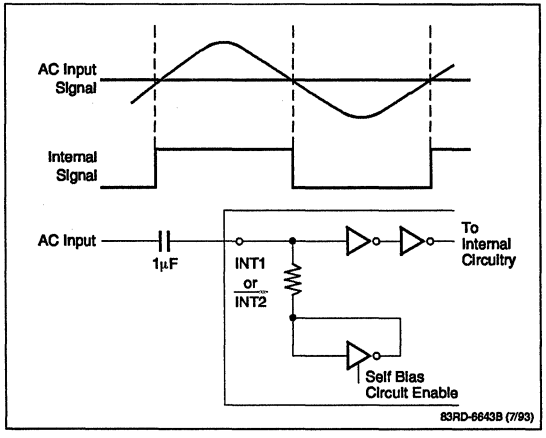
Figure 6. Universal Serial Interface Block Diagram



Zero-Crossing Detector

The INT1 and $\overline{INT2}$ terminals (used common to TI and PC_3) can detect the zero-crossing point of low-frequency AC signals. When driven directly, these pins respond as a normal digital input. Figure 7 shows the zero-crossing detection circuitry.

Figure 7. Zero-Crossing Detection Circuit



The zero-crossing detection capability allows you to make the 50-60 Hz power signal the basis for system timing and to control voltage phase-sensitive devices.

To use the zero-cross detection mode, an AC signal of 1.0 to 1.8 V (peak-to-peak) and a maximum frequency of 1 kHz is coupled through an external capacitor to the INT1 and $\overline{INT2}$ pins.

For the INT1 pin, the internal digital state is sensed as a 0 until the rising edge crosses the average DC level, when it becomes a 1 and an INT1 interrupt is generated.

For the $\overline{INT2}$ pin, the state is sensed as a 1 until the falling edge crosses the average DC level, when it becomes a 0 and $\overline{INT2}$ is generated.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = 25°C

Power supply voltage, V_{DD}	−0.5 to +7.0 V
Power supply voltage, AV_{DD}	AV_{SS} to V_{DD} + 0.5 V
Power supply voltage, AV_{SS}	−0.5 to + 0.5 V
Power supply voltage, V_{PP} (μPD78CP14 only)	−0.5 to +13.5 V
Input voltage, V_I	−0.5 to V_{DD} + .5 V
$\overline{STOP}$ pin (μPD78CP14 only)	−0.5 to +13.5 V
Output voltage, V_O	−0.5 to V_{DD} + .5 V
Output current, low; I_{OL}	
Each output pin	4.0 mA
Total	100 mA
Output current, high; I_{OH}	
Each output pin	−2.0 mA
Total	−50 mA
Reference input voltage, VA_{REF}	−0.5 to AV_{DD} + 0.3 V
Operating temperature, T_{OPR} $f_{XTAL} \leq$ 15 MHz	−40 to +85°C
Storage temperature, T_{STG}	−65 to +150°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC Characteristics.

Capacitance

T_A = 25°C; V_{DD} = V_{SS} = 0 V

Parameter	Symbol	Max	Unit	Conditions
Input capacitance	C_I	10	pF	f_c = 1 MHz; unmeasured pins returned to 0 V
Output capacitance	C_O	20	pF	
I/O capacitance	C_{IO}	20	pF	

2a

Oscillation Characteristics

T_A = −40 to 85°C; V_{DD} = AV_{DD} = 5 V ± 10% (±5% μPD78CP14); V_{SS} = AV_{SS} = 0 V; V_{DD} − 0.8 V ≤ AV_{DD} ≤ V_{DD}; 3.4 V ≤ V_{AREF} ≤ AV_{DD}

Resonator	Recommended Circuit	Parameter	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Note 1) or crystal oscillator (XTAL)(Note 4)	(Note 2)	Oscillation frequency (f_{XX})	4		15	MHz	A/D converter not used
			5.8		15	MHz	A/D converter used
			6		15	MHz	μPD78CP14 only
External clock	(Note 3)	X1 input frequency (f_X)	4		15	MHz	A/D converter not used
			5.8		15	MHz	A/D converter used
			6		15	MHz	μPD78CP14 only
		X1 input, rise, fall time (t_r, t_f)	0		20	ns	
		X_1 input low- and high-level width ($t_{\phi L}$, $t_{\phi H}$)	20		250	ns	
			20		167	ns	μPD78CP14

Notes:

(1) Refer to the Resonator and Capacitance Requirements table for the recommended ceramic resonators.

(2) For XTAL, see the Recommended XTAL or Ceramic Resonator Oscillation Circuit Diagram.

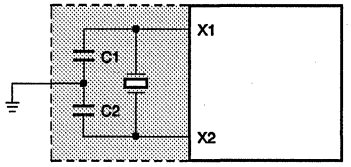
(3) See the following recommended external clock diagram.

(4) When using a crystal oscillator, it should be a parallel-resonant, fundamental mode, "AT cut" crystal. Capacitors C1 and C2 are required for frequency stability. The values of C1 and C2 (C1 = C2) can be calculated from the load capacitance (C_L), specified by the crystal manufacturer:

$$C_L = \frac{C1 \times C2}{C1 + C2} + C_S$$

Where C_S is any stray capacitance in parallel with the crystal such as the μPD78C10A, μPD78C11A, or μPD78C14/14A input capacitance between X1 and X2.

Recommended XTAL or Ceramic Resonator Oscillation Circuit Diagram

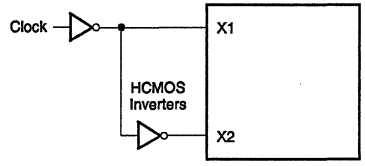


External oscillation circuit should be as close to the X1 and X2 pins as possible.

Do not place other signal lines in the shaded area.

83RD-6946A

Recommended External Clock Diagram



83RD-6946A

Resonator and Capacitance Requirements

T_A = −40 to +85°C

Manufacturer	Product Number	C1, C2 (pF)	Conditions
Murata	CSA15.0MX3	22	μPD78C14/C14A
	CSA10.0MT	30	
	CST10.0MT	Not required	
	CSA6.00MG	30	
	CST6.00MG	Not required	
	CSA12.0MT	30	μPD78C10A/C11A/C12A/C14/C14A
	CST12.0MT	Not required	
	CSA15.00MX001	15	μPD78C10A/C11A/C12A
	CSA7.37MT	30	
	CST7.37MT	Not required	
TDK	FCR12.9MC	Not required	μPD787C14/C14A

DC Characteristics

T_A = −40 to +85°C; V_{SS} = 0 V; V_{DD} = +5.0 V ±10% (μPD78C10A/C11A/C12A/C14/C14A); V_{DD} = +5.0 V ± 5% (μPD78CP14 only)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input voltage, low	V_{IL1}	0		0.8	V	All except Note 1 inputs
	V_{IL2}	0		0.2 V_{DD}	V	Note 1 inputs
Input voltage, high	V_{IH1}	2.2		V_{DD}	V	All except X1, X2, and Note 1 inputs
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	X1, X2, and Note 1 inputs
Output voltage, low	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output voltage, high	V_{OH}	V_{DD} − 1.0			V	I_{OH} = 1.0 mA
		V_{DD} − 0.5			V	I_{OH} = −100 μA
Data retention voltage	V_{DDDR}	2.5			V	STOP mode
Input current	I_{I1}			±200	μA	INT1 (Note 2); TI (PC_3) (Note 3); 0 V ≤ V_I ≤ V_{DD}
Input current (μPD78C14 only)	I_{I2}			±200	μA	INT1 (Note 2); TI (PC_3) (Note 3); 0 V ≤ V_I ≤ V_{DD}
Input current (μPD78C14 only)	I_{I3}			−300	μA	I_0–I_7 (upper input pin); V_I = 0
Input leakage current	I_{LI}			±10	μA	All except INT1, TI (PC_3), 0 V ≤ V_I ≤ V_{DD}
				±1	μA	AN7-0, 0 V ≤ V_1 ≤ V_{DD} (μPDC10A(A)/C11A(A)/C12A(A)/CP14A(A) only)
Output leakage current	I_{LO}			±10	μA	0 V ≤ V_O ≤ V_{DD}
AV_{DD} supply current	AI_{DD1}		0.5	1.3	mA	f = 15 MHz
	AI_{DD2}		10	20	μA	STOP mode
V_{DD} supply current	I_{DD1}		13	25	mA	Normal operation; f = 15 MHz; (μPD78C10A/C11A/C12A only)
	I_{DD2}		7	13	mA	HALT mode; f = 15 MHz; (μPD78C10A/C11A/C12A only)
	I_{DD3}		16	30	mA	Normal operation; f = 15 MHz (μPD78C14/C14A)
	I_{DD4}			32	mA	Normal operation; f = 15 MHz; (μPD78CP14 only)
	I_{DD5}		8	15	mA	HALT mode; f = 15 MHz; (μPD78C14/C14A/CP14 only)
Data retention current	I_{DDDR}		1	15	μA	V_{DDDR} = 2.5 V (Note 4)
				300		(μPD78CP14 only-Note 4)
			10	50	μA	V_{DDDR} = 5.0 V ±10% (Note 4)
				1	mA	(μPD78CP14 only-Note 4)
Pullup resistor	R_L	17	27	75	KΩ	Ports A, B, C; 3.5 V ≤ V_{DD} ≤ 5.5 V; V_I = 0 V (μPD78C11A/C12A/C14A only)

Notes:

(1) Inputs $\overline{\text{RESET}}$, $\overline{\text{STOP}}$, $\overline{\text{NMI}}$, $\overline{\text{SCK}}$, INTP1, TI, and AN4-AN7.

(2) Assuming ZCM register is set to self-bias.

(3) Assuming ZCM register is set to self-bias and the MCC register is set to control mode.

(4) Hardware/software STOP mode and assuming ZCM register is set to self-bias not selected.

AC Characteristics

T_A = −40 to +85°C; V_{SS} = 0 V; V_{DD} = +5.0 V ±10% (μPD78C10A/C11A/C12A/C14/C19A); V_{DD} = +5.0 V ± 5% (μPD78CP14 only)

Parameter	Symbol	Min	Max	Unit	Conditions
$\overline{RESET}$ pulse width high, low	t_{RSH}, t_{RSL}	10		μs	
$\overline{NMI}$ pulse width high, low	t_{NIH}, t_{NIH}	10		μs	
X1 input cycle time	t_{CYC}	66	250	ns	
			167	ns	(Note 1)
Address setup to ALE ↓	t_{AL}	30		ns	(Notes 2, 3)
Address hold to ALE ↓	t_{LA}	35		ns	(Notes 2, 3)
Address to $\overline{RD}$ ↓ delay time	t_{AR}	100		ns	(Notes 2, 3)
$\overline{RD}$ ↓ to address floating	t_{AFR}		20	ns	(Note 2)
Address to data input	t_{AD}		250	ns	(Notes 2, 3)
ALE ↓ to data input	t_{LDR}		135	ns	(Notes 2, 3)
$\overline{RD}$ ↓ to data input	t_{RD}		120	ns	(Notes 2, 3)
ALE ↓ to $\overline{RD}$ ↓ delay time	t_{LR}	15		ns	(Notes 2, 3)
Data hold time $\overline{RD}$ ↑	t_{RDH}	0		ns	(Note 2)
$\overline{RD}$ ↑ to ALE ↑ delay time	t_{RL}	80		ns	(Notes 2, 3)
$\overline{RD}$ width low	t_{RR}	215		ns	Data read (Notes 2, 3)
		415		ns	Opcode fetch (Notes 2, 3)
ALE width high	t_{LL}	90		ns	(Notes 2, 3)
$\overline{M1}$ setup time to ALE ↓	t_{ML}	30		ns	(Note 3)
$\overline{M1}$ hold time after ALE ↓	t_{LM}	35		ns	(Note 3)
$\overline{IO}$/M setup time to ALE ↓	t_{IL}	30		ns	(Note 3)
$\overline{IO}$/M hold time after ALE ↓	t_{LI}	35		ns	(Note 3)
Address to $\overline{WR}$ ↓ delay	t_{AW}	100		ns	(Notes 2, 3)
ALE ↓ to data output	t_{LDW}		180	ns	(Notes 2, 3)
$\overline{WR}$ ↓ to data output	t_{WD}		100	ns	(Note 2)
ALE ↓ to $\overline{WR}$ ↓ delay time	t_{LW}	15		ns	(Notes 2, 3)
Data setup time to $\overline{WR}$ ↑	t_{DW}	165		ns	(Notes 2, 3)
Data hold time to $\overline{WR}$ ↑	t_{WDH}	60		ns	(Notes 2, 3)
$\overline{WR}$ ↑ to ALE ↑ delay time	t_{WL}	80		ns	(Notes 2, 3)
$\overline{WR}$ width low	t_{WW}	215		ns	(Notes 2, 3)
Address to data input	t_{ACC}		250	ns	(Notes 2, 3)
Data hold time from address	t_{IH}	0		ns	(Note 2)

Notes:

(1) Applies to μPD78CP14 only.

(2) Load capacitance C_L = 150 pF.

(3) Values are for 15-MHz operation. For operation at other frequencies, refer to the Bus Timing Dependent on t_{CYK} table.

Serial Operation

Parameter	Symbol	Min	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{CYK}	0.8		μs	$\overline{SCK}$ input (Notes 1, 3)
		0.4		μs	$\overline{SCK}$ input (Note 2)
		1.6		μs	$\overline{SCK}$ output (Note 3)

Serial Operation (cont)

Parameter	Symbol	Min	Max	Unit	Conditions
$\overline{SCK}$ width low	t_{KKL}	335		ns	$\overline{SCK}$ input (Notes 1, 3)
		160		ns	$\overline{SCK}$ input (Note 2)
		700		ns	$\overline{SCK}$ output (Note 3)
$\overline{SCK}$ width high	t_{KKH}	335		ns	$\overline{SCK}$ input (Notes 1, 3)
		160		ns	$\overline{SCK}$ input (Note 2)
		700		ns	$\overline{SCK}$ output (Note 3)
RxD setup time to $\overline{SCK}$ ↑	t_{RXK}	80		ns	(Note 1)
RxD hold time after $\overline{SCK}$ ↑	t_{KRX}	80		ns	(Note 1)
$\overline{SCK}$ ↓ TxD delay time	t_{KTX}		210	ns	(Note 1)

Notes:

(1) 1 x baud rate in synchronous or I/O interface mode.

(2) 16 x baud rate or 64 x baud rate in asynchronous mode.

(3) f_{XTAL} = 15 MHz.

Zero-Cross Characteristics

Parameter	Symbol	Min	Max	Unit	Condition
Zero-cross detection input	V_{ZX}	1	1.8	VAC_{p-p}	AC-coupled 60-Hz sine wave
Zero-cross accuracy	A_{ZX}		±135	mV	
Zero-cross detection input frequency	f_{ZX}	0.05	1	kHz	

A/D Converter Characteristics

T_A = −40 to +85°C; V_{DD} = +5.0 V ±10% (±5% on μPD78CP14); V_{SS} = AV_{SS} 0 V;
V_{DD} −0.5 V ≤ AV_{DD} ≤ V_{DD}; 3.4 V ≤ V_{AREF} ≤ AV_{DD}

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		8			bits	
Absolute accuracy (Note 1)				±0.4	%FSR	T_A = −10 to +70°C; 66 ns ≤ t_{CYC} ≤ 170 ns; 4.0 V ≤ V_{AREF} ≤ AV_{DD}
				±0.6	%FSR	66 ns ≤ t_{CYC} ≤ 170 ns; 4.0 V ≤ V_{AREF} ≤ AV_{DD}
				±0.8	%FSR	66 ns ≤ t_{CYC} ≤ 170 ns; 3.4 V ≤ V_{AREF} ≤ AV_{DD}
Conversion time	t_{CONV}	576			t_{CYC}	66 ns ≤ t_{CYC} ≤ 110 ns
		432			t_{CYC}	110 ns ≤ t_{CYC} ≤ 170 ns
Sampling time	t_{SAMP}	96			t_{CYC}	66 ns ≤ t_{CYC} ≤ 110 ns
		72			t_{CYC}	110 ns ≤ t_{CYC} ≤ 170 ns
Analog input voltage	V_{IAN}	0		V_{AREF}	V	
Analog input impedance	R_{AN}		1000		MΩ	
Reference voltage	V_{AREF}	3.4		AV_{DD}	V	
V_{AREF} current	I_{AREF1}		1.5	3.0	mA	Operation mode
	I_{AREF2}		0.7	1.5	mA	STOP mode
AV_{DD} supply current	AI_{DD1}		0.5	1.3	mA	Operation mode
	AI_{DD2}		10	20	μA	STOP mode

Notes:

(1) Quantizing error (±1/2 LSB) is not included.

(2) FSR = Full-scale resolution.

Bus Timing Dependent on t_{CYK}

Symbol	Min/Max (ns)	Calculation Formula
t_{TIH}, t_{TIL}	Min	6T (TI input - PC_3)
t_{CI1H}, t_{CI1L} (Note 2)	Min	6T (TI input - PC_5)
t_{CI2H}, t_{CI2L} (Note 3)	Min	48T (TI input - PC_5)
t_{I1H}, t_{I1L}	Min	36T (INT1)
t_{I2H}, t_{I2L}	Min	36T ($\overline{INT2}$)
t_{ANH}, t_{ANL}	Min	36T (AN4-AN7)
t_{AL}	Min	2T – 100
t_{LA}	Min	T – 30
t_{AR}	Min	3T – 100
t_{AD}	Max	7T – 220
t_{LDR}	Max	5T – 200
t_{RD}	Max	4T – 150
t_{LR}	Min	T – 50
t_{RL}	Min	2T – 50
t_{RR}	Min	4T – 50 (Data read)
	Min	7T – 50 (Opcode fetch)
t_{LL}	Min	2T – 40
t_{ML}	Min	2T – 100
t_{LM}	Min	T – 30
t_{IL}	Min	2T – 100
t_{LI}	Min	T – 30
t_{AW}	Min	3T – 100
t_{LDW}	Max	T + 110
t_{LW}	Min	T – 50
t_{DW}	Min	4T – 100
t_{WDH}	Min	2T – 70
t_{WL}	Min	2T – 50
t_{WW}	Min	4T – 50
t_{CYK}	Min	12T ($\overline{SCK}$ input) (Note 1)
	Min	24T ($\overline{SCK}$ output)
t_{KKL}	Min	5T + 5 ($\overline{SCK}$ input) (Note 1)
	Min	12T – 100 ($\overline{SCK}$ output)
t_{KKH}	Min	5T + 5 ($\overline{SCK}$ input) (Note 1)
	Min	12T – 100 ($\overline{SCK}$ output)

Notes:

(1) 1 x baud rate in synchronous or I/O interface mode; T = t_{CYC} = $1/f_{XTAL}$.

The items not included in this list are independent of oscillator frequency.

(2) Event counter mode.

(3) Pulse-width measurement mode.

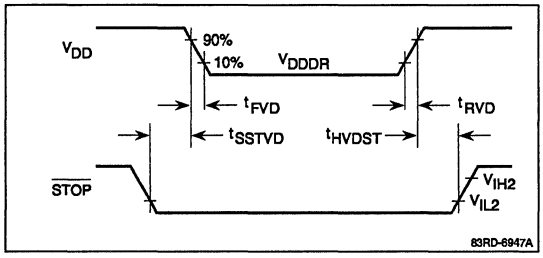
Data Memory STOP Mode Data Retention Characteristics

T_A = −40 to 85°C

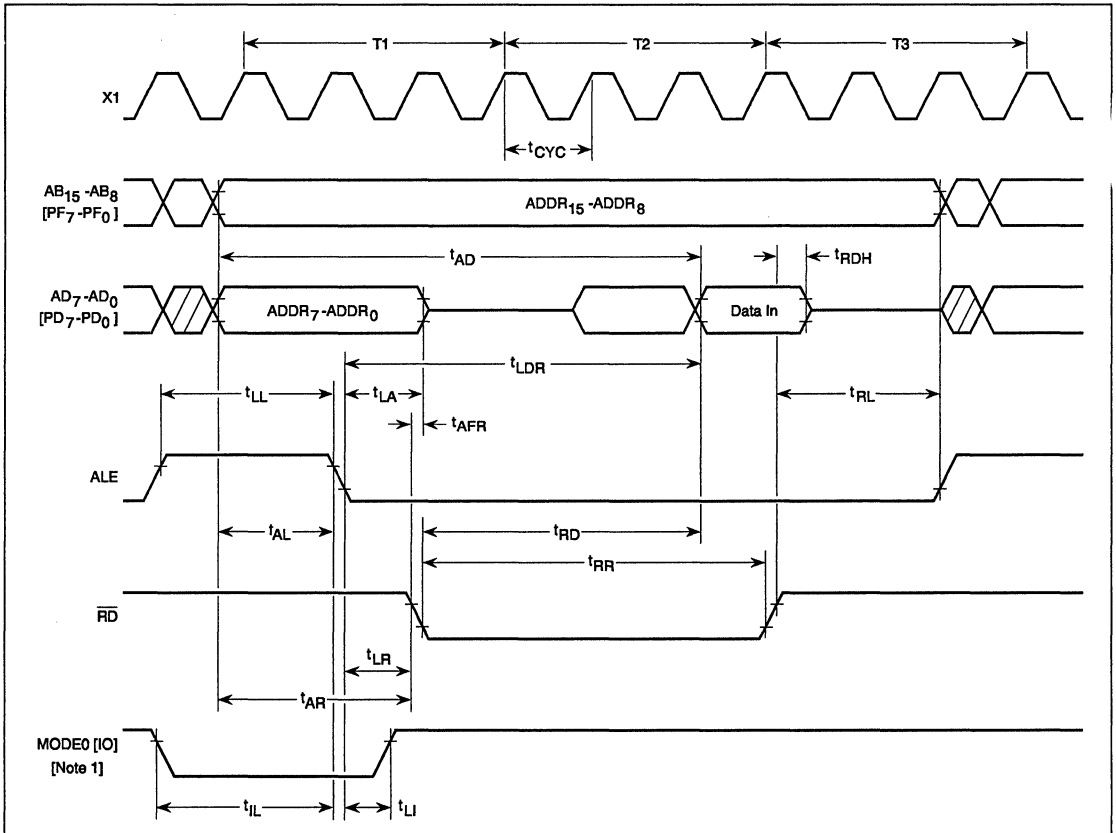
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention power supply voltage	V_{DDDR}	2.5		5.5	V	
Data retention power supply current	I_{DDDR}		1	15	μA	V_{DDDR} = 2.5 V
			15	50	μA	V_{DDDR} = 5.0 V ±10%
				300	μA	V_{DDDR} = 2.4 V (μPD78CP14)
				1	mA	V_{DDDR} = 5.0 V ±5% (μPD78CP14)
V_{DD} rise, fall time	t_{RVD}, t_{FVD}	200			μs	
$\overline{STOP}$ setup time to V_{DD}	t_{SSTVD}	12T+ 0.5			μs	
$\overline{STOP}$ hold time from V_{DD}	t_{HVDST}	12T+ 0.5			μs	

Timing Waveforms

Data Retention



Read Operation



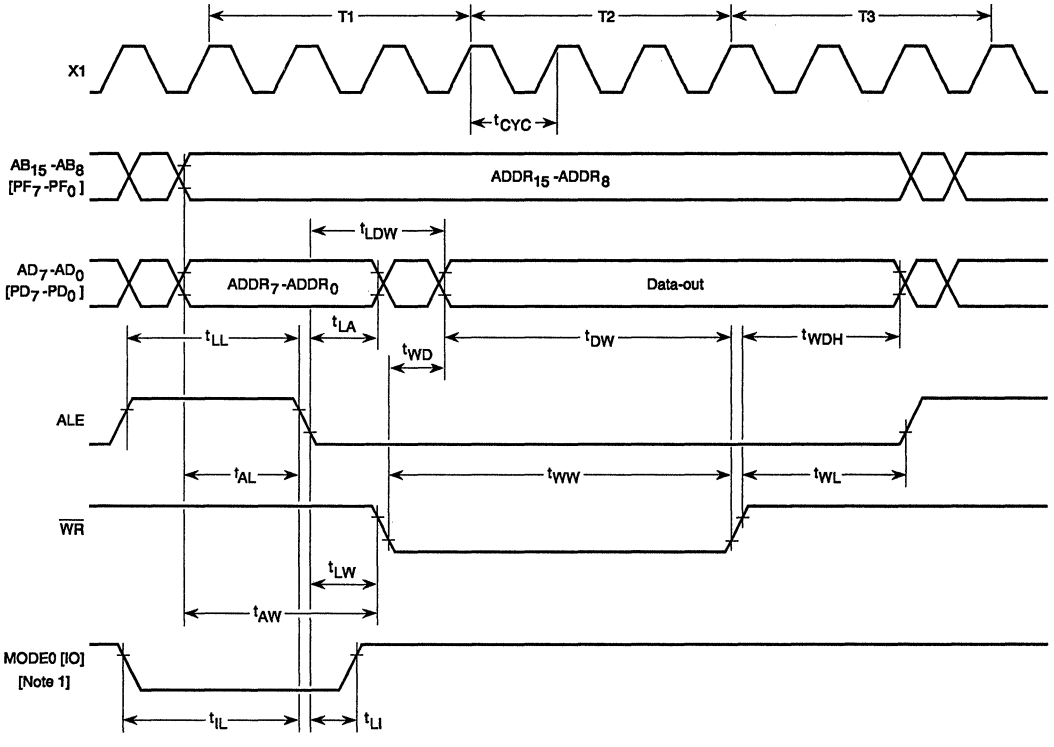
Note:

[1] $\overline{IO}$ signal is output to the MODE0 pin [if MODE0 is pulled up to V_{DD}] during a read or write of special registers sr-sr2 or a write to register MM or MF. Refer to description of Port Emulation Mode [PEM] in the User's Manual for further explanation. This signal is not output on the μPD78CP14.

83YL-9253B

Timing Waveforms (cont)

Write Operation



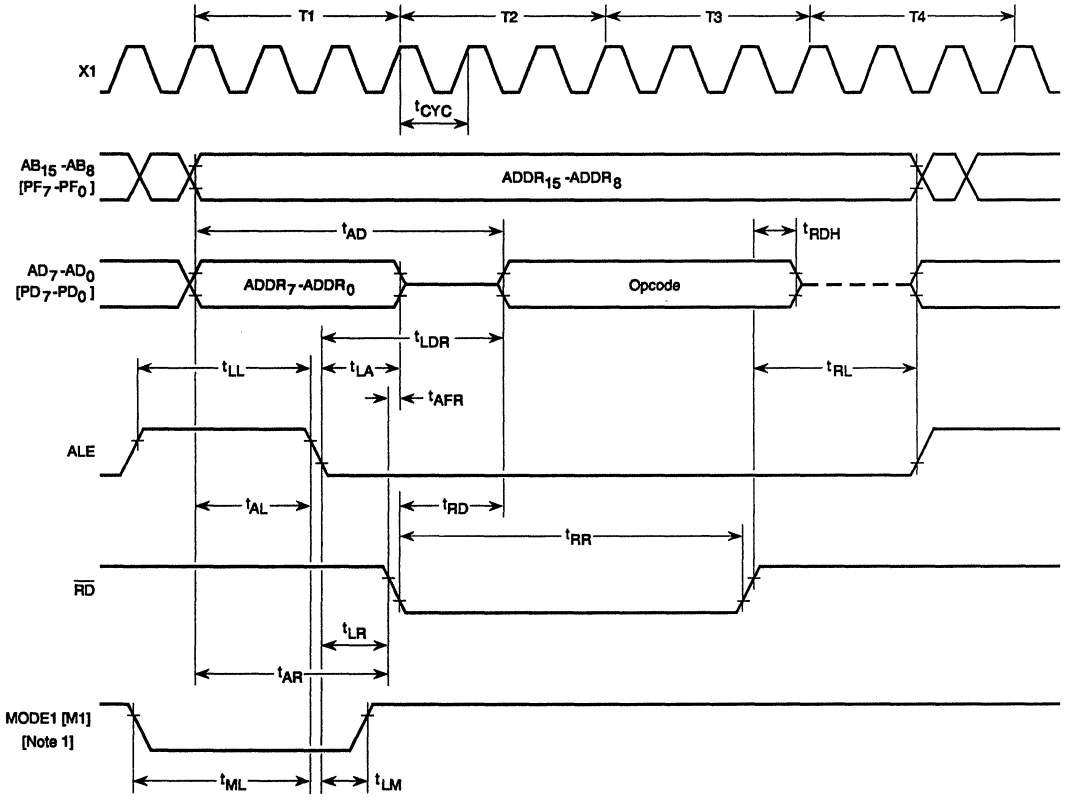
Note:

[1] $\overline{IO}$ signal is output to the MODE0 pin [if MODE0 is pulled up to V_{DD}] during a read or write of special registers sr-sr2 or a write to register MM or MF. Refer to description of Port Emulation Mode [PEM] in the User's Manual for further explanation. This signal is not output on the μPD78CP14.

83YL-9254B

Timing Waveforms (cont)

Opcode Fetch Operation



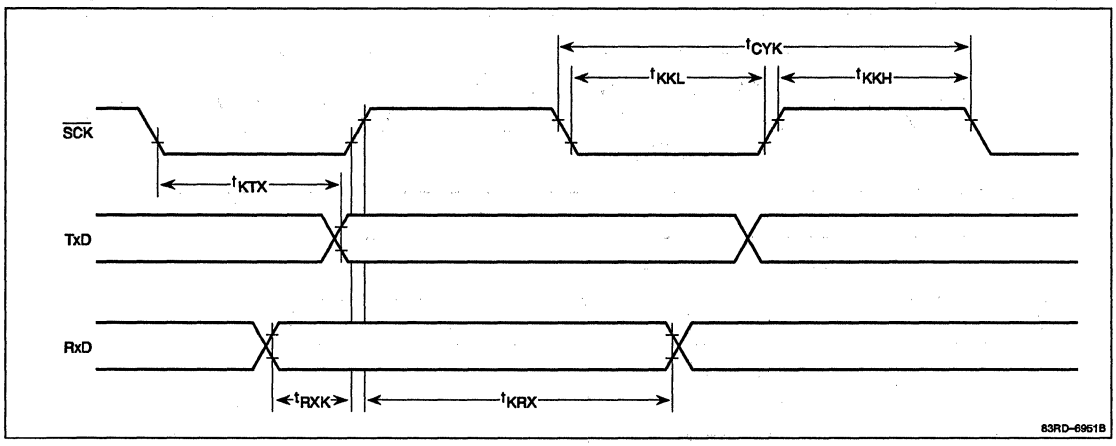
Note:

[1] $\overline{M1}$ signal is output to the MODE1 pin during every Opcode Fetch if MODE1 pin is pulled up to V_{DD}. This signal is not output on the μPD78CP14.

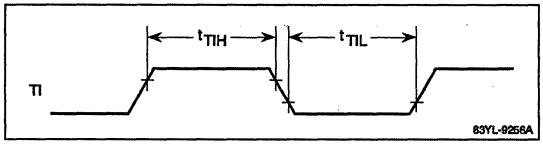
83YL-9255B

Timing Waveforms (cont)

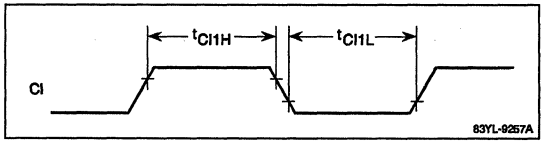
Serial Operation Transmit/Receive



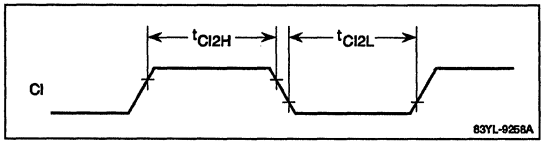
Timer Input



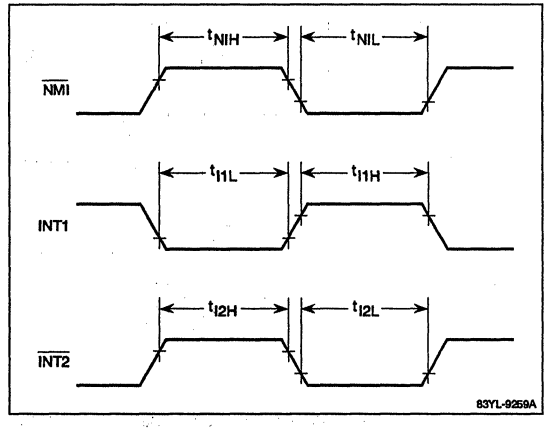
Timer/Event Counter Input: Event Counter Mode



Timer/Event Counter Input: Pulse-Width Measurement Mode

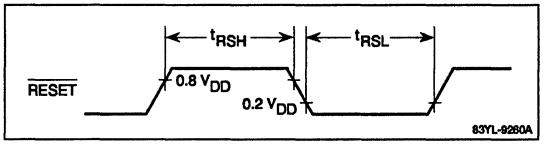


Interrupt Input

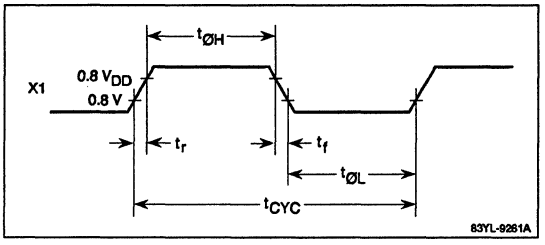


Timing Waveforms (cont)

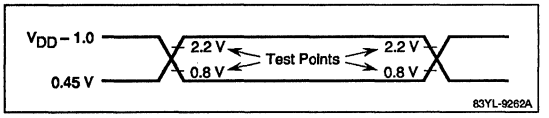
$\overline{\text{RESET}}$ Input



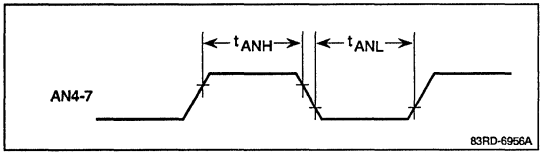
External Clock



AC Timing Test Points



AN4-AN7 Edge Detection



2a

μPD78CP14 PROGRAMMING

In the μPD78CP14, the mask ROM of the μPD78C14 family is replaced by a one-time programmable ROM (OTP ROM) or a reprogrammable, ultraviolet erasable ROM (UV EPROM). The ROM is 16,384 by 8 bits and can be programmed using a general-purpose PROM writer with a μPD27C256A programming mode. Refer to tables 3 through 5 and the DC and AC Programming Characteristics tables for specific information applicable to programming the μPD78CP14.

Table 3. Pin Functions during EPROM Programming

Pin	Function	Description
PA_0 - PA_7	A_0 - A_7	Low-order 8-bit address
PF_0	A_8	High-order 7-bit address
$\overline{NMI}$	A_9	
PF_2 - PF_6	A_{10} - A_{14}	
PD_0 - PD_7	D_0 - D_7	Data input/output
PB_6	$\overline{CE}$	Chip enable input
PB_7	$\overline{OE}$	Output enable input
$\overline{RESET}$	$\overline{RESET}$	PROM programming mode requires a low voltage on this pin
Mode 0	Mode 0	Enter PROM programming mode by applying a high voltage to this pin
Mode 1	Mode 1	Enter PROM programming mode by applying a low voltage to this pin
$\overline{STOP}$	V_{PP}	High-voltage input (write/verify) high level (read)

Table 4. Summary of Operation Modes for EPROM Programming

Operation Mode	$\overline{CE}$	$\overline{OE}$	V_{PP}	V_{DD}	$\overline{RESET}$	MODE0	MODE1	A_{14}
Program write	L	H	+12.5 V	+6 V	L	H	L	L
Program verify	H	L	+12.5 V	+6 V	L	H	L	L
Program inhibit	H	H	+12.5 V	+6 V	L	H	L	L
Read	L	L	+5 V	+5 V	L	H	L	L
Output disable	L	H	+5 V	+5 V	L	H	L	L
Standby	H	L/H	+5 V	+5 V	L	H	L	L

Notes:

(1) The $\overline{CE}$, $\overline{OE}$, V_{pp}, and V_{DD} pins are all compatible with the μPD27C256A pins.

Caution: When Vpp is set to +12.5 V and V_{DD} is set to +6 V, you cannot set both $\overline{CE}$ and $\overline{OE}$ to low level (L).

Table 5. Recommended Connections for Unused Pins (EPROM Programming Mode)

Pin	Recommended Connection Method
INT1	Connect to V_{SS}
X1	Connect to V_{SS}
X2	Leave this pin disconnected
AN0-AN7	Connect to V_{SS}
VA_{REF}	Connect to V_{SS}
AV_{DD}	Connect to V_{SS}
AV_{SS}	Connect to V_{SS}
Remaining pins	Connect each pin via a resistor to V_{SS}

PROM Write Procedure

(1) Connect the $\overline{\text{RESET}}$ pin, the MODE1 pin, and A_{14} pin to a low level and connect the MODE0 pin to a high level. Connect all unused pins as recommended in Table 5.

(2) Apply +6 V to the V_{DD} pin and +12.5 V to the V_{pp} pin.

(3) Provide the initial address.

(4) Provide write data.

(5) Provide 1-ms program pulse (active low) to the $\overline{\text{CE}}$ pin.

(6) This bit is now verified with a pulse (active low) to the $\overline{\text{OE}}$ pin. If the data has been written, proceed to step 8; if not, repeat steps 4 to 6. If the data cannot be correctly written after 25 attempts, go to step 7.

(7) Classify as defective and stop write operation.

(8) Provide write data and supply program pulse (for additional writing) for 3 ms times the number of repeats performed between steps 4 to 6.

(9) Increment the address.

(10) Repeat steps 4 to 9 until the end address.

PROM Read Procedure

(1) Connect the $\overline{\text{RESET}}$ pin, the MODE1 pin, and A_{14} pin to a low level and connect the MODE0 pin to a high level.

(2) Apply +5 V to the V_{DD} and V_{pp} pins.

(3) Input the address of the data to be read to pins A_0 - A_{14}.

(4) Read mode is entered with a pulse (active low) on both the $\overline{\text{CE}}$ and $\overline{\text{OE}}$ pins.

(5) Data is output to the D_0 - D_7 pins.

EPROM Erasure

Data in an EPROM is erased by exposing the quartz window in the ceramic package to light having a wavelength shorter than 400 nm, including ultraviolet rays, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15W-s/cm^2 (ultraviolet ray intensity x exposure time) is required to completely erase written data. Erasure by an ultraviolet lamp rated at 12 mW/cm^2 takes approximately 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

μPD78CP14 DC Programming Characteristics

T_A = 25 ±5°C; MODE1 = V_{IL}; MODE0 = V_{IH}; V_{SS} = 0 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH}	V_{IH}	2.2		V_{DDP}+ 0.3	V	
Low-level input voltage	V_{IL}	V_{IL}	−0.3		0.8	V	
Input leakage current	I_{LIP}	I_{LI}			±10	μA	$0 \le V_I \le V_{DDP}$
High-level output voltage	V_{OH}	V_{OH}	V_{DD} − 1.0			V	I_{OH} = −1.0 mA
Low-level output voltage	V_{OL}	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output leakage current	I_{LO}				±10	μA	$0 \le V_O \le V_{DDP}$; $\overline{OE}$ = V_{IH}
V_{DDP} power voltage	V_{DDP}	V_{CC}	5.75	6.0	6.25	V	Program memory write mode
			4.5	5.0	5.5	V	Program memory read mode
V_{PP} power voltage	V_{PP}	V_{PP}	12.2	12.5	12.8	V	Program memory write mode
				V_{PP} = V_{DDP}		V	Program memory read mode
V_{DDP} power current	I_{DD}	I_{CC}			30	mA	Program memory write mode
					30	mA	Program memory read mode; $\overline{CE}$ = V_{IL}; V_I = V_{IH}
V_{PP} power current	I_{PP}	I_{PP}			30	mA	Program memory read mode; $\overline{CE}$ = V_{IL}; $\overline{OE}$ = V_{IH}
				1	100	μA	Program memory write mode

* Corresponding symbols of the μPD27C256A.

μPD78CP14 AC Programming Characteristics

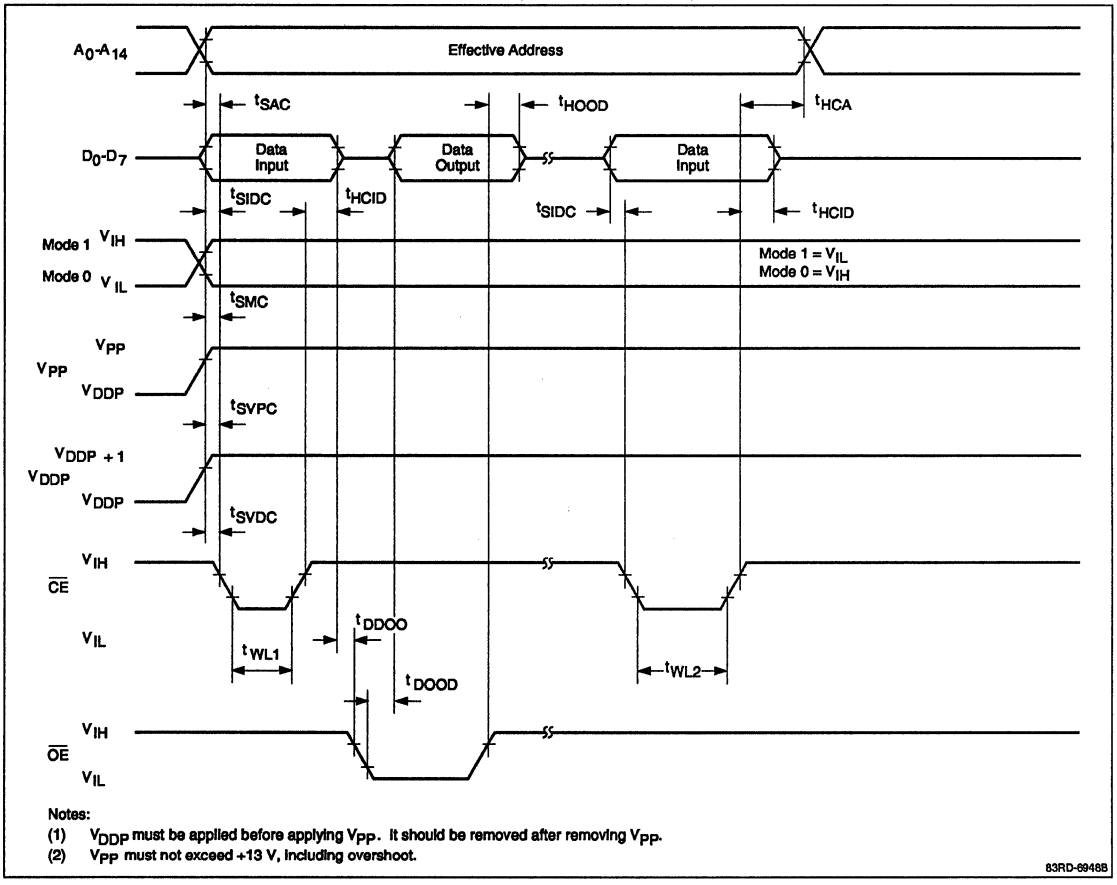
T_A = 25 ±5°C; MODE1 = V_{IL}; V_{SS} = 0 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}	2			μs	
Data to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}	2			μs	
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}	2			μs	
Address hold time from $\overline{CE}$ ↑	t_{HCA}	t_{AH}	2			μs	
Input data hold time from $\overline{CE}$ ↑	t_{HCID}	t_{DH}	2			μs	
Output data hold time from $\overline{OE}$ ↑	t_{HOOD}	t_{DF}	0		130	ns	
V_{pp} setup time to $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}	2			μs	
V_{DDP} setup time to $\overline{CE}$ ↓	t_{SVDC}	t_{VDS}	2			μs	
Initial program pulse width	t_{WL1}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{WL2}	t_{OPW}	2.85		78.75	ms	
MODE0/MODE1 setup time vs. $\overline{CE}$ ↓	t_{SMC}		2			μs	MODE1 = V_{IL} and MODE0 = V_{IH}
Address to data output time	t_{DAOD}	t_{ACC}			2	μs	$\overline{OE}$ = V_{IL}
$\overline{CE}$ ↓ to data output time	t_{DCOD}	t_{CE}			1	μs	
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			1	μs	
Data hold time from $\overline{OE}$ ↑ or $\overline{CE}$ ↑	t_{HCOD}	t_{DF}	0		130	ns	
Data hold time from address	t_{HAOD}	t_{OH}	0			ns	$\overline{OE}$ = V_{IL}

* Corresponding symbols of the μPD27C256A.

PROM Timing Diagrams

μPD78CP14 PROM Write Mode



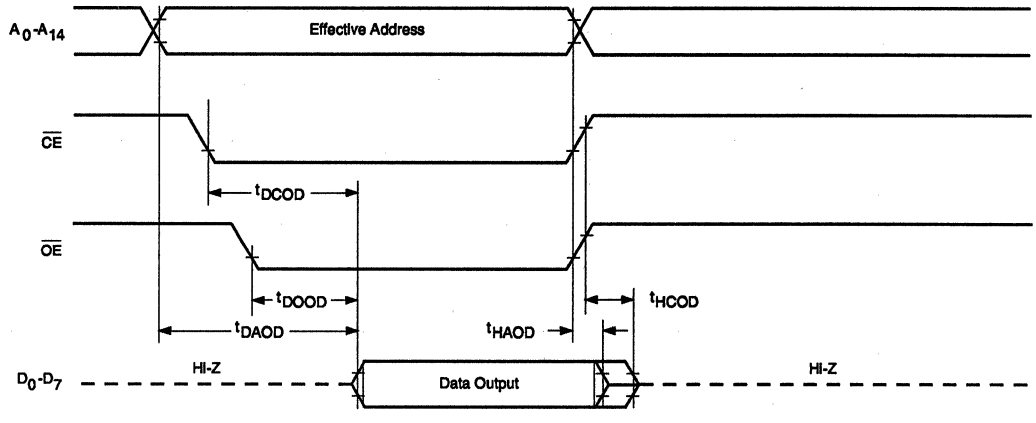
Notes:
(1) V_{DDP} must be applied before applying V_{PP}. It should be removed after removing V_{PP}.
(2) V_{PP} must not exceed +13 V, including overshoot.

83RD-6946B

2a

PROM Timing Diagrams (cont)

μPD78CP14 PROM Read Mode



Notes:

(1) To read PROM within the t_{DAOD} range, the delay of $\overline{OE}\downarrow$ from $\overline{CE}\downarrow$ must be within $t_{DAOD} - t_{DOOD}$.

(2) t_{HCOD} is the time from the state in which either $\overline{OE}$ or $\overline{CE}$ first becomes V_{IH}.

83RD-6949B

NEC
NEC Electronics Inc.

μPD78C18 Family
μPD78C17/C18/CP18
8-Bit, Single-Chip Microcontrollers With A/D Converter

September 1993

Description

The μPD78C18 family is an expanded memory version of the μPD78C14 family of 8-bit CMOS single-chip microcontrollers.

These microcontrollers integrate sophisticated on-chip peripheral functions normally provided by external components. Their internal 16-bit ALU and data paths, combined with a powerful instruction set and addressing capability, make the devices appropriate in data processing as well as control applications.

The devices integrate a 16-bit ALU, 32K-byte ROM, 1024-byte RAM, an eight-channel A/D converter, a multifunction 16-bit timer/event counter, two 8-bit timers, a USART, and two zero-crossing detect inputs on a single die, allowing their use in fast, high-end processing applications.

The μPD78C18 family includes a 32K-byte mask ROM device, embedded with a customer program, a ROM-less device for use with up to 64K bytes of external memory, and a 32K-byte EPROM or OTP ROM device for prototyping and low-volume production. The μPD78C18 may also be ordered with pullup resistors that are available as a mask option for ports A, B, and C.

Features

- CMOS technology
- 30 mA operating current (μPD78C17/C18)
- Complete single-chip microcontroller
  - 16-bit ALU
  - 32K-byte ROM
  - 1024-byte RAM
- 40 I/O lines
- Pullup resistors for the mask option
  - Ports A, B, and C
  - μPD78C18 device only
- Two zero-crossing detect inputs
- Two 8-bit timers
- Four edge-detection inputs (AN4-AN7)
- Expansion capabilities
  - 8085A-like bus
  - 64K-byte external memory address range
- Eight-channel, 8-bit A/D converter
  - Autoscan mode
  - Channel select mode
- Full-duplex USART (synchronous and asynchronous)
- 159 instructions
  - 16-bit arithmetic, multiply, and divide
  - HALT and STOP instructions
- 0.8 μs instruction cycle time (15 MHz operation)
- Prioritized interrupt structure
  - Three external
  - Eight internal
- Standby function
- On-chip clock generator

Ordering Information

Part Number (Note 1)	Package	Package Drawing	Quality Grade (Note 3)
ROMless			
μPD78C17CW	64-pin SDIP	P64C-70-750A, C	Standard
GF-3BE	64-pin QFP (Note 2)	P64GF-100-3B8, 3BE-1	Standard
GF(A)-3BE			Special
GQ-36	64-pin QUIP	P64GQ-100-36	Standard
GQ(A)-36			Special
32K Mask ROM			
μPD78C18CW-xxx	64-pin SDIP	P64C-70-750A, C	Standard
GF-xxx-3BE	64-pin QFP (Note 2)	P64GF-100-3B8, 3BE-1	Standard
GF(A)-xxx-3BE			Special
GQ-xxx-36	64-pin QUIP	P64GQ-100-36	Standard
GQ(A)-xxx-36			Special
32K OTP ROM			
μPD78CP18CW	64-pin SDIP	P64C-70-750A, C	Standard
GF-3BE	64-pin QFP (Note 2)	P64GF-100-3B8, 3BE-1	Standard
GF(A)-3BE			Special
GQ-36	64-pin QUIP	P64GQ-100-36	Standard
GQ(A)-36			Special
32K UV EPROM			
μPD78CP18DW	64-pin SDIP w/window	P64C-70-750A, C	Standard
KB	64-pin ceramic LCC w/window	X64KW-100A-1	

Note:

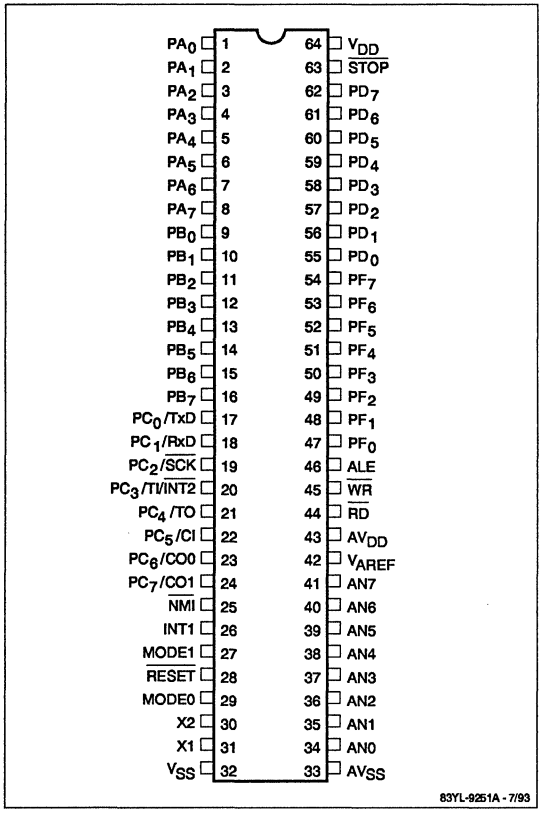
(1) xxx indicates ROM code suffix.

(2) Engineering samples supplied in a ceramic QFP package

(3) Special grade devices have the symbol (A) embedded in the part number

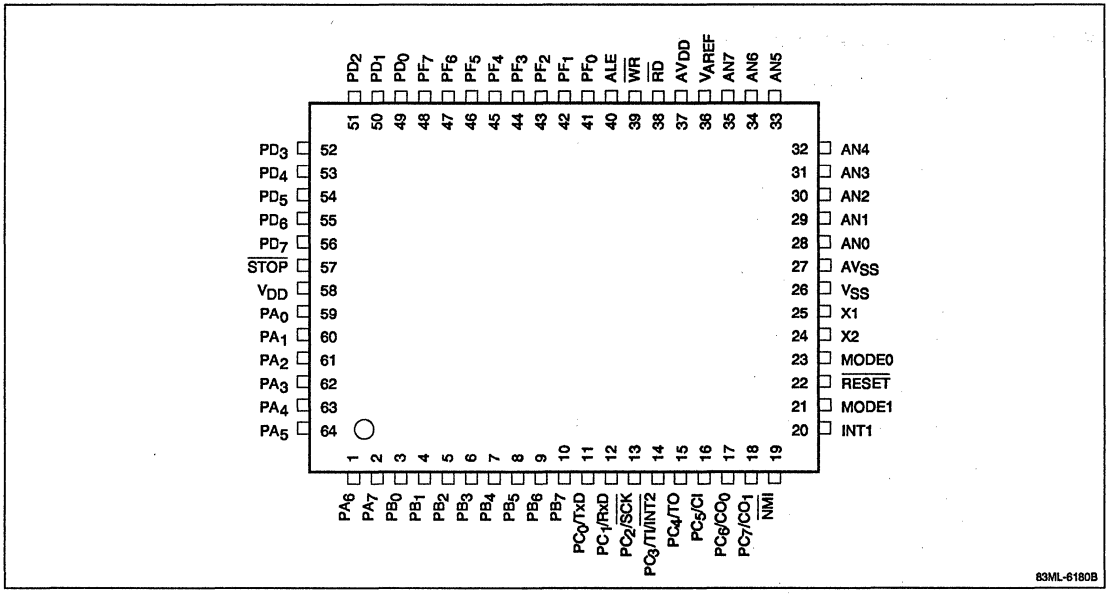
Pin Configurations

64-Pin Plastic QUIP or Plastic Shrink DIP



2b

64-Pin Plastic QFP or Ceramic LCC



Pin Identification

Symbol	Function
ALE	Address latch enable output
AN0-AN7	A/D converter analog inputs 0-7
INT1	Interrupt request 1 input
MODE0	Mode 0 input; I/O memory output
MODE1	Mode 1 input
$\overline{NMI}$	Nonmaskable interrupt input
PA_0-PA_7	Port A I/O lines 0-7
PB_0-PB_7	Port B I/O lines 0-7
PC_0/TxD	Port C I/O line 0; transmit data output
PC_1/RxD	Port C I/O line 1; receive data input
$PC_2/\overline{SCK}$	Port C I/O line 2; serial clock I/O
PC_3/TI/$\overline{INT2}$	Port C I/O line 3; timer input; interrupt request 2 input
PC_4/TO	Port C I/O line 4; timer output
PC_5/CI	Port C I/O line 5; counter input
PC_6 and PC_7/ CO_0 and CO_1	Port C I/O lines 6 and 7; counter outputs 0 and 1
PD_0-PD_7	Port D I/O; expansion memory address, data bus (bits AD_0-AD_7)
PF_0-PF_7	Port F I/O; expansion memory address, (bits AB_8-AB_{15})
$\overline{RD}$	Read strobe output
$\overline{RESET}$	Reset input
$\overline{STOP}$	Stop mode control input
V_{AREF}	A/D converter reference voltage
$\overline{WR}$	Write strobe output
X1 and X2	Crystal connections 1 and 2
AV_{DD}	A/D converter power supply voltage
AV_{SS}	A/D converter power supply ground
V_{DD}	+5 V power supply
V_{SS}	Ground
IC	Internal connection

Pin Functions

ALE (Address Latch Enable). The ALE output is used to strobe the address of PD_0-PD_7 into an external latch.

AN0-AN7 (Analog Inputs). AN0-AN7 are the eight analog inputs to the A/D converter. AN4-AN7 can also be used as digital inputs for falling edge detection.

CI (Counter Input). CI is the external pulse input to the timer/event counter.

CO_0 and CO_1 (Counter Outputs). CO_0 and CO_1 are programmable waveform outputs from the timer/event counter.

INT1 (Interrupt Request 1). INT1 is a rising edge-triggered, maskable interrupt input, as well as an ac-input, zero-crossing detection terminal.

$\overline{INT2}$ (Interrupt Request 2). $\overline{INT2}$ is a falling edge-triggered, maskable interrupt input, as well as an ac-input, zero-crossing detection terminal.

If the optional pullup resistor is specified for this pin on the μPD78C18, the zero-crossing detection circuitry will not function.

MODE0 and MODE1 (Mode). For the μPD78C17, the size of the externally installed memory can be selected as 4K, 16K, or 63K bytes by setting the MODE0 and MODE1 pins.

For the μPD78C18, the MODE0 pin is set to 0 (logic low). The MODE1 pin is pulled high with a pullup resistor.

For the μPD78C17/C18, an external pullup resistor to V_{DD} is required, if the mode pin is to be a logic high. The value of this pullup resistor, R, is dependent on t_{CYC} and is calculated as follows: R in kΩ is $4 \leq R \leq 0.4\ t_{CYC}$, where t_{CYC} is in ns units.

$\overline{NMI}$ (Nonmaskable Interrupt). NMI is a falling edge, Schmitt-triggered nonmaskable interrupt input.

PA_0-PA_7 (Port A). Port A is an 8-bit three-state port. Each bit is independently programmable as either input or output. The reset signal causes all lines of port A to be inputs. Pullup resistors are available as a mask option on the μPD78C18.

PB_0-PB_7 (Port B). Port B is an 8-bit three-state port. Each bit is independently programmable as either input or output. The reset signal causes all lines of port B to be inputs. Pullup resistors are available as a mask option on the μPD78C18.

PC_0-PC_7 (Port C). Port C is an 8-bit three-state port. Each bit is independently programmable as either input or output. Alternatively, the lines of port C can be used as control lines for the USART, interrupts, and timer. The reset signal causes all lines of port C to be inputs. Pullup resistors are available as a mask option on the μPD78C18.

PD_0-PD_7 (Port D). Port D is an 8-bit three-state port. It can be programmed as either 8 bits of input or 8 bits of output. When external expansion memory is used, port D functions as the multiplexed address/data bus.

PF_0-PF_7 (Port F). Port F is an 8-bit three-state port. Each bit is independently programmable as either input or output. When external expansion memory is used, port F outputs the high-order address bits.

$\overline{RD}$ (Read Strobe). The strobe signal, when output for read operation of external memory, operates as follows. The signal is high, except during a data read machine cycle. It becomes a high output impedance when the $\overline{RESET}$ signal is low or when the device is in hardware stop mode.

$\overline{RESET}$ (Reset). When the Schmitt-triggered $\overline{RESET}$ input goes low, it initializes the device.

RxD (Receive Data). RxD is the serial data input terminal.

$\overline{SCK}$ (Serial Clock). $\overline{SCK}$ is the serial clock output when the internal clock is used. $\overline{SCK}$ is the input for the serial clock when the external clock is used.

$\overline{STOP}$ (Stop Mode Control Input). A low-level input on $\overline{STOP}$ (Schmitt-triggered input) stops the system clock oscillator.

TI (TImer Input). TI is the timer input terminal.

TO (Timer Output). The output of TO is a square wave with a frequency determined by the timer/counter.

TxD (Transmit Data). TxD is the serial data output terminal.

V_{AREF} (A/D Converter Reference) . V_{AREF} functions as an input pin for the A/D converter reference voltage and as the control pin for A/D converter operation.

$\overline{WR}$ (Write Strobe). The strobe signal, when output for the write operation of external memory, operates as follows. The signal is high, except during a data write machine cycle. It becomes a high output impedance when the $\overline{RESET}$ signal is low or when the device is in hardware stop mode.

X1 and X2 (Crystal Connections). X1 and X2 are the system clock crystal oscillator terminals. X1 is also the input for an external clock.

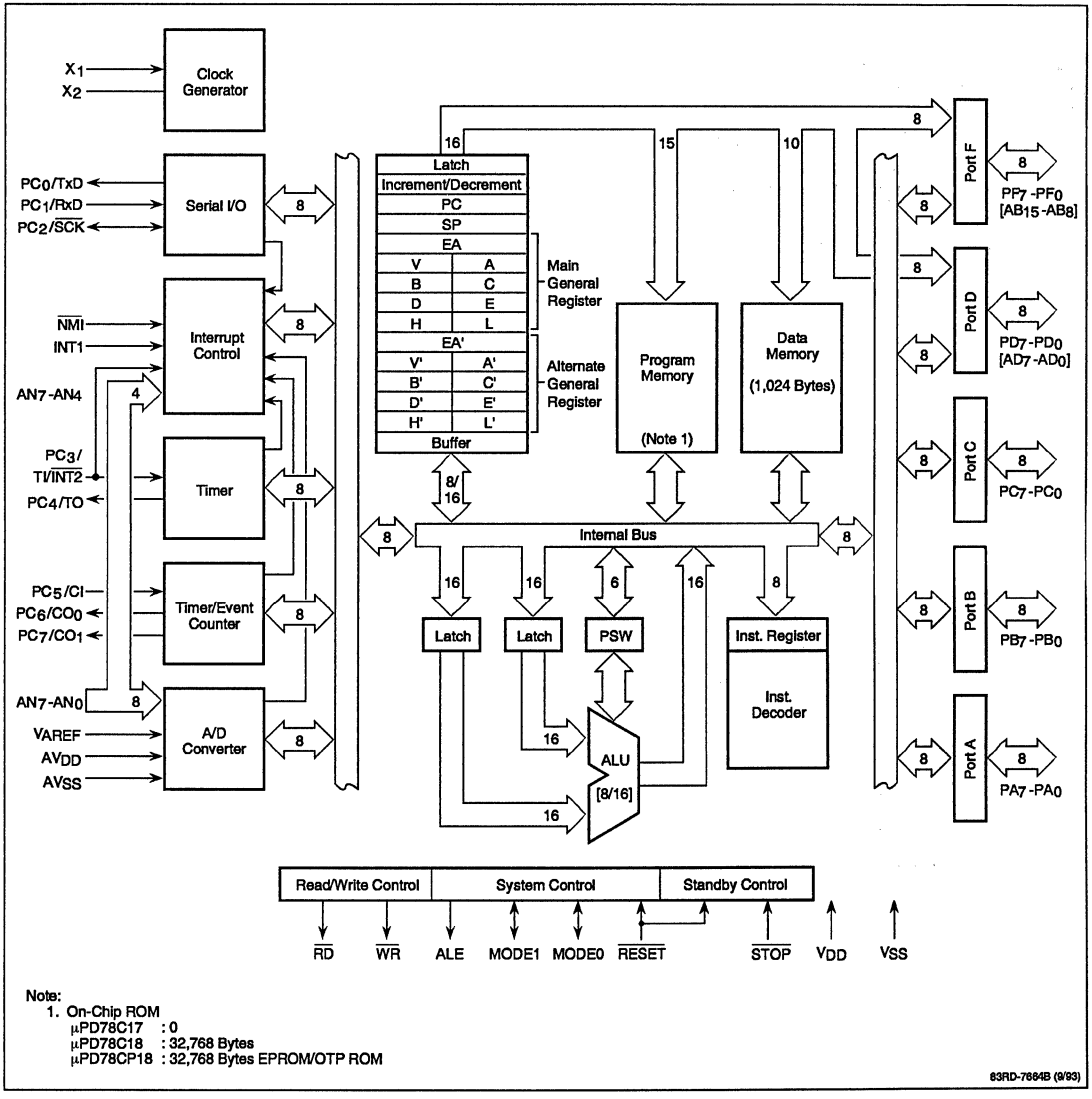
AV_{DD} (A/D Converter Power). This is the power supply voltage for the A/D converter.

AV_{SS} (A/D Converter Ground). AV_{SS} is the ground potential for the A/D converter power supply.

V_{DD} (Power Supply). V_{DD} is the +5-volt power supply.

V_{SS} (Ground). V_{SS} is the ground potential for the +5-volt device power supply.

Block Diagram



Note:
1. On-Chip ROM
   μPD78C17 : 0
   μPD78C18 : 32,768 Bytes
   μPD78CP18 : 32,768 Bytes EPROM/OTP ROM

63RD-7664B (9/93)

2b

FUNCTIONAL DESCRIPTION

Memory Map

The μPD78C18 family can directly address up to 64K bytes of memory. Except for the on-chip ROM (or PROM) and RAM (FC00H-FFFFH), any memory location can be used as ROM or RAM. The memory maps, shown in figures 1 through 3, define the 0 to 64K-byte memory space for the μPD78C18 family.

The μPD78CP18 can be programmed by software to have 4K, 8K, 16K, or 32K bytes of internal program memory. This programming is transparent to a ROM-based device, allowing easy transfer of code.

Figure 1. Memory Map (μPD78C17)

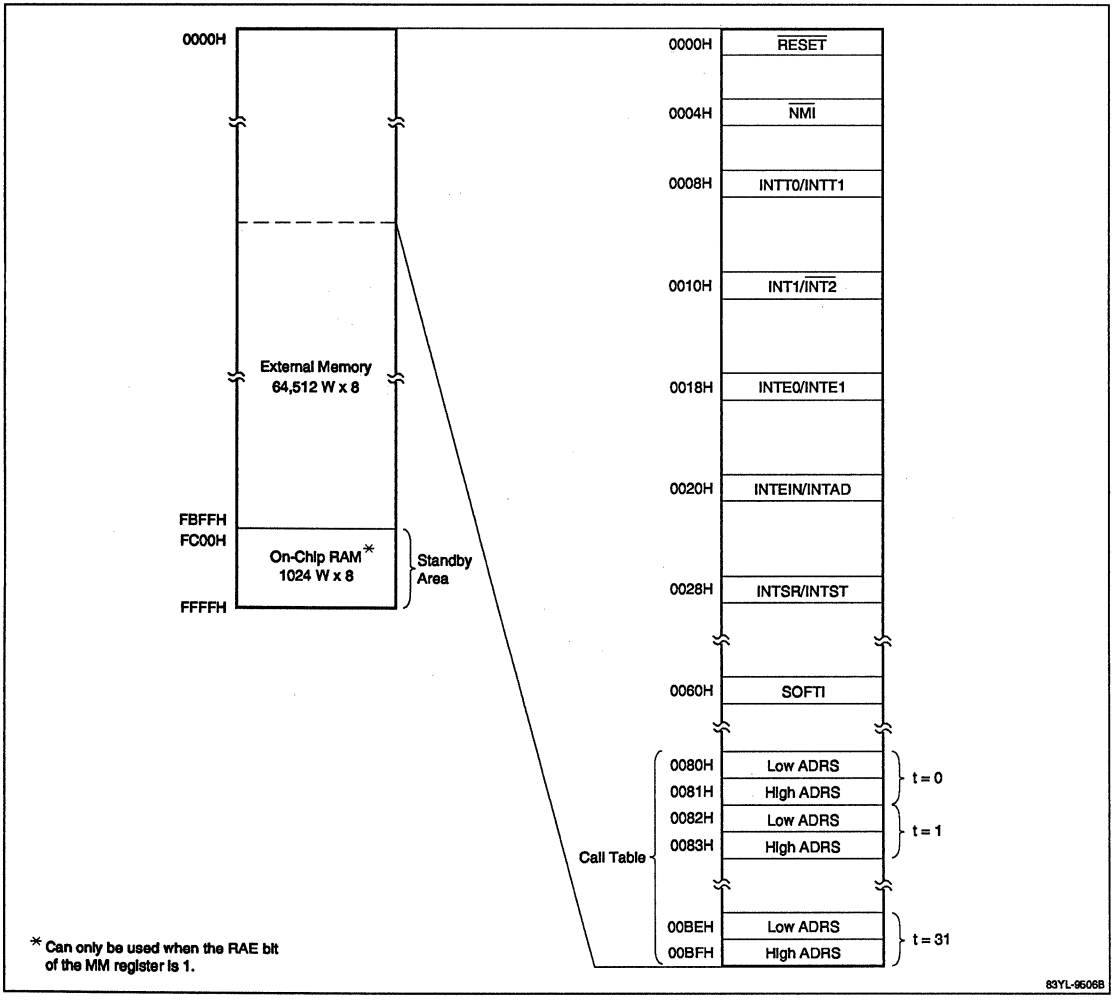
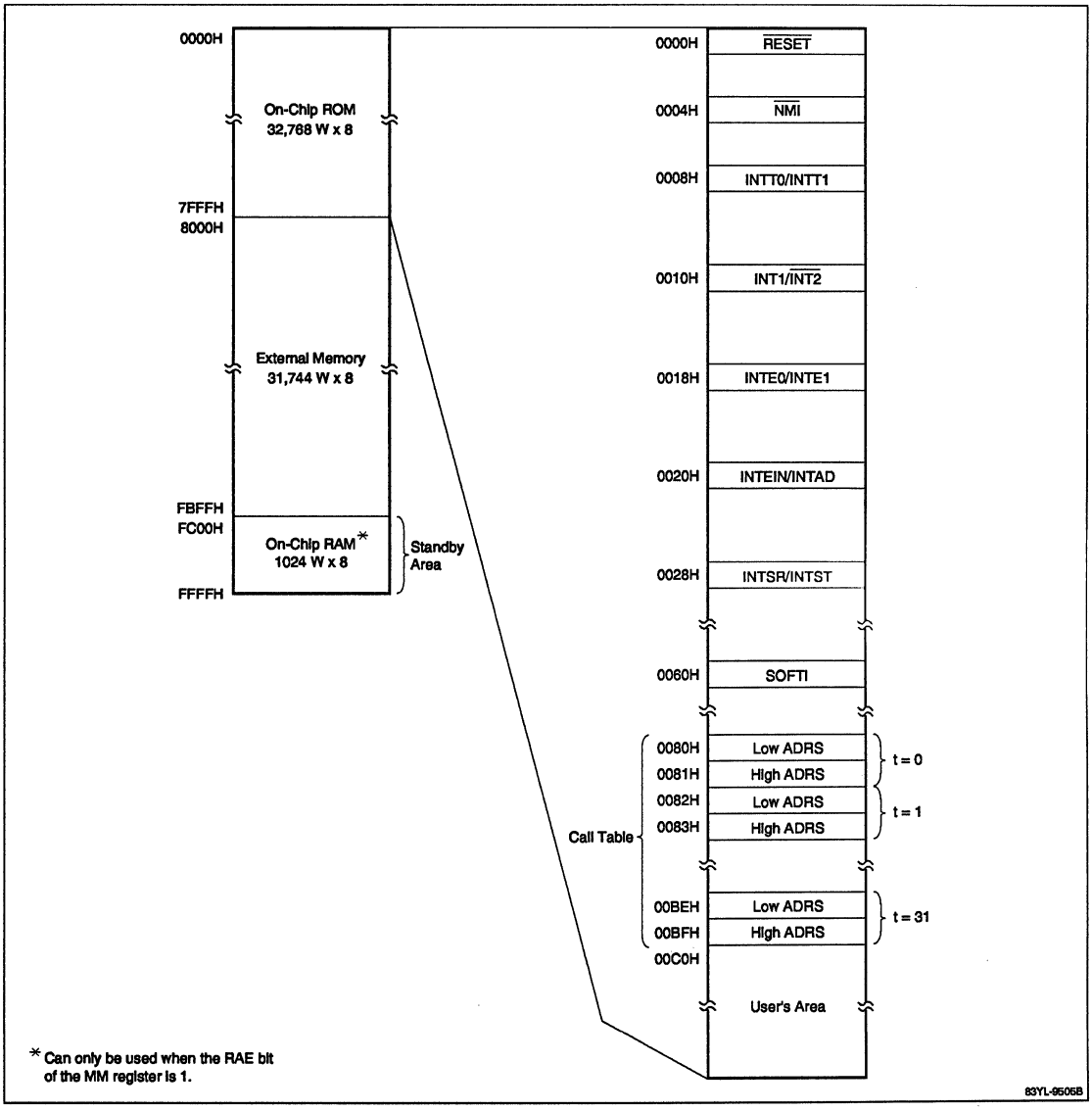
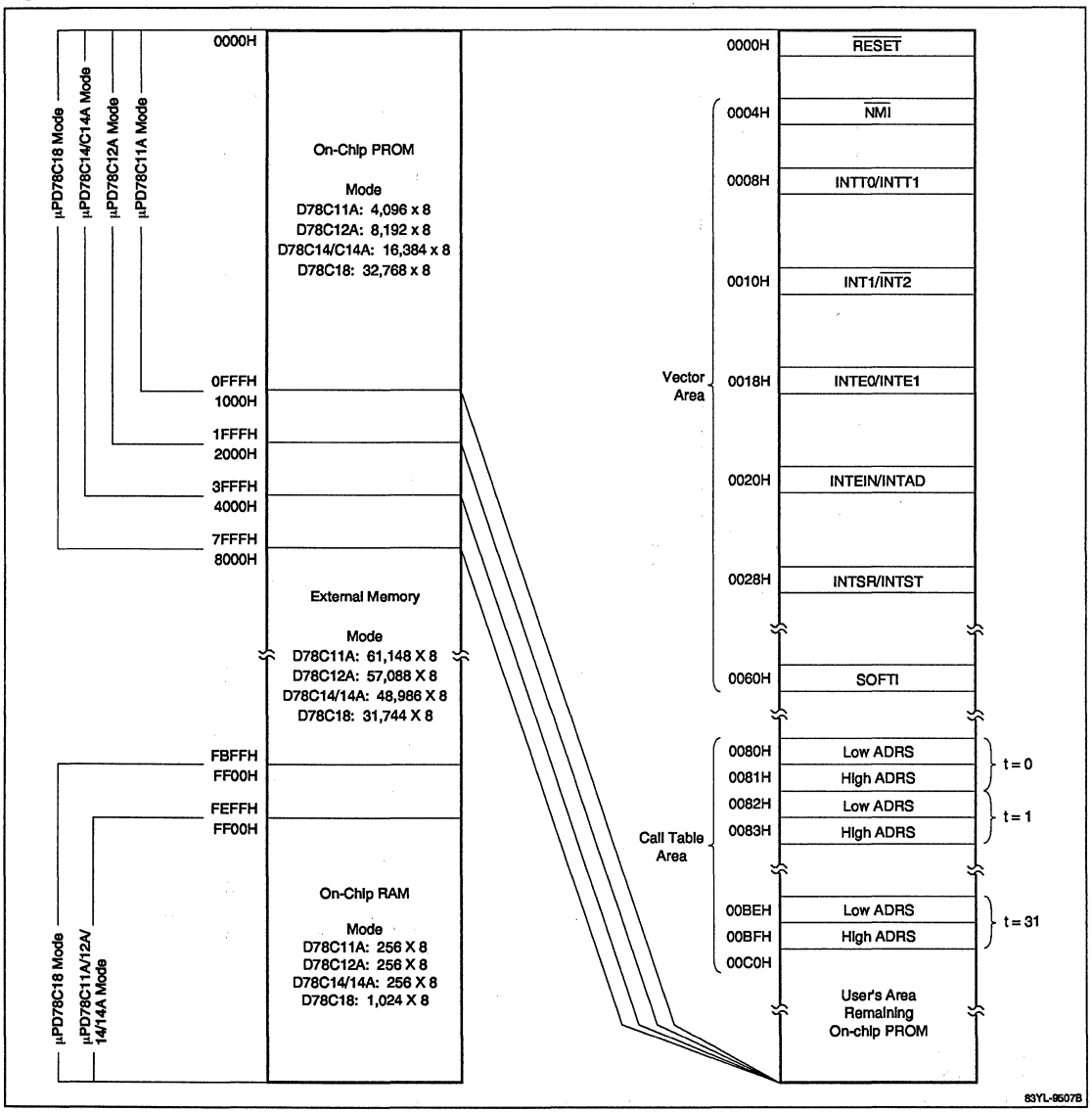


Figure 2. Memory Map (μPD78C18)



2b

Figure 3. Memory Map (μPD78CP18)



Input/Output

The μPD78C18 family has 40 digital I/O lines, consisting of five 8-bit ports (ports A, B, C, D, and F), and four digital input lines (AN4-AN7).

Analog Input Lines. AN0-AN7 are configured as analog input lines for the on-chip A/D converter. Lines AN4-AN7 can be used as digital input lines for falling edge detection.

Port A, Port B, Port C, and Port F. Each line of these ports can be individually programmed as an input or output. When used as I/O ports, all have latched outputs and high impedance inputs. On the μPD78C18, pullup resistors are available as a mask option for ports A, B, and C.

Port D. Port D can be programmed as a byte input or a byte output.

Control Lines. Under software control, each line of port C can be configured individually as a control line for the serial interface, timer, and timer/counter or as an I/O port.

Memory Expansion. In addition to the single-chip operation mode, the μPD78C18 family has four memory expansion modes. Under software control, port D can provide a multiplexed low-order address and data bus; port F can provide a high-order address bus. Table 1 shows the relationship between the memory expansion modes and the pin configurations of port D and port F.

Table 1. Memory Expansion Modes and Port Configurations

Memory Expansion	Port	Port Configuration
None	Port D	I/O port
	Port F	I/O port
256 bytes	Port D	Multiplexed address/data bus
	Port F	I/O port
4K bytes	Port D	Multiplexed address/data bus
	Port F (PF_0-PF_3)	Address bus
	Port F (PF_4-PF_7)	I/O port
16K bytes	Port D	Multiplexed address/data bus
	Port F (PF_0-PF_5)	Address bus
	Port F (PF_6-PF_7)	I/O port
32K/48K/56K/60K bytes (Note 1)	Port D	Multiplexed address/data bus
	Port F	Address bus

Note:

(1) Set according to bits MM7 to MM5.

2b

Timers

The two 8-bit timers can be programmed independently or cascaded as a 16-bit timer. The timer can be set by software to increment at intervals of four machine cycles (0.8 μs at 15 MHz operation) or 128 machine cycles (25.6 μs at 15 MHz), or to increment on receipt of a pulse at TI. Figure 4 is the block diagram for the timer.

Timer/Event Counter

The 16-bit multifunctional timer/event counter, shown in figure 5, can be used for the following operations:

- Interval timing
- External event counting
- Frequency measurement
- Pulse-width measurement
- Programmable frequency and duty cycle waveform output
- Single-pulse output

8-Bit A/D Converter

The 8-bit A/D converter provides the following:

- Eight input channels
- Four conversion result registers
- Two powerful operation modes
  - —Autoscan
  - —Channel select
- Successive approximation technique
- Absolute accuracy: 0.6% FSR ±1/2 LSB
- Conversion range: 0 to 5 V
- Conversion time: 38.4 μs
- Interrupt generation

Figure 4. Timer Block Diagram

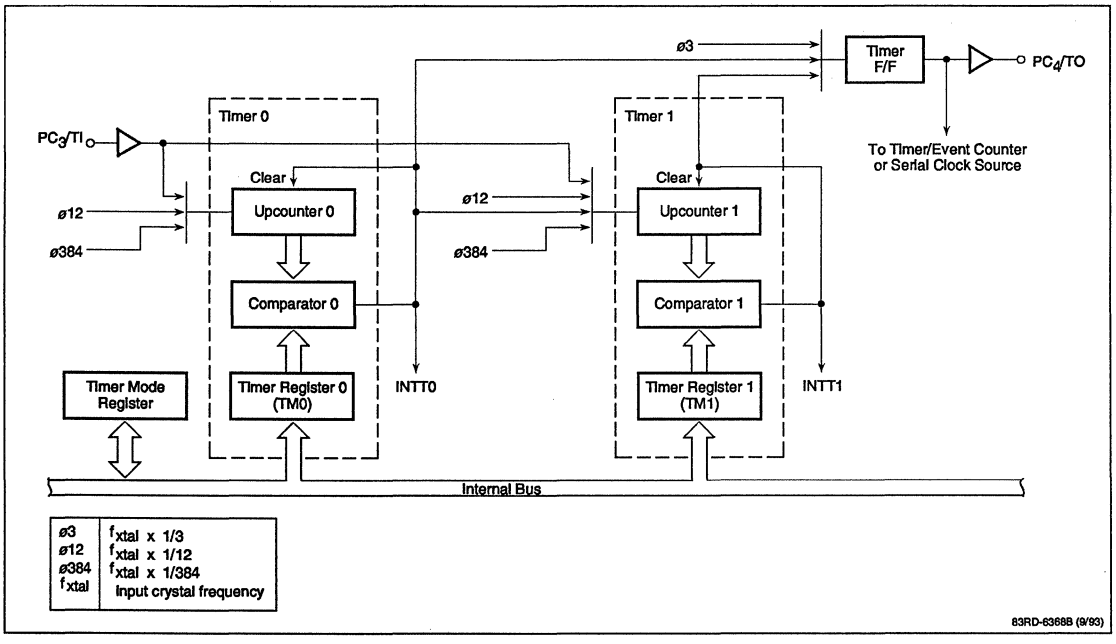
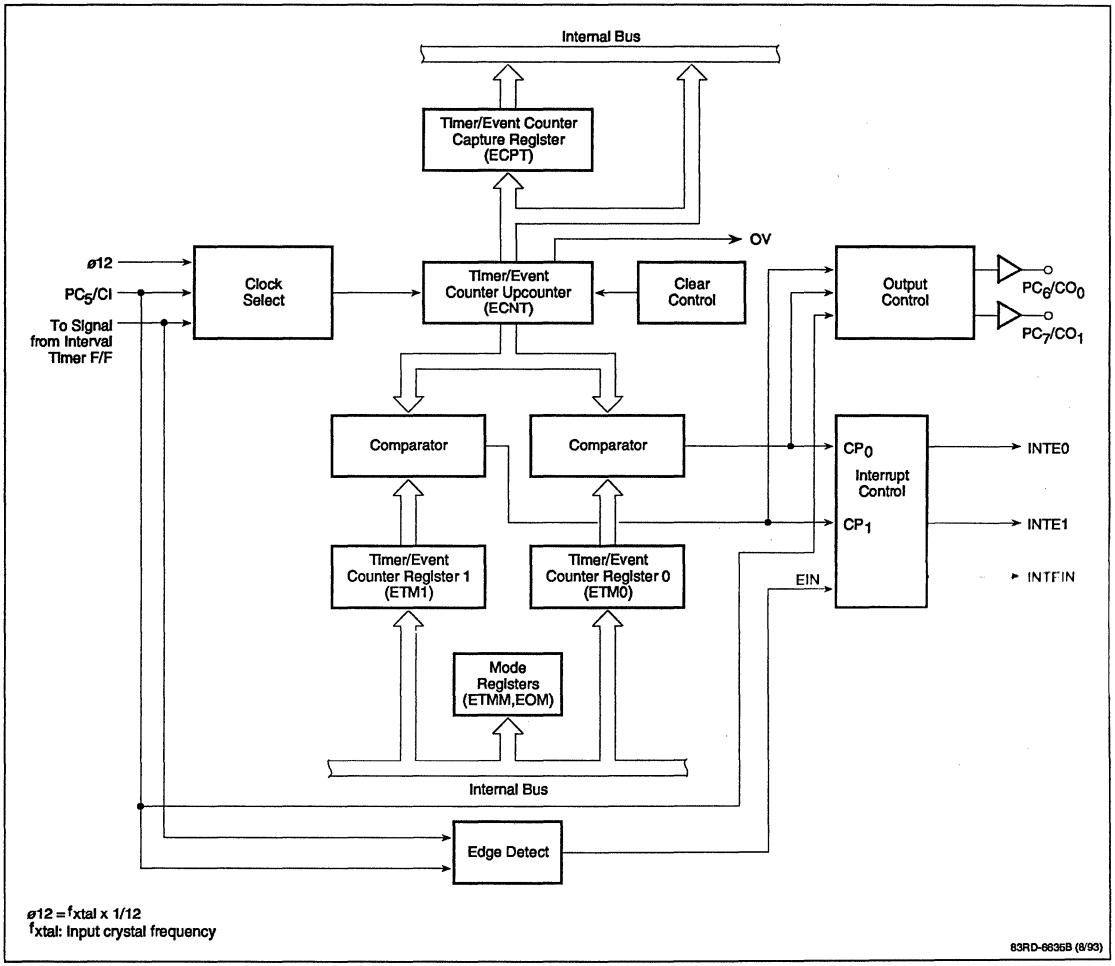


Figure 5. Block Diagram for the Timer/Event Counter



2b

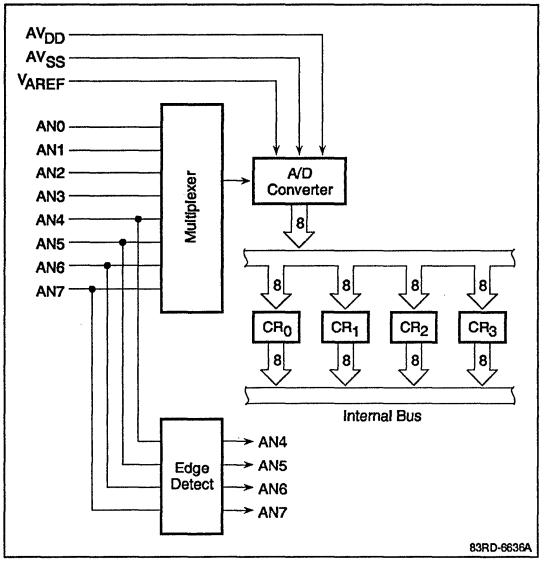
Analog/Digital Converter

The μPD78C18 family features an 8-bit, high-speed, high accuracy A/D converter. The A/D converter is comprised of a 256-resistor ladder and a successive approximation register (SAR). There are four conversion result registers (CR0-CR3).

The eight-channel analog input can be operated in two different modes. In the select mode, the conversion value of one analog input is sequentially stored in CR0-CR3. In the scan mode, either the upper four channels or the lower four channels may be specified. The four channels specified will be consecutively selected and the conversion results stored sequentially in the four conversion result registers.

Figure 6 is the block diagram for the A/D converter. To stop the operation of the A/D converter and reduce the power consumption, set V_{AREF} = 0 V.

Figure 6. A/D Converter Block Diagram



Interrupt Structure

There are 12 interrupt sources in the μPD78C18 family. Three are external and nine are internal interrupt sources. Table 2 shows 11 interrupt sources divided into seven priority levels, where IRQ0 is the highest and IRQ6 is the lowest. See figure 7.

Figure 7. Interrupt Structure Block Diagram

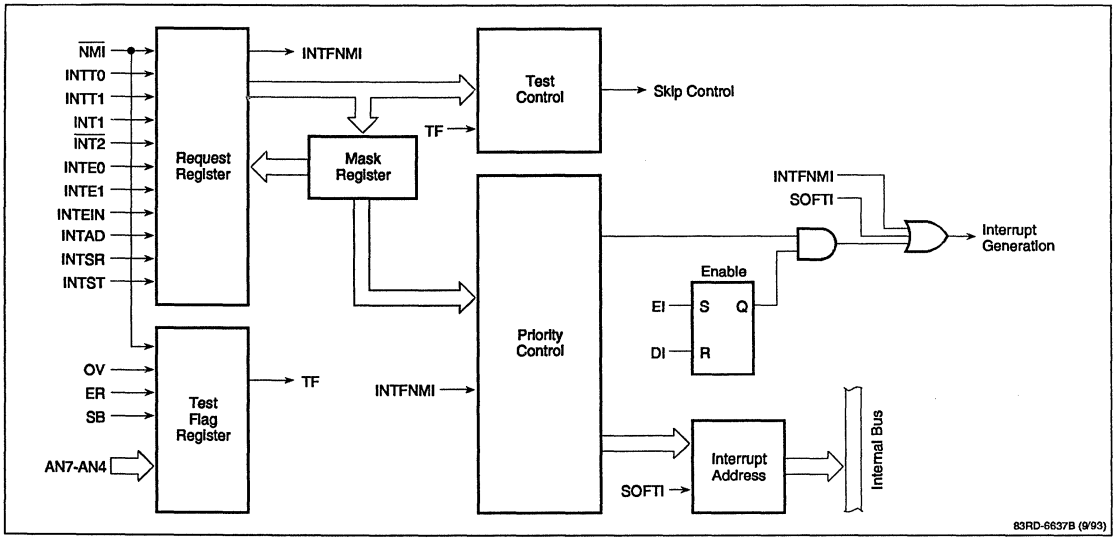


Table 2. Interrupt Sources

Interrupt Request	Interrupt Address	Type of Interrupt	Internal/External
IRQ0	4	$\overline{NMI}$ (nonmaskable interrupt)	External
IRQ1	8	INTT0, INTT1 (coincidence signals from timers 0, 1)	Internal
IRQ2	16	INT1, $\overline{INT2}$ (maskable interrupts)	External
IRQ3	24	INTE0, INTE1 (coincidence signals from timer/event counter)	Internal
IRQ4	32	INTEIN (falling signal of CI or TO into the timer/event counter)	Internal or External
		INTAD (A/D converter interrupt)	Internal
IRQ5	40	INTSR (serial receive interrupt)	Internal
		INST (serial send interrupt)	
IRQ6	96	SOFTI instruction	Internal

2b

Standby Modes

The μPD78C18 family has two standby modes: HALT and STOP.

HALT Mode. The HALT mode reduces power consumption to 50% of normal operating requirements, while maintaining the contents of on-chip registers, RAM, and control status. The system clock and on-board peripherals continue to operate, but the CPU stops executing instructions. The HALT mode is initiated by executing the HLT instruction and can be released by any nonmasked interrupt or by $\overline{RESET}$.

STOP Mode. The STOP mode reduces power consumption to less than 0.1% of normal operating requirements. There are two stop modes: type A and type B.

Type A is initiated by executing a STOP instruction. If V_{DD} is held above 2.5 V, the contents of the on-board RAM are saved. The oscillator is stopped. The stop mode can be released by an input on $\overline{NMI}$ or $\overline{RESET}$. The user can program oscillator stabilization time up to 52.4 ms via timer 1. By checking the standby flag (SB), the user can determine whether the processor has been in the standby mode or has been powered up.

Type B is initiated by inputting a low level on the $\overline{STOP}$ input. The RAM contents are saved if V_{DD} is held above 2.5 V. The oscillator is stopped. The stop mode is released by raising $\overline{STOP}$ to a high level. The oscillator stabilization time is fixed at 52.4 ms; instructions will automatically begin executing at location 0, 52.4 ms after $\overline{STOP}$ is raised. You can increase the stabilization time by holding $\overline{RESET}$ low for the required time period.

Universal Serial Interface

The serial interface can operate in one of three modes: synchronous, asynchronous, and I/O interface. I/O interface mode transfers data most significant bit (MSB) first, for ease of interfacing to certain NEC peripheral devices. Synchronous and asynchronous modes transfer data least significant bit (LSB) first. Synchronous operation offers two modes of data reception: search and nonsearch. In the search mode, data is transferred one bit at a time from the serial register to the receive buffer. This allows a software search for a synchronous character. In the nonsearch mode, data going from the serial register to the transmit buffer is transferred eight bits at a time. Figure 8 shows the universal serial interface block diagram.

Zero-Crossing Detector

The INT1 and $\overline{INT2}$ (common to TI and PC_3) terminals can detect the zero-crossing point of low-frequency AC signals. When driven directly, these pins respond as a normal digital input. Figure 9 shows the zero-crossing detection circuitry.

The zero-crossing detection capability allows you to make the 50-60 Hz power signal the basis for system timing and to control voltage phase-sensitive devices.

To use the zero-crossing detection mode, an AC signal of 1.0 to 1.8 V (peak to peak) and a maximum frequency of 1 kHz is coupled through an external capacitor to the INT1 and $\overline{INT2}$ pins.

For the INT1 pin, the internal digital state is sensed as a 0 until the rising edge crosses the average DC level. It then becomes a 1 and an INT1 interrupt is generated.

For the $\overline{INT2}$ pin, the state is sensed as a 1 until the falling edge crosses the average DC level. It then becomes a 0 and $\overline{INT2}$ is generated.

Figure 8. Universal Serial Interface Block Diagram

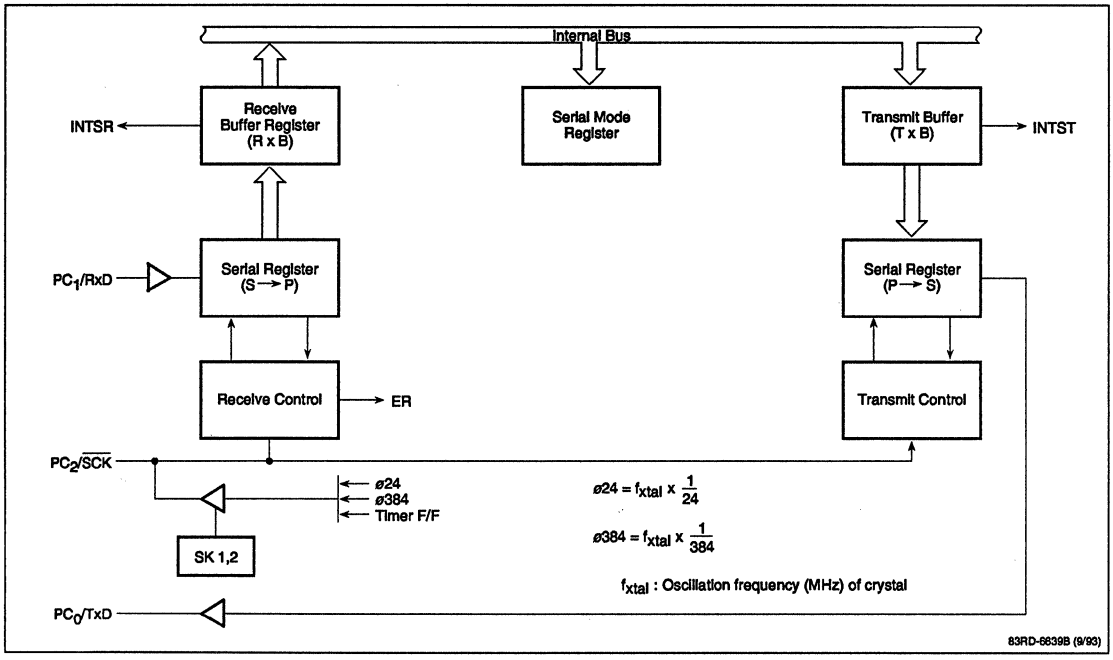
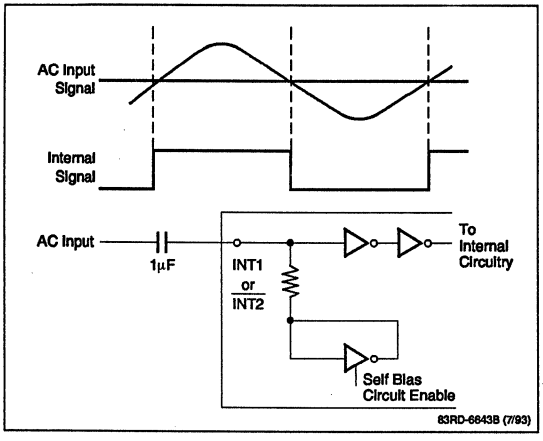


Figure 9. Zero-Crossing Detection Circuit



ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings, μPD78C17/C18

T_A = 25°C

Power supply voltage, V_{DD}	−0.5 to +7.0 V
AV_{DD}	AV_{SS} to V_{DD} +0.5 V
AV_{SS}	−0.5 to +0.5 V
V_{PP} (μPD78CP18 only)	−0.5 to +13.5 V
Input voltage, V_I	−0.5 to V_{DD}+0.5 V
Output voltage, V_O	−0.5 to V_{DD}+0.5 V
Output current, low; I_{OL}	
Each output pin	4.0 mA
Total	100 mA
Output current, high; I_{OH}	
Each output pin	−2.0 mA
Total	−50 mA
Reference input voltage, VA_{REF}	−0.5 to AV_{DD} +0.3 V
Operating temperature, T_{OPR} ($f_{XTAL} \leq$ 15 MHz)	−40 to +85°C
Storage temperature, T_{STG}	−65 to +150°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC Characteristics.

Capacitance

T_A = 25°C; V_{DD} = V_{SS} = 0 V

Parameter	Symbol	Max	Unit	Conditions
Input capacitance	C_I	10	pF	f_c = 1 MHz; unmeasured pins returned to 0 V
Output capacitance	C_O	20	pF	
I/O capacitance	C_{IO}	20	pF	

2b

Oscillation Characteristics

T_A = −40 to +85°C; V_{DD} = AV_{DD} = 5 V ± 10%; V_{SS} = AV_{SS} = 0 V; V_{DD} − 0.8 V ≤ AV_{DD} ≤ V_{DD}; 3.4 V ≤ V_{AREF} ≤ AV_{DD}

Resonator	Recommended Circuit	Parameter	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Note 1) or crystal oscillator (XTAL) (Note 2)	(Note 3)	Oscillation frequency (f_{XX})	4		15	MHz	A/D converter not used
			5.8		15	MHz	A/D converter used
External clock	(Note 4)	X1 input frequency (f_X)	4		15	MHz	A/D converter not used
			5.8		15	MHz	A/D converter used
		X1 input, rise, fall time (t_r, t_f)	0		20	ns	
		X_1 input low- and high-level width ($t_{\phi L}$, $t_{\phi H}$)	20		250	ns	

Notes:

(1) Refer to the Resonator and Capacitance Requirements table for the recommended ceramic resonators.

(2) When using a crystal oscillator, it should be a parallel-resonant, fundamental mode, "AT cut" crystal. Capacitors C1 and C2 are required for frequency stability. The values of C1 and C2 (C1 = C2) can be calculated from the load capacitance (C_L), specified by the crystal manufacturer:

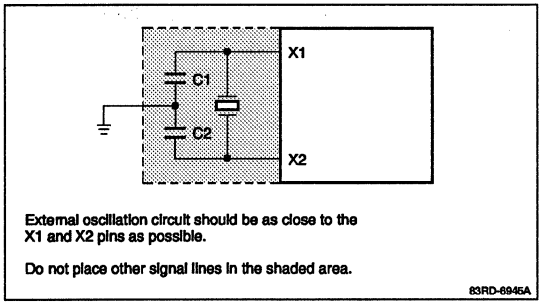
$$C_L = \frac{C1 \times C2}{C1 + C2} + C_S$$

Where C_S is any stray capacitance in parallel with the crystal.

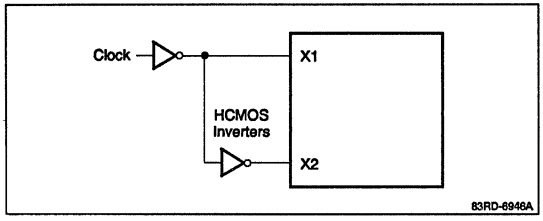
(3) For XTAL, see the Recommended XTAL or Ceramic Resonator Oscillation Circuit Diagram.

(4) See the Recommended External Clock Diagram.

Recommended XTAL or Ceramic Resonator Oscillation Circuit Diagram



Recommended External Clock Diagram



Resonator and Capacitance Requirements

T_A = −40 to +85°C

Manufacturer	Product Number	C1, C2 (pF)
Murata	CSA15.00MX001	22
	CST15.00MXW001	None required
	CSA10.0MT	30
	CST10.0MTW	None required
	CSA8.00MT	30
	CST8.00MTW	None required
TDK	FCR15.0MC	None required
	FCR10.0	None required
	FCR8.0	None required

DC Characteristics

T_A = −40 to +85°C; V_{DD} = AV_{DD} = +5.0 V ±10%; V_{SS} = AV_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input voltage, low	V_{IL1}	0		0.8	V	All except the Note 1 inputs
	V_{IL2}	0		0.2 V_{DD}	V	Note 1 inputs
Input voltage, high	V_{IH1}	2.2		V_{DD}	V	All except X1, X2, and the Note 1 inputs
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	X1, X2, and the Note 1 inputs
Output voltage, low	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output voltage, high	V_{OH}	V_{DD} − 1.0			V	I_{OH} = −1.0 mA
		V_{DD} − 0.5			V	I_{OH} = −100 μA
Input current	I_{I1}			±200	μA	INT1 (Note 2); TI (PC_3) (Note 3); 0 V ≤ V_I ≤ V_{DD}
Input leakage current	I_{LI}			±10	μA	All except INT1; TI (PC_3); 0 V ≤ V_I ≤ V_{DD}
				±1	μA	AN7-0; 0 V ≤ V_I ≤ V_{DD} (μPD78C17(A)/C18(A)/CP18(A) only)
Output leakage current	I_{LO}			±10	μA	0 V ≤ V_O ≤ V_{DD}
AV_{DD} supply current	AI_{DD1}		0.5	1.3	mA	f = 15 MHz
	AI_{DD2}		10	20	μA	Stop mode
V_{DD} supply current	I_{DD1}		16	30	mA	Normal operation; f = 15 MHz
	I_{DD2}		7	13	mA	Halt mode; f = 15 MHz
Data retention voltage	V_{DDDR}	2.5			V	Stop mode
Data retention current	I_{DDDR}		1	15	μA	V_{DDDR} = 2.5 V (Note 4)
			10	50	μA	V_{DDDR} = 5.0 V ±10% (Note 4)
Pullup resistor	R_L	17	27	75	kΩ	Ports A, B, C; 3.5 V ≤ V_{DD} ≤ 5.5 V; V_I = 0 V (μPD78C18/C18(A) only)

Notes:

(1) Inputs $\overline{\text{RESET}}$, $\overline{\text{STOP}}$, $\overline{\text{NMI}}$, $\overline{\text{SCK}}$, INT1, TI, and AN4-AN7.

(2) Assuming ZCM register is set to self-bias.

(3) Assuming ZCM register is set to self-bias and the MCC register is set to the control mode.

(4) Hardware/software stop mode and assuming ZCM register is set so that self-bias is not selected.

AC Characteristics

T_A = −40 to +85°C; V_{DD} = AV_{DD} = +5.0 V ±10%; V_{SS} = AV_{SS} = 0 V

Parameter	Symbol	Min	Max	Unit	Conditions
$\overline{RESET}$ pulse width high, low	t_{RSH}, t_{RSL}	10		μs	
$\overline{NMI}$ pulse width high, low	t_{NIH}, t_{NIH}	10		μs	
X1 input cycle time	t_{CYC}	66	250	ns	
			167	ns	(Note 1)
Address setup to ALE ↓	t_{AL}	30		ns	(Notes 2, 3)
Address hold from ALE ↓	t_{LA}	35		ns	(Notes 2, 3)
Address to $\overline{RD}$ ↓ delay time	t_{AR}	100		ns	(Notes 2, 3)
$\overline{RD}$ ↓ to address floating	t_{AFR}		20	ns	(Note 3)
Address to data input	t_{AD}		250	ns	(Notes 2, 3)
ALE ↓ to data input	t_{LDR}		135	ns	(Notes 2, 3)
$\overline{RD}$ ↓ to data input	t_{RD}		120	ns	(Notes 2, 3)
ALE ↓ to $\overline{RD}$ ↓ delay time	t_{LR}	15		ns	(Notes 2, 3)
Data hold time from $\overline{RD}$ ↑	t_{RDH}	0		ns	(Note 3)
$\overline{RD}$ ↑ to ALE ↑ delay time	t_{RL}	80		ns	(Notes 2, 3)
$\overline{RD}$ width low	t_{RR}	215		ns	Data read (Notes 2, 3)
		415		ns	Opcode fetch (Notes 2, 3)
ALE width high	t_{LL}	90		ns	(Notes 2, 3)
$\overline{M1}$ setup time to ALE ↓	t_{ML}	30		ns	(Note 3)
$\overline{M1}$ hold time after ALE ↓	t_{LM}	35		ns	(Note 3)
$\overline{IO}$/M setup time to ALE ↓	t_{IL}	30		ns	(Note 3)
$\overline{IO}$/M hold time after ALE ↓	t_{LI}	35		ns	(Note 3)
Address to $\overline{WR}$ ↓ delay	t_{AW}	100		ns	(Notes 2, 3)
ALE ↓ to data output	t_{LDW}		180	ns	(Notes 2, 3)
$\overline{WR}$ ↓ to data output	t_{WD}		100	ns	(Note 3)
ALE ↓ to $\overline{WR}$ ↓ delay time	t_{LW}	15		ns	(Notes 2, 3)
Data setup time to $\overline{WR}$ ↑	t_{DW}	165		ns	(Notes 2, 3)
		127		ns	(Note 1)
Data hold time from $\overline{WR}$ ↑	t_{WDH}	60		ns	(Notes 2, 3)
$\overline{WR}$ ↑ to ALE ↑ delay time	t_{WL}	80		ns	(Notes 2, 3)
$\overline{WR}$ width low	t_{WW}	215		ns	(Notes 2, 3)

Notes:

(1) μPD78CP18 only.

(2) Load capacitance C_L = 100 pF.

(3) Values are for 15 MHz operation. For operation at other frequencies, refer to the table labeled Bus Timing Dependent on t_{CYC}.

Serial Operation

Parameter	Symbol	Min	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{CYK}	0.8		μs	$\overline{SCK}$ input (Notes 1, 3)
		0.4		μs	$\overline{SCK}$ input (Note 2)
		1.6		μs	$\overline{SCK}$ output (Note 3)
$\overline{SCK}$ width low	t_{KKL}	335		ns	$\overline{SCK}$ input (Notes 1, 3)
		160		ns	$\overline{SCK}$ input (Note 2)
		700		ns	$\overline{SCK}$ output (Note 3)
$\overline{SCK}$ width high	t_{KKH}	335		ns	$\overline{SCK}$ input (Notes 1, 3)
		160		ns	$\overline{SCK}$ input (Note 2)
		700		ns	$\overline{SCK}$ output (Note 3)
RxD setup time to $\overline{SCK}$ ↑	t_{RXK}	80		ns	(Note 1)
RxD hold time after $\overline{SCK}$ ↑	t_{KRX}	80		ns	(Note 1)
$\overline{SCK}$ ↓ TxD delay time	t_{KTX}		210	ns	(Note 1)

Notes:

(1) 1 x baud rate in asynchronous, synchronous, and I/O interface modes.

(2) 16 x baud rate or 64 x baud rate in asynchronous mode.

(3) f_{XTAL} = 15 MHz.

Zero-Crossing Characteristics

Parameter	Symbol	Min	Max	Unit	Condition
Zero-crossing detection input	V_{ZX}	1	1.8	VAC_{p-p}	AC coupled 60 Hz sine wave
Zero-crossing accuracy	A_{ZX}		±135	mV	
Zero-crossing detection input frequency	f_{ZX}	0.05	1	kHz	

A/D Converter Characteristics

T_A = −40 to +85°C; V_{DD} = +5.0 V ±10%; V_{SS} = AV_{SS} = 0 V; V_{DD} − 0.5 V ≤ AV_{DD} ≤ V_{DD}; 3.4 V ≤ V_{AREF} ≤ AV_{DD}

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		8			bits	
Absolute accuracy (Note 1)				±0.4	%FSR	T_A = −10 to +70°C; 66 ns ≤ t_{CYC} ≤ 170 ns; 4.0 V ≤ V_{AREF} ≤ AV_{DD}
				±0.6	%FSR	66 ns ≤ t_{CYC} ≤ 170 ns; 4.0 V ≤ V_{AREF} ≤ AV_{DD}
				±0.8	%FSR	66 ns ≤ t_{CYC} ≤ 170 ns; 3.4 V ≤ V_{AREF} ≤ AV_{DD}
Conversion time	t_{CONV}	576			t_{CYC}	66 ns ≤ t_{CYC} ≤ 110 ns
		432			t_{CYC}	110 ns ≤ t_{CYC} ≤ 170 ns
Sampling time	t_{SAMP}	96			t_{CYC}	66 ns ≤ t_{CYC} ≤ 110 ns
		72			t_{CYC}	110 ns ≤ t_{CYC} ≤ 170 ns
Analog input voltage	V_{IAN}	0		V_{AREF}	V	
Analog input impedance	R_{AN}		1000		MΩ	
Reference voltage	V_{AREF}	3.4		AV_{DD}	V	
V_{AREF} current	I_{AREF1}		1.5	3.0	mA	Operation mode
	I_{AREF2}		0.7	1.5	mA	Stop mode

A/D Converter Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
AV_{DD} supply current	AI_{DD1}		0.5	1.3	mA	Operation mode; f_{XTAL} = 15 MHz
	AI_{DD2}		10	20	μA	Stop mode

Notes:

(1) Quantizing error (±1/2 LSB) is not included.

(2) FSR = full-scale resolution.

Bus Timing Dependent on t_{CYC}

Symbol	Min/Max (ns)	Calculation Formula
t_{TIH}, t_{TIL}	Min	6T (TI input - PC_3)
t_{CI1H}, t_{CI1L} (Note 2)	Min	6T (TI input - PC_5)
t_{CI2H}, t_{CI2L} (Note 3)	Min	48T (TI input - PC_5)
t_{I1H}, t_{I1L}	Min	36T (INT1)
t_{I2H}, t_{I2L}	Min	36T ($\overline{INT2}$)
t_{ANH}, t_{ANL}	Min	36T (AN4-AN7)
t_{AL}	Min	2T – 100
t_{LA}	Min	T – 30
t_{AR}	Min	3T – 100
t_{AD}	Max	7T – 220
t_{LDR}	Max	5T – 200
t_{RD}	Max	4T – 150
t_{LR}	Min	T – 50
t_{RL}	Min	2T – 50
t_{RR}	Min	4T – 50 (Data read)
	Min	7T – 50 (Opcode fetch)
t_{LL}	Min	2T – 40
t_{ML}	Min	2T – 100
t_{LM}	Min	T – 30
t_{IL}	Min	2T – 100
t_{LI}	Min	T – 30
t_{AW}	Min	3T – 100
t_{LDW}	Max	T + 110
		T + 130 (Note 7)

Symbol	Min/Max (ns)	Calculation Formula
t_{LW}	Min	T – 50
t_{DW}	Min	4T – 100/4T –140 (Note 7)
t_{WDH}	Min	2T – 70
t_{WL}	Min	2T – 50
t_{WW}	Min	4T – 50
t_{CYK}	Min	24T ($\overline{SCK}$ output)
	Min	12T ($\overline{SCK}$ input) (Note 1)
	Min	6T (Note 6)
t_{KKL}	Min	12T – 100 ($\overline{SCK}$ output)
	Min	5T + 5 ($\overline{SCK}$ input) (Note 1)
	Min	2.5T + 5 (Note 6)
t_{KKH}	Min	12T – 100 ($\overline{SCK}$ output)
	Min	5T + 5 ($\overline{SCK}$ input) (Note 1)
	Min	2.5T + 5 (Note 6)

[redacted]s:

(1) 1 x baud rate in asynchronous, synchronous, and I/O interface modes.

(2) Event counter mode.

(3) Pulse width measurement mode.

(4) $T = t_{CYC} = 1/f_{XTAL}$.

(5) The items not included in this list are independent of oscillator frequency (f_{XTAL}).

(6) 16 x baud rate or 64 x baud rate in asynchronous mode

(7) μPD78CP18/CP18(A) only.

Data Memory Stop Mode Data Retention Characteristics

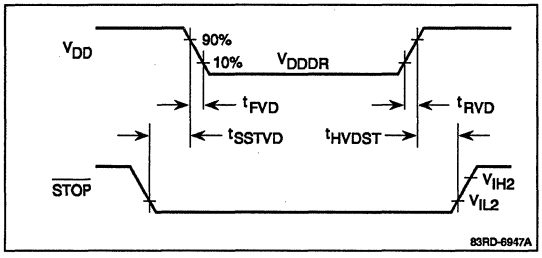
T_A = −40 to +85°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention power supply voltage	V_{DDDR}	2.5		5.5	V	
Data retention power supply current	I_{DDDR}		1	15	μA	V_{DDDR} = 2.5 V
			15	50	μA	V_{DDDR} = 5.0 V ±10%
V_{DD} rise, fall time	t_{RVD}, t_{FVD}	200			μs	
$\overline{STOP}$ setup time to V_{DD}	t_{SSTVD}	12T+0.5			μs	
$\overline{STOP}$ hold time from V_{DD}	t_{HVDST}	12T+0.5			μs	

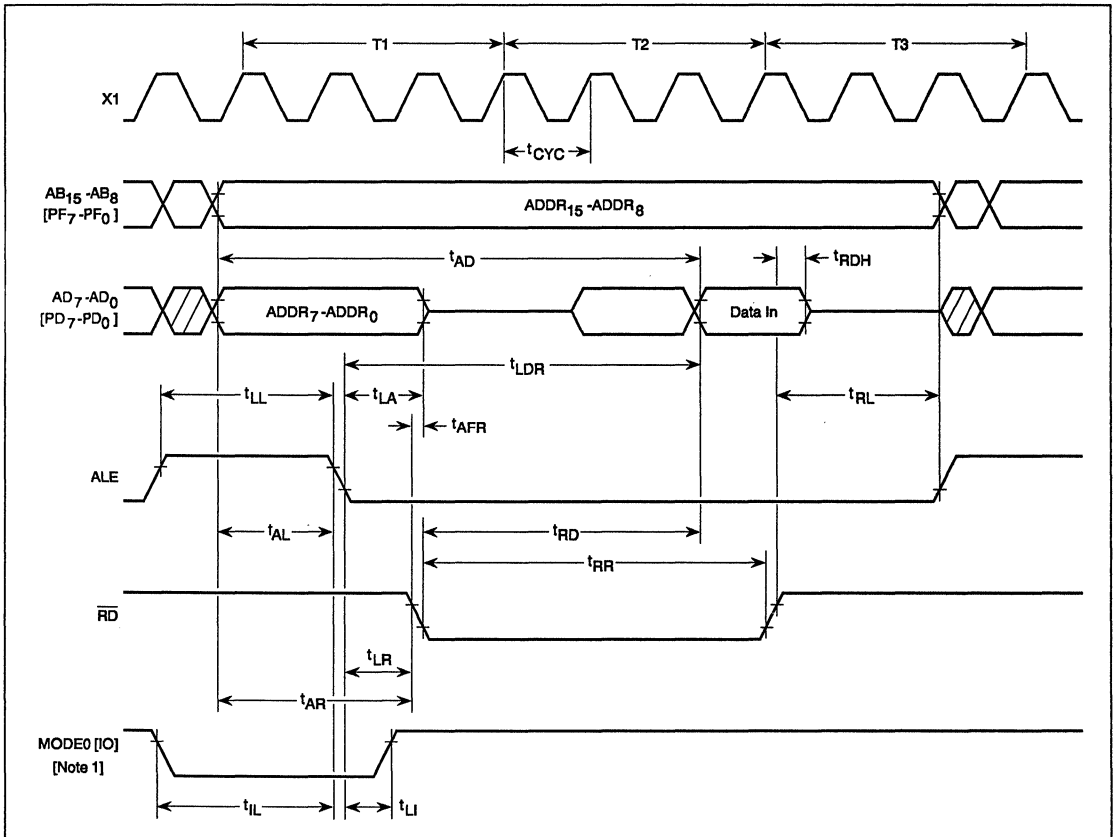
2b

Timing Waveforms

Data Retention



Data Read Operation



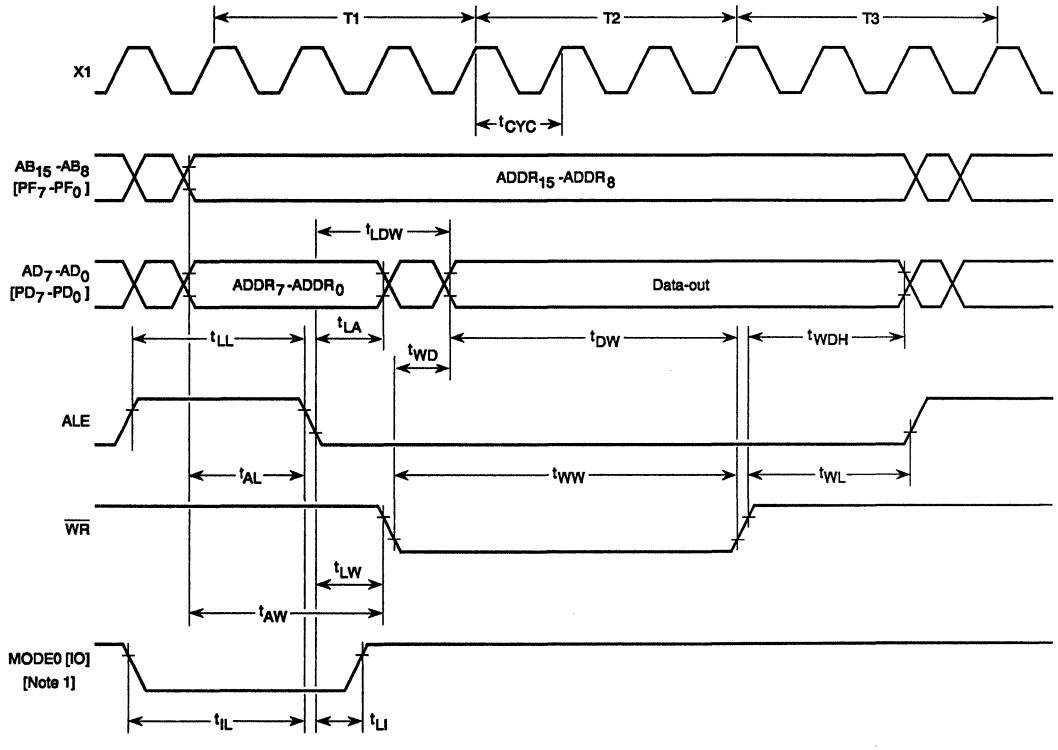
Note:

[1] $\overline{IO}$ signal is output to the MODE0 pin [if MODE0 is pulled up to V_{DD}] during a read or write of special registers sr-sr2.

83YL-9511B

Timing Waveforms (cont)

Data Write Operation



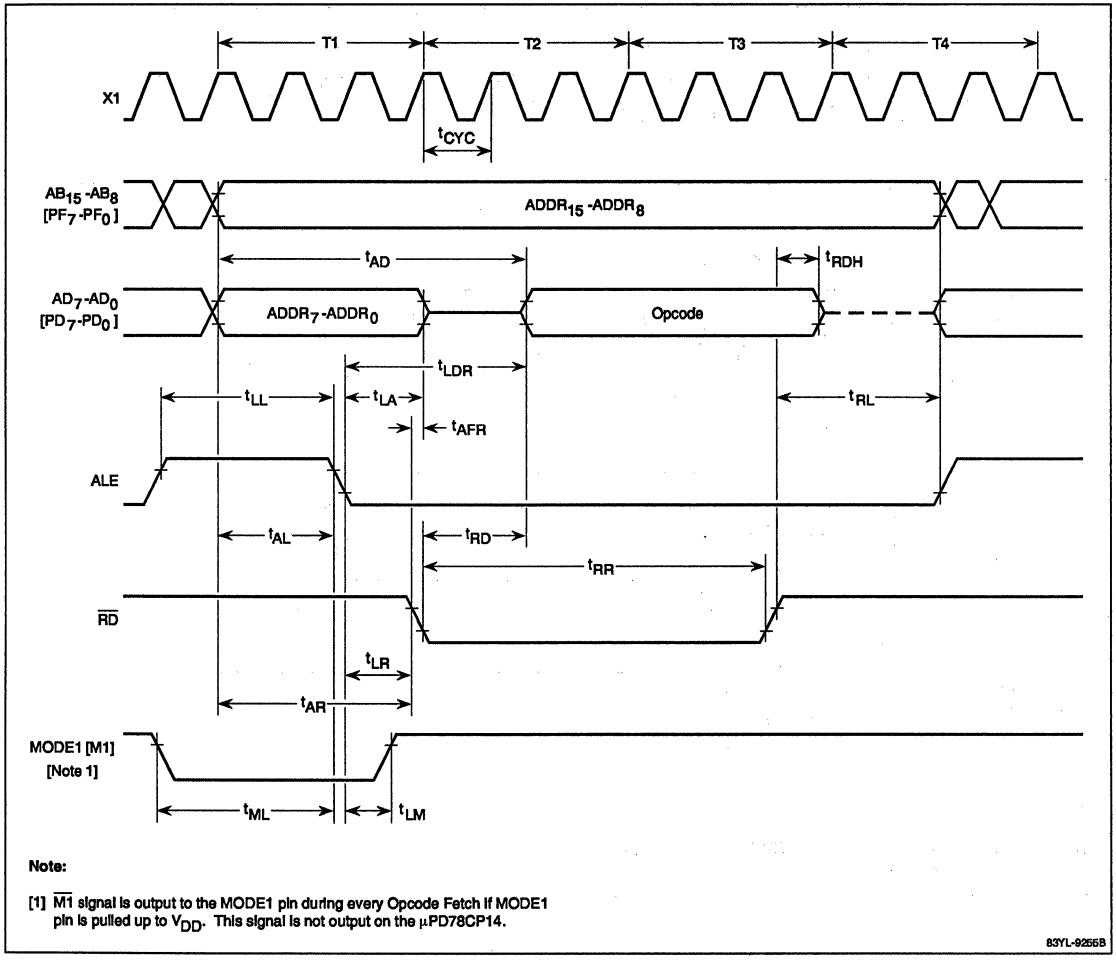
Note:

[1] $\overline{IO}$ signal is output to the MODE0 pin [if MODE0 is pulled up to V_{DD}] during a read or write of special registers sr-sr2.

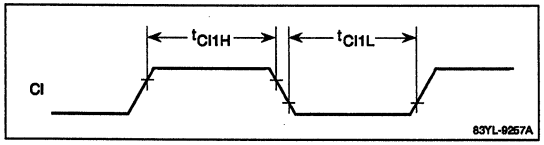
83YL-9512B

Timing Waveforms (cont)

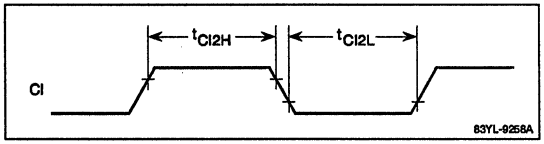
Opcode Fetch Operation



Timer/Event Counter Input: Event Counter Mode

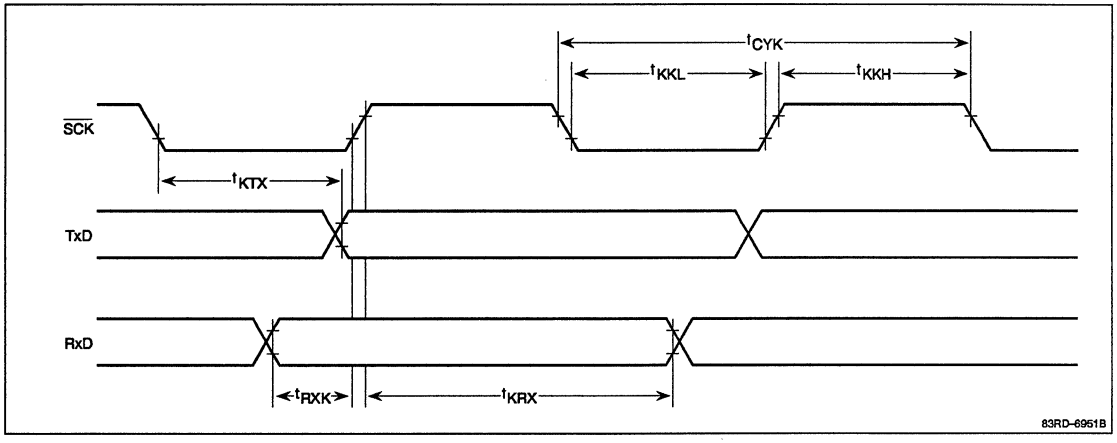


Timer/Event Counter Input: Pulse Width Measurement Mode

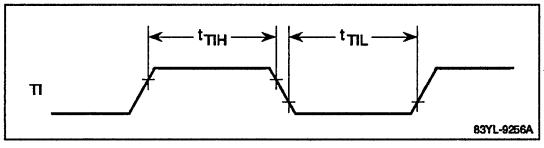


Timing Waveforms (cont)

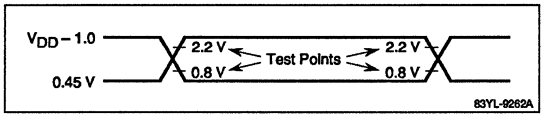
Serial Operation Transmit/Receive



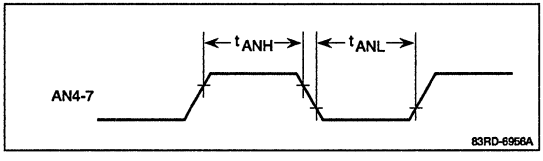
Timer Input



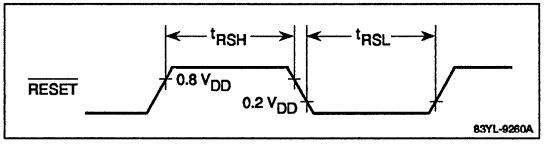
AC Timing Test Points



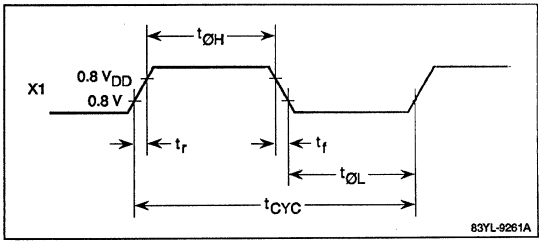
AN4-AN7 Edge Detection



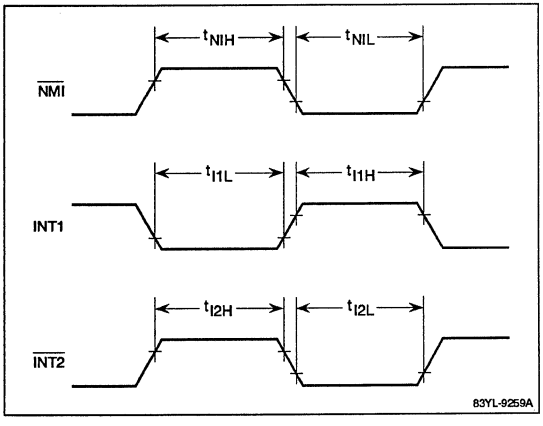
RESET Input



External Clock



Interrupt Input



2b

μPD78CP18 PROGRAMMING

In the μPD78CP18, the mask ROM of the μPD78C18 family is replaced by a one-time programmable ROM (OTP ROM) or a reprogrammable, ultraviolet erasable ROM (UV EPROM). The ROM is 32,768 by 8 bits and can be programmed using a general-purpose PROM writer with a μPD27C256A programming mode. Refer to tables 3 through 5 and the DC and AC Programming Characteristics tables for specific information applicable to programming the μPD78CP18.

Table 3. Pin Functions during EPROM Programming

Pin	Function	Description
PA_0-PA_7	A_0-A_7	Low-order 8-bit address
PF_0	A_8	High-order 7-bit address
$\overline{NMI}$	A_9	
PF_2-PF_6	A_{10}-A_{14}	
PD_0-PD_7	D_0-D_7	Data input/output
PB_6	$\overline{CE}$	Chip enable input
PB_7	$\overline{OE}$	Output enable input
$\overline{RESET}$	$\overline{RESET}$	PROM programming mode requires a low voltage on this pin
Mode 0	Mode 0	Enter PROM programming mode by applying a high voltage to this pin
Mode 1	Mode 1	Enter PROM programming mode by applying a low voltage to this pin
$\overline{STOP}$	V_{PP}	High-voltage input (write/verify) high level (read)

Table 4. Summary of Operation Modes for EPROM Programming

Operation Mode	$\overline{CE}$	$\overline{OE}$	V_{PP}	V_{DD}	$\overline{RESET}$	MODE0	MODE1
Program write	L	H	+12.5 V	+6 V	L	H	L
Program verify	H	L	+12.5 V	+6 V	L	H	L
Program inhibit	H	H	+12.5 V	+6 V	L	H	L
Read	L	L	+5 V	+5 V	L	H	L
Output disable	L	H	+5 V	+5 V	L	H	L
Standby	H	L/H	+5 V	+5 V	L	H	L

Notes:

(1) The $\overline{CE}$, $\overline{OE}$, V_{pp}, and V_{DD} pins are all compatible with the μPD27C256A pins.

Caution: When Vpp is set to +12.5 V and V_{DD} is set to +6 V, you cannot set both $\overline{CE}$ and $\overline{OE}$ to low level (L).

Table 5. Recommended Connections for Unused Pins (EPROM Programming Mode)

Pin	Recommended Connection Method
INT1	Connect to V_{SS}
X1	Connect to V_{SS}
X2	Leave this pin disconnected
AN0-AN7	Connect to V_{SS}
VA_{REF}	Connect to V_{SS}
AV_{DD}	Connect to V_{SS}
AV_{SS}	Connect to V_{SS}
Remaining pins	Connect each pin via a resistor to V_{SS}

PROM Write Procedure

(1) Connect the $\overline{\text{RESET}}$ pin, the MODE1 pin, and A_{14} pin to a low level and connect the MODE0 pin to a high level. Connect all unused pins as recommended in Table 5.

(2) Apply +6 V to the V_{DD} pin and +12.5 V to the V_{pp} pin.

(3) Provide the initial address.

(4) Provide write data.

(5) Provide 1-ms program pulse (active low) to the $\overline{\text{CE}}$ pin.

(6) This bit is now verified with a pulse (active low) to the $\overline{\text{OE}}$ pin. If the data has been written, proceed to step 8; if not, repeat steps 4 to 6. If the data cannot be correctly written after 25 attempts, go to step 7.

(7) Classify as defective and stop write operation.

(8) Provide write data and supply program pulse (for additional writing) for 3 ms times the number of repeats performed between steps 4 to 6.

(9) Increment the address.

(10) Repeat steps 4 to 9 until the end address.

PROM Read Procedure

(1) Connect the $\overline{\text{RESET}}$ pin and the MODE1 pin to a low level and connect the MODE0 pin to a high level.

(2) Apply +5 V to the V_{DD} and V_{pp} pins.

(3) Input the address of the data to be read to pins A_0-A_{14}.

(4) Read mode is entered with a pulse (active low) on both the $\overline{\text{CE}}$ and $\overline{\text{OE}}$ pins.

(5) Data is output to the D_0-D_7 pins.

EPROM Erasure

Data in an EPROM is erased by exposing the quartz window in the ceramic package to light having a wavelength shorter than 400 nm, including ultraviolet rays, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15W-s/cm^2 (ultraviolet ray intensity x exposure time) is required to completely erase written data. Erasure by an ultraviolet lamp rated at 12 mW/cm^2 takes approximately 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

μPD78CP18 DC Programming Characteristics

T_A = 25 ±5°C; MODE1 = V_{IL}; MODE0 = V_{IH}; V_{SS} = 0 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH}	V_{IH}	2.2		V_{DDP}+0.3	V	
Low-level input voltage	V_{IL}	V_{IL}	−0.3		0.8	V	
Input leakage current	I_{LIP}	I_{LI}			±10	μA	$0 \le V_I \le V_{DDP}$
High-level output voltage	V_{OH}	V_{OH}	V_{DD} − 1.0			V	I_{OH} = −1.0 mA
Low-level output voltage	V_{OL}	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output leakage current	I_{LO}				±10	μA	$0 \le V_O \le V_{DDP}$; $\overline{OE}$ = V_{IH}
V_{DDP} power voltage	V_{DDP}	V_{CC}	5.75	6.0	6.25	V	Program memory write mode
			4.5	5.0	5.5	V	Program memory read mode
V_{PP} power voltage	V_{PP}	V_{PP}	12.2	12.5	12.8	V	Program memory write mode
				V_{PP}=V_{DDP}		V	Program memory read mode
V_{DDP} power current	I_{DD}	I_{CC}		5.0	50	mA	Program memory write mode
				5.0	50	mA	Program memory read mode; $\overline{CE}$ = V_{IL}; V_I = V_{IH}
V_{PP} power current	I_{PP}	I_{PP}			30	mA	Program memory read mode; $\overline{CE}$ = V_{IL}; $\overline{OE}$ = V_{IH}
				1	100	μA	Program memory write mode

* Corresponding symbols of the μPD27C256A.

μPD78CP18 AC Programming Characteristics

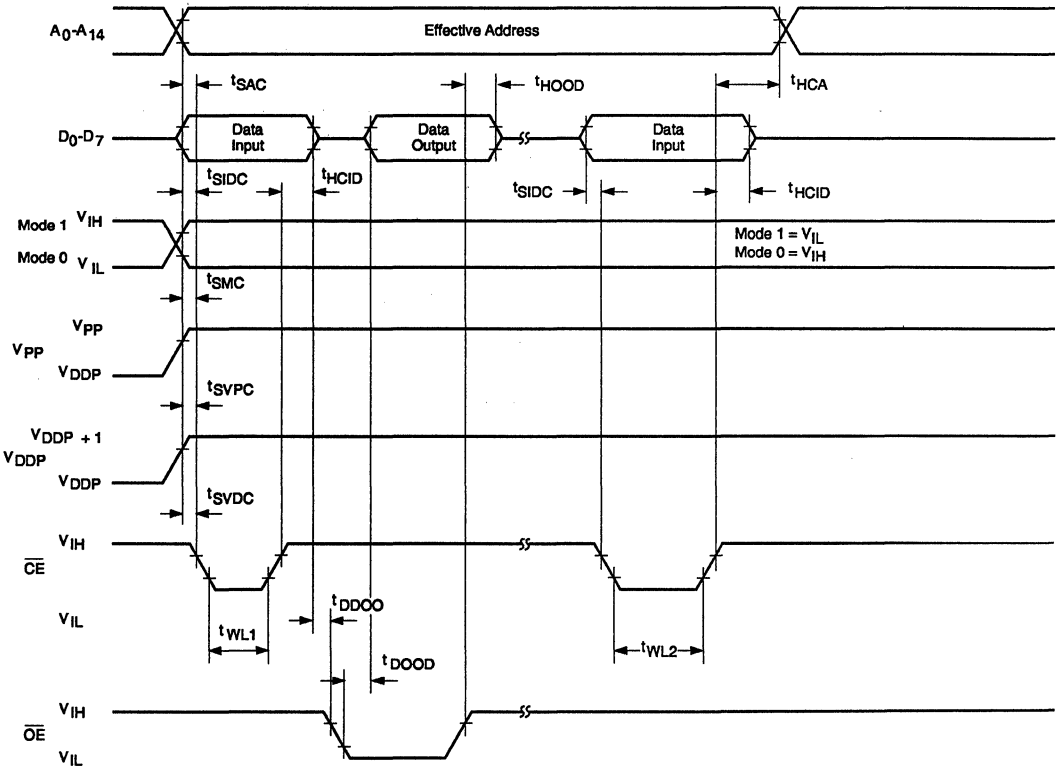
T_A = 25 ±5°C; MODE1 = V_{IL}; V_{SS} = 0 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}	2			μs	
Data to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}	2			μs	
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}	2			μs	
Address hold time from $\overline{CE}$ ↑	t_{HCA}	t_{AH}	2			μs	
Input data hold time from $\overline{CE}$ ↑	t_{HCID}	t_{DH}	2			μs	
Output data hold time from $\overline{OE}$ ↑	t_{HOOD}	t_{DF}	0		130	ns	
V_{PP} setup time to $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}	2			μs	
V_{DDP} setup time to $\overline{CE}$ ↓	t_{SVDC}	t_{VDS}	2			μs	
Initial program pulse width	t_{WL1}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{WL2}	t_{OPW}	2.85		78.75	ms	
MODE0/MODE1 setup time vs. $\overline{CE}$ ↓	t_{SMC}		2			μs	MODE1 = V_{IL} and MODE0 = V_{IH}
Address to data output time	t_{DAOD}	t_{ACC}			2	μs	$\overline{OE}$ = V_{IL}
$\overline{CE}$ ↓ to data output time	t_{DCOD}	t_{CE}			1	μs	
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			1	μs	
Data hold time from $\overline{OE}$ ↑ or $\overline{CE}$ ↑	t_{HCOD}	t_{DF}	0		130	ns	
Data hold time from address	t_{HAOD}	t_{OH}	0			ns	$\overline{OE}$ = V_{IL}

* Corresponding symbols of the μPD27C256A.

PROM Timing Diagrams

μPD78CP18 PROM Write Mode



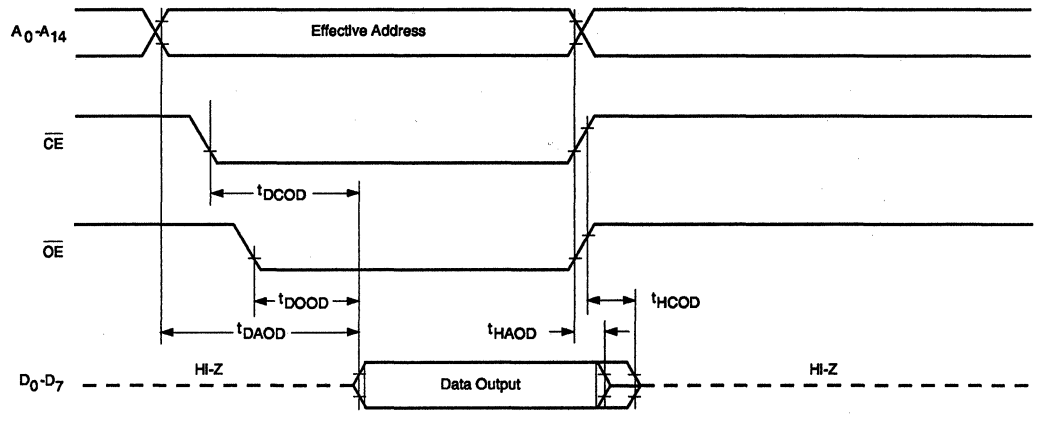
Notes:
(1) V_{DDP} must be applied before applying V_{PP}. It should be removed after removing V_{PP}.
(2) V_{PP} must not exceed +13 V, including overshoot.

83RD-6948B

2b

PROM Timing Diagrams (cont)

μPD78CP18 PROM Read Mode



Notes:

(1) To read PROM within the t_{DAOD} range, the delay of $\overline{OE}\downarrow$ from $\overline{CE}\downarrow$ must be within t_{DAOD}- t_{DOOD}.

(2) t_{HCOD} is the time from the state in which either $\overline{OE}$ or $\overline{CE}$ first becomes V_{IH}.

83RD-6949B

INSTRUCTION SET

Operand Symbols

Symbol	Allowable Operands
Registers	
r	V, A, B, C, D, E, H, L
r1	EAH, EAL, B, C, D, E, H, L
r2	A, B, C
Special Registers	
sr	PA, PB, PC, PD, PF, MKH, MKL, ANM, SMH, SML, EOM, ETMM, TMM, MM, MCC, MA, MB, MC, MF, TXB, TM0, TM1, ZCM
sr1	PA, PB, PC, PD, PF, MKH, MKL, ANM, SMH, EOM, TMM, RXB, CR0, CR1, CR2, CR3
sr2	PA, PB, PC, PD, PF, MKH, ANM, MKL, SMH, EOM, TMM
sr3	ETM0, ETM1
sr4	ECNT, ECPT
Register Pairs	
rp	SP, B, D, H
rp1	V, B, D, H, EA
rp2	SP, B, D, H, EA
rp3	B, D, H
Register Pair Addressing	
rpa	B, D, H, D+, H+, D–, H–
rpa1	B, D, H
rpa2	B, D, H, D+, H+, D–, H–, D+ byte, H+A, H+B, H+EA, H+ byte
rpa3	D, H, D++, H++, D+ byte, H+A, H+B, H+EA, H+ byte
Flags	
f	CY, HC, Z
Interrupt Flags	
irf	INTFNMI, INTFT0, INTFT1, INTF1, INTF2, INTFE0, INTFE1, INTFEIN, INTFAD, INTFSR, INTFST, ER, OV, AN4, AN5, AN6, AN7, SB
Immediate Data	
wa	8-bit immediate data (low byte of working register address)
word	16-bit immediate data
byte	8-bit immediate data
bit	3-bit immediate data (b_2, b_1, b_0)

Operand Definitions

Special registers (sr-sr4)

PA = Port A	ECNT = Timer/event counter upcounter
PB = Port B	ECPT = Timer/event counter capture
PC = Port C	ETMM = Timer/event counter mode
PD = Port D	
PF = Port F	
MA = Mode A	
MB = Mode B	
MC = Mode C	EOM = Timer/event counter output mode
MCC = Mode control C	
MF = Mode F	
	TXB = Transmit buffer
MM = Memory mapping	RXB = Receive buffer
TM0 = Timer register 0	SMH = Serial mode high
TM1 = Timer register 1	SML = Serial mode low
TMM = Timing mode	MKH = Mask high
ETM0 = Timer/ event counter register 0	MKL = Mask low
	ANM = A/D channel mode
ETM1 = Timer/event counter register 1	CR0 to CR3 = A/D conversion result 0-3
ZCM = Zero-cross mode control register	

Register Pairs (rp-rp3)

SP = Stack pointer	H = HL
B = BC	V = VA
D = DE	EA = Extended accumulator

Register Pair Addressing (rpa-rpa3)

B = (BC)	D++ = (DE)++
D = (DE)	H++ = (HL)++
H = (HL)	D+ byte = (DE+ byte)
D+ = (DE)+	H+ byte = (HL+ byte)
H+ = (HL)+	H+A = (HL+A)
D– = (DE)–	H+B = (HL+B)
H– = (HL)–	H+EA = (HL+EA)

Flags (f)

CY = Carry	HC = Half-carry	Z = Zero

Interrupt Flags (Irf)

INTFNMI = NMI interrupt flag	INTFEIN = FEIN
	INTFAD = FAD
	INTFSR = FSR
INTFT0 = FT0	INTFST = FST
INTFT1 = FT1	ER = Error
INTF1 = F1	OV = Overflow
INTF2 = F2	AN4 to AN7 = Analog input 4-7
INTFE0 = FE0	SB = Standby
INTFE1 = FE1	

Operand Codes

Registers (r, r2)

R_2	R_1	R_0	Reg	Applicable to
0	0	0	V	r
0	0	1	A	r, r2
0	1	0	B	
0	1	1	C	
1	0	0	D	r
1	0	1	E	
1	1	0	H	
1	1	1	L	

Register (r1)

T_2	T_1	T_0	Reg
0	0	0	EAH
0	0	1	EAL
0	1	0	B
0	1	1	C
1	0	0	D
1	0	1	E
1	1	0	H
1	1	1	L

Special Registers (sr, sr1, sr2)

S_5	S_4	S_3	S_2	S_1	S_0	Special Reg	Applicable to
0	0	0	0	0	0	PA	sr, sr1, sr2
0	0	0	0	0	1	PB	
0	0	0	0	1	0	PC	
0	0	0	0	1	1	PD	
0	0	0	1	0	1	PF	
0	0	0	1	1	0	MKH	
0	0	0	1	1	1	MKL	
0	0	1	0	0	0	ANM	
0	0	1	0	0	1	SMH	
0	0	1	0	1	0	SML	sr
0	0	1	0	1	1	EOM	sr, sr1, sr2
0	0	1	1	0	0	ETMM	sr
0	0	1	1	0	1	TMM	sr, sr1, sr2
0	1	0	0	0	0	MM	sr
0	1	0	0	0	1	MCC	
0	1	0	0	1	0	MA	
0	1	0	0	1	1	MB	
0	1	0	1	0	0	MC	
0	1	0	1	1	1	MF	
0	1	1	0	0	0	TXB	
0	1	1	0	0	1	RXB	sr1
0	1	1	0	1	0	TM0	sr
0	1	1	0	1	1	TM1	
1	0	0	0	0	0	CR0	sr1
1	0	0	0	0	1	CR1	
1	0	0	0	1	0	CR2	
1	0	0	0	1	1	CR3	
1	0	1	0	0	0	ZCM	sr

Special Registers (sr3)

U_0	Special Reg
0	ETM0
1	ETM1

Special Registers (sr4)

V_0	Special Reg
0	ECNT
1	ECPT

Register Pairs (rp, rp2, rp3)

P_2	P_1	P_0	Reg Pair	Applicable to
0	0	0	SP	rp, rp2
0	0	1	BC	rp, rp2, rp3
0	1	0	DE	
0	1	1	HL	
1	0	0	EA	rp2

Register Pairs (rp1)

Q_2	Q_1	Q_0	Reg Pair
0	0	0	VA
0	0	1	BC
0	1	0	DE
0	1	1	HL
1	0	0	EA

Register Pair Addressing (rpa, rpa1, rpa2)

A_3	A_2	A_1	A_0	Addressing	Applicable to
0	0	0	0	—	rpa, rpa1, rpa2
0	0	0	1	(BC)	
0	0	1	0	(DE)	
0	0	1	1	(HL)	
0	1	0	0	(DE)+	rpa, rpa2
0	1	0	1	(HL)+	
0	1	1	0	(DE)–	
0	1	1	1	(HL)–	
1	0	1	1	(DE+ byte)	rpa2
1	1	0	0	(HL+ A)	
1	1	0	1	(HL+ B)	
1	1	1	0	(HL+ EA)	
1	1	1	1	(HL+ byte)	

Register Pair Addressing (rpa3)

C_3	C_2	C_1	C_0	Addressing
0	0	1	0	(DE)
0	0	1	1	(HL)
0	1	0	0	(DE)+ +
0	1	0	1	(HL)+ +
1	0	1	1	(DE+ byte)
1	1	0	0	(HL+ A)
1	1	0	1	(HL+ B)
1	1	1	0	(HL+ EA)
1	1	1	1	(HL+ byte)

Operand Codes (cont)

Flags (f)

F_2	F_1	F_0	Flag
0	0	0	—
0	1	0	CY
0	1	1	HC
1	0	0	Z

Interrupt Flags (irf)

I_4	I_3	I_2	I_1	I_0	Flag
0	0	0	0	0	NMI
0	0	0	0	1	FT0
0	0	0	1	0	FT1
0	0	0	1	1	F1
0	0	1	0	0	F2
0	0	1	0	1	FE0
0	0	1	1	0	FE1
0	0	1	1	1	FEIN
0	1	0	0	0	FAD
0	1	0	0	1	FSR
0	1	0	1	0	FST
0	1	0	1	1	ER
0	1	1	0	0	OV
1	0	0	0	0	AN4
1	0	0	0	1	AN5
1	0	0	1	0	AN6
1	0	0	1	1	AN7
1	0	1	0	0	SB

Graphic Symbols

Symbol	Description
←	Transfer direction, result
∧	Logical product (logical AND)
∨	Logical sum (logical OR)
⊻	Exclusive-OR
—	Complement
•	Concatenation

2c

Instruction Set

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
8-Bit Data Transfer													
MOV	r1, A	r1 ← A	1	4		0	0	0	1	1	T_2	T_1	T_0
	A, r1	A ← r1	1	4		0	0	0	0	1	T_2	T_1	T_0
	*sr, A	sr ← A	2	10		0	1	0	0	1	1	0	1
						1	1	S_5	S_4	S_3	S_2	S_1	S_0
	*A, sr1	A ← sr1	2	10		0	1	0	0	1	1	0	0
						1	1	S_5	S_4	S_3	S_2	S_1	S_0
	r, word	r ← (word)	4	17		0	1	1	1	0	0	0	0
						0	1	1	0	1	R_2	R_1	R_0
						Low addr							
						High addr							
	word, r	(word) ← r	4	17		0	1	1	1	0	0	0	0
						0	1	1	1	1	R_2	R_1	R_0
						Low addr							
						High addr							
MVI	*r, byte	r ← byte	2	7		0	1	1	0	1	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 ← byte	3	14		0	1	1	0	0	1	0	0
						S_3	0	0	0	0	S_2	S_1	S_0
						Data							
MVIW	*wa, byte	(V•wa) ← byte	3	13		0	1	1	1	0	0	0	1
						Offset							
						Data							
MVIX	*rpa1, byte	(rpa1) ← byte	2	10		0	1	0	0	1	0	A_1	A_0
						Data							
STAW	*wa	(V•wa) ← A	2	10		0	1	1	0	0	0	1	1
						Offset							
LDAW	*wa	A ← (V•wa)	2	10		0	0	0	0	0	0	0	1
						Offset							
STAX	*rpa2	(rpa2) ← A	2	7/13 (Note 3)		A_3	0	1	1	1	A_2	A_1	A_0
						Data (Note 2)							
LDAX	*rpa2	A ← (rpa2)	2	7/13 (Note 3)		A_3	0	1	0	1	A_2	A_1	A_0
						Data (Note 2)							
EXX		B ↔ B', C ↔ C', D ↔ D' E ↔ E', H ↔ H', L ↔ L'	1	4		0	0	0	1	0	0	0	1
EXA		V ↔ V', A ↔ A', EA ↔ EA'	1	4		0	0	0	1	0	0	0	0
EXH		H ↔ H', L ↔ L'	1	4		0	1	0	1	0	0	0	0
BLOCK		(DE) ← (HL), DE ← DE + 1, HL ← HL + 1, C ← C − 1 End if borrow	1	13 x (C + 1)		0	0	1	1	0	0	0	1

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
16-Bit Data Transfer													
DMOV	rp3, EA	$rp3_L \leftarrow EAL$, $rp3_H \leftarrow EAH$	1	4		1	0	1	1	0	1	P_1	P_0
	EA, rp3	$EAL \leftarrow rp3_L$, $EAH \leftarrow rp3_H$	1	4		1	0	1	0	0	1	P_1	P_0
	sr3, EA	sr3 ← EA	2	14		0	1	0	0	1	0	0	0
						1	1	0	1	0	0	1	U_0
	EA, sr4	EA ← sr4	2	14		0	1	0	0	1	0	0	0
						1	1	0	0	0	0	0	V_0
SBCD	word	(word) ← C, (word + 1) ← B	4	20		0	1	1	1	0	0	0	0
						0	0	0	1	1	1	1	0
						Low addr							
						High addr							
SDED	word	(word) ← E, (word + 1) ← D	4	20		0	1	1	1	0	0	0	0
						0	0	1	0	1	1	1	0
						Low addr							
						High addr							
SHLD	word	(word) ← L, (word + 1) ← H	4	20		0	1	1	1	0	0	0	0
						0	0	1	1	1	1	1	0
						Low addr							
						High addr							
SSPD	word	(word) ← SP_L, (word + 1) ← SP_H	4	20		0	1	1	1	0	0	0	0
						0	0	0	0	1	1	1	0
						Low addr							
						High addr							
STEAX	rpa3	(rpa3) ← EAL, (rpa3 + 1) ← EAH	3	14/20 (Note 3)		0	1	0	0	1	0	0	0
						1	0	0	1	C_3	C_2	C_1	C_0
						Data (Note 4)							
LBCD	word	C ← (word), B ← (word + 1)	4	20		0	1	1	1	0	0	0	0
						0	0	0	1	1	1	1	1
						Low addr							
						High addr							
LDED	word	E ← (word), D ← (word + 1)	4	20		0	1	1	1	0	0	0	0
						0	0	1	0	1	1	1	1
						Low addr							
						High addr							
LHLD	word	L ← (word), H ← (word + 1)	4	20		0	1	1	1	0	0	0	0
						0	0	1	1	1	1	1	1
						Low addr							
						High addr							

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
16-Bit Data Transfer (cont)													
LSPD	word	$SP_L \leftarrow$ (word), $SP_H \leftarrow$ (word + 1)	4	20		0	1	1	1	0	0	0	0
						0	0	0	0	1	1	1	1
						Low addr							
						High addr							
LDEAX	rpa3	EAL ← (rpa3), EAH ← (rpa3 + 1)	3	14/20 (Note 3)		0	1	0	0	1	0	0	0
						1	0	0	0	C_3	C_2	C_1	C_0
						Data (Note 4)							
PUSH	rp1	$(SP - 1) \leftarrow rp1_H$, $(SP - 2) \leftarrow rp1_L$, SP ← SP − 2	1	13		1	0	1	1	0	Q_2	Q_1	Q_0
POP	rp1	$rp1_L \leftarrow$ (SP), $rp1_H \leftarrow$ (SP + 1) SP ← SP + 2	1	10		1	0	1	0	0	Q_2	Q_1	Q_0
LXI	*rp2, word	rp2 ← word	3	10		0	P_2	P_1	P_0	0	1	0	0
						Low byte							
						High byte							
TABLE		C ← (PC + 3 + A), B ← (PC + 3 + A + 1)	2	17		0	1	0	0	1	0	0	0
						1	0	1	0	1	0	0	0
8-Bit Arithmetic (Register)													
ADD	A, r	A ← A + r	2	8		0	1	1	0	0	0	0	0
						1	1	0	0	0	R_2	R_1	R_0
	r, A	r ← r + A	2	8		0	1	1	0	0	0	0	0
						0	1	0	0	0	R_2	R_1	R_0
ADC	A, r	A ← A + r + CY	2	8		0	1	1	0	0	0	0	0
						1	1	0	1	0	R_2	R_1	R_0
	r, A	r ← r + A + CY	2	8		0	1	1	0	0	0	0	0
						0	1	0	1	0	R_2	R_1	R_0
ADDDNC	A, r	A ← A + r	2	8	No carry	0	1	1	0	0	0	0	0
						1	0	1	0	0	R_2	R_1	R_0
	r, A	r ← r + A	2	8	No carry	0	1	1	0	0	0	0	0
						0	0	1	0	0	R_2	R_1	R_0
SUB	A, r	A ← A − r	2	8		0	1	1	0	0	0	0	0
						1	1	1	0	0	R_2	R_1	R_0
	r, A	r ← r − A	2	8		0	1	1	0	0	0	0	0
						0	1	1	0	0	R_2	R_1	R_0
SBB	A, r	A ← A − r − CY	2	8		0	1	1	0	0	0	0	0
						1	1	1	1	0	R_2	R_1	R_0
	r, A	r ← r − A − CY	2	8		0	1	1	0	0	0	0	0
						0	1	1	1	0	R_2	R_1	R_0

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
8-Bit Arithmetic (Register) (cont)													
SUBNB	A, r	A ← A – r	2	8	No borrow	0	1	1	0	0	0	0	0
						1	0	1	1	0	R_2	R_1	R_0
	r, A	r ← r – A	2	8	No borrow	0	1	1	0	0	0	0	0
						0	0	1	1	0	R_2	R_1	R_0
ANA	A, r	A ← A ∧ r	2	8		0	1	1	0	0	0	0	0
						1	0	0	0	1	R_2	R_1	R_0
	r, A	r ← r ∧ A	2	8		0	1	1	0	0	0	0	0
						0	0	0	0	1	R_2	R_1	R_0
ORA	A, r	A ← A ∨ r	2	8		0	1	1	0	0	0	0	0
						1	0	0	1	1	R_2	R_1	R_0
	r, A	r ← r ∨ A	2	8		0	1	1	0	0	0	0	0
						0	0	0	1	1	R_2	R_1	R_0
XRA	A, r	A ← A ⊻ r	2	8		0	1	1	0	0	0	0	0
						1	0	0	1	0	R_2	R_1	R_0
	r, A	r ← r ⊻ A	2	8		0	1	1	0	0	0	0	0
						0	0	0	1	0	R_2	R_1	R_0
GTA	A, r	A – r – 1	2	8	No borrow	0	1	1	0	0	0	0	0
						1	0	1	0	1	R_2	R_1	R_0
	r, A	r – A – 1	2	8	No borrow	0	1	1	0	0	0	0	0
						0	0	1	0	1	R_2	R_1	R_0
LTA	A, r	A – r	2	8	Borrow	0	1	1	0	0	0	0	0
						1	0	1	1	1	R_2	R_1	R_0
	r, A	r – A	2	8	Borrow	0	1	1	0	0	0	0	0
						0	0	1	1	1	R_2	R_1	R_0
NEA	A, r	A – r	2	8	No zero	0	1	1	0	0	0	0	0
						1	1	1	0	1	R_2	R_1	R_0
	r, A	r – A	2	8	No zero	0	1	1	0	0	0	0	0
						0	1	1	0	1	R_2	R_1	R_0
EQA	A, r	A – r	2	8	Zero	0	1	1	0	0	0	0	0
						1	1	1	1	1	R_2	R_1	R_0
	r, A	r – A	2	8	Zero	0	1	1	0	0	0	0	0
						0	1	1	1	1	R_2	R_1	R_0
ONA	A, r	A ∧ r	2	8	No zero	0	1	1	0	0	0	0	0
						1	1	0	0	1	R_2	R_1	R_0
OFFA	A, r	A ∧ r	2	8	Zero	0	1	1	0	0	0	0	0
						1	1	0	1	1	R_2	R_1	R_0

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
8-Bit Arithmetic (Memory)													
ADDX	rpa	A ← A + (rpa)	2	11		0	1	1	1	0	0	0	0
						1	1	0	0	0	A_2	A_1	A_0
ADCX	rpa	A ← A + (rpa) + CY	2	11		0	1	1	1	0	0	0	0
						1	1	0	1	0	A_2	A_1	A_0
ADDNCX	rpa	A ← A + (rpa)	2	11	No carry	0	1	1	1	0	0	0	0
						1	0	1	0	0	A_2	A_1	A_0
SUBX	rpa	A ← A − (rpa)	2	11		0	1	1	1	0	0	0	0
						1	1	1	0	0	A_2	A_1	A_0
SBBX	rpa	A ← A − (rpa) − CY	2	11		0	1	1	1	0	0	0	0
						1	1	1	1	0	A_2	A_1	A_0
SUBNBX	rpa	A ← A − (rpa)	2	11	No borrow	0	1	1	1	0	0	0	0
						1	0	1	1	0	A_2	A_1	A_0
ANAX	rpa	A ← A ∧ (rpa)	2	11		0	1	1	1	0	0	0	0
						1	0	0	0	1	A_2	A_1	A_0
ORAX	rpa	A ← A ∨ (rpa)	2	11		0	1	1	1	0	0	0	0
						1	0	0	1	1	A_2	A_1	A_0
XRAX	rpa	A ← A ⊻ (rpa)	2	11		0	1	1	1	0	0	0	0
						1	0	0	1	0	A_2	A_1	A_0
GTAX	rpa	A − (rpa) − 1	2	11	No borrow	0	1	1	1	0	0	0	0
						1	0	1	0	1	A_2	A_1	A_0
LTAX	rpa	A − (rpa)	2	11	Borrow	0	1	1	1	0	0	0	0
						1	0	1	1	1	A_2	A_1	A_0
NEAX	rpa	A − (rpa)	2	11	No zero	0	1	1	1	0	0	0	0
						1	1	1	0	1	A_2	A_1	A_0
EQAX	rpa	A − (rpa)	2	11	Zero	0	1	1	1	0	0	0	0
						1	1	1	1	1	A_2	A_1	A_0
ONAX	rpa	A ∧ (rpa)	2	11	No zero	0	1	1	1	0	0	0	0
						1	1	0	0	1	A_2	A_1	A_0
OFFAX	rpa	A ∧ (rpa)	2	11	Zero	0	1	1	1	0	0	0	0
						1	1	0	1	1	A_2	A_1	A_0

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
Immediate Data													
ADI	*A, byte	A ← A + byte	2	7		0	1	0	0	0	1	1	0
						Data							
	r, byte	r ← r + byte	3	11		0	1	1	1	0	1	0	0
						0	1	0	0	0	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 ← sr2 + byte	3	20		0	1	1	0	0	1	0	0
						S_3	1	0	0	0	S_2	S_1	S_0
						Data							
ACI	*A, byte	A ← A + byte + CY	2	7		0	1	0	1	0	1	1	0
						Data							
	r, byte	r ← r + byte + CY	3	11		0	1	1	1	0	1	0	0
						0	1	0	1	0	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 ← sr2 + byte + CY	3	20		0	1	1	0	0	1	0	0
						S_3	1	0	1	0	S_2	S_1	S_0
						Data							
ADINC	*A, byte	A ← A + byte	2	7	No carry	0	0	1	0	0	1	1	0
						Data							
	r, byte	r ← r + byte	3	11	No carry	0	1	1	1	0	1	0	0
						0	0	1	0	0	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 ← sr2 + byte	3	20	No carry	0	1	1	0	0	1	0	0
						S_3	0	1	0	0	S_2	S_1	S_0
						Data							
SUI	*A, byte	A ← A – byte	2	7		0	1	1	0	0	1	1	0
						Data							
	r, byte	r ← r – byte	3	11		0	1	1	1	0	1	0	0
						0	1	1	0	0	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 ← sr2 – byte	3	20		0	1	1	0	0	1	0	0
						S_3	1	1	0	0	S_2	S_1	S_0
						Data							

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
Immediate Data (cont)													
SBI	*A, byte	A ← A – byte – CY	2	7		0	1	1	1	0	1	1	0
						Data							
	r, byte	r ← r – byte – CY	3	11		0	1	1	1	0	1	0	0
						0	1	1	1	0	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 ← sr2 – byte – CY	3	20		0	1	1	0	0	1	0	0
						S_3	1	1	1	0	S_2	S_1	S_0
						Data							
SUINB	*A, byte	A ← A – byte	2	7	No borrow	0	0	1	1	0	1	1	0
						Data							
	r, byte	r ← r – byte	3	11	No borrow	0	1	1	1	0	1	0	0
						0	0	1	1	0	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 ← sr2 – byte	3	20	No borrow	0	1	1	0	0	1	0	0
						S_3	0	1	1	0	S_2	S_1	S_0
						Data							
ANI	*A, byte	A ← A ∧ byte	2	7		0	0	0	0	0	1	1	1
						Data							
	r, byte	r ← r ∧ byte	3	11		0	1	1	1	0	1	0	0
						0	0	0	0	1	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 ← sr2 ∧ byte	3	20		0	1	1	0	0	1	0	0
						S_3	0	0	0	1	S_2	S_1	S_0
						Data							
ORI	*A, byte	A ← A V byte	2	7		0	0	0	1	0	1	1	1
						Data							
	r, byte	r ← r V byte	3	11		0	1	1	1	0	1	0	0
						0	0	0	1	1	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 ← sr2 V byte	3	20		0	1	1	0	0	1	0	0
						S_3	0	0	1	1	S_2	S_1	S_0
						Data							

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
Immediate Data (cont)													
XRI	*A, byte	A ← A ∀ byte	2	7		0	0	0	1	0	1	1	0
						Data							
	r, byte	r ← r ∀ byte	3	11		0	1	1	1	0	1	0	0
						0	0	0	1	0	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 ← sr2 ∀ byte	3	20		0	1	1	0	0	1	0	0
						S_3	0	0	1	0	S_2	S_1	S_0
						Data							
GTI	*A, byte	A – byte – 1	2	7	No borrow	0	0	1	0	0	1	1	1
						Data							
	r, byte	r – byte – 1	3	11	No borrow	0	1	1	1	0	1	0	0
						0	0	1	0	1	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 – byte – 1	3	14	No borrow	0	1	1	0	0	1	0	0
						S_3	0	1	0	1	S_2	S_1	S_0
						Data							
LTI	*A, byte	A – byte	2	7	Borrow	0	0	1	1	0	1	1	1
						Data							
	r, byte	r – byte	3	11	Borrow	0	1	1	1	0	1	0	0
						0	0	1	1	1	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 – byte	3	14	Borrow	0	1	1	0	0	1	0	0
						S_3	0	1	1	1	S_2	S_1	S_0
						Data							
NEI	*A, byte	A – byte	2	7	No zero	0	1	1	0	0	1	1	1
						Data							
	r, byte	r – byte	3	11	No zero	0	1	1	1	0	1	0	0
						0	1	1	0	1	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 – byte	3	14	No zero	0	1	1	0	0	1	0	0
						S_3	1	1	0	1	S_2	S_1	S_0
						Data							

2c

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
Immediate Data (cont)													
EQI	*A, byte	A – byte	2	7	Zero	0	1	1	1	0	1	1	1
						Data							
	r, byte	r – byte	3	11	Zero	0	1	1	1	0	1	0	0
						0	1	1	1	1	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 – byte	3	14	Zero	0	1	1	0	0	1	0	0
						S_3	1	1	1	1	S_2	S_1	S_0
						Data							
ONI	*A, byte	A ∧ byte	2	7	no zero	0	1	0	0	0	1	1	1
						Data							
	r, byte	r ∧ byte	3	11	no zero	0	1	1	1	0	1	0	0
						0	1	0	0	1	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 ∧ byte	3	14	no zero	0	1	1	0	0	1	0	0
						S_3	1	0	0	1	S_2	S_1	S_0
						Data							
OFFI	*A, byte	A ∧ byte	2	7	zero	0	1	0	1	0	1	1	1
						Data							
	r, byte	r ∧ byte	3	11	zero	0	1	1	1	0	1	0	0
						0	1	0	1	1	R_2	R_1	R_0
						Data							
	sr2, byte	sr2 ∧ byte	3	14	zero	0	1	1	0	0	1	0	0
						S_3	1	0	1	1	S_2	S_1	S_0
						Data							
Working Register													
ADDW	wa	A ← A + (V•wa)	3	14		0	1	1	1	0	1	0	0
						1	1	0	0	0	0	0	0
						Offset							
ADCW	wa	A ← A + (V•wa) + CY	3	14		0	1	1	1	0	1	0	0
						1	1	0	1	0	0	0	0
						Offset							
ADDNCW	wa	A ← A + (V•wa)	3	14	No carry	0	1	1	1	0	1	0	0
						1	0	1	0	0	0	0	0
						Offset							
SUBW	wa	A ← A – (V•wa)	3	14		0	1	1	1	0	1	0	0
						1	1	1	0	0	0	0	0
						Offset							

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
Working Register (cont)													
SBBW	wa	A ← A – (V•wa) – CY	3	14		0	1	1	1	0	1	0	0
						1	1	1	1	0	0	0	0
						Offset							
SUBNBW	wa	A ← A – (V•wa)	3	14	No borrow	0	1	1	1	0	1	0	0
						1	0	1	1	0	0	0	0
						Offset							
ANAW	wa	A ← A ∧ (V•wa)	3	14		0	1	1	1	0	1	0	0
						1	0	0	0	1	0	0	0
						Offset							
ORAW	wa	A ← A ∨ (V•wa)	3	14		0	1	1	1	0	1	0	0
						1	0	0	1	1	0	0	0
						Offset							
XRAW	wa	A ← A ∀ (V•wa)	3	14		0	1	1	1	0	1	0	0
						1	0	0	1	0	0	0	0
						Offset							
GTAW	wa	A – (V•wa) – 1	3	14	No borrow	0	1	1	1	0	1	0	0
						1	0	1	0	1	0	0	0
						Offset							
LTAW	wa	A – (V•wa)	3	14	Borrow	0	1	1	1	0	1	0	0
						1	0	1	1	1	0	0	0
						Offset							
NEAW	wa	A – (V•wa)	3	14	No Zero	0	1	1	1	0	1	0	0
						1	1	1	0	1	0	0	0
						Offset							
EQAW	wa	A – (V•wa)	3	14	Zero	0	1	1	1	*	1	0	0
						1	1	1	1	1	0	0	0
						Offset							
ONAW	wa	A ∧ (V•wa)	3	14	No zero	0	1	1	1	0	1	0	0
						1	1	0	0	1	0	0	0
						Offset							
OFFAW	wa	A ∧ (V•wa)	3	14	Zero	0	1	1	1	0	1	0	0
						1	1	0	1	1	0	0	0
						Offset							
ANIW	*wa, byte	(V•wa) ← (V•wa) ∧ byte	3	19		0	0	0	0	0	1	0	1
						Offset							
						Data							
ORIW	*wa, byte	(V•wa) ← (V•wa) ∨ byte	3	19		0	0	0	1	0	1	0	1
						Offset							
						Data							

2c

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
Working Register (cont)													
GTIW	*wa, byte	(V•wa) – byte – 1	3	13	No borrow	0	0	1	0	0	1	0	1
						Offset							
						Data							
LTIW	*wa, byte	(V•wa) – byte	3	13	Borrow	0	0	1	1	0	1	0	1
						Offset							
						Data							
NEIW	*wa, byte	(V•wa) – byte	3	13	No zero	0	1	1	0	0	1	0	1
						Offset							
						Data							
EQIW	*wa, byte	(V•wa) – byte	3	13	Zero	0	1	1	1	0	1	0	1
						Offset							
						Data							
ONIW	*wa, byte	(V•wa) ∧ byte	3	13	No zero	0	1	0	0	0	1	0	1
						Offset							
						Data							
OFFIW	*wa, byte	(V•wa) ∧ byte	3	13	Zero	0	1	0	1	0	1	0	1
						Offset							
						Data							
16-Bit Arithmetic													
EADD	EA, r2	EA ← EA + r2	2	11		0	1	1	1	0	0	0	0
						0	1	0	0	0	0	R_1	R_0
DADD	EA, rp3	EA ← EA + rp3	2	11		0	1	1	1	0	1	0	0
						1	1	0	0	0	1	P_1	P_0
DADC	EA, rp3	EA ← EA + rp3 + CY	2	11		0	1	1	1	0	1	0	0
						1	1	0	1	0	1	P_1	P_0
DADDNC	EA, rp3	EA ← EA + rp3	2	11	No carry	0	1	1	1	0	1	0	0
						1	0	1	0	0	1	P_1	P_0
ESUB	EA, r2	EA ← EA – r2	2	11		0	1	1	1	0	0	0	0
						0	1	1	0	0	0	R_1	R_0
DSUB	EA, rp3	EA ← EA – rp3	2	11		0	1	1	1	0	1	0	0
						1	1	1	0	0	1	P_1	P_0
DSBB	EA, rp3	EA ← EA – rp3 – CY	2	11		0	1	1	1	0	1	0	0
						1	1	1	1	0	1	P_1	P_0
DSUBNB	EA, rp3	EA ← EA – rp3	2	11	No borrow	0	1	1	1	0	1	0	0
						1	0	1	1	0	1	P_1	P_0
DAN	EA, rp3	EA ← EA ∧ rp3	2	11		0	1	1	1	0	1	0	0
						1	0	0	0	1	1	P_1	P_0
DOR	EA, rp3	EA ← EA ∨ rp3	2	11		0	1	1	1	0	1	0	0
						1	0	0	1	1	1	P_1	P_0

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
16-Bit Arithmetic (cont)													
DXR	EA, rp3	EA ← EA ∀ rp3	2	11		0	1	1	1	0	1	0	0
						1	0	0	1	0	1	P_1	P_0
DGT	EA, rp3	EA – rp3 – 1	2	11	No borrow	0	1	1	1	0	1	0	0
						1	0	1	0	1	1	P_1	P_0
DLT	EA, rp3	EA – rp3	2	11	Borrow	0	1	1	1	0	1	0	0
						1	0	1	1	1	1	P_1	P_0
DNE	EA, rp3	EA – rp3	2	11	No zero	0	1	1	1	0	1	0	0
						1	1	1	0	1	1	P_1	P_0
DEQ	EA, rp3	EA – rp3	2	11	Zero	0	1	1	1	0	1	0	0
						1	1	1	1	1	1	P_1	P_0
DON	EA, rp3	EA ∧ rp3	2	11	No zero	0	1	1	1	0	1	0	0
						1	1	0	0	1	1	P_1	P_0
DOFF	EA, rp3	EA ∧ rp3	2	11	Zero	0	1	1	1	0	1	0	0
						1	1	0	1	1	1	P_1	P_0
Multiply/Divide													
MUL	r2	EA ← A x r2	2	32		0	1	0	0	1	0	0	0
						0	0	1	0	1	1	R_1	R_0
DIV	r2	EA ← EA ÷ r2, r2 ← Remainder	2	59		0	1	0	0	1	0	0	0
						0	0	1	1	1	1	R_1	R_0
Increment/Decrement													
INR	r2	r2 ← r2 + 1	1	4	Carry	0	1	0	0	0	0	R_1	R_0
INRW	*wa	(V•wa) ← (V•wa) + 1	2	16	Carry	0	0	1	0	0	0	0	0
										Offset			
INX	rp	rp ← rp + 1	1	7		0	0	P_1	P_0	0	0	1	0
	EA	EA ← EA + 1	1	7		1	0	1	0	1	0	0	0
DCR	r2	r2 ← r2 – 1	1	4	Borrow	0	1	0	1	0	0	R_1	R_0
DCRW	*wa	(V•wa) ← (V•wa) – 1	2	16	Borrow	0	0	1	1	0	0	0	0
										Offset			
DCX	rp	rp ← rp – 1	1	7		0	0	P_1	P_0	0	0	1	1
	EA	EA ← EA – 1	1	7		1	0	1	0	1	0	0	1
Others													
DAA		Decimal Adjust Accumulator	1	4		0	1	1	0	0	0	0	1
STC		CY ← 1	2	8		0	1	0	0	1	0	0	0
						0	0	1	0	1	0	1	1
CLC		CY ← 0	2	8		0	1	0	0	1	0	0	0
						0	0	1	0	1	0	1	0
NEGA		A ← $\overline{A}$ + 1	2	8		0	1	0	0	1	0	0	0
						0	0	1	1	1	0	1	0

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
Rotate and Shift													
RLD		Rotate left digit $A_{3-0} \leftarrow (HL)_{7-4}$, $(HL)_{7-4} \leftarrow (HL)_{3-0}$, $(HL)_{3-0} \leftarrow A_{3-0}$	2	17		0	1	0	0	1	0	0	0
						0	0	1	1	1	0	0	0
RRD		Rotate right digit $(HL)_{7-4} \leftarrow A_{3-0}$, $(HL)_{3-0} \leftarrow (HL)_{7-4}$, $A_{3-0} \leftarrow (HL)_{3-0}$	2	17		0	1	0	0	1	0	0	0
						0	0	1	1	1	0	0	1
RLL	r2	$r2_{m+1} \leftarrow r2_m$, $r2_0 \leftarrow CY$, $CY \leftarrow r2_7$	2	8		0	1	0	0	1	0	0	0
						0	0	1	1	0	1	R_1	R_0
RLR	r2	$r2_{m-1} \leftarrow r2_m$, $r2_7 \leftarrow CY$, $CY \leftarrow r2_0$	2	8		0	1	0	0	1	0	0	0
						0	0	1	1	0	0	R_1	R_0
SLL	r2	$r2_{m+1} \leftarrow r2_m$, $r2_0 \leftarrow 0$, $CY \leftarrow r2_7$	2	8		0	1	0	0	1	0	0	0
						0	0	1	0	0	1	R_1	R_0
SLR	r2	$r2_{m-1} \leftarrow r2_m$, $r2_7 \leftarrow 0$, $CY \leftarrow r2_0$	2	8		0	1	0	0	1	0	0	0
						0	0	1	0	0	0	R_1	R_0
SLLC	r2	$r2_{m+1} \leftarrow r2_m$, $r2_0 \leftarrow 0$, $CY \leftarrow r2_7$	2	8	Carry	0	1	0	0	1	0	0	0
						0	0	0	0	0	1	R_1	R_0
SLRC	r2	$r2_{m-1} \leftarrow r2_m$, $r2_7 \leftarrow 0$, $CY \leftarrow r2_0$	2	8	Carry	0	1	0	0	1	0	0	0
						0	0	0	0	0	0	R_1	R_0
DRLL	EA	$EA_{n+1} \leftarrow EA_n$, $EA_0 \leftarrow CY$, $CY \leftarrow EA_{15}$	2	8		0	1	0	0	1	0	0	0
						1	0	1	1	0	1	0	0
DRLR	EA	$EA_{n-1} \leftarrow EA_n$, $EA_{15} \leftarrow CY$, $CY \leftarrow EA_0$	2	8		0	1	0	0	1	0	0	0
						1	0	1	1	0	0	0	0
DSLL	EA	$EA_{n+1} \leftarrow EA_n$, $EA_0 \leftarrow 0$, $CY \leftarrow EA_{15}$	2	8		0	1	0	0	1	0	0	0
						1	0	1	0	0	1	0	0
DSLR	EA	$EA_{n-1} \leftarrow EA_n$, $EA_{15} \leftarrow 0$, $CY \leftarrow EA_0$	2	8		0	1	0	0	1	0	0	0
						1	0	1	0	0	0	0	0
Jump													
JMP	*word	PC ← word	3	10		0	1	0	1	0	1	0	0
						Low addr							
						High addr							
JB		$PC_H \leftarrow B$, $PC_L \leftarrow C$	1	4		0	0	1	0	0	0	0	1
JR	word	PC ← PC + 1 + jdisp1	1	10		1	1	jdisp1					
JRE	*word	PC ← PC + 2 + jdisp	2	10		0	1	0	0	1	1	1	
						jdisp							
JEA		PC ← EA	2	8		0	1	0	0	1	0	0	0
						0	0	1	0	1	0	0	0

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
Call													
CALL	*word	$(SP-1) \leftarrow (PC+3)_H$, $(SP-2) \leftarrow (PC+3)_L$, $PC \leftarrow word$, $SP \leftarrow SP-2$	3	16		0	1	0	0	0	0	0	0
						Low Addr							
						High Addr							
CALB		$(SP-1) \leftarrow (PC+2)_H$, $(SP-2) \leftarrow (PC+2)_L$, $PC_H \leftarrow B$, $PC_L \leftarrow C$, $SP \leftarrow SP-2$	2	17		0	1	0	0	1	0	0	0
						0	0	1	0	1	0	0	1
CALF	*word	$(SP-1) \leftarrow (PC+2)_H$, $(SP-2) \leftarrow (PC+2)_L$, $PC_{15-11} \leftarrow 00001$, $PC_{10-0} \leftarrow fa$, $SP \leftarrow SP-2$	2	13		0	1	1	1	1		fa_H	
										fa_L			
CALT	word	$(SP-1) \leftarrow (PC+1)_H$, $(SP-2) \leftarrow (PC+1)_L$, $PC_L \leftarrow (128+2ta)$, $PC_H \leftarrow (129+2ta)$, $SP \leftarrow SP-2$	1	16		1	0	0			ta		
SOFTI	word	$(SP-1) \leftarrow PSW$, $(SP-2) \leftarrow (PC+1)_H$, $(SP-3) \leftarrow (PC+1)_L$, $PC \leftarrow 0060H$, $SP \leftarrow SP-3$	1	16		0	1	1	1	0	0	1	0
Return													
RET		$PC_L \leftarrow (SP)$, $PC_H \leftarrow (SP+1)$, $SP \leftarrow SP+2$	1	10		1	0	1	1	1	0	0	0
RETS		$PC_L \leftarrow (SP)$, $PC_H \leftarrow (SP+1)$, $SP \leftarrow SP+2$, $PC \leftarrow PC+n$	1	10	Unconditional Skip	1	0	1	1	1	0	0	1
RETI		$PC_L \leftarrow (SP)$, $PC_H \leftarrow (SP+1)$, $PSW \leftarrow SP+2$, $SP \leftarrow SP+3$	1	13		0	1	1	0	0	0	1	0
Skip													
BIT	*bit, wa	Skip if (V•wa) bit = 1	2	10	Bit Test	0	1	0	1	1	B_2	B_1	B_0
						Offset							
SK	f	Skip if f = 1	2	8	f = 1	0	1	0	0	1	0	0	0
						0	0	0	0	1	F_2	F_1	F_0
SKN	f	Skip if f = 0	2	8	f = 0	0	1	0	0	1	0	0	0
						0	0	0	1	1	F_2	F_1	F_0
SKIT	irf	Skip if irf = 1, then reset irf	2	8	irf = 1	0	1	0	0	1	0	0	0
						0	1	0	I_4	I_3	I_2	I_1	I_0
SKNIT	irf	Skip if irf = 0; Reset irf if irf = 1 and don't skip	2	8	irf = 0	0	1	0	0	1	0	0	0
						0	1	1	I_4	I_3	I_2	I_1	I_0

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States (Note 1)	Skip Conditions	Operation Code 7	6	5	4	3	2	1	0
CPU Control													
NOP		No operation	1	4		0	0	0	0	0	0	0	0
EI		Enable interrupt	1	4		1	0	1	0	1	0	1	0
DI		Disable interrupt	1	4		1	0	1	1	1	0	1	0
HLT		Set HALT mode	2	12		0	1	0	0	1	0	0	0
						0	0	1	1	1	0	1	1
STOP		Set STOP mode	2	12		0	1	0	0	1	0	0	0
						1	0	1	1	1	0	1	1

Notes:

(1) For the skip condition, the idle states are as follows:
1-byte instructions: 4 states
2-byte instructions: 8 states
3-byte instructions: 11 states
2-byte instructions with*: 7 states
3-byte instructions with*: 10 states
4-byte instructions: 14 states

(2) B2 (Data): rpa2 = D+ byte or H+ byte.

(3) Right side of slash (/) in states indicates the case when rpa2 or rpa3 = D+ byte, H+A, H+B, H+EA, or H+ byte.

(4) B3 (Data): rpa3 = D+ byte or H+ byte.

NEC

Section 3
μPD78K0 Product Line
8-Bit, K-Series Microcontrollers

NEC
NEC Electronics Inc.

μPD78002 Family (μPD78001B/002B/P014) 8-Bit, K-Series Microcontrollers General Purpose

September 1993

Description

The μPD78001B, μPD78002B, and μPD78P014* are members of the K-Series® of microcontrollers. The μPD78P014 is used for prototyping since the μPD78014 family is pin and function compatible with, and the features are a superset of, the μPD78002 family. The features of the μPD78002 family include bit manipulation instructions, four banks of main registers, an advanced interrupt handling facility, and a powerful set of memory-mapped on-chip peripherals. On-board data memory includes 256, 384, or 1024 bytes of internal high-speed RAM. Program memory options include 8K or 16K bytes of mask ROM, or 32K bytes of internal UV EPROM or one-time programmable (OTP) ROM.

The μPD78002 family operates over a wide voltage range: 2.7 to 6.0 volts. Timing is generated by two built-in oscillators. A main oscillator normally drives the CPU and most peripherals and at 10 MHz provides a minimum instruction time of 0.4 μs. A subsystem oscillator at 32.768 kHz provides time keeping, and optionally a slow clock for the CPU. Since CMOS power dissipation is directly proportional to clock rate, the μPD78002 family provides a software variable CPU clock. The HALT and STOP modes are two additional power saving features that turn off parts of the microcontroller to reduce power consumption. A data retention mode permits RAM contents to be saved down to 2 volts.

The range of peripherals, including timers and a serial port, makes these devices ideal for applications in portable battery-powered equipment, office automation, communications, consumer electronics, home appliances, and fitness equipment.

Features

- One-channel serial communication interface
  - 8-bit clock synchronous interface 0
  - Full-duplex, three-wire mode
  - NEC serial bus interface (SBI) mode
  - Half-duplex, two-wire mode
- Timers
  - Watchdog timer
  - Two 8-bit timer/event counters usable as one 16-bit timer/event counter
  - Clock (watch) timer (time of day tick from either oscillator)
- 53 I/O lines
  - Two CMOS input-only lines
  - 47 CMOS I/O lines
  - Four n-channel, open-drain I/O lines at 15 V maximum
- I/O port pullup resistors
  - Software controllable on 47 lines
  - Mask option on four lines on ROM versions
- Program memory
  - μPD78001B: 8K bytes ROM
  - μPD78002B: 16K bytes ROM
  - μPD78P014: 32K bytes EPROM/OTP
- Internal high-speed data memory
  - μPD78001B: 256 bytes RAM
  - μPD78002B: 384 bytes RAM
  - μPD78P014: 1024 bytes RAM
- External memory expansion
  - 64K byte total memory space
- Powerful instruction set
  - 16-bit arithmetic and data transfer instructions
  - 1-bit and 8-bit logic instructions
- Minimum instruction execution times:
  - 0.4/0.8/1.6/3.2/6.4 μs program selectable using 10-MHz main system clock
  - 122 μs selectable using 32.768-kHz subsystem clock
- Memory-mapped on-chip peripherals (special function registers)
- Programmable priority, vectored-interrupt controller (two levels)
- Buzzer and clock outputs
- Power saving and battery operation features
  - Variable CPU clock rate
  - HALT mode
  - STOP mode
  - 2-V data retention mode
- CMOS operation; V_{DD} from 2.7 to 6.0 V

3a

* See the μPD78014 family data sheet for the μPD78P014 electrical and functional specifications.

K-Series is a registered trademark of NEC Electronics, Inc.

Ordering Information

Part Number	ROM	Package (Package Dwg.)
μPD78001BCW-xxx	8K mask ROM	64-pin plastic shrink DIP (P64C-70-750 A, C)
μPD78002BCW-xxx	16K mask ROM	
μPD78001BGC-xxx-AB8	8K mask ROM	64-pin plastic QFP (P64GC-80-AB8-2)
μPD78002BGC-xxx-AB8	16K mask ROM	
μPD78P014GC-AB8 (Note 3)	32K OTP ROM	
μPD78P014DW (Note 3)	32K UV EPROM	64-pin ceramic shrink DIP w/window (P64DW-70-750A)

Notes:

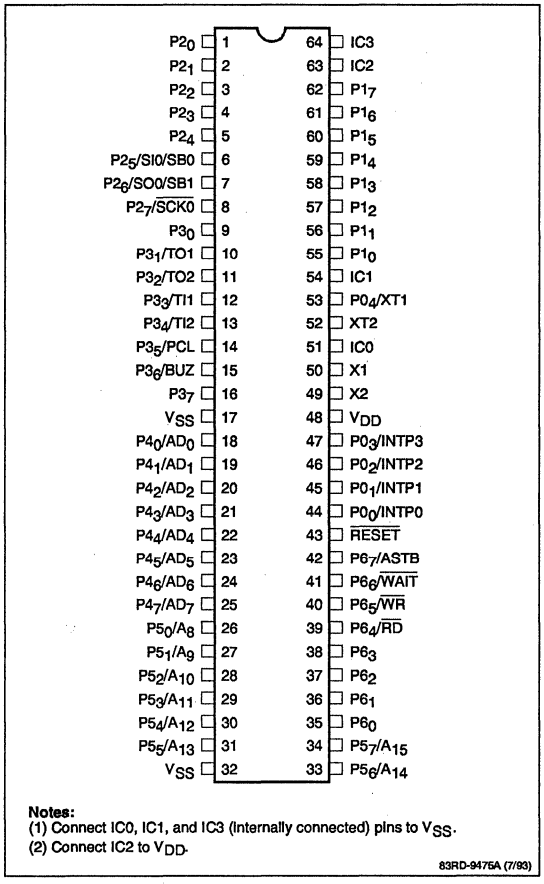
(1) xxx indicates ROM code suffix

(2) All devices listed are standard quality grade

(3) See the μPD78014 family data sheet for the μPD78P014 electrical and functional specifications

Pin Configurations

64-Pin Plastic Shrink DIP

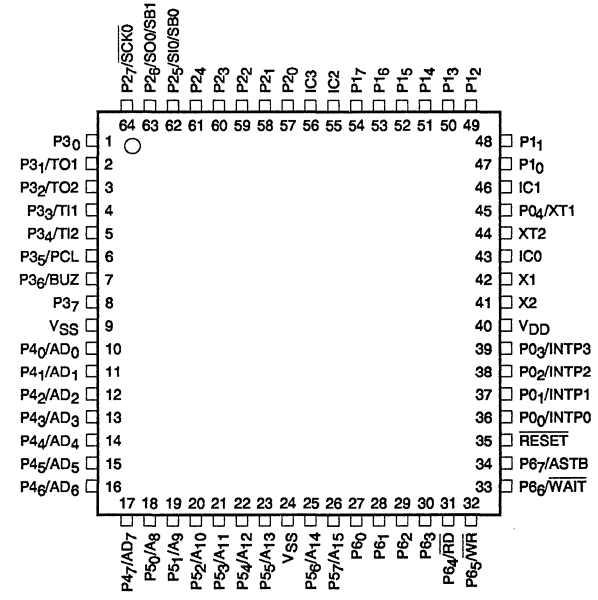


Notes:
(1) Connect IC0, IC1, and IC3 (internally connected) pins to V_{SS}.
(2) Connect IC2 to V_{DD}.

83RD-9476A (7/93)

Pin Configurations (cont)

64-Pin Plastic QFP



Notes:
(1) Connect IC0, IC1, and IC3 (internally connected) pins to V_{SS}.
(2) Connect IC2 to V_{DD}.

83RD-9476B

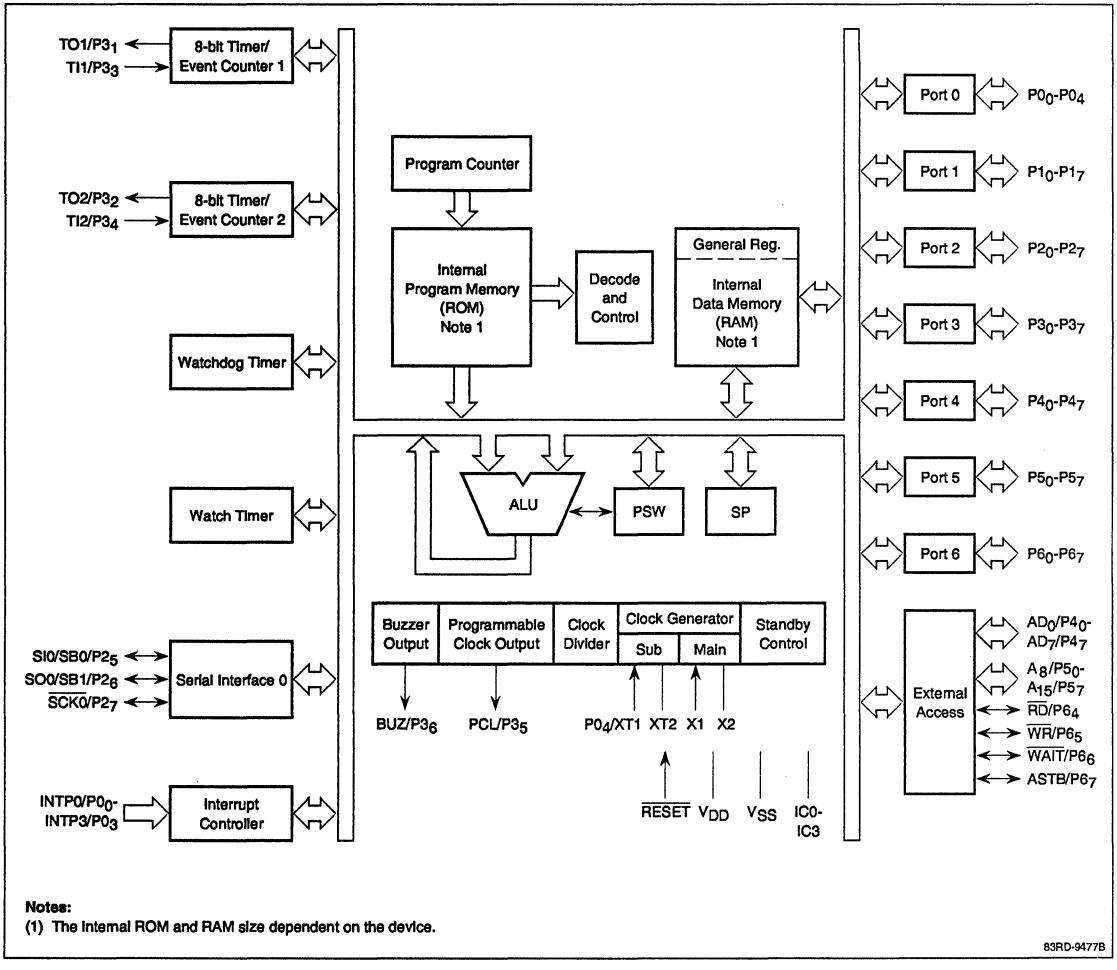
3a

Pin Functions; Normal Operating Mode

Symbol	First Function	Symbol	Alternate Function
$P0_0$ $P0_1$ $P0_2$ $P0_3$	Port 0; 5-bit, bit selectable I/O port (Bits 0 and 4 are input only)	INTP0 INTP1 INTP2 INTP3	External maskable interrupt
$P0_4$		XT1	Crystal oscillator or external clock input for subsystem clock
$P1_0$ - $P1_7$	Port 1; 8-bit, bit-selectable I/O port	—	
$P2_0$ - $P2_4$	Port 2; 8-bit, bit-selectable I/O port	—	
$P2_5$		SIO SB0	Serial data input 3-wire serial I/O mode 2/3-wire serial I/O mode
$P2_6$		SO0 SB1	Serial data output 3-wire serial I/O mode 2/3-wire serial I/O mode
$P2_7$		$\overline{SCK0}$	Serial clock I/O for serial interface 0
$P3_0$	Port 3; 8-bit, bit-selectable I/O port	—	
$P3_1$		TO1	Timer output from timer 1
$P3_2$		TO2	Timer output from timer 2
$P3_3$		TI1	External count clock input to timer 1
$P3_4$		TI2	External count clock input to timer 2
$P3_5$		PCL	Programmable clock output
$P3_6$		BUZ	Programmable buzzer output
$P3_7$		—	
$P4_0$ - $P4_7$	Port 4; 8-bit I/O port	AD_0 - AD_7	Low-order 8-bit multiplexed address/data bus for external memory
$P5_0$ - $P5_7$	Port 5; 8-bit, bit selectable I/O port	A_8 - A_{15}	High-order 8-bit address bus for external memory
$P6_0$ - $P6_3$	Port 6; 8-bit, bit selectable ($P6_0$ to $P6_3$ n-channel open-drain I/O with mask option pullup resistors; $P6_4$ - $P6_7$ I/O). See note.	—	
$P6_4$		$\overline{RD}$	External memory read strobe
$P6_5$		$\overline{WR}$	External memory write strobe
$P6_6$		$\overline{WAIT}$	External memory wait signal input
$P6_7$		ASTB	Address strobe used to latch address for external memory
$\overline{RESET}$	External system reset input		
X1	Crystal/ceramic resonator connection or external clock input for main system clock		
X2	Crystal/ceramic resonator connection or inverse of external clock for main system clock		
XT2	Crystal oscillator or left open when using external clock for subsystem clock		
V_{DD}	Power supply input		
V_{SS}	Power supply ground		
IC0 to IC3	Internal connection		

Note: See table 3 and figure 4 for details.

Block Diagram



Notes:
(1) The internal ROM and RAM size dependent on the device.

83RD-9477B

3a

μPD78002 and μPD78014 Family Differences

The μPD78002 family is pin compatible with the μPD78014 family and shares the same programmable device, the μPD78P014. The μPD78002 family offers a reduced set of features. Table 1 lists only the differences between the two families. All other features not listed are identical for both.

Table 1. Differences Between μPD78002 and μPD78014 Families

Item	μPD78002 Family	μPD78014 Family
Maximum internal ROM	16K bytes	32K bytes
Maximum internal high-speed RAM	384 bytes	1024 bytes
Buffer RAM	None	32 bytes
Multiply/divide instructions	None	Available
16-bit timer/event counter	None	One
Serial interface 1 (3-wire and 3-wire with automatic transmit/receive)	None	One
Vectored internal interrupts	7	10
A/D converter	None	8-bit, 8 channels

FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processing unit (CPU) of the μPD78002 family features 8- and 16-bit arithmetic.

A CALLT vector table and a CALLF area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction can access up to 32 subroutines through their addresses contained in the CALLT vector table (40H to 7FH). A 2-byte call instruction can access any routine beginning in a specific CALLF area (0800H to 0FFFH).

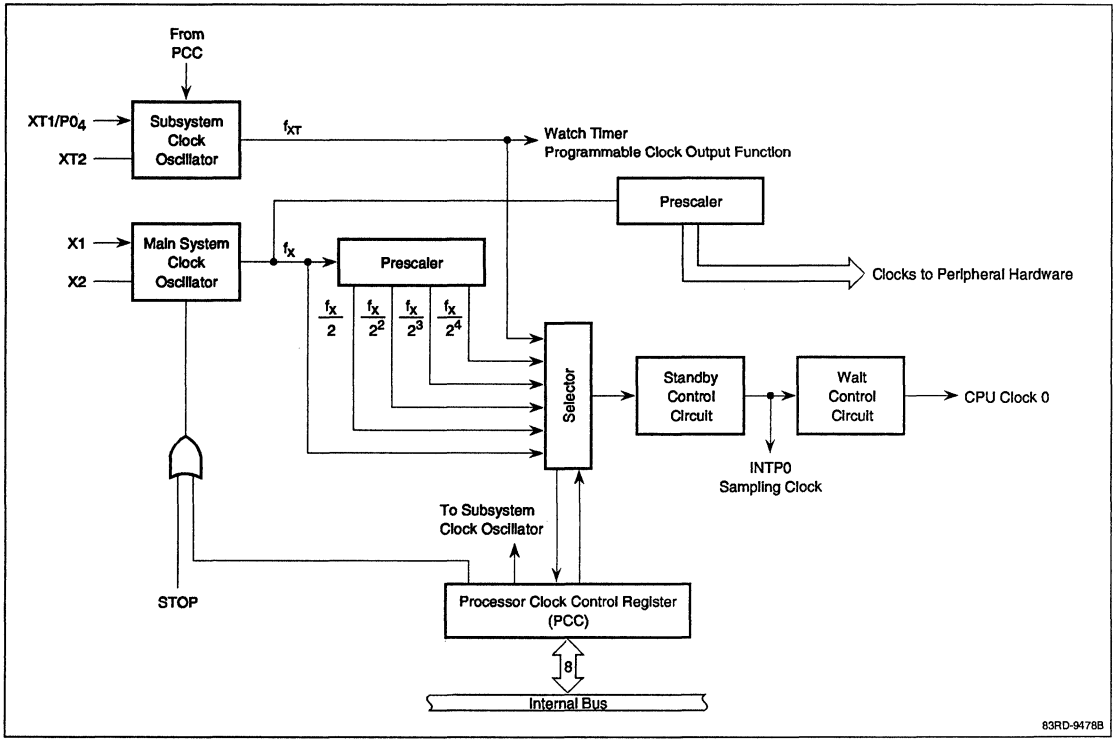
Internal System Clock Generator

The internal system clocks of the μPD78002 family are derived from either the main system or the subsystem oscillator. See figure 1. The clocks for the watch timer and programmable clock output are derived from either the subsystem clock (f_{XT}) or the main system clock. The clocks for all other peripheral hardware are derived from the main system clock.

The CPU clock (ϕ) can be supplied from either the main system clock (f_X) or the subsystem clock (f_{XT}). Using the processor clock control register (PCC), a CPU clock frequency equal to f_X, $f_X/2$, $f_X/4$, $f_X/8$, $f_X/16$ or the subsystem clock f_{XT} can be selected. The CPU clock selected should be based on the power supply voltage available and the desired power consumption. On power up, the CPU clock defaults to the lowest speed from the main system clock and can be changed while the microcontroller is running.

Since the shortest instruction takes four CPU clocks to execute, the fastest minimum instruction execution time (t_{CY}) of 0.4 μs is achieved when using a main system clock at 10 MHz (V_{DD} equals 4.5 to 6.0 V). However, if the clock timer must generate an interrupt every 0.5 or 0.25 seconds, t_{CY} is 0.48 μs at 8.38 MHz. The fastest minimum instruction execution time available across the full voltage range of 2.7 to 6.0 V is 0.96 μs when using a main system clock of 8.38 MHz. For the lowest power consumption, the CPU can be operated from the subsystem clock and the minimum instruction execution time is 122 μs at 32.768 kHz.

Figure 1. Internal System Clock Generator

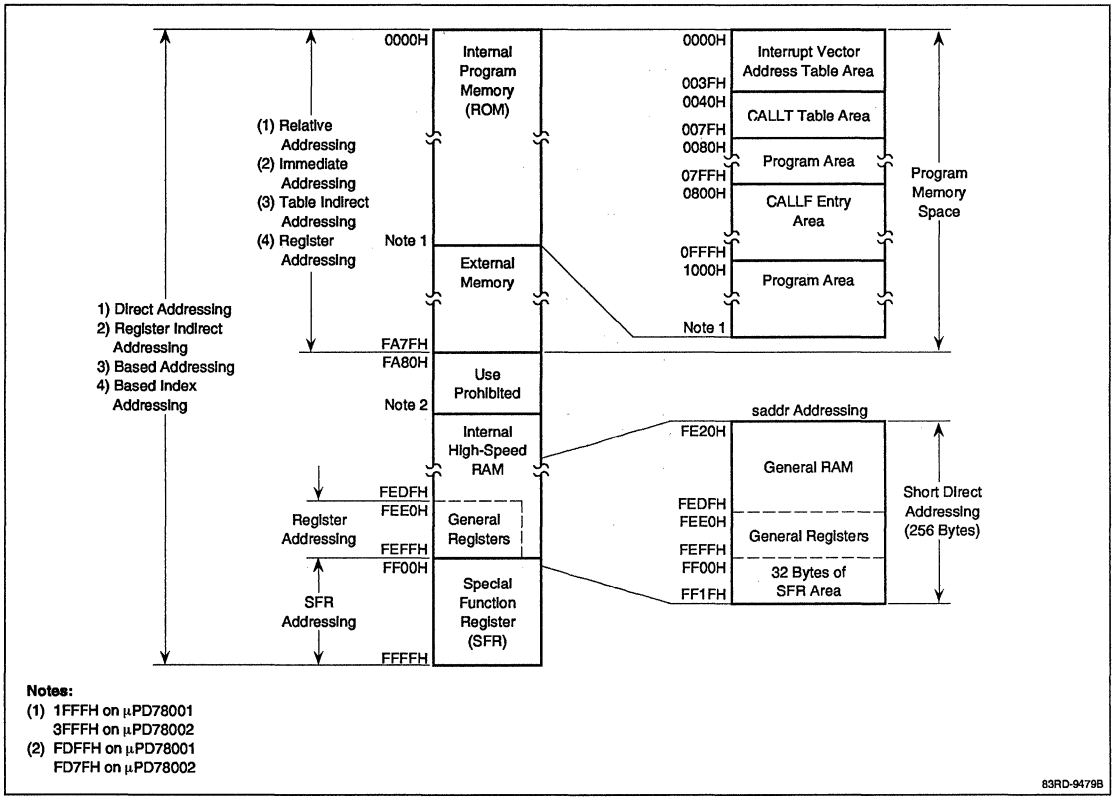


3a

Memory Space

The μPD78002 family has a 64K-byte address space. Some of this address space (0000H-FFFFH) can be used as both program and data memory as shown in figure 2.

Figure 2. Memory Map



Notes:
(1) 1FFFH on μPD78001
3FFFH on μPD78002
(2) FDFFH on μPD78001
FD7FH on μPD78002

83RD-9479B

Internal Program Memory

All devices in the μPD78002 family have internal program memory. The μPD78001B contains 8K bytes while the μPD78002B contains 16K bytes of internal ROM. The μPD78P014 contains 32K bytes of UV EPROM or one time programmable ROM. To allow the μPD78P014 to emulate the mask ROM devices, the amount of internal program memory available in the μPD78P014 can be selected using the memory size switching register (IMS).

Internal RAM

The μPD78001B contains 256 bytes (FE00H to FEFFH) while the μPD78002B contains 384 bytes (FD80H to FEFFH) of high-speed Internal RAM. The high-speed Internal RAM contains the general register banks and the stack. The remainder of the high-speed Internal RAM and any unused register bank locations are available for general storage.

To allow the μPD78P014 to emulate the mask ROM devices, the amount of high-speed Internal RAM in the μPD78P014 can also be selected using the IMS.

External Memory

The μPD78002 family can access 0, 256, 4K, 16K or all available bytes of external memory. The μPD78002 family has an 8-bit wide external data bus and a 16-bit wide external address bus. The low-order 8 bits of the address bus are multiplexed and also provide the 8-bit data bus and are supplied by port 4. The high-order address bits of the 16-bit address bus are taken from port 5 as required. The address latch, read, and write strobes, and the external $\overline{WAIT}$ signal are supplied by port 6.

The memory expansion mode register (MM) controls the size of external memory. It can be programmed to use 0, 4, 6, or 8 bits from port 5 for the high-order address. Any remaining port 5 bits can be used for I/O. The MM register also can be used to specify one additional wait state or the use of the external $\overline{WAIT}$ signal for low-speed external memory or external peripheral devices.

When only internal ROM and RAM are used and no external memory is required, ports 4, 5 and 6 are available as general purpose I/O ports.

CPU Control Registers

Program Counter. The program counter is a 16-bit binary counter register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 0001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

Program Status Word. The program status word (PSW) is an 8-bit register that contains flags that are set or reset depending on the results obtained during the execution of an instruction. This register can be written to or read from 1 bit or 8 bits at a time. The assignment of PSW bits follows.

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

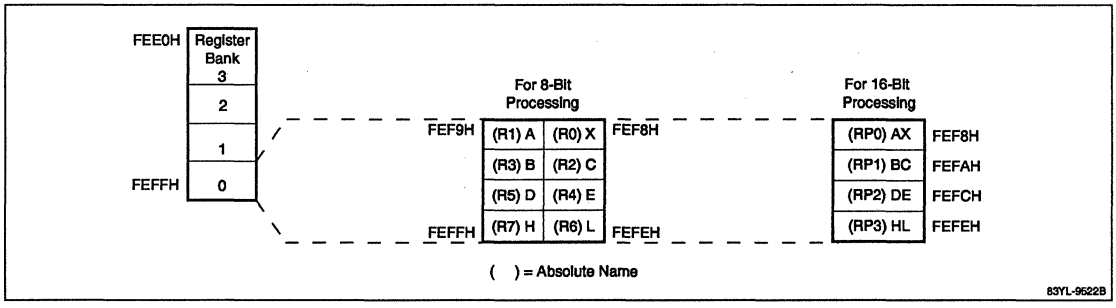
CY	Carry flag
ISP	In-service (interrupt) priority flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

General Registers

The general-purpose registers (figure 3) consist of four banks of registers located at addresses FEE0H to FEFFH in Internal RAM. Each bank consists of eight 8-bit general registers that can also be used in pairs to function as four 16-bit registers. Two bits in the PSW (RBS0 and RSB1) specify which of the register banks is active at any time and are set under program control.

Registers have both functional names A, X, B, C, D, E, H or L for 8-bit registers and AX, BC, DE, and HL for 16-bit registers) and absolute names (like R1, R0, R3, R2, R5, R4, R7, or R6 for 8-bit registers and RP0, RP1, RP2 or RP3 for 16- bit registers). Either the functional or absolute register names can be used in instructions that use the operand identifiers r and rp.

Figure 3. General Registers



Addressing

The program memory addressing (ROM) modes provided are relative, immediate, table indirect, and register addressing. The operand addressing modes provided are implied, register, direct, short direct (saddr), special function (SFR), register indirect, based, based indexed, and stack addressing.

The 'SFR addressing' and 'saddr addressing' modes use direct addressing and require only 1 byte in the instruction to address RAM. Normally a 65K byte address space requires 2 bytes to address it. One-byte addressing results in faster access times, since the instructions are shorter. SFR addressing addresses the entire 256-byte SFR address space from FF00H to FFFFH. Saddr addressing (see figure 2) addresses the 256-byte address space FE20H to FF1FH. FE20H to FEFFH are composed of 224 bytes of internal high speed RAM; FF00H to FF1FH contain the first 32 bytes in the special function register area.

One-byte addressing is accomplished by using the first byte of the instruction for the opcode (and one operand if register A or AX is used) and the second byte of the instruction as an address (offset) into the 256-byte area. If register A or AX is used, the instructions are 2 bytes long, thereby providing fast access times. If immediate data is used, the instruction will be 3 or 4 bytes long depending upon whether the immediate data is a byte or a word. Many 16-bit SFRs are in the space FF00H to FF1FH. Using AX as an operand to these SFRs will provide fast access, since the instructions will be only 2 bytes long.

Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by SFR addressing. FF00H to FF1FH can also be accessed using saddr addressing. They are either 8 or 16 bits as required, and many of the 8-bit registers are bit addressable.

Locations FFD0H through FFDFH are known as the external access SFR area. Registers in external circuitry interfaced and mapped to these addresses can only be addressed by main memory addressing. Table 2 lists the special function registers.

Table 2. Special Function Registers

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF00H	Port 0	P0	R/W	x	x	—	00H
FF01H	Port 1	P1	R/W	x	x	—	00H
FF02H	Port 2	P2	R/W	x	x	—	00H
FF03H	Port 3	P3	R/W	x	x	—	00H
FF04H	Port 4	P4	R/W	x	x	—	Undefined
FF05H	Port 5	P5	R/W	x	x	—	Undefined
FF06H	Port 6	P6	R/W	x	x	—	Undefined
FF16H	Compare register 10	CR10	R/W	—	x	—	Undefined
FF17H	Compare register 20	CR20	R/W	—	x	—	Undefined
FF18H	8-bit timer register 1	TM1	R	x	x	—	00H
FF19H	8-bit timer register 2	TM2	R	x	x	—	00H
FF18H-FF19H	16-bit timer register	TMS	R	—	—	x	0000H
FF1AH	Serial I/O shift register 0	SIO0	R/W	—	x	—	Undefined
FF20H	Port mode register 0	PM0	R/W	x	x	—	1FH
FF21H	Port mode register 1	PM1	R/W	x	x	—	FFH
FF22H	Port mode register 2	PM2	R/W	x	x	—	FFH
FF23H	Port mode register 3	PM3	R/W	x	x	—	FFH
FF25H	Port mode register 5	PM5	R/W	x	x	—	FFH
FF26H	Port mode register 6	PM6	R/W	x	x	—	FFH
FF40H	Timer clock select register 0	TCL0	R/W	x	x	—	00H
FF41H	Timer clock select register 1	TCL1	R/W	—	x	—	00H
FF42H	Timer clock select register 2	TCL2	R/W	—	x	—	00H
FF43H	Timer clock select register 3	TCL3	R/W	—	x	—	88H
FF47H	Sampling clock select register	SCS	R/W	—	x	—	00H
FF49H	8-bit timer mode control register	TMC1	R/W	x	x	—	00H
FF4AH	Watch timer mode control register	TMC2	R/W	x	x	—	00H
FF4FH	8-bit timer output control register	TOC1	R/W	x	x	—	00H
FF60H	Serial operating mode register 0	CSIM0	R/W	x	x	—	00H
FF61H	Serial bus interface control register	SBIC	R/W	x	x	—	00H
FF62H	Slave address register	SVA	R/W	—	x	—	Undefined
FF63H	Interrupt timing specify register	SINT	R/W	x	x	—	00H
FFD0H-FFDFH	External SFR access area(Note 1)	—	R/W	x	x	—	Undefined
FFE0H	Interrupt flag register L	IF0L	R/W	x	x	—	00H
FFE1H	Interrupt flag register H	IF0H	R/W	x	x	—	00H
FFE0H-FFE1H	Interrupt flag register	IF0	R/W	—	—	x	0000H
FFE4H	Interrupt mask flag register L	MK0L	R/W	x	x	—	FFH
FFE5H	Interrupt mask flag register H	MK0H	R/W	x	x	—	FFH
FFE4H-FFE5H	Interrrupt mask flag register	MK0	R/W	—	—	x	FFFFH
FFE8H	Priority order specify flag register L	PR0L	R/W	x	x	—	FFH
FFE9H	Priority order specify flag register H	PR0H	R/W	x	x	—	FFH

3a

Table 2. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FFE8H-FFE9H	Priority order specify flag register	PR0	R/W	—	—	x	FFFFH
FFECH	External interrupt mode register	INTM0	R/W	—	x	—	00H
FFF6H	Key return mode register	KRM	R/W	x	x	—	02H
FFF7H	Pullup resistor option register	PU0	R/W	x	x	—	00H
FFF8H	Memory expanded mode register	MM	R/W	x	x	—	10H
FFF9H	Watchdog timer mode register	WDTM	R/W	x	x	—	00H
FFFAH	Oscillation stabilization time select register	OSTS	R/W	—	x	—	04H
FFFBH	Processor clock control register	PCC	R/W	x	x	—	04H

Note: The external access area cannot be accessed using SFR addressing. It can only be accessed using main memory addressing.

Input/Output Ports

The μPD78002 family has up to 53 port lines. Table 3 lists the features of each port and figure 4 shows the structure of each port pin.

Table 3. Digital Port Functions

Port	Operational Features	Configuration	Direct Drive Capability	Software Pullup Resistor Connection (Note 1)
Port 0 (Note 2)	5-bit input or output	Bit selectable		Byte selectable, input bits only
Port 1	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 2	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 3	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 4	8-bit input or output	Byte selectable		Byte selectable, input bits only
Port 5	8-bit input or output	Bit selectable	LED	Byte selectable, input bits only
Port 6	8-bit input or output ($P6_0$ - $P6_3$ n-channel)	Bit selectable	15 V max ($P6_0$ - $P6_3$)	Byte selectable, input bits only $P6_0$ - $P6_3$ - mask option only (Note 3) $P6_4$ - $P6_7$ - software

Notes:

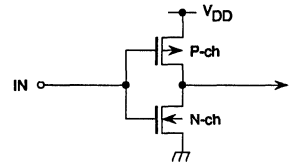
(1) Software pullup resistors can be internally connected (only on a port-by-port basis) to port bits set to input mode. Pullup resistors are not connected to port bits set to output mode.

(2) $P0_0$ and $P0_4$ are input only and do not have a software pullup resistor.

(3) All devices except μPD78P014.

Figure 4. Pin Input/Output Circuits

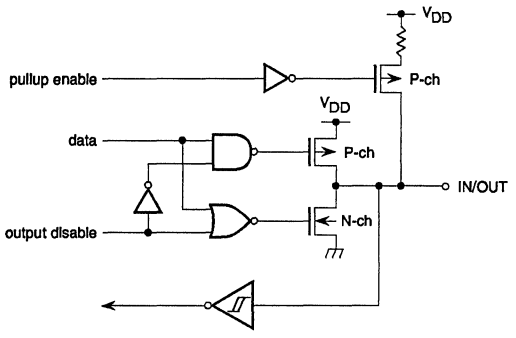
Type 1 ($P0_4$)



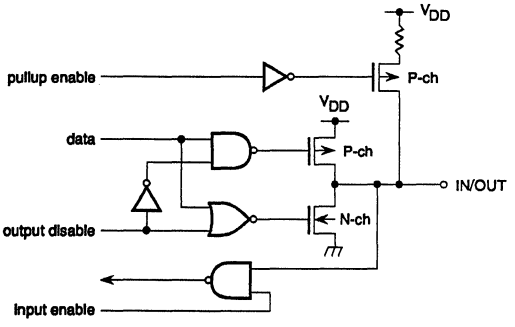
Type 2 ($P0_0$, $\overline{RESET}$)

Schmitt trigger input with hysteresis characteristics

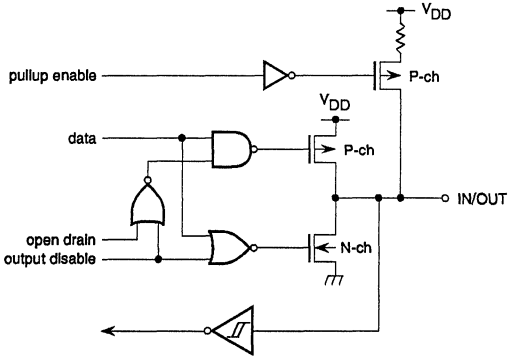
Type 8-A ($P0_1$ - $P0_3$, $P3_3$ - $P3_4$)



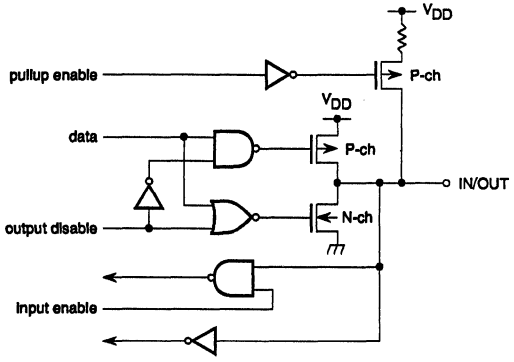
Type 5-A ($P1_0$ - $P1_7$, $P2_0$ - $P2_4$, $P3_0$ - $P3_2$, $P3_5$ - $P3_7$, $P5_0$ - $P5_7$, $P6_4$ - $P6_7$)



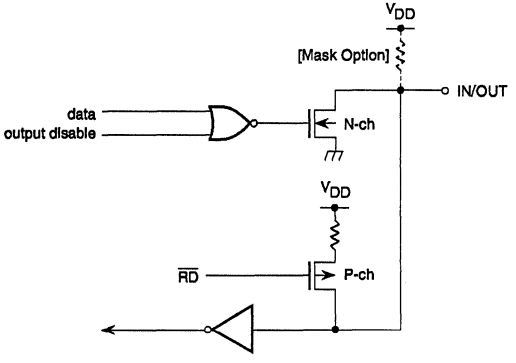
Type 10-A ($P2_5$ - $P2_7$)



Type 5-B ($P4_0$ - $P4_7$)



Type 13-B ($P6_0$ - $P6_3$)



Middle-High Voltage Input Buffer

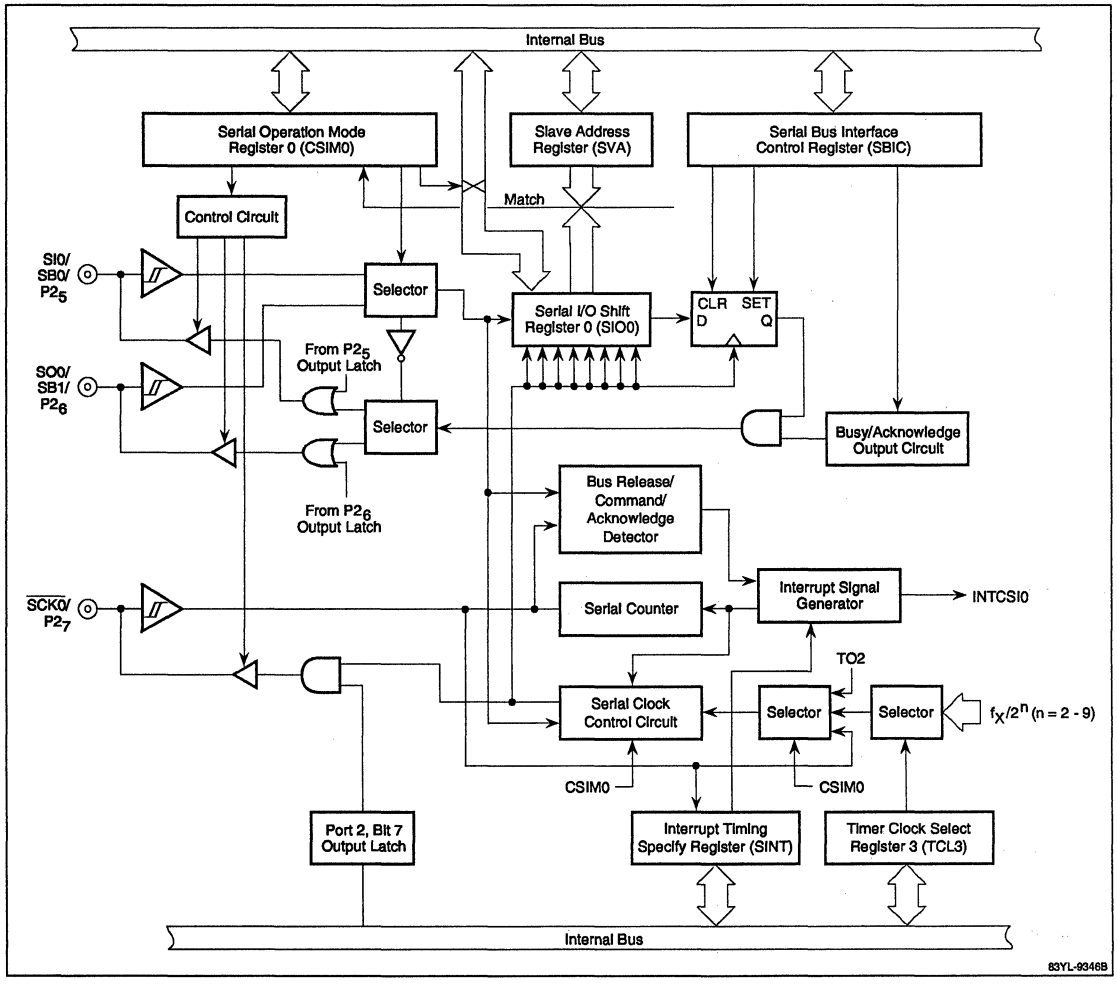
83CL-9481B (9/93)

3a

Serial Interface

The μPD78002 family has one serial interface. Serial interface 0 is an 8-bit clock synchronous serial interface (figure 5). It can be operated in either a three-wire serial I/O mode, NEC serial bus interface (SBI) mode, or two-wire serial I/O mode. The serial clock can be provided from one of eight internal clocks, the output of 8-bit timer register 2, or the external clock line $\overline{SCK0}$.

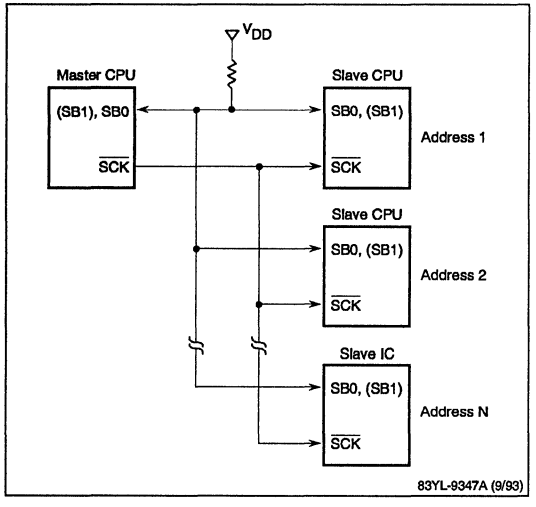
Figure 5. Serial Interface 0



In the three-wire serial I/O mode, the 8-bit shift register (SIO0) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO0 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI0 line providing full-duplex operation. The INTCSI0 interrupt is generated after each 8-bit transfer.

The NEC SBI mode is a two-wire high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are connected in a master/slave configuration (see figure 6). There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over one of the serial bus lines (SB0 or SB1) using a fixed hardware protocol synchronized with the $\overline{SCK0}$ line . Each slave device of the μPD78002 family can be programmed to respond in hardware to any one of 256 addresses set in its slave address register (SVA). There are also 256 commands and 256 data types. Since all commands are user definable, many software protocols, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

Figure 6. SBI Mode Master/Slave Configuration



The two-wire serial I/O mode provides half-duplex operation using either the SB0 or SB1 line and the $\overline{SCK0}$ line. Communication format and handshaking can be handled in software by controlling the output levels of the data and clock lines between transfers. For data transmission, the 8-bit shift register (SIO0) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of either the SB0 or SB1 line MSB first. In addition, this byte of data is also shifted back into SIO0 on the rising edge of these pulses providing a way of verifying that the transmission was correct.

For data reception, the SIO0 register is preloaded with the value FFH. As this data value is shifted out on the falling edge of the serial clock, it disables the n-channel open-drain driver. This allows the receive data to be driven on to the serial line and shifted into the SIO0 register on the rising edge of the serial clock. The INTCSI0 interrupt is generated after each 8-bit transfer.

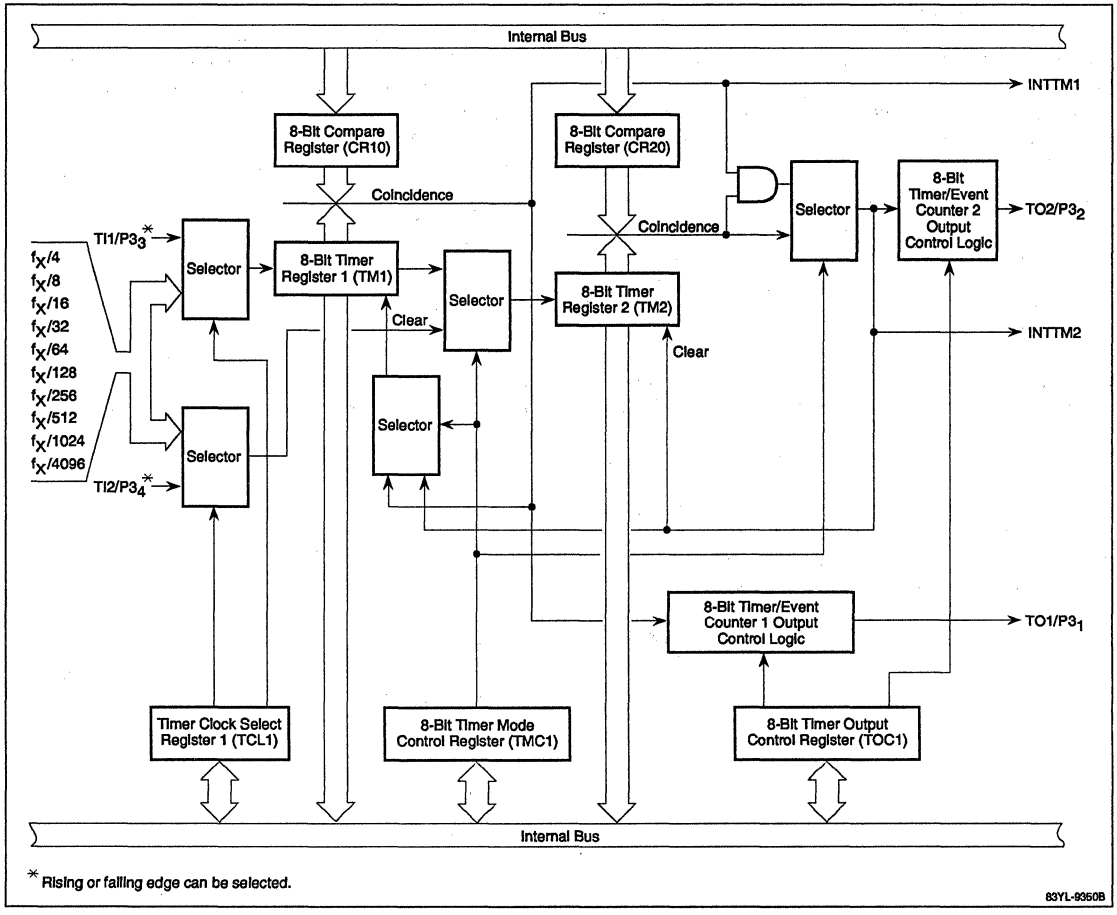
3a

Timers

The μPD78002 family has two 8-bit timer/event counters that can be combined for use as a 16-bit timer/event counter, a watch timer, and a watchdog timer. All of these can be programmed to count a number of prescaled values of the main system clock. In addition, the watch timer can also count the subsystem clock. The two timer/event counters can count external events.

8-Bit Timer/Event Counters 1 and 2. Timer/event counters 1 and 2 (figure 7) each consist of an 8-bit timer (TM1 or TM2), an 8-bit compare register (CR10 or CR20), and a timer output control logic (TO1 or TO2). The timers are controlled by registers TCL1, TMC1, and TOC1 via five seletors. Timer/event counters 1 and 2 can each be used as an 8-bit interval timer, to count external events on the timer input pins (TI1 or TI2), or to output a programmable square wave. In addition, timers 1 and 2 also can be combined as a 16-bit timer/event counter and used as a 16-bit interval timer, to count external events on TI1, or to output a programmable square wave on TO2.

Figure 7. 8-Bit Timer/Event Counters 1 and 2

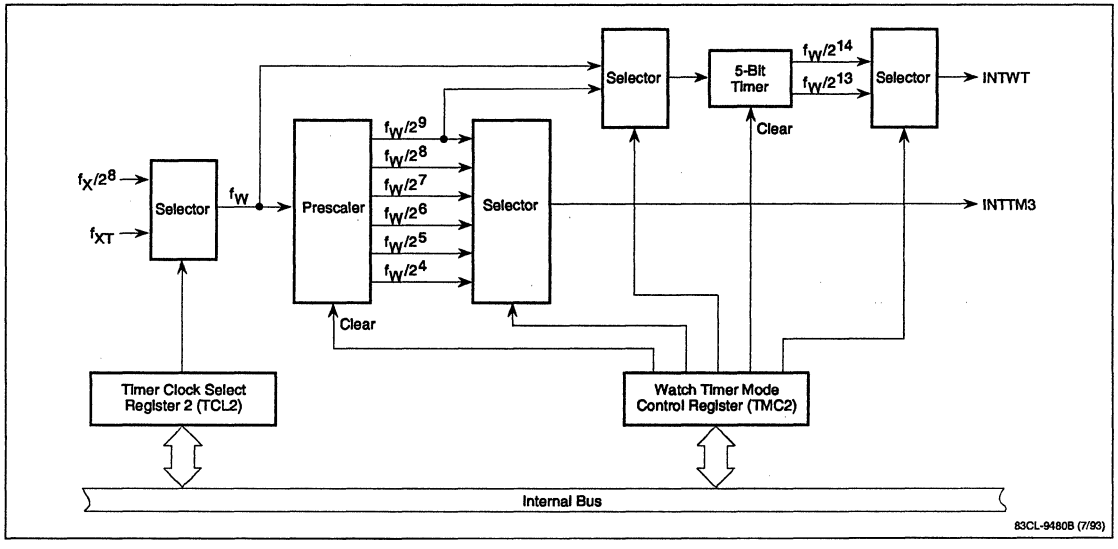


Watch Timer. The watch timer (figure 8) is a 5-bit timer that can be used as a time source to keep track of time of day, to release the STOP or HALT modes at regular intervals, or to initiate any other task that must be performed at regular intervals. When driven by the subsystem clock, the watch timer continues to operate in the STOP mode.

The watch timer can function as both a watch timer and an interval timer simultaneously. When used as a watch timer, interrupt request INTWT (not a vectored interrupt) can be generated using a main system clock or a subsystem clock every 0.5 or 0.25 seconds.

When used as an interval timer, vectored interrupt request INTTM3 is generated at preselected time intervals. With a main system clock of 8.38 MHz or a subsystem clock of 32.768 kHz, the following time intervals can be selected: 489 μs, 978 μs, 1.96 ms, 3.91 ms, 7.82 ms or 15.6 ms.

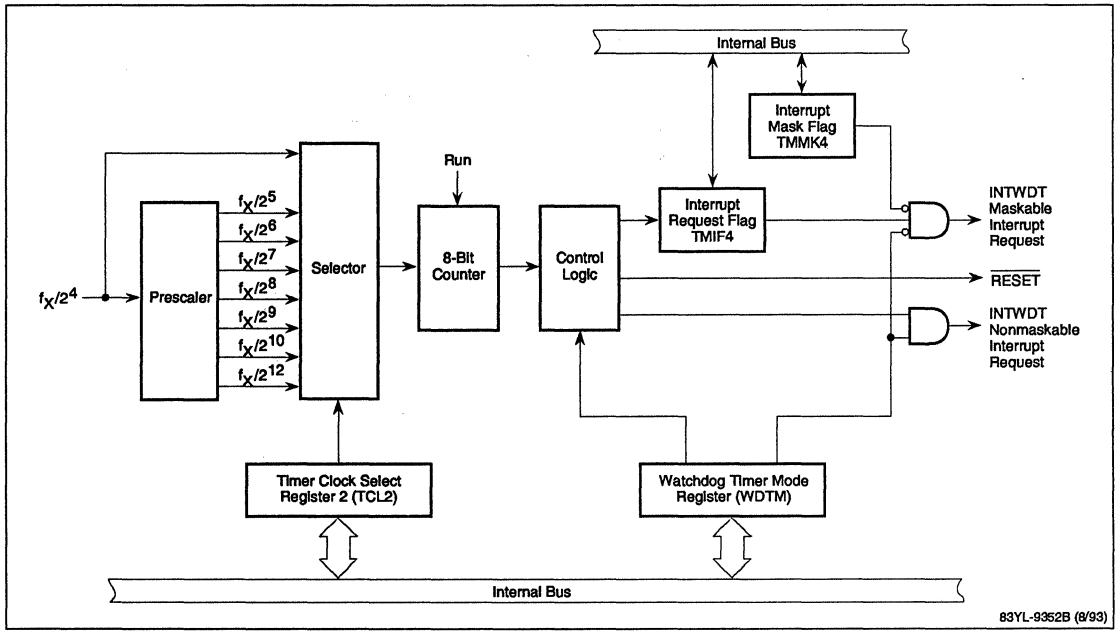
Figure 8. Watch Timer



Watchdog Timer. The watchdog timer (figure 9) can be used as either a watchdog timer or an interval timer. When used as a watchdog timer, it protects against program run-away. It can be selected to generate a nonmaskable interrupt (INTWDT), which vectors to address 0004H, or to generate an internal reset signal, which vectors to the restart address 0000H if the timer is not cleared by the program before it overflows. Eight program-selectable intervals based on the main system clock are available. With a main system clock of 8.38 MHz, the program selectable intervals are 0.489, 0.978, 1.96, 3.91, 7.82, 15.6, 31.3, and 125 ms. Once the watchdog timer is initialized and started, the timer's mode cannot be changed and the timer can only be stopped by an external reset.

When used as an interval timer, maskable interrupts (INTWDT), which vector to address 0004H, are generated repeatedly at a preset interval. The time intervals available are the same as in the watchdog timer mode.

Figure 9. Watchdog Timer



Programmable Clock Output

The μPD78002 family has a programmable clock output (PCL) that can be used for carrier output for remote controlled transmissions or as a clock output for peripheral devices. The main system clock (f_X) divided by 8, 16, 32, 64, 128, or 256 or the subsystem clock (f_{XT}) can be output on the PCL pin. Frequencies of 1050, 524, 262, 131, 65.5 and 32.7 kHz are available with a main system clock of 8.38 MHz. See figure 10.

Buzzer Output

The μPD78002 family also has a programmable buzzer output (BUZ). The buzzer output frequency can be programmed to be equal to the main system clock (f_X) divided by 1024, 2048, or 4096. With a main system clock of 8.38 MHz, the buzzer can be set to 8.2, 4.1 or 2.0 kHz. See figure 11.

Figure 10. Programmable Clock Output

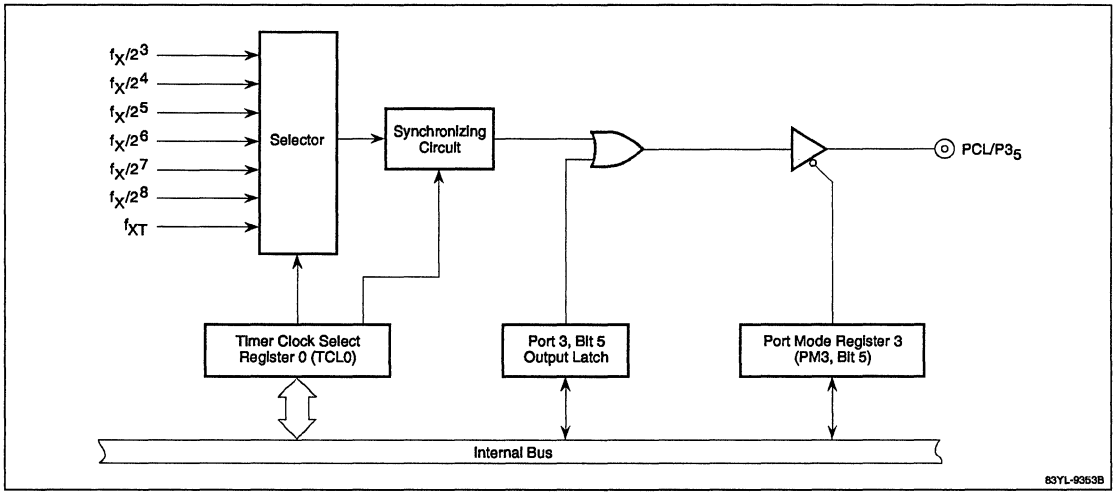
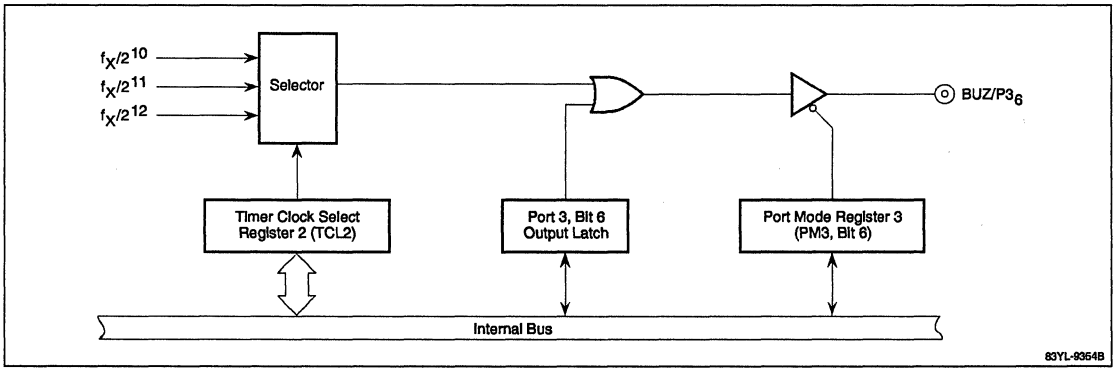


Figure 11. Buzzer Output



Interrupts

The μPD78002 family has 11 maskable hardware interrupt sources (5 external and 6 internal). Of these 11 interrupt sources, 9 cause a vectored interrupt while the 2 testable inputs only generate an interrupt request. All of the 11 maskable interrupts can be used to release the HALT mode except INTP0. INTP0 cannot be used to release the STOP mode and cannot release the HALT mode when register SCS = 0. In addition, there is one nonmaskable interrupt from the watchdog timer, one software interrupt, and a RESET interrupt. The watchdog timer overflow interrupt (interrupt vector table address 0004H) can be initialized to be a nonmaskable interrupt or the highest default priority maskable interrupt. The software interrupt, generated by the BRK instruction, is not maskable. See table 4 and figure 12.

Table 4. Interrupt Sources and Vector Addresses

Type of Request	Default Priority	Signal Name	Interupt Source	Location	Vector Address	Interrupt Configuration
Restart	—	RESET	RESET input pin	External	0000H	—
	—	INTWDT	Watchdog timer overflow (when reset mode selected)	Internal		
Nonmaskable	—	INTWDT	Watchdog timer overflow (when nonmaskable interrupt selected)	Internal	0004H	A
Maskable	0	INTWDT	Watchdog timer overflow (when interval timer selected)	Internal	0004H	B
	1	INTP0	External interrupt edge detection	External	0006H	C
	2	INTP1	External interrupt edge detection	External	0008H	D
	3	INTP2	External interrupt edge detection	External	000AH	D
	4	INTP3	External interrupt edge detection	External	000CH	D
	5	INTCSI0	End of clocked serial interface 0 transfer	Internal	000EH	B
	6	INTTM3	Watch timer reference time interval signal	Internal	0012H	B
	7	INTTM1	8-bit timer/event counter 1 coincidence signal	Internal	0016H	B
	8	INTTM2	8-bit timer/event counter 2 coincidence signal	Internal	0018H	B
Software	—	—	BRK instruction	Internal	003EH	E
Test input	—	INTWT	Watch timer overflow	Internal	—	F
	—	INTPT4	Port 4 falling edge detection	External	—	F

Interrupt Servicing. The μPD78002 family provides two levels of programmable hardware priority control and services all interrupt requests, except the two testable interrupts (INTWT and INTPT4). Using vectored interrupts, the programmer can choose the priority of servicing each maskable interrupt by using the interrupt control registers.

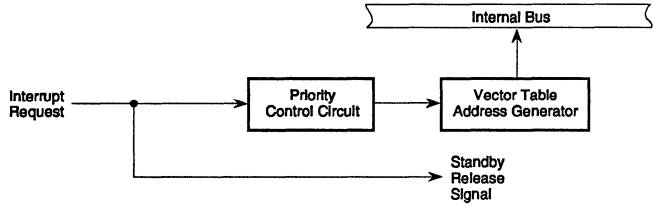
Interrupt Control Registers. The μPD78002 family has three 16-bit interrupt control registers. The interrupt request flag register (IF0) contains an interrupt request flag for each interrupt except INTPT4. The interrupt mask register (MK0) is used to enable or disable any interrupt except INTPT4. The priority flag register (PR0) can be used to specify a high or a low priority level for each interrupt except the two testable interrupts.

Four other 8-bit registers are associated with interrupt processing. The key return mode register (KRM) contains the KRIF interrupt request flag associated with falling-edge detection on port 4 and the KRMK mask flag used to enable or disable clearing of the standby mode if a falling edge is detected on port 4. The external interrupt mode register (INTM0) is used to select a rising, falling, or both edges as the valid edge for each of the external interrupts INTP0 to INTP2 (INTP3 is always falling edge). The sampling clock select register (SCS) is used to select a sampling clock for the noise eliminator circuit on external interrupt INTP0.

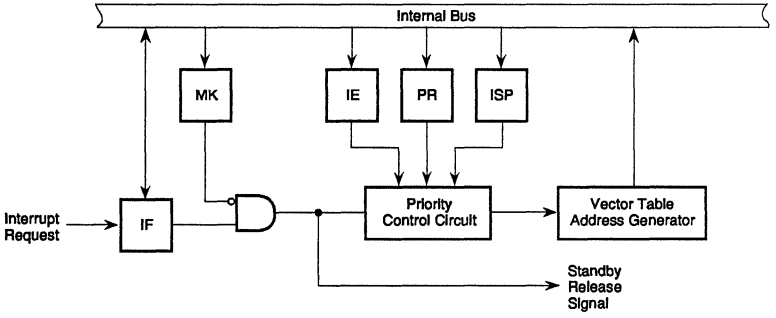
The IE and the ISP bit of the program status word are also used to control interrupts. If the IE bit is 0, all maskable interrupts are disabled. The IE bit can be set or cleared using the EI and DI instructions, respectively, or by directly writing to the PSW. The IE bit is cleared each time an interrupt is accepted. The ISP bit is used by hardware to hold the priority level flag of the interrupt being serviced.

Figure 12. Interrupt Configurations

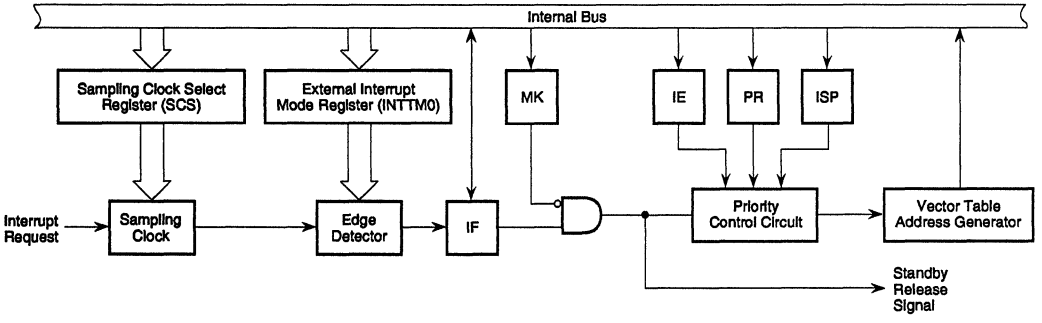
Type A: Internal nonmaskable interrupt



Type B: Internal maskable interrupt



Type C: External maskable interrupt (INTP0)

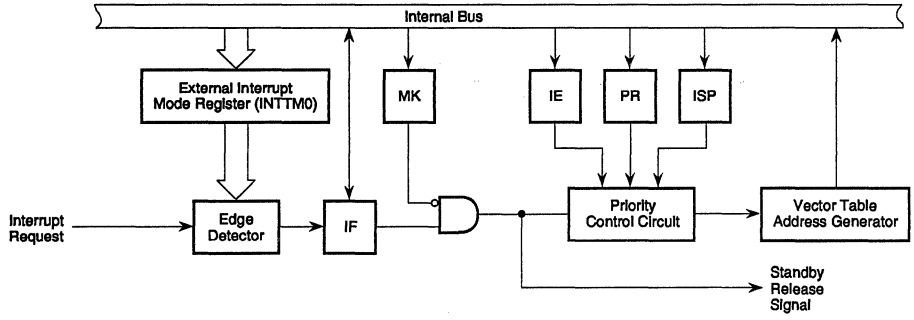


83YL-9356B (1)

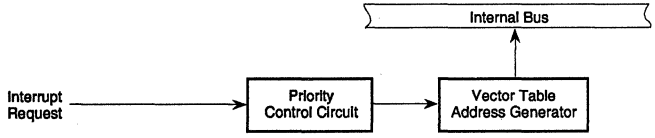
3a

Figure 12. Interrupt Configurations (cont)

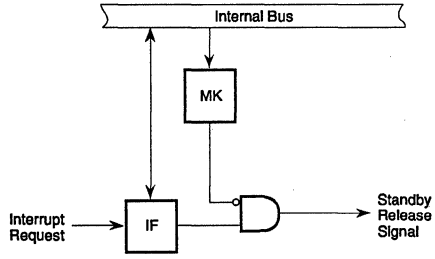
Type D: External maskable interrupt (except INTP0)



Type E: Software interrupt



Type F: Test input



Abbreviations:

- IF: Interrupt request flag
- IE: Interrupt enable flag
- ISP: In-service priority flag
- MK: Interrupt mask flag
- PR: Priority specify flag

83YL-9355B (2)

Interrupt Priority. If the watchdog timer overflow interrupt (INTWDT) has been initialized to be a nonmaskable interrupt, it has priority over all other interrupts. Two hardware-controlled priority levels are available for all maskable interrupts that generate a vectored interrupt (i.e., all except the two testable interrupts). Either a high or a low priority level can be assigned by software to each of the maskable interrupts. Interrupt requests of the same priority or a priority higher than the processor's current priority level are held pending until interrupts in the current service routine are enabled by software or until one instruction has been executed after returning from the current service routine. Interrupt requests of a lower priority are always held pending until one instruction has been executed after returning from the current service routine.

The default priorities listed in table 4 are fixed by hardware and are effective only when it is necessary to choose between two interrupt requests of the same software-assigned priority. For example, the default priorities would be used after the completion of a high-priority routine, if two interrupts of the same software priority were pending.

The software interrupt, initiated by the BRK instruction, is executed regardless of the processor's priority level and the state of the IE bit. It does not alter the processor's priority level.

Vectored Interrupt Servicing. When a vectored interrupt is acknowledged, the program status word and the program counter are saved on the stack, the processor's priority is set to that specified for the interrupt, the IE bit in the PSW is set to zero, and the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (RETB instruction for the software interrupt) reverses the process and the μPD78002 family microcontroller resumes the interrupted routine.

Standby Modes

HALT, STOP, and data retention modes are provided to reduce power consumption when CPU action is not required.

The HALT mode is entered by executing a HALT instruction while the CPU is operating from the main system or subsystem clock. In HALT mode, the CPU clock is stopped while the main system and the subsystem clock continue to run. The HALT mode is released by any unmasked interrupt request (except INTP0 if register SCS= 0), a nonmaskable interrupt request, an unmasked test input, or an external reset pulse.

Power consumption may be further reduced by using the STOP mode. The STOP mode is entered by executing a STOP instruction while operating from the main system clock. In STOP mode, the main system clock input pin X1 is internally grounded stopping both the CPU and the peripheral hardware clock. The STOP mode is released by any unmasked interrupt request except INTP0, a nonmaskable interrupt request, an unmasked test input, or an external reset pulse. Any peripheral using the main oscillator as a clock source will also be disabled in the STOP mode and interrupts from such a peripheral cannot be used to exit the STOP mode. Table 5 summarizes both the HALT and STOP standby modes.

Table 5. Standby Mode Operation Status

Item	HALT Mode	STOP Mode
Setting instruction	HALT instruction	STOP instruction
System clock when setting	Main system or subsystem clock	Main system clock
Clock oscillator	Main system and subsystem clocks can oscillate; CPU clock is stopped.	Subsystem clock can oscillate; CPU clock and main system clock are stopped.
CPU	Operation stopped	Operation stopped
Ports	Maintain previous state	Maintain previous state
8-bit timer/ event counters	Operational from main system clock	Operational only with TI1 and TI2 as count clock
Watch timer	Operational from main system clock or with f_{XT} as count clock	Operational only with f_{XT} as count clock
Watchdog timer	Operational from main system clock	Operation stopped
Serial interface 0	Operational from main system clock	Operational only with external clock
External interrupts	Operational except for INTP0 when its sampling clock is based on the CPU clock	INTP0 not operational; INTP1 to INTP3 operational

When exiting the STOP mode, a wait time occurs before the CPU begins execution to allow the main system clock oscillator circuit to stabilize. The oscillator stabilization time is selected by programming the OSTS register with one of five values before entering the STOP mode. The values range from 0.98 msec to 31.3 msec at f_X= 8.38 MHz.

Once in the STOP mode, power consumption can be further minimized by lowering the power supply voltage V_{DD} to as little as 2 V. This places the device in the data retention mode. The contents of Internal RAM and the registers are retained. This mode is released by first raising V_{DD} to the proper operating range and then releasing the STOP mode.

External Reset

The μPD78002 family is reset by taking the $\overline{\text{RESET}}$ pin low or by an overflow of the watchdog timer (if enabled). The $\overline{\text{RESET}}$ input pin is a schmitt-trigger input with hysteresis characteristics to protect against spurious system resets caused by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low for a minimum of 10 μs after the power supply reaches its operating voltage.

There is no functional difference between an external reset and an internal reset caused by the overflow of the watchdog timer. In both cases, the main system clock oscillation is stopped and the subsystem clock oscillation continues. During reset, the program counter is loaded with the address contained in the reset vector (addresses 0000H, 0001H). Once the reset is cleared and the oscillation stabilization time of $2^{18}/f_X$ has elapsed, program execution starts at that address.

ELECTRICAL SPECIFICATIONS

The following specifications are for the μPD78001B/002B devices only. Refer to the μPD78014 data sheet for μPD78P014 device specifications.

Absolute Maximum Ratings

T_A = +25°C

Supply voltage, V_{DD}	−0.3 to +7.0 V
Input voltage, V_{I1} (except $P6_0$ to $P6_3$)	−0.3 to V_{DD}+0.3 V
Input voltage, V_{I2} ($P6_0$ to $P6_3$; open drain)	−0.3 to +16 V
Output voltage, V_O	−0.3 to V_{DD}+0.3 V
Output current, high; I_{OH}	
Each output pin	−10 mA
Total: ports 1 to 3	−15 mA
Total: ports 0 and ports 4 to 6	−15 mA
Output current, low, I_{OL} †	
Each output pin	30 mA peak, 15 mA rms
Total: $P4_0$ to $P4_7$ and $P5_0$ to $P5_5$	100 mA peak, 70 mA rms
Total: $P0_1$ to $P0_3$, $P5_6$, $P5_7$, and $P6_0$ to $P6_7$	100 mA peak, 70 mA rms
Total: $P0_1$ to $P0_3$ and $P6_4$ to $P6_7$	50 mA peak, 20 mA rms
Total: ports 1 to 3	50 mA peak, 20 mA rms
Operating temperature, T_{OPT}	−40 to +85°C
Storage temperature, T_{STG}	−65 to +150°C

† rms value = peak value x (duty cycle) $^{1/2}$

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC characteristics.

Capacitance

T_A = +25°C; V_{DD} = V_{SS} = 0 V

Parameter	Symbol	Max	Unit	Conditions	
Input capacitance	C_{IN}	15	pF	Except $P6_0$ to $P6_3$	f = 1 MHz; unmeasured pins returned to ground
		20	pF	$P6_0$ to $P6_3$	
Output capacitance	C_{OUT}	15	pF	Except $P6_0$ to $P6_3$	
		20	pF	$P6_0$ to $P6_3$	
Input/output capacitance	C_{IO}	15	pF	Except $P6_0$ to $P6_3$	
		20	pF	$P6_0$ to $P6_3$	

3a

Main System Clock Oscillator

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 13.

Type	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Figure 13A)	Oscillation frequency (Note 1)	f_X	1.0		10.0	MHz	V_{DD} = oscillator voltage range
	Oscillation stabilization time (Note 2)				4.0	ms	After V_{DD} reaches oscillator operating voltage
Crystal resonator (Figure 13A)	Oscillation frequency (Note 1)	f_X	1.0	8.38	10.0	MHz	
	Oscillation stabilization time (Note 2)				10	ms	V_{DD} = 4.5 to 6.0 V
					30	ms	
External clock (Figure 13B)	X1 input frequency (Note 1)	f_X	1.0		10.0	MHz	
	X1 input high/low-level width	t_{XH}, t_{XL}	50		500	ns	

Notes:

(1) Oscillator and X1 input frequencies are included only to show the oscillator characteristics. Refer to the AC Characteristics table for actual instruction execution times.

(2) Time required for the oscillator to stabilize after reset or STOP mode is released. The values shown are for the recommended resonators. Values for resonators not shown in this data sheet should be obtained from the manufacturer's specification sheets.

Subsystem Clock Oscillator

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 14.

Type	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Crystal resonator (Figure 14A)	Oscillation frequency (Note 1)	f_{XT}	32	32.768	35	kHz	
	Oscillation stabilization time (Note 2)			1.2	2	s	V_{DD} = 4.5 to 6.0 V
					10	s	
External clock (Figure 14B)	XT1 input frequency (Note 1)	f_{XT}	32		100	kHz	
	XT1 input high/low-level width	t_{XTH}, t_{XTL}	5		15	μs	

Notes:

(1) The oscillator and XT1 input frequencies are included only to show the oscillator characteristics. Refer to the AC Characteristics table for actual instruction execution times.

(2) Time required for the oscillator to stabilize after reset or STOP mode is released. The values shown are for the recommended resonators. Values for resonators not shown in this data sheet should be obtained from the manufacturer's specification sheets.

Figure 13. Main System Clock Configurations

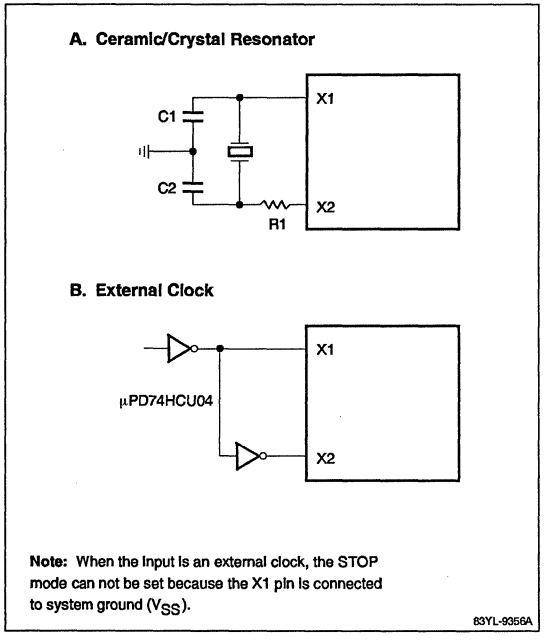
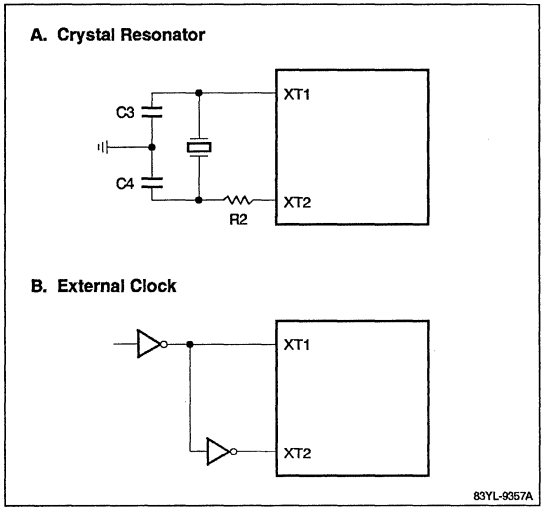


Figure 14. Subsystem Clock Configurations



Recommended Main System Clock Ceramic Resonators

T_A = −40 to +85°C; refer to figure 13A

Part Number (Notes 1 and 2)	Recommended Circuit Constant			Oscillator Voltage Range		Frequency (MHz)
	C1 (pF)	C2 (pF)	R1 (kΩ)	Min (V)	Max (V)	
CSB1000J	100	100	6.8	2.7	6.0	1.00
CSBxxxxJ	100	100	4.7	2.7	6.0	1.01 to 1.25
CSAx.xxxMK	100	100	0	2.7	6.0	1.26 to 1.79
CSAx.xxMG	100	100	0	2.7	6.0	1.80 to 2.44
CSTx.xxMG	(Note 3)	(Note 3)	0	2.7	6.0	1.80 to 2.44
CSAx.xxMG	30	30	0	2.7	6.0	2.45 to 4.18
CSTx.xxMGW	(Note 3)	(Note 3)	0	2.7	6.0	2.45 to 4.18
CSAx.xxMG	30	30	0	2.7	6.0	4.19 to 6.00
CSTx.xxMGW	(Note 3)	(Note 3)	0	2.7	6.0	4.19 to 6.00
CSAx.xxMT	30	30	0	2.7	6.0	6.01 to 10.0
CSTx.xxMTW	(Note 3)	(Note 3)	0	2.7	6.0	6.01 to 10.0

Notes:

(1) Manufactured by Murata Mfg. Co., Ltd.

(2) x.xx indicates frequency

(3) C1 and C2 are contained in the ceramic resonators.

Recommended Subsystem Clock Crystal Resonators

T_A = −40 to +60°C; refer to figure 14A

Part Number †	Frequency (kHz)	Recommended Circuit Constant			Oscillator Voltage Range	
		C3 (pF)	C4 (pF)	R2 (kΩ)	Min (V)	Max (V)
DT-38 (1TA252 E00, load capacitance 6.3 pF)	32.768	12	12	100	2.7	6.0

† Manufactured by Daishinku

DC Characteristics

T_A = −40 to +85°C; V_{DD} = +2.7 to 6.0 V; refer to figures 15-20

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Other than below
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	$P0_0$ to $P0_4$, $P2_0$, $P2_2$, $P2_4$ to $P2_7$, $P3_3$, $P3_4$, $\overline{RESET}$
	V_{IH3}	0.7 V_{DD}		15	V	$P6_0$ to $P6_3$; open-drain
	V_{IH4}	V_{DD} − 0.5		V_{DD}	V	X1, X2
	V_{IH5}	V_{DD} − 0.5		V_{DD}	V	V_{DD} = 4.5 to 6.0 V; XT1, XT2
Low-level input voltage	V_{IL1}	0		0.3 V_{DD}	V	Other than below
	V_{IL2}	0		0.2 V_{DD}	V	$P0_0$ to $P0_4$, $P2_0$, $P2_2$, $P2_4$ to $P2_7$, $P3_3$, $P3_4$, $\overline{RESET}$
	V_{IL3}	0		0.3 V_{DD}	V	$P6_0$ to $P6_3$; V_{DD} = 4.5 to 6.0 V
		0		0.2 V_{DD}	V	$P6_0$ to $P6_3$
	V_{IL4}	0		0.4	V	X1, X2
	V_{IL5}	0		0.4	V	XT1, XT2; V_{DD} = 4.5 to 6.0 V
		0		0.3	V	XT1, XT2
High-level output voltage	V_{OH1}	V_{DD} − 1.0			V	V_{DD} = 4.5 to 6.0 V, I_{OH} = −1.0 mA
		V_{DD} − 0.5			V	I_{OH} = −100 μA
Low-level output voltage	V_{OL1}		0.4	2.0	V	$P5_0$ to $P5_7$, $P6_0$ to $P6_3$; V_{DD} = 4.5 to 6.0 V, I_{OL} = 15 mA
				0.4	V	Other than above; V_{DD} = 4.5 to 6.0 V, I_{OL} = 1.6 mA
	V_{OL2}			0.2 V_{DD}	V	SB0, SB1, $\overline{SCK0}$; V_{DD} = 4.5 to 6.0 V, open-drain, pullup resistance = 1 kΩ
	V_{OL3}			0.5	V	I_{OL} = 400 μA
High-level input leakage current	I_{LIH1}			3	μA	V_{IN} = V_{DD}; except X1, X2, XT1, XT2
	I_{LIH2}			20	μA	V_{IN} = V_{DD}; X1, X2, XT1, XT2
	I_{LIH3}			80	μA	V_{IN} = 15 V; $P6_0$ to $P6_3$
Low-level input leakage current	I_{LIL1}			−3	μA	V_{IN} = 0 V; except X1, X2, XT1, XT2
	I_{LIL2}			−20	μA	V_{IN} = 0 V; X1, X2, XT1, XT2
	I_{LIL3}			−3	μA	V_{IN} = 0 V; $P6_0$ to $P6_3$ (Note 1)
Output leakage current high	I_{LOH1}			3	μA	V_{OUT} = V_{DD}
Output leakage current low	I_{LOL}			−3	μA	V_{OUT} = 0 V
Mask option pullup resistor	R_1	20	40	90	kΩ	V_{IN} = 0 V; $P6_0$ to $P6_3$
Software pullup resistor	R_2	15	40	90	kΩ	V_{DD} = 4.5 to 6.0 V; V_{IN} = 0 V, $P0_1$ to $P0_3$, ports 1 to 5, $P6_4$ to $P6_7$
		20		500	kΩ	V_{DD} = 2.7 to 4.5 V; V_{IN} = 0 V, $P0_1$ to $P0_3$, ports 1 to 5, $P6_4$ to $P6_7$

DC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Power supply current	I_{DD1}		7.5	22.5	mA	8.38 MHz crystal oscillation operating mode; V_{DD} = 5.0 V ±10% (Note 2)
			0.8	2.4	mA	8.38 MHz crystal oscillation operating mode; V_{DD} = 3.0 V ±10% (Note 3)
	I_{DD2}		1.4	4.2	mA	8.38 MHz crystal oscillation HALT mode; V_{DD} = 5.0 V ±10%
			550	1650	μA	8.38 MHz crystal oscillation HALT mode; V_{DD} = 3.0 V ±10%
	I_{DD3}		60	120	μA	32.768 kHz crystal oscillation operating mode; V_{DD} = 5.0 V ±10%, X1 STOP mode, CPU operating from subsystem clock
			35	70	μA	32.768 kHz crystal oscillation operating mode; V_{DD} = 3.0 V ±10%, X1 STOP mode, CPU operating from subsystem clock
	I_{DD4}		25	50	μA	32.768 kHz crystal oscillation HALT mode; V_{DD} = 5.0 V ±10%, X1 STOP mode
			5	10	μA	32.768 kHz crystal oscillation HALT mode; V_{DD} = 3.0 V ±10%, X1 STOP mode
	I_{DD5}		1	20	μA	XT1 = 0 V STOP mode when feedback resistor is connected; V_{DD} = 5.0 V ±10%
			0.5	10	μA	XT1 = 0 V STOP mode when feedback resistor is connected; V_{DD} = 3.0 V ±10%
	I_{DD6}		0.1	20	μA	XT1 = 0 V STOP mode when feedback resistor is disconnected; V_{DD} = 5.0 V ±10%
			0.05	10	μA	XT1 = 0 V STOP mode when feedback resistor is disconnected; V_{DD} = 3.0 V ±10%

Notes:

(1) $P6_0$ to $P6_3$ become −200 μA (max.) for only 1 clock cycle during input instruction execution (no wait) and −3 μA (max.) during instruction other than input.

(2) When operated in the high-speed mode with the processor clock control register set to 00H.

(3) When operated in low-speed mode with the processor clock control register set to 04H.

3a

Figure 15. I_{DD} vs V_{DD} (f_X = 8.38 MHz)

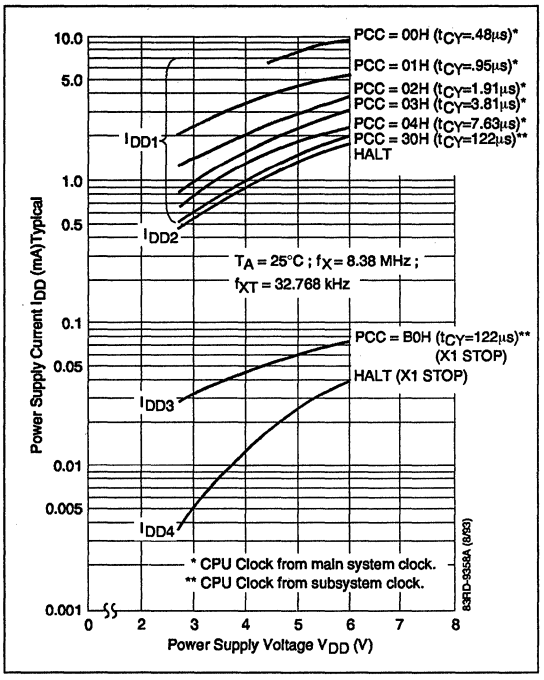


Figure 16. I_{DD} vs V_{DD} (f_X = 4.19 MHz)

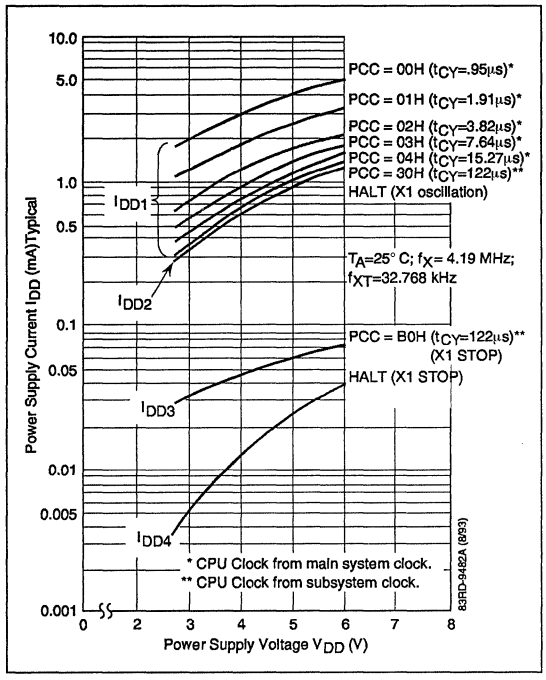


Figure 17. I_{OL} vs V_{OL} (Ports 0, 2-5, $P6_4$ - $P6_7$)

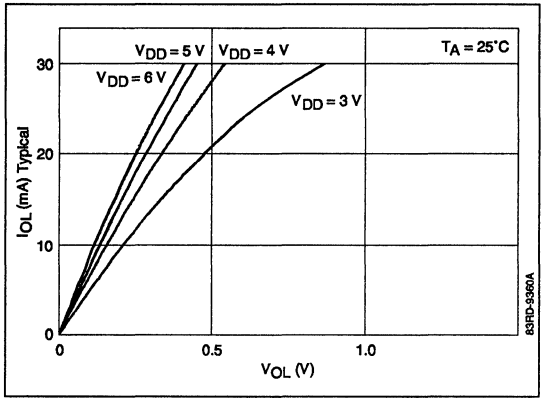


Figure 18. I_{OL} vs V_{OL} (Port 1)

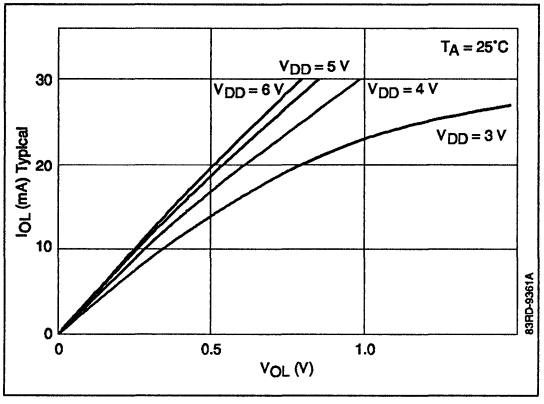


Figure 19. I_{OL} vs V_{OL} ($P6_0$ - $P6_3$)

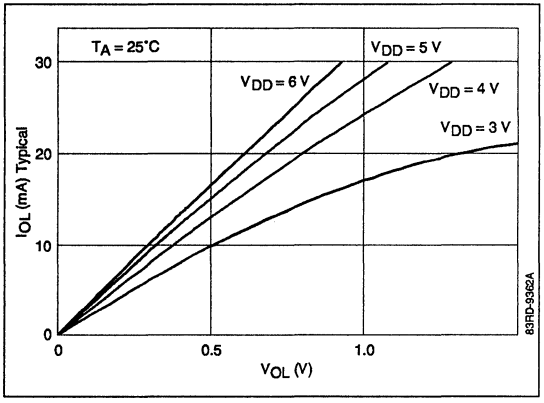
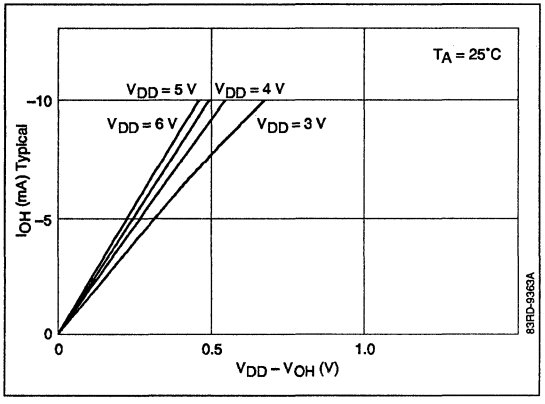


Figure 20. I_{OH} vs $V_{DD} - V_{OH}$ (Ports 0-5, $P6_4$ - $P6_7$)



3a

AC Characteristics

T_A = −40 to +85°C; V_{DD} = 2.7 to 6 V ; refer to figures 21 through 26

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time (Min. instruction execution time)	t_{CY}	0.4		64	μs	V_{DD} = 4.5 to 6.0 V; operating on main system clock
		0.96		64	μs	Operating on main system clock
		114	122	125	μs	Operating on subsystem clock
TI input frequency	f_{TI}	0		4	MHz	V_{DD} = 4.5 to 6.0 V
		0		275	kHz	
TI input high/low-level width	t_{TIH}, t_{TIL}	100			ns	V_{DD} = 4.5 to 6.0 V
		1.8			μs	
Interrupt input high/low-level width	t_{INTH}, t_{INTL}	$8/f_{sam}$ (Note 1)			μs	INTP0
		10			μs	INTP1 to INTP3
		10			μs	KR0 to KR7 (Note 2)
$\overline{\text{RESET}}$ low-level width	t_{RST}	10			μs	

Notes:

(1) By using bits 0 and 1 of the sampling clock select (SCS) register in conjunction with bits 0 to 2 of the processor clock control (PCC) register, f_{sam} can be set to $f_X/2^{N+1}$ (where N = 0 to 4), $f_X/64$, or $f_X/128$.

(2) Port 4 falling-edge detection input.

Figure 21. Main System Clock Operation; t_{CY} vs V_{DD}

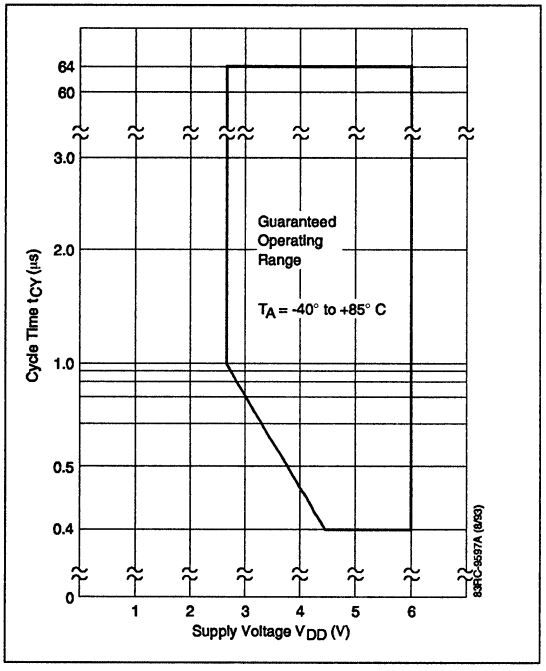


Figure 22. AC Timing Measurements Points (except X1 and XT1)

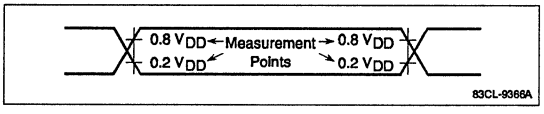


Figure 23. Clock AC Timing Points X1 and XT1

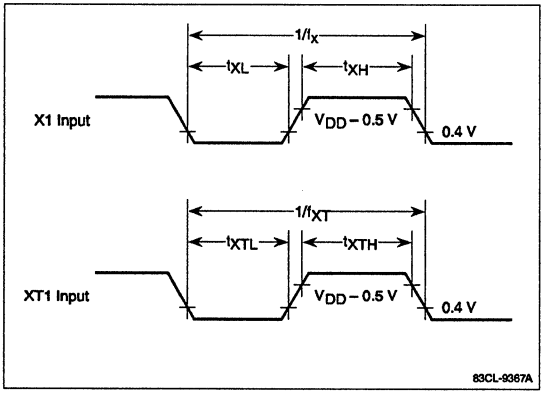


Figure 24. TI Timing

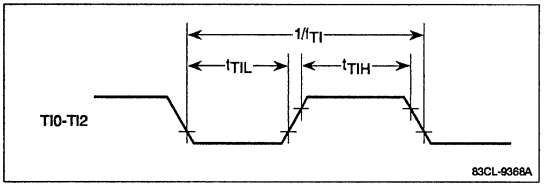


Figure 25. Interrupt Input Timing

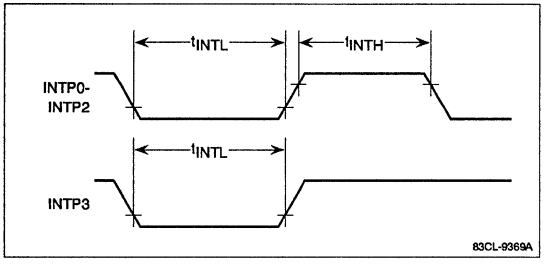
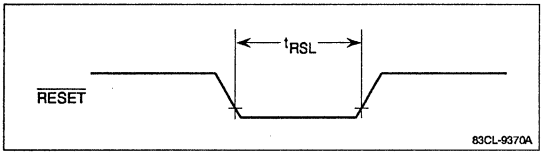


Figure 26. $\overline{RESET}$ Input Timing



3a

Read/Write Operation

T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 V; refer to figures 27 through 30

Parameter	Symbol	Min	Max	Unit	Conditions
ASTB high-level width	t_{ASTH}	0.5 t_{CY}		ns	
Address setup time to ASTB ↓	t_{ADS}	0.5 t_{CY} − 30		ns	
Address hold time from ASTB ↓	t_{ADH}	10		ns	Load resistor ≥ 5 kΩ
Data input time from address	t_{ADD1}		(2+2n)t_{CY} − 50	ns	Instruction fetch
	t_{ADD2}	5	(3+2n)t_{CY} − 100	ns	Data access
Data input time from $\overline{RD}$ ↓	t_{RDD1}		(1+2n)t_{CY} − 25	ns	Instruction fetch
	t_{RDD2}		(2.5+2n)t_{CY} − 100	ns	Data access
Read data hold time	t_{RDH}	0		ns	
$\overline{RD}$ low-level width	t_{RDL1}	(1.5+2n)t_{CY} − 20		ns	Instruction fetch
	t_{RDL2}	(2.5+2n)t_{CY} − 20		ns	Data access
$\overline{WAIT}$ ↓ input time from $\overline{RD}$ ↓	t_{RDWT1}		0.5 t_{CY}	ns	Instruction fetch
	t_{RDWT2}		1.5 t_{CY}	ns	Data access
$\overline{WAIT}$ ↓ input time from $\overline{WR}$ ↓	t_{WRWT}		0.5 t_{CY}	ns	
$\overline{WAIT}$ low-level width	t_{WTL}	(0.5+2n)t_{CY} + 10	(2+2n)t_{CY}	ns	
Write data setup time to $\overline{WR}$ ↑	t_{WDS}	100		ns	
Write data hold time from $\overline{WR}$ ↑	t_{WDH}	5		ns	
$\overline{WR}$ low-level width	t_{WRL1}	(2.5+2n)t_{CY} − 20		ns	
$\overline{RD}$ ↓ delay time from ASTB ↓	t_{ASTRD}	0.5 t_{CY} − 30		ns	
$\overline{WR}$ ↓ delay time from ASTB ↓	t_{ASTWR}	1.5 t_{CY} − 30		ns	
ASTB ↑ delay time from $\overline{RD}$ ↑ (external fetch)	t_{RDAST}	t_{CY} − 10	t_{CY} + 40	ns	
Address hold time from $\overline{RD}$ ↑ (external fetch)	t_{RDADH}	t_{CY}	t_{CY} + 50	ns	
Write data output time from $\overline{RD}$ ↑	t_{RDWD}	10		ns	
WR ↓ delay time from write data	t_{WDWR}	0.5 t_{CY} − 120	0.5 t_{CY}	ns	V_{DD} = 4.5 to 6.0 V
		0.5 t_{CY} − 170	0.5 t_{CY}	ns	
Address hold time from $\overline{WR}$ ↑	t_{WRADH}	t_{CY}	t_{CY} + 60	ns	V_{DD} = 4.5 to 6.0 V
		t_{CY}	t_{CY} + 100	ns	
$\overline{RD}$ ↑ delay time from $\overline{WAIT}$ ↑	t_{WTRD}	0.5 t_{CY}	2.5 t_{CY} + 80	ns	
$\overline{WR}$ ↑ delay time from $\overline{WAIT}$ ↑	t_{WTWR}	0.5 t_{CY}	2.5 t_{CY} + 80	ns	

Notes:

(1) $t_{CY} = t_{CY}/4$

(2) n indicates number of waits.

(3) C_L = 100 pF

Figure 27. Read Operation; External Fetch (No Wait)

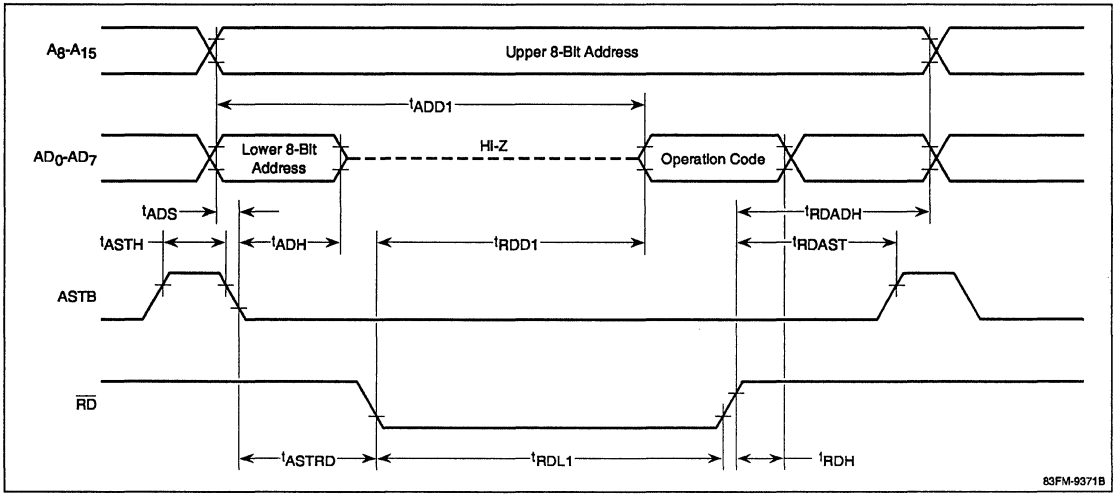
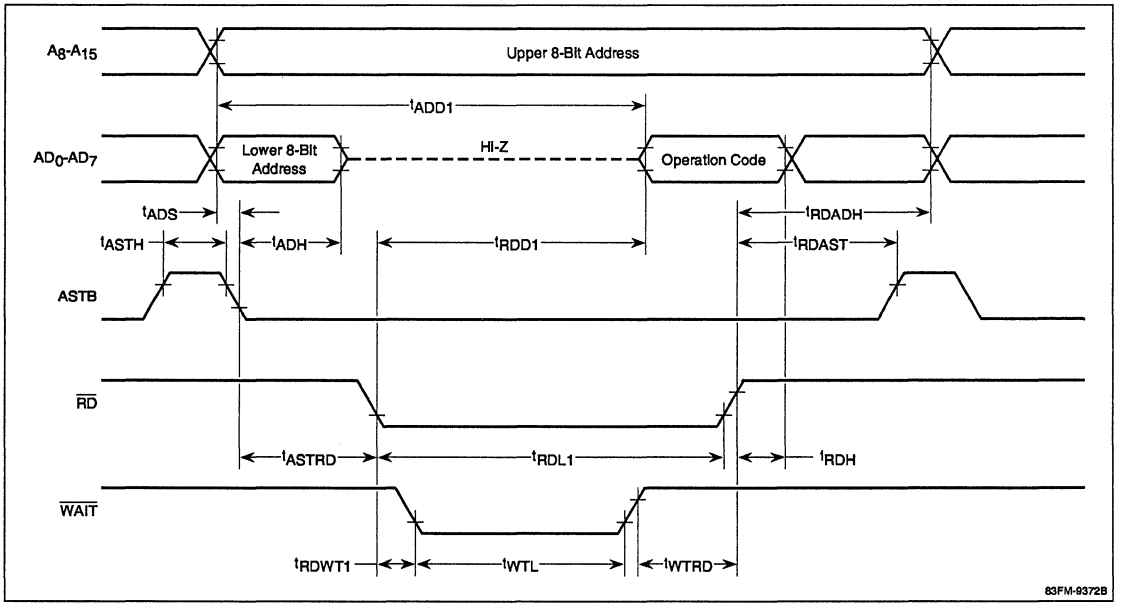


Figure 28. Read Operation; External Fetch (Wait Insertion)



3a

Figure 29. Read/Write Operation; External Data Access (No Wait)

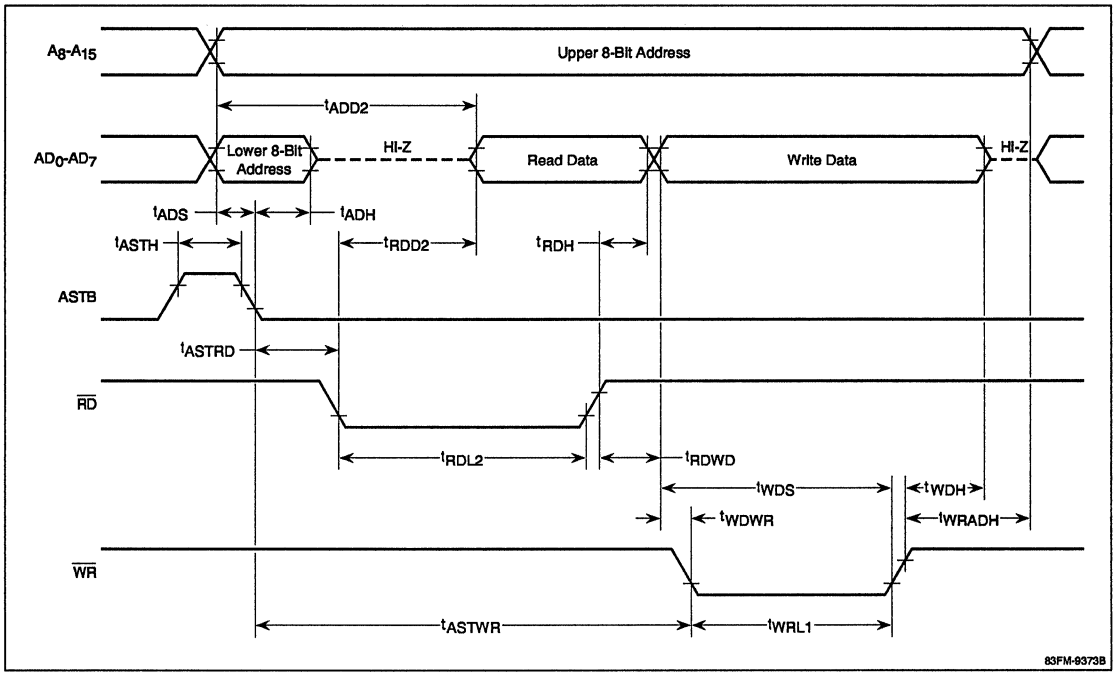
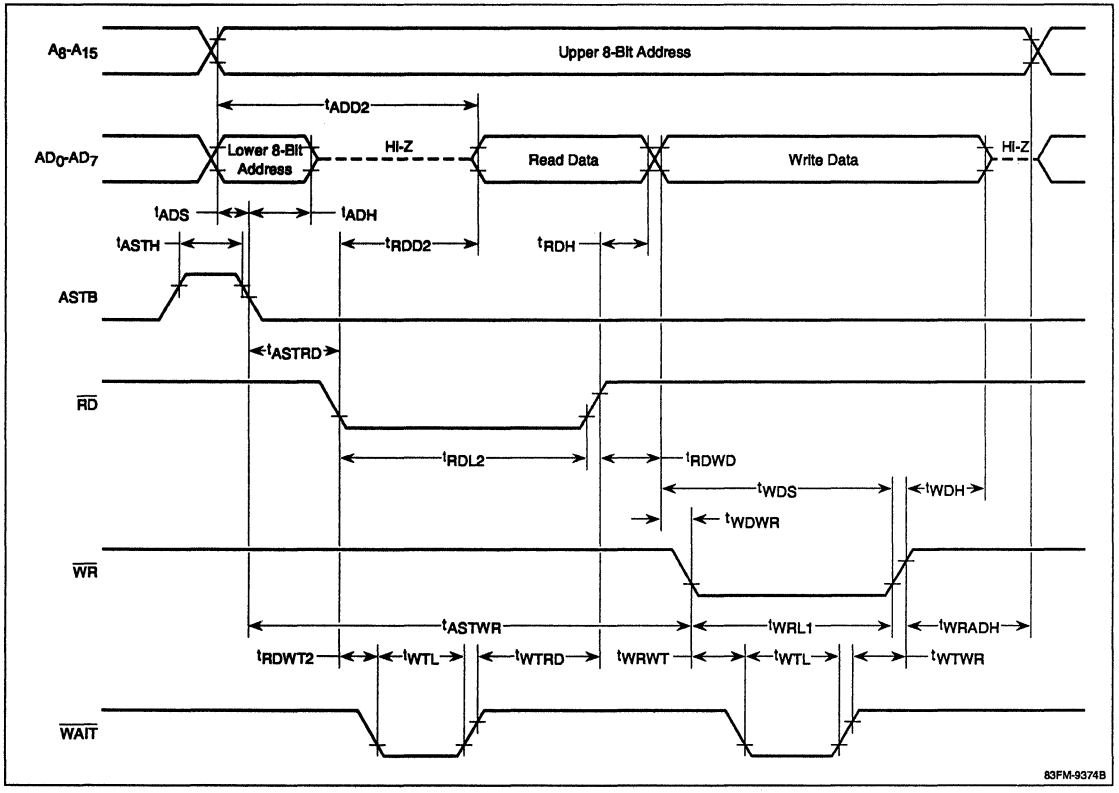


Figure 30. Read/Write Operation; External Data Access (Wait Insertion)



3a

Serial Interface, 3-Wire, I/O Mode; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 31

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY1}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH1}, t_{KL1}	$t_{KCY1}/2-50$			ns	V_{DD} = 4.5 to 6.0 V
		$t_{KCY1}/2-150$			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK1}	100			ns	
SI hold time from $\overline{SCK}$ ↑	t_{KSI1}	400			ns	
SO output delay time from $\overline{SCK}$ ↓	t_{KSO1}			300	ns	V_{DD} = 4.5 to 6.0 V; C = 100 pF (Note 1)
				1000	ns	C = 100 pF (Note 1)

Note 1: C is the load capacitance of the SO output line.

Serial Interface, 3-Wire I/O Mode; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 31

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY2}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH2}, t_{KL2}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK2}	100			ns	
SI hold time from $\overline{SCK}$ ↑	t_{KSI2}	400			ns	
SO output delay time from $\overline{SCK}$ ↓	t_{KSO2}			300	ns	V_{DD} = 4.5 to 6.0 V; C = 100 pF (Note 1)
				1000	ns	C = 100 pF (Note 1)

Note 1: C is the load capacitance of the SO output line.

Serial Interface, SBI Mode; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 32

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY3}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH3}, t_{KL3}	$t_{KCY3}/2-50$			ns	V_{DD} = 4.5 to 6.0 V
		$t_{KCY3}/2-150$			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK3}	100			ns	V_{DD} = 4.5 to 6.0 V
		300			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI3}	$t_{KCY3}/2$			ns	
SB0,SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO3}	0		250	ns	V_{DD} = 4.5 to 6.0 V; R = 1 kΩ, C = 100 pF (Note 1)
		0		1000	ns	R = 1 kΩ, C = 100 pF (Note 1)
SB0, SB1 ↓ from $\overline{SCK}$ ↑	t_{KSB}	t_{KCY3}			ns	
$\overline{SCK}$ ↓ from SB0, SB1 ↓	t_{SBK}	t_{KCY3}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY3}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY3}			ns	

Note 1: R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

Serial Interface, SBI Mode; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 32

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY4}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH4}, t_{KL4}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK4}	100			ns	V_{DD} = 4.5 to 6.0 V
		300			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI4}	t_{KCY4}/2			ns	
SB0, SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO4}	0		300	ns	V_{DD} = 4.5 to 6.0 V; R = 1 kΩ, C = 100 pF (Note 1)
		0		1000	ns	R = 1 kΩ, C = 100 pF (Note 1)
SB0, SB1 ↓ from $\overline{SCK}$ ↑	t_{KSB}	t_{KCY4}			ns	
$\overline{SCK}$ ↓ from SB0, SB1 ↓	t_{SBK}	t_{KCY4}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY4}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY4}			ns	

Note 1: R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

Serial Interface, 2-Wire, I/O Mode; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 33

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY5}	1600			ns	V_{DD} = 4.5 to 6.0 V
		3800			ns	
$\overline{SCK}$ high-level width	t_{KH5}	t_{KCY5}/2 − 50			ns	
$\overline{SCK}$ low-level width	t_{KL5}	t_{KCY5}/2 − 50			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK5}	300			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI5}	600			ns	
SB0, SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO5}	0		250	ns	V_{DD} = 4.5 to 6.0 V R = 1 kΩ, C = 100 pF (Note 1)
		0		1000	ns	R = 1 kΩ, C = 100 pF (Note 1)

Note 1: R and C are load resistance and load capacitance of the $\overline{SCK0}$, SB0, and SB1 output lines.

3a

Serial Interface, 2-Wire, I/O Mode; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 33

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY6}	1600			ns	V_{DD} = 4.5 to 6.0 V
		3800			ns	
$\overline{SCK}$ high-level width	t_{KH6}	650			ns	
$\overline{SCK}$ low-level width	t_{KL6}	800			ns	
SB0, SB1 setup time (to $\overline{SCK}$ ↑)	t_{SIK6}	100			ns	
SB0, SB1 hold time (from $\overline{SCK}$ ↑)	t_{KSI6}	$t_{KCY6}/2$			ns	
SB0, SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO6}	0		300	ns	V_{DD} = 4.5 to 6.0 V R = 1kΩ, C = 100 pF (Note 1)
		0		1000	ns	R = 1kΩ, C = 100 pF (Note 1)

Note 1: R and C are load resistance and load capacitance of the $\overline{SCK0}$, SB0, and SB1 output lines.

Figure 31. Serial Interface Timing; 3-Wire Serial I/O Mode

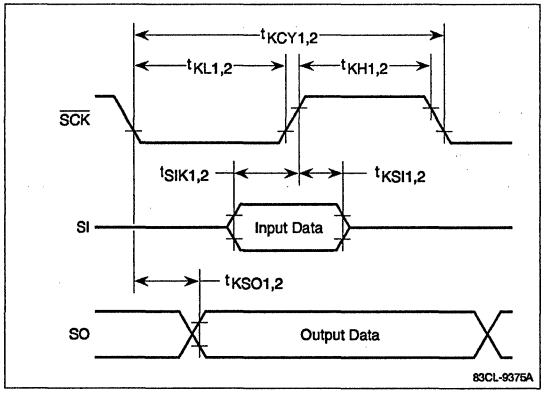
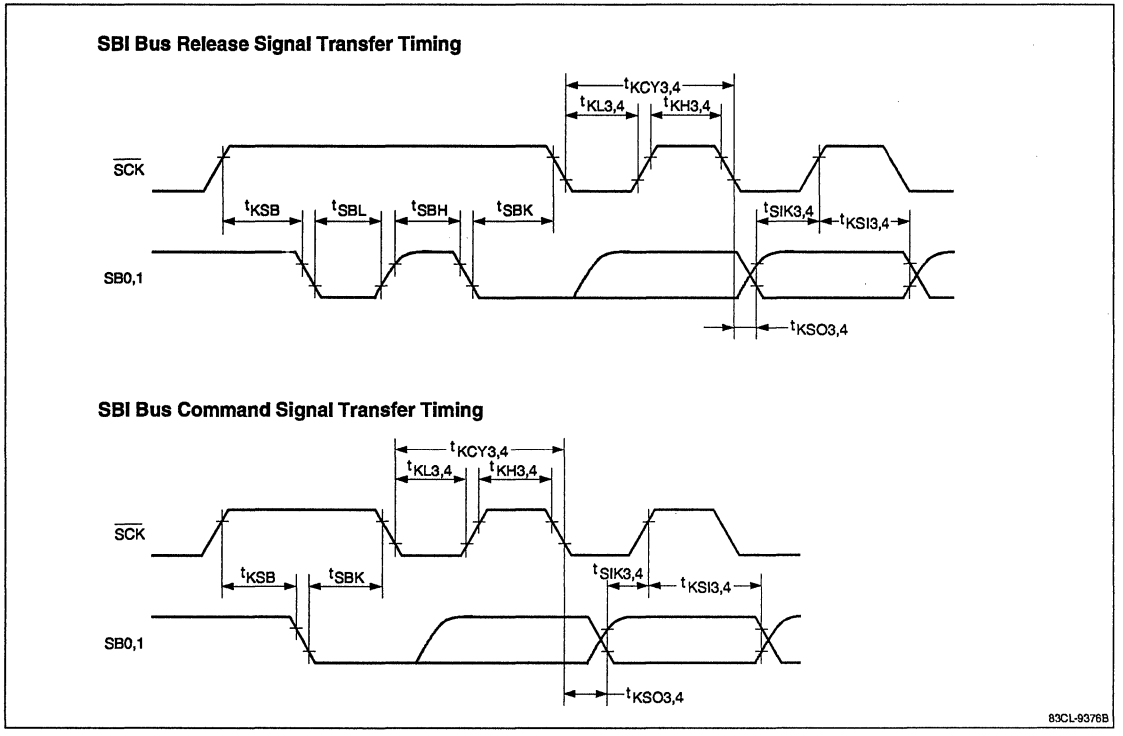
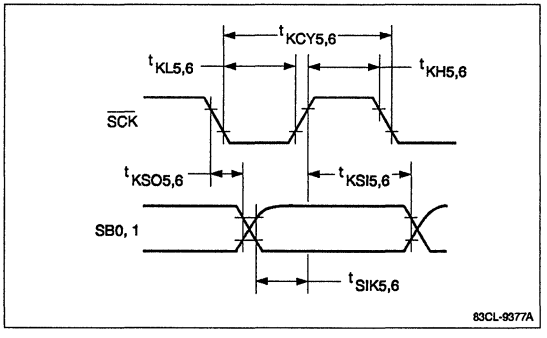


Figure 32. Serial Interface Timing; SBI Mode



3a

Figure 33. Serial Interface Timing; 2-Wire Serial I/O Mode



Data Memory STOP Mode; Low-Voltage Data Retention

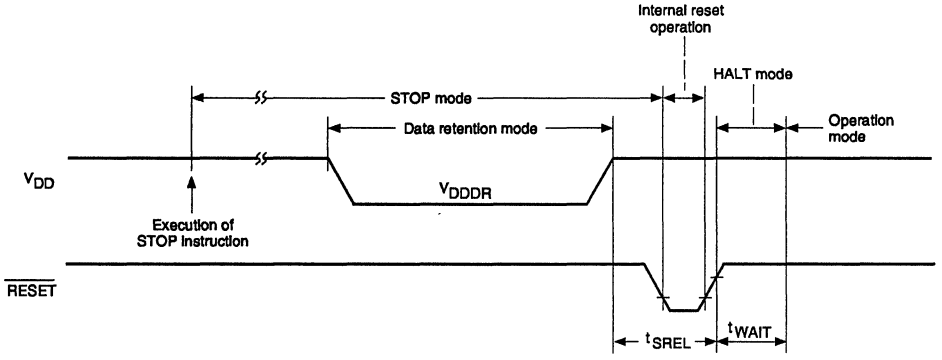
T_A = −40 to +85°C; refer to figure 34

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention supply voltage	V_{DDDR}	2.0		6.0	V	
Data retention power supply current	I_{DDDR}		0.1	10	μA	V_{DDDR} = 2.0 V; subsystem clock stop and feedback resistor disconnected
Release signal set time	t_{SREL}	0			μs	
Oscillation stabilization wait time	t_{WAIT}		$2^{18}/f_X$		ms	Release by $\overline{RESET}$
			(Note 1)		ms	Release by interrupt

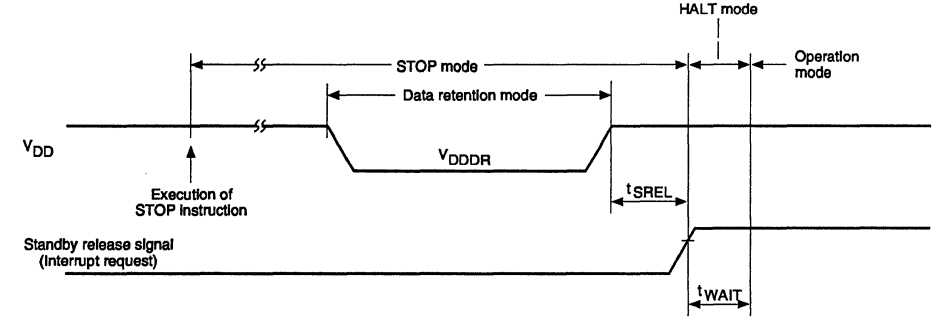
Note: $2^{13}/f_X$, $2^{15}/f_X$, $2^{16}/f_X$, $2^{17}/f_X$ or $2^{18}/f_X$ can be chosen by using bits 0 to 2 of the oscillation stabilization time select (OSTS) register.

Figure 34. Data Retention Timing

A. STOP mode is released by $\overline{\text{RESET}}$ input



B. STOP mode is released by interrupt signal



83RD-6456B

3a

NEC
NEC Electronics Inc.

μPD78002Y Family (μPD78001BY/002BY/P014Y) 8-Bit, K-Series Microcontrollers General Purpose With I2C Bus

Preliminary

September 1993

Description

The μPD78001BY, μPD78002BY, and μPD78P014Y* are members of the K-Series® of microcontrollers. The μPD78002Y family is a variation of the μPD78002 family with the addition of an I2C bus mode in serial interface 0. The μPD78P014Y is used for prototyping since the μPD78014Y family is pin and function compatible with, and the features are a superset of, the μPD78002Y family. The μPD78002Y features include bit manipulation instructions, four banks of main registers, an advanced interrupt handling facility, and a powerful set of memory-mapped on-chip peripherals. On-board data memory includes 256, 384, or 1024 bytes of internal high-speed RAM. Program memory options include 8K or 16K bytes of mask ROM, or 32K bytes of internal UV EPROM or one-time programmable (OTP) ROM.

The μPD78002Y family operates over a wide voltage range: 2.7 to 6.0 volts. Timing is generated by two built-in oscillators. A main oscillator normally drives the CPU and most peripherals and at 10 MHz provides a minimum instruction time of 0.4 μs. A subsystem oscillator at 32.768 kHz provides time keeping, and optionally a slow clock for the CPU. Since CMOS power dissipation is directly proportional to clock rate, the μPD78002Y family provides a software variable CPU clock. The HALT and STOP modes are two additional power saving features that turn off parts of the microcontroller for additional power saving. A data retention mode permits RAM contents to be saved down to 2 volts.

The range of peripherals, including timers and a serial port, makes these devices ideal for applications in portable battery-powered equipment, office automation, communications, consumer electronics, home appliances, and fitness equipment.

* See the μPD78014Y family data sheet for the μPD78P014Y electrical and functional specifications.

K-Series is a registered trademark of NEC Electronics, Inc.

Purchase of NEC I2C components conveys a license under the Philips I2C Patents Right to use these components in an I2C system, provided that the system conforms to the I2C specification as defined by Philips.

Features

- One-channel serial communication interface
  - 8-bit clock synchronous interface 0
  - I2C bus mode
  - Full-duplex, three-wire mode
  - NEC serial bus interface (SBI) mode
  - Half-duplex, two-wire mode
- Timers
  - Watchdog timer
  - Two 8-bit timer/event counters usable as one 16-bit timer/event counter
  - Clock (watch) timer (time of day tick from either oscillator)
- 53 I/O lines
  - Two CMOS input-only lines
  - 47 CMOS I/O lines
  - Four n-channel, open-drain I/O lines at 15 V maximum
- I/O port pullup resistors
  - Software controllable on 47 lines
  - Mask option on four lines on ROM versions
- Program memory
  - μPD78001BY: 8K bytes ROM
  - μPD78002BY: 16K bytes ROM
  - μPD78P014Y: 32K bytes EPROM/OTP
- Internal high-speed data memory
  - μPD78001BY: 256 bytes RAM
  - μPD78002BY: 384 bytes RAM
  - μPD78P014Y: 1024 bytes RAM
- External memory expansion
  - 64K byte total memory space
- Powerful instruction set
  - 16-bit arithmetic and data transfer instructions
  - 1-bit and 8-bit logic instructions
- Minimum instruction execution times:
  - 0.4/0.8/1.6/3.2/6.4 μs program selectable using 10-MHz main system clock
  - 122 μs selectable using 32.768-kHz subsystem clock
- Memory-mapped on-chip peripherals (special function registers)
- Programmable priority, vectored-interrupt controller (two levels)

Features (cont)

- Buzzer and clock outputs
- Power saving and battery operation features
  - Variable CPU clock rate
  - HALT mode
  - STOP mode
  - 2-V data retention mode
- CMOS operation; V_{DD} from 2.7 to 6.0 V

Ordering Information

Part Number	ROM	Package (Package Dwg.)
μPD78001BYCW-xxx	8K mask ROM	64-pin plastic shrink DIP (P64C-70-750 A, C)
μPD78002BYCW-xxx	16K mask ROM	
μPD78001BYGC-xxx-AB8	8K mask ROM	64-pin plastic QFP (P64GC-80-AB8-2)
μPD78002BYGC-xxx-AB8	16K mask ROM	
μPD78P014YGC-AB8 (Note 3)	32K OTP ROM	
μPD78P014YDW (Note 3)	32K UV EPROM	64-pin ceramic shrink DIP w/window (P64DW-70-750A)

Notes:

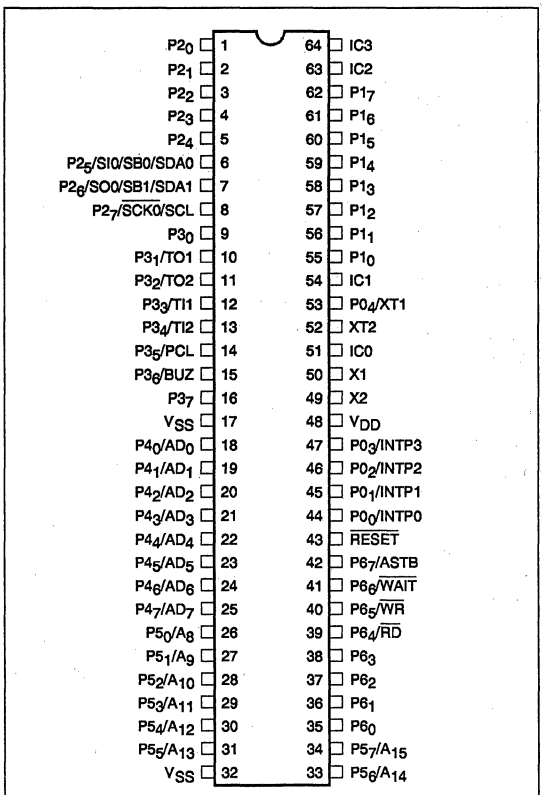
(1) xxx indicates ROM code suffix

(2) All devices listed are standard quality grade

(3) See the μPD78014Y family data sheet for the μPD78P014Y electrical and functional specifications

Pin Configurations

64-Pin Plastic Shrink DIP



Notes:

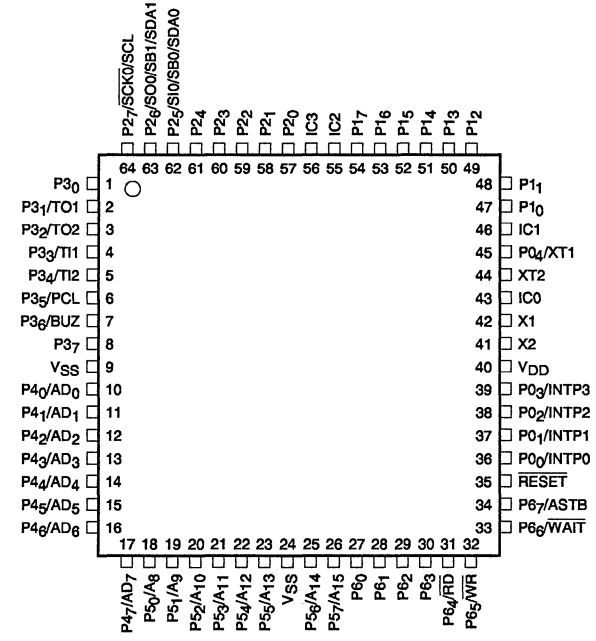
(1) Connect IC0, IC1, and IC3 (internally connected) pins to V_{SS}.

(2) Connect IC2 to V_{DD}.

83YL-9490A

Pin Configurations (cont)

64-Pin Plastic QFP



Notes:

(1) Connect IC0, IC1, and IC3 (internally connected) pins to V_{SS}.
(2) Connect IC2 to V_{DD}.

83YL-9491B

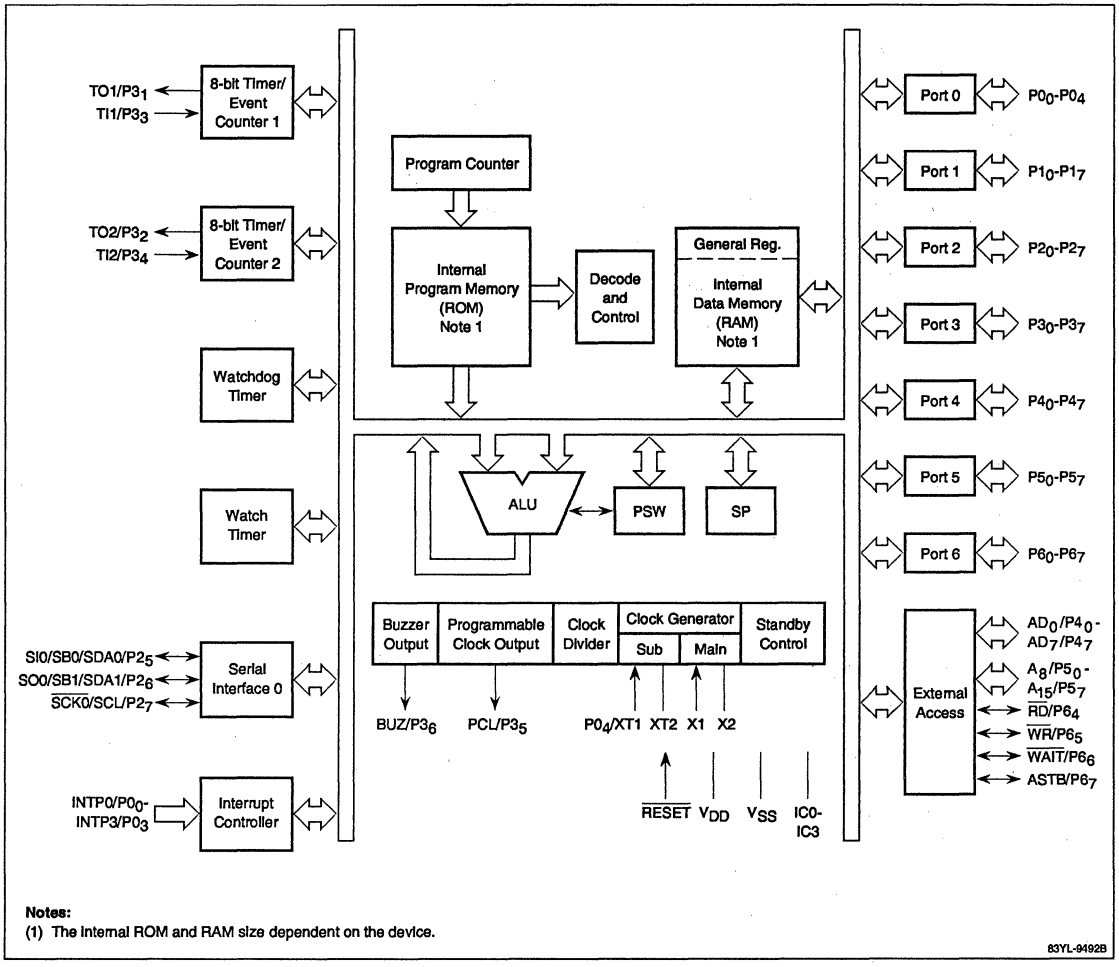
3b

Pin Functions; Normal Operating Mode

Symbol	First Function	Symbol	Alternate Function
$P0_0$	Port 0; 5-bit, bit selectable I/O port (Bits 0 and 4 are input only)	INTP0	External maskable interrupt
$P0_1$		INTP1	
$P0_2$		INTP2	
$P0_3$		INTP3	
$P0_4$		XT1	Crystal oscillator or external clock input for subsystem clock
$P1_0$ - $P1_7$	Port 1; 8-bit, bit-selectable I/O port	—	
$P2_0$ - $P2_4$	Port 2; 8-bit, bit-selectable I/O port	—	
$P2_5$		SIO SB0 SDA0	Serial data input 3-wire serial I/O mode 2/3-wire serial I/O mode Serial data bus 0 for I^2C bus mode
$P2_6$		SO0 SB1 SDA1	Serial data output 3-wire serial I/O mode 2/3-wire serial I/O mode Serial data bus 1 for I^2C bus mode
$P2_7$		$\overline{SCK0}$ SCL	Serial clock I/O for serial interface 0 Serial clock I/O for I^2C bus mode
$P3_0$	Port 3; 8-bit, bit-selectable I/O port	—	
$P3_1$		TO1	Timer output from timer 1
$P3_2$		TO2	Timer output from timer 2
$P3_3$		TI1	External count clock input to timer 1
$P3_4$		TI2	External count clock input to timer 2
$P3_5$		PCL	Programmable clock output
$P3_6$		BUZ	Programmable buzzer output
$P3_7$		—	
$P4_0$ - $P4_7$	Port 4; 8-bit I/O port	AD_0 - AD_7	Low-order 8-bit multiplexed address/data bus for external memory
$P5_0$ - $P5_7$	Port 5; 8-bit, bit selectable I/O port	A_8 - A_{15}	High-order 8-bit address bus for external memory
$P6_0$ - $P6_3$	Port 6; 8-bit, bit selectable ($P6_0$ to $P6_3$ n-channel open-drain I/O with mask option pullup resistors; $P6_4$ - $P6_7$ I/O). See note.	—	
$P6_4$		$\overline{RD}$	External memory read strobe
$P6_5$		$\overline{WR}$	External memory write strobe
$P6_6$		$\overline{WAIT}$	External memory wait signal input
$P6_7$		ASTB	Address strobe used to latch address for external memory
$\overline{RESET}$	External system reset input		
X1	Crystal/ceramic resonator connection or external clock input for main system clock		
X2	Crystal/ceramic resonator connection or inverse of external clock for main system clock		
XT2	Crystal oscillator or left open when using external clock for subsystem clock		
V_{DD}	Power supply input		
V_{SS}	Power supply ground		
IC0 to IC3	Internal connection		

Note: See table 3 and figure 4 for details

Block Diagram



Notes:
(1) The internal ROM and RAM size dependent on the device.

83YL-9492B

3b

μPD78002Y and μPD78014Y Family Differences

The μPD78002Y family is pin compatible with the μPD78014Y family and shares the same programmable device, the μPD78P014Y. The μPD78002Y family offers a reduced set of features. Table 1 lists only the differences between the two families. All other features not listed are identical for both families.

Table 1. Differences Between μPD78002Y and μPD78014Y Families

Item	μPD78002Y Family	μPD78014Y Family
Maximum internal ROM	16K bytes	32K bytes
Maximum internal high-speed RAM	384 bytes	1024 bytes
Buffer RAM	None	32 bytes
Multiply/divide instructions	None	Available
16-bit timer/event counter	None	One
Serial interface 1 (3-wire and 3-wire with automatic transmit/receive)	None	One
Vectored internal interrupts	7	10
A/D converter	None	8-bit, 8 channels

FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processing unit (CPU) of the μPD78002Y family features 8- and 16-bit arithmetic.

A CALLT vector table and a CALLF area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction can access up to 32 subroutines through their addresses contained in the CALLT vector table (40H to 7FH). A 2-byte call instruction can access any routine beginning in a specific CALLF area (0800H to 0FFFH).

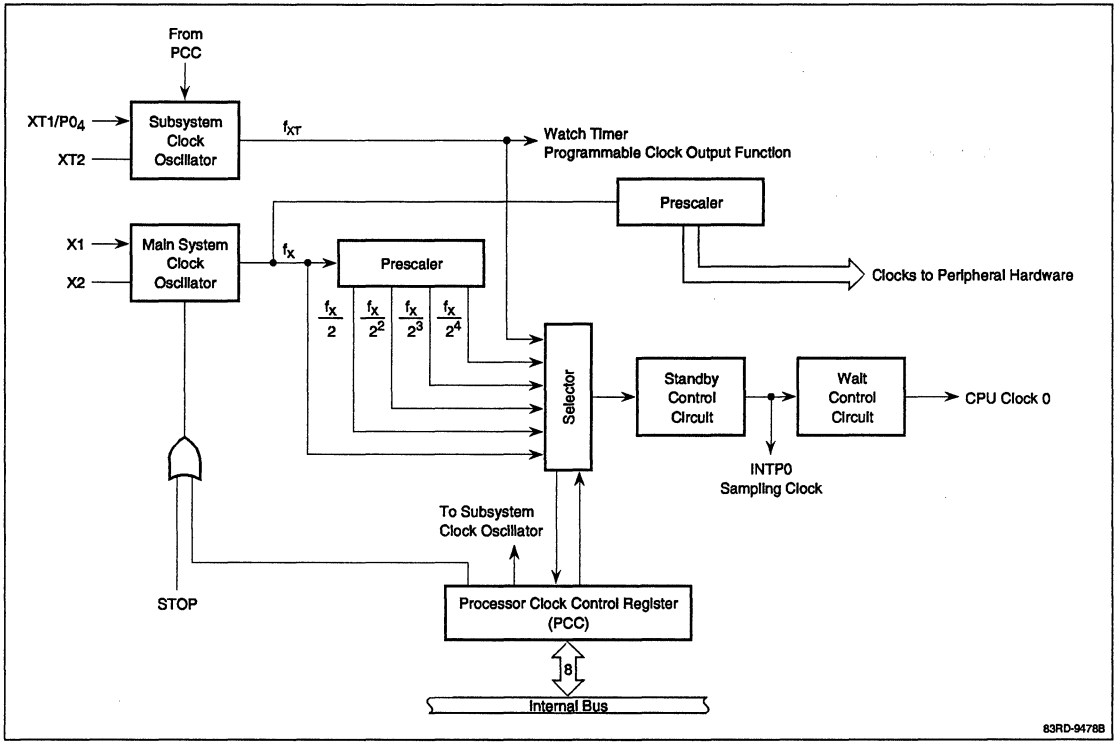
Internal System Clock Generator

The internal system clocks of the μPD78002Y family are derived from either the main system or the subsystem oscillator. See figure 1. The clocks for the watch timer and programmable clock output are derived from either the subsystem clock (f_{XT}) or the main system clock. The clocks for all other peripheral hardware are derived from the main system clock.

The CPU clock (ϕ) can be supplied from either the main system clock (f_X) or the subsystem clock (f_{XT}). Using the processor clock control register (PCC), a CPU clock frequency equal to f_X, $f_X/2$, $f_X/4$, $f_X/8$, $f_X/16$ or the subsystem clock f_{XT} can be selected. The CPU clock selected should be based on the power supply voltage available and the desired power consumption. On power up, the CPU clock defaults to the lowest speed from the main system clock and can be changed while the microcontroller is running.

Since the shortest instruction takes four CPU clocks to execute, the fastest minimum instruction execution time (t_{CY}) of 0.4 μs is achieved when using a main system clock at 10 MHz (V_{DD} equals 4.5 to 6.0 V). However, if the clock timer must generate an interrupt every 0.5 or 0.25 seconds, t_{CY} is 0.48 μs at 8.38 MHz. The fastest minimum instruction execution time available across the full voltage range of 2.7 to 6.0 V is 0.96 μs when using a main system clock of 8.38 MHz. For the lowest power consumption, the CPU can be operated from the subsystem clock and the minimum instruction execution time is 122 μs at 32.768 kHz.

Figure 1. Internal System Clock Generator

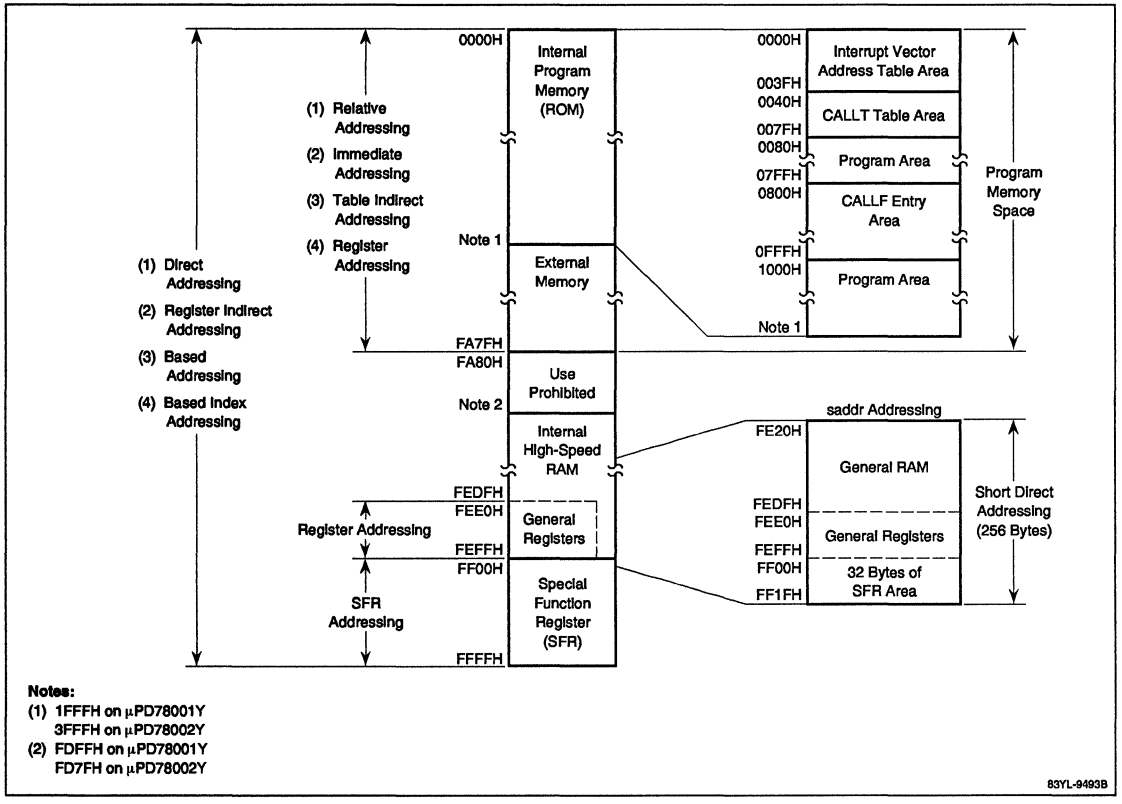


3b

Memory Space

The μPD78002Y family has a 64K-byte address space. Some of this address space (0000H-FFFFH) can be used as both program and data memory as shown in figure 2.

Figure 2. Memory Map



Notes:
(1) 1FFFH on μPD78001Y
3FFFH on μPD78002Y
(2) FDFFH on μPD78001Y
FD7FH on μPD78002Y

83YL-9493B

Internal Program Memory

All devices in the μPD78002Y family have internal program memory. The μPD78001BY contains 8K bytes while the μPD78002BY contains 16K bytes of internal ROM. The μPD78P014Y contains 32K bytes of UV EPROM or one time programmable ROM. To allow the μPD78P014Y to emulate the mask ROM devices, the amount of internal program memory available in the μPD78P014Y can be selected using the memory size switching register (IMS).

Internal RAM

The μPD78001BY contains 256 bytes (FE00H to FEFFH) while the μPD78002BY contains 384 bytes (FD80H to FEFFH) of high-speed Internal RAM. The high-speed Internal RAM contains the general register banks and the stack. The remainder of the high-speed Internal RAM and any unused register bank locations are available for general storage.

To allow the μPD78P014Y to emulate the mask ROM devices, the amount of high-speed Internal RAM in the μPD78P014Y can also be selected using the IMS.

External Memory

The μPD78002Y family can access 0, 256, 4K, 16K or all available bytes of external memory. The μPD78002Y family has an 8-bit wide external data bus and a 16-bit wide external address bus. The low-order 8 bits of the address bus are multiplexed and also provide the 8-bit data bus and are supplied by port 4. The high-order address bits of the 16-bit address bus are taken from port 5 as required. The address latch, read, and write strobes, and the external $\overline{WAIT}$ signal are supplied by port 6.

The memory expansion mode register (MM) controls the size of external memory. It can be programmed to use 0, 4, 6, or 8 bits from port 5 for the high-order address. Any remaining port 5 bits can be used for I/O. The MM register also can be used to specify one additional wait state or the use of the external $\overline{WAIT}$ signal for low-speed external memory or external peripheral devices.

When only internal ROM and RAM are used and no external memory is required, ports 4, 5 and 6 are available as general purpose I/O ports.

CPU Control Registers

Program Counter. The program counter is a 16-bit binary counter register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 0001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

Program Status Word. The program status word (PSW) is an 8-bit register that contains flags that are set or reset depending on the results obtained during the execution of an instruction. This register can be written to or read from 1 bit or 8 bits at a time. The assignment of PSW bits follows.

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

CY	Carry flag
ISP	In-service (interrupt) priority flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

3b

General Registers

The general-purpose registers (figure 3) consist of four banks of registers located at addresses FEE0H to FEFFH in Internal RAM. Each bank consists of eight 8-bit general registers that can also be used in pairs to function as four 16-bit registers. Two bits in the PSW (RBS0 and RSB1) specify which of the register banks is active at any time and are set under program control.

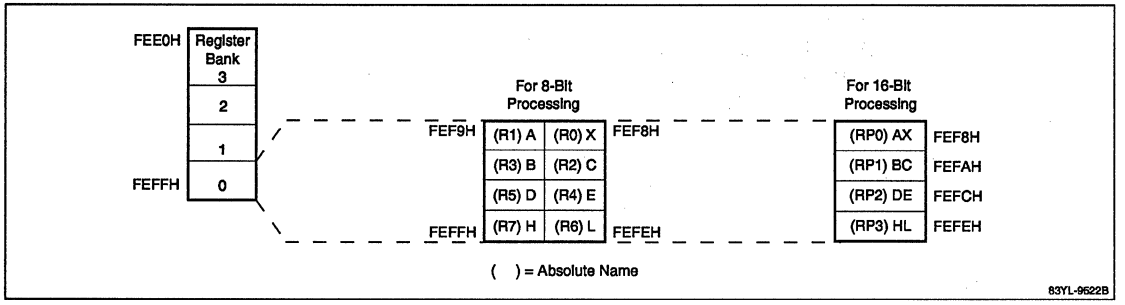
Registers have both functional names (A, X, B, C, D, E, H or L for 8-bit registers and AX, BC, DE and HL for 16-bit registers) and absolute names (R1, R0, R3, R2, R5, R4, R7, or R6 for 8-bit registers and RP0, RP1, RP2, or RP3 for 16-bit registers). Either the functional or absolute register names can be used in instructions that use the operand identifiers r and rp.

Addressing

The program memory addressing (ROM) modes provided are relative, imediate, table indirect and register addressing. The operand addressing modes provided are implied, register, direct, short direct (saddr), special-function (SFR), register indirect, based, based indexed, and stack addressing.

The 'SFR addressing' and 'saddr addressing' modes use direct addressing, and require only 1 byte in the instruction to address RAM. Normally a 65K byte ad-

Figure 3. General Registers



dress space requires 2 bytes to address it. One-byte addressing results in faster access times, since the instructions are shorter. SFR addressing addresses the entire 256-byte SFR address space from FF00H to FFFFH. Saddr addressing (see figure 2) addresses the 256 byte address space FE20H to FF1FH. FE20H to FEFFH are composed of 224 bytes of internal high speed RAM; FF00H to FF1FH contain the first 32 bytes in the special function register area.

One byte addressing is accomplished by using the first byte of the instruction for the opcode (and one operand if register A or AX is used) and the second byte of the instruction as an address (offset) into the 256-byte area. If register A or AX is used, the instructions are 2 bytes long, thereby providing fast access times. If immediate data is used, the instruction will be 3 or 4 bytes long depending upon whether the immediate data is a byte or a word. Many 16-bit SFRs are in the space FF00H to FF1FH. Using AX as an operand to these SFRs will provide fast access, since the instructions will be only 2 bytes long.

Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by SFR addressing. FF00H to FF1FH can also be accessed using saddr addressing. They are either 8 or 16 bits as required, and many of the 8-bit registers are bit addressable.

Locations FFD0H through FFDFH are known as the external access SFR area. Registers in external circuitry interfaced and mapped to these addresses can only be addressed by main memory addressing. Table 2 lists the special function registers.

Table 2. Special Function Registers

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF00H	Port 0	P0	R/W	x	x	—	00H
FF01H	Port 1	P1	R/W	x	x	—	00H
FF02H	Port 2	P2	R/W	x	x	—	00H
FF03H	Port 3	P3	R/W	x	x	—	00H
FF04H	Port 4	P4	R/W	x	x	—	Undefined
FF05H	Port 5	P5	R/W	x	x	—	Undefined
FF06H	Port 6	P6	R/W	x	x	—	Undefined
FF16H	Compare register 10	CR10	R/W	—	x	—	Undefined
FF17H	Compare register 20	CR20	R/W	—	x	—	Undefined
FF18H	8-bit timer register 1	TM1	R	x	x	—	00H
FF19H	8-bit timer register 2	TM2	R	x	x	—	00H
FF18H-FF19H	16-bit timer register	TMS	R	—	—	x	0000H
FF1AH	Serial I/O shift register 0	SIO0	R/W	—	x	—	Undefined
FF20H	Port mode register 0	PM0	R/W	x	x	—	1FH
FF21H	Port mode register 1	PM1	R/W	x	x	—	FFH
FF22H	Port mode register 2	PM2	R/W	x	x	—	FFH
FF23H	Port mode register 3	PM3	R/W	x	x	—	FFH
FF25H	Port mode register 5	PM5	R/W	x	x	—	FFH
FF26H	Port mode register 6	PM6	R/W	x	x	—	FFH
FF40H	Timer clock select register 0	TCL0	R/W	x	x	—	00H
FF41H	Timer clock select register 1	TCL1	R/W	—	x	—	00H
FF42H	Timer clock select register 2	TCL2	R/W	—	x	—	00H
FF43H	Timer clock select register 3	TCL3	R/W	—	x	—	88H
FF47H	Sampling clock select register	SCS	R/W	—	x	—	00H
FF49H	8-bit timer mode control register	TMC1	R/W	x	x	—	00H
FF4AH	Watch timer mode control register	TMC2	R/W	x	x	—	00H
FF4FH	8-bit timer output control register	TOC1	R/W	x	x	—	00H
FF60H	Serial operating mode register 0	CSIM0	R/W	x	x	—	00H
FF61H	Serial bus interface control register	SBIC	R/W	x	x	—	00H
FF62H	Slave address register	SVA	R/W	—	x	—	Undefined
FF63H	Interrupt timing specify register	SINT	R/W	x	x	—	00H
FFD0H-FFDFH	External SFR access area(Note 1)	—	R/W	x	x	—	Undefined
FFE0H	Interrupt flag register L	IF0L	R/W	x	x	—	00H
FFE1H	Interrupt flag register H	IF0H	R/W	x	x	—	00H
FFE0H-FFE1H	Interrupt flag register	IF0	R/W	—	—	x	0000H
FFE4H	Interrupt mask flag register L	MK0L	R/W	x	x	—	FFH
FFE5H	Interrupt mask flag register H	MK0H	R/W	x	x	—	FFH
FFE4H-FFE5H	Interrrupt mask flag register	MK0	R/W	—	—	x	FFFFH
FFE8H	Priority order specify flag register L	PR0L	R/W	x	x	—	FFH
FFE9H	Priority order specify flag register H	PR0H	R/W	x	x	—	FFH

Table 2. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FFE8H-FFE9H	Priority order specify flag register	PR0	R/W	—	—	x	FFFFH
FFECH	External interrupt mode register	INTM0	R/W	—	x	—	00H
FFF6H	Key return mode register	KRM	R/W	x	x	—	02H
FFF7H	Pullup resistor option register	PUO	R/W	x	x	—	00H
FFF8H	Memory expanded mode register	MM	R/W	x	x	—	10H
FFF9H	Watchdog timer mode register	WDTM	R/W	x	x	—	00H
FFFAH	Oscillation stabilization time select register	OSTS	R/W	—	x	—	04H
FFFBH	Processor clock control register	PCC	R/W	x	x	—	04H

Note: The external access area cannot be accessed using SFR addressing. It can only be accessed using main memory addressing.

Input/Output Ports

The μPD78002Y family has up to 53 port lines. Table 3 lists the features of each port and figure 4 shows the structure of each port pin.

Table 3. Digital Port Functions

Port	Operational Features	Configuration	Direct Drive Capability	Software Pullup Resistor Connection (Note 1)
Port 0 (Note 2)	5-bit input or output	Bit selectable		Byte selectable, input bits only
Port 1	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 2	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 3	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 4	8-bit input or output	Byte selectable		Byte selectable, input bits only
Port 5	8-bit input or output	Bit selectable	LED	Byte selectable, input bits only
Port 6	8-bit input or output ($P6_0$ - $P6_3$ n-channel)	Bit selectable	15 V max ($P6_0$ - $P6_3$)	Byte selectable, input bits only $P6_0$ - $P6_3$ - mask option only (Note 3) $P6_4$ - $P6_7$ - software

Notes:

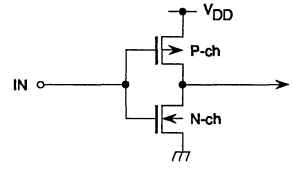
(1) Software pullup resistors can be internally connected (only on a port-by-port basis) to port bits set to input mode. Pullup resistors are not connected to port bits set to output mode.

(2) $P0_0$ and $P0_4$ are input only and do not have a software pullup resistor.

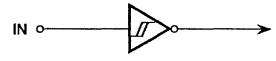
(3) All devices except μPD78P014Y.

Figure 4. Pin Input/Output Circuits

Type 1 ($P0_4$)

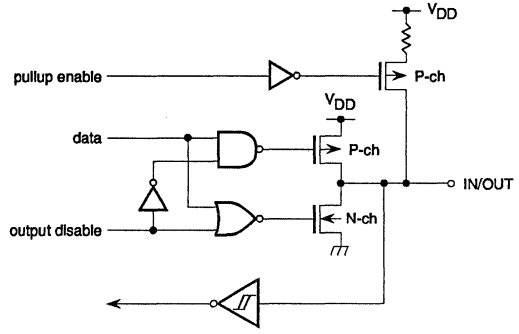


Type 2 ($P0_0$, $\overline{RESET}$)

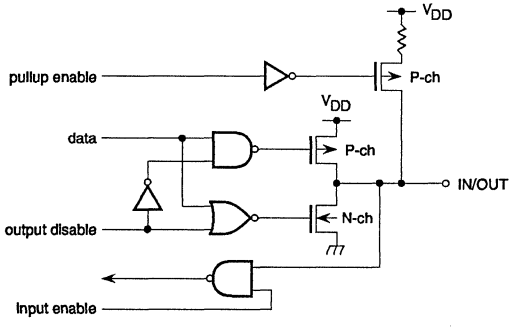


Schmitt trigger input with hysteresis characteristics

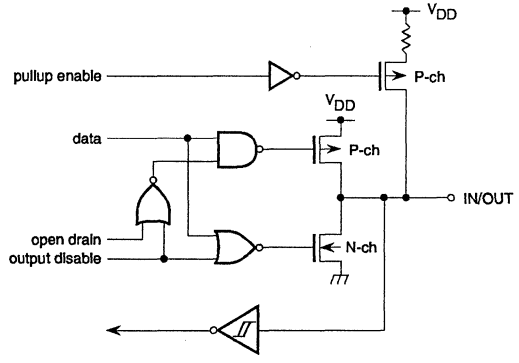
Type 8-A ($P0_1$ - $P0_3$, $P3_3$ - $P3_4$)



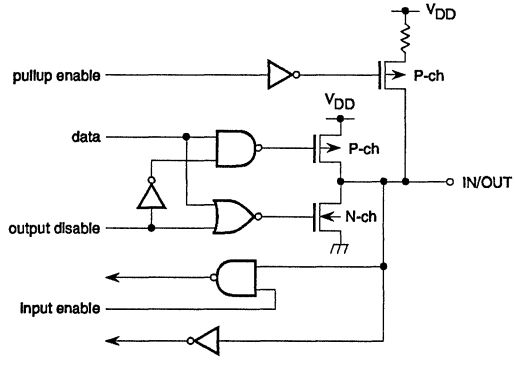
Type 5-A ($P1_0$ - $P1_7$, $P2_0$ - $P2_4$, $P3_0$ - $P3_2$, $P3_5$ - $P3_7$, $P5_0$ - $P5_7$, $P6_4$ - $P6_7$)



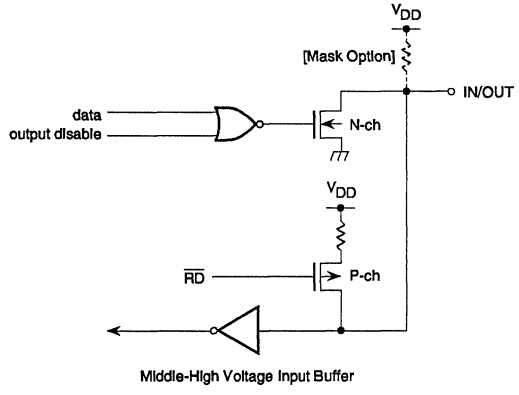
Type 10-A ($P2_5$ - $P2_7$)



Type 5-B ($P4_0$ - $P4_7$)



Type 13-B ($P6_0$ - $P6_3$)



Middle-High Voltage Input Buffer

83CL-9481B (9/93)

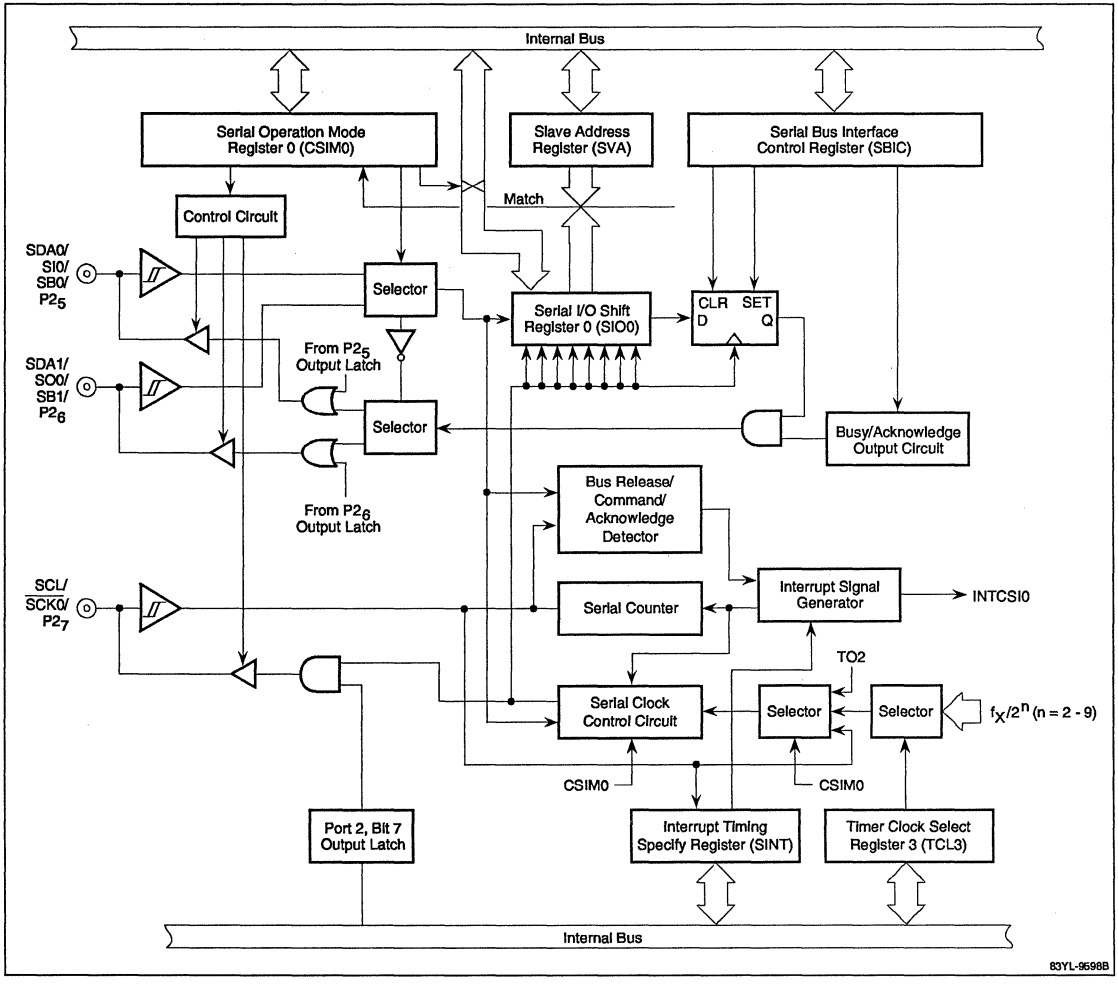
3b

Serial Interface

The μPD78002Y family has one serial interface. Serial interface 0 is an 8-bit clock synchronous serial interface (figure 5). It can be operated in either a three-wire serial I/O mode, NEC serial bus interface (SBI) mode, two-wire serial I/O mode, or I²C bus mode. The serial clock can be provided from one of eight internal clocks, the output of 8-bit timer register 2, or the external clock line $\overline{SCK0}$ (SCL for I²C bus mode).

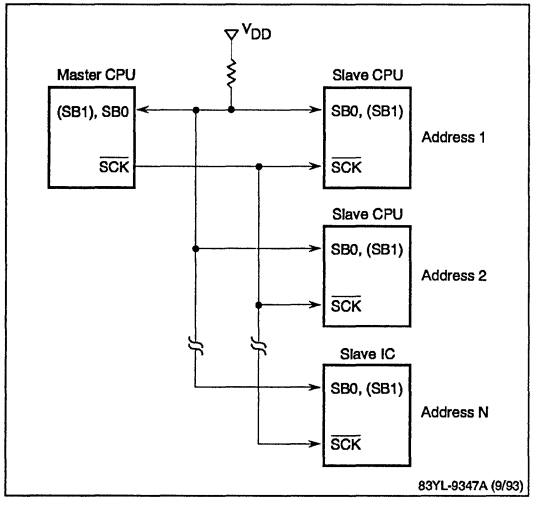
In the three-wire serial I/O mode, the 8-bit shift register (SIO0) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO0 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI0 line providing full-duplex operation. The INTCSI0 interrupt is generated after each 8-bit transfer.

Figure 5. Serial Interface 0



The NEC SBI mode is a two-wire high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are connected in a master/slave configuration (see figure 6). There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over one of the serial bus lines (SB0 or SB1) using a fixed hardware protocol synchronized with the $\overline{SCK0}$ line . Each slave device of the μPD78002Y family can be programmed to respond in hardware to any one of 256 addresses set in its slave address register (SVA). There are also 256 commands and 256 data types. Since all commands are user definable, many software protocols, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

Figure 6. SBI Mode Master/Slave Configuration



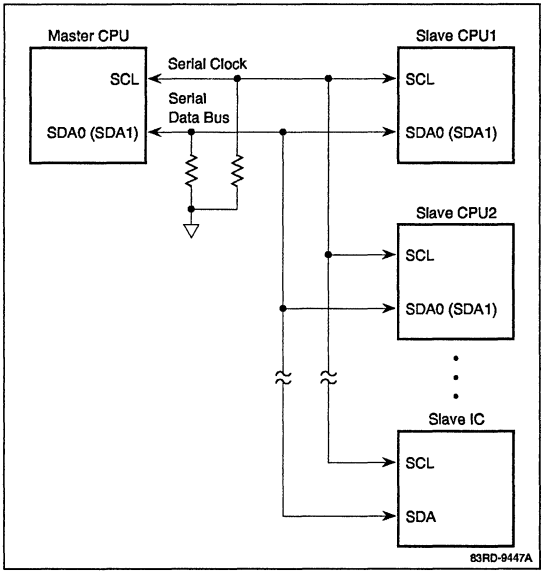
The two-wire serial I/O mode provides half-duplex operation using either the SB0 or SB1 line and the $\overline{SCK0}$ line. Communication format and handshaking can be handled in software by controlling the output levels of the data and clock lines between transfers. For data transmission, the 8-bit shift register (SIO0) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of either the SB0 or SB1 line MSB first. In addition, this byte of data is also shifted back into SIO0 on the rising edge of these pulses providing a way of verifying that the transmission was correct.

For data reception, the SIO0 register is preloaded with the value FFH. As this data value is shifted out on the falling edge of the serial clock, it disables the n-channel open-drain driver. This allows the receive data to be driven on to the serial line and shifted into the SIO0 register on the rising edge of the serial clock. The INTCSI0 interrupt is generated after each 8-bit transfer.

The I²C bus is a two-wire, high-speed serial bus developed by Philips. The I²C bus configuration has a single master and up to 128 slave devices (see figure 7). The master sends the start condition, 7-bit slave address, one bit indicating the direction of the upcoming data transfer, and the stop condition over one of the serial bus lines (SDA0 or SDA1) using a fixed hardware protocol synchronized with the serial clock line (SCL).

Each slave device of the μPD78002Y family can be programmed to respond in hardware to any one of 128 addresses set in its slave address register (SVA). Depending on the state of the transfer direction bit, either the master or the slave device places additional data on the I²C bus. The device receiving the data returns an acknowledge signal each time it receives 8 bits of data. The slave device can also notify the master device when it is busy by holding SCL low.

Figure 7. I²C Bus Master/Slave Configuration

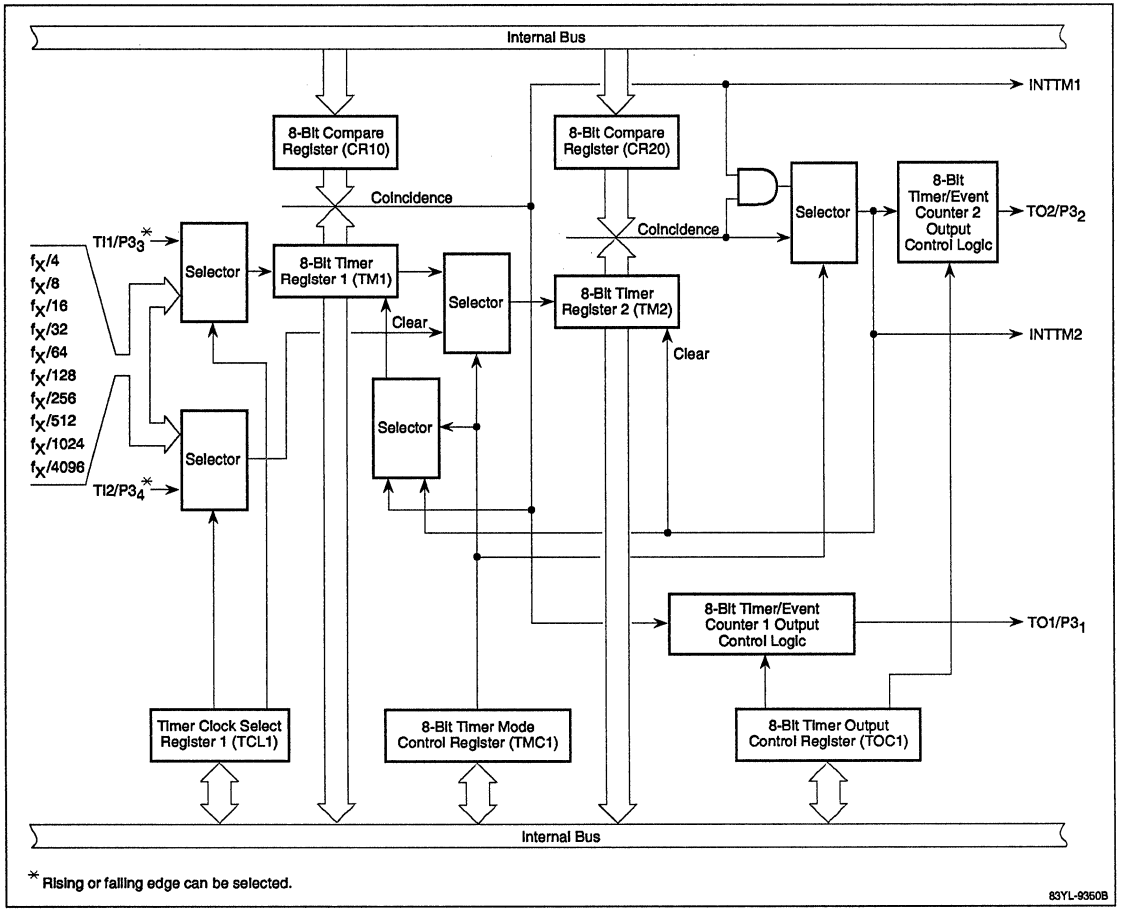


Timers

The μPD78002Y family has two 8-bit timer/event counters that can be combined for use as a 16-bit timer/event counter, a watch timer, and a watchdog timer. All of these can be programmed to count a number of prescaled values of the main system clock. In addition, the watch timer can also count the subsystem clock. The two timer/event counters can count external events.

8-Bit Timer/Event Counters 1 and 2. Timer/event counters 1 and 2 (figure 8) each consist of an 8-bit timer (TM1 or TM2), an 8-bit compare register (CR10 or CR20), and a timer output control logic (TO1 or TO2). The timers are controlled by registers TCL1, TMC1, and TOC1 via five selectors. Timer/event counters 1 and 2 can each be used as an 8-bit interval timer, to count external events on the timer input pins (TI1 or TI2), or to output a programmable square wave. In addition, timers 1 and 2 also can be combined as a 16-bit timer/event counter and used as a 16-bit interval timer, to count external events on TI1, or to output a programmable square wave on TO2.

Figure 8. 8-Bit Timer/Event Counters 1 and 2

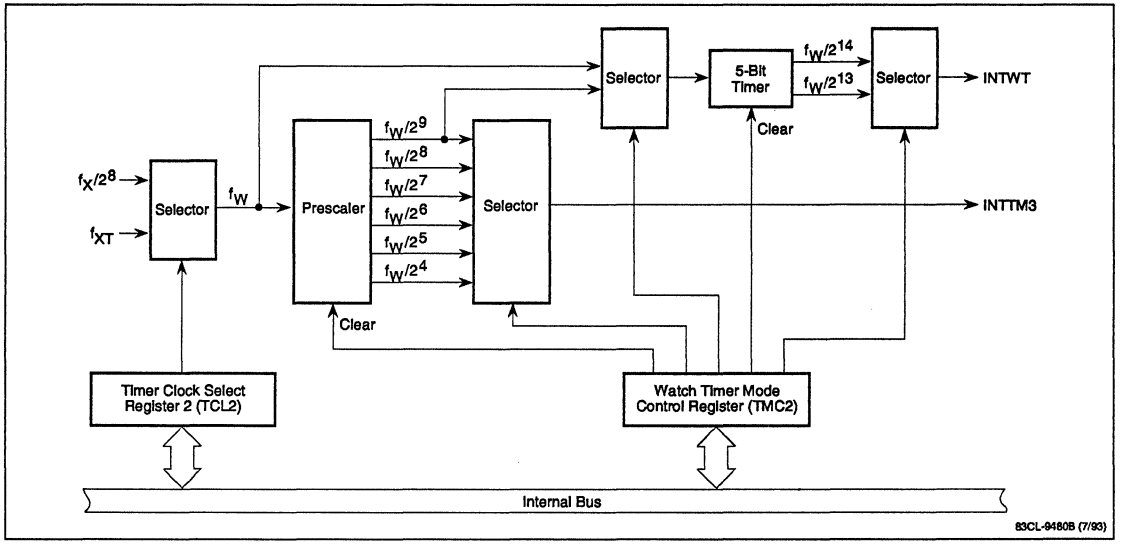


Watch Timer. The watch timer 3 (figure 9) is a 5-bit timer that can be used as a time source to keep track of time of day, to release the STOP or HALT modes at regular intervals, or to initiate any other task that must be performed at regular intervals. When driven by the subsystem clock, the watch timer continues to operate in the STOP mode.

The watch timer can function as both watch timer and an interval timer simultaneously.

When used as a watch timer, interrupt requests INTWT (not a vectored interrupt) can be generated using the main system or subsystem clock every 0.5 or 0.25 seconds. When used as an interval timer, vectored interrupt request INTTM3 is generated at preselected time intervals. With a main system clock of 8.38 MHz or a subsystem clock of 32.768 kHz, the following time intervals can be selected: 489 μs, 978 μs, 1.96 ms, 3.91 ms, 7.82 ms or 15.6 ms.

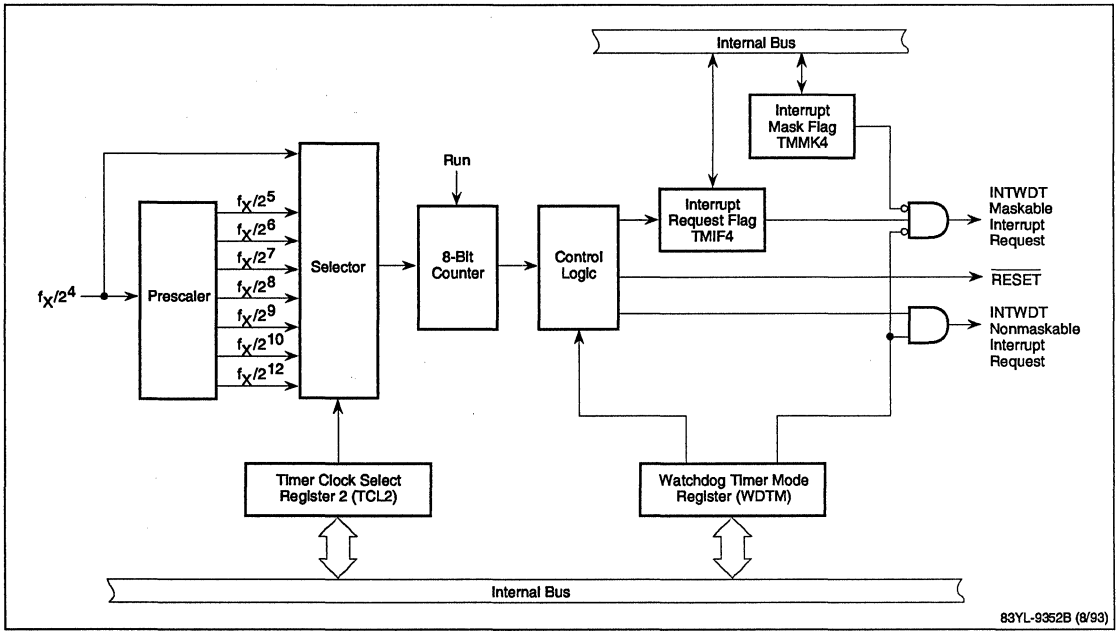
Figure 9. Watch Timer



Watchdog Timer. The watchdog timer (figure 10) can be used as either a watchdog timer or an interval timer. When used as a watchdog timer, it protects against program run-away. It can be selected to generate a nonmaskable interrupt (INTWDT), which vectors to address 0004H, or to generate an internal reset signal, which vectors to the restart address 0000H if the timer is not cleared by the program before it overflows. Eight program-selectable intervals based on the main system clock are available. With a main system clock of 8.38 MHz, the program selectable intervals are 0.489, 0.978, 1.96, 3.91, 7.82, 15.6, 31.3, and 125 ms. Once the watchdog timer is initialized and started, the timer's mode cannot be changed and the timer can only be stopped by an external reset.

When used as an interval timer, maskable interrupts (INTWDT), which vector to address 0004H, are generated repeatedly at a preset interval. The time intervals available are the same as in the watchdog timer mode.

Figure 10. Watchdog Timer



Programmable Clock Output

The μPD78002Y family has a programmable clock output (PCL) that can be used for carrier output for remote controlled transmissions or as a clock output for peripheral devices. The main system clock (f_X) divided by 8, 16, 32, 64, 128, or 256 or the subsystem clock (f_{XT}) can be output on the PCL pin. Frequencies of 1050, 524, 262, 131, 65.5 and 32.7 kHz are available with a main system clock of 8.38 MHz. See figure 11.

Buzzer Output

The μPD78002Y family also has a programmable buzzer output (BUZ). The buzzer output frequency can be programmed to be equal to the main system clock (f_X) divided by 1024, 2048, or 4096. With a main system clock of 8.38 MHz, the buzzer can be set to 8.2, 4.1 or 2.0 kHz. See figure 12.

Figure 11. Programmable Clock Output

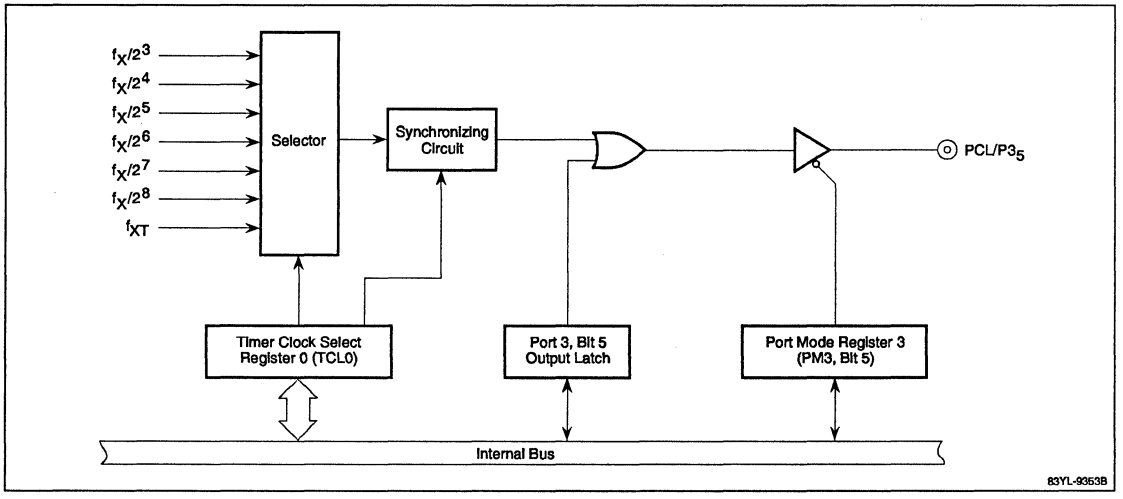
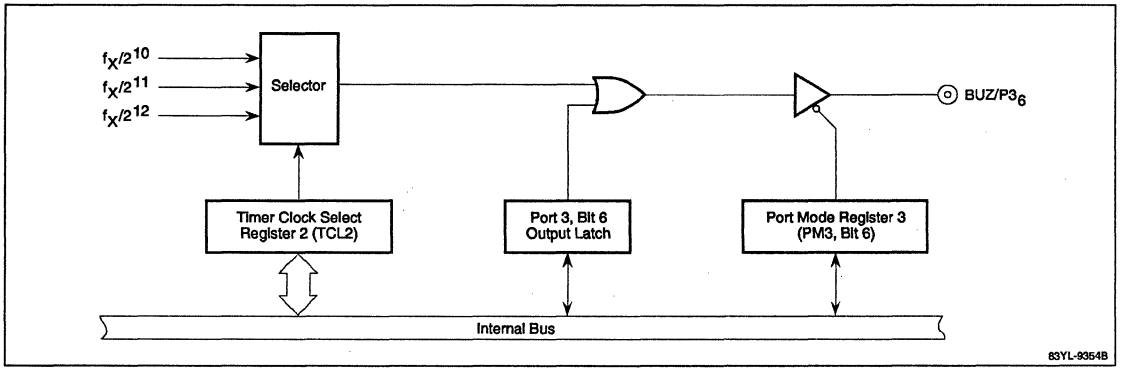


Figure 12. Buzzer Output



Interrupts

The μPD78002Y family has 11 maskable hardware interrupt sources (5 external and 6 internal). Of these 11 interrupt sources, 9 cause a vectored interrupt while the 2 testable inputs only generate an interrupt request. All of the 11 maskable interrupts can be used to release the HALT mode except INTP0. INTP0 cannot be used to release the STOP mode and cannot release the HALT mode when regiser SCS = 0. In addition, there is one nonmaskable interrupt from the watchdog timer, one software interrupt, and a RESET interrupt. The watchdog timer overflow interrupt (interrupt vector table address 0004H) can be initialized to be a nonmaskable interrupt or the highest default priority maskable interrupt. The software interrupt, generated by the BRK instruction, is not maskable. See table 4 and figure 13.

Table 4. Interrupt Sources and Vector Addresses

Type of Request	Default Priority	Signal Name	Interupt Source	Location	Vector Address	Interrupt Configuration
Restart	—	RESET	RESET Input Pin	External	0000H	—
	—	INTWDT	Watchdog timer overflow (when reset mode selected)	Internal		
Nonmaskable	—	INTWDT	Watchdog timer overflow (when nonmaskable interrupt selected)	Internal	0004H	A
Maskable	0	INTWDT	Watchdog timer overflow (when interval timer selected)	Internal	0004H	B
	1	INTP0	External interrupt edge detection	External	0006H	C
	2	INTP1	External interrupt edge detection	External	0008H	D
	3	INTP2	External interrupt edge detection	External	000AH	D
	4	INTP3	External interrupt edge detection	External	000CH	D
	5	INTCSI0	End of clocked serial interface 0 transfer	Internal	000EH	B
	6	INTTM3	Watch timer reference time interval signal	Internal	0012H	B
	7	INTTM1	8-bit timer/event counter 1 coincidence signal	Internal	0016H	B
	8	INTTM2	8-bit timer/event counter 2 coincidence signal	Internal	0018H	B
Software	—	—	BRK instruction	Internal	003EH	E
Test input	—	INTWT	Watch timer overflow	Internal	—	F
	—	INTPT4	Port 4 falling edge detection	External	—	F

Interrupt Servicing. The μPD78002Y family provides two levels of programmable hardware priority control and services all interrupt requests, except the two testable interrupts (INTWT and INTPT4). Using vectored interrupts, the programmer can choose the priority of servicing each maskable interrupt by using the interrupt control registers.

Interrupt Control Registers. The μPD78002Y family has three 16-bit interrupt control registers. The interrupt request flag register (IF0) contains an interrupt request flag for each interrupt except INTPT4. The interrupt mask register (MK0) is used to enable or disable any interrupt except INTPT4. The priority flag register (PR0) can be used to specify a high or a low priority level for each interrupt except the two testable interrupts.

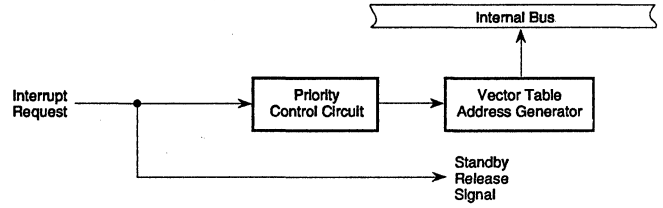
Four other 8-bit registers are associated with interrupt processing. The key return mode register (KRM) contains the KRIF interrupt request flag associated with falling-edge detection on port 4 and the KRMK mask flag used to enable or disable clearing of the standby mode if a falling edge is detected on port 4. The external interrupt mode register (INTM0) is used to select a rising, falling, or both edges as the valid edge for each of the external interrupts INTP0 to INTP2 (INTP3 is always falling edge). The sampling clock select register (SCS) is used to select a sampling clock for the noise eliminator circuit on external interrupt INTP0.

The IE and the ISP bit of the program status word are also used to control interrupts. If the IE bit is 0, all maskable interrupts are disabled. The IE bit can be set or cleared using the EI and DI instructions, respectively, or by directly writing to the PSW. The IE bit is

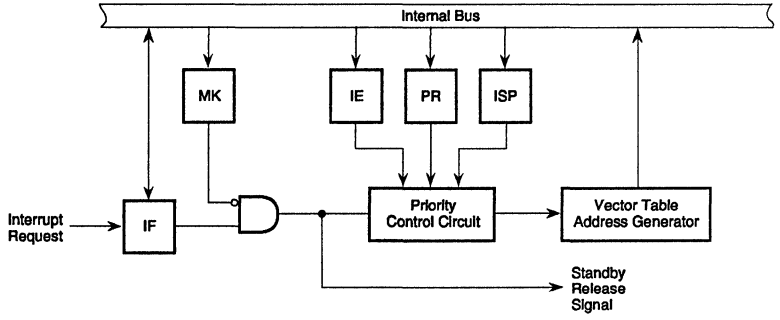
cleared each time an interrupt is accepted. The ISP bit is used by hardware to hold the priority level flag of the interrupt being serviced.

Figure 13. Interrupt Configurations

Type A: Internal nonmaskable interrupt



Type B: Internal maskable interrupt



Type C: External maskable interrupt (INTP0)

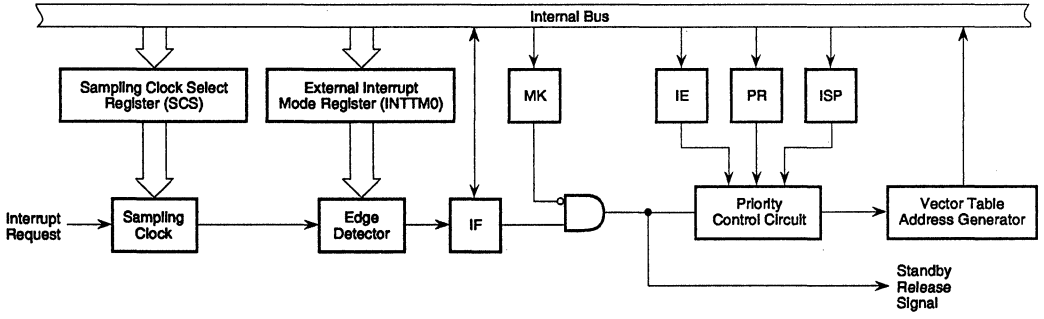
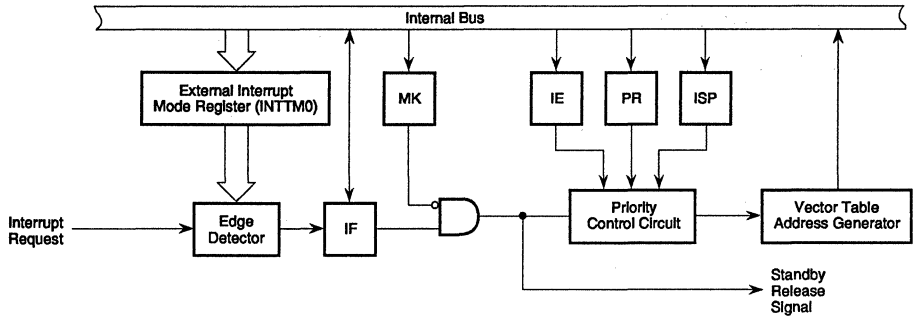
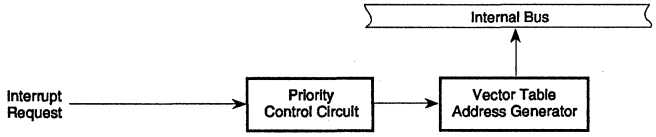


Figure 13. Interrupt Configurations (cont)

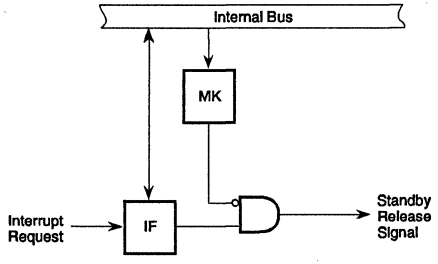
Type D: External maskable interrupt (except INTP0)



Type E: Software interrupt



Type F: Test input



Abbreviations:

IF: Interrupt request flag
IE: Interrupt enable flag
ISP: In-service priority flag
MK: Interrupt mask flag
PR: Priority specify flag

83YL-9356B (2)

Interrupt Priority. If the watchdog timer overflow interrupt (INTWDT) has been initialized to be a nonmaskable interrupt, it has priority over all other interrupts. Two hardware-controlled priority levels are available for all maskable interrupts that generate a vectored interrupt (i.e., all except the two testable interrupts). Either a high or a low priority level can be assigned by software to each of the maskable interrupts. Interrupt requests of the same priority or a priority higher than the processor's current priority level are held pending until interrupts in the current service routine are enabled by software or until one instruction has been executed after returning from the current service routine. Interrupt requests of a lower priority are always held pending until one instruction has been executed after returning from the current service routine.

The default priorities listed in table 4 are fixed by hardware and are effective only when it is necessary to choose between two interrupt requests of the same software-assigned priority. For example, the default priorities would be used after the completion of a high-priority routine, if two interrupts of the same software priority were pending.

The software interrupt, initiated by the BRK instruction, is executed regardless of the processor's priority level and the state of the IE bit. It does not alter the processor's priority level.

Vectored Interrupt Servicing. When a vectored interrupt is acknowledged, the program status word and the program counter are saved on the stack, the processor's priority is set to that specified for the interrupt, the IE bit in the PSW is set to zero, and the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (RETB instruction for the software interrupt) reverses the process and the μPD78002Y family microcontroller resumes the interrupted routine.

Standby Modes

HALT, STOP, and data retention modes are provided to reduce power consumption when CPU action is not required.

The HALT mode is entered by executing a HALT instruction while the CPU is operating from the main system or subsystem clock. In HALT mode, the CPU clock is stopped while the main system and the subsystem clock continue to run. The HALT mode is released by any unmasked interrupt request (except INTP0 if register SCS = 0), a nonmaskable interrupt request, an unmasked test input, or an external reset pulse.

Power consumption may be further reduced by using the STOP mode. The STOP mode is entered by executing a STOP instruction while operating from the main system clock. In STOP mode, the main system clock input pin X1 is internally grounded stopping both the CPU and the peripheral hardware clock. The STOP mode is released by any unmasked interrupt request except INTP0, a nonmaskable interrupt request, an unmasked test input, or an external reset pulse. Any peripheral using the main oscillator as a clock source will also be disabled in the STOP mode and interrupts from such a peripheral cannot be used to exit the STOP mode. Table 5 summarizes both the HALT and STOP standby modes.

Table 5. Standby Mode Operation Status

Item	HALT Mode	STOP Mode
Setting instruction	HALT instruction	STOP instruction
System clock when setting	Main system or subsystem clock	Main system clock
Clock oscillator	Main system and subsystem clocks can oscillate; CPU clock is stopped.	Subsystem clock can oscillate; CPU clock and main system clock are stopped.
CPU	Operation stopped	Operation stopped
Ports	Maintain previous state	Maintain previous state
8-bit timer/ event counters	Operational from main system clock	Operational only with TI1 and TI2 as count clock
Watch timer	Operational from main system clock or with f_{XT} as count clock	Operational only with f_{XT} as count clock
Watchdog timer	Operational from main system clock	Operation stopped
Serial interface 0	Operational from main system clock	Operational only with external clock
External interrupts	Operational except for INTP0 when its sampling clock is based on the CPU clock	INTP0 not operational; INTP1 to INTP3 operational

When exiting the STOP mode, a wait time occurs before the CPU begins execution to allow the main system clock oscillator circuit to stabilize. The oscillator stabilization time is selected by programming the OSTS register with one of five values before entering the STOP mode. The values range from 0.98 msec to 31.3 msec at f_x = 8.38 MHZ.

Once in the STOP mode, power consumption can be further minimized by lowering the power supply voltage V_{DD} to as little as 2 V. This places the device in the data retention mode. The contents of Internal RAM and the registers are retained. This mode is released by first raising V_{DD} to the proper operating range and then releasing the STOP mode.

External Reset

The μPD78002Y family is reset by taking the $\overline{\text{RESET}}$ pin low or by an overflow of the watchdog timer (if enabled). The $\overline{\text{RESET}}$ input pin is a schmitt-trigger input with hysteresis characteristics to protect against spurious system resets caused by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low for a minimum of 10 μs after the power supply reaches its operating voltage.

There is no functional difference between an external reset and an internal reset caused by the overflow of the watchdog timer. In both cases, the main system clock oscillation is stopped and the subsystem clock oscillation continues. During reset, the program counter is loaded with the address contained in the reset vector (addresses 0000H, 0001H). Once the reset is cleared and the oscillation stabilization time of $2^{18}/f_X$ has elapsed, program execution starts at that address.

ELECTRICAL SPECIFICATIONS

The following specifications are for the μPD78001BY/002BY devices only. Refer to the μPD78014Y data sheet for μPD78P014Y device specifications.

Absolute Maximum Ratings

T_A = +25°C

Supply voltage, V_{DD}	−0.3 to +7.0 V
Input voltage, V_{I1} (except $P6_0$ to $P6_3$)	−0.3 to V_{DD} + 0.3 V
Input voltage, V_{I2} ($P6_0$ to $P6_3$; open drain)	−0.3 to +16 V
Output voltage, V_O	−0.3 to V_{DD} + 0.3 V
Output current, high; I_{OH}	
Each output pin	−10 mA
Total: ports 1 to 3	−15 mA
Total: port 0 and ports 4 to 6	−15 mA
Output current, low, I_{OL} †	
Each output pin	30 mA peak, 15 mA rms
Total: $P4_0$ to $P4_7$ and $P5_0$ to $P5_5$	100 mA peak, 70 mA rms
Total: $P0_1$ to $P0_3$, $P5_6$, $P5_7$, and $P6_0$ to $P6_7$	100 mA peak, 70 mA rms
Total: $P0_1$ to $P0_3$ and $P6_4$ to $P6_7$	50 mA peak, 20 mA rms
Total: ports 1 to 3	50 mA peak, 20 mA rms
Operating temperature, T_{OPT}	−40 to +85°C
Storage temperature, T_{STG}	−65 to +150°C

† rms value = peak value x (duty cycle) $^{1/2}$

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC characteristics.

Capacitance

T_A = +25°C; V_{DD} = V_{SS} = 0 V

Parameter	Symbol	Max	Unit	Conditions	
Input capacitance	C_{IN}	15	pF	Except $P6_0$ to $P6_3$	f = 1 MHz; unmeasured pins returned to ground
		20	pF	$P6_0$ to $P6_3$	
Output capacitance	C_{OUT}	15	pF	Except $P6_0$ to $P6_3$	
		20	pF	$P6_0$ to $P6_3$	
Input/output capacitance	C_{IO}	15	pF	Except $P6_0$ to $P6_3$	
		20	pF	$P6_0$ to $P6_3$	

3b

Main System Clock Oscillator

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 14.

Type	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Figure 14A)	Oscillation frequency (Note 1)	f_X	1.0		10.0	MHz	V_{DD} = oscillator voltage range
	Oscillation stabilization time (Note 2)				4.0	ms	After V_{DD} reaches oscillator operating voltage
Crystal resonator (Figure 14A)	Oscillation frequency (Note 1)	f_X	1.0	8.38	10.0	MHz	
	Oscillation stabilization time (Note 2)				10	ms	V_{DD} = 4.5 to 6.0 V
					30	ms	
External clock (Figure 14B)	X1 input frequency (Note 1)	f_X	1.0		10.0	MHz	
	X1 input high/low-level width	t_{XH}, t_{XL}	50		500	ns	

Notes:

(1) Oscillator and X1 input frequencies are included only to show the oscillator characteristics. Refer to the AC Characteristics table for actual instruction execution times.

(2) Time required for the oscillator to stabilize after reset or STOP mode is released. The values shown are for the recommended resonators. Values for resonators not shown in this data sheet should be obtained from the manufacturer's specification sheets.

Subsystem Clock Oscillator

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 15.

Type	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Crystal resonator (Figure 15A)	Oscillation frequency (Note 1)	f_{XT}	32	32.768	35	kHz	
	Oscillation stabilization time (Note 2)			1.2	2	s	V_{DD} = 4.5 to 6.0 V
					10	s	
External clock (Figure 15B)	XT1 input frequency (Note 1)	f_{XT}	32		100	kHz	
	XT1 input high/low-level width	t_{XTH}, t_{XTL}	5		15	μs	

Notes:

(1) The oscillator and XT1 input frequencies are included only to show the oscillator characteristics. Refer to the AC Characteristics table for actual instruction execution times.

(2) Time required for the oscillator to stabilize after reset or STOP mode is released. The values shown are for the recommended resonators. Values for resonators not shown in this data sheet should be obtained from the manufacturer's specification sheets.

Figure 14. Main System Clock Configurations

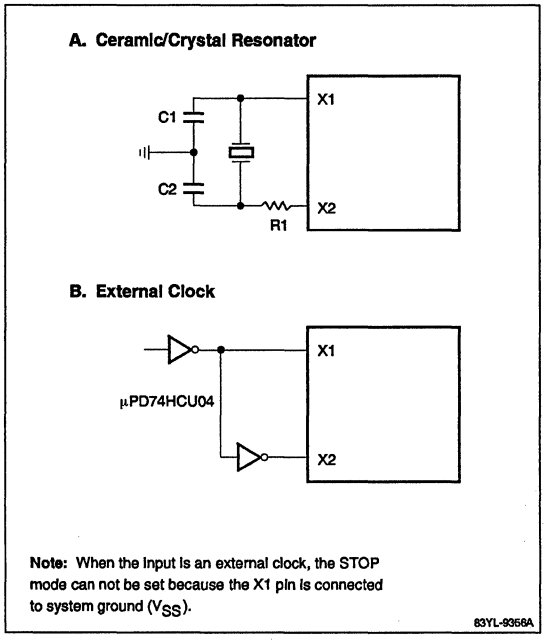
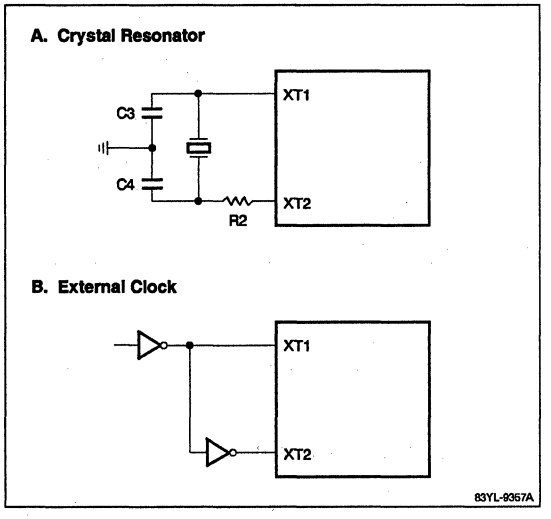


Figure 15. Subsystem Clock Configurations



Recommended Main System Clock Ceramic Resonators

T_A = −40 to +85°C; refer to figure 14A.

Part Number (Notes 1 and 2)	Recommended Circuit Constant			Oscillator Voltage Range		Frequency (MHz)
	C1 (pF)	C2 (pF)	R1 (kΩ)	Min (V)	Max (V)	
CSB1000J	100	100	6.8	2.7	6.0	1.00
CSBxxxxJ	100	100	4.7	2.7	6.0	1.01 to 1.25
CSAx.xxxMK	100	100	0	2.7	6.0	1.26 to 1.79
CSAx.xxMG	100	100	0	2.7	6.0	1.80 to 2.44
CSTx.xxMG	(Note 3)	(Note 3)	0	2.7	6.0	1.80 to 2.44
CSAx.xxMG	30	30	0	2.7	6.0	2.45 to 4.18
CSTx.xxMGW	(Note 3)	(Note 3)	0	2.7	6.0	2.45 to 4.18
CSAx.xxMG	30	30	0	2.7	6.0	4.19 to 6.00
CSTx.xxMGW	(Note 3)	(Note 3)	0	2.7	6.0	4.19 to 6.00
CSAx.xxMT	30	30	0	2.7	6.0	6.01 to 10.0
CSTx.xxMTW	(Note 3)	(Note 3)	0	2.7	6.0	6.01 to 10.0

Notes:

(1) Manufactured by Murata Mfg. Co., Ltd.

(2) x.xx indicates frequency

(3) C1 and C2 are contained in the ceramic resonators.

3b

Recommended Subsystem Clock Crystal Resonators

T_A = −40 to +60°C; refer to figure 15A.

Part Number †	Frequency (kHz)	Recommended Circuit Constant			Oscillator Voltage Range	
		C3 (pF)	C4 (pF)	R2 (kΩ)	Min (V)	Max (V)
DT-38 (1TA252 E00, load capacitance 6.3 pF)	32.768	12	12	100	2.7	6.0

† Manufactured by Daishinku

DC Characteristics

T_A = −40 to +85°C; V_{DD} = +2.7 to 6.0 V; refer to figures 16-21

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Other than below
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	$P0_0$ to $P0_4$, $P2_0$, $P2_2$, $P2_4$ to $P2_7$, $P3_3$, $P3_4$, $\overline{RESET}$
	V_{IH3}	0.7 V_{DD}		15	V	$P6_0$ to $P6_3$; open-drain
	V_{IH4}	V_{DD} − 0.5		V_{DD}	V	X1, X2
	V_{IH5}	V_{DD} − 0.5		V_{DD}	V	V_{DD} = 4.5 to 6.0 V; XT1, XT2
Low-level input voltage	V_{IL1}	0		0.3 V_{DD}	V	Other than below
	V_{IL2}	0		0.2 V_{DD}	V	$P0_0$ to $P0_4$, $P2_0$, $P2_2$, $P2_4$ to $P2_7$, $P3_3$, $P3_4$, $\overline{RESET}$
	V_{IL3}	0		0.3 V_{DD}	V	$P6_0$ to $P6_3$; V_{DD} = 4.5 to 6.0 V
		0		0.2 V_{DD}	V	$P6_0$ to $P6_3$
	V_{IL4}	0		0.4	V	X1, X2
	V_{IL5}	0		0.4	V	XT1, XT2; V_{DD} = 4.5 to 6.0 V
		0		0.3	V	XT1, XT2
High-level output voltage	V_{OH1}	V_{DD} − 1.0			V	V_{DD} = 4.5 to 6.0 V, I_{OH} = −1.0 mA
		V_{DD} − 0.5			V	I_{OH} = −100 μA
Low-level output voltage	V_{OL1}		0.4	2.0	V	$P5_0$ to $P5_7$, $P6_0$ to $P6_3$; V_{DD} = 4.5 to 6.0 V, I_{OL} = 15 mA
				0.4	V	Other than above; V_{DD} = 4.5 to 6.0 V, I_{OL} = 1.6 mA
	V_{OL2}			0.2 V_{DD}	V	SB0, SB1, $\overline{SCK0}$; V_{DD} = 4.5 to 6.0 V, open-drain, pullup resistance = 1 kΩ
	V_{OL3}			0.5	V	I_{OL} = 400 μA
High-level input leakage current	I_{LIH1}			3	μA	V_{IN} = V_{DD}; except X1, X2, XT1, XT2
	I_{LIH2}			20	μA	V_{IN} = V_{DD}; X1, X2, XT1, XT2
	I_{LIH3}			80	μA	V_{IN} = 15 V; $P6_0$ to $P6_3$
Low-level input leakage current	I_{LIL1}			−3	μA	V_{IN} = 0 V; except X1, X2, XT1, XT2
	I_{LIL2}			−20	μA	V_{IN} = 0 V; X1, X2, XT1, XT2
	I_{LIL3}			−3	μA	V_{IN} = 0 V; $P6_0$ to $P6_3$ (Note 1)
Output leakage current high	I_{LOH1}			3	μA	V_{OUT} = V_{DD}
Output leakage current low	I_{LOL}			−3	μA	V_{OUT} = 0 V
Mask option pullup resistor	R_1	20	40	90	kΩ	V_{IN} = 0 V; $P6_0$ to $P6_3$
Software pullup resistor	R_2	15	40	90	kΩ	V_{DD} = 4.5 to 6.0 V; V_{IN} = 0 V, $P0_1$ to $P0_3$, ports 1 to 5, $P6_4$ to $P6_7$
		20		500	kΩ	V_{DD} = 2.7 to 4.5 V; V_{IN} = 0 V, $P0_1$ to $P0_3$, ports 1 to 5, $P6_4$ to $P6_7$

DC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Power supply current	I_{DD1}		7.5	22.5	mA	8.38 MHz crystal oscillation operating mode; V_{DD} = 5.0 V ±10% (Note 2)
			0.8	2.4	mA	8.38 MHz crystal oscillation operating mode; V_{DD} = 3.0 V ±10% (Note 3)
	I_{DD2}		1.4	4.2	mA	8.38 MHz crystal oscillation HALT mode; V_{DD} = 5.0 V ±10%
			550	1650	μA	8.38 MHz crystal oscillation HALT mode; V_{DD} = 3.0 V ±10%
	I_{DD3}		60	120	μA	32.768 kHz crystal oscillation operating mode; V_{DD} = 5.0 V ±10%, X1 STOP mode, CPU operating from subsystem clock
			35	70	μA	32.768 kHz crystal oscillation operating mode; V_{DD} = 3.0 V ±10%, X1 STOP mode, CPU operating from subsystem clock
	I_{DD4}		25	50	μA	32.768 kHz crystal oscillation HALT mode; V_{DD} = 5.0 V ±10%, X1 STOP mode
			5	10	μA	32.768 kHz crystal oscillation HALT mode; V_{DD} = 3.0 V ±10%, X1 STOP mode
	I_{DD5}		1	20	μA	XT1 = 0 V STOP mode when feedback resistor is connected; V_{DD} = 5.0 V ±10%
			0.5	10	μA	XT1 = 0 V STOP mode when feedback resistor is connected; V_{DD} = 3.0 V ±10%
	I_{DD6}		0.1	20	μA	XT1 = 0 V STOP mode when feedback resistor is disconnected; V_{DD} = 5.0 V ±10%
			0.05	10	μA	XT1 = 0 V STOP mode when feedback resistor is disconnected; V_{DD} = 3.0 V ±10%

Notes:

(1) $P6_0$ to $P6_3$ become −200 μA (max.) for only 1 clock cycle during input instruction execution (no wait) and −3 μA (max.) during instruction other than input.

(2) When operated in the high-speed mode with the processor clock control register set to 00H.

(3) When operated in low-speed mode with the processor clock control register set to 04H.

Figure 16. I_{DD} vs V_{DD} (f_X = 8.38 MHz)

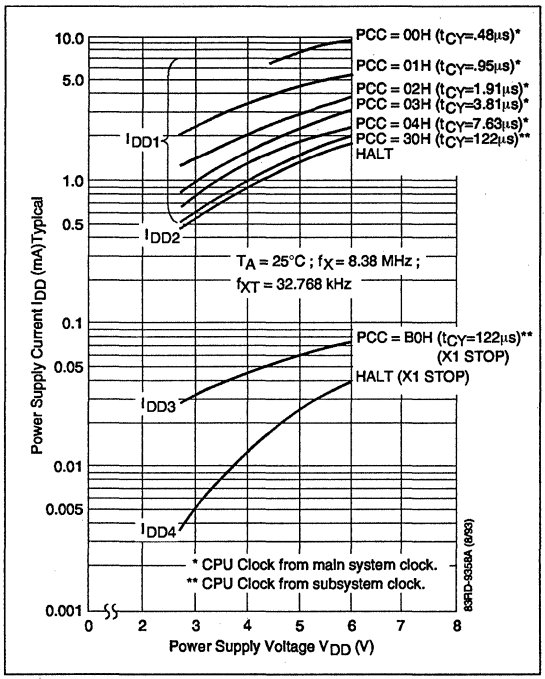


Figure 17. I_{DD} vs V_{DD} (f_X = 4.19 MHz)

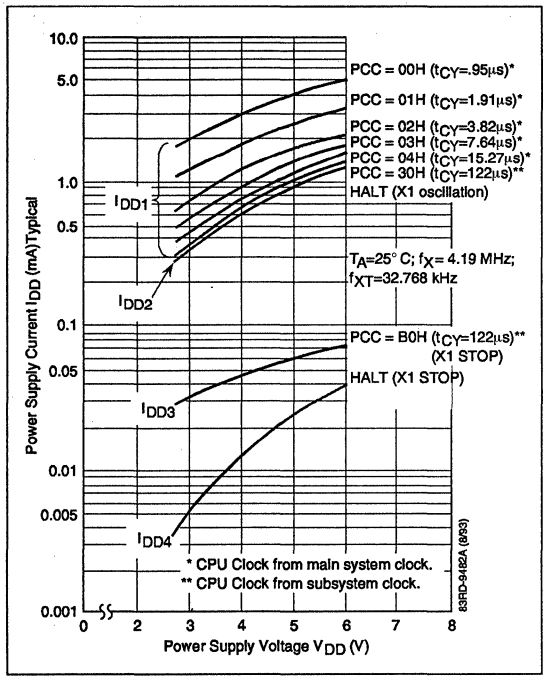


Figure 18. I_{OL} vs V_{OL} (Ports 0, 2-5, $P6_4$-$P6_7$)

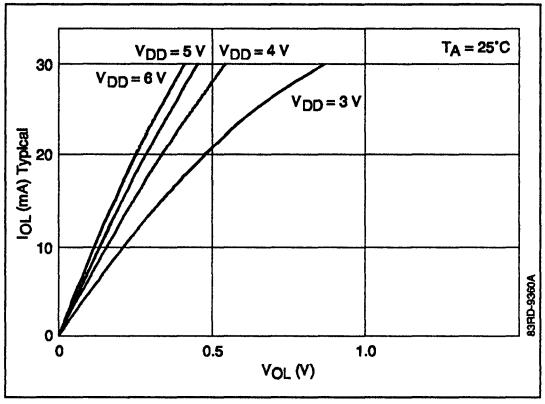


Figure 19. I_{OL} vs V_{OL} (Port 1)

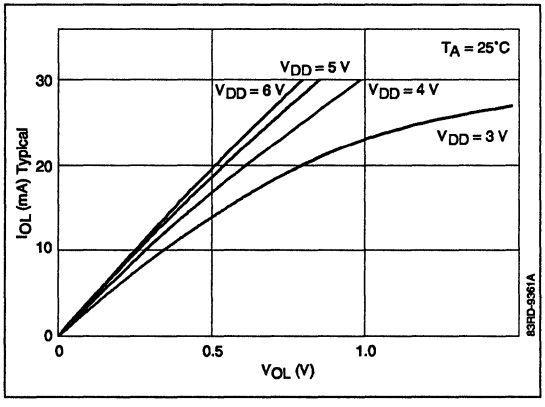


Figure 20. I_{OL} vs V_{OL} ($P6_0$-$P6_3$)

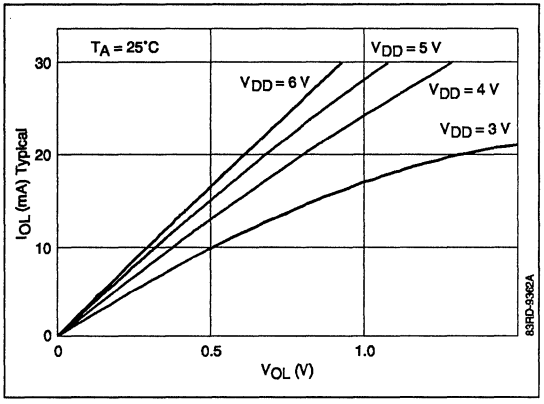
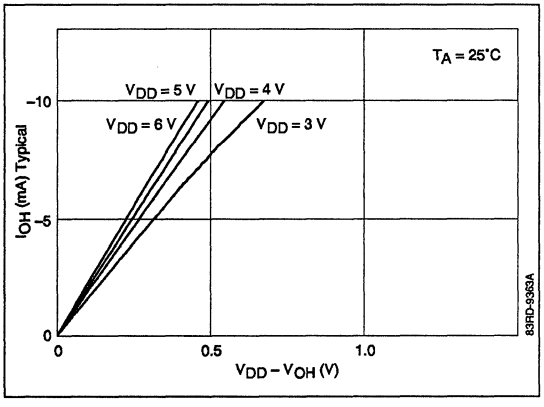


Figure 21. I_{OH} vs $V_{DD} - V_{OH}$ (Ports 0-5, $P6_4$-$P6_7$)



3b

AC Characteristics

T_A = −40 to +85°C; V_{DD} = 2.7 to 6 V ; refer to figures 22 through 27

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time (Min. instruction execution time)	t_{CY}	0.4		64	μs	V_{DD} = 4.5 to 6.0 V; operating on main system clock
		0.96		64	μs	Operating on main system clock
		114	122	125	μs	Operating on subsystem clock
TI input frequency	f_{TI}	0		4	MHz	V_{DD} = 4.5 to 6.0 V
		0		275	kHz	
TI input high/low-level width	t_{TIH}, t_{TIL}	100			ns	V_{DD} = 4.5 to 6.0 V
		1.8			μs	
Interrupt input high/low-level width	t_{INTH}, t_{INTL}	$8/f_{sam}$ (Note 1)			μs	INTP0
		10			μs	INTP1 to INTP3
		10			μs	KR0 to KR7 (Note 2)
$\overline{\text{RESET}}$ low-level width	t_{RST}	10			μs	

Notes:

(1) By using bits 0 and 1 of the sampling clock select (SCS) register in conjunction with bits 0 to 2 of the processor clock control (PCC) register, f_{sam} can be set to $f_X/2^{N+1}$ (where N = 0 to 4), $f_X/64$, or $f_X/128$.

(2) Port 4 falling-edge detection input.

Figure 22. Main System Clock Operation t_{CY} vs V_{DD}

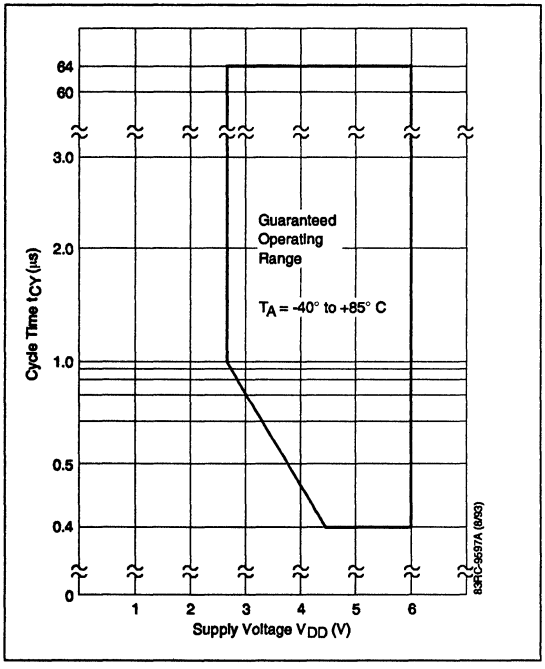


Figure 23. AC Timing Measurement Points (except X1 and XT1)

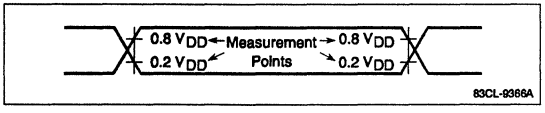


Figure 24. Clock AC Timing Points X1 and XT1

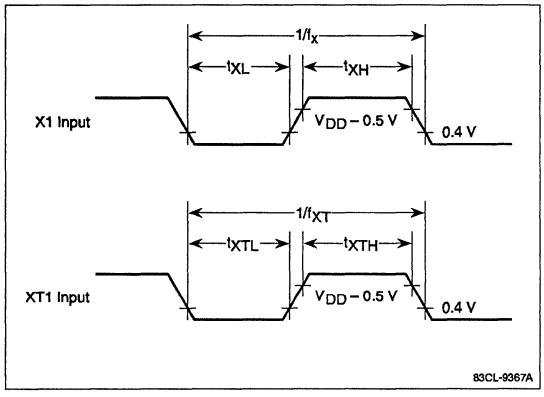


Figure 25. TI Timing

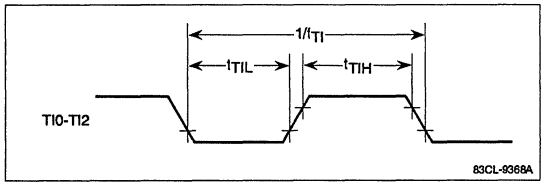


Figure 26. Interrupt Input Timing

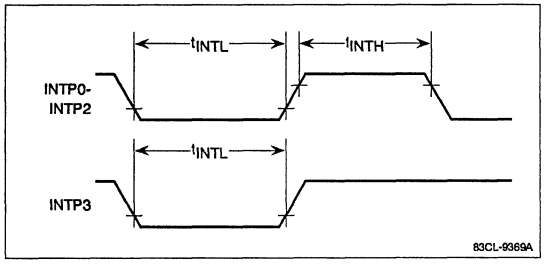
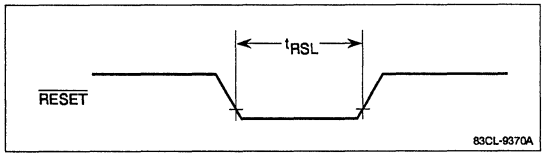


Figure 27. $\overline{\text{RESET}}$ Input Timing



3b

Read/Write Operation

T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 V; refer to figures 28 through 31

Parameter	Symbol	Min	Max	Unit	Conditions
ASTB high-level width	t_{ASTH}	0.5 t_{CY}		ns	
Address setup time to ASTB ↓	t_{ADS}	0.5 t_{CY} − 30		ns	
Address hold time from ASTB ↓	t_{ADH}	10		ns	Load resistor ≥ 5 kΩ
Data input time from address	t_{ADD1}		(2+2n)t_{CY} − 50	ns	Instruction fetch
	t_{ADD2}	5	(3+2n)t_{CY} − 100	ns	Data access
Data input time from $\overline{RD}$ ↓	t_{RDD1}		(1+2n)t_{CY} − 25	ns	Instruction fetch
	t_{RDD2}		(2.5+2n)t_{CY} − 100	ns	Data access
Read data hold time	t_{RDH}	0		ns	
$\overline{RD}$ low-level width	t_{RDL1}	(1.5+2n)t_{CY} − 20		ns	Instruction fetch
	t_{RDL2}	(2.5+2n)t_{CY} − 20		ns	Data access
$\overline{WAIT}$ ↓ input time from $\overline{RD}$ ↓	t_{RDWT1}		0.5 t_{CY}	ns	Instruction fetch
	t_{RDWT2}		1.5 t_{CY}	ns	Data access
$\overline{WAIT}$ ↓ input time from $\overline{WR}$ ↓	t_{WRWT}		0.5 t_{CY}	ns	
$\overline{WAIT}$ low-level width	t_{WTL}	(0.5+2n)t_{CY} + 10	(2+2n)t_{CY}	ns	
Write data setup time to $\overline{WR}$ ↑	t_{WDS}	100		ns	
Write data hold time from $\overline{WR}$ ↑	t_{WDH}	5		ns	
$\overline{WR}$ low-level width	t_{WRL1}	(2.5+2n)t_{CY-20}		ns	
$\overline{RD}$ ↓ delay time from ASTB ↓	t_{ASTRD}	0.5 t_{CY} − 30		ns	
$\overline{WR}$ ↓ delay time from ASTB ↓	t_{ASTWR}	1.5 t_{CY} − 30		ns	
ASTB ↑ delay time from $\overline{RD}$ ↑ (external fetch)	t_{RDAST}	t_{CY} − 10	t_{CY} + 40	ns	
Address hold time from $\overline{RD}$ ↑ (external fetch)	t_{RDADH}	t_{CY}	t_{CY} + 50	ns	
Write data output time from $\overline{RD}$ ↑	t_{RDWD}	10		ns	
WR ↓ delay time from write data	t_{WDWR}	0.5 t_{CY} − 120	0.5 t_{CY}	ns	V_{DD} = 4.5 to 6.0 V
		0.5 t_{CY} − 170	0.5 t_{CY}	ns	
Address hold time from $\overline{WR}$ ↑	t_{WRADH}	t_{CY}	t_{CY} + 60	ns	V_{DD} = 4.5 to 6.0 V
		t_{CY}	t_{CY} + 100	ns	
$\overline{RD}$ ↑ delay time from $\overline{WAIT}$ ↑	t_{WTRD}	0.5 t_{CY}	2.5 t_{CY} + 80	ns	
$\overline{WR}$ ↑ delay time from $\overline{WAIT}$ ↑	t_{WTWR}	0.5 t_{CY}	2.5 t_{CY} + 80	ns	

Notes:

(1) $t_{CY} = t_{CY}/4$

(2) n indicates number of waits.

(3) C_L = 100 pF

Figure 28. Read Operation; External Fetch (No Wait)

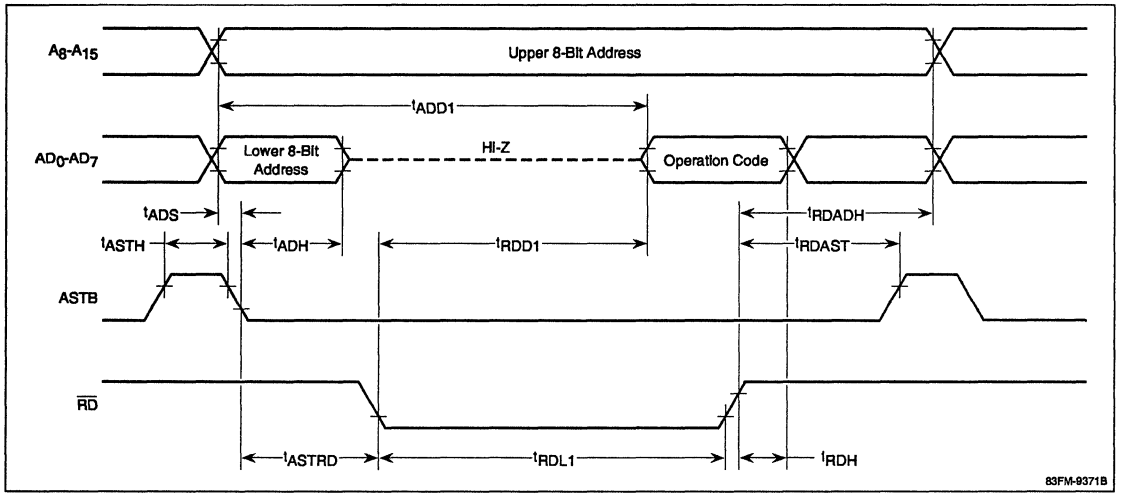
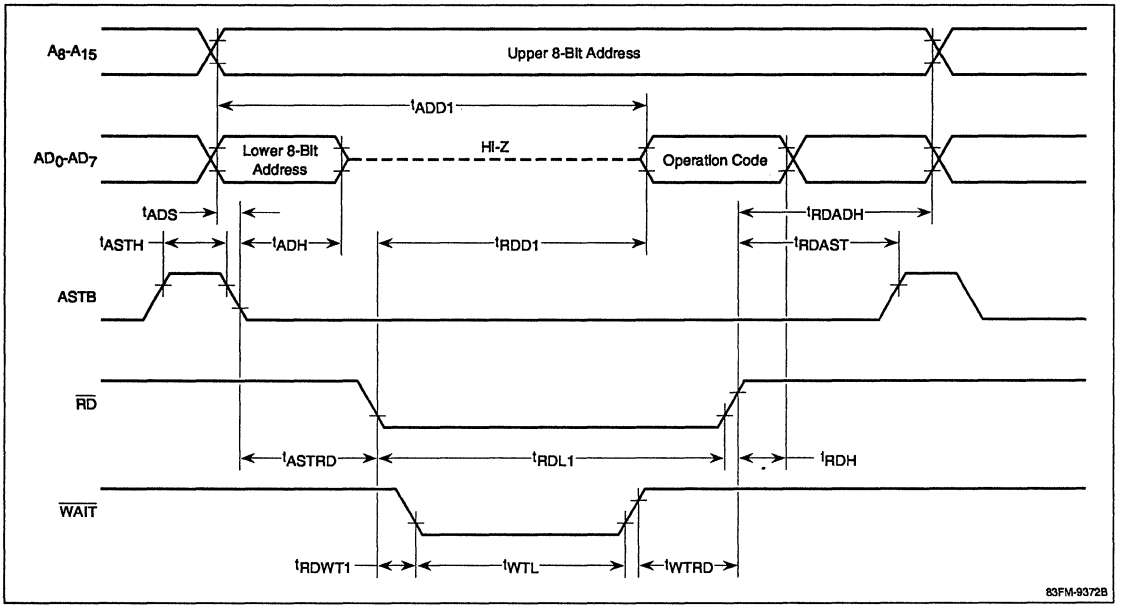


Figure 29. Read Operation; External Fetch (Wait Insertion)



3b

Figure 30. Read/Write Operation; External Data Access (No Wait)

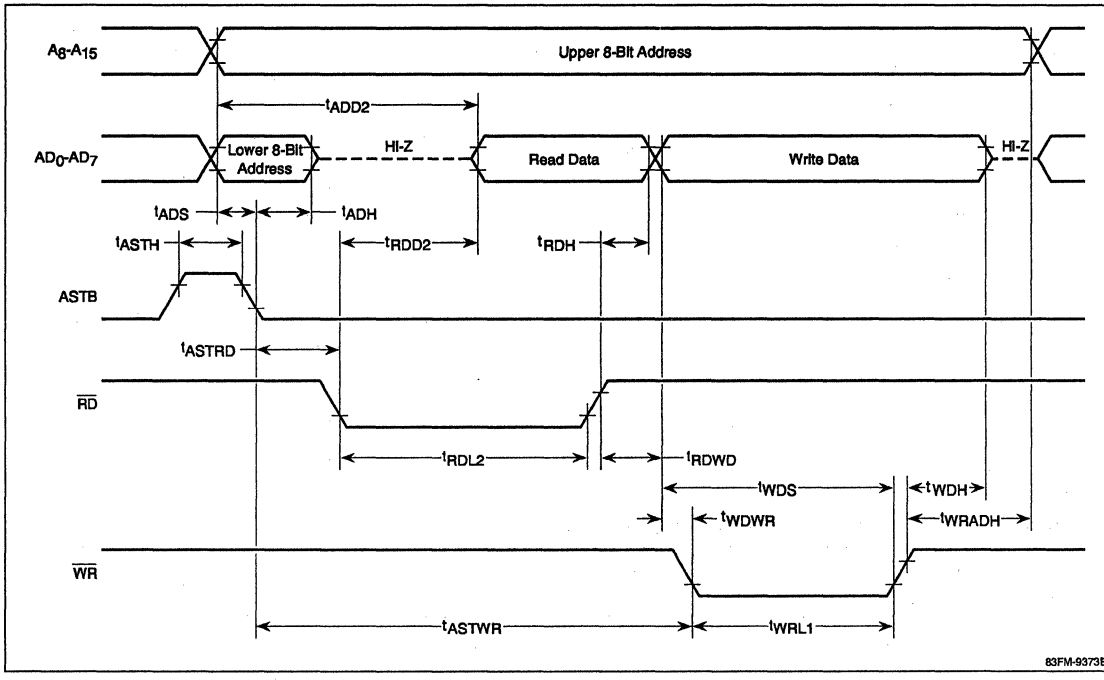
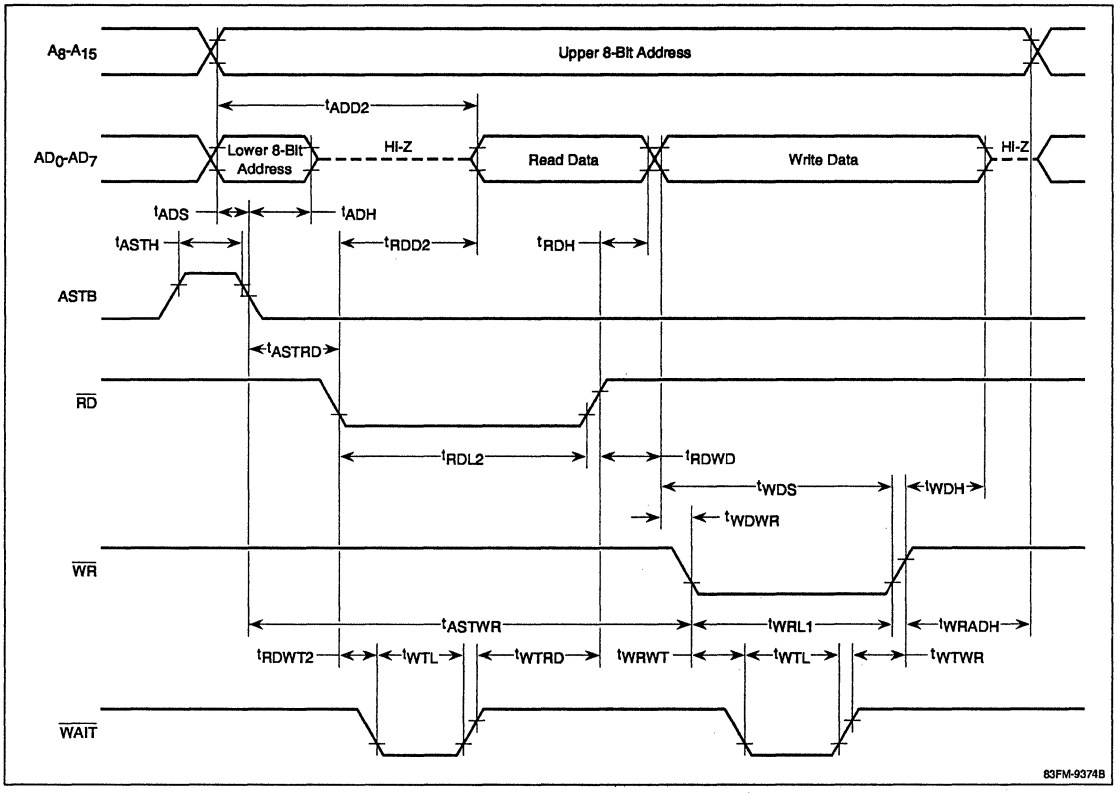


Figure 31. Read/Write Operation; External Data Access (Wait Insertion)



Serial Interface, 3-Wire, I/O Mode; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 32

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY1}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH1}, t_{KL1}	$t_{KCY1}/2 - 50$			ns	V_{DD} = 4.5 to 6.0 V
		$t_{KCY1}/2 - 150$			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK1}	100			ns	
SI hold time from $\overline{SCK}$ ↑	t_{KSI1}	400			ns	
SO output delay time from $\overline{SCK}$ ↓	t_{KSO1}			300	ns	V_{DD} = 4.5 to 6.0 V; C = 100 pF (Note 1)
				1000	ns	C = 100 pF (Note 1)

Note 1: C is the load capacitance of the SO output line.

Serial Interface, 3-Wire, I/O Mode; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 32

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY2}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH2}, t_{KL2}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK2}	100			ns	
SI hold time from $\overline{SCK}$ ↑	t_{KSI2}	400			ns	
SO output delay time from $\overline{SCK}$ ↓	t_{KSO2}			300	ns	V_{DD} = 4.5 to 6.0 V; C = 100 pF (Note 1)
				1000	ns	C = 100 pF (Note 1)

Note 1: C is the load capacitance of the SO output line.

Serial Interface, SBI Mode; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 33

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY3}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH3}, t_{KL3}	$t_{KCY3}/2 - 50$			ns	V_{DD} = 4.5 to 6.0 V
		$t_{KCY3}/2 - 150$			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK3}	100			ns	V_{DD} = 4.5 to 6.0 V
		300			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI3}	$t_{KCY3}/2$			ns	
SB0,SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO3}	0		250	ns	V_{DD} = 4.5 to 6.0 V; R = 1 kΩ, C = 100 pF (Note 1)
		0		1000	ns	R = 1 kΩ, C = 100 pF (Note 1)
SB0, SB1 ↓ from $\overline{SCK}$ ↑	t_{KSB}	t_{KCY3}			ns	
$\overline{SCK}$ ↓ from SB0, SB1 ↓	t_{SBK}	t_{KCY3}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY3}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY3}			ns	

Note 1: R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

Serial Interface, SBI Mode; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 33

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY4}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH4}, t_{KL4}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK4}	100			ns	V_{DD} = 4.5 to 6.0 V
		300			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI4}	t_{KCY4}/2			ns	
SB0, SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO4}	0		300	ns	V_{DD} = 4.5 to 6.0 V; R = 1 kΩ, C = 100 pF. (Note 1)
		0		1000	ns	R = 1 kΩ, C = 100 pF (Note 1)
SB0, SB1 ↓ from $\overline{SCK}$ ↑	t_{KSB}	t_{KCY4}			ns	
$\overline{SCK}$ ↓ from SB0, SB1 ↓	t_{SBK}	t_{KCY4}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY4}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY4}			ns	

Note 1: R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

Serial Interface, 2-Wire, I/O Mode; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 34

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY5}	1600			ns	V_{DD} = 4.5 to 6.0 V
		3800			ns	
$\overline{SCK}$ high-level width	t_{KH5}	t_{KCY5}/2 − 50			ns	
$\overline{SCK}$ low-level width	t_{KL5}	t_{KCY5}/2 − 50			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK5}	300			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI5}	600			ns	
SB0, SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO5}	0		250	ns	V_{DD} = 4.5 to 6.0 V R = 1 kΩ, C = 100 pF (Note 1)
		0		1000	ns	R = 1 kΩ, C = 100 pF (Note 1)

Note 1: R and C are load resistance and load capacitance of the $\overline{SCK0}$, SB0, and SB1 output lines.

3b

Serial Interface, 2-Wire, I/O Mode; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 34

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY6}	1600			ns	V_{DD} = 4.5 to 6.0 V
		3800			ns	
$\overline{SCK}$ high-level width	t_{KH6}	650			ns	
$\overline{SCK}$ low-level width	t_{KL6}	800			ns	
SB0, SB1 setup time (to $\overline{SCK}$ ↑)	t_{SIK6}	100			ns	
SB0, SB1 hold time (from $\overline{SCK}$ ↑)	t_{KSI6}	$t_{KCY6}/2$			ns	
SB0, SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO6}	0		300	ns	V_{DD} = 4.5 to 6.0 V R = 1kΩ, C = 100 pF (Note 1)
		0		1000	ns	R = 1kΩ, C = 100 pF (Note 1)

Note 1: R and C are load resistance and load capacitance of the $\overline{SCK0}$, SB0, and SB1 output lines.

Figure 32. Serial Interface Timing; 3-Wire Serial I/O Mode

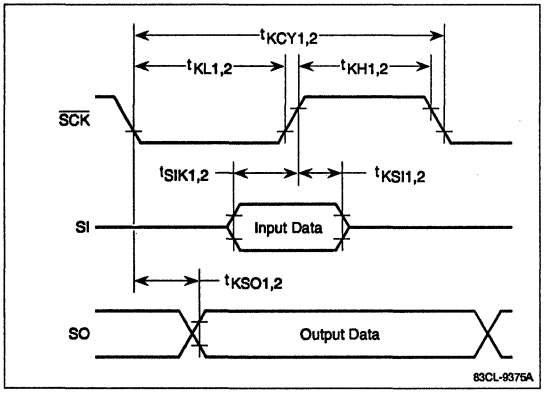


Figure 33. Serial Interface Timing; SBI Mode

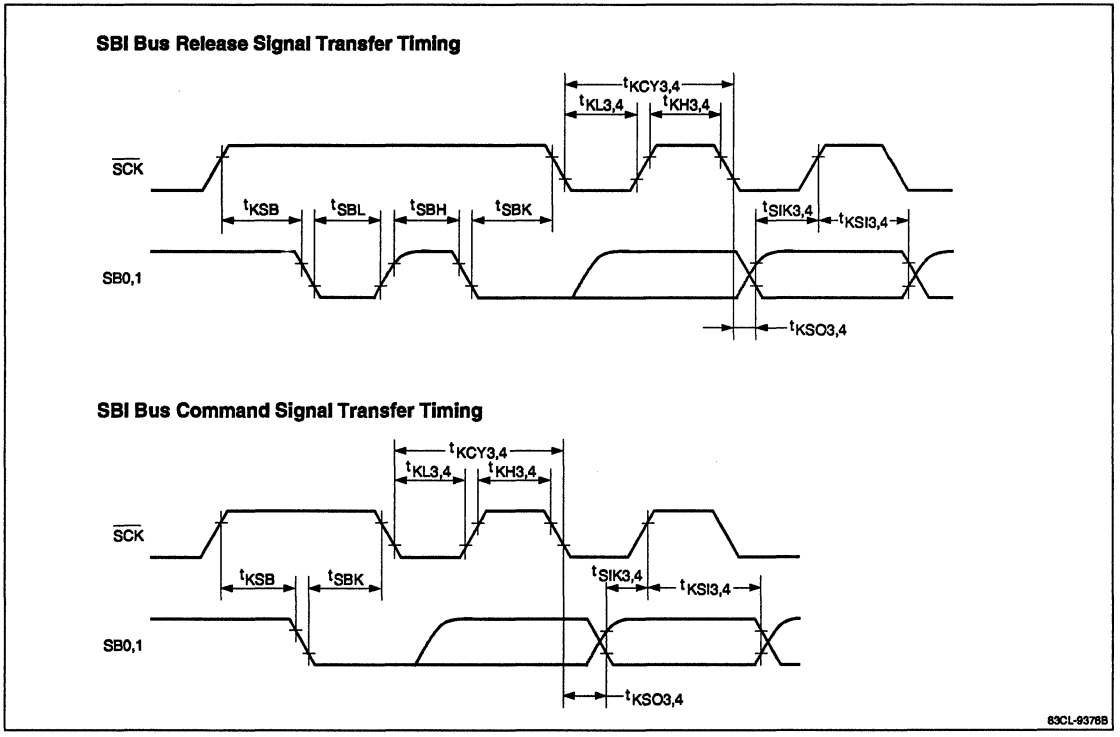
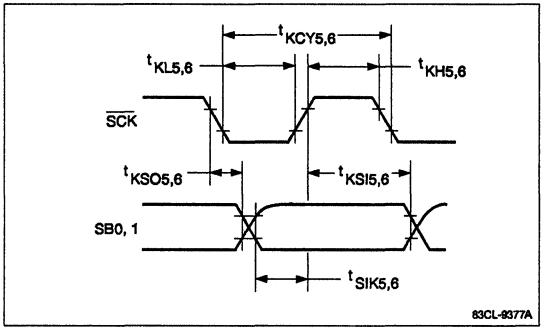


Figure 34. Serial Interface Timing; 2-Wire Serial I/O Mode



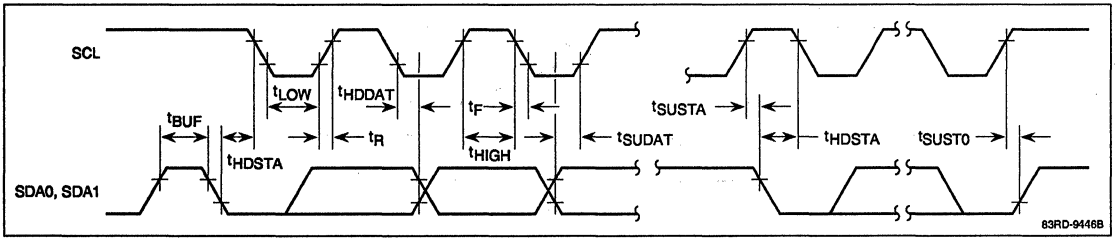
3b

Serial Interface, I²C Bus

Refer to figure 35

Parameter	Symbol	Min	typ	Max	Unit	Conditions
SCL input clock frequency	f_{SCL}	0		100	kHz	
Bus release time before start of transfer	t_{BUF}	4.7			μs	
Start condition hold time	t_{HDSTA}	4.0			μs	
SCL low-level time	t_{LOW}	4.7			μs	
SCL high-level time	t_{HIGH}	4.0			μs	
Start condition setup time	t_{SUSTA}	4.7			μs	
Data hold time	t_{HDDAT}	0			μs	SCL fall time: data retention
Data setup time	t_{SUDAT}	250			ns	
SDA, SDA0, SDA1, SCL signal rise time	t_R			1	μs	
SDA, SDA0, SDA1, SCL signal fall time	t_F			300	ns	
Stop condition setup time	t_{SUSTO}	4.7			μs	

Figure 35. Serial Interface Timing; I²C Bus Mode



Data Memory STOP Mode; Low-Voltage Data Retention

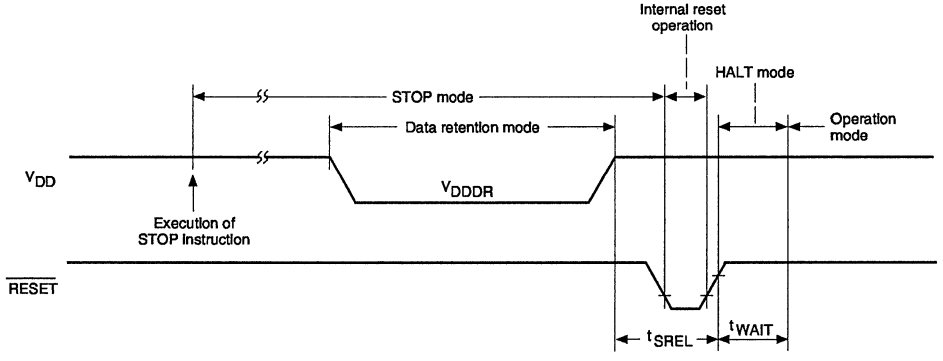
T_A = −40 to +85°C; refer to figure 36

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention supply voltage	V_{DDDR}	2.0		6.0	V	
Data retention power supply current	I_{DDDR}		0.1	10	μA	V_{DDDR} = 2.0 V; subsystem clock stop and feedback resistor disconnected
Release signal set time	t_{SREL}	0			μs	
Oscillation stabilization wait time	t_{WAIT}		$2^{18}/f_X$		ms	Release by $\overline{RESET}$
			(Note 1)		ms	Release by interrupt

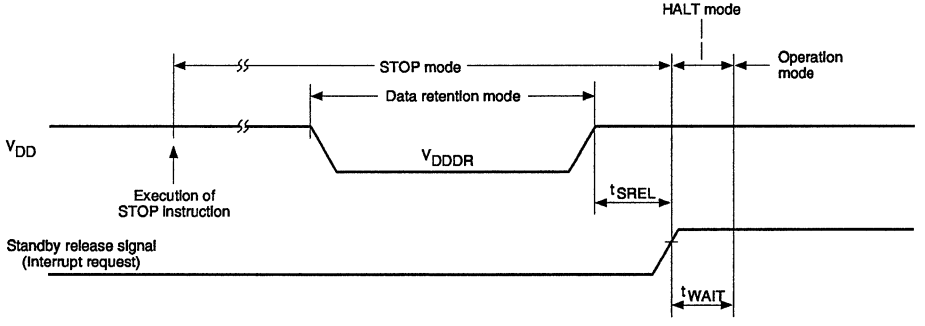
Note: $2^{13}/f_X$, $2^{15}/f_X$, $2^{16}/f_X$, $2^{17}/f_X$ or $2^{18}/f_X$ can be chosen by using bits 0 to 2 of the oscillation stabilization time select (OSTS) register.

Figure 36. Data Retention Timing

A. STOP mode is released by $\overline{RESET}$ input



B. STOP mode is released by interrupt signal



83RD-6456B

NEC
NEC Electronics Inc.

μPD78014 Family (μPD78011B/012B/013/014/P014) 8-Bit, K-Series Microcontrollers General Purpose With A/D Converter

September 1993

Description

The μPD78011B, μPD78012B, μPD78013, μPD78014, and μPD78P014 are members of the K-Series® of microcontrollers. These 8-bit, single-chip microcontrollers have an A/D converter, two serial interface ports, 8-bit hardware multiply and divide instructions, bit manipulation instructions, four banks of main registers, an advanced interrupt handling facility, and a powerful set of memory-mapped on-chip peripherals.

On-board data memory includes 512 or 1024 bytes of internal high-speed RAM plus 32 bytes of serial buffer RAM. Program memory options include 8K, 16K, 24K, or 32K bytes of mask ROM, or 32K bytes of UV EPROM or one-time programmable (OTP) ROM.

The μPD78014 family operates over a wide voltage range: 2.7 to 6.0 volts. Timing is generated by two built-in oscillators. A main oscillator normally drives the CPU and most peripherals and at 10 MHz provides a minimum instruction time of 0.4 μsec. A subsystem oscillator at 32.768 kHz provides time keeping, and optionally a slow clock for the CPU. Since CMOS power dissipation is directly proportional to clock rate, the μPD78014 family provides a software variable CPU clock. The HALT and STOP modes are two additional power saving features that turn off parts of the microcontroller to reduce power consumption. A data retention mode permits RAM contents to be saved down to 2 volts.

The range of peripherals, including an A/D converter, timers, and two serial ports makes these devices ideal for applications in portable battery-powered equipment, office automation, comunications, consumer electronics, home appliances, and fitness equipment.

Features

- Eight-channel 8-bit A/D converter
  - Operates from 2.7 to 6.0 V
- Two-channel serial communication interface
  - 8-bit clock-synchronous interface 0
    - Full-duplex, three-wire mode
    - NEC serial bus interface (SBI) mode
    - Half-duplex, two-wire mode
  - 8-bit clock-synchronous interface 1
    - Full-duplex, three-wire mode with automatic transmit/receive
    - Half-duplex, two-wire mode
- Timers
  - Watchdog timer
  - 16-bit timer/event counter
  - Two 8-bit timer/event counters usable as one 16-bit timer event/counter
  - Clock (watch) timer (time of day tick from either oscillator)
- 53 I/O lines
  - Two CMOS input-only lines
  - 47 CMOS I/O lines
  - Four n-channel, open-drain I/O lines at 15 V maximum
- I/O port pullup resistors
  - Software controllable on 47 lines
  - Mask option on four lines on ROM version
- Program memory
  - μPD78011B: 8K bytes ROM
  - μPD78012B: 16K bytes ROM
  - μPD78013: 24K bytes ROM
  - μPD78014: 32K bytes ROM
  - μPD78P014: 32K bytes EPROM/OTP
- Internal high-speed data memory RAM
  - μPD78011B/012B: 512 bytes
  - μPD78013/014/P014: 1024 bytes
- Specialized memory
  - Serial buffer RAM: 32 bytes
- External memory expansion
  - 64K bytes total memory space
- Powerful instruction set
  - 8-bit unsigned multiply and divide
  - 16-bit arithmetic and data transfer instructions
  - 1-bit and 8-bit logic instructions
- Minimum instruction times:
  - 0.4/0.8/1.6/3.2/6.4 μs program selectable using 10 MHz main system clock
  - 122 μs selectable using 32.768-kHz subsystem clock
- Memory-mapped on-chip peripherals (special function registers)
- Programmable priority, vectored-interrupt controller (two levels)
- Buzzer and clock outputs

K-Series is a registered trademark of NEC Electronics, Inc.

Features (cont)

- Power saving and battery operation features
  - Variable CPU clock rate
  - HALT mode
  - STOP mode
  - 2-V data retention mode
  - CMOS operation; V_{DD} from 2.7 to 6.0 V

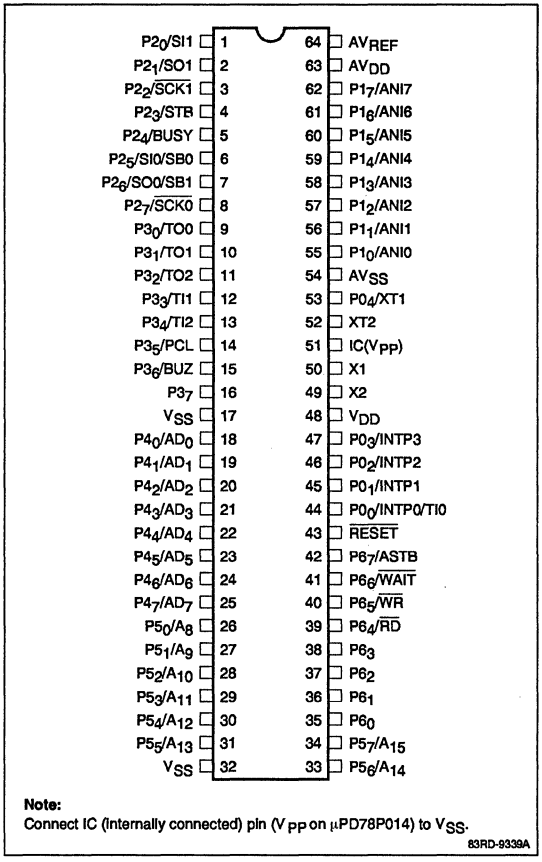
Ordering Information

Part Number	ROM	Package	Package Drawing
μPD78011BCW-xxx	8K mask ROM	64-pin plastic shrink DIP	P64C-70-750A, C
μPD78012BCW-xxx	16K mask ROM		
μPD78013CW-xxx	24K mask ROM		
μPD78014CW-xxx	32K mask ROM		
μPD78P014CW	32K OTP ROM		
μPD78011BGC-xxx-AB8	8K mask ROM	64-pin plastic QFP	P64GC-80-AB8-2
μPD78012BGC-xxx-AB8	16K mask ROM		
μPD78013GC-xxx-AB8	24K mask ROM		
μPD78014GC-xxx-AB8	32K mask ROM		
μPD78P014GC-AB8	32K OTP ROM		
μPD78P014DW	32K UV EPROM	64-pin ceramic shrink DIP w/window	P64DW-70-750A

xxx indicates ROM code suffix

Pin Configurations

64-Pin Plastic or Ceramic Shrink DIP

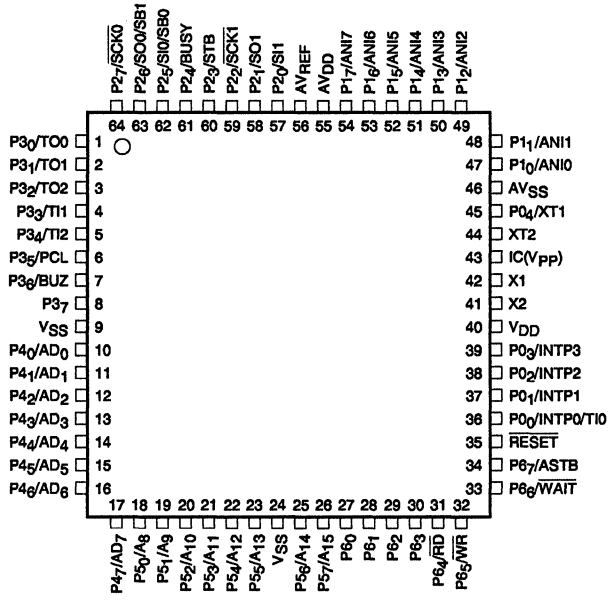


Note:
Connect IC (Internally connected) pin (V_{PP} on μPD78P014) to V_{SS}.

83RD-9339A

Pin Configurations (cont)

64-Pin Plastic QFP



Notes:
Connect IC (Internally connected) pin (V_{PP} on μPD78P014) to V_{SS}.

83RD-9340B

Pin Functions; Normal Operating Mode

Symbol	First Function	Symbol	Alternate Function
$P0_0$	Port 0; 5-bit, bit selectable I/O port (Bits 0 and 4 are input only)	INTP0 TI0	External maskable interrupt External count clock input to timer 0
$P0_1$ $P0_2$ $P0_3$		INTP1 INTP2 INTP3	External maskable interrupt
$P0_4$		XT1	Crystal oscillator or external clock input for subsystem clock
$P1_0$ - $P1_7$	Port 1; 8-bit, bit-selectable I/O port	ANI0-ANI7	Analog input to A/D converter
$P2_0$	Port 2; 8-bit, bit-selectable I/O port	SI1	Serial data input three-wire serial I/O mode
$P2_1$		SO1	Serial data output three-wire serial I/O mode
$P2_2$		$\overline{SCK1}$	Serial clock I/O for serial interface 1
$P2_3$		STB	Serial interface automatic transmit/receive strobe output
$P2_4$		BUSY	Serial interface automatic transmit/receive busy input
$P2_5$		SI0 SB0	Serial data input three-wire serial I/O mode 2/3-wire serial I/O mode
$P2_6$		SO0 SB1	Serial data output three-wire serial I/O mode 2/3-wire serial I/O mode
$P2_7$		$\overline{SCK0}$	Serial clock I/O for serial interface 0
$P3_0$	Port 3; 8-bit, bit-selectable I/O port	TO0	Timer output from timer 0
$P3_1$		TO1	Timer output from timer 1
$P3_2$		TO2	Timer output from timer 2
$P3_3$		TI1	External count clock input to timer 1
$P3_4$		TI2	External count clock input to timer 2
$P3_5$		PCL	Programmable clock output
$P3_6$		BUZ	Programmable buzzer output
$P3_7$		—	
$P4_0$ - $P4_7$	Port 4; 8-bit I/O port	AD_0 - AD_7	Low-order 8-bit multiplexed address/data bus for external memory
$P5_0$ - $P5_7$	Port 5; 8-bit, bit selectable I/O port	A_8 - A_{15}	High-order 8-bit address bus for external memory
$P6_0$-$P6_3$	Port 6; 8-bit, bit selectable ($P6_0$ to $P6_3$ n-channel open-drain I/O with mask option pullup resistors; $P6_4$-$P6_7$ I/O). See note.	—	
$P6_4$		$\overline{RD}$	External memory read strobe
$P6_5$		$\overline{WR}$	External memory write strobe
$P6_6$		$\overline{WAIT}$	External memory wait signal input
$P6_7$		ASTB	Address strobe used to latch address for external memory

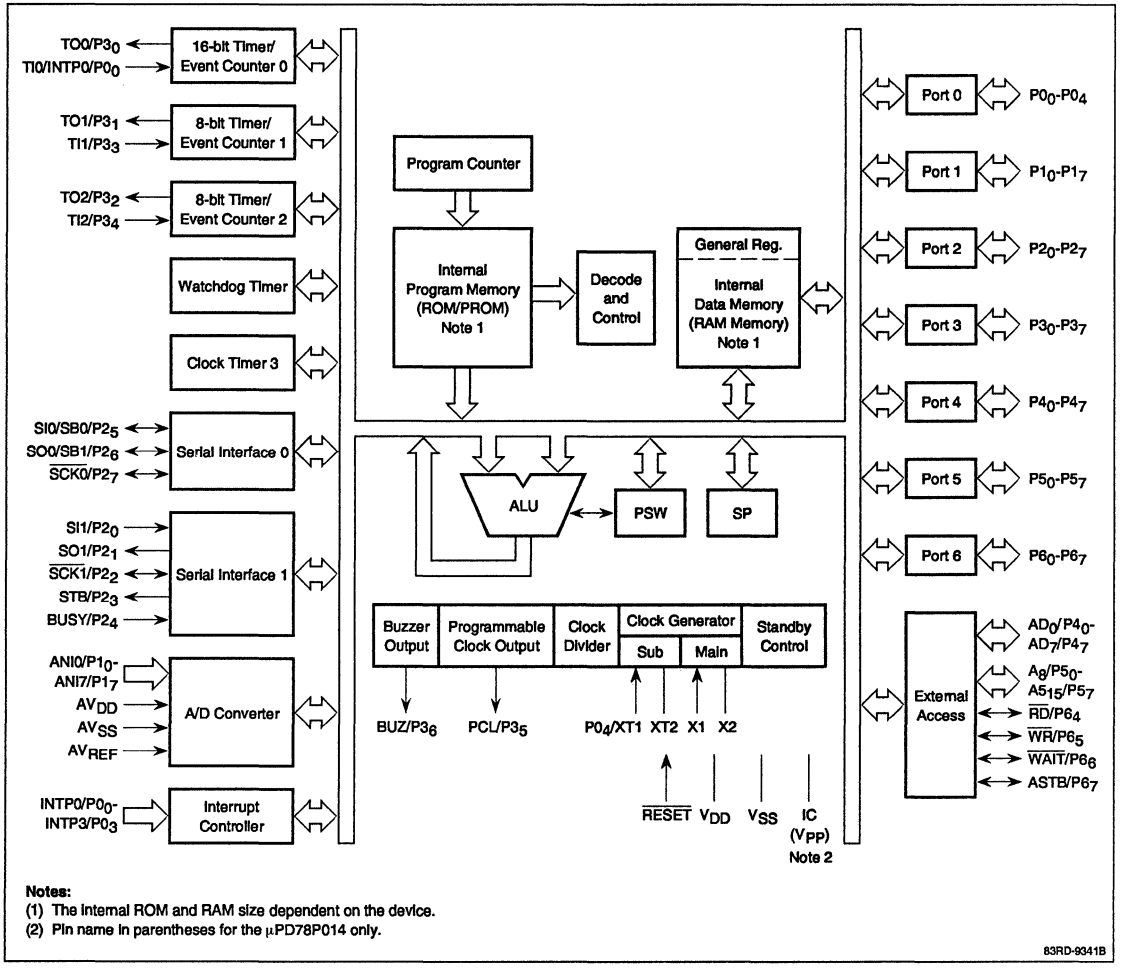
3c

Pin Functions; Normal Operating Mode (cont)

Symbol	First Function	Symbol	Alternate Function
$\overline{RESET}$	External system reset input		
X1	Crystal/ceramic reaonator connection or external clock input for main system clock		
X2	Crystal/ceramic resonator connection or inverse of external clock for main system clock		
XT2	Crystal oscillator or left open when using external clock for subsystem clock		
AV_{REF}	A/D converter reference voltage		
AV_{DD}	A/D converter power supply input		
AV_{SS}	A/D converter ground		
V_{DD}	Power-supply input		
V_{PP}	μPD78P014 PROM programming power-supply input		
V_{SS}	Power-supply ground		
IC	Internal connection		

Note: See table 2 and figure 4 for details

Block Diagram



Notes:
(1) The internal ROM and RAM size dependent on the device.
(2) Pin name in parentheses for the μPD78P014 only.

83RD-9341B

3c

FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processing unit (CPU) of the μPD78014 family features 8- and 16-bit arithmetic including an 8 x 8-bit unsigned multiply and 16 x 8-bit unsigned divide (producing a 16-bit quotient and an 8-bit remainder). The multiply executes in 3.2 μs and the divide in 5 μs using the fastest clock cycle with a main system clock of 10 MHz.

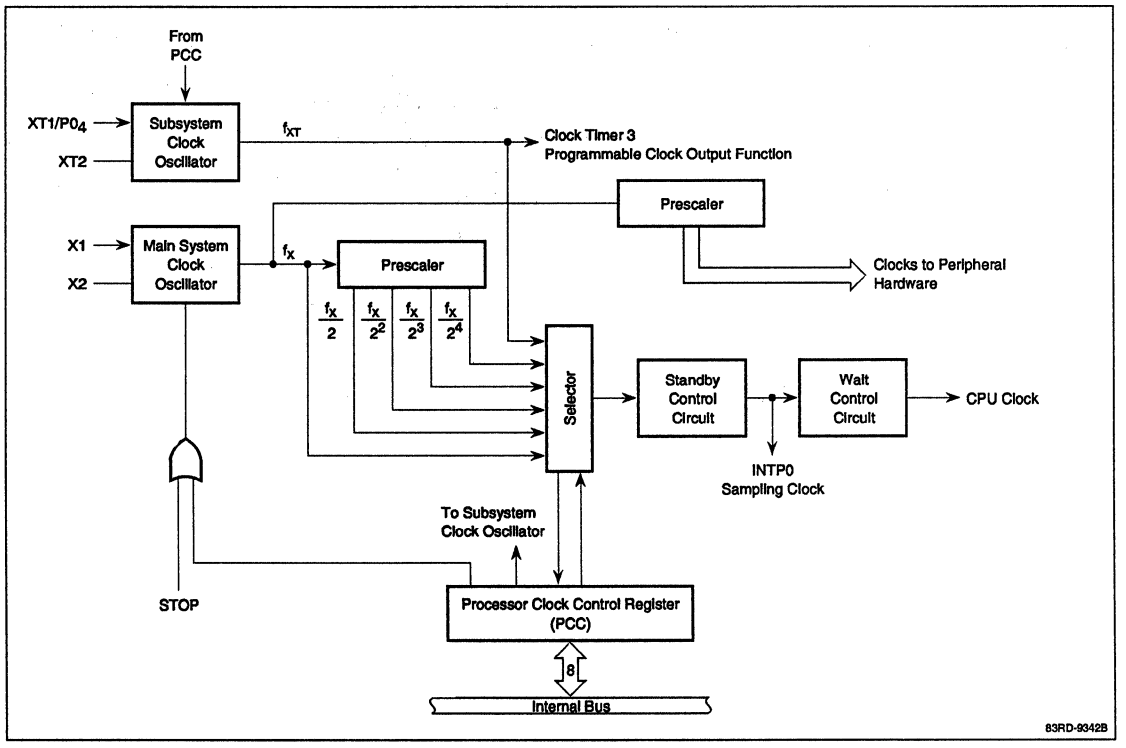
A CALLT vector table and a CALLF area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction can access up to 32 subroutines through their addresses contained in the CALLT vector table (40H to 7FH). A 2-byte call instruction can access any routine beginning in a specific CALLF area (0800H to 0FFFH).

Internal System Clock Generator

The internal system clocks of the μPD78014 family are derived from either the main system or the subsystem oscillator. See figure 1. The clocks for the clock timer and programmable clock output are derived from either the subsystem clock (f_{XT}) or the main system clock. The clocks for all other peripheral hardware are derived from the main system clock.

The CPU clock (ϕ) can be supplied from either the main system clock (f_X) or the subsystem clock (f_{XT}). Using the processor clock control register (PCC), a CPU clock frequency equal to f_X, $f_X/2$, $f_X/4$, $f_X/8$, $f_X/16$ or the subsystem clock f_{XT} can be selected. The CPU clock selected should be based on the power supply voltage available and the desired power consumption. On power up, the CPU clock defaults to the lowest speed from the main system clock and can be changed while the microcontroller is running.

Figure 1. Internal System Clock Generator

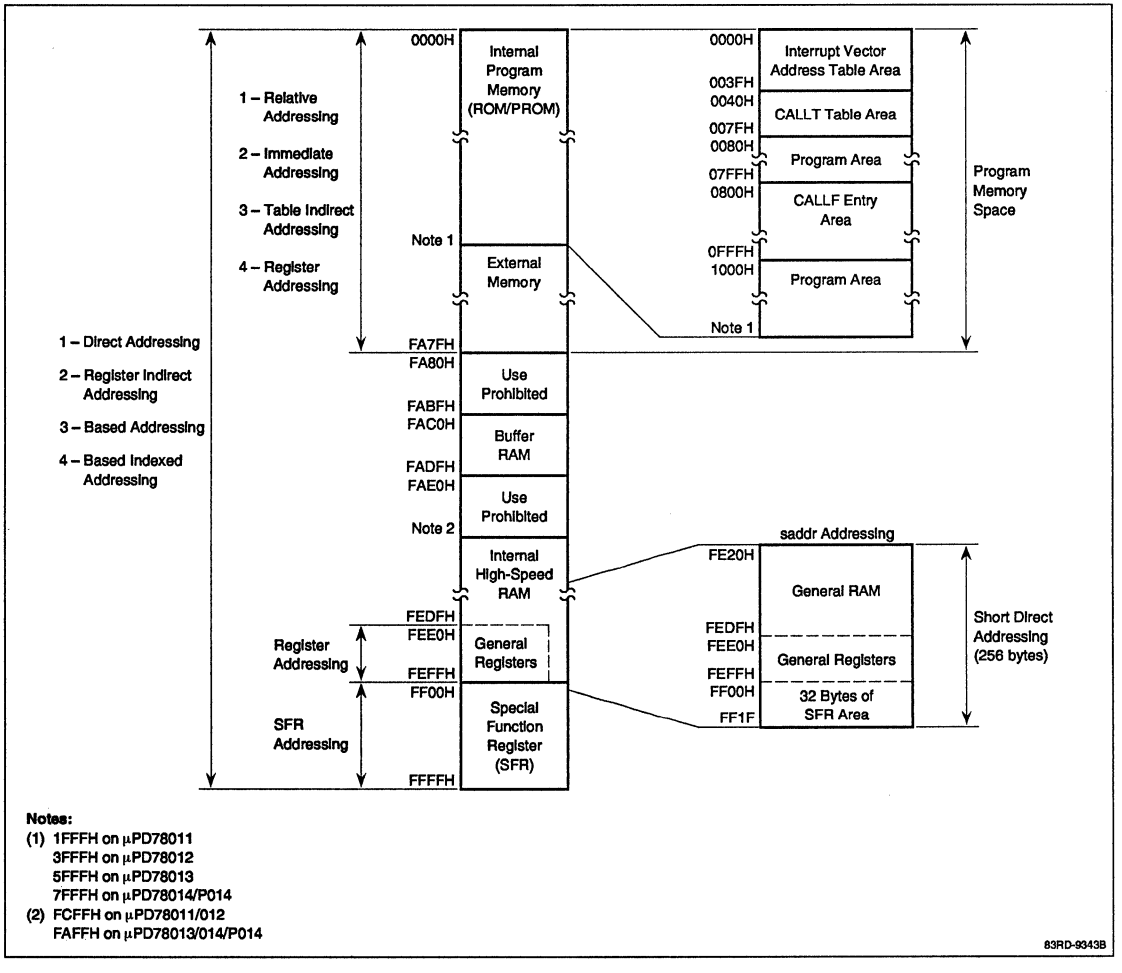


Since the shortest instruction takes four CPU clocks to execute, the fastest minimum instruction execution time (t_{CY}) of 0.4 μs is achieved when using a main system clock at 10 MHz (V_{DD} equals 4.5 to 6.0 V). However, if the clock timer must generate an interrupt every 0.5 or 0.25 seconds, t_{CY} is 0.48 μs at 8.38 MHz. The fastest minimum instruction execution time available across the full voltage range of 2.7 to 6.0 V is 0.96 μs when using a main system clock of 8.38 MHz. For the lowest power consumption, the CPU can be operated from the subsystem clock and the minimum instruction execution time is 122 μs at 32.768 kHz.

Memory Space

The μPD78014 family has a 64K-byte address space. Some of this address space (0000H-FFFFH) can be used as both program and data memory as shown in figure 2.

Figure 2. Memory Map



Notes:
(1) 1FFFH on μPD78011
3FFFH on μPD78012
5FFFH on μPD78013
7FFFH on μPD78014/P014
(2) FCFFH on μPD78011/012
FAFFH on μPD78013/014/P014

83RD-9343B

3c

Internal Program Memory

All devices in the μPD78014 family have internal program memory. The μPD78011B/012B/013/014 contain 8K, 16K, 24K, and 32K bytes of internal ROM, respectively. The μPD78P014 contains 32K bytes of UV EPROM or one time programmable ROM. To allow the μPD78P014 to emulate the mask ROM devices, the amount of internal program memory available in the μPD78P014 can be selected using the memory size switching register (IMS).

Internal RAM

The μPD78011B/012B have 544 bytes and the μPD78013/014/P014 have 1056 bytes of Internal RAM. This Internal RAM consists of two types: high-speed Internal RAM and buffer RAM.

The μPD78011B/012B contain 512 bytes (FD00H to FEFFH) while the μPD78013/014/P014 contain 1024 bytes (FB00H to FEFFH) of high-speed Internal RAM. The high-speed Internal RAM contains the general register banks and the stack. The remainder of the high-speed Internal RAM and any unused register bank locations are available for general storage.

All devices also contain 32 bytes of buffer RAM (FAC0H to FADFH). The buffer RAM is accessed at the same speed as external memory and is used as the buffer area for the automatic transfer mode of serial interface 1 or for general storage.

To allow the μPD78P014 to emulate the mask ROM devices, the amount of high-speed Internal RAM available in the μPD78P014 can also be selected using the IMS.

External Memory

The μPD78014 family can access 0, 256, 4K, 16K or all available bytes of external memory. The μPD78014 family has an 8-bit wide external data bus and a 16-bit wide external address bus. The low-order 8 bits of the address bus are multiplexed to provide the 8-bit data bus and are supplied by port 4. The high-order address bits of the 16-bit address bus are taken from port 5 as required. The address latch, read, and write strobes, and the external $\overline{WAIT}$ signal are supplied by port 6.

The memory expansion mode register (MM) controls the size of external memory. It can be programmed to use 0, 4, 6, or 8 bits from port 5 for the high-order address. Any remaining port 5 bits can be used for I/O. The MM register also can be used to specify one additional wait state or the use of the external $\overline{WAIT}$ signal for low-speed external memory or external peripheral devices.

When only internal ROM and RAM are used and no external memory is required, ports 4, 5 and 6 are available as general purpose I/O ports.

CPU Control Registers

Program Counter. The program counter is a 16-bit binary counter register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 0001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

Program Status Word. The program status word (PSW) is an 8-bit register that contains flags that are set or reset depending on the results of an instruction. This register can be written to or read from 8 bits at a time. The individual flags can also be manipulated on a bit-by-bit basis. The assignment of PSW bits follows.

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

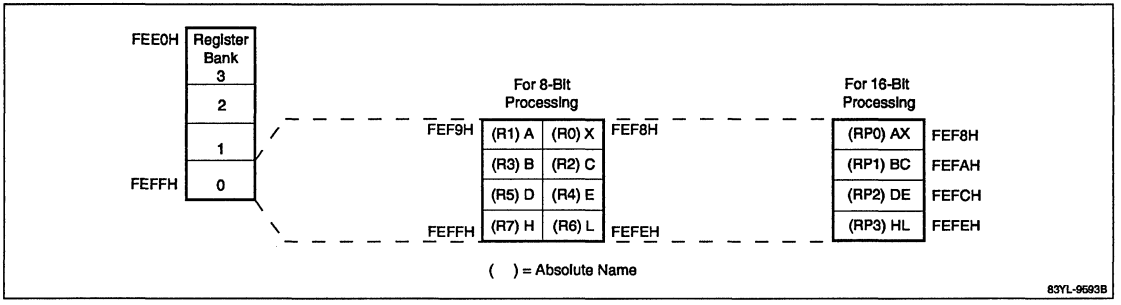
CY	Carry flag
ISP	In-service (interrupt) priority flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

General Registers

The general-purpose registers (figure 3) consist of four banks of registers located at addresses FEE0H to FEFFH in Internal RAM. Each bank consists of eight 8-bit general registers that can also be used in pairs to function as four 16-bit registers. Two bits in the PSW (RBS0 and RBS1) specify which of the register banks is active at any time and are set under program control.

Registers have both functional names (like A, X, B, C, D, E, H or L for 8-bit registers and AX, BC, DE and HL for 16-bit registers) and absolute names (like R1, R0, R3, R2, R5, R4, R7 or R6 for 8-bit registers and RP0, RP1, RP2 or RP3 for 16-bit registers). Either the functional or absolute register names can be used in instructions that use the operand identifiers r and rp.

Figure 3. General Registers



Addressing

The program memory addressing (ROM) modes provided are relative, immediate, table indirect and register addressing. The operand addressing modes provided are relative, immediate, table indirect and register addressing. The operand addressing modes provided are implied, register, direct, short direct (saddr), special function (SFR), register indirect, based, based indexed, and stack addressing.

The 'SFR addressing' and 'saddr addressing' modes use direct addressing, and require only 1 byte in the instruction to address RAM. Normally, a 65K byte address space requires 2 bytes to address it. One-byte addressing results in faster access times, since the instructions are shorter. SFR addressing addresses the entire 256-byte SFR address space from FF00H to FFFFH. Saddr addressing (see figure 2) addresses the 256-byte address space FE20H to FF1FH. FE20H to FEFFH are composed of 224 bytes of internal high speed RAM; FF00H to FF1FH contain the first 32 bytes in the special function register area.

One-byte addressing is accomplished by using the first byte of the instruction for the opcode (and one operand if register A or AX is used) and the second byte of the instruction as an address (offset) into the 256-byte area. If register A or AX is used, the instructions are 2 bytes long, thereby providing fast access times. If immediate data is used, the instruction will be 3 or 4 bytes long depending upon whether the immediate data is a byte or a word. Many 16-bit SFRs are in the space FF00H to FF1FH. Using AX as an operand to these SFRs will provide fast access, since the instructions will be only 2 bytes long.

3c

Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by SFR addressing. FF00H to FF1H can also be accessed using saddr addressing. They are either 8 or 16 bits as required, and many of the 8-bit registers are bit addressable.

Locations FFD0H through FFDFH are known as the external SFR area. Registers in external circuitry interfaced and mapped to these addresses can only be addressed by main memory addressing. Table 1 lists the special function registers.

Table 1. Special Function Registers

				Access Units (Bits)			
Address	**Register (SFR)**	**Symbol**	**R/W**	**1**	**8**	**16**	**State After Reset**
FF00H	Port 0	P0	R/W	x	x	—	00H
FF01H	Port 1	P1	R/W	x	x	—	00H
FF02H	Port 2	P2	R/W	x	x	—	00H
FF03H	Port 3	P3	R/W	x	x	—	00H
FF04H	Port 4	P4	R/W	x	x	—	Undefined
FF05H	Port 5	P5	R/W	x	x	—	Undefined
FF06H	Port 6	P6	R/W	x	x	—	Undefined

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF10H-FF11H	Compare register 00	CR00	R/W	—	—	x	Undefined
FF12H-FF13H	Compare register 01	CR01	R	—	—	x	Undefined
FF14H-FF15H	16-bit timer register	TM0	R	—	—	x	00H
FF16H	Compare register 10	CR10	R/W	—	x	—	Undefined
FF17H	Compare register 20	CR20	R/W	—	x	—	Undefined
FF18H	8-bit timer register 1	TM1	R	x	x	—	00H
FF19H	8-bit timer register 2	TM2	R	x	x	—	00H
FF18H-FF19H	16-bit timer register 1	TMS	R	—	—	x	0000H
FF1AH	Serial I/O shift register 0	SIO0	R/W	—	x	—	Undefined
FF1BH	Serial I/O shift register 1	SIO1	R/W	—	x	—	Undefined
FF1FH	A/D conversion result register	ADCR	R	—	x	—	Undefined
FF20H	Port mode register 0	PM0	R/W	x	x	—	1FH
FF21H	Port mode register 1	PM1	R/W	x	x	—	FFH
FF22H	Port mode register 2	PM2	R/W	x	x	—	FFH
FF23H	Port mode register 3	PM3	R/W	x	x	—	FFH
FF25H	Port mode register 5	PM5	R/W	x	x	—	FFH
FF26H	Port mode register 6	PM6	R/W	x	x	—	FFH
FF40H	Timer clock select register 0	TCL0	R/W	x	x	—	00H
FF41H	Timer clock select register 1	TCL1	R/W	—	x	—	00H
FF42H	Timer clock select register 2	TCL2	R/W	—	x	—	00H
FF43H	Timer clock select register 3	TCL3	R/W	—	x	—	88H
FF47H	Sampling clock select register	SCS	R/W	—	x	—	00H
FF48H	16-bit timer mode control register	TMC0	R/W	x	x	—	00H
FF49H	8-bit timer mode control register	TMC1	R/W	x	x	—	00H
FF4AH	Watch (clock) timer mode control register	TMC2	R/W	x	x	—	00H
FF4EH	16-bit timer output control register	TOC0	R/W	x	x	—	00H
FF4FH	8-bit timer output control register	TOC1	R/W	x	x	—	00H
FF60H	Serial operating mode register 0	CSIM0	R/W	x	x	—	00H
FF61H	Serial bus interface control register	SBIC	R/W	x	x	—	00H
FF62H	Slave address register	SVA	R/W	—	x	—	Undefined
FF63H	Interrupt timing specify register	SINT	R/W	x	x	—	00H
FF68H	Serial operation mode register 1	CSIM1	R/W	x	x	—	00H
FF69H	Automatic data transmit/receive control register	ADTC	R/W	x	x	—	00H
FF6AH	Automatic data transmit/receive address pointer register	ADTP	R/W	—	x	—	00H
FF80H	A/D converter mode register	ADM	R/W	x	x	—	01H
FF84H	A/D converter input select register	ADIS	R/W	—	x	—	00H
FFD0H-FFDFH	External SFR access area(Note 1)	—	R/W	x	x	—	Undefined
FFE0H	Interrupt flag register L	IF0L	R/W	x	x	—	00H
FFE1H	Interrupt flag register H	IF0H	R/W	x	x	—	00H

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FFE0H-FFE1H	Interrupt flag register	IF0	R/W	—	—	x	0000H
FFE4H	Interrupt mask flag register L	MK0L	R/W	x	x	—	FFH
FFE5H	Interrupt mask flag register H	MK0H	R/W	x	x	—	FFH
FFE4H-FFE5H	Interrrupt mask flag register	MK0	R/W	—	—	x	FFFFH
FFE8H	Priority order specify flag register L	PR0L	R/W	x	x	—	FFH
FFE9H	Priority order specify flag register H	PR0H	R/W	x	x	—	FFH
FFE8H-FFE9H	Priority order specify flag register	PR0	R/W	—	—	x	FFFFH
FFECH	External interrupt mode register	INTM0	R/W	—	x	—	00H
FFF0H	Memory size switch register (Note 2)	IMS	W	—	x	—	C8H
FFF6H	Key return mode register	KRM	R/W	x	x	—	02H
FFF7H	Pullup resistor option register	PUO	R/W	x	x	—	00H
FFF8H	Memory expanded mode register	MM	R/W	x	x	—	10H
FFF9H	Watchdog timer mode register	WDTM	R/W	x	x	—	00H
FFFAH	Oscillation stabilization time select register	OSTS	R/W	—	x	—	04H
FFFBH	Processor clock control register	PCC	R/W	x	x	—	04H

Notes:

(1) The external access area cannot be accessed using SFR addressing. It can only be accessed using main memory addressing.

(2) μPD78P014 only.

3c

Input/Output Ports

The μPD78014 family has up to 53 port lines. Table 2 lists the features of each port and figure 4 shows the structure of each port pin.

Table 2. Digital Port Functions

Port	Operational Features	Configuration	Direct Drive Capability	Software Pullup Resistor Connection (Note 1)
Port 0 (Note 2)	5-bit input or output	Bit selectable		Byte selectable, input bits only
Port 1	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 2	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 3	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 4	8-bit input or output	Byte selectable		Byte selectable, input bits only
Port 5	8-bit input or output	Bit selectable	LED	Byte selectable, input bits only
Port 6	8-bit input or output ($P6_0$ - $P6_3$ n-channel)	Bit selectable	15 V max ($P6_0$ - $P6_3$)	Byte selectable, input bits only $P6_0$ - $P6_3$ - mask option only (Note 3) $P6_4$ - $P6_7$ - software

Notes:

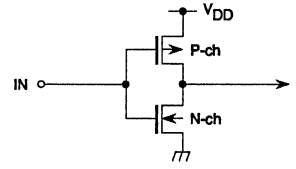
(1) Software pullup resistors can be internally connected (only on a port-by-port basis) to port bits set to input mode. Pullup resistors are not connected to port bits set to output mode.

(2) $P0_0$ and $P0_4$ are input only and do not have a software pullup resistor.

(3) All devices except μPD78P014

Figure 4. Pin Input/Output Circuits

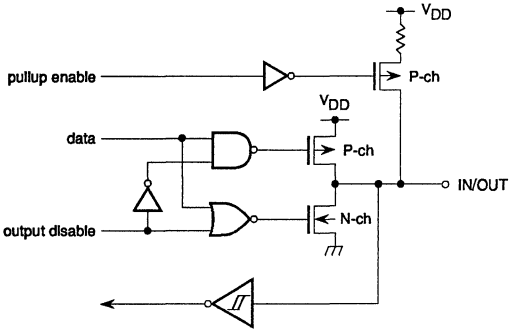
Type 1 ($P0_4$)



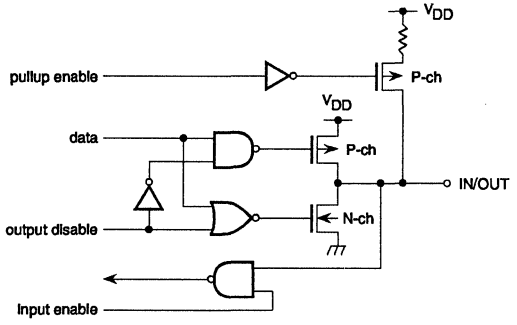
Type 2 ($P0_0$, $\overline{RESET}$)

Schmitt trigger input with hysteresis characteristics

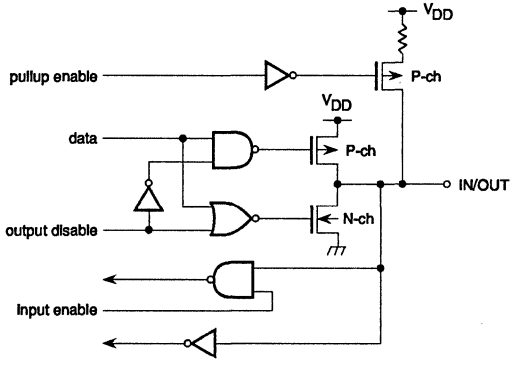
Type 8-A ($P0_1$ - $P0_3$, $P2_0$, $P2_2$, $P2_4$, $P3_3$ - $P3_4$)



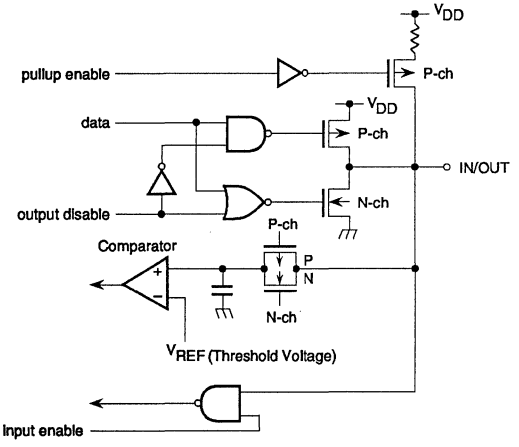
Type 5-A ($P2_1$, $P2_3$, $P3_0$- $P3_2$, $P3_5$-$P3_7$, $P5_0$-$P5_7$, $P6_4$-$P6_7$)



Type 5-B ($P4_0$-$P4_7$)



Type 9-B ($P1_0$-$P1_7$)

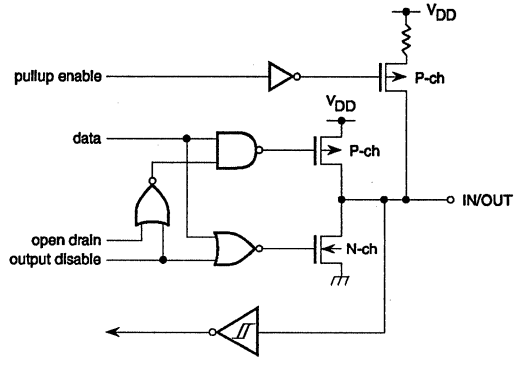


83YL-9344B (1)

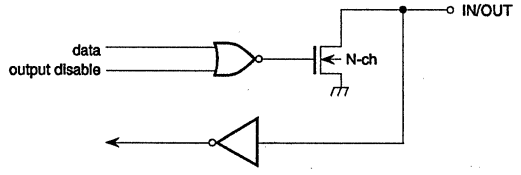
3c

Figure 4. Pin Input/Output Circuits (cont)

Type 10-A ($P2_5$ - $P2_7$)

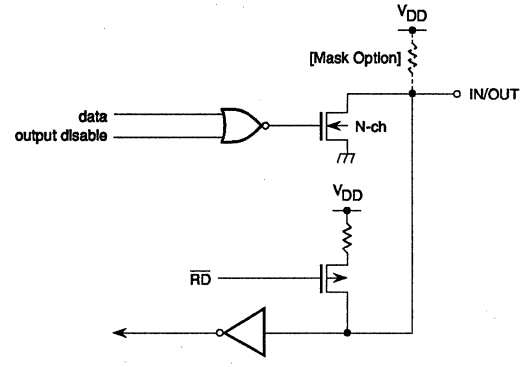


Type 13 ($P6_0$ - $P6_3$: μPD78PO14)



Middle-High Voltage Input Buffer

Type 13-B ($P6_0$ - $P6_3$: μPD780 1x)



Middle-High Voltage Input Buffer

83YL-9344B (2)

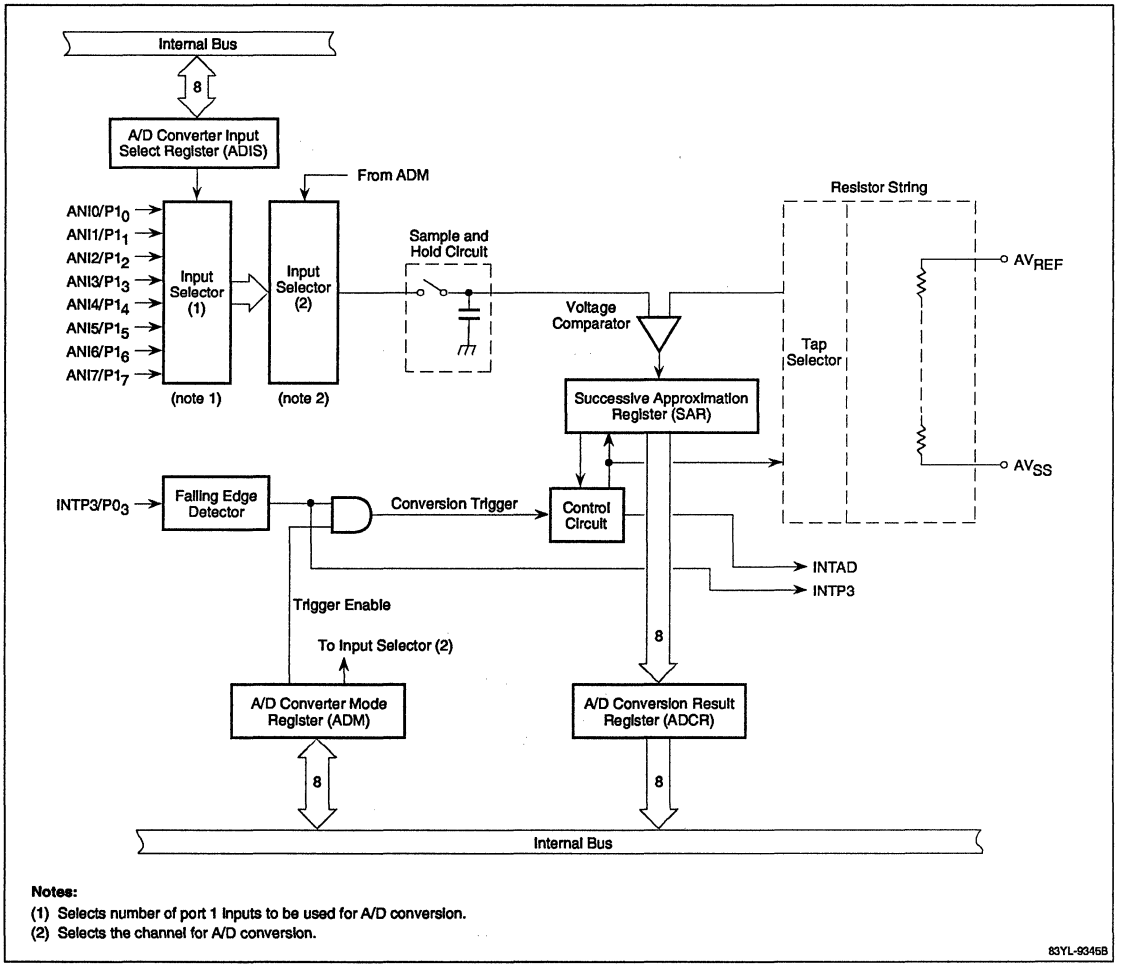
Analog-to-Digital (A/D) Converter

The μPD78014 family A/D converter (see figure 5) uses the successive-approximation method for converting one of eight multiplexed analog inputs into 8-bit digital data. The conversion time per input is 19.1 μs at 8.38 MHz operation.

The A/D converter input select register (ADIS) selects the number of inputs that are used in A/D conversion. The remaining inputs are used as ports. The A/D input to be converted is selected by programming the A/D converter mode register (ADM). A/D conversion is started by external interrupt INTP3, or by writing to ADM. When the conversion is completed, the results are stored in the A/D conversion result register (ADCR) and an INTAD interrupt is generated.

If the A/D converter was started by an external interrupt, the A/D converter stops after the interrupt is generated. If the A/D converter was started by software, the A/D converter repeats the conversion until new data is written to the ADM register.

Figure 5. A/D Converter



Notes:
(1) Selects number of port 1 inputs to be used for A/D conversion.
(2) Selects the channel for A/D conversion.

83YL-9346B

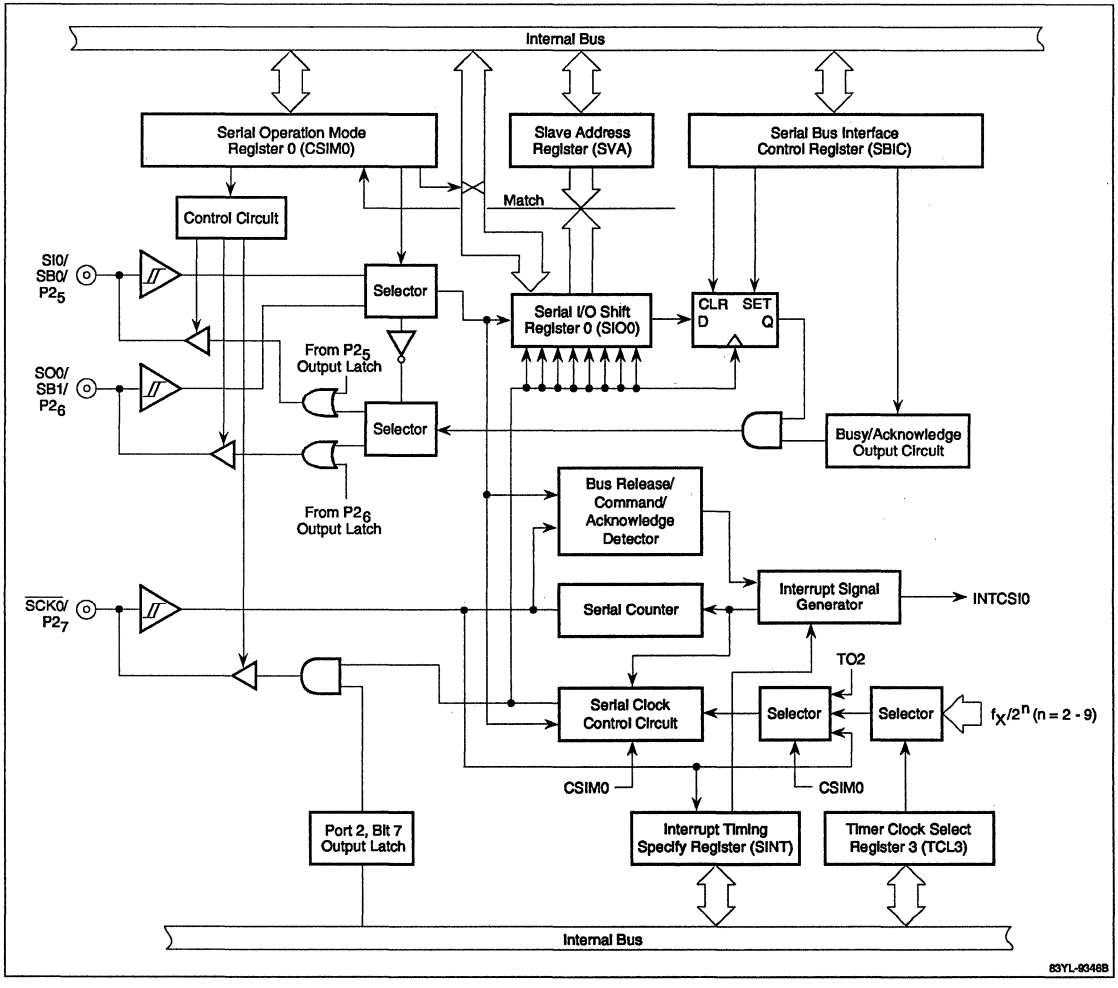
Serial Interfaces

The μPD78014 family has two independent serial interfaces: serial interface 0 and serial interface 1.

Serial Interface 0. Serial interface 0 is an 8-bit clock synchronous serial interface (figure 6). It can be operated in either a three-wire serial I/O mode, NEC serial bus interface (SBI) mode, or two-wire serial I/O mode. The serial clock can be provided from one of eight internal clocks, the output of 8-bit timer register 2, or the external clock line $\overline{SCK0}$.

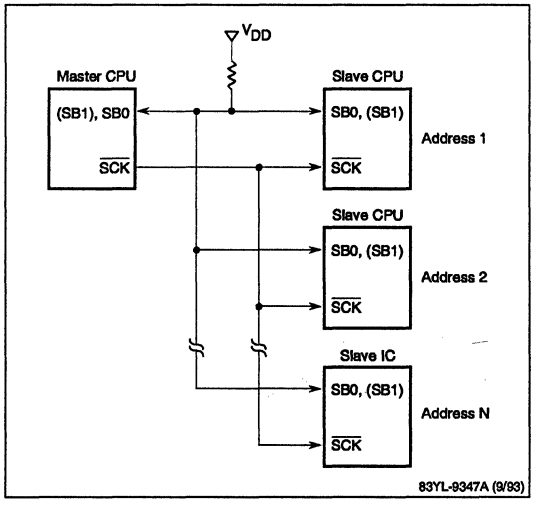
In the three-wire serial I/O mode, the 8-bit shift register (SIO0) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO0 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI0 line providing full-duplex operation. The INTCSI0 interrupt is generated after each 8-bit transfer.

Figure 6. Serial Interface 0



The NEC SBI mode is a two-wire high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are connected in a master/slave configuration (see figure 7). There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over one of the serial bus lines (SB0 or SB1) using a fixed hardware protocol synchronized with the $\overline{SCK0}$ line . Each slave device of the μPD78014 family can be programmed to respond in hardware to any one of 256 addresses set in its slave address register (SVA). There are also 256 commands and 256 data types. Since all commands are user definable, many software protocols, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

Figure 7. SBI Mode Master/Slave Configuration



The two-wire serial I/O mode provides half-duplex operation using either the SB0 or SB1 line and the $\overline{SCK0}$ line. Communication format and handshaking can be handled in software by controlling the output levels of the data and clock lines between transfers. For data transmission, the 8-bit shift register (SIO0) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of either the SB0 or SB1 line (MSB first). In addition, this byte of data is also shifted back into SIO0 on the rising edge of these pulses providing a way of verifying that the transmission was correct.

For data reception, the SIO0 register is preloaded with the value FFH. As this data value is shifted out on the falling edge of the serial clock, it disables the n-channel open-drain driver. This allows the receive data to be driven on to the serial line and shifted into the SIO0 register on the rising edge of the serial clock. The INTCSI0 interrupt is generated after each 8-bit transfer.

Serial Interface 1. Serial interface 1 is also an 8-bit clock synchronous serial interface (figure 8). It can be operated in either a three-wire serial I/O mode, or three-wire serial I/O mode with automatic transmit/receive. The serial clock can also be provided from one of eight internal clocks (common clock for both interfaces), the output of 8-bit timer register 2, or the external clock line $\overline{SCK1}$.

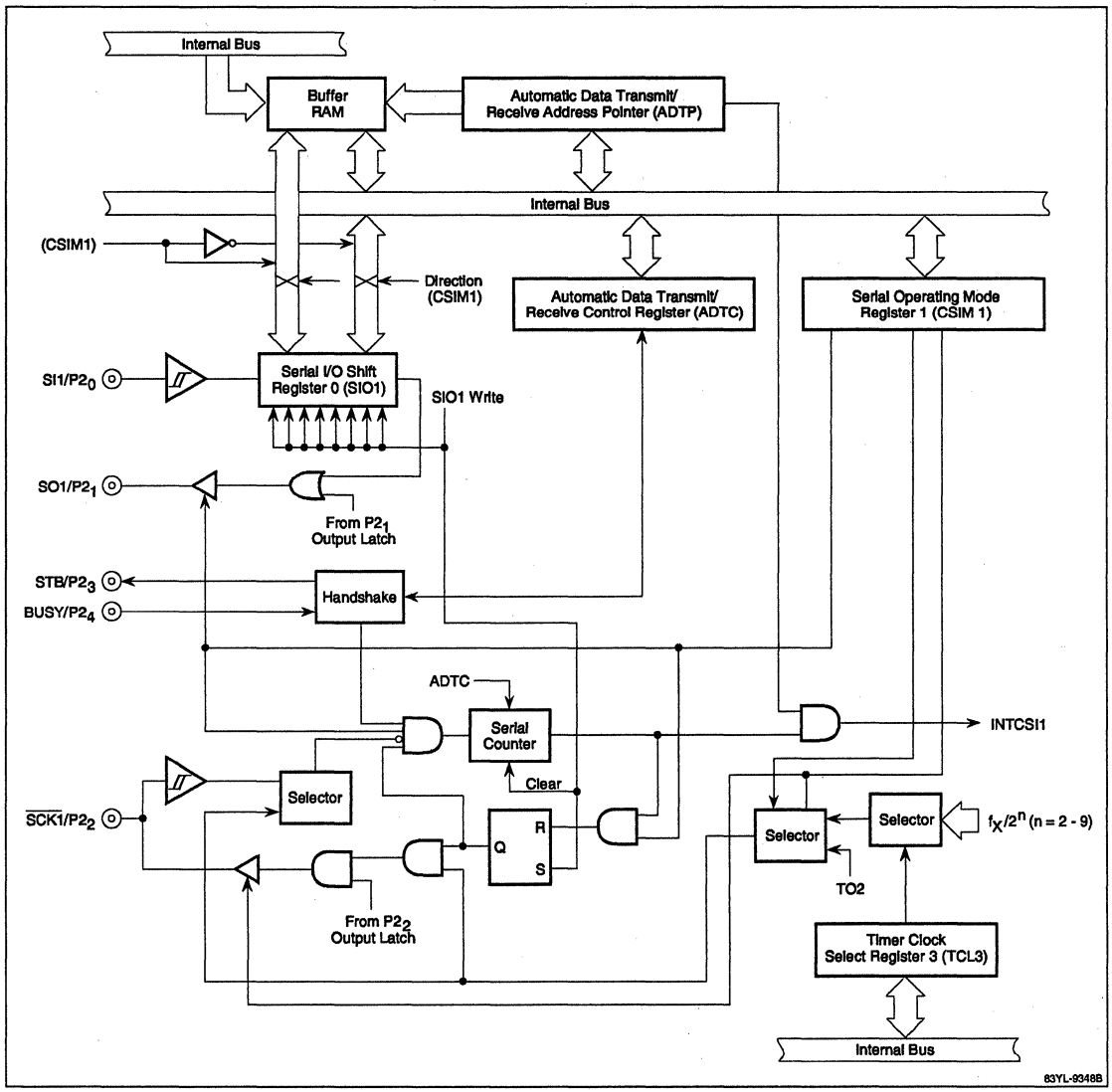
In the three-wire serial I/O mode, the 8-bit shift register (SIO1) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO1 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI1 line providing full-duplex operation. The INTCSI1 interrupt is generated after each 8-bit transfer.

In the three-wire serial I/O mode with automatic transmit/receive, up to 32 bytes of data can be transferred with minimal CPU overhead. The data to be transmitted and received is stored in the buffer RAM. Handshaking using either the BUSY input line, the strobe (STB) output line, or both, can be selected by the program. Error detection of bit drift due to noise is available for each byte transferred when using the BUSY input line. This automatic transmit/receive mode is ideally suited for transferring data to/from external peripheral devices such as onscreen display (OSD) and LCD controller/driver devices.

While in three-wire serial interface mode with automatic data transfer, the interface can be operated as either a full-duplex interface or a transmit-only interface in single or repetitive operation mode. In the full-duplex mode, a byte of data is transferred from the first location in the buffer RAM and shifted out of the SO1 line (either MSB or LSB first) while the received data is shifted into the SI1 line and stored back in the first buffer location. After the preset number of bytes has been transferred, the INTCSI1 interrupt is generated.

In single-operation transmit mode, the preset number of bytes from the buffer RAM are transmitted out of the SO1 line (either MSB or LSB first) and the INTCSI1 interrupt is generated after all bytes are transferred. In the repetitive operation transmit mode, data in the buffer is transmitted repeatedly.

Figure 8. Serial Interface 1

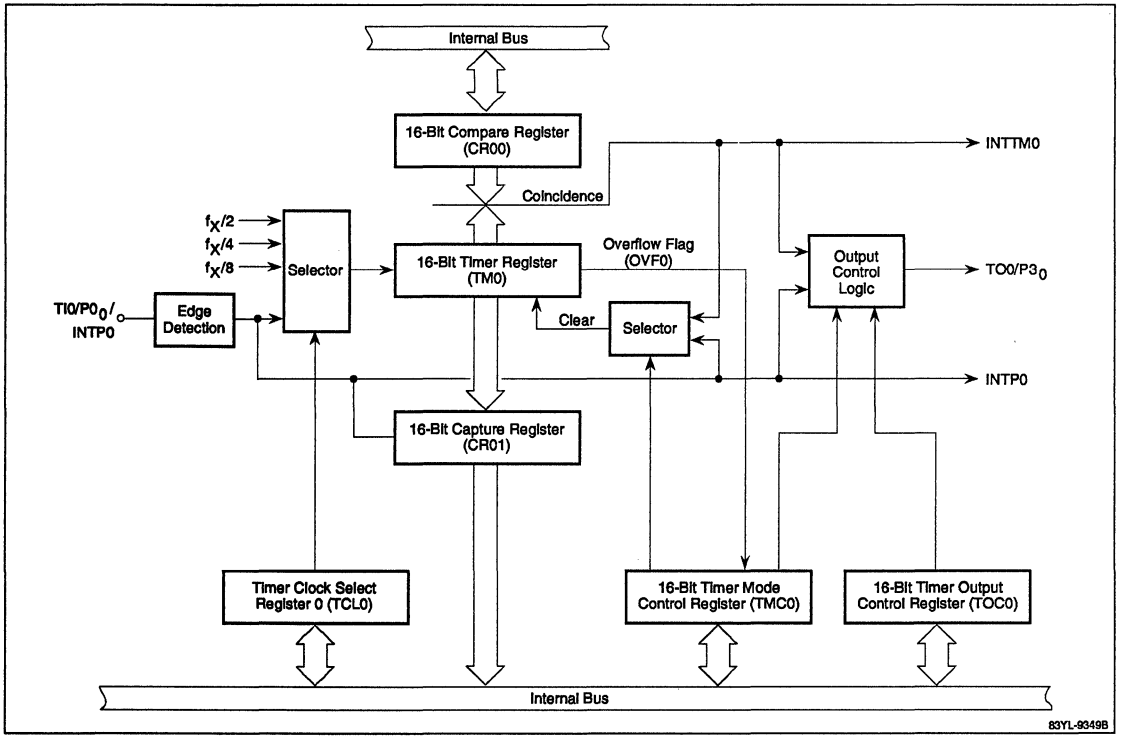


Timers

The μPD78014 family has one 16-bit timer/event counter, two 8-bit timer/event counters that can be combined for use as a 16-bit timer/event counter, a clock timer, and a watchdog timer. All of these can be programmed to count a number of prescaled values of the main system clock. In addition, the clock timer can also count the subsystem clock. All of the timer/event counters can count external events.

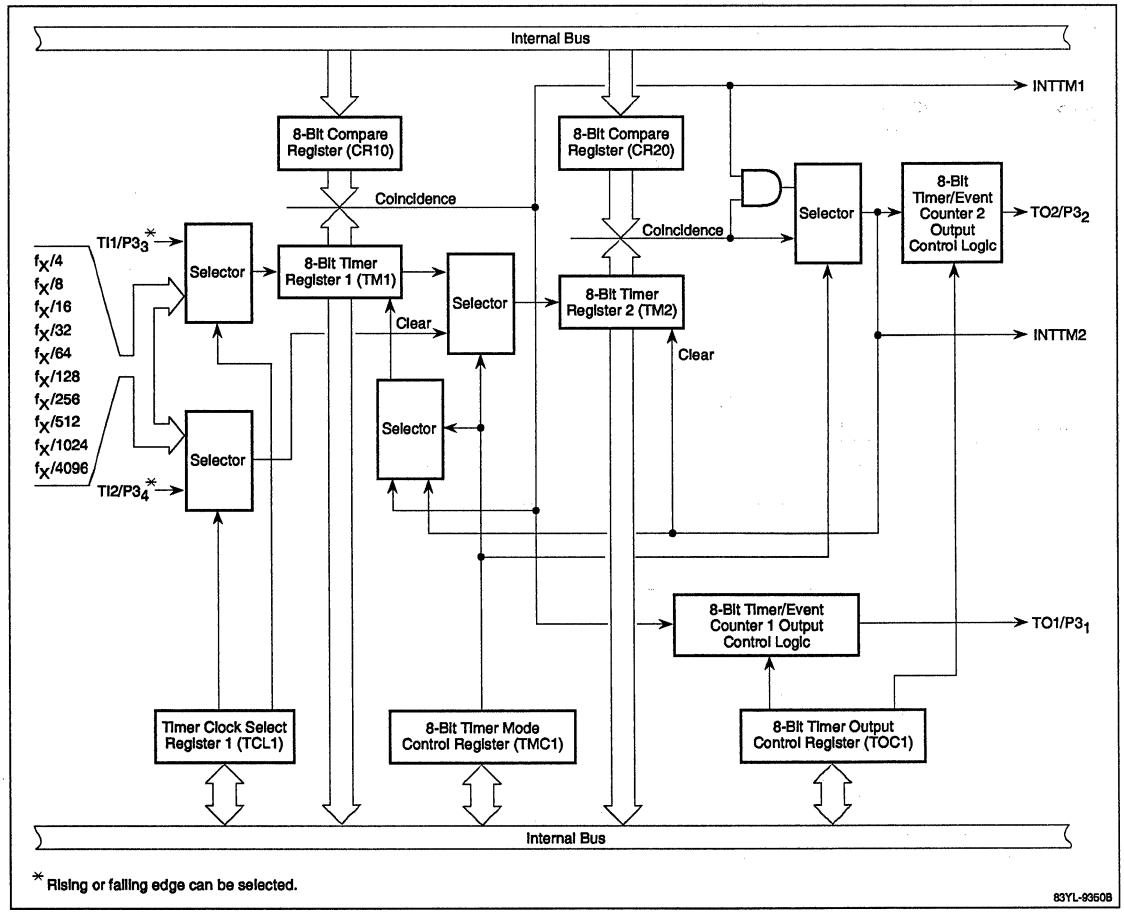
16-Bit Timer/Event Counter 0. Timer/event counter 0 (figure 9) consists of a 16-bit counter (TM0), a 16-bit compare register (CR00), a 16-bit capture register (CR01), and a timer output (TO0). Timer 0 can be used as an interval timer, to count external events on the timer input (TI0) pin, to output a programmable square wave, a 14-bit pulse width modulated output, or to measure pulse widths.

Figure 9. 16-Bit Timer/Event Counter 0



8-Bit Timer/Event Counters 1 and 2. Timer/event counters 1 and 2 (figure 10) each consist of an 8-bit timer (TM1 or TM2), an 8-bit compare register (CR10 or CR20), and timer output control logic (TO1 or TO2). The timers are controlled by registers TCL1, TMC1, and TOC1 via five selectors. Timer/event counters 1 and 2 can each be used as an 8-bit interval timer, to count external events on the timer input pins (TI1 or TI2), or to output a programmable square wave. In addition, timers 1 and 2 also can be combined as a 16-bit timer/event counter and used as a 16-bit interval timer, to count external events on TI1, or to output a programmable square wave on TO2.

Figure 10. 8-Bit Timer/Event Counters 1 and 2



Clock Timer 3. Clock timer 3 (figure 11) is a 5-bit timer that can be used as a time source to keep track of time of day, to release the STOP or HALT modes at regular intervals, or to initiate any other task that must be performed at regular intervals. When driven by the subsystem clock, the clock timer continues to operate in the STOP mode.

The clock timer can function as both an interval timer and a clock timer simultaneously. When used as a clock timer, interrupt request INTWT (not a vectored interrupts) can be generated using the main system or subsystem clock every 0.5 or 0.25 seconds.When used as an interval timer, vectored interrupt request INTTM3 is generated at preselected time intervals. With a main system clock of 8.38 MHz or a subsystem clock of 32.768 kHz, the following time intervals can be selected: 489 μs, 978 μs, 1.96 ms, 3.91 ms, 7.82 ms or 15.6 ms.

Watchdog Timer. The watchdog timer (figure 12) can be used as either a watchdog timer or an interval timer. When used as a watchdog timer, it protects against inadvertent program run-away. It can be selected to generate a nonmaskable interrupt (INTWDT), which vectors to address 0004H, or to generate an internal reset signal, which vectors to the restart address 0000H if the timer is not cleared by the program before it overflows. Eight program-selectable intervals based on the main system clock are available. With a main system clock of 8.38 MHz, the program selectable intervals are 0.489, 0.978, 1.96, 3.91, 7.82, 15.6, 31.3, and 125 ms. Once the watchdog timer is initialized and started, the timer's mode cannot be changed and the timer can only be stopped by an external reset.

When used as an interval timer, maskable interrupts (INTWDT), which vector to address 0004H, are generated repeatedly at a preset interval. The time intervals available are the same as in the watchdog timer mode.

Figure 11. Clock Timer 3

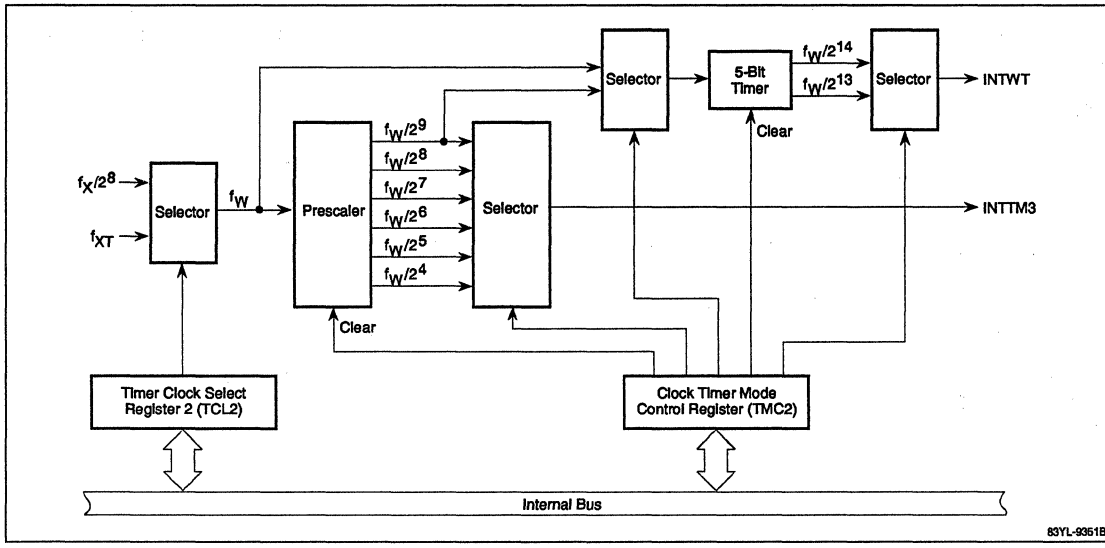
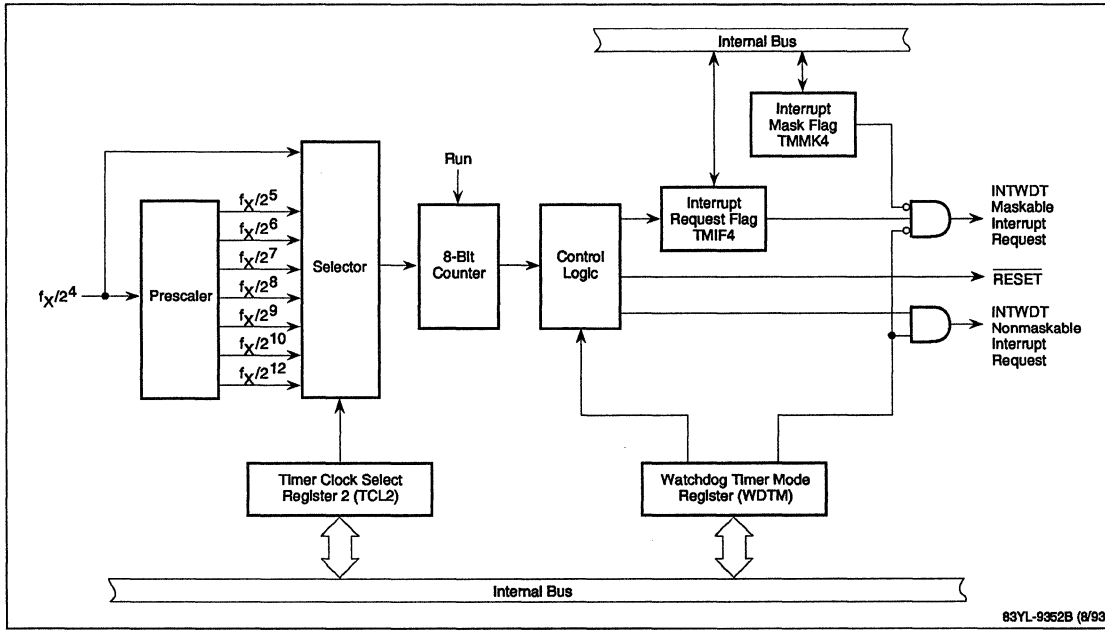


Figure 12. Watchdog Timer



Programmable Clock Output

The μPD78014 family has a programmable clock output (PCL) that can be used for carrier output for remote controlled transmissions or as a clock output for peripheral devices. The main system clock (f_X) divided by 8, 16, 32, 64, 128, or 256 or the subsystem clock (f_{XT}) can be output on the PCL pin. Frequencies of 1050, 524, 262, 131, 65.5 and 32.7 kHz are available with a main system clock of 8.38 MHz. See figure 13.

Buzzer Output

The μPD78014 family also has a programmable buzzer output (BUZ). The buzzer output frequency can be programmed to be equal to the main system clock (f_X) divided by 1024, 2048, or 4096. With a main system clock of 8.38 MHz, the buzzer can be set to 8.2, 4.1 or 2.0 kHz. See figure 14.

Figure 13. Programmable Clock Output

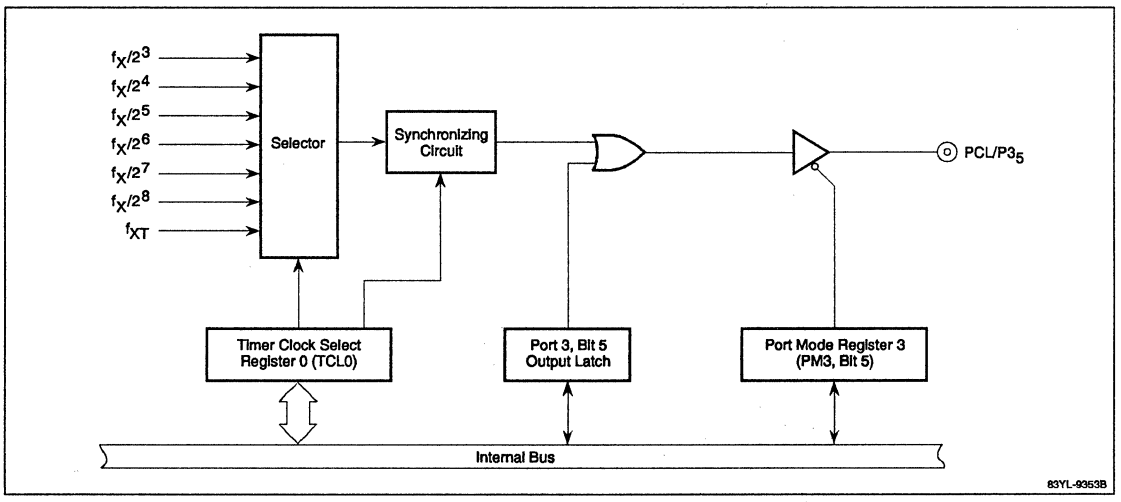
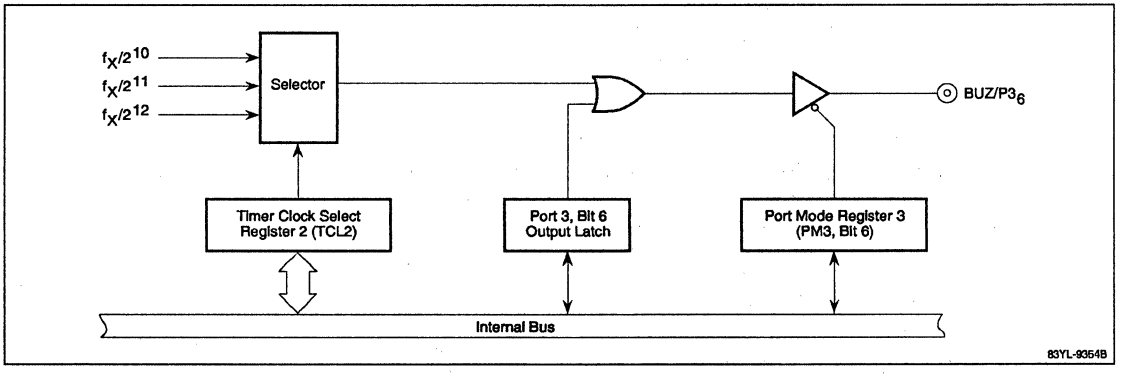


Figure 14. Buzzer Output



Interrupts

The μPD78014 family has 14 maskable hardware interrupt sources (5 external and 9 internal). Of these 14 interrupt sources, 12 cause a vectored interrupt while the 2 testable inputs only generate an interrupt request. All of the 14 maskable interrupts can be used to release the HALT mode except INTP0. INTP0 cannot be used to release the STOP mode and cannot release the HALT mode when register SCS = 0. In addition, there is one nonmaskable interrupt from the watchdog timer, one software interrupt, and a reset interrupt. The watchdog timer overflow interrupt (interrupt vector table address 0004H) can be initialized to be a nonmaskable interrupt or the highest default priority maskable interrupt. The software interrupt, generated by the BRK instruction, is not maskable. See table 3 and figure 15.

Table 3. Interrupt Sources and Vector Addresses

Type of Request	Default Priority	Signal Name	Interupt Source	Location	Vector Address	Interrupt Configuration
Restart	—	RESET	RESET input pin	External	0000H	—
	—	INTWDT	Watchdog timer overflow (when reset mode selected)	Internal		
Nonmaskable	—	INTWDT	Watchdog timer overflow (when nonmaskable interrupt selected)	Internal	0004H	A
Maskable	0	INTWDT	Watchdog timer overflow (when interval timer selected)	Internal	0004H	B
	1	INTP0	External interrupt edge detection	External	0006H	C
	2	INTP1	External interrupt edge detection	External	0008H	D
	3	INTP2	External interrupt edge detection	External	000AH	D
	4	INTP3	External interrupt edge detection	External	000CH	D
	5	INTCSI0	End of clocked serial interface 0 transfer	Internal	000EH	B
	6	INTCSI1	End of clocked serial interface 1 transfer	Internal	0010H	B
	7	INTTM3	Clock timer reference time interval signal	Internal	0012H	B
	8	INTTM0	16-bit timer/event counter coincidence signal	Internal	0014H	B
	9	INTTM1	8-bit timer/event counter 1 coincidence signal	Internal	0016H	B
	10	INTTM2	8-bit timer/event counter 2 coincidence signal	Internal	0018H	B
	11	INTAD	End of A/D Conversion	Internal	001AH	B
Software	—	—	BRK instruction	Internal	003EH	E
Test input	—	INTWT	Clock timer overflow	Internal	—	F
	—	INTPT4	Port 4 falling edge detection	External	—	F

Interrupt Servicing. The μPD78014 family provides two levels of programmable hardware priority control and services all interrupt requests, except the two testable interrupts (INTWT and INTPT4) using vectored interrupts. The programmer can choose the priority of servicing each maskable interrupt by using the interrupt control registers.

Interrupt Control Registers. The μPD78014 family has three 16-bit interrupt control registers. The interrupt request flag register (IF0) contains an interrupt request flag for each interrupt except INTPT4. The interrupt mask register (MK0) is used to enable or disable any interrupt except INTPT4. The priority flag register (PR0) can be used to specify a high or a low priority level for each interrupt except the two testable interrupts.

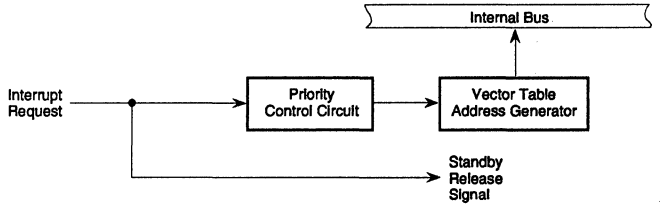
Four other 8-bit registers are associated with interrupt processing. The key return mode register (KRM) contains the KRIF interrupt request flag associated with falling-edge detection on port 4 and the KRMK mask flag used to enable or disable clearing of the standby mode if a falling edge is detected on port 4. The external interrupt mode register (INTM0) is used to select a rising, falling, or both edges as the valid edge for each of the external interrupts INTP0 to INTP2 (INTP3 is always falling edge). The sampling clock

select register (SCS) is used to select a sampling clock for the noise eliminator circuit on external interrupt INTP0.

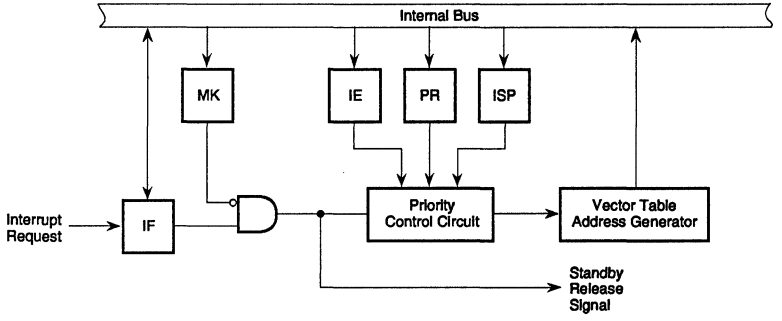
The IE and the ISP bit of the program status word are also used to control interrupts. If the IE bit is 0, all maskable interrupts are disabled. The IE bit can be set or cleared using the EI and DI instructions, respectively, or by directly writing to the PSW. The IE bit is cleared each time an interrupt is accepted. The ISP bit is used by hardware to hold the priority level flag of the interrupt being serviced.

Figure 15. Interrupt Configurations

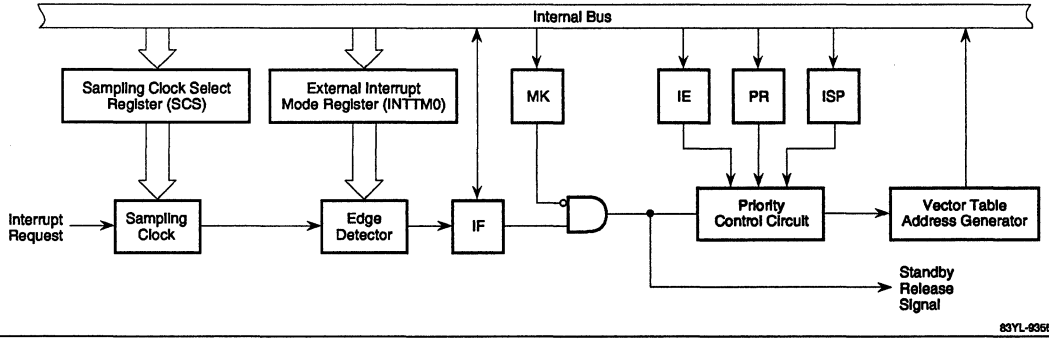
Type A: Internal nonmaskable Interrupt



Type B: Internal maskable Interrupt



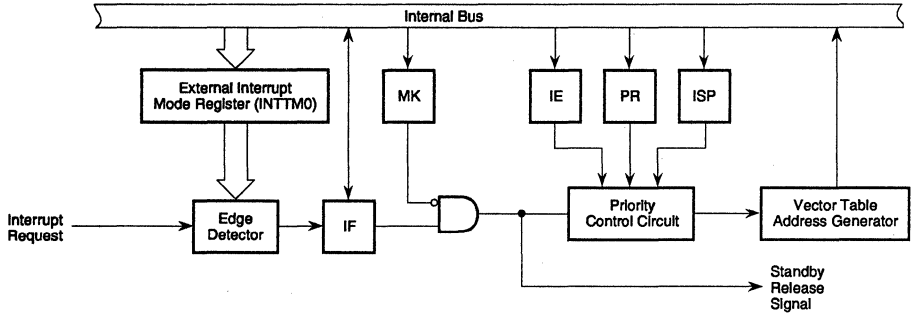
Type C: External maskable Interrupt (INTP0)



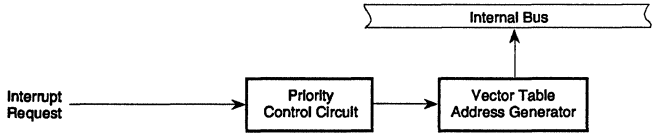
3c

Figure 15. Interrupt Configurations (cont)

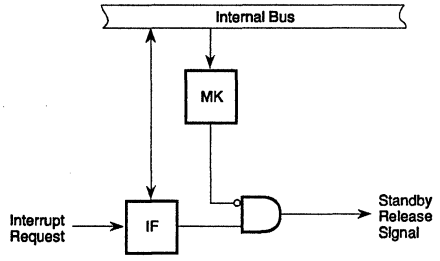
Type D: External maskable interrupt (except INTP0)



Type E: Software interrupt



Type F: Test input



Abbreviations:

IF: Interrupt request flag
IE: Interrupt enable flag
ISP: In-service priority flag
MK: Interrupt mask flag
PR: Priority specify flag

83YL-9356B (2)

Interrupt Priority. If the watchdog timer overflow interrupt (INTWDT) has been initialized to be a nonmaskable interrupt, it has priority over all other interrupts. Two hardware-controlled priority levels are available for all maskable interrupts that generate a vectored interrupt (i.e., all except the two testable interrupts). Either a high or a low priority level can be assigned by software to each of the maskable interrupts. Interrupt requests of the same priority or a priority higher than the processor's current priority level are held pending until interrupts in the current service routine are enabled by software or until one instruction has been executed after returning from the current service routine. Interrupt requests of a lower priority are always held pending until one instruction has been executed after returning from the current service routine.

The default priorities listed in table 3 are fixed by hardware and are effective only when it is necessary to choose between two interrupt requests of the same software-assigned priority. For example, the default priorities would be used after the completion of a high-priority routine, if two interrupts of the same software priority were pending.

The software interrupt, initiated by the BRK instruction, is executed regardless of the processor's priority level and the state of the IE bit. It does not alter the processor's priority level.

Vectored Interrupt Servicing. When a vectored interrupt is acknowledged, the program status word and the program counter are saved on the stack, the processor's priority is set to that specified for the interrupt, the IE bit in the PSW is set to zero, and the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (RETB instruction for the software interrupt) reverses the process and the μPD78014 family microcontroller resumes the interrupted routine.

Standby Modes

HALT, STOP, and data retention modes are provided to reduce power consumption when CPU action is not required.

The HALT mode is entered by executing a HALT instruction while the CPU is operating from the main system or subsystem clock. In HALT mode, the CPU clock is stopped while the main system and the subsystem clock continue to run. The HALT mode is released by any unmasked interrupt request (except INTP0 if register SCS = 0), a nonmaskable interrupt request , an unmasked test input, or an external reset pulse.

Power consumption may be further reduced by using the STOP mode. The STOP mode is entered by executing a STOP instruction while operating from the main system clock. In STOP mode, the main system clock input pin X1 is internally grounded stopping both the CPU and the peripheral hardware clock. The STOP mode is released by any unmasked interrupt request except INTP0, a nonmaskable interrupt request, an unmasked test input, or an external reset pulse. Any peripheral using the main oscillator as a clock source will also be disabled in the STOP mode and interrupts from such a peripheral cannot be used to exit the STOP mode. Table 4 summarizes both the HALT and STOP standby modes.

3c

Table 4. Standby Mode Operation Status

Item	HALT Mode	STOP Mode
Setting instruction	HALT instruction	STOP instruction
System clock when setting	Main system or subsystem clock	Main system clock
Clock oscillator	Main system and subsystem clocks can oscillate; CPU clock is stopped.	Subsystem clock can oscillate; CPU clock and main system clock are stopped.
CPU	Operation stopped	Operation stopped
Ports	Maintain previous state	Maintain previous state
16-bit timer/event counter	Operational from main system clock	Operation stopped
8-bit timer/event counters	Operational from main system clock	Operational only with TI1 and TI2 as count clock
Clock timer	Operational from main system clock and with f_{XT} as count clock	Operational only with f_{XT} as count clock
Watchdog timer	Operational from main system clock	Operation stopped
Serial interface 0	Operational from main system clock	Operational only with external clock
Serial interface 1	Operational from main system clock; no automatic transmit/ receive mode	Operational only with external clock; no automatic transmit/ receive mode
A/D converter	Operational from main system clock	Operation stopped
External interrupts	Operational except for INTP0 when its sampling clock is based on the CPU clock	INTP0 not operational; INTP1 to INTP3 operational

When exiting the STOP mode, a wait time occurs before the CPU begins execution to allow the main system clock oscillator circuit to stabilize. The oscillator stabilization time is selected by programming the OSTS register with one of five values before entering the STOP mode. The values range from 0.98 msec to 31.3 msec at f_X = 8.38 MHz. Once in the STOP mode, power consumption can be further minimized by lowering the power supply voltage V_{DD} to 2 V. This places the device in the data retention mode. The contents of internal RAM and the registers are retained. This mode is released by first raising V_{DD} to the proper operating range and then releasing the STOP mode.

External Reset

The μPD78014 family is reset by taking the $\overline{\text{RESET}}$ pin low or by an overflow of the watchdog timer (if enabled). The $\overline{\text{RESET}}$ input pin is a schmitt-trigger input with hysteresis characteristics to protect against spurious system resets caused by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low for a minimum of 10 μs after the power supply reaches its operating voltage.

There is no functional difference between an external reset and an internal reset caused by the overflow of the watchdog timer. In both cases, the main system clock oscillation is stopped and the subsystem clock oscillation continues. During reset, the program counter is loaded with the address contained in the reset vector (addresses 0000H, 0001H). Once the reset is cleared and the oscillation stabilization time of $2^{18}/f_X$ has elapsed, program execution starts at that address.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = +25°C

Supply voltage, V_{DD}	−0.3 to +7.0 V
Supply voltage, V_{PP}	−0.3 to +13.5 V
Supply voltage, AV_{DD}	−0.3 to V_{DD}+0.3 V
Supply voltage, AV_{REF}	−0.3 to V_{DD}+0.3 V
Supply voltage, AV_{SS}	−0.3 to +0.3 V
Input voltage, V_{I1} (except $P6_0$ to $P6_3$)	−0.3 to V_{DD}+0.3 V
Input voltage, V_{I2} ($P6_0$ to $P6_3$; open drain)	−0.3 to +16 V
Output voltage, V_O	−0.3 to V_{DD}+0.3 V
Analog input voltage, V_{AN} (port 1; analog input pin)	AV_{SS} −0.3 to AV_{REF}+0.3 V
Output current, high; I_{OH}	
Each output pin	−10 mA
Total: ports 2 and 3	−15 mA
Total: port 0 and ports 4 to 6	−15 mA
Output current, low, I_{OL} †	
Each output pin	30 mA peak, 15 mA rms
Total: $P4_0$ to $P4_7$ and $P5_0$ to $P5_5$	100 mA peak, 70 mA rms
Total: $P0_1$ to $P0_3$, $P5_6$, $P5_7$, and $P6_0$ to $P6_7$	100 mA peak, 70 mA rms
Total: $P0_1$ to $P0_3$ and $P6_4$ to $P6_7$	50 mA peak, 20 mA rms
Total: ports 2 and 3	50 mA peak, 20 mA rms
Operating temperature, T_{OPT}	−40 to +85°C
Storage temperature, T_{STG}	−65 to +150°C

† rms value = peak value x (duty cycle) $^{1/2}$

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC characteristics.

Capacitance

T_A = +25°C; V_{DD} = V_{SS} = 0 V

Parameter	Symbol	Max	Unit	Conditions	
Input capacitance	C_{IN}	15	pF	Except $P6_0$ to $P6_3$	f = 1 MHz; unmeasured pins returned to ground
		20	pF	$P6_0$ to $P6_3$	
Output capacitance	C_{OUT}	15	pF	Except $P6_0$ to $P6_3$	
		20	pF	$P6_0$ to $P6_3$	
Input/output capacitance	C_{IO}	15	pF	Except $P6_0$ to $P6_3$	
		20	pF	$P6_0$ to $P6_3$	

Main System Clock Oscillator

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 16.

Type	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Figure 16A)	Oscillation frequency (Note 1)	f_X	1.0		10.0	MHz	V_{DD} = oscillator voltage range
	Oscillation stabilization time (Note 2)				4.0	ms	After V_{DD} reaches oscillator operating voltage
Crystal resonator (Figure 16A)	Oscillation frequency (Note 1)	f_X	1.0	8.38	10.0	MHz	
	Oscillation stabilization time (Note 2)				10	ms	V_{DD} = 4.5 to 6.0 V
					30	ms	
External clock (Figure 16B)	X1 input frequency (Note 1)	f_X	1.0		10.0	MHz	
	X1 input high/low-level width	t_{XH}, t_{XL}	50		500	ns	

Notes:

(1) Oscillator and X1 input frequencies are included only to show the oscillator characteristics. Refer to the AC Characteristics table for actual instruction execution times.

(2) Time required for the oscillator to stabilize after reset or STOP mode is released. The values shown are for the recommended resonators. Values for resonators not shown in this data sheet should be obtained from the manufacturer's specification sheets.

3c

Subsystem Clock Oscillator

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 17.

Type	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Crystal resonator (Figure 17A)	Oscillation frequency (Note 1)	f_{XT}	32	32.768	35	kHz	
	Oscillation stabilization time (Note 2)			1.2	2	s	V_{DD} = 4.5 to 6.0 V
					10	s	
External clock (Figure 17B)	XT1 input frequency (Note 1)	f_{XT}	32		100	kHz	
	XT1 input high/low-level width	t_{XTH}, t_{XTL}	5		15	μs	

Notes:

(1) The oscillator and XT1 input frequencies are included only to show the oscillator characteristics. Refer to the AC Characteristics table for actual instruction execution times.

(2) Time required for the oscillator to stabilize after reset or STOP mode is released. The values shown are for the recommended resonators. Values for resonators not shown in this data sheet should be obtained from the manufacturer's specification sheets.

Figure 16. Main System Clock Configurations

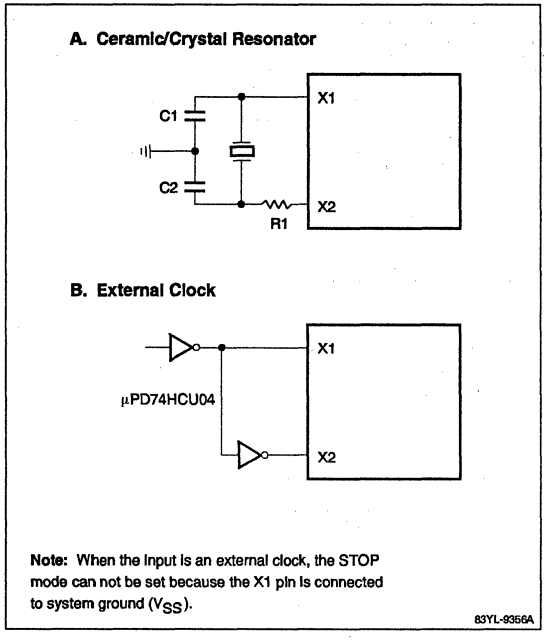
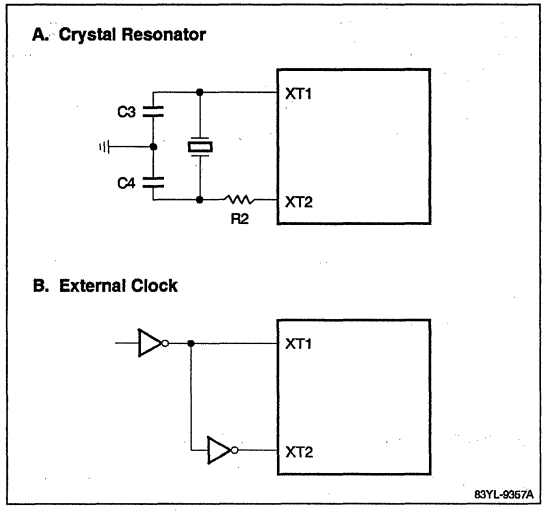


Figure 17. Subsystem Clock Configurations



Recommended Main System Clock Ceramic Resonators

T_A = −40 to +85°C, refer to figure 16a

Part Number (Notes 1 and 2)	Recommended Circuit Constant			Oscillator Voltage Range		Frequency (MHz)
	C1 (pF)	C2 (pF)	R1 (kΩ)	Min (V)	Max (V)	
CSB1000J	100	100	6.8	2.7 (Note 3) 2.8 (Note 4)	6.0	1.00
CSBxxxxJ	100	100	4.7	2.7 (Note 3) 2.8 (Note 4)	6.0	1.01 to 1.25
CSAx.xxxMK	100	100	0	2.7 (Note 3) 2.8 (Note 4)	6.0	1.26 to 1.79
CSAx.xxMG (Note 3)	100	100	0	2.7	6.0	1.8 to 2.44
CSAx.xxMG093(Note 4)	100	100	0	2.7	6.0	
CSTx.xxMG (Note 3)	(Note 5)	(Note 5)	0	2.7	6.0	
CSTx.xxMG093 (Note 4)	0 (Note 5)	0 (Note 5)	0	2.7	6.0	
CSAx.xxMG	30	30	0	2.7	6.0	2.45 to 4.18
CSTx.xxMGW	(Note 5)	(Note 5)	0	2.7	6.0	
CSAx.xxMG (Note 3)	30	30	0	2.7	6.0	4.19 to 6.00
CSAx.xxMGU(Note 4)	30	30	0	2.7	6.0	
CSTx.xxMGW (Note 3)	(Note 5)	(Note 5)	0	2.7	6.0	
CSTx.xxMGWU (Note 4)	0 (Note 5)	0 (Note 5)	0	2.7	6.0	
CSAx.xxMT	30	30	0	2.7 (Note 3) 3.0 (Note 4)	6.0	6.01 to 10.0
CSTx.xxMTW	(Note 5)	(Note 5)	0	2.7 (Note 3) 3.0 (Note 4)	6.0	

Notes:

(1) Manufactured by Murata Mfg. Co., Ltd.

(2) x.xx indicates frequency

(3) μPD7801x only

(4) μPD78P014 only

(5) C1 and C2 are contained in the ceramic resonators.

Recommended Subsystem Clock Crystal Resonators (μPD7801x)

T_A = −40 to +60°C , refer to figure 17a

Part Number †	Frequency (kHz)	Recommended Circuit Constant			Oscillator Voltage Range	
		C3 (pF)	C4 (pF)	R2 (kΩ)	Min (V)	Max (V)
DT-38 (1TA252 E00, load capacitance 6.3 pF)	32.768	12	12	100	2.7	6.0

† Manufactured by Daishinku

DC Characteristics

T_A = −40 to +85°C; V_{DD} = +2.7 to 6.0 V; refer to figures 18-23

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Other than below
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	$P0_0$ to $P0_4$, $P2_0$, $P2_2$, $P2_4$ to $P2_7$, $P3_3$, $P3_4$, $\overline{RESET}$
	V_{IH3}	0.7 V_{DD}		15	V	$P6_0$ to $P6_3$; open-drain
	V_{IH4}	V_{DD} − 0.5		V_{DD}	V	X1, X2
	V_{IH5}	V_{DD} − 0.5		V_{DD}	V	V_{DD} = 4.5 to 6.0 V; XT1, XT2
		V_{DD} − 0.3		V_{DD}	V	μPD7801x; XT1, XT2
		V_{DD} − 0.2		V_{DD}	V	μPD78P014

3c

DC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Low-level input voltage	V_{IL1}	0		0.3 V_{DD}	V	Other than below
	V_{IL2}	0		0.2 V_{DD}	V	$P0_0$ to $P0_4$, $P2_0$, $P2_2$, $P2_4$ to $P2_7$, $P3_3$, $P3_4$, $\overline{RESET}$
	V_{IL3}	0		0.3 V_{DD}	V	$P6_0$ to $P6_3$; V_{DD} = 4.5 to 6.0 V
		0		0.2 V_{DD}	V	$P6_0$ to $P6_3$
	V_{IL4}	0		0.4	V	X1, X2
	V_{IL5}	0		0.4	V	XT1, XT2; V_{DD} = 4.5 to 6.0 V
		0		0.3	V	XT1, XT2
High-level output voltage	V_{OH1}	V_{DD} – 1.0			V	V_{DD} = 4.5 to 6.0 V, I_{OH} = –1.0 mA
		V_{DD} – 0.5			V	I_{OH} = –100 μA
Low-level output voltage	V_{OL1}		0.4	2.0	V	$P5_0$ to $P5_7$, $P6_0$ to $P6_3$; V_{DD} = 4.5 to 6.0 V, I_{OL} = 15 mA
				0.4	V	Other than above; V_{DD} = 4.5 to 6.0 V, I_{OL} = 1.6 mA
	V_{OL2}			0.2 V_{DD}	V	SB0, SB1, $\overline{SCK0}$; V_{DD} = 4.5 to 6.0 V, open-drain, pullup resistance = 1 kΩ
	V_{OL3}			0.5	V	I_{OL} = 400 μA
High-level input leakage current	I_{LIH1}			3	μA	V_{IN} = V_{DD}; except X1, X2, XT1, XT2
	I_{LIH2}			20	μA	V_{IN} = V_{DD}; X1, X2, XT1, XT2
	I_{LIH3}			80	μA	V_{IN} = 15 V; $P6_0$ to $P6_3$
Low-level input leakage current	I_{LIL1}			–3	μA	V_{IN} = 0 V; except X1, X2, XT1, XT2
	I_{LIL2}			–20	μA	V_{IN} = 0 V; X1, X2, XT1, XT2
	I_{LIL3}			–3	μA	V_{IN} = 0 V; $P6_0$ to $P6_3$ (Note 1)
Output leakage current high	I_{LOH1}			3	μA	V_{OUT} = V_{DD}
Output leakage current low	I_{LOL}			–3	μA	V_{OUT} = 0 V
Mask option pullup resistor	R_1	20	40	90	kΩ	V_{IN} = 0 V; $P6_0$ TO $P6_3$, μPD7801x only
Software pullup resistor	R_2	15	40	90	kΩ	V_{DD} = 4.5 to 6.0 V; V_{IN} = 0 V, $P0_1$ to $P0_3$, ports 1 to 5, $P6_4$ to $P6_7$
		20		500	kΩ	V_{DD} = =2.7 to 4.5 V; V_{IN} = 0 V, $P0_1$ to $P0_3$, ports 1 to 5, $P6_4$ to $P6_7$
Power supply current (μPD7801x)	I_{DD1}		7.5	22.5	mA	8.38 MHz crystal oscillation operating mode; V_{DD} = 5.0 V ±10% (Note 2)
			0.8	2.4	mA	8.38 MHz crystal oscillation operating mode; V_{DD} = 3.0 V ±10% (Note 3)
	I_{DD2}		1.4	4.2	mA	8.38 MHz crystal oscillation HALT mode; V_{DD} = 5.0 V ±10%
			550	1650	μA	8.38 MHz crystal oscillation HALT mode; V_{DD} = 3.0 V ±10%
	I_{DD3}		60	120	μA	32.768 kHz crystal oscillation operating mode; V_{DD} = 5.0 V ±10%, X1 STOP mode, CPU operating from subsystem clock
			35	70	μA	32.768 kHz crystal oscillation operating mode; V_{DD} = 3.0 V ±10%, X1 STOP mode, CPU operating from subsystem clock

DC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Power supply current (μPD7801x) (cont)	I_{DD4}		25	50	μA	32.768 kHz crystal oscillation HALT mode; V_{DD} = 5.0 V ±10%, X1 STOP mode
			5	10	μA	32.768 kHz crystal oscillation HALT mode; V_{DD} = 3.0 V ±10%, X1 STOP mode
	I_{DD5}		1	20	μA	XT1 = 0 V STOP mode when feedback resistor is connected; V_{DD} = 5.0 V ±10%
			0.5	10	μA	XT1 = 0 V STOP mode when feedback resistor is connected; V_{DD} = 3.0 V ±10%
	I_{DD6}		0.1	20	μA	XT1 = 0 V STOP mode when feedback resistor is disconnected; V_{DD} = 5.0 V ±10%
			0.05	10	μA	XT1 = 0 V STOP mode when feedback resistor is disconnected; V_{DD} = 3.0 V ±10%
Power supply current (μPD78P014)	I_{DD1}		9	27	mA	8.38 MHz crystal oscillation operating mode; V_{DD} = 5.0 V ±10% (Note 2)
			1	3	mA	8.38 MHz crystal oscillation operating mode; V_{DD} = 3.0 V ±10% (Note 3)
	I_{DD2}		1.4	4.2	mA	8.38 MHz crystal oscillation HALT mode; V_{DD} = 5.0 V ±10%
			550	1650	μA	8.38 MHz crystal oscillation HALT mode; V_{DD} = 3.0 V ±10%
	I_{DD3}		90	180	μA	32.768 kHz crystal oscillation operating mode; V_{DD} = 5.0 V ±10%, X1 STOP mode, CPU operating from subsystem clock
			50	100	μA	32.768 kHz crystal oscillation operating mode; V_{DD} = 3.0 V ±10%, X1 STOP mode, CPU operating from subsystem clock
	I_{DD4}		25	50	μA	32.768 kHz crystal oscillation HALT mode; V_{DD} = 5.0 V ±10%, X1 STOP mode
			5	10	μA	32.768 kHz crystal oscillation HALT mode; V_{DD} = 3.0 V ±10%, X1 STOP mode
	I_{DD5}		1	30	μA	XT1 = 0 V STOP mode when feedback resistor is connected; V_{DD} = 5.0 V ±10%
			0.5	10	μA	XT1 = 0 V STOP mode when feedback resistor is connected; V_{DD} = 3.0 V ±10%
	I_{DD6}		0.1	30	μA	XT1 = 0 V STOP mode when feedback resistor is disconnected; V_{DD} = 5.0 V ±10%
			0.05	10	μA	XT1 = 0 V STOP mode when feedback resistor is disconnected; V_{DD} = 3.0 V ±10%

Notes:

(1) $P6_0$ to $P6_3$ become −200 μA (max.) for only 1 clock cycle during input instruction execution (no wait) and −3 μA (max.) during instruction other than input.

(2) When operated in the high-speed mode with the processor clock control register set to 00H.

(3) When operated in low-speed mode with the processor clock control register set to 04H.

3c

Figure 18. I_{DD} vs V_{DD} (μPD7801x)

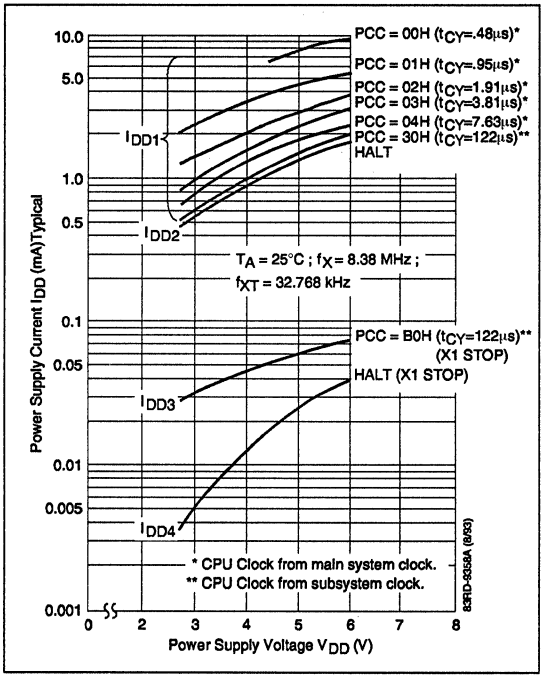


Figure 19. I_{DD} vs V_{DD} (μPD78P014)

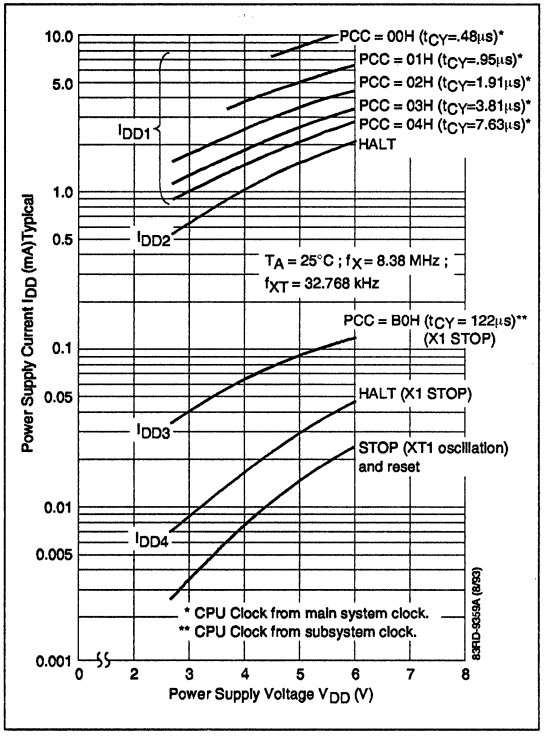


Figure 20. I_{OL} vs V_{OL} (Ports 0, 2-5, $P6_4$-$P6_7$)

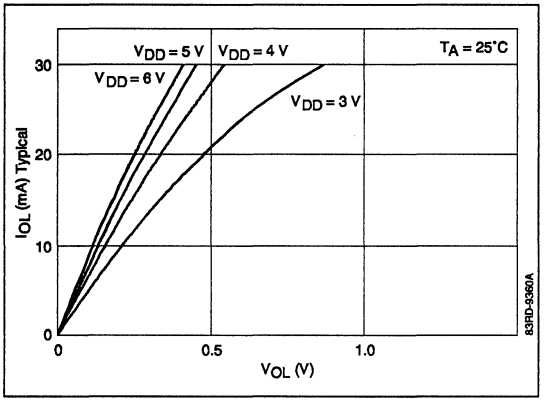


Figure 21. I_{OL} vs V_{OL} (Port 1)

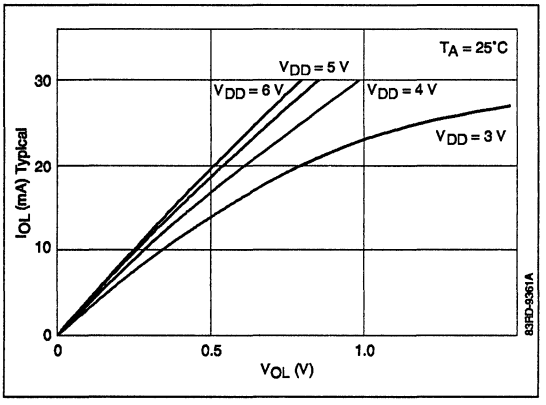


Figure 22. I_{OL} vs V_{OL} ($P6_0$-$P6_3$)

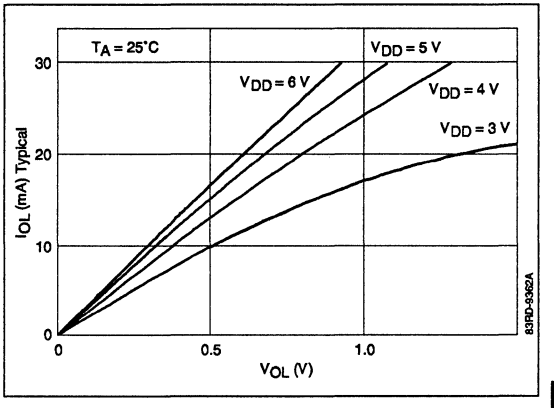
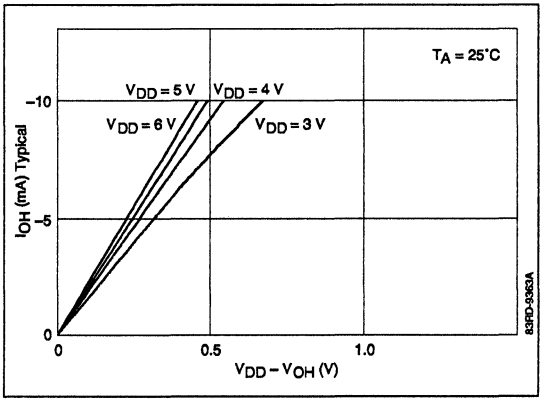


Figure 23. I_{OH} vs $V_{DD} - V_{OH}$ (Ports 0-5, $P6_4$-$P6_7$)



3c

AC Characteristics

T_A = −40 to +85°C; V_{DD} = 2.7 to 6 V ; refer to figures 24 through 30

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time (Min. instruction execution time)	t_{CY}	0.4		64	μs	V_{DD} = 4.5 to 6.0 V; operating on main system clock (μPD7801x)
		0.96		64	μs	Operating on main system clock (μPD7801x)
		0.48		64	μs	V_{DD} = 4.5 to 6.0 V; operating on main system clock (μPD78P014)
		1.91		64	μs	Operating on main system clock (μPD78P014)
		0.4		64	μs	T_A = −40 to +40°C, V_{DD} = 4.75 to 6.0 V; operating on main system clock (μPD78P014)
		0.96		64	μs	T_A = −40 to +40°C; operating on main system clock (μPD78P014)
		114	122	125	μs	Operating on subsystem clock
TI input frequency	f_{TI}	0		4	MHz	V_{DD} = 4.5 to 6.0 V
		0		275	kHz	
TI input high/low-level width	t_{TIH}, t_{TIL}	100			ns	V_{DD} = 4.5 to 6.0 V
		1.8			μs	
Interrupt input high/low-level width	t_{INTH}, t_{INTL}	$8/f_{sam}$ (Note 1)			μs	INTP0
		10			μs	INTP1 to INTP3
		10			μs	KR0 to KR7 (Note 2)
$\overline{\text{RESET}}$ low-level width	t_{RST}	10			μs	

Notes:

(1) By using bits 0 and 1 of the sampling clock select (SCS) register in conjunction with bits 0 to 2 of the processor clock control (PCC) register, f_{sam} can be set to $f_X/2^{N+1}$ (where N = 0 to 4), $f_X/64$, or $f_X/128$.

(2) Port 4 falling-edge detection input.

Figure 24. Main System Clock Operation t_{CY} vs V_{DD} (μPD7801x)

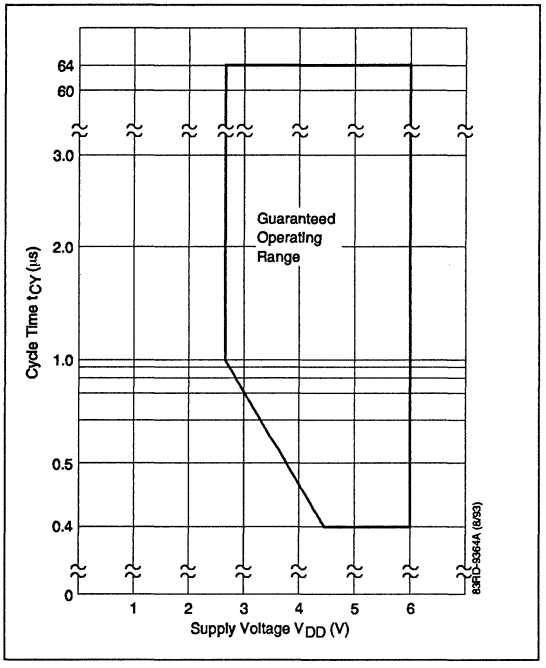
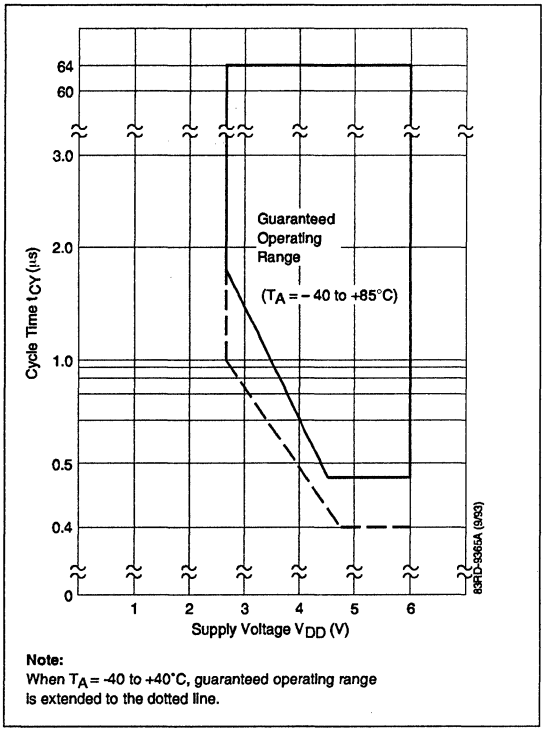


Figure 25. Main System Clock Operation t_{CY} vs V_{DD} (μPD78P014)



Note:
When T_A = -40 to +40°C, guaranteed operating range is extended to the dotted line.

3c

Figure 26. AC Timing Measurements Points (except X1 and XT1)

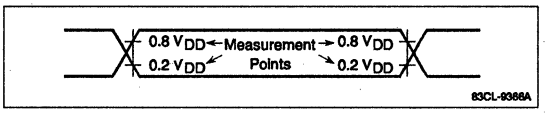


Figure 27. Clock AC Timing Points X1 and XT1

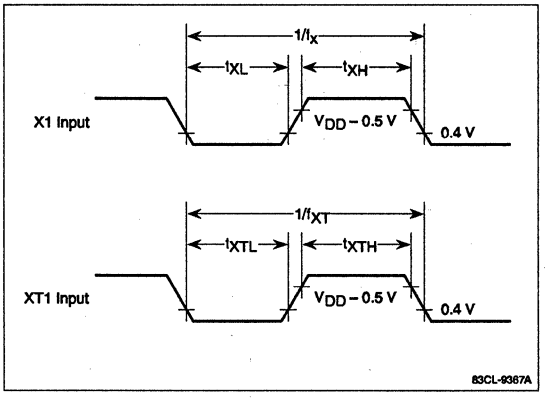


Figure 28. TI Timing

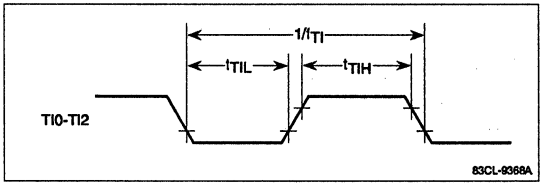


Figure 29. Interrupt Input Timing

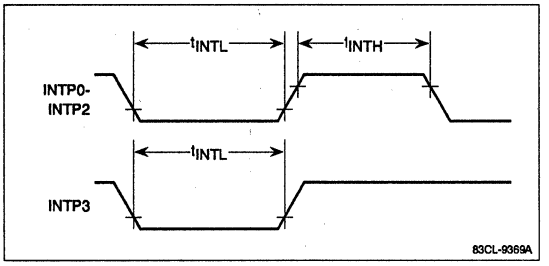
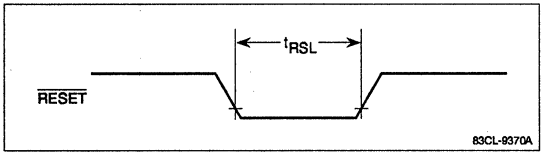


Figure 30. $\overline{\text{RESET}}$ Input Timing



Read/Write Operation

T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 V; refer to figures 31 through 34

Parameter	Symbol	Min	Max	Unit	Conditions
ASTB high-level width	t_{ASTH}	0.5 t_{CY}		ns	
Address setup time to ASTB ↓	t_{ADS}	0.5 t_{CY} − 30		ns	
Address hold time from ASTB ↓	t_{ADH}	10		ns	Load resistor ≥ 5 kΩ
Data input time from address	t_{ADD1}		(2+2n)t_{CY} − 50	ns	Instruction fetch
	t_{ADD2}	5	(3+2n)t_{CY} − 100	ns	Data access
Data input time from $\overline{RD}$ ↓	t_{RDD1}		(1+2n)t_{CY} − 25	ns	Instruction fetch
	t_{RDD2}		(2.5+2n)t_{CY} − 100	ns	Data access
Read data hold time	t_{RDH}	0		ns	
$\overline{RD}$ low-level width	t_{RDL1}	(1.5+2n)t_{CY} − 20		ns	Instruction fetch
	t_{RDL2}	(2.5+2n)t_{CY} − 20		ns	Data access
$\overline{WAIT}$ ↓ input time from $\overline{RD}$ ↓	t_{RDWT1}		0.5 t_{CY}	ns	Instruction fetch
	t_{RDWT2}		1.5 t_{CY}	ns	Data access
$\overline{WAIT}$ ↓ input time from $\overline{WR}$ ↓	t_{WRWT}		0.5 t_{CY}	ns	
$\overline{WAIT}$ low-level width	t_{WTL}	(0.5+2n)t_{CY}+10	(2+2n)t_{CY}	ns	
Write data setup time to $\overline{WR}$ ↑	t_{WDS}	100		ns	
Write data hold time from $\overline{WR}$ ↑	t_{WDH}	5		ns	
$\overline{WR}$ low-level width	t_{WRL1}	(2.5+2n)t_{CY} − 20		ns	
$\overline{RD}$ ↓ delay time from ASTB ↓	t_{ASTRD}	0.5 t_{CY} − 30		ns	
$\overline{WR}$ ↓ delay time from ASTB ↓	t_{ASTWR}	1.5 t_{CY} − 30		ns	
ASTB ↑ delay time from $\overline{RD}$ ↑ (external fetch)	t_{RDAST}	t_{CY} − 10	t_{CY}+40	ns	
Address hold time from $\overline{RD}$ ↑ (external fetch)	t_{RDADH}	t_{CY}	t_{CY}+50	ns	
Write data output time from $\overline{RD}$ ↑	t_{RDWD}	10		ns	
WR ↓ delay time from write data	t_{WDWR}	0.5 t_{CY} − 120	0.5 t_{CY}	ns	V_{DD} = 4.5 to 6.0 V
		0.5 t_{CY} − 170	0.5 t_{CY}	ns	
Address hold time from $\overline{WR}$ ↑	t_{WRADH}	t_{CY}	t_{CY}+60	ns	V_{DD} = 4.5 to 6.0 V
		t_{CY}	t_{CY}+100	ns	
$\overline{RD}$ ↑ delay time from $\overline{WAIT}$ ↑	t_{WTRD}	0.5 t_{CY}	2.5 t_{CY}+80	ns	
$\overline{WR}$ ↑ delay time from $\overline{WAIT}$ ↑	t_{WTWR}	0.5 t_{CY}	2.5 t_{CY}+80	ns	

Notes:

(1) $t_{CY} = t_{CY}/4$

(2) n indicates number of waits.

(3) C_L = 100 pF

3c

Figure 31. Read Operation; External Fetch (No Wait)

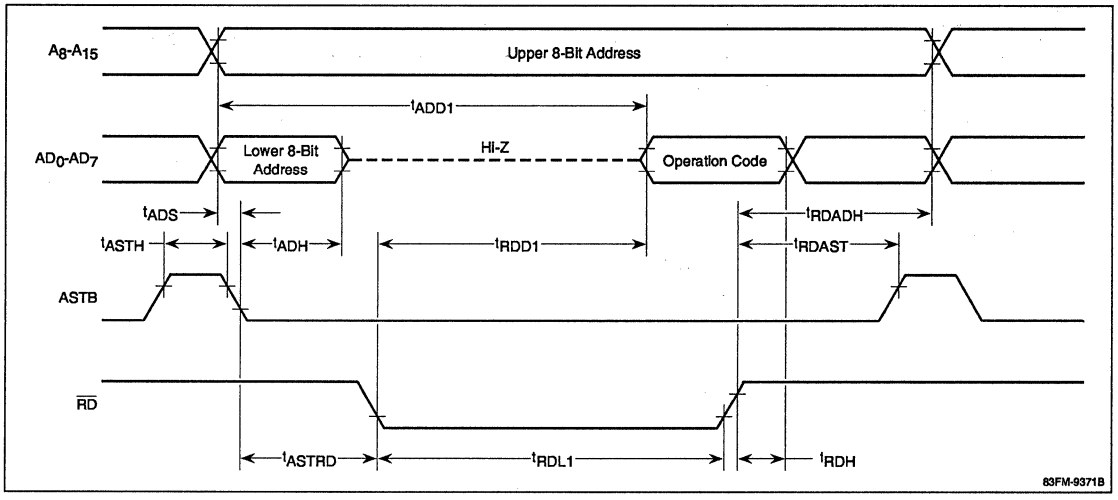


Figure 32. Read Operation; External Fetch (Wait Insertion)

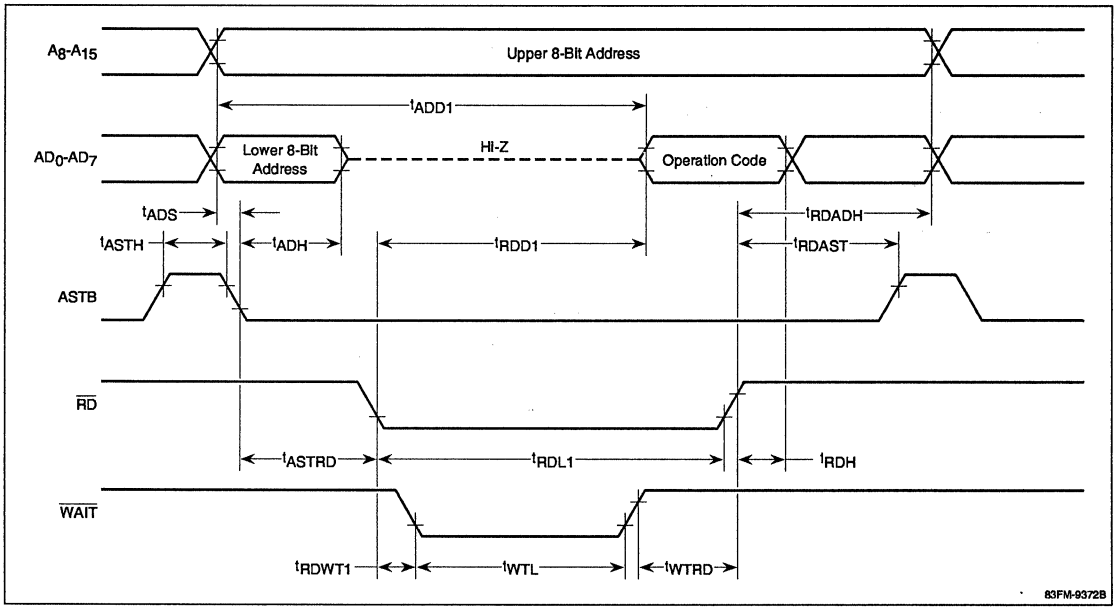


Figure 33. Read/Write Operation; External Data Access (No Wait)

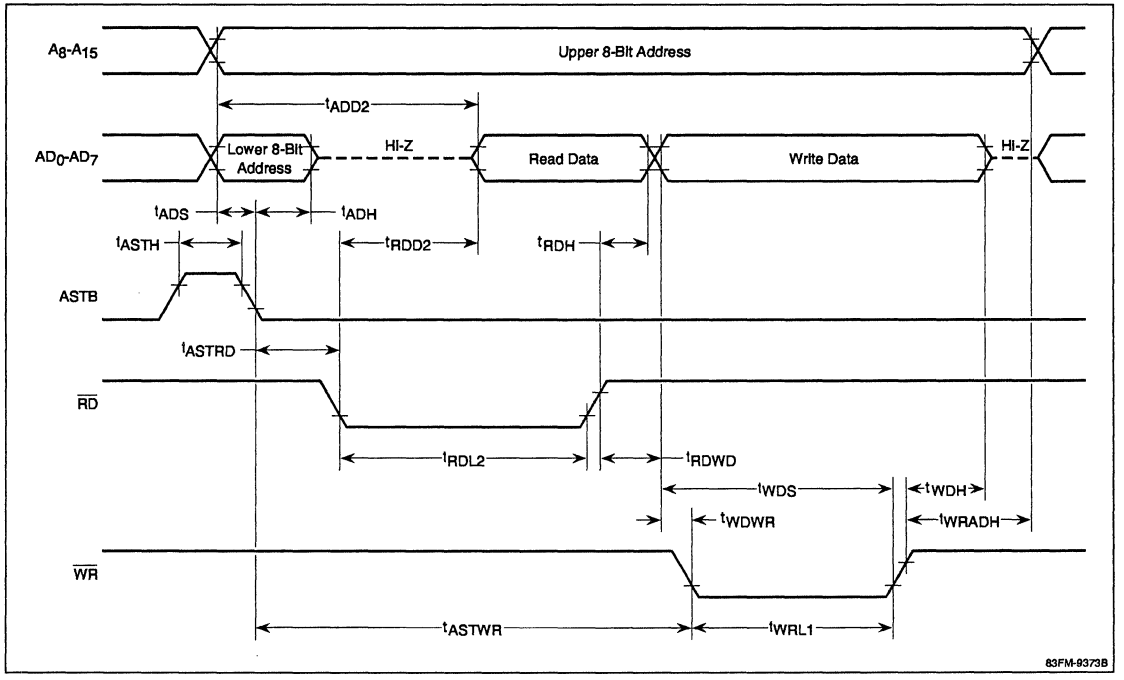
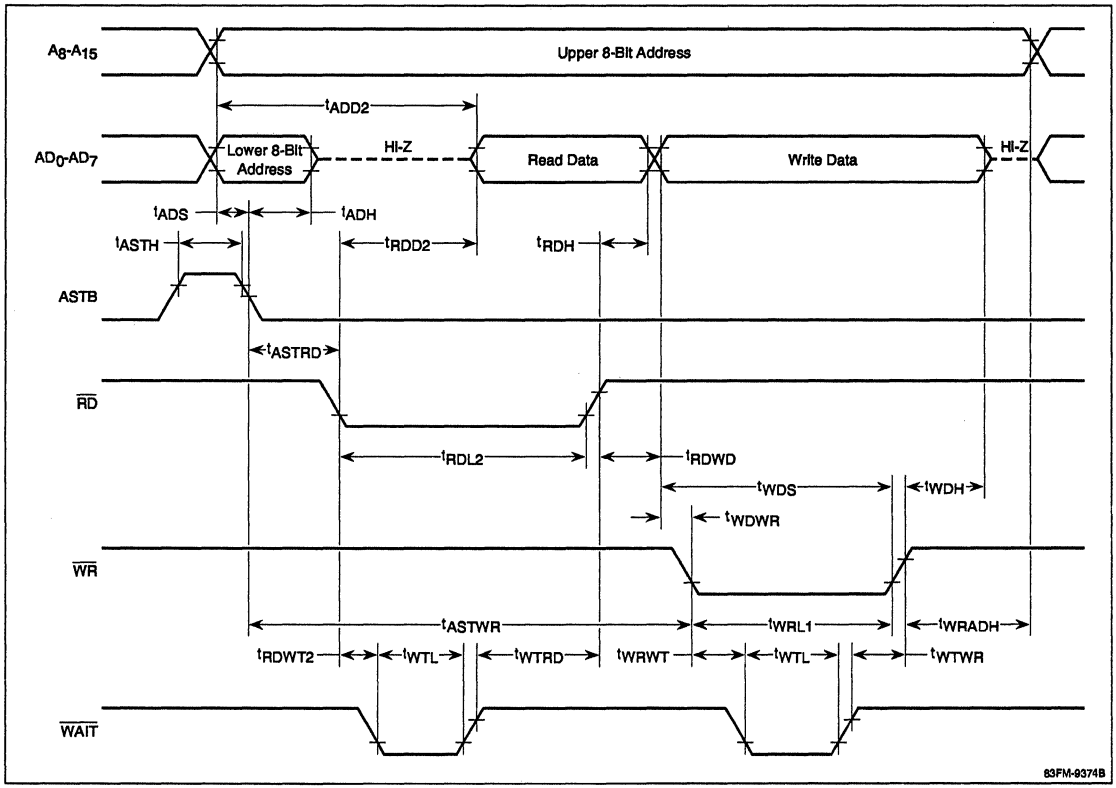


Figure 34. Read/Write Operation; External Data Access (Wait Insertion)



Serial Interface, 3-Wire, I/O Mode; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 35

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY1}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH1}, t_{KL1}	$t_{KCY1}/2 - 50$			ns	V_{DD} = 4.5 to 6.0 V
		$t_{KCY1}/2 - 150$			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK1}	100			ns	
SI hold time from $\overline{SCK}$ ↑	t_{KSI1}	400			ns	
SO output delay time from $\overline{SCK}$ ↓	t_{KSO1}			300	ns	V_{DD} = 4.5 to 6.0 V; C = 100 pF (Note 1)
				1000	ns	C = 100 pF (Note 1)

Note 1: C is the load capacitance of the SO output line.

Serial Interface, 3-Wire, I/O Mode; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 35

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY2}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH2}, t_{KL2}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK2}	100			ns	
SI hold time from $\overline{SCK}$ ↑	t_{KSI2}	400			ns	
SO output delay time from $\overline{SCK}$ ↓	t_{KSO2}			300	ns	V_{DD} = 4.5 to 6.0 V; C = 100 pF (Note 1)
				1000	ns	C = 100 pF (Note 1)

Note 1: C is the load capacitance of the SO output line.

Serial Interface, SBI Mode; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 36

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY3}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH3}, t_{KL3}	$t_{KCY3}/2 - 50$			ns	V_{DD} = 4.5 to 6.0 V
		$t_{KCY3}/2 - 150$			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK3}	100			ns	V_{DD} = 4.5 to 6.0 V
		300			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI3}	$t_{KCY3}/2$			ns	
SB0,SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO3}	0		250	ns	V_{DD} = 4.5 to 6.0 V; R = 1 kΩ, C = 100 pF (Note 1)
		0		1000	ns	R = 1 kΩ, C = 100 pF (Note 1)
SB0, SB1 ↓ from $\overline{SCK}$ ↑	t_{KSB}	t_{KCY3}			ns	
$\overline{SCK}$ ↓ from SB0, SB1 ↓	t_{SBK}	t_{KCY3}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY3}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY3}			ns	

Note 1: R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

Serial Interface, SBI Mode; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 36

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY4}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH4}, t_{KL4}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK4}	100			ns	V_{DD} = 4.5 to 6.0 V
		300			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI4}	$t_{KCY4}/2$			ns	
SB0, SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO4}	0		300	ns	V_{DD} = 4.5 to 6.0 V; R = 1 kΩ, C = 100 pF. (Note 1)
		0		1000	ns	R = 1 kΩ, C = 100 pF
SB0, SB1 ↓ from $\overline{SCK}$ ↑	t_{KSB}	t_{KCY4}			ns	
$\overline{SCK}$ ↓ from SB0, SB1 ↓	t_{SBK}	t_{KCY4}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY4}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY4}			ns	

Note 1: R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

Serial Interface, 2-Wire, I/O Mode; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 37

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY5}	1600			ns	V_{DD} = 4.5 to 6.0 V
		3800			ns	
$\overline{SCK}$ high-level width	t_{KH5}	$t_{KCY5}/2 - 50$			ns	
$\overline{SCK}$ low-level width	t_{KL5}	$t_{KCY5}/2 - 50$			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK1}	300			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI5}	600			ns	
SB0, SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO5}	0		250	ns	V_{DD} = 4.5 to 6.0 V R = 1 kΩ, C = 100 pF (Note 1)
		0		1000	ns	R = 1 kΩ, C = 100 pF (Note 1)

Note 1: R and C are load resistance and load capacitance of the $\overline{SCK0}$, SB0, and SB1 output lines.

Serial Interface, 2-Wire, I/O Mode; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 37

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY6}	1600			ns	V_{DD} = 4.5 to 6.0 V
		3800			ns	
$\overline{SCK}$ high-level width	t_{KH6}	650			ns	
$\overline{SCK}$ low-level width	t_{KL6}	800			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK6}	100			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI6}	$t_{KCY6}/2$			ns	
SB0, SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO6}	0		300	ns	V_{DD} = 4.5 to 6.0 V R = 1kΩ, C = 100 pF (Note 1)
		0		1000	ns	

Note 1: R and C are load resistance and load capacitance of the $\overline{SCK0}$, SB0, and SB1 output lines.

3c

Serial Interface, 3-Wire, I/O Mode with Automatic Transmit/Receive Function; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 38

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY7}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH7}, t_{KL7}	$t_{KCY7}/2 - 50$			ns	V_{DD} = 4.5 to 6.0 V
		$t_{KCY7}/2 - 150$			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK7}	100			ns	
SI hold time from $\overline{SCK}$ ↑	t_{KSI7}	400			ns	
$\overline{SCK}$ ↓ to SO output delay time	t_{KSO7}			300	ns	V_{DD} = 4.5 to 6.0 V; C = 100 pF (Note 1)
				1000	ns	C = 100 pF (Note 1)
STB ↑ from $\overline{SCK}$ ↑	t_{SBD}	400		t_{KCY7}	ns	
Strobe signal high-level width	t_{SBW}	$t_{KCY7} - 30$		$t_{KCY7} + 30$	ns	
Busy signal set-up time (to busy signal detection timing)	t_{BYS}	100			ns	
Busy signal hold time (from busy signal detection timing)	t_{BYH}	100			ns	
$\overline{SCK}$ ↓ from busy inactive	t_{SPS}			2 t_{KCY7}	ns	

Note 1: C is the load capacitance for the SO output line.

Serial Interface, 3-Wire, I/O Mode with Automatic Transmit/Receive Function; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 38

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY8}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH8}, t_{KL8}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK8}	100			ns	
SI hold time from $\overline{SCK}$ ↑	t_{KSI8}	400			ns	
$\overline{SCK}$ ↓ to SO output delay time	t_{KSO8}			300	ns	V_{DD} = 4.5 to 6.0 V; C = 100 pF (Note 1)
				1000	ns	C = 100 pF (Note 1)
$\overline{SCK}$ ↓ (after STB) from $\overline{SCK}$ ↑	t_{SPS1}	$2t_{KCY8}$			ns	

Note 1: C is the load capacitance for the SO output line.

Figure 35. Serial Interface Timing; 3-Wire Serial I/O Mode

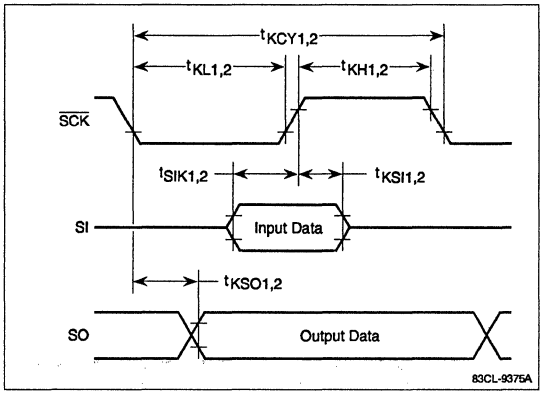
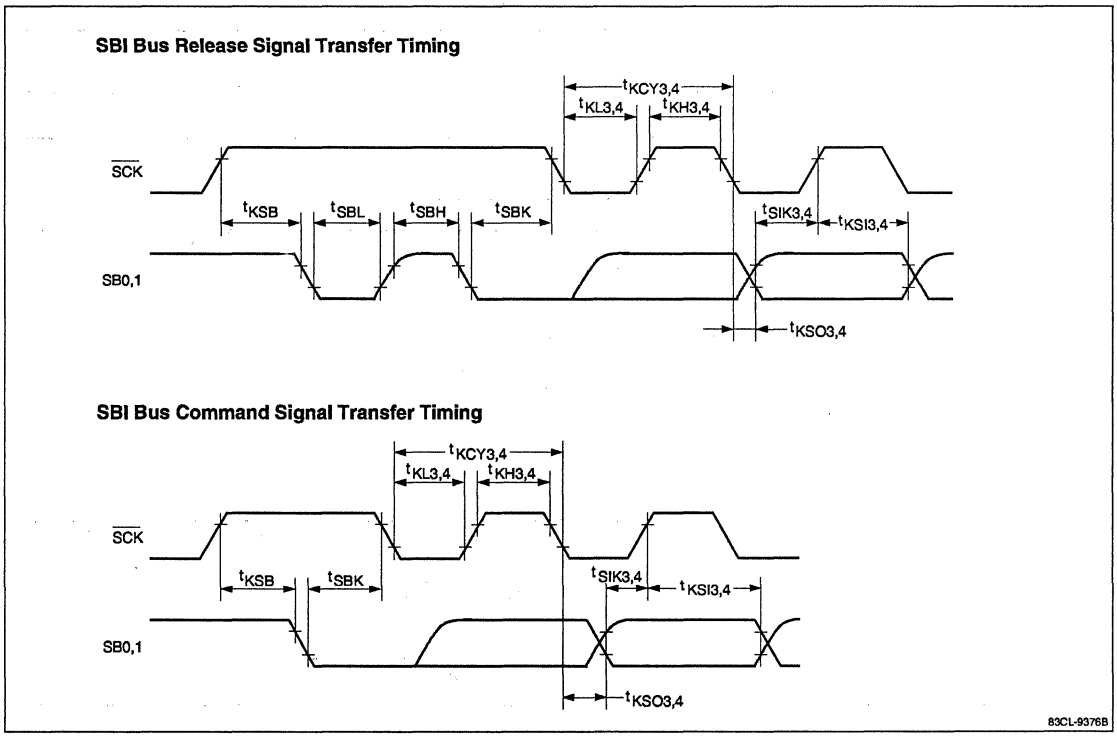


Figure 36. Serial Interface Timing; SBI Mode



3c

Figure 37. Serial Interface Timing; 2-Wire Serial I/O Mode

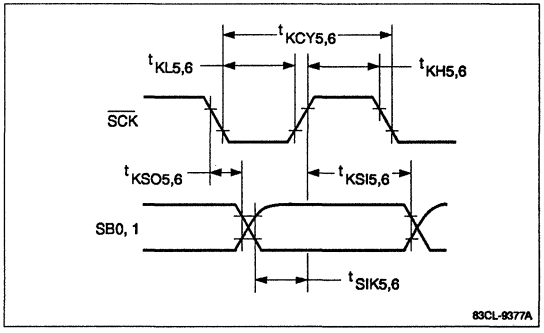
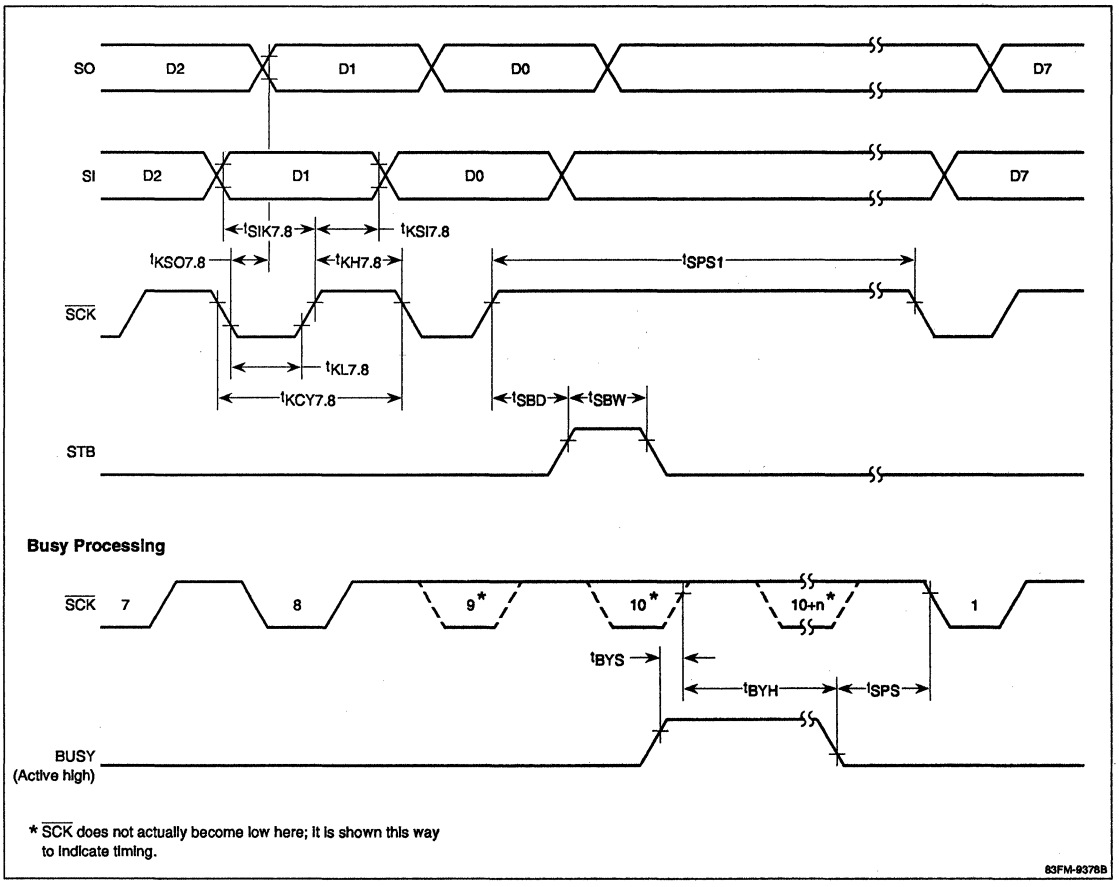


Figure 38. Serial Interface Timing; 3-Wire Serial I/O Mode with Automatic Transmit/Receive Function



A/D Converter

T_A = −40 to +85°C; AV_{DD} = V_{DD} = 2.7 to 6.0 V, AV_{SS} = V_{SS} = 0V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		8	8	8	bit	
Absolute accuracy (Note 1)				±1.5	LSB	
Conversion time	t_{CONV}	160/f_X			μs	f_X = 4.19 to 8.38 MHz
		80/f_X			μs	f_X = 1 to 4.19 MHz
Sampling time	t_{SAMP}	24/f_X			μs	
Analog input voltage	V_{IAN}	AV_{SS}		AV_{REF}	V	
Reference voltage	AV_{REF}	2.7		AV_{DD}	V	
AV_{REF} current	I_{REF}		0.5	1.5	mA	

Note 1: Absolute accuracy does not include the quantization error (±1/2 LSB).

Data Memory STOP Mode; Low-Voltage Data Retention

T_A = −40 to +85°C; refer to figure 39

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention supply voltage	V_{DDDR}	2.0		6.0	V	
Data retention power supply current	I_{DDDR}		0.1	10	μA	V_{DDDR} = 2.0 V; subsystem clock stop and feedback resistor disconnected
Release signal set time	t_{SREL}	0			μs	
Oscillation stabilization wait time	t_{WAIT}		$2^{18}/f_X$		ms	Release by $\overline{RESET}$
			(Note 1)		ms	Release by interrupt

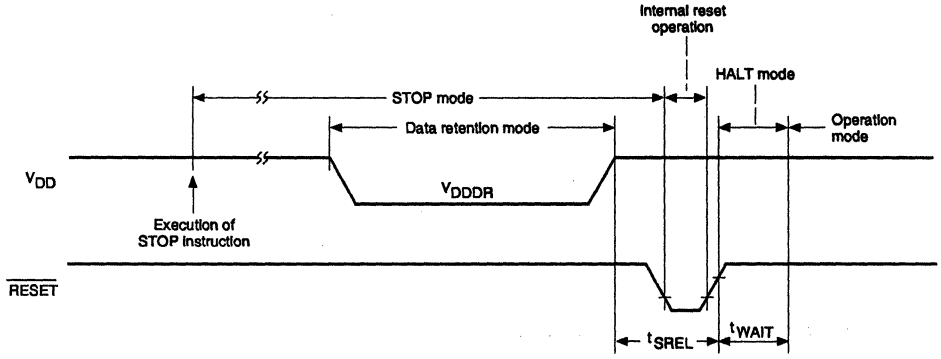
Note:

(1) $2^{13}/f_X$, $2^{15}/f_X$, $2^{16}/f_X$, $2^{17}/f_X$ or $2^{18}/f_X$ can be chosen by using bits 0 to 2 of the oscillation stabilization time select (OSTS) register.

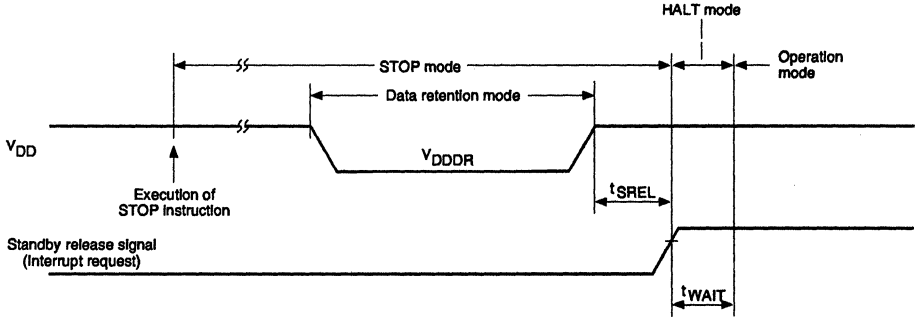
3c

Figure 39. Data Retention Timing

A. STOP mode is released by $\overline{\text{RESET}}$ input



B. STOP mode is released by interrupt signal



83RD-6456B

PROM PROGRAMMING

The PROM in the μPD78P014 is an OTP or UV EPROM. The 32,768 x 8-bit PROM has the programming characteristics of an NEC μPD27C256A. Table 5 shows the functions of the μPD78P014 pins in both normal operating and PROM programming mode.

Table 5. Pin Functions During PROM Programming

Function	Normal Operating Mode	Programming Mode
Address input	$P4_0$ - $P4_7$, $P5_0$, $P0_0$, $P5_2$ - $P5_6$	A_0 - A_{14}
Data input	$P3_0$ - $P3_7$	D_0 - D_7
Chip enable/ program pulse	$P6_5/\overline{WR}$	$\overline{CE}$
Output enable	$P6_4/\overline{RD}$	$\overline{OE}$
Program voltage	IC	V_{PP}
Mode voltage	$\overline{RESET}$	Logical 0

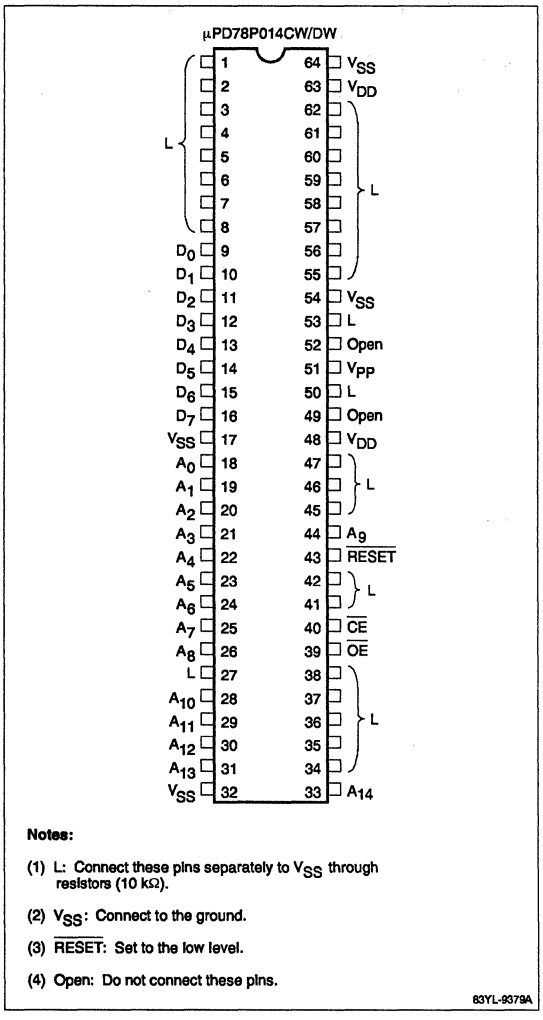
PROM Programming Modes

When the $\overline{RESET}$ pin is set low and V_{PP} is set to +5 V or +12.5 V, the μPD78P014 enters the programming mode of operation. Operation in this mode is determined by the setting of the $\overline{CE}$, $\overline{OE}$, V_{PP} and V_{DD} pins as indicated in Table 6.

Table 6. Programming Operation Modes

Mode	$\overline{RESET}$	V_{PP}	V_{DD}	$\overline{CE}$	$\overline{OE}$	D_0 to D_7
Program write	L	+12.5 V	+6 V	L	H	Data input
Program verify	L	+12.5 V	+6 V	H	L	Data output
Program inhibit	L	+12.5 V	+6 V	H	H	High impedance
Read	L	+5 V	+5 V	L	L	Data output
Output disble	L	+5 V	+5 V	L	H	High impedance
Standby	L	+5 V	+5 V	H	L/H	High impedance

Figure 40. PROM Programming Mode Pin Function; 64-Pin Plastic or Ceramic Shrink DIP



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors (10 kΩ).

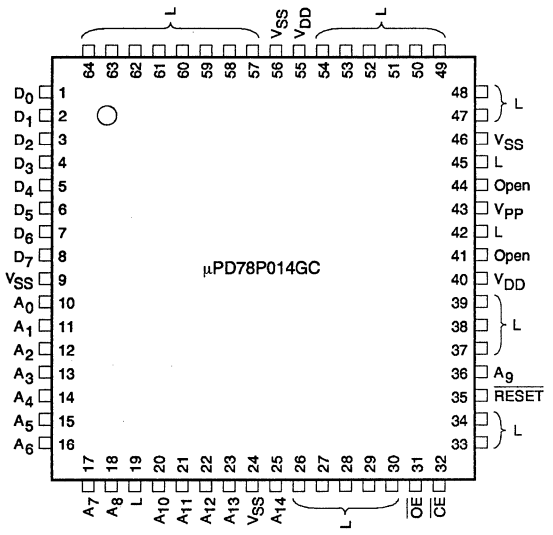
(2) V_{SS}: Connect to the ground.

(3) $\overline{RESET}$: Set to the low level.

(4) Open: Do not connect these pins.

83YL-9379A

Figure 41. PROM Programming Mode Pin Functions; 64-pin Plastic QFP



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors (10 kΩ).

(2) V_{SS}: Connect to the ground.

(3) $\overline{RESET}$: Set to the low level.

(4) Open: Do not connect these pins.

83YL-9380B

3c

PROM Write Procedure

Data can be written to the PROM by using the following procedure.

(1) Set the pins not used for programming as indicated in figures 40 and 41. Set the $\overline{\text{RESET}}$ pin low and the V_{DD} and V_{PP} pins to +5 V. The $\overline{CE}$ and $\overline{OE}$ pins should be high.

(2) Supply +6.0 V to the V_{DD} pin and +12.5 V to the V_{PP} pin.

(3) Provide the initial address to the A_0 - A_{14} pins.

(4) Provide the write data.

(5) Provide a 1-ms program pulse (active low) to the $\overline{CE}$ pin.

(6) Use the verify mode (pulse $\overline{OE}$ low) to test the data. If data is written correctly, proceed to step 8; if data is not written correctly, repeat steps 4 to 6 up to 25 times. If data is still incorrect, go to step 7.

(7) Classify the PROM as defective and cease write operation.

(8) Perform one additional write with a program pulse width (in ms) equal to three times the number of writes performed in step 5.

(9) Increment the address.

(10) Repeat steps 4-9 until the last adress is programmed.

PROM Read Procedure

The contents of the PROM can be read out of the external data bus (D_0 - D_7) by using the following procedure.

(1) Set the pins not used for programming as indicated in figures 40 and 41. Set the $\overline{\text{RESET}}$ pin low and the V_{PP} pin and V_{DD} pin to +5 V. The $\overline{CE}$ and $\overline{OE}$ pins should be high.

(2) Input the address of the data to be read to the A_0 - A_{14} pins.

(3) Put an active-low pulse on $\overline{CE}$ and $\overline{OE}$ pins.

(4) Data is output to pins D_0 - D_7.

Program Erasure

The UV EPROM can be erased (all locations FFH) by exposing the window to light having a wavelength shorter than 400 nm, including ultraviolet, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15 W s/cm^2 (ultraviolet ray intensity x exposure time) is required to completely erase the written data. Erasure by an ultraviolet lamp rated at 12,000 μW/cm^2 takes approximately 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

DC Programming Characteristics

T_A = 25 ±5°C, V_{SS} = 0 V.

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH}	V_{IH}	0.7 V_{DDp}		V_{DDP}	V	
Low-level input voltage	V_{IL}	V_{IL}	0		0.3 V_{DDP}	V	
Input leakage current	I_{LIP}	I_{LI}			10	μA	$0 \le V_I \le V_{DDP}$
High-level output voltage	V_{OH1}	V_{OH1}	2.4			V	I_{OH} = −400 μA
	V_{OH2}	V_{OH2}	V_{DD} − 0.7			V	I_{OH} = −100 μA
Low-level output voltage	V_{OL}	V_{OL}			0.45	V	I_{OL} = 2.1 mA
Output leakage current	I_{LO}				10	μA	$0 \le V_O \le V_{DDP}$, $\overline{OE}$ = V_{IH}
V_{DDP} power voltage	V_{DDP}	V_{CC}	5.75	6.0	6.25	V	Program memory write mode
			4.5	5.0	5.5	V	Program memory read mode
V_{PP} power voltage	V_{PP}	V_{PP}	12.2	12.5	12.8	V	Program memory write mode
				V_{PP} = V_{DDP}		V	Program memory read mode
V_{DDP} power current	I_{DD}	I_{CC}		5	30	mA	Program memory write mode
				5	30	mA	Program memory read mode $\overline{CE}$ = V_{IL}, V_I = V_{IH}
V_{PP} power current	I_{PP}	I_{PP}		5	30	mA	Program memory write mode $\overline{CE}$ = V_{IL}, $\overline{OE}$ = V_{IH}
				1	100	μA	Program memory read mode

*Corresponding symbols of the μPD27C256A.

AC Programming Characteristics (Write Mode)

T_A = 25 ±5°C, V_{SS} = 0 V, V_{DD} = 6 ±0.25 V, V_{PP} = 12.5 ±0.3 V.

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Conditions
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}	2			μs	
Data input to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}	2			μs	
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}	2			μs	
Address hold time from $\overline{CE}$ ↑	t_{HCA}	t_{AH}	2			μs	
Input data hold time from $\overline{CE}$ ↑	t_{HCID}	t_{DH}	2			μs	
Output data hold time from $\overline{OE}$ ↑	t_{HOOD}	t_{DF}	0		130	ns	
V_{PP} setup time to $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}	1			ms	
V_{DDP} setup time to $\overline{CE}$ ↓	t_{SVDC}	t_{VCS}	1			ms	
Initial program pulse width	t_{WL1}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{WL2}	t_{OPW}	2.85		78.75	ms	
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			150	ns	

* Corresponding symbols of the μPD27C256A.

AC Programming Characteristics (Read Mode)

T_A = 25 ±5°C V_{SS} = 0 V, V_{DD} = 5 ±0.5 V, V_{PP} = V_{DD}.

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
Address to data output time	t_{DAOD}	t_{ACC}			200	ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}
$\overline{CE}$ ↓ to data output time	t_{DCOD}	t_{CE}			200	ns	$\overline{OE}$ = V_{IL}
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			75	ns	$\overline{CE}$ = V_{IL}
Data hold time from $\overline{OE}$ ↑	t_{HCOD}	t_{DF}	0		60	ns	$\overline{CE}$ = V_{IL}
Data hold time from address	t_{HAOD}	t_{OH}	0			ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}

* Corresponding symbols of the μPD27C256A.

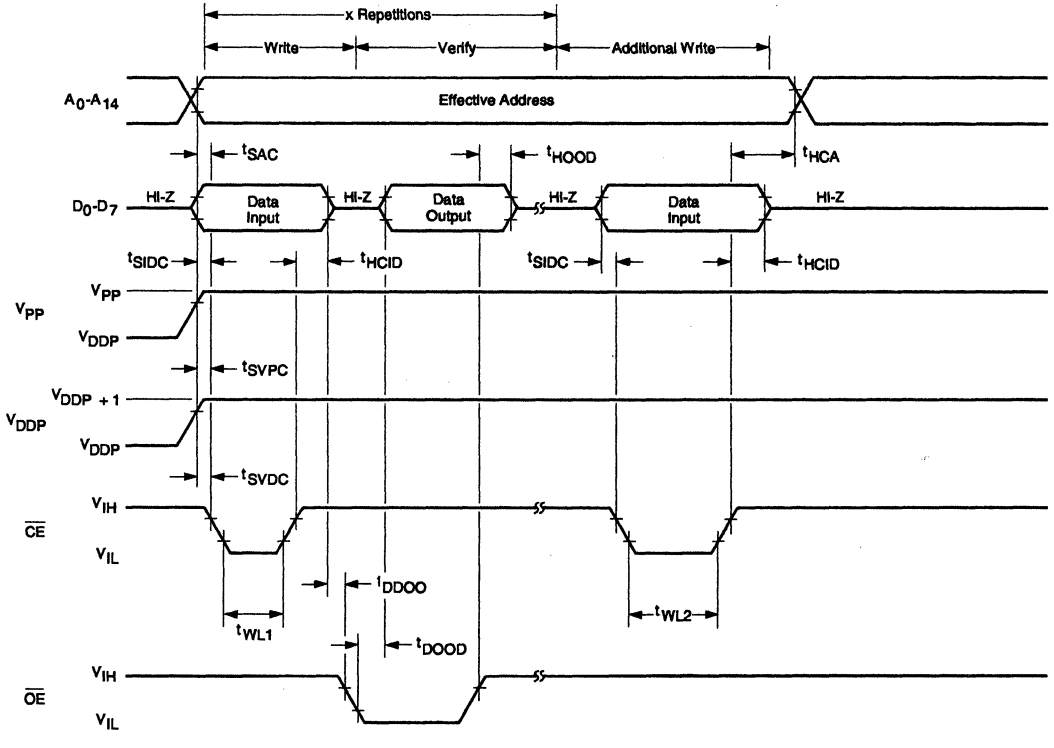
AC Programming Characteristics (PROM Mode)

T_A = 25 ±5°C, V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Condition
PROM mode setup time	t_{SMA}			10	μs	

PROM Timing Diagrams

PROM Write/Verify Mode



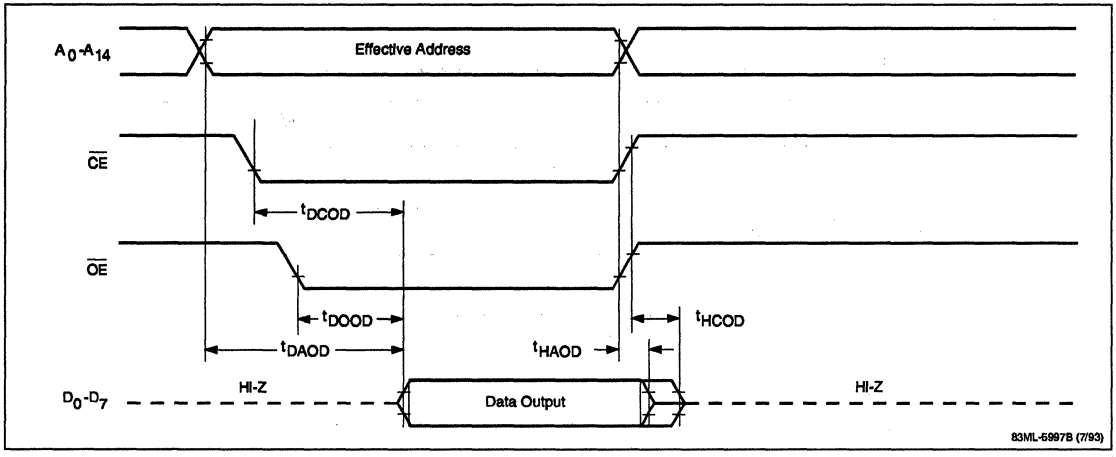
Notes:
(1) V_{DDP} must be applied before applying V_{PP}. It should be removed before removing V_{PP}.
(2) V_{PP} must not exceed +13V, including overshoot.

83YL-9382B (8/93)

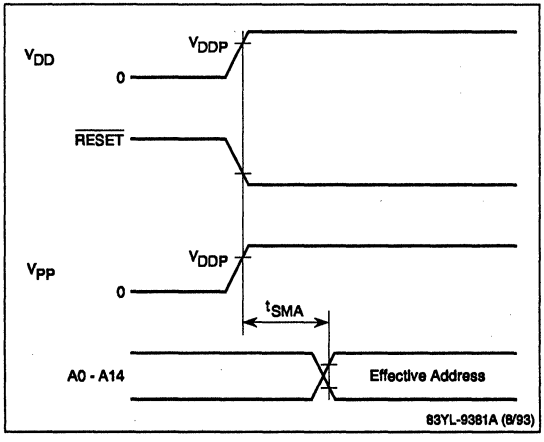
3c

PROM Timing Diagrams (cont)

PROM Read Mode



PROM Mode Setting



NEC

NEC Electronics Inc.

μPD78014Y Family (μPD78011BY/012BY/013Y/014Y/P014Y) 8-Bit, K-Series Microcontrollers General Purpose With A/D Converter and I²C Bus

Preliminary

September 1993

Description

The μPD78011BY, μPD78012BY, μPD78013Y, μPD78014Y, and μPD78P014Y are members of NEC's K-Series® of microcontrollers. The μPD78014Y family is a variation of the μPD78014 family with the addition of an I²C bus mode in serial interface 0. These 8-bit, single-chip microcontrollers feature an A/D converter, two serial interface ports, 8-bit hardware multiply and divide instructions, bit manipulation instructions, four banks of main registers, an advanced interrupt handling facility, and a powerful set of memory-mapped on-chip peripherals.

On-board data memory includes 512 or 1024 bytes of internal high-speed RAM plus 32 bytes of serial buffer RAM. Program memory options include 8K, 16K, 24K, or 32K bytes of mask ROM, or 32K bytes of UV EPROM or one-time programmable (OTP) ROM.

The μPD78014Y family operates over a wide voltage range: 2.7 to 6.0 volts. Timing is generated by two built-in oscillators. A main oscillator normally drives the CPU and most peripherals and at 10 MHz provides a minimum instruction time of 0.4 μsec. A subsystem oscillator at 32.768 kHz provides time keeping, and optionally a slow clock for the CPU. Since CMOS power dissipation is directly proportional to clock rate, the μPD78014Y family provides a software variable CPU clock. The HALT and STOP modes are two additional power saving features that turn off parts of the microcontroller to reduce power consumption. A data retention mode permits RAM contents to be saved down to 2 volts.

The range of peripherals, including an A/D converter, timers, and two serial ports (one with an IC² bus interface) makes these devices ideal for applications in portable battery-powered equipment, office automation, comunications, consumer electronics, home appliances, and fitness equipment.

K-Series is a registered trademark of NEC Electronics, Inc.

Purchase of NEC I²C components conveys a license under the Philips I²C Patents Rights to use these components in an I²C system, provided that the system conforms to the I²C standard specification as defined by Philips.

Features

- Eight-channel 8-bit A/D converter
  - —Operates from 2.7 to 6.0 V
- Two-channel serial communication interface
  - —8-bit clock-synchronous interface 0
    - I²C bus mode
    - Full-duplex, three-wire mode
    - NEC serial bus interface (SBI) mode
    - Half-duplex, two-wire mode
  - —8-bit clock-synchronous interface 1
    - Full-duplex, three-wire mode with automatic transmit/receive
    - Half-duplex, two-wire mode
- Timers
  - —Watchdog timer
  - —16-bit timer/event counter
  - —Two 8-bit timer/event counters usable as one 16-bit timer event/counter
  - —Clock (watch) timer (time of day tick from either oscillator)
- 53 I/O lines
  - —Two CMOS input-only lines
  - —47 CMOS I/O lines
  - —Four n-channel, open-drain I/O lines at 15 V maximum
- I/O port pullup resistors
  - —Software controllable on 47 lines
  - —Mask option on four lines on ROM version
- Program memory
  - —μPD78011BY: 8K bytes ROM
  - —μPD78012BY: 16K bytes ROM
  - —μPD78013Y: 24K bytes ROM
  - —μPD78014Y: 32K bytes ROM
  - —μPD78P014Y: 32K bytes EPROM/OTP
- Internal high-speed data memory (RAM)
  - —μPD78011BY/012BY: 512 bytes
  - —μPD78013Y/014Y/P014Y: 1024 bytes
- Specialized memory
  - —Serial buffer RAM: 32 bytes
- External memory expansion
  - —64K bytes total memory space
- Powerful instruction set
  - —8-bit unsigned multiply and divide
  - —16-bit arithmetic and data transfer instructions
  - —1-bit and 8-bit logic instructions

3d

Features (cont)

- Minimum instruction times:
  - 0.4/0.8/1.6/3.2/6.4 μs program selectable using 10-MHz main system clock
  - 122 μs selectable using 32.768-kHz subsystem clock
- Memory-mapped on-chip peripherals (special function registers)
- Programmable priority, vectored-interrupt controller (two levels)
- Buzzer and clock outputs
- Power saving and battery operation features
  - Variable CPU clock rate
  - HALT mode
  - STOP mode
  - 2-V data retention mode
  - CMOS operation; V_{DD} from 2.7 to 6.0 V

Ordering Information

Part Number	ROM	Package	Package Drawing
μPD78011BYCW-xxx	8K mask ROM	64-pin plastic shrink DIP	P64C-70-750 A, C
μPD78012BYCW-xxx	16K mask ROM		
μPD78013YCW-xxx	24K mask ROM		
μPD78014YCW-xxx	32K mask ROM		
μPD78P014YCW	32K OTP ROM		
μPD78011BYGC-xxx-AB8	8K mask ROM	64-pin plastic QFP	P64GC-80-AB8-2
μPD78012BYGC-xxx-AB8	16K mask ROM		
μPD78013YGC-xxx-AB8	24K mask ROM		
μPD78014YGC-xxx-AB8	32K mask ROM		
μPD78P014YGC-AB8	32K OTP ROM		
μPD78P014YDW	32K UV EPROM	64-pin ceramic shrink DIP w/window	P64DW-70-750 A

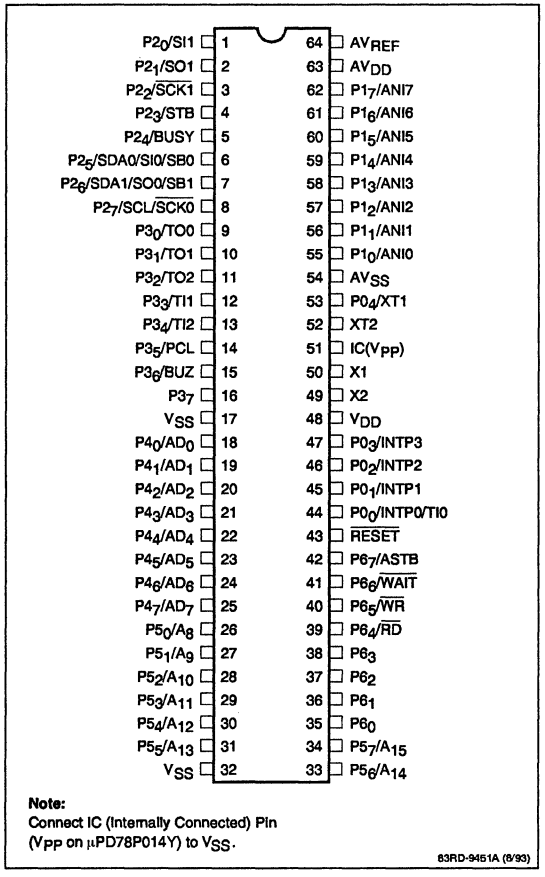
Notes:

(1) xxx indicates ROM code suffix

(2) All devices listed are standard quality grade

Pin Configurations

64-Pin Plastic or Ceramic Shrink DIP

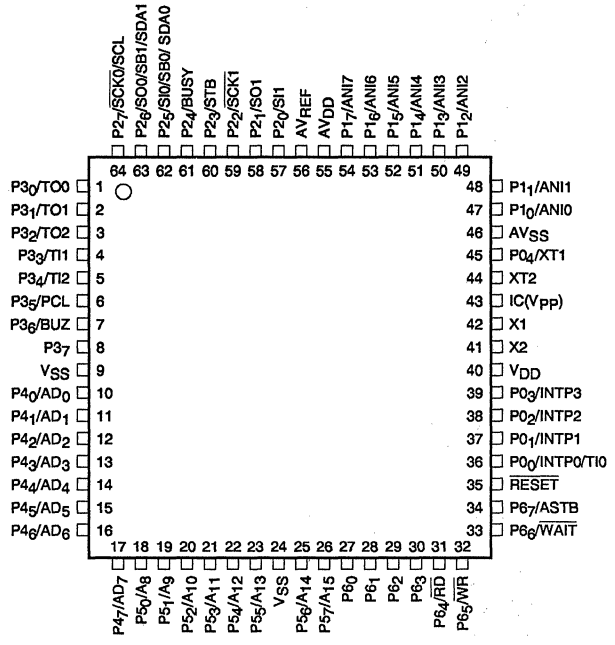


Note:
Connect IC (Internally Connected) Pin (VPP on μPD78P014Y) to VSS.

83RD-9451A (8/93)

Pin Configurations (cont)

64-Pin Plastic QFP



Note:
Connect IC (Internally Connected) Pin (V_{PP} on μPD78P014Y) to V_{SS}.

83RD-9438B (8/93)

Pin Functions; Normal Operating Mode

Symbol	First Function	Symbol	Alternate Function
$P0_0$	Port 0; 5-bit, bit selectable I/O port (Bits 0 and 4 are input only)	INTP0 TI0	External maskable interrupt External count clock input to timer 0
$P0_1$ $P0_2$ $P0_3$		INTP1 INTP2 INTP3	External maskable interrupt
$P0_4$		XT1	Crystal oscillator or external clock input for subsystem clock
$P1_0$ - $P1_7$	Port 1; 8-bit, bit-selectable I/O port	ANI0-ANI7	Analog input to A/D converter
$P2_0$	Port 2; 8-bit, bit-selectable I/O port	SI1	Serial data input three-wire serial I/O mode
$P2_1$		SO1	Serial data output three-wire serial I/O mode
$P2_2$		$\overline{SCK1}$	Serial clock I/O for serial interface 1
$P2_3$		STB	Serial interface automatic transmit/receive strobe output
$P2_4$		BUSY	Serial interface automatic transmit/receive busy input
$P2_5$		SI0 SB0 SDA0	Serial data input three-wire serial I/O mode 2/3-wire serial I/O mode Serial data bus 0 for I^2C bus mode
$P2_6$		SO0 SB1 SDA1	Serial data output three-wire serial I/O mode 2/3-wire serial I/O mode Serial data bus 1 for I^2C bus mode
$P2_7$		$\overline{SCK0}$ SCL	Serial clock I/O for serial interface 0 Serial clock I/O for I^2C bus mode
$P3_0$	Port 3; 8-bit, bit-selectable I/O port	TO0	Timer output from timer 0
$P3_1$		TO1	Timer output from timer 1
$P3_2$		TO2	Timer output from timer 2
$P3_3$		TI1	External count clock input to timer 1
$P3_4$		TI2	External count clock input to timer 2
$P3_5$		PCL	Programmable clock output
$P3_6$		BUZ	Programmable buzzer output
$P3_7$		—	
$P4_0$ - $P4_7$	Port 4; 8-bit I/O port	AD_0 - AD_7	Low-order 8-bit multiplexed address/data bus for external memory
$P5_0$ - $P5_7$	Port 5; 8-bit, bit selectable I/O port	A_8 - A_{15}	High-order 8-bit address bus for external memory
$P6_0$ - $P6_3$	Port 6; 8-bit, bit selectable ($P6_0$ to $P6_3$ n-channel open-drain I/O with mask option pullup resistors ($P6_4$ - $P6_7$ I/O). See note.	—	
$P6_4$		$\overline{RD}$	External memory read strobe
$P6_5$		$\overline{WR}$	External memory write strobe
$P6_6$		$\overline{WAIT}$	External memory wait signal input
$P6_7$		ASTB	Address strobe used to latch address for external memory

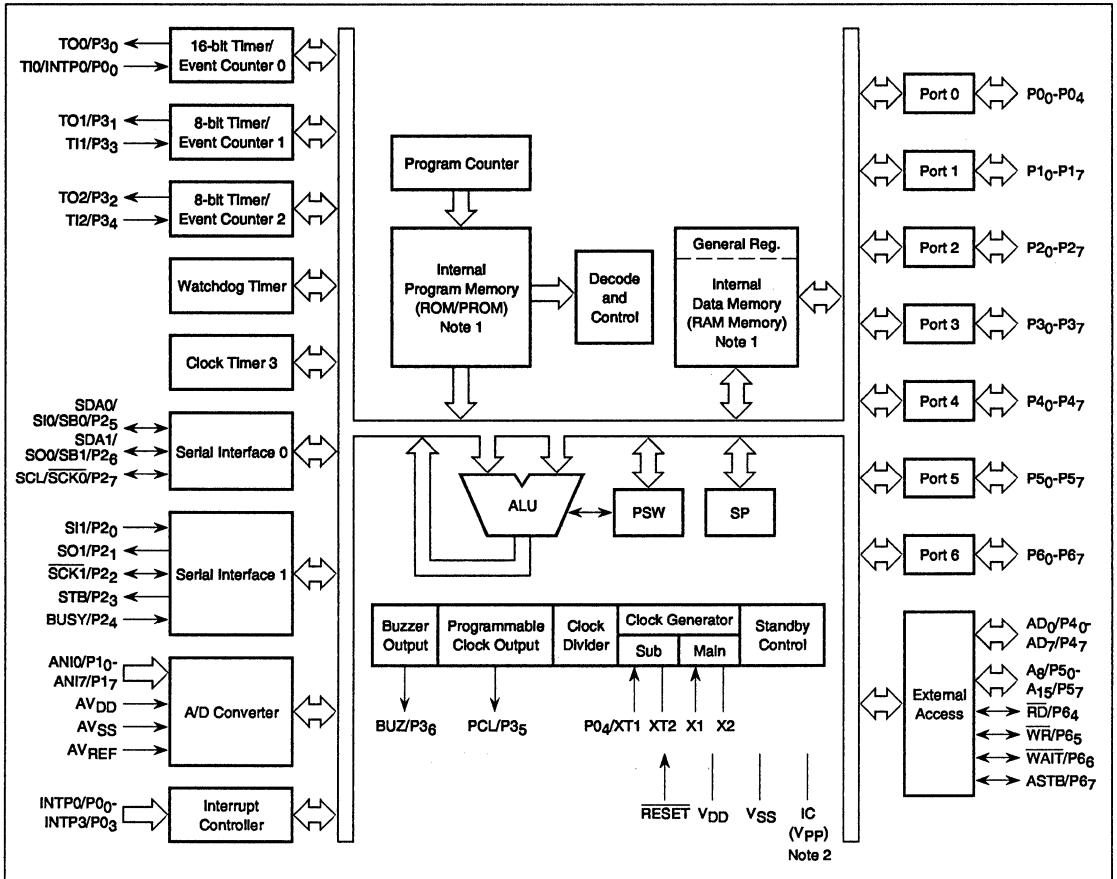
3d

Pin Functions; Normal Operating Mode (cont)

Symbol	First Function	Symbol	Alternate Function
$\overline{RESET}$	External system reset input		
X1	Crystal/ceramic reaonator connection or external clock input for main system clock		
X2	Crystal/ceramic resonator connection or inverse of external clock for main system clock		
XT2	Crystal oscillator or left open when using external clock for subsystem clock		
AV_{REF}	A/D converter reference voltage		
AV_{DD}	A/D converter power supply input		
AV_{SS}	A/D converter ground		
V_{DD}	Power-supply input		
V_{PP}	μPD78P014Y PROM programming power-supply input		
V_{SS}	Power-supply ground		
IC	Internal connection		

Note: See table 2 and figure 4 for details

Block Diagram



Notes:
(1) The internal ROM and RAM size depends on the device.
(2) Pin name in parentheses for the μPD78P014Y only.

83RD-9439B (8/93)

3d

FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processing unit (CPU) of the μPD78014Y family features 8- and 16-bit arithmetic including an 8 x 8-bit unsigned multiply and 16 x 8-bit unsigned divide (producing a 16-bit quotient and an 8-bit remainder). The multiply executes in 3.2 μs and the divide in 5 μs using the fastest clock cycle with a main system clock of 10 MHz.

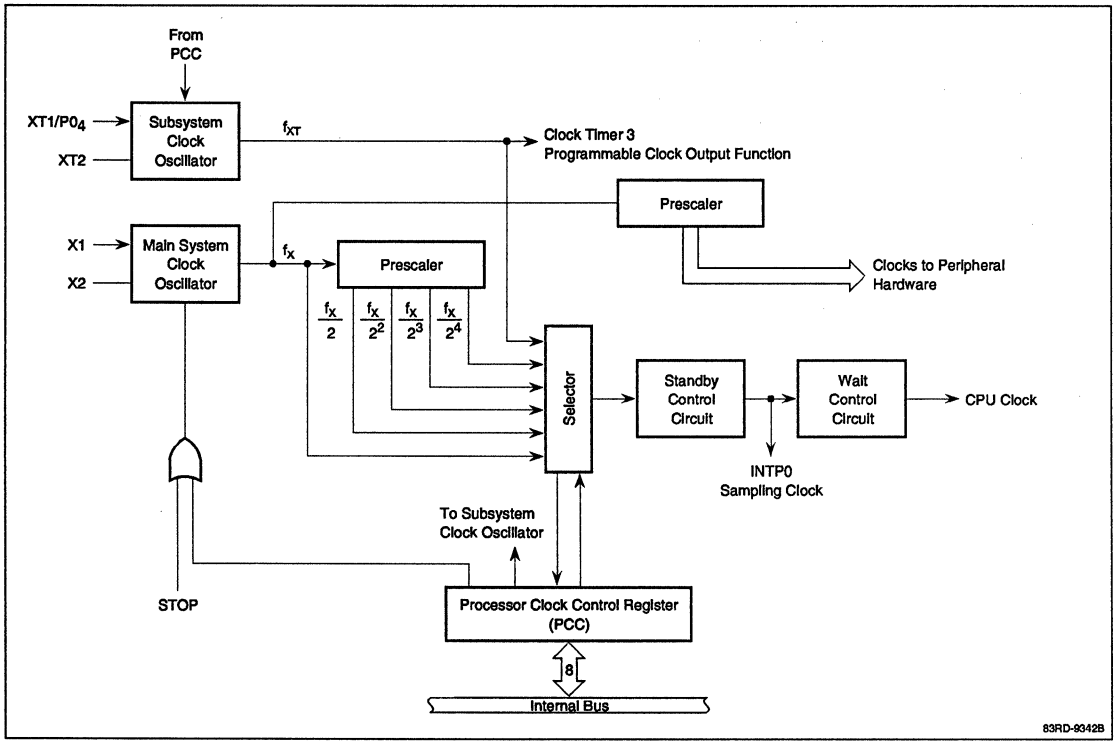
A CALLT vector table and a CALLF area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction can access up to 32 subroutines through their addresses contained in the CALLT vector table (40H to 7FH). A 2-byte call instruction can access any routine beginning in a specific CALLF area (0800H to 0FFFH).

Internal System Clock Generator

The internal system clocks of the μPD78014Y family are derived from either the main system or the subsystem oscillator. See figure 1. The clocks for the clock timer and programmable clock output are derived from either the subsystem clock (f_{XT}) or the main system clock. The clocks for all other peripheral hardware are derived from the main system clock.

The CPU clock (ϕ) can be supplied from either the main system clock (f_X) or the subsystem clock (f_{XT}). Using the processor clock control register (PCC), a CPU clock frequency equal to f_X, $f_X/2$, $f_X/4$, $f_X/8$, $f_X/16$ or the subsystem clock f_{XT} can be selected. The CPU clock selected should be based on the power supply voltage available and the desired power consumption. On power up, the CPU clock defaults to the lowest speed from the main system clock and can be changed while the microcontroller is running.

Figure 1. Internal System Clock Generator

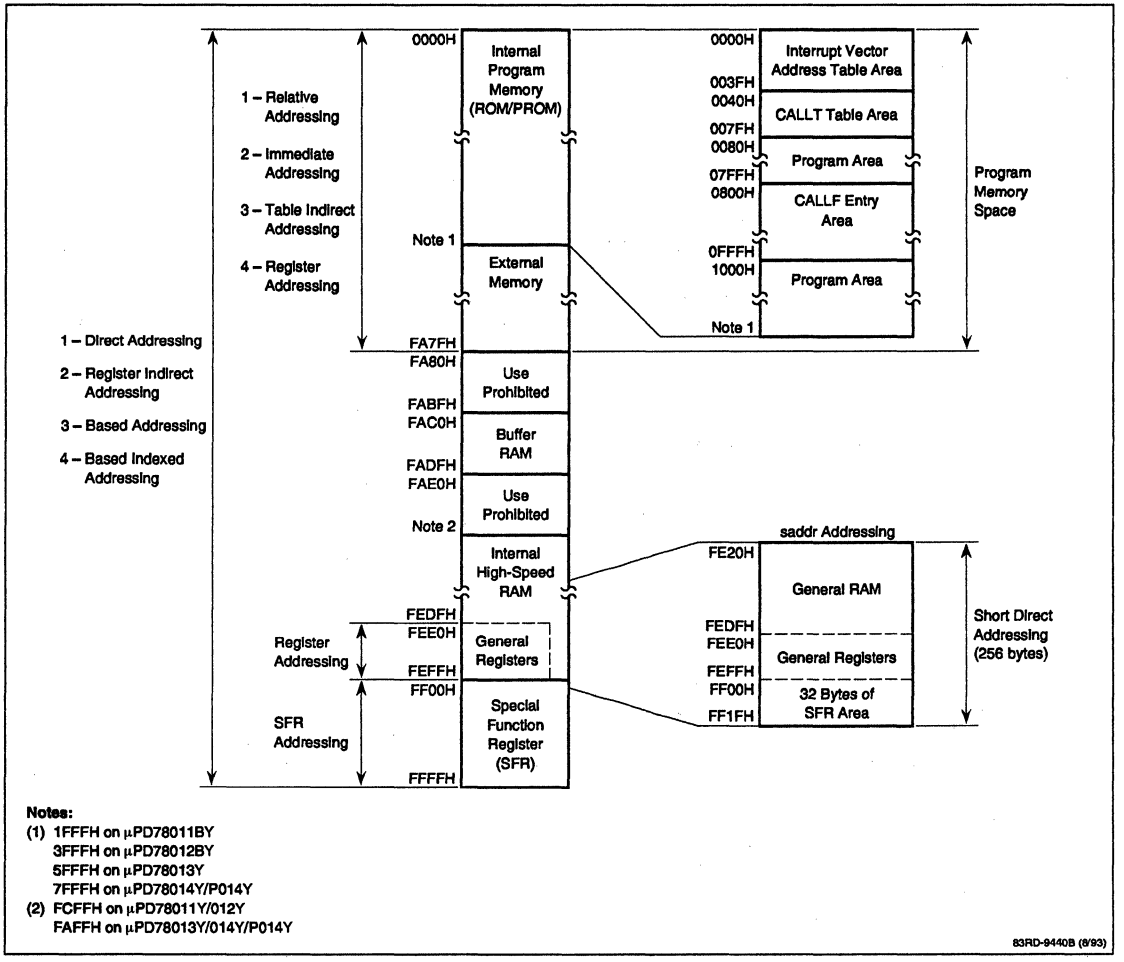


Since the shortest instruction takes four CPU clocks to execute, the fastest minimum instruction execution time (t_{CY}) of 0.4 μs is achieved when using a main system clock at 10 MHz (V_{DD} equals 4.5 to 6.0 V). However, if the clock timer must generate an interrupt every 0.5 or 0.25 seconds, t_{CY} is 0.48 μs at 8.38 MHz. The fastest minimum instruction execution time available across the full voltage range of 2.7 to 6.0 V is 0.96 μs when using a main system clock of 8.38 MHz. For the lowest power consumption, the CPU can be operated from the subsystem clock and the minimum instruction execution time is 122 μs at 32.768 kHz.

Memory Space

The μPD78014Y family has a 64K-byte address space. Some of this address space (0000H-FFFFH) can be used as both program and data memory as shown in figure 2.

Figure 2. Memory Map



Notes:
(1) 1FFFH on μPD78011BY
3FFFH on μPD78012BY
5FFFH on μPD78013Y
7FFFH on μPD78014Y/P014Y
(2) FCFFH on μPD78011Y/012Y
FAFFH on μPD78013Y/014Y/P014Y

83RD-9440B (8/93)

3d

Internal Program Memory

All devices in the μPD78014Y family have internal program memory. The μPD78011Y/012Y/013Y/014Y contain 8K, 16K, 24K, and 32K bytes of internal ROM, respectively. The μPD78P014Y contains 32K bytes of UV EPROM or one time programmable ROM. To allow the μPD78P014Y to emulate the mask ROM devices, the amount of internal program memory available in the μPD78P014Y can be selected using the memory size switching register (IMS).

Internal RAM

The μPD78011Y/012Y have 544 bytes and the μPD78013Y/014Y/P014Y have 1056 bytes of Internal RAM. This Internal RAM consists of two types: high-speed Internal RAM and buffer RAM.

The μPD78011Y/012Y contain 512 bytes (FD00H to FEFFH) while the μPD78013Y/014Y/P014Y contain 1024 bytes (FB00H to FEFFH) of high-speed Internal RAM. The high-speed Internal RAM contains the general register banks and the stack. The remainder of the high-speed Internal RAM and any unused register bank locations are available for general storage.

All devices also contain 32 bytes of buffer RAM (FAC0H to FADFH). The buffer RAM is accessed at the same speed as external memory and is used as the buffer area for the automatic transfer mode of serial interface 1 or for general storage.

To allow the μPD78P014Y to emulate the mask ROM devices, the amount of high-speed Internal RAM available in the μPD78P014Y can also be selected using the IMS.

External Memory

The μPD78014Y family can access 0, 256, 4K, 16K or all available bytes of external memory. The μPD78014Y family has an 8-bit wide external data bus and a 16-bit wide external address bus. The low-order 8 bits of the address bus are multiplexed and also provide the 8-bit data bus and are supplied by port 4. The high-order address bits of the 16-bit address bus are taken from port 5 as required. The address latch, read, and write strobes, and the external $\overline{WAIT}$ signal are supplied by port 6.

The memory expansion mode register (MM) controls the size of external memory. It can be programmed to use 0, 4, 6, or 8 bits from port 5 for the high-order address. Any remaining port 5 bits can be used for I/O. The MM register also can be used to specify one additional wait state or the use of the external $\overline{WAIT}$ signal for low-speed external memory or external peripheral devices.

When only internal ROM and RAM are used and no external memory is required, ports 4, 5 and 6 are available as general purpose I/O ports.

CPU Control Registers

Program Counter. The program counter is a 16-bit binary counter register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 0001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

Program Status Word. The program status word (PSW) is an 8-bit register that contains flags that are set or reset depending on the results of an instruction. This register can be written to or read from 8 bits at a time. The individual flags can also be manipulated on a bit-by-bit basis. The assignment of PSW bits follows.

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

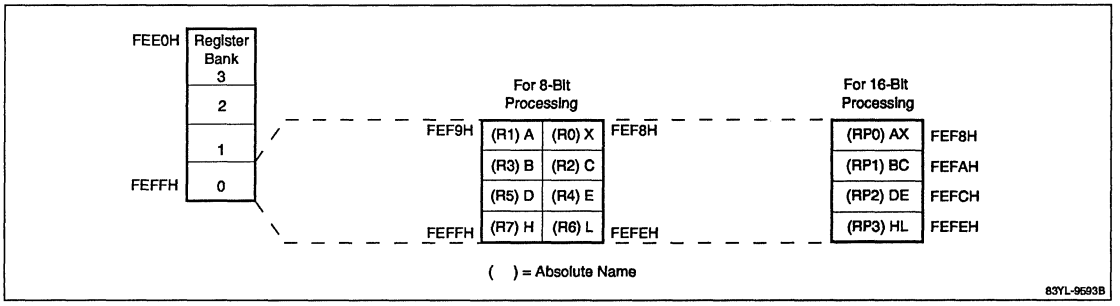
CY	Carry flag
ISP	In-service (interrupt) priority flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

General Registers

The general-purpose registers (figure 3) consist of four banks of registers located at addresses FEE0H to FEFFH in Internal RAM. Each bank consists of eight 8-bit general registers that can also be used in pairs to function as four 16-bit registers. Two bits in the PSW (RBS0 and RSB1) specify which of the register banks is active at any time and are set under program control.

Registers have both functional names (like A, X, B, C, D, E, H or L for 8-bit registers and AX, BC, DE, and HL for 16-bit registers) and absolute names (like R1, R0, R3, R2, R5, R4, R7, or R6 for 8-bit registers and RP0, RP1 RP2, or RP3 for 16-bit registers). Either the functional or absolute register names can be used in instructions that use the operand identifiers r and rp.

Figure 3. General Registers



Addressing

The program memory addressing (ROM) modes provided are relative, immediate, table indirect and register addressing. The operand addressing modes provided are implied, register, direct, short direct (saddr), special function (SFR), register indirect, based, based indexed, and stack addressing.

The 'SFR addressing' and 'saddr addressing' modes use direct addressing, and require only 1 byte in the instruction to address RAM. Normally, a 65K byte address space requires 2 bytes to address it. One-byte addressing results in faster access times, since the instructions are shorter. SFR addressing addresses the entire 256-byte SFR address space from FF00H to FFFFH. Saddr addressing (see figure 2) addresses the 256-byte address space FE20H to FF1FH. FE20H to FEFFH are composed of 224 bytes of internal high speed RAM; FF00H to FF1FH contain the first 32 bytes in the special function register area.

One-byte addressing is accomplished by using the first byte of the instruction for the opcode (and one operand if register A or AX is used) and the second byte of the instruction as an address (offset) into the 256-byte area. If register A or AX is used, the instructions are 2 bytes long, thereby providing fast access times. If immediate data is used, the instruction will be 3 or 4 bytes long depending upon whether the immediate data is a byte or a word. Many 16-bit SFRs are in the space FF00H to FF1FH. Using AX as an operand to these SFRs will provide fast access, since the instructions will be only 2 bytes long.

3d

Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by SFR addressing. FF00H to FF1H can also be accessed using saddr addressing. They are either 8 or 16 bits as required, and many of the 8-bit registers are bit addressable.

Locations FFD0H through FFDFH are known as the external SFR area. Registers in external circuitry interfaced and mapped to these addresses can only be addressed by main memory addressing. Table 1 lists the special function registers.

Table 1. Special Function Registers

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF00H	Port 0	P0	R/W	x	x	—	00H
FF01H	Port 1	P1	R/W	x	x	—	00H
FF02H	Port 2	P2	R/W	x	x	—	00H
FF03H	Port 3	P3	R/W	x	x	—	00H
FF04H	Port 4	P4	R/W	x	x	—	Undefined
FF05H	Port 5	P5	R/W	x	x	—	Undefined
FF06H	Port 6	P6	R/W	x	x	—	Undefined
FF10H-FF11H	Compare register 00	CR00	R/W	—	—	x	Undefined
FF12H-FF13H	Capture register 01	CR01	R	—	—	x	Undefined
FF14H-FF15H	16-bit timer register	TM0	R	—	—	x	00H
FF16H	Compare register 10	CR10	R/W	—	x	—	Undefined
FF17H	Compare register 20	CR20	R/W	—	x	—	Undefined
FF18H	8-bit timer register 1	TM1	R	x	x	—	00H
FF19H	8-bit timer register 2	TM2	R	x	x	—	00H
FF18H-FF19H	8-bit timer registers 1 and 2	TMS	R	—	—	x	0000H
FF1AH	Serial I/O shift register 0	SIO0	R/W	—	x	—	Undefined
FF1BH	Serial I/O shift register 1	SIO1	R/W	—	x	—	Undefined
FF1FH	A/D conversion result register	ADCR	R	—	x	—	Undefined
FF20H	Port mode register 0	PM0	R/W	x	x	—	1FH
FF21H	Port mode register 1	PM1	R/W	x	x	—	FFH
FF22H	Port mode register 2	PM2	R/W	x	x	—	FFH
FF23H	Port mode register 3	PM3	R/W	x	x	—	FFH
FF25H	Port mode register 5	PM5	R/W	x	x	—	FFH
FF26H	Port mode register 6	PM6	R/W	x	x	—	FFH
FF40H	Timer clock select register 0	TCL0	R/W	x	x	—	00H
FF41H	Timer clock select register 1	TCL1	R/W	—	x	—	00H
FF42H	Timer clock select register 2	TCL2	R/W	—	x	—	00H
FF43H	Timer clock select register 3	TCL3	R/W	—	x	—	88H
FF47H	Sampling clock select register	SCS	R/W	—	x	—	00H
FF48H	16-bit timer mode control register	TMC0	R/W	x	x	—	00H
FF49H	8-bit timer mode control register	TMC1	R/W	x	x	—	00H
FF4AH	Watch (clock) timer mode control register	TMC2	R/W	x	x	—	00H
FF4EH	16-bit timer output control register	TOC0	R/W	x	x	—	00H
FF4FH	8-bit timer output control register	TOC1	R/W	x	x	—	00H
FF60H	Serial operating mode register 0	CSIM0	R/W	x	x	—	00H
FF61H	Serial bus interface control register	SBIC	R/W	x	x	—	00H
FF62H	Slave address register	SVA	R/W	—	x	—	Undefined
FF63H	Interrupt timing specify register	SINT	R/W	x	x	—	00H
FF68H	Serial operation mode register 1	CSIM1	R/W	x	x	—	00H
FF69H	Automatic data transmit/receive control register	ADTC	R/W	x	x	—	00H

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF6AH	Automatic data transmit/receive address pointer register	ADTP	R/W	—	x	—	00H
FF80H	A/D converter mode register	ADM	R/W	x	x	—	01H
FF84H	A/D converter input select register	ADIS	R/W	—	x	—	00H
FFD0H-FFDFH	External SFR access area(Note 1)	—	R/W	x	x	—	Undefined
FFE0H	Interrupt request flag register L	IF0L	R/W	x	x	—	00H
FFE1H	Interrupt request flag register H	IF0H	R/W	x	x	—	00H
FFE0H-FFE1H	Interrupt request flag register	IF0	R/W	—	—	x	0000H
FFE4H	Interrupt mask flag register L	MK0L	R/W	x	x	—	FFH
FFE5H	Interrupt mask flag register H	MK0H	R/W	x	x	—	FFH
FFE4H-FFE5H	Interrrupt mask flag register	MK0	R/W	—	—	x	FFFFH
FFE8H	Priority order specify flag register L	PR0L	R/W	x	x	—	FFH
FFE9H	Priority order specify flag register H	PR0H	R/W	x	x	—	FFH
FFE8H-FFE9H	Priority order specify flag register	PR0	R/W	—	—	x	FFFFH
FFECH	External interrupt mode register	INTM0	R/W	—	x	—	00H
FFF0H	Memory size switch register (Note 2)	IMS	W	—	x	—	C8H
FFF6H	Key return mode register	KRM	R/W	x	x	—	02H
FFF7H	Pullup resistor option register	PU0	R/W	x	x	—	00H
FFF8H	Memory expanded mode register	MM	R/W	x	x	—	10H
FFF9H	Watchdog timer mode register	WDTM	R/W	x	x	—	00H
FFFAH	Oscillation stabilization time select register	OSTS	R/W	—	x	—	04H
FFFBH	Processor clock control register	PCC	R/W	x	x	—	04H

Notes:

(1) The external access area cannot be accessed using SFR addressing. It can only be accessed using main memory addressing.

(2) μPD78P014Y only.

3d

Input/Output Ports

The μPD78014Y family has up to 53 port lines. Table 2 lists the features of each port and figure 4 shows the structure of each port pin.

Table 2. Digital Port Functions

Port	Operational Features	Configuration	Direct Drive Capability	Software Pullup Resistor Connection (Note 1)
Port 0 (Note 2)	5-bit input or output	Bit selectable		Byte selectable, input bits only
Port 1	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 2	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 3	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 4	8-bit input or output	Byte selectable		Byte selectable, input bits only
Port 5	8-bit input or output	Bit selectable	LED	Byte selectable, input bits only
Port 6	8-bit input or output ($P6_0$ - $P6_3$ n-channel)	Bit selectable	15 V max ($P6_0$ - $P6_3$)	Byte selectable, input bits only $P6_0$ - $P6_3$ - mask option only (Note 3) $P6_4$ - $P6_7$ - software

Notes:

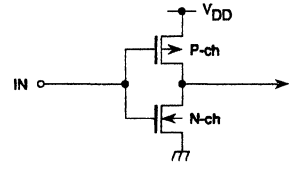
(1) Software pullup resistors can be internally connected only (on a port-by-port basis) to port bits set to input mode. Pullup resistors are not connected to port bits set to output mode.

(2) $P0_0$ and $P0_4$ are input only and do not have a software pullup resistor.

(3) All devices except μPD78P014Y.

Figure 4. Pin Input/Output Circuits

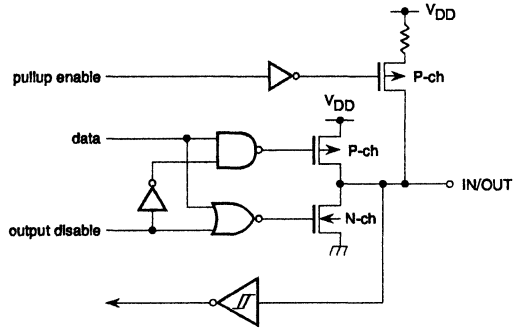
Type 1 ($P0_4$)



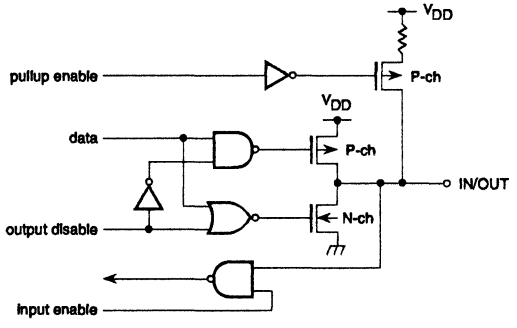
Type 2 ($P0_0$, $\overline{RESET}$)

Schmitt trigger input with hysteresis characteristics

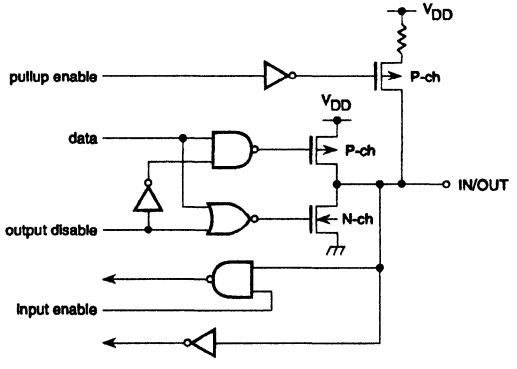
Type 8-A ($P0_1$ - $P0_3$, $P2_0$, $P2_2$, $P2_4$, $P3_3$ - $P3_4$)



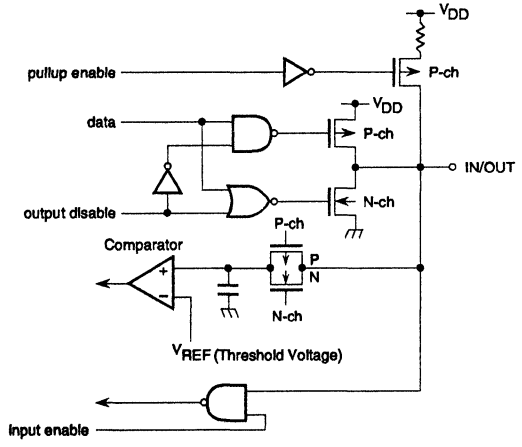
Type 5-A ($P2_1$, $P2_3$, $P3_0$ - $P3_2$, $P3_5$ - $P3_7$, $P5_0$ - $P5_7$, $P6_4$ - $P6_7$)



Type 5-B ($P4_0$ - $P4_7$)



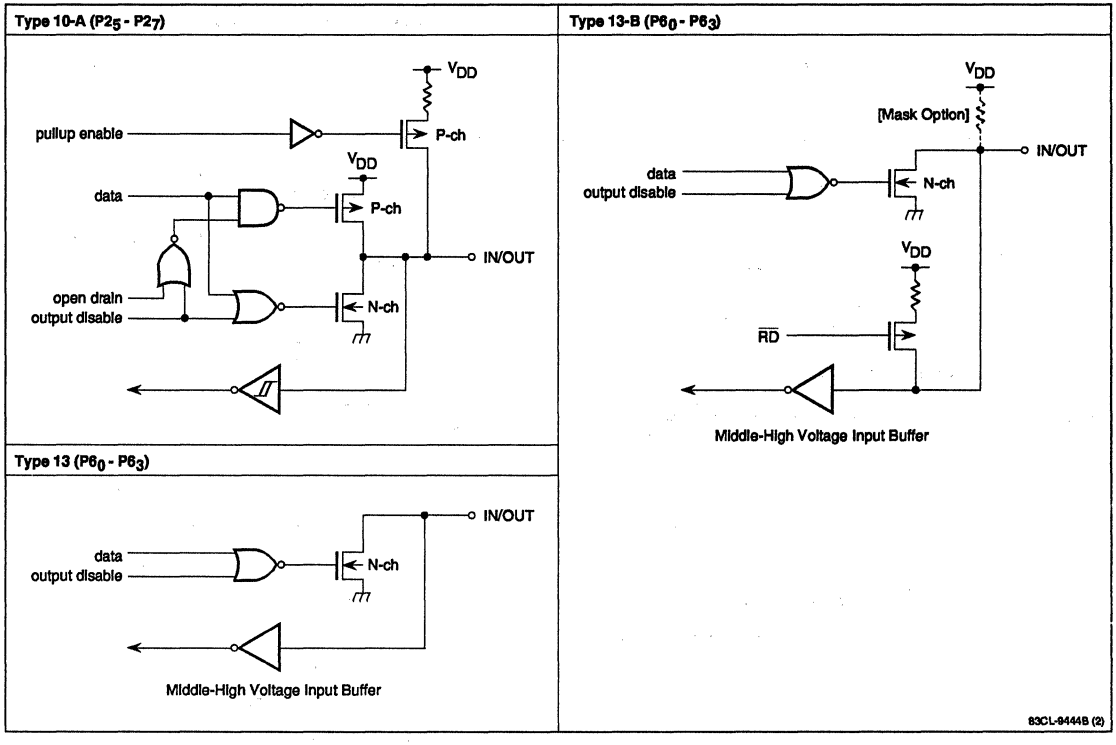
Type 9-B ($P1_0$ - $P1_7$)



83CL-9444B (1)

3d

Figure 4. Pin Input/Output Circuits (cont)



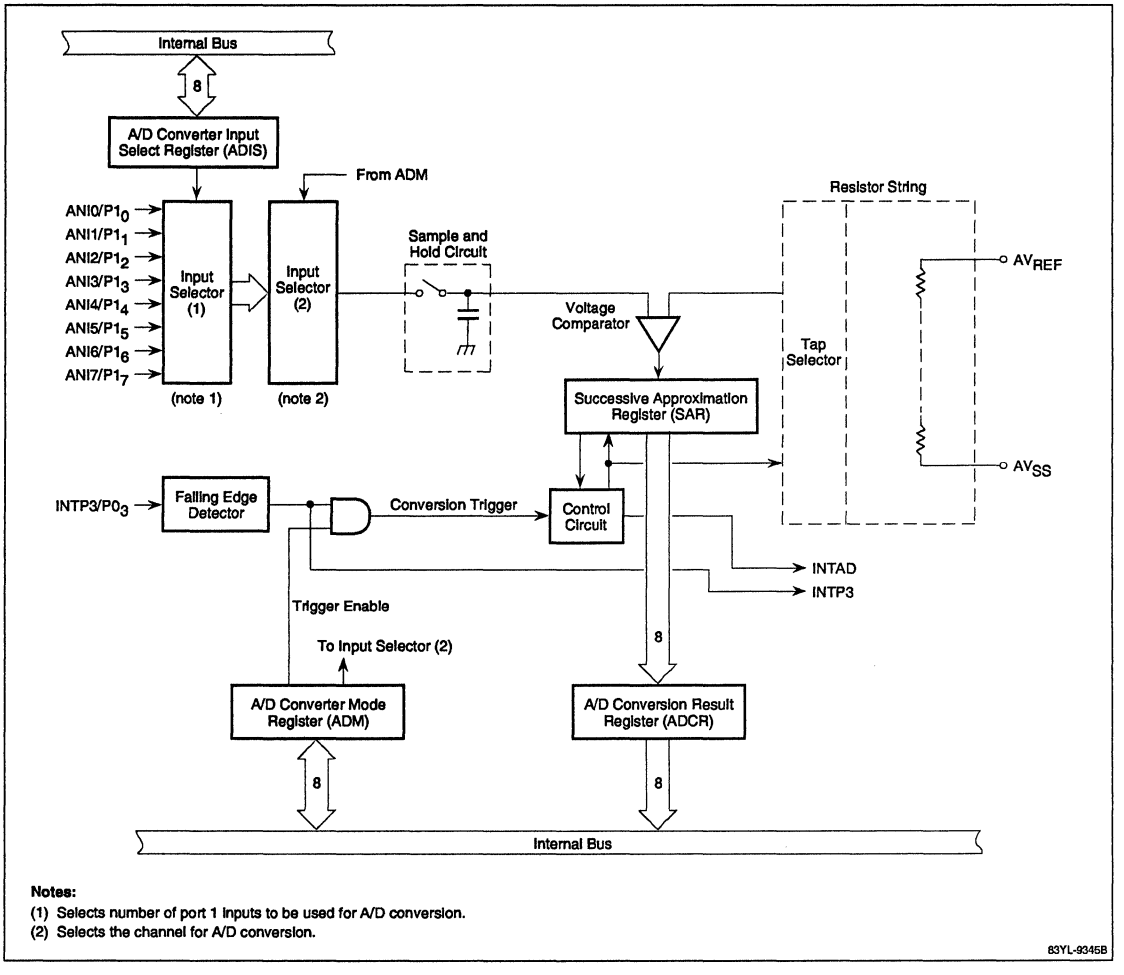
Analog-to-Digital (A/D) Converter

The μPD78014Y family A/D converter (see figure 5) uses the successive-approximation method for converting one of eight multiplexed analog inputs into 8-bit digital data. The conversion time per input is 19.1 μs at 8.38 MHz operation.

The A/D converter input select register (ADIS) selects the number of inputs that are used in A/D conversion. The remaining inputs are used as ports. The A/D input to be converted is selected by programming the A/D converter mode register (ADM). A/D conversion is started by external interrupt INTP3, or by writing to the ADM. When the conversion is completed, the results are stored in the A/D conversion result register (ADCR) and an INTAD interrupt is generated.

If the A/D converter was started by an external interrupt, the A/D converter stops after the interrupt is generated. If the A/D converter was started by software, the A/D converter repeats the conversion until new data is written to the ADM register.

Figure 5. A/D Converter



Notes:
(1) Selects number of port 1 inputs to be used for A/D conversion.
(2) Selects the channel for A/D conversion.

83YL-9345B

3d

Serial Interfaces

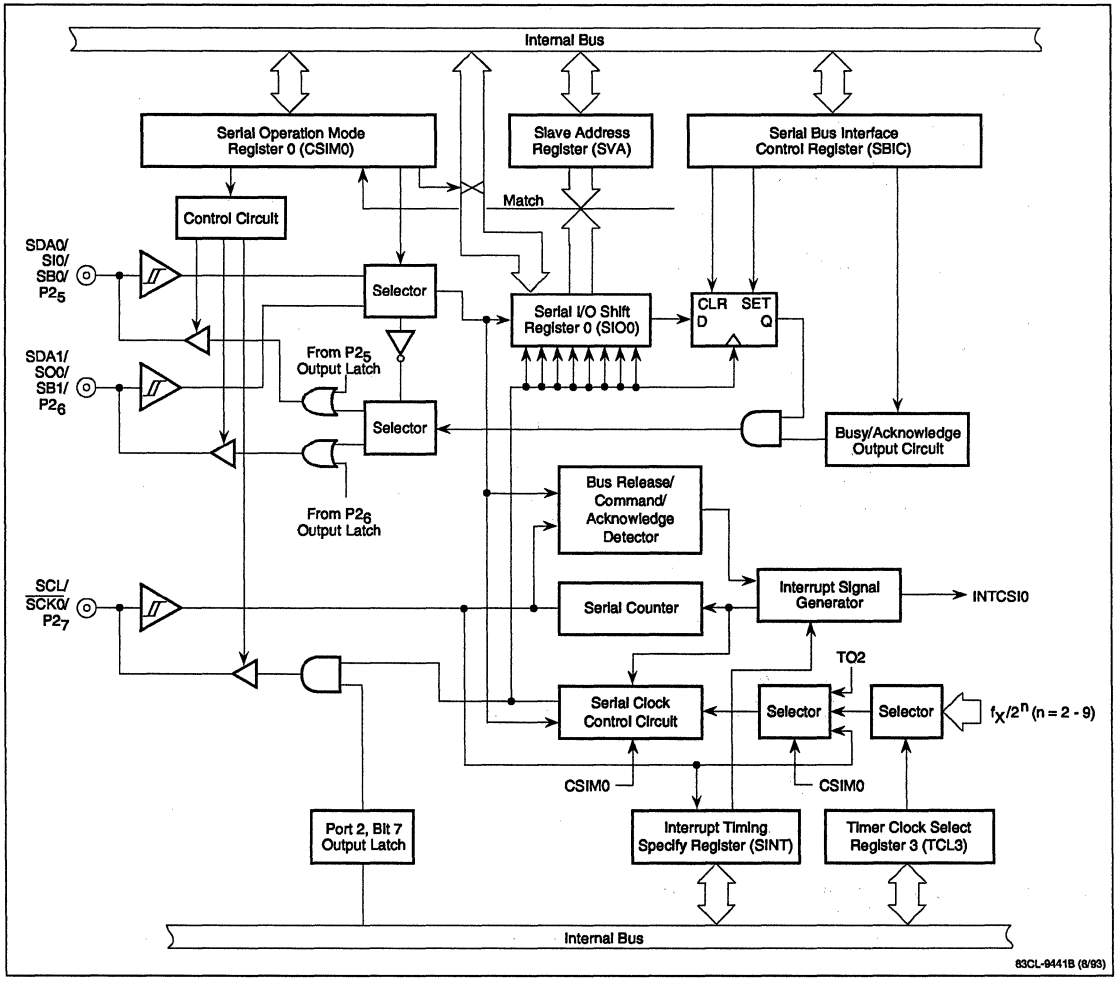
The μPD78014Y family has two independent serial interfaces: serial interface 0 and serial interface 1.

Serial Interface 0. Serial interface 0 is an 8-bit clock synchronous serial interface (figure 6). It can be operated in either a three-wire serial I/O mode, NEC serial bus interface (SBI) mode, two-wire serial I/O mode, or I²C bus mode. The serial clock can be provided from one of eight internal clocks, the output of 8-bit timer register 2, or the external clock line $\overline{SCK0}$ (SCL for I²C bus mode).

In the three-wire serial I/O mode, the 8-bit shift register (SIO0) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO0 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI0 line providing full-duplex operation. The INTCSI0 interrupt is generated after each 8-bit transfer.

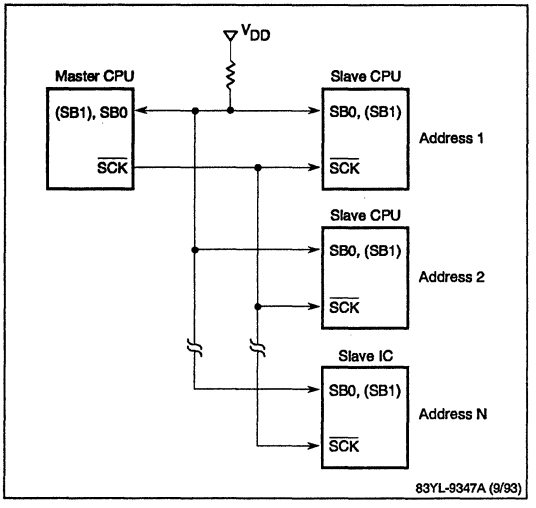
The NEC SBI mode is a two-wire high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are

Figure 6. Serial Interface 0



connected in a master/slave configuration (see figure 7). There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over one of the serial bus lines (SB0 or SB1) using a fixed hardware protocol synchronized with the $\overline{SCK0}$ line. Each slave device of the μPD78014Y family can be programmed to respond in hardware to any one of 256 addresses set in its slave address register (SVA). There are also 256 commands and 256 data types. Since all commands are user definable, many software protocols, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

Figure 7. SBI Mode Master/Slave Configuration



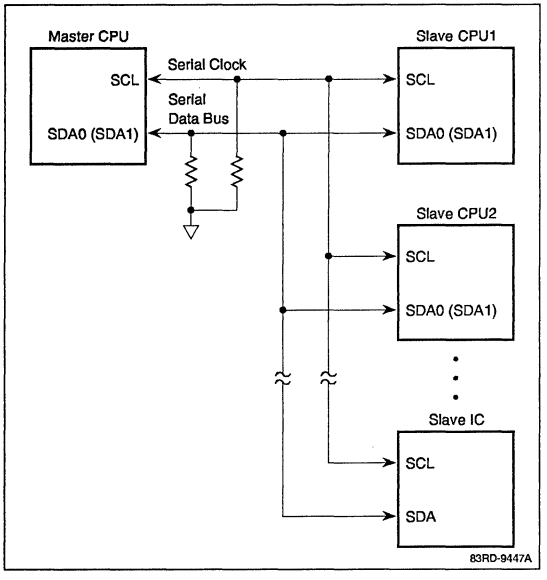
The two-wire serial I/O mode provides half-duplex operation using either the SB0 or SB1 line and the $\overline{SCK0}$ line. Communication format and handshaking can be handled in software by controlling the output levels of the data and clock lines between transfers. When the 8-bit shift register (SIO0) is loaded with a byte of data, eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of either the SB0 or SB1 line MSB first. In addition, this byte of data is also shifted back into SIO0 on the rising edge of these pulses providing a way of verifying that the transmission was correct.

For data reception, the SIO0 register is preloaded with the value FFH. As this data value is shifted out on the falling edge of the serial clock, it disables the n-channel open-drain driver. This allows the receive data to be driven on to the serial line and shifted into the SIO0 register on the rising edge of the serial clock. The INTCSI0 interrupt is generated after each 8-bit transfer.

The I²C bus is a two-wire, high-speed serial bus developed by Philips. The I²C bus configuration has a single master and up to 128 slave devices (see figure 8). The master sends the start condition, 7-bit slave address, one bit indicating the direction of the upcoming data transfer, and the stop condition over one of the serial bus lines (SDA0 or SDA1) using a fixed hardware protocol synchronized with the output of the serial clock line (SCL).

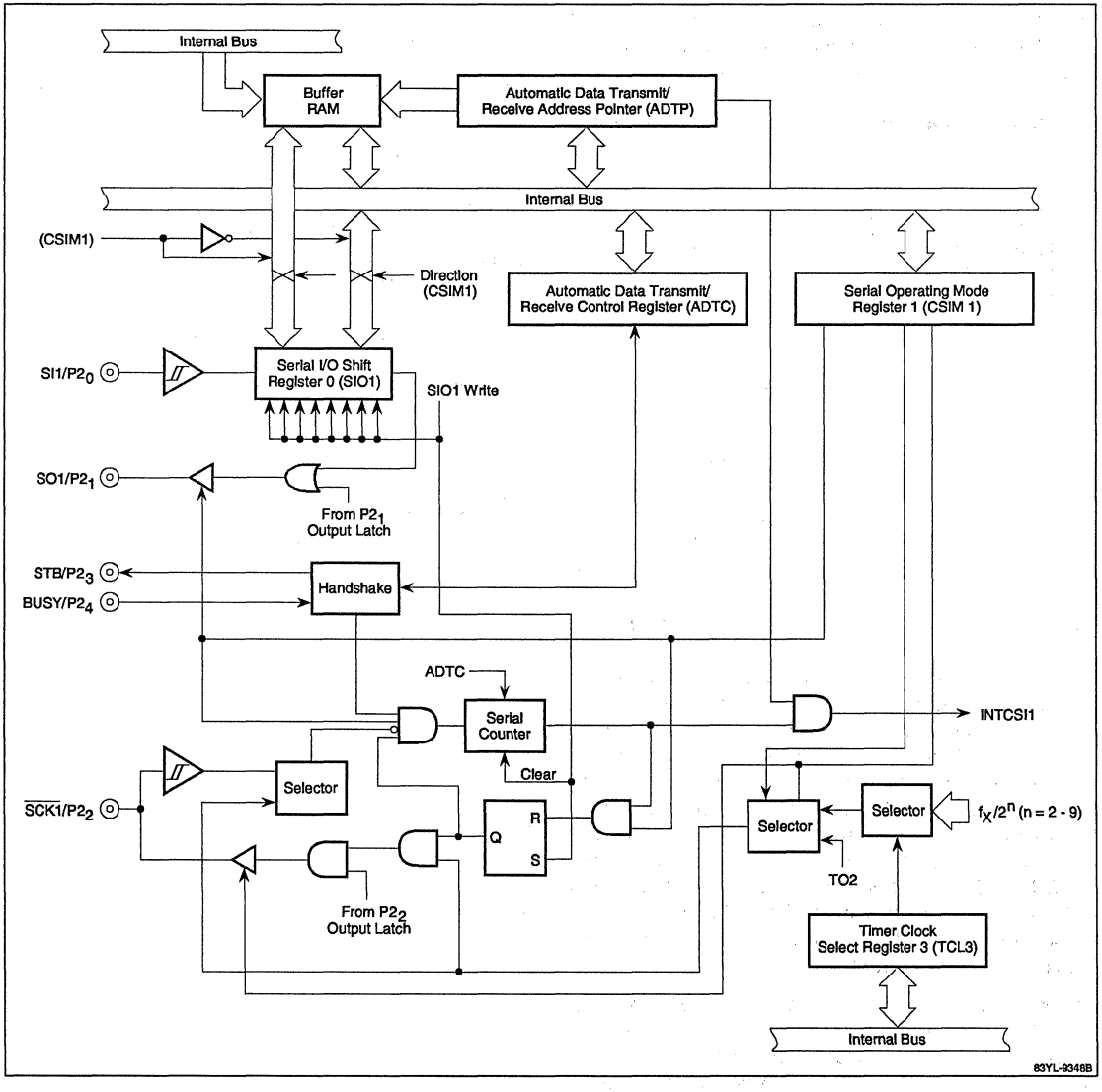
Each slave device of the μPD78014Y family can be programmed to respond in hardware to any of 128 addresses set in its SVA. Depending on the state of the transfer direction bit, either the master or the slave device places additional data on the I²C bus. The device receiving the data returns an acknowledge signal each time it receives 8 bits of data. The slave device can also notify the master device when it is busy by holding SCL low.

Figure 8. I²C Bus Master/Slave Configuration



Serial Interface 1. Serial interface 1 is also an 8-bit clock synchronous serial interface (figure 9). It can be operated in either a three-wire serial I/O mode, or three-wire serial I/O mode with automatic transmit/ receive. The serial clock can also be provided from one of eight internal clocks (common clock for both interfaces), the output of 8-bit timer register 2, or the external clock line $\overline{SCK1}$.

Figure 9. Serial Interface 1



In the three-wire serial I/O mode, the 8-bit shift register (SIO1) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO1 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI1 line providing full-duplex operation. The INTCSI1 interrupt is generated after each 8-bit transfer.

In the three-wire serial I/O mode with automatic transmit/receive, up to 32 bytes of data can be transferred with minimal CPU overhead. The data to be transmitted and received is stored in the buffer RAM. Handshaking using either the BUSY input line, the strobe (STB) output line, or both, can be selected by the program. Error detection of bit drift due to noise is available for each byte transferred when using the BUSY input line. This automatic transmit/receive mode is ideally suited for transferring data to/from external peripheral devices such as onscreen display (OSD) and LCD controller/driver devices.

While in three-wire serial interface mode with automatic data transfer, the interface can be operated as either a full-duplex interface or a transmit-only interface in single or repetitive operation mode. In the full-duplex mode, a byte of data is transferred from the first location in the buffer RAM and shifted out of the SO1 line (either MSB or LSB first) while the received data is shifted into the SI1 line and stored back in the first buffer location. After the preset number of bytes has been transferred, the INTCSI1 interrupt is generated.

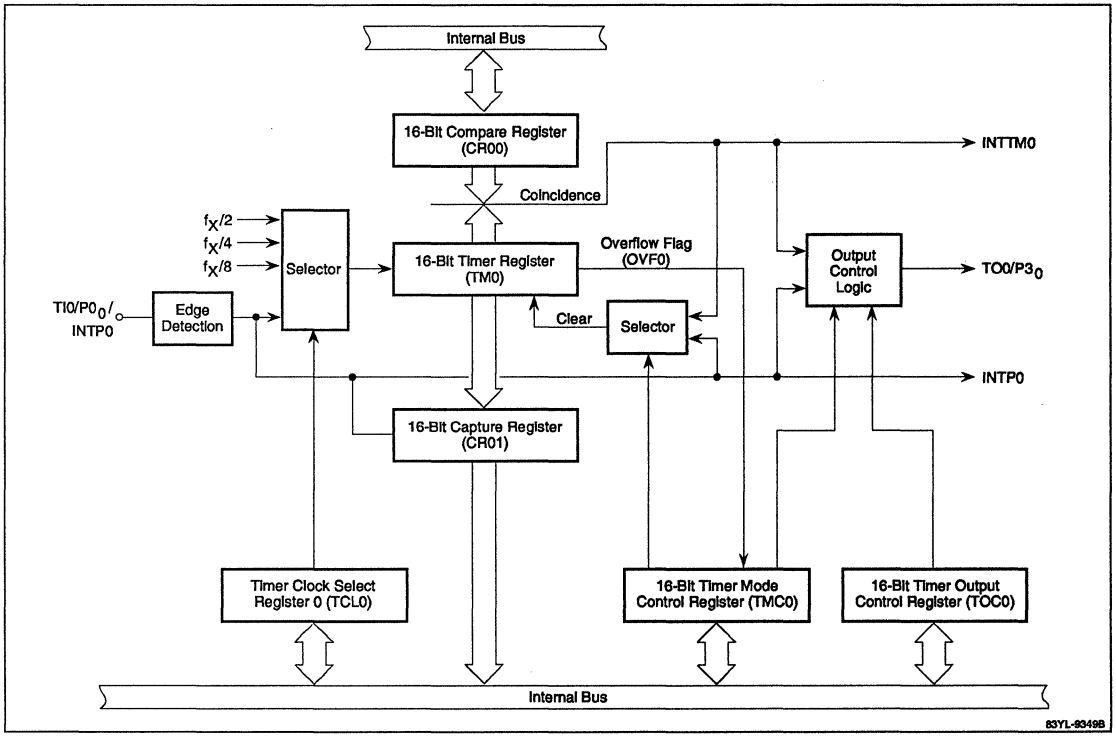
In single-operation transmit mode, the preset number of bytes from the buffer RAM are transmitted out of the SO1 line (either MSB or LSB first) and the INTCSI1 interrupt is generated after all bytes are transferred. In the repetitive operation transmit mode, data in the buffer is transmitted repeatedly.

Timers

The μPD78014Y family has one 16-bit timer/event counter, two 8-bit timer/event counters that can be combined for use as a 16-bit timer/event counter, a clock timer, and a watchdog timer. All of these can be programmed to count a number of prescaled values of the main system clock. In addition, the clock timer can also count the subsystem clock. All of the timer/event counters can count external events.

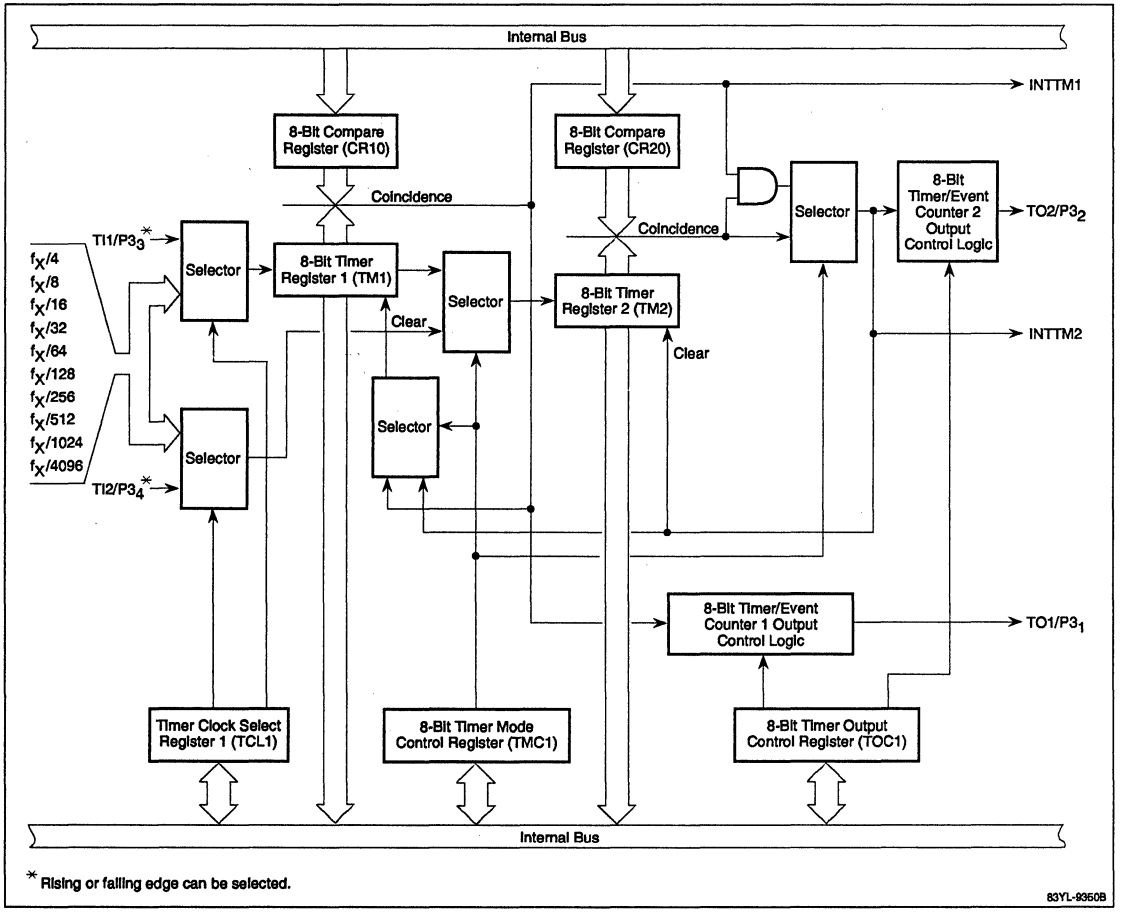
16-Bit Timer/Event Counter 0. Timer/event counter 0 (figure 10) consists of a 16-bit counter (TM0), a 16-bit compare register (CR00), a 16-bit capture register (CR01), and a timer output (TO0). Timer 0 can be used as an interval timer, to count external events on the timer input (TI0) pin, to output a programmable square wave, a 14-bit pulse width modulated output, or to measure pulse widths.

Figure 10. 16-Bit Timer/Event Counter 0



8-Bit Timer/Event Counters 1 and 2. Timer/event counters 1 and 2 (figure 11) each consist of an 8-bit timer (TM1 or TM2), an 8-bit compare register (CR10 or CR20), and timer output control logic (TO1 or TO2). The timers are controlled by registers TCL1, TMC1, and TOC1 via five selectors. Timer/event counters 1 and 2 can each be used as an 8-bit interval timer, to count external events on the timer input pins (TI1 or TI2), or to output a programmable square wave. In addition, timers 1 and 2 also can be combined as a 16-bit timer/event counter and used as a 16-bit interval timer, to count external events on TI1, or to output a programmable square wave on TO2.

Figure 11. 8-Bit Timer/Event Counters 1 and 2



Clock Timer 3. Clock timer 3 (figure 12) is a 5-bit timer that can be used as a time source to keep track of time of day, to release the STOP or HALT modes at regular intervals, or to initiate any other task that must be performed at regular intervals. When driven by the subsystem clock, the clock timer continues to operate in the STOP mode. The clock timer can function as both an interval timer and a clock timer simultaneously. When used as an interval timer, vectored interrupt request INTTM3 is generated at preselected time intervals. With a main system clock of 8.38 MHz or a subsystem clock of 32.768 kHz, the following time intervals can be selected: 489 μs, 978 μs, 1.96 ms, 3.91 ms, 7.82 ms or 15.6 ms.

When used as a clock timer, interrupt request INTWT (not a vectored interrupt) can be generated using the main system clock or subsystem clock every 0.5 or 0.25 seconds.

Watchdog Timer. The watchdog timer (figure 13) can be used as either a watchdog timer or an interval timer. When used as a watchdog timer, it protects against inadvertent program run-away. It can be selected to generate a nonmaskable interupt (INTWDT), which vectors to address 0004H, or to generate an internal reset signal, which vectors to the restart address 0000H if the timer is not cleared by the program before it overflows. Eight program-selectable intervals based on the main system clock are available. With a main system clock of 8.38 MHz, the program selectable intervals are 0.489, 0.978, 1.96, 3.91, 7.82, 15.6, 31.3, and 125 ms. Once the watchdog timer is initialized and started, the timer's mode cannot be changed and the timer can only be stopped by an external reset. When used as an interval timer, maskable interrupts (INTWDT), which vector to address 0004H, are generated repeatedly at a preset interval. The time intervals available are the same as in the watchdog timer mode.

Programmable Clock Output

The μPD78014Y family has a programmable clock output (PCL) that can be used for carrier output for remote controlled transmissions or as a clock output for peripheral devices. The main system clock (f_X) divided by 8, 16, 32, 64, 128, or 256 or the subsystem clock (f_{XT}) can be output on the PCL pin. Frequencies of 1050, 524, 262, 131, 65.5 and 32.7 kHz are available with a main system clock of 8.38 MHz. See figure 14.

Figure 12. Clock Timer 3

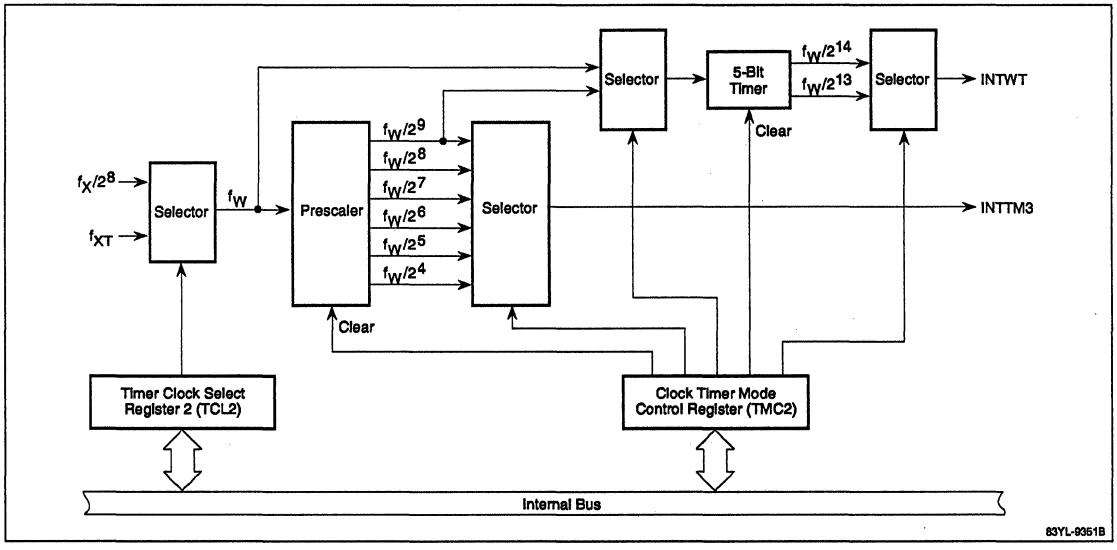


Figure 13. Watchdog Timer

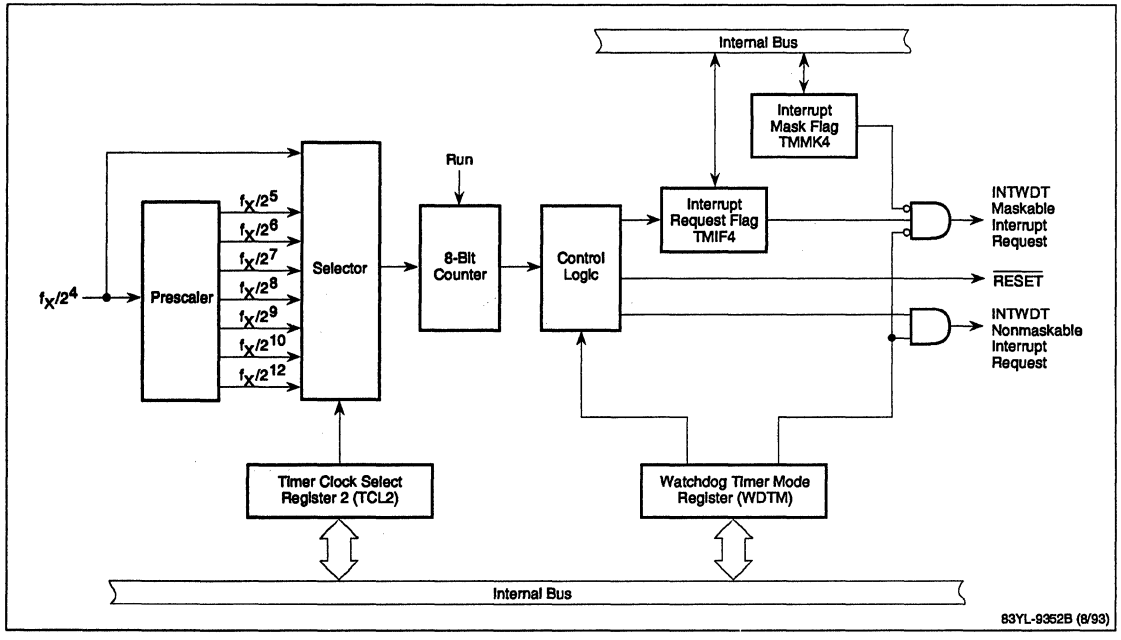
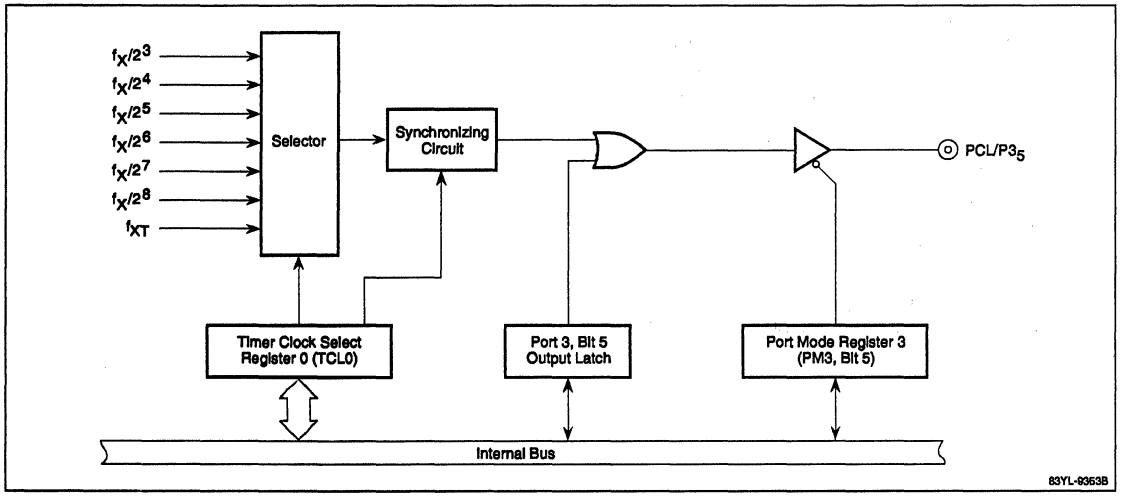


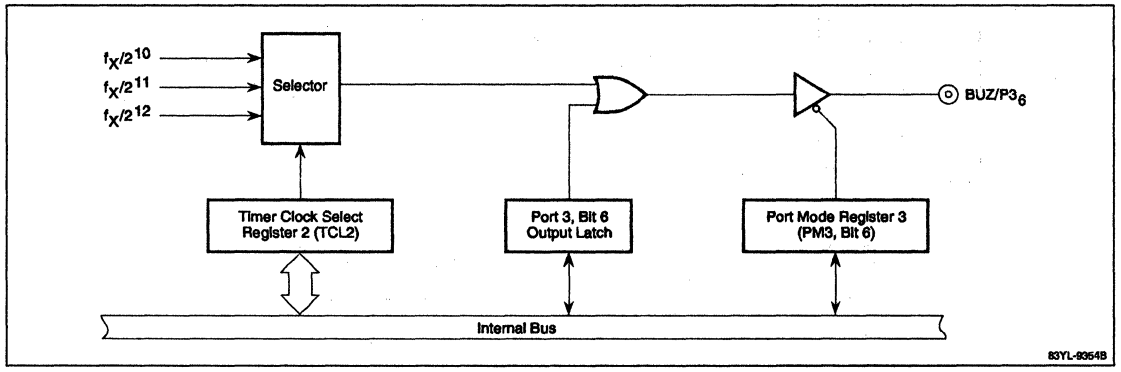
Figure 14. Programmable Clock Output



Buzzer Output

The μPD78014Y family also has a programmable buzzer output (BUZ). The buzzer output frequency can be programmed to be equal to the main system clock (f_X) divided by 1024, 2048, or 4096. With a main system clock of 8.38 MHz, the buzzer can be set to 8.2, 4.1 or 2.0 kHz. See figure 15.

Figure 15. Buzzer Output



Interrupts

The μPD78014Y family has 14 maskable hardware interrupt sources (5 external and 9 internal). Of these 14 interrupt sources, 12 cause a vectored interrupt while the 2 testable inputs only generate an interrupt request. All of the 14 maskable interrupts can be used to release the HALT mode except INTP0. INTP0 cannot be used to release the STOP mode and cannot release the HALT mode when register SCS = 0. In addition, there is one nonmaskable interrupt from the watchdog timer, one software interrupt, and a reset interrupt. The watchdog timer overflow interrupt (interrupt vector table address 0004H) can be initialized to be a nonmaskable interrupt or the highest default priority maskable interrupt. The software interrupt, generated by the BRK instruction, is not maskable. See table 3 and figure 16.

Table 3. Interrupt Sources and Vector Addresses

Type of Request	Default Priority	Signal Name	Interupt Source	Location	Vector Address	Interrupt Configuration
Restart	—	RESET	RESET input pin	External	0000H	—
	—	INTWDT	Watchdog timer overflow (when reset mode selected)	Internal		
Nonmaskable	—	INTWDT	Watchdog timer overflow (when nonmaskable interrupt selected)	Internal	0004H	A
Maskable	0	INTWDT	Watchdog timer overflow (when interval timer selected)	Internal	0004H	B
	1	INTP0	External interrupt edge detection	External	0006H	C
	2	INTP1	External interrupt edge detection	External	0008H	D
	3	INTP2	External interrupt edge detection	External	000AH	D
	4	INTP3	External interrupt edge detection	External	000CH	D
	5	INTCSI0	End of clocked serial interface 0 transfer	Internal	000EH	B
	6	INTCSI1	End of clocked serial interface 1 transfer	Internal	0010H	B
	7	INTTM3	Clock timer reference time interval signal	Internal	0012H	B
	8	INTTM0	16-bit timer/event counter coincidence signal	Internal	0014H	B
	9	INTTM1	8-bit timer/event counter 1 coincidence signal	Internal	0016H	B
	10	INTTM2	8-bit timer/event counter 2 coincidence signal	Internal	0018H	B
	11	INTAD	End of A/D Conversion	Internal	001AH	B
Software	—	—	BRK instruction	Internal	003EH	E
Test input	—	INTWT	Clock timer overflow	Internal	—	F
	—	INTPT4	Port 4 falling edge detection	External	—	F

Interrupt Servicing. The μPD78014Y family provides two levels of programmable hardware priority control and services all interrupt requests, except the two testable interrupts (INTWT and INTPT4), using vectored interrupts. The programmer can choose the priority of servicing each maskable interrupt by using the interrupt control registers.

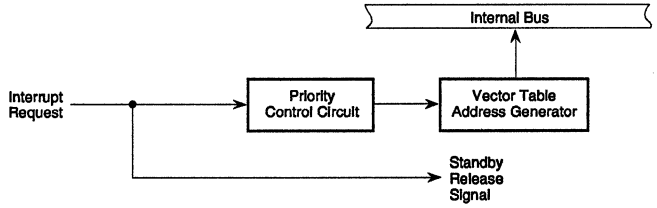
Interrupt Control Registers. The μPD78014Y family has three 16-bit interrupt control registers. The interrupt request flag register (IF0) contains an interrupt request flag for each interrupt except INTPT4. The interrupt mask register (MK0) is used to enable or disable any interrupt except INTPT4. The priority flag register (PR0) can be used to specify a high or a low priority level for each interrupt except the two testable interrupts.

Four other 8-bit registers are associated with interrupt processing. The key return mode register (KRM) contains the KRIF interrupt request flag associated with falling-edge detection on port 4 and the KRMK mask flag used to enable or disable clearing of the standby mode if a falling edge is detected on port 4. The external interrupt mode register (INTM0) is used to select a rising, falling, or both edges as the valid edge for each of the external interrupts INTP0 to INTP2 (INTP3 is always falling edge). The sampling clock select register (SCS) is used to select a sampling clock for the noise eliminator circuit on external interrupt INTP0.

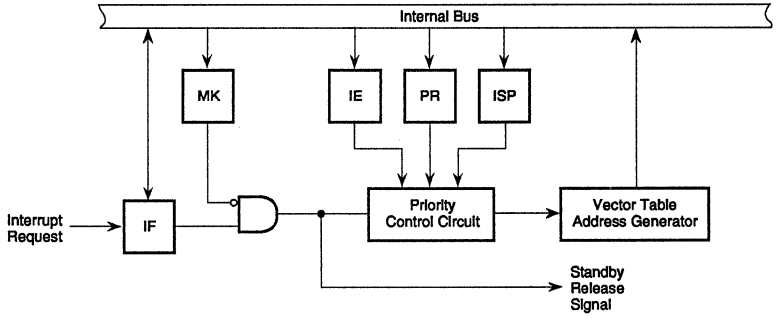
The IE and the ISP bit of the program status word are also used to control interrupts. If the IE bit is 0, all maskable interrupts are disabled. The IE bit can be set or cleared using the EI and DI instructions, respectively, or by directly writing to the PSW. The IE bit is cleared each time an interrupt is accepted. The ISP bit is used by hardware to hold the priority level flag of the interrupt being serviced.

Figure 16. Interrupt Configurations

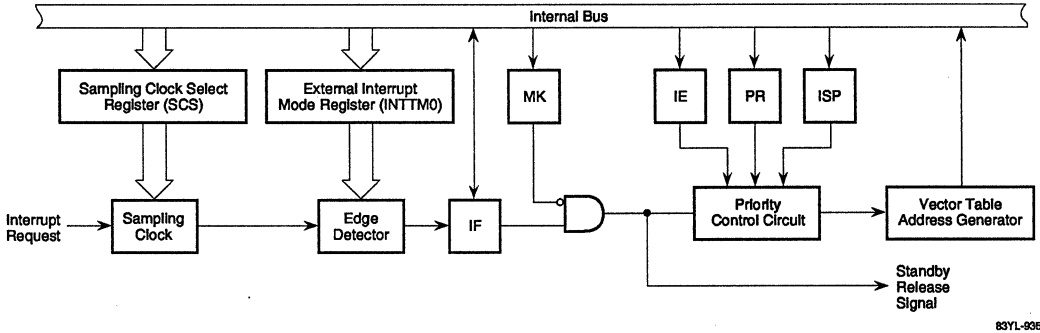
Type A: Internal nonmaskable interrupt



Type B: Internal maskable interrupt



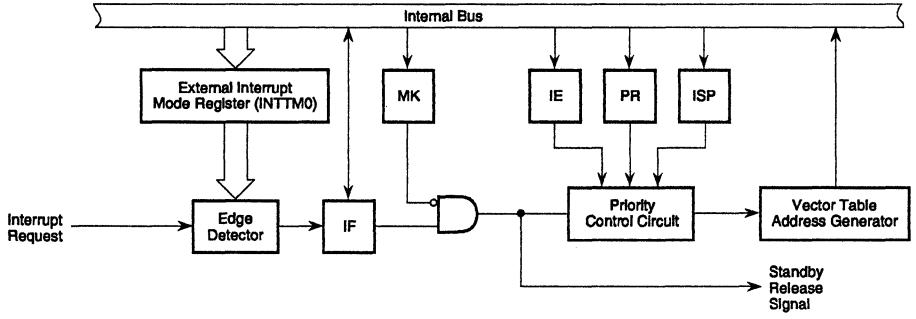
Type C: External maskable interrupt (INTP0)



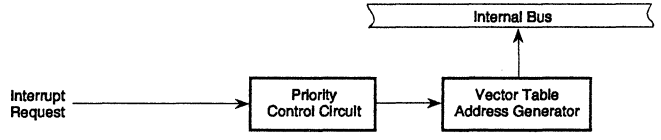
83YL-9365B (1)

Figure 16. Interrupt Configurations (cont)

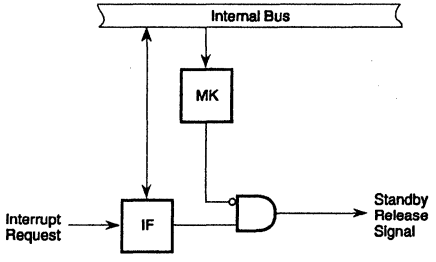
Type D: External maskable interrupt (except INTP0)



Type E: Software interrupt



Type F: Test input



Abbreviations:

IF: Interrupt request flag
IE: Interrupt enable flag
ISP: In-service priority flag
MK: Interrupt mask flag
PR: Priority specify flag

83YL-9355B (2)

3d

Interrupt Priority. If the watchdog timer overflow interrupt (INTWDT) has been initialized to be a nonmaskable interrupt, it has priority over all other interrupts. Two hardware-controlled priority levels are available for all maskable interrupts that generate a vectored interrupt (i.e., all except the two testable interrupts). Either a high or a low priority level can be assigned by software to each of the maskable interrupts. Interrupt requests of the same priority or a priority higher than the processor's current priority level are held pending until interrupts in the current service routine are enabled by software or until one instruction has been executed after returning from the current service routine. Interrupt requests of a lower priority are always held pending until one instruction has been executed after returning from the current service routine.

The default priorities listed in table 3 are fixed by hardware and are effective only when it is necessary to choose between two interrupt requests of the same software-assigned priority. For example, the default priorities would be used after the completion of a high-priority routine, if two interrupts of the same software priority were pending.

The software interrupt, initiated by the BRK instruction, is executed regardless of the processor's priority level and the state of the IE bit. It does not alter the processor's priority level.

Vectored Interrupt Servicing. When a vectored interrupt is acknowledged, the program status word and the program counter are saved on the stack, the processor's priority is set to that specified for the interrupt, the IE bit in the PSW is set to zero, and the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (RETB instruction for the software interrupt) reverses the process and the μPD78014Y family microcontroller resumes the interrupted routine.

Standby Modes

HALT, STOP, and data retention modes are provided to reduce power consumption when CPU action is not required.

The HALT mode is entered by executing a HALT instruction while the CPU is operating from the main system or subsystem clock. In HALT mode, the CPU clock is stopped while the main system and the subsystem clock continue to run. The HALT mode is released by any unmasked interrupt request (except INTP0 if register SCS = 0), a nonmaskable interrupt request, an unmasked test input, or an external reset pulse.

Power consumption may be further reduced by using the STOP mode. The STOP mode is entered by executing a STOP instruction while operating from the main system clock. In STOP mode, the main system clock input pin X1 is internally grounded stopping both the CPU and the peripheral hardware clock. The STOP mode is released by any unmasked interrupt request except INTP0, a nonmaskable interrupt request, an unmasked test input, or an external reset pulse. Any peripheral using the main oscillator as a clock source will also be disabled in the STOP mode and interrupts from such a peripheral cannot be used to exit the STOP mode. Table 4 summarizes both the HALT and STOP standby modes.

Table 4. Standby Mode Operation Status

Item	HALT Mode	STOP Mode
Setting instruction	HALT instruction	STOP instruction
System clock when setting	Main system or subsystem clock	Main system clock
Clock oscillator	Main system and subsystem clocks can oscillate; CPU clock is stopped.	Subsystem clock can oscillate; CPU clock and main system clock are stopped.
CPU	Operation stopped	Operation stopped
Ports	Maintain previous state	Maintain previous state
16-bit timer/event counter	Operational from main system clock	Operation stopped
8-bit timer/event counters	Operational from main system clock	Operational only with TI1 and TI2 as count clock
Clock timer	Operational from main system clock or with f_{XT} as a count clock	Operational only with f_{XT} as count clock
Watchdog timer	Operational from main system clock	Operation stopped
Serial interface 0	Operational from main system clock	Operational only with external clock
Serial interface 1	Operational from main system clock; no automatic transmit/ receive mode	Operational only with external clock; no automatic transmit/ receive mode
A/D converter	Operational from main system clock	Operation stopped
External interrupts	Operational except for INTP0 when its sampling clock is based on the CPU clock	INTP0 not operational; INTP1 to INTP3 operational

When exiting the STOP mode, a wait time occurs before the CPU begins execution to allow the main system clock oscillator circuit to stabilize. The oscillator stabilization time is selected by programming the OSTS register with one of the five values before entering the STOP mode. The values range from 0.98 msec to 31.3 msec at f_X = 8.38 MHz.

Once in the STOP mode, power consumption can be further minimized by lowering the power supply voltage V_{DD} to as little as 2 V. This places the device in the data retention mode. The contents of Internal RAM and the registers are retained. This mode is released by first raising V_{DD} to the proper operating range and then releasing the STOP mode.

External Reset

The μPD78014Y family is reset by taking the $\overline{\text{RESET}}$ pin low or by an overflow of the watchdog timer (if enabled). The $\overline{\text{RESET}}$ input pin is a schmitt-trigger input with hysteresis characteristics to protect against spurious system resets caused by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low for a minimum of 10 μs after the power supply reaches its operating voltage.

There is no functional difference between an external reset and an internal reset caused by the overflow of the watchdog timer. In both cases, the main system clock oscillation is stopped and the subsystem clock oscillation continues. During reset, the program counter is loaded with the address contained in the reset vector (addresses 0000H, 0001H). Once the reset is cleared and the oscillation stabilization time of $2^{18}/f_X$ has elapsed, program execution starts at that address.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = +25°C

Supply voltage, V_{DD}	−0.3 to +7.0 V
Supply voltage, V_{PP}	−0.3 to +13.5 V
Supply voltage, AV_{DD}	−0.3 to V_{DD}+0.3 V
Supply voltage, AV_{REF}	−0.3 to V_{DD}+0.3 V
Supply voltage, AV_{SS}	−0.3 to +0.3 V
Input voltage, V_{I1} (except $P6_0$ to $P6_3$)	−0.3 to V_{DD}+0.3 V
Input voltage, V_{I2} ($P6_0$ to $P6_3$; open drain)	−0.3 to +16 V
Output voltage, V_O	−0.3 to V_{DD}+0.3 V
Analog input voltage, V_{AN} (port 1; analog input pin)	AV_{SS} −0.3 to AV_{REF}+0.3 V
Output current, high; I_{OH}	
Each output pin	−10 mA
Total: ports 2 and 3	−15 mA
Total: port 0 and ports 4 to 6	−15 mA
Output current, low, I_{OL} †	
Each output pin	30 mA peak, 15 mA rms
Total: $P4_0$ to $P4_7$ and $P5_0$ to $P5_5$	100 mA peak, 70 mA rms
Total: $P0_1$ to $P0_3$, $P5_6$, $P5_7$, and $P6_0$ to $P6_7$	100 mA peak, 70 mA rms
Total: $P0_1$ to $P0_3$ and $P6_4$ to $P6_7$	50 mA peak, 20 mA rms
Total: ports 2 and 3	50 mA peak, 20 mA rms
Operating temperature, T_{OPT}	−40 to +85°C
Storage temperature, T_{STG}	−65 to +150°C

† rms value = peak value x (duty cycle) $^{1/2}$

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC characteristics.

Capacitance

T_A = +25°C; V_{DD} = V_{SS} = 0 V

Parameter	Symbol	Max	Unit	Conditions	
Input capacitance	C_{IN}	15	pF	Except $P6_0$ to $P6_3$	f = 1 MHz; unmeasured pins returned to ground
		20	pF	$P6_0$ to $P6_3$	
Output capacitance	C_{OUT}	15	pF	Except $P6_0$ to $P6_3$	
		20	pF	$P6_0$ to $P6_3$	
Input/output capacitance	C_{IO}	15	pF	Except $P6_0$ to $P6_3$	
		20	pF	$P6_0$ to $P6_3$	

Main System Clock Oscillator

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 17.

Type	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Figure 17A)	Oscillation frequency (Note 1)	f_X	1.0		10.0	MHz	V_{DD} = oscillator voltage range
	Oscillation stabilization time (Note 2)				4.0	ms	After V_{DD} reaches oscillator operating voltage
Crystal resonator (Figure 17A)	Oscillation frequency (Note 1)	f_X	1.0	8.38	10.0	MHz	
	Oscillation stabilization time (Note 2)				10	ms	V_{DD} = 4.5 to 6.0 V
					30	ms	
External clock (Figure 17B)	X1 input frequency (Note 1)	f_X	1.0		10.0	MHz	
	X1 input high/low-level width	t_{XH}, t_{XL}	50		500	ns	

Notes:

(1) Oscillator and X1 input frequencies are included only to show the oscillator characteristics. Refer to the AC Characteristics table for actual instruction execution times.

(2) Time required for the oscillator to stabilize after reset or STOP mode is released. The values shown are for the recommended resonators. Values for resonators not shown in this data sheet should be obtained from the manufacturer's specification sheets.

Subsystem Clock Oscillator

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 18.

Type	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Crystal resonator (Figure 18A)	Oscillation frequency (Note 1)	f_{XT}	32	32.768	35	kHz	
	Oscillation stabilization time (Note 2)			1.2	2	s	V_{DD} = 4.5 to 6.0 V
					10	s	
External clock (Figure 18B)	XT1 input frequency (Note 1)	f_{XT}	32		100	kHz	
	XT1 input high/low-level width	t_{XTH}, t_{XTL}	5		15	μs	

Notes:

(1) The oscillator and XT1 input frequencies are included only to show the oscillator characteristics. Refer to the AC Characteristics table for actual instruction execution times.

(2) Time required for the oscillator to stabilize after reset or STOP mode is released. The values shown are for the recommended resonators. Values for resonators not shown in this data sheet should be obtained from the manufacturer's specification sheets.

Figure 17. Main System Clock Configurations

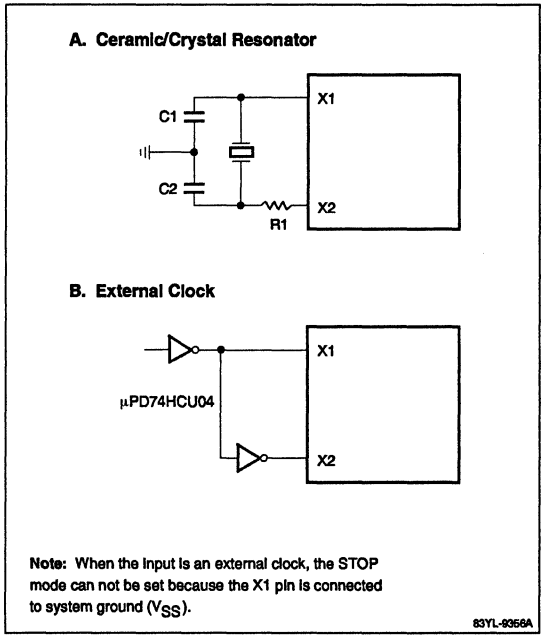
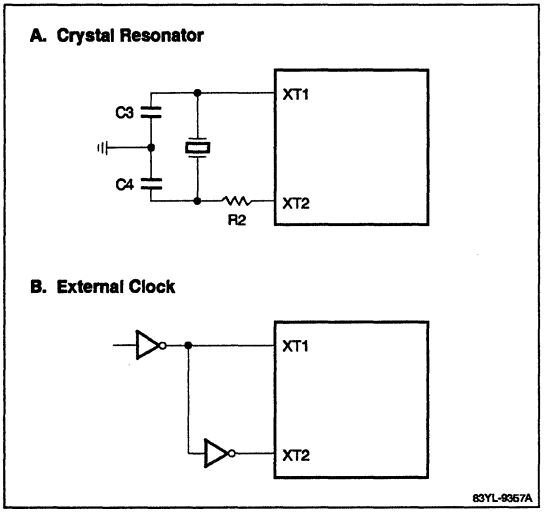


Figure 18. Subsystem Clock Configurations



3d

Recommended Main System Clock Ceramic Resonators

T_A = −40 to +85°C, refer to figure 17A

Part Number (Notes 1 and 2)	Recommended Circuit Constant			Oscillator Voltage Range		Frequency (MHz)
	C1 (pF)	C2 (pF)	R1 (kΩ)	Min (V)	Max (V)	
CSB1000J	100	100	6.8	2.7 (Note 3) 2.8 (Note 4)	6.0	1.00
CSBxxxxJ	100	100	4.7	2.7 (Note 3) 2.8 (Note 4)	6.0	1.01 to 1.25
CSAx.xxxMK	100	100	0	2.7 (Note 3) 2.8 (Note 4)	6.0	1.26 to 1.79
CSAx.xxMG (Note 3)	100	100	0	2.7	6.0	1.8 to 2.44
CSAx.xxMG093 (Note 4)	100	100	0	2.7	6.0	
CSTx.xxMG (Note 3)	0 (Note 5)	0 (Note 5)		2.7	6.0	
CSTx.xxMG093 (Note 4)	0 (Note 5)	0 (Note 5)		2.7	6.0	
CSAx.xxMG	30	30	0	2.7	6.0	2.45 to 4.18
CSTx.xxMGW	0 (Note 5)	0 (Note 5)	0	2.7	6.0	
CSAx.xxMG (Note 3)	30	30	0	2.7	6.0	4.19 to 6.00
CSAx.xxMGU (Note 4)	30	30	0	2.7	6.0	
CSTx.xxMGW (Note 3)	0 (Note 5)	0 (Note 5)	0	2.7	6.0	
CSTx.xxMGWU (Note 4)	0 (Note 5)	0 (Note 5)	0	2.7	6.0	
CSAx.xxMT	30	30	0	2.7 (Note 3) 3.0 (Note 4)	6.0	6.01 to 10.0
CSTx.xxMTW	0 (Note 5)	0 (Note 5)	0	2.7 (Note 3) 3.0 (Note 4)	6.0	

Notes:

(1) Manufactured by Murata Mfg. Co., Ltd.

(2) x.xx indicates frequency

(3) μPD7801xY only.

(4) μPD78P014Y only

(5) C1 and C2 are contained in the ceramic resonators.

Recommended Subsystem Clock Crystal Resonators (μPD7801xY)

T_A = −40 to +60°C, refer to figure 18A

Part Number †	Frequency (kHz)	Recommended Circuit constant			Oscillator Voltage Range	
		C3 (pF)	C4 (pF)	R2 (kΩ)	Min (V)	Max (V)
DT-38 (1TA252 E00, load capacitance 6.3 pF)	32.768	12	12	100	2.7	6.0

† Manufactured by Daishinku

DC Characteristics

T_A = −40 to +85°C; V_{DD} = +2.7 to 6.0 V; refer to figures 19-25

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Other than below
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	$P0_0$ to $P0_4$, $P2_0$, $P2_2$, $P2_4$ to $P2_7$, $P3_3$, $P3_4$, $\overline{RESET}$
	V_{IH3}	0.7 V_{DD}		15	V	$P6_0$ to $P6_3$; open-drain
	V_{IH4}	V_{DD} − 0.5		V_{DD}	V	X1, X2
	V_{IH5}	V_{DD} − 0.5		V_{DD}	V	V_{DD} = 4.5 to 6.0 V; XT1, XT2
		V_{DD} − 0.3		V_{DD}	V	μPD7801xY; XT1, XT2
		V_{DD} − 0.2		V_{DD}	V	μPD78P014Y; XT1, XT2
Low-level input voltage	V_{IL1}	0		0.3 V_{DD}	V	Other than below
	V_{IL2}	0		0.2 V_{DD}	V	$P0_0$ to $P0_4$, $P2_0$, $P2_2$, $P2_4$ to $P2_7$, $P3_3$, $P3_4$, $\overline{RESET}$
	V_{IL3}	0		0.3 V_{DD}	V	$P6_0$ to $P6_3$; V_{DD} = 4.5 to 6.0 V
		0		0.2 V_{DD}	V	$P6_0$ to $P6_3$
	V_{IL4}	0		0.4	V	X1, X2
	V_{IL5}	0		0.4	V	XT1, XT2; V_{DD} = 4.5 to 6.0 V
		0		0.3	V	XT1, XT2
High-level output voltage	V_{OH1}	V_{DD} − 1.0			V	V_{DD} = 4.5 to 6.0 V, I_{OH} = −1.0 mA
		V_{DD} − 0.5			V	I_{OH} = −100 μA
Low-level output voltage	V_{OL1}		0.4	2.0	V	$P5_0$ to $P5_7$, $P6_0$ to $P6_3$; V_{DD} = 4.5 to 6.0 V, I_{OL} = 15 mA
				0.4	V	Other than above; V_{DD} = 4.5 to 6.0 V, I_{OL} = 1.6 mA
	V_{OL2}			0.2 V_{DD}	V	SB0, SB1, $\overline{SCK0}$; V_{DD} = 4.5 to 6.0 V, open-drain, pullup resistance = 1 kΩ
	V_{OL3}			0.5	V	I_{OL} = 400 μA
High-level input leakage current	I_{LIH1}			3	μA	V_{IN} = V_{DD}; except X1, X2, XT1, XT2
	I_{LIH2}			20	μA	V_{IN} = V_{DD}; X1, X2, XT1, XT2
	I_{LIH3}			80	μA	V_{IN} = 15 V; $P6_0$ to $P6_3$
Low-level input leakage current	I_{LIL1}			−3	μA	V_{IN} = 0 V; except X1, X2, XT1, XT2
	I_{LIL2}			−20	μA	V_{IN} = 0 V; X1, X2, XT1, XT2
	I_{LIL3}			−3	μA	V_{IN} = 0 V; $P6_0$ to $P6_3$ (Note 1)
Output leakage current high	I_{LOH1}			3	μA	V_{OUT} = V_{DD}
Output leakage current low	I_{LOL}			−3	μA	V_{OUT} = 0 V
Mask option pullup resistor	R_1	20	40	90	kΩ	V_{IN} = 0 V; $P6_0$ TO $P6_3$, μPD7801xY only
Software pullup resistor	R_2	15	40	90	kΩ	V_{DD} = 4.5 to 6.0 V; V_{IN} = 0 V, $P0_1$ to $P0_3$, ports 1 to 5, $P6_4$ to $P6_7$
		20		500	kΩ	V_{DD} = 2.7 to 4.5 V; V_{IN} = 0 V, $P0_1$ to $P0_3$, ports 1 to 5, $P6_4$ to $P6_7$

3d

DC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Power supply current (μPD7801xY)	I_{DD1}		7.5	22.5	mA	8.38 MHz crystal oscillation operating mode; V_{DD} = 5.0 V ±10% (Note 2)
			0.8	2.4	mA	8.38 MHz crystal oscillation operating mode; V_{DD} = 3.0 V ±10% (Note 3)
	I_{DD2}		1.4	4.2	mA	8.38 MHz crystal oscillation HALT mode; V_{DD} = 5.0 V ±10%
			550	1650	μA	8.38 MHz crystal oscillation HALT mode; V_{DD} = 3.0 V ±10%
	I_{DD3}		60	120	μA	32.768 kHz crystal oscillation operating mode; V_{DD} = 5.0 V ±10%, X1 STOP mode, CPU operating from subsystem clock.
			35	70	μA	32.768 kHz crystal oscillation operating mode; V_{DD} = 3.0 V ±10%, X1 STOP mode, CPU operating from subsystem clock.
	I_{DD4}		25	50	μA	32.768 kHz crystal oscillation HALT mode; V_{DD} = 5.0 V ±10%, X1 STOP mode
			5	10	μA	32.768 kHz crystal oscillation HALT mode; V_{DD} = 3.0 V ±10%, X1 STOP mode
	I_{DD5}		1	20	μA	XT1 = 0 V STOP mode when feedback resistor is connected; V_{DD} = 5.0 V ±10%
			0.5	10	μA	XT1 = 0 V STOP mode when feedback resistor is connected; V_{DD} = 3.0 V ±10%
	I_{DD6}		0.1	20	μA	XT1 = 0 V STOP mode when feedback resistor is disconnected; V_{DD} = 5.0 V ±10%
			0.05	10	μA	XT1 = 0 V STOP mode when feedback resistor is disconnected; V_{DD} = 3.0 V ±10%
Power supply current (μPD78P014Y)	I_{DD1}		9	27	mA	8.38 MHz crystal oscillation operating mode; V_{DD} = 5.0 V ±10% (Note 2)
			1	3	mA	8.38 MHz crystal oscillation operating mode; V_{DD} = 3.0 V ±10% (Note 3)
	I_{DD2}		1.4	4.2	mA	8.38 MHz crystal oscillation HALT mode; V_{DD} = 5.0 V ±10%
			550	1650	μA	8.38 MHz crystal oscillation HALT mode; V_{DD} = 3.0 V ±10%
	I_{DD3}		90	180	μA	32.768 kHz crystal oscillation operating mode; V_{DD} = 5.0 V ±10%, X1 STOP mode, CPU operating from subsystem clock.
			50	100	μA	32.768 kHz crystal oscillation operating mode; V_{DD} = 3.0 V ±10%, X1 STOP mode, CPU operating from subsystem clock.
	I_{DD4}		25	50	μA	32.768 kHz crystal oscillation HALT mode; V_{DD} = 5.0 V ±10%, X1 STOP mode
			5	10	μA	32.768 kHz crystal oscillation HALT mode; V_{DD} = 3.0 V ±10%, X1 STOP mode
	I_{DD5}		1	30	μA	XT1 = 0 V STOP mode when feedback resistor is connected; V_{DD} = 5.0 V ±10%
			0.5	10	μA	XT1 = 0 V STOP mode when feedback resistor is connected; V_{DD} = 3.0 V ±10%

DC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Power supply current (μPD78P014Y) (cont)	I_{DD6}		0.1	30	μA	XT1 = 0 V STOP mode when feedback resistor is disconnected; V_{DD} = 5.0 V ±10%
			0.05	10	μA	XT1 = 0 V STOP mode when feedback resistor is disconnected; V_{DD} = 3.0 V ±10%

Notes:

(1) $P6_0$ to $P6_3$ become −200 μA (max.) for only 1 clock cycle during input instruction execution (no wait) and −3 μA (max.) during instruction other than input.

(2) When operated in the high-speed mode with the processor clock control register set to 00H.

(3) When operated in low-speed mode with the processor clock control register set to 04H.

Figure 19. I_{DD} vs V_{DD}, f_x = 8.38 MHz (μPD7801xY)

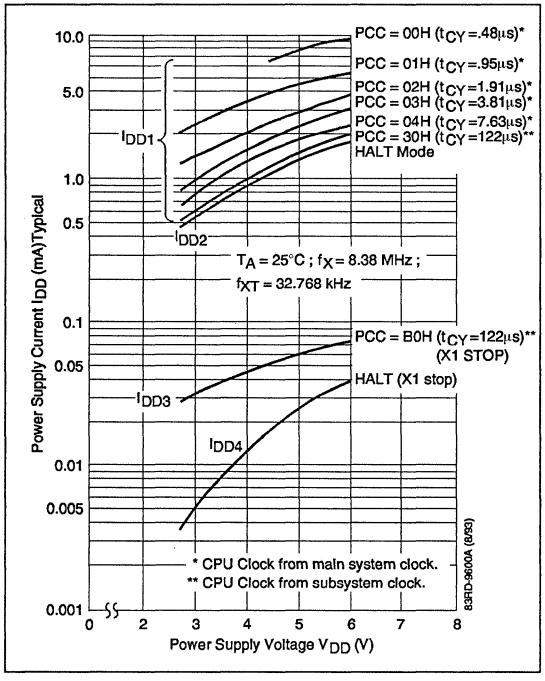
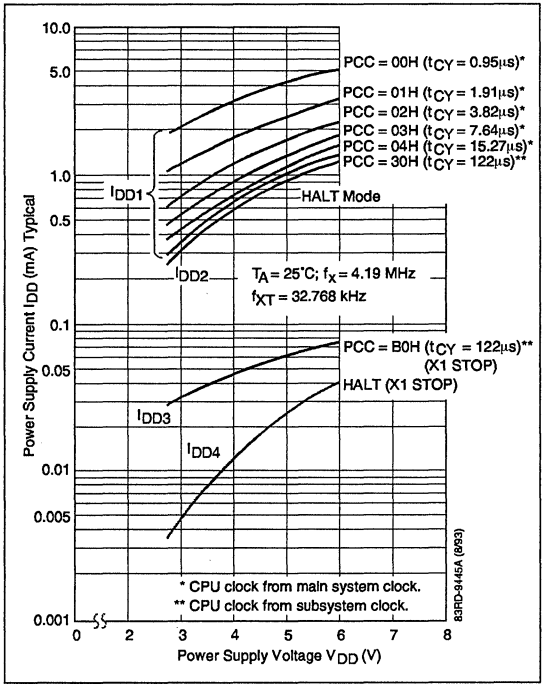


Figure 20. I_{DD} vs V_{DD}, f_x = 4.19 MHz (μPD7801xY)



3d

Figure 21. I_{DD} vs V_{DD} (μPD78P014Y)

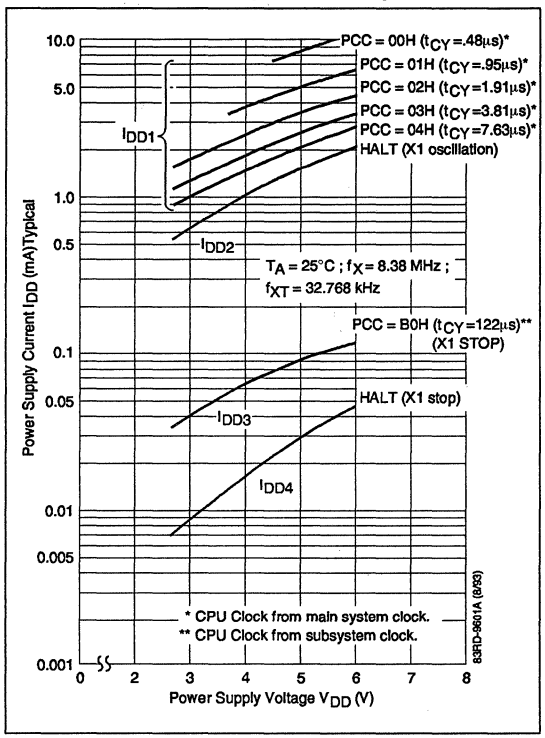


Figure 22. I_{OL} vs V_{OL} (Ports 0, Ports 2-5, $P6_4$-$P6_7$)

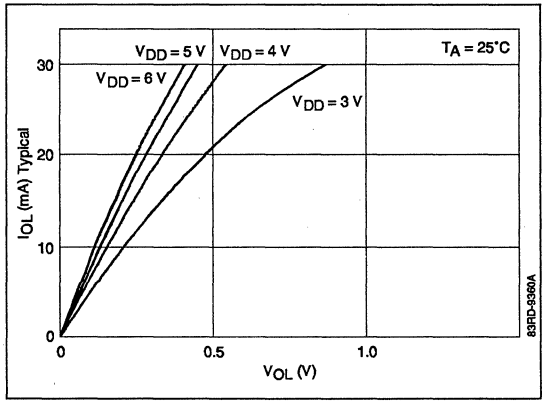


Figure 23. I_{OL} vs V_{OL} (Port 1)

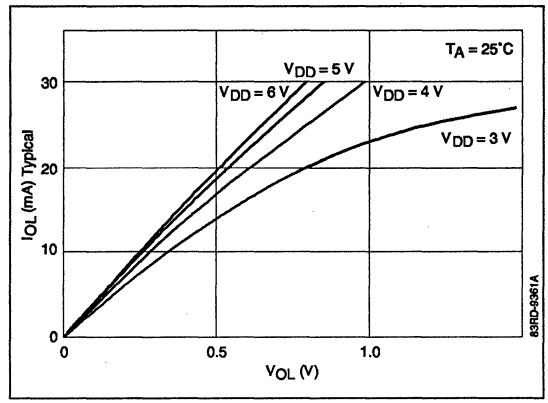


Figure 24. I_{OL} vs V_{OL} ($P6_0$-$P6_3$)

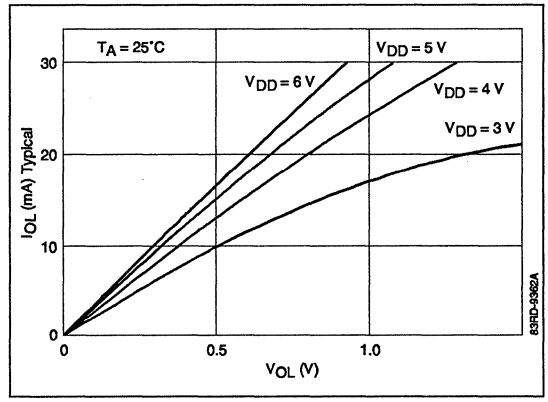
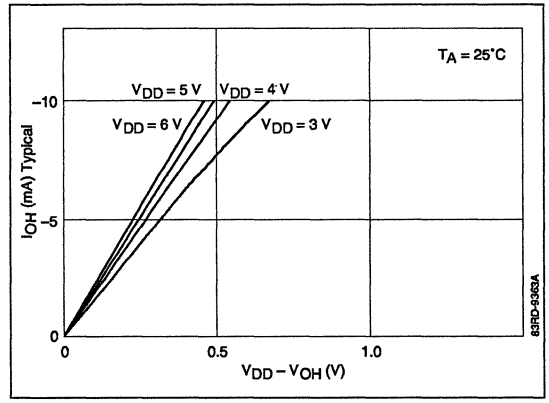


Figure 25. I_{OH} vs V_{DD} – V_{OH} (Ports 0-5, $P6_4$-$P6_7$)



AC Characteristics

T_A = −40 to +85°C; V_{DD} = 2.7 to 6 V ; refer to figures 26 through 32

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time (Min. instruction execution time)	t_{CY}	0.4		64	μs	V_{DD} = 4.5 to 6.0 V; operating on main system clock (μPD7801xY)
		0.96		64	μs	Operating on main system clock (μPD7801xY)
		0.48		64	μs	V_{DD} = 4.5 to 6.0 V; operating on main system clock (μPD78P014Y)
		1.91		64	μs	Operating on main system clock (μPD78P014Y)
		0.4		64	μs	T_A = −40 to +40°C, V_{DD} = 4.75 to 6.0 V; operating on main system clock (μPD78P014Y)
		0.96		64	μs	T_A = −40 to +40°C; operating on main system clock (μPD78P014Y)
		114	122	125	μs	Operating on subsystem clock
TI input frequency	f_{TI}	0		4	MHz	V_{DD} = 4.5 to 6.0 V
		0		275	kHz	
TI input high/low-level width	t_{TIH}, t_{TIL}	100			ns	V_{DD} = 4.5 to 6.0 V
		1.8			μs	
Interrupt input high/low-level width	t_{INTH}, t_{INTL}	$8/f_{sam}$ (Note 1)			μs	INTP0
		10			μs	INTP1 to INTP3
		10			μs	KR0 to KR7 (Note 2)
$\overline{\text{RESET}}$ low-level width	t_{RST}	10			μs	

Notes:

(1) By using bits 0 and 1 of the sampling clock select (SCS) register in conjunction with bits 0 to 2 of the processor clock control (PCC) register, f_{sam} can be set to $f_X/2^{N+1}$ (where N = 0 to 4), $f_X/64$, or $f_X/128$.

(2) Port 4 falling-edge detection input.

3d

Figure 26. Main System Clock Operation t_{CY} vs V_{DD} (μPD7801xY)

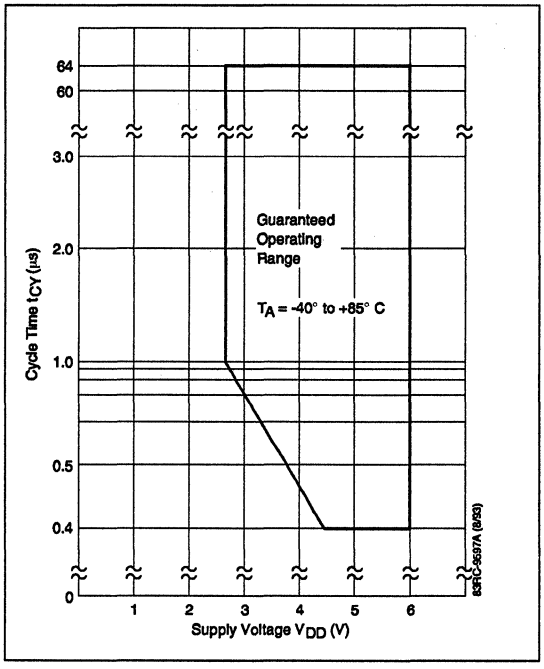
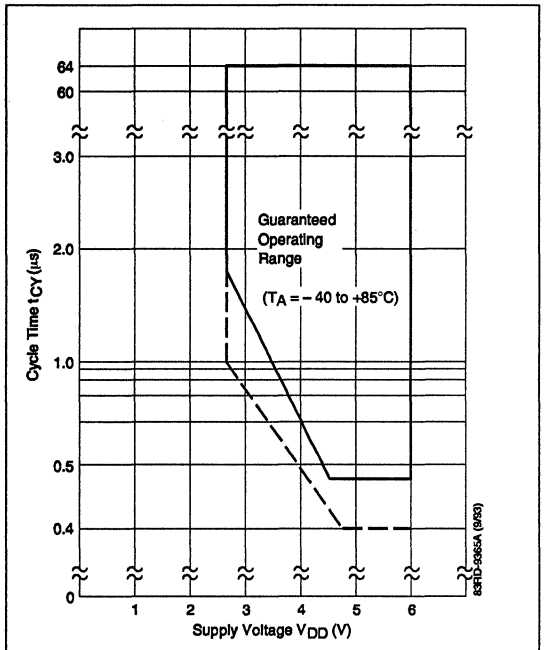


Figure 27. Main System Clock Operation t_{CY} vs V_{DD} (μPD78P014Y)



Note:
When T_A = -40 to +40°C, guaranteed operating range is extended to the dotted line.

Figure 28. AC Timing Measurements Points (except X1 and XT1)

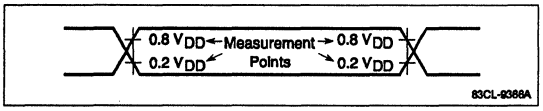


Figure 29. Clock AC Timing Points X1 and XT1

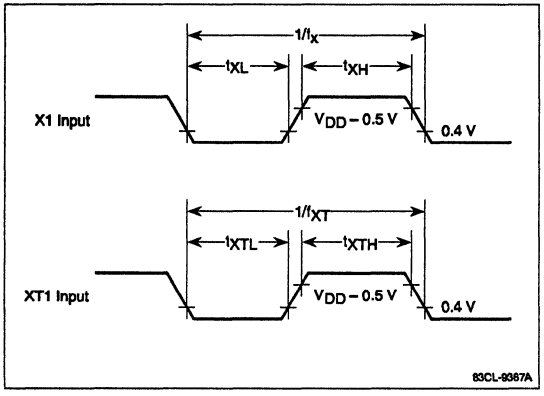


Figure 30. TI Timing

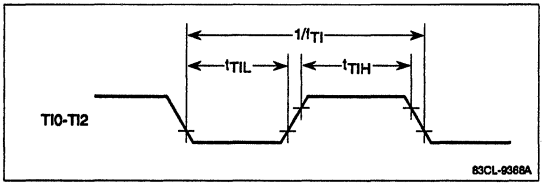


Figure 31. Interrupt Input Timing

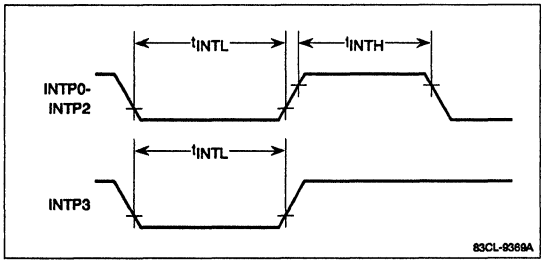
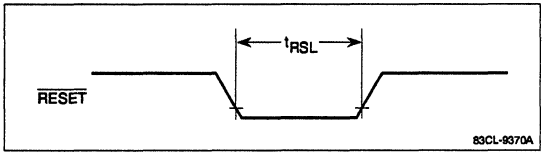


Figure 32. $\overline{\text{RESET}}$ Input Timing



3d

Read/Write Operation

T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 V; refer to figures 33 through 36

Parameter	Symbol	Min	Max	Unit	Conditions
ASTB high-level width	t_{ASTH}	0.5 t_{CY}		ns	
Address setup time to ASTB ↓	t_{ADS}	0.5 t_{CY} − 30		ns	
Address hold time from ASTB ↓	t_{ADH}	10		ns	Load resistor ≥ 5 kΩ
Data input time from address	t_{ADD1}		(2+2n)t_{CY} − 50	ns	Instruction fetch
	t_{ADD2}	5	(3+2n)t_{CY} − 100	ns	Data access
Data input time from $\overline{RD}$ ↓	t_{RDD1}		(1+2n)t_{CY} − 25	ns	Instruction fetch
	t_{RDD2}		(2.5+2n)t_{CY} − 100	ns	Data access
Read data hold time	t_{RDH}	0		ns	
$\overline{RD}$ low-level width	t_{RDL1}	(1.5+2n)t_{CY} − 20		ns	Instruction fetch
	t_{RDL2}	(2.5+2n)t_{CY} − 20		ns	Data access
$\overline{WAIT}$ ↓ input time from $\overline{RD}$ ↓	t_{RDWT1}		0.5 t_{CY}	ns	Instruction fetch
	t_{RDWT2}		1.5 t_{CY}	ns	Data access
$\overline{WAIT}$ ↓ input time from $\overline{WR}$ ↓	t_{WRWT}		0.5 t_{CY}	ns	
$\overline{WAIT}$ low-level width	t_{WTL}	(0.5+2n)t_{CY} + 10	(2+2n)t_{CY}	ns	
Write data setup time to $\overline{WR}$ ↑	t_{WDS}	100		ns	
Write data hold time from $\overline{WR}$ ↑	t_{WDH}	5		ns	
$\overline{WR}$ low-level width	t_{WRL1}	(2.5+2n)t_{CY} − 20		ns	
$\overline{RD}$ ↓ delay time from ASTB ↓	t_{ASTRD}	0.5 t_{CY} − 30		ns	
$\overline{WR}$ ↓ delay time from ASTB ↓	t_{ASTWR}	1.5 t_{CY} − 30		ns	
ASTB ↑ delay time from $\overline{RD}$ ↑ (external fetch)	t_{RDAST}	t_{CY} − 10	t_{CY} + 40	ns	
Address hold time from $\overline{RD}$ ↑ (external fetch)	t_{RDADH}	t_{CY}	t_{CY} + 50	ns	
Write data output time from $\overline{RD}$ ↑	t_{RDWD}	10		ns	
WR ↓ delay time from write data	t_{WDWR}	0.5 t_{CY} − 120	0.5 t_{CY}	ns	V_{DD} = 4.5 to 6.0 V
		0.5 t_{CY} − 170	0.5 t_{CY}	ns	
Address hold time from $\overline{WR}$ ↑	t_{WRADH}	t_{CY}	t_{CY} + 60	ns	V_{DD} = 4.5 to 6.0 V
		t_{CY}	t_{CY} + 100	ns	
$\overline{RD}$ ↑ delay time from $\overline{WAIT}$ ↑	t_{WTRD}	0.5 t_{CY}	2.5 t_{CY} + 80	ns	
$\overline{WR}$ ↑ delay time from $\overline{WAIT}$ ↑	t_{WTWR}	0.5 t_{CY}	2.5 t_{CY} + 80	ns	

Notes:

(1) $t_{CY} = t_{CY}/4$

(2) n indicates number of waits.

(3) C_L = 100 pF

Figure 33. Read Operation; External Fetch (No Wait)

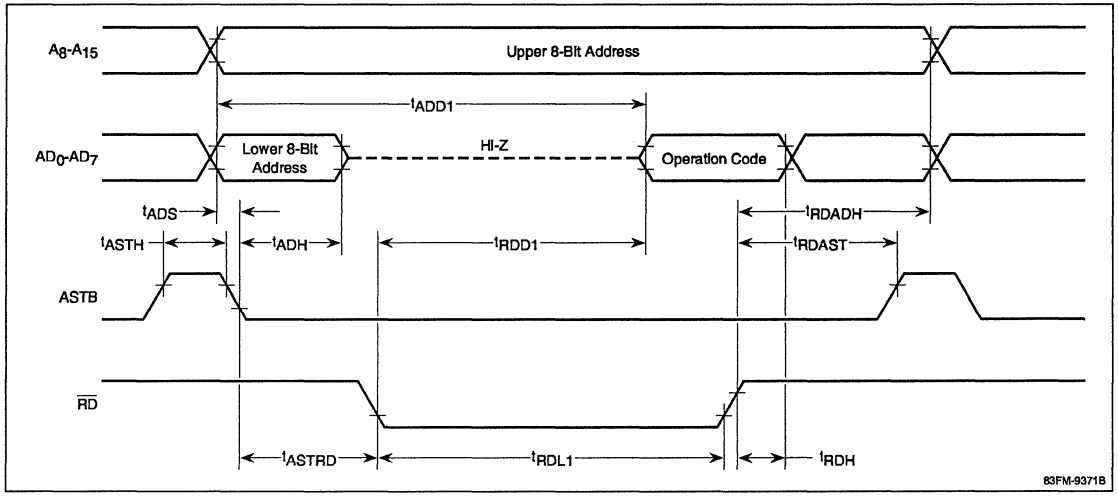
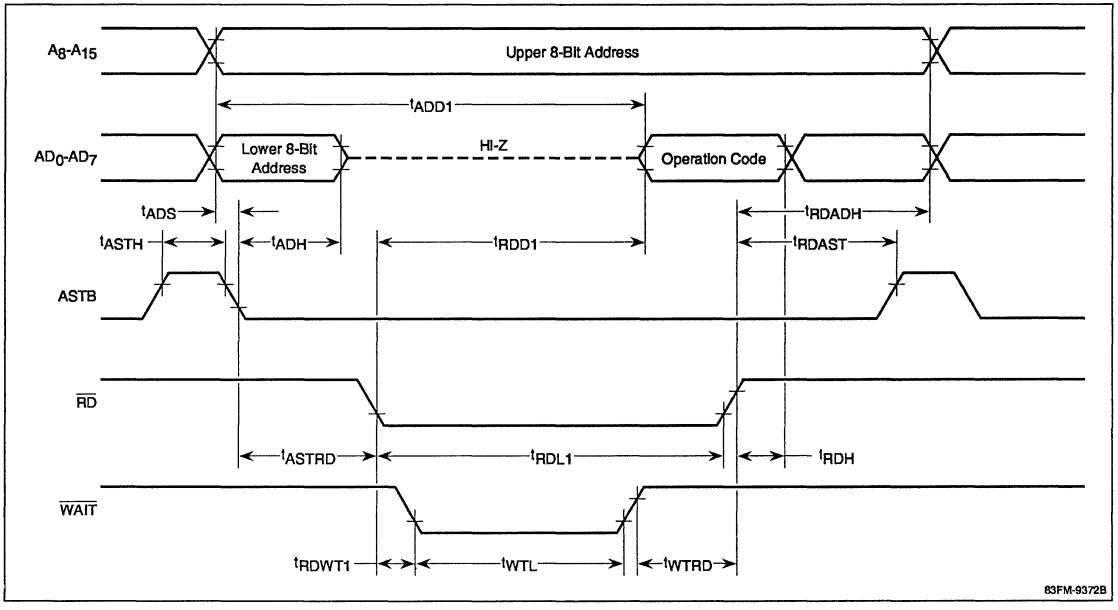


Figure 34. Read Operation; External Fetch (Wait Insertion)



3d

Figure 35. Read/Write Operation; External Data Access (No Wait)

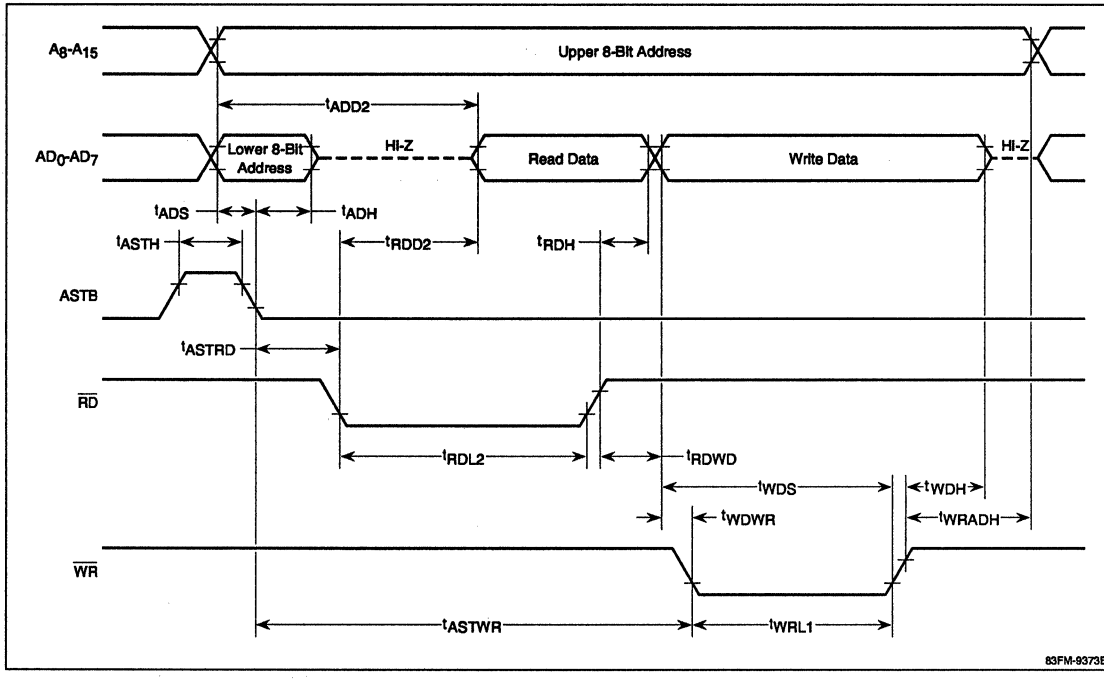
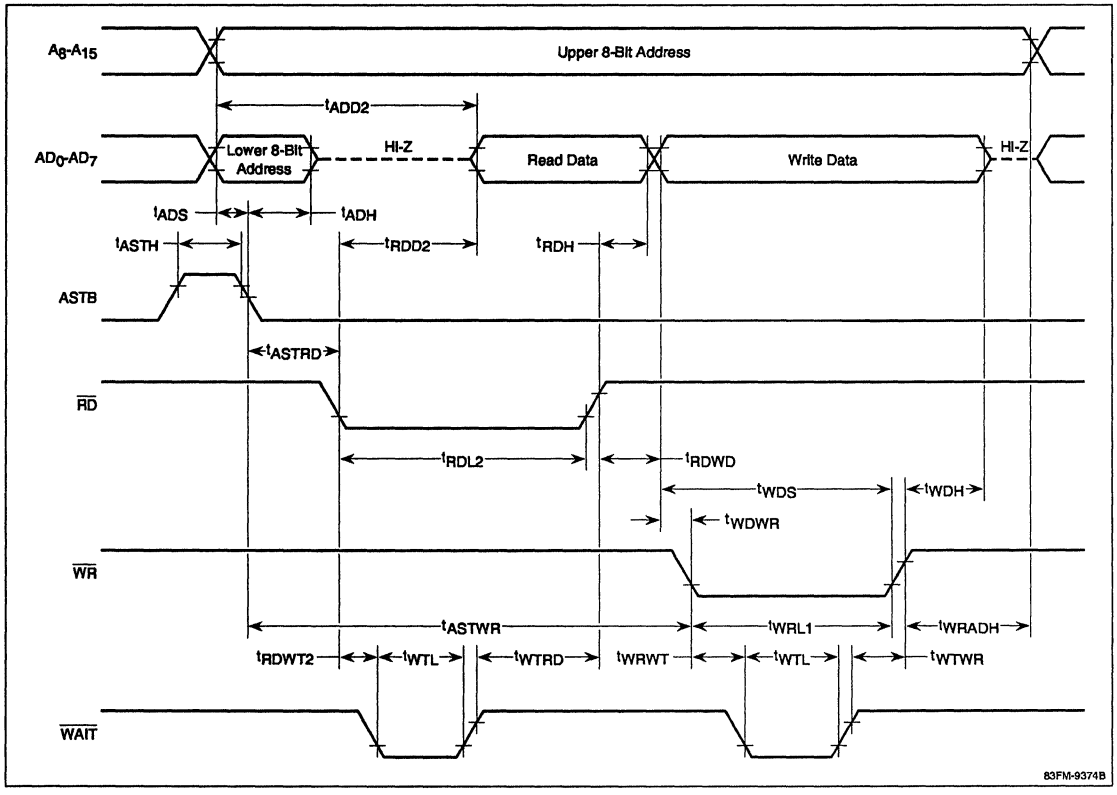


Figure 36. Read/Write Operation; External Data Access (Wait Insertion)



Serial Interface, 3-Wire, I/O Mode; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 37

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY1}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH1}, t_{KL1}	t_{KCY1}/2-50			ns	V_{DD} = 4.5 to 6.0 V
		t_{KCY1}/2-150			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK1}	100			ns	
SI hold time from $\overline{SCK}$ ↑	t_{KSI1}	400			ns	
SO output delay time from $\overline{SCK}$ ↓	t_{KSO1}			300	ns	V_{DD} = 4.5 to 6.0 V; C = 100 pF (See note)
				1000	ns	C = 100 pF (See note)

Note: C is the load capacitance of the SO output line.

Serial Interface, 3-Wire, I/O Mode; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 37

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY2}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH2}, t_{KL2}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK2}	100			ns	
SI hold time from $\overline{SCK}$ ↑	t_{KSI2}	400			ns	
SO output delay time from $\overline{SCK}$ ↓	t_{KSO2}			300	ns	V_{DD} = 4.5 to 6.0 V; C = 100 pF (See note)
				1000	ns	C = 100 pF (See note)

Note: C is the load capacitance of the SO output line.

Serial Interface, SBI Mode; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 38

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY3}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH3}, t_{KL3}	$t_{KCY3}/2$ - 50			ns	V_{DD} = 4.5 to 6.0 V
		$t_{KCY3}/2$ - 150			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK3}	100			ns	V_{DD} = 4.5 to 6.0 V
		300			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI3}	$t_{KCY3}/2$			ns	
SB0,SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO3}	0		250	ns	V_{DD} = 4.5 to 6.0 V; R = 1 kΩ, C = 100 pF (See note)
		0		1000	ns	R = 1 kΩ, C = 100 pF (See note)
SB0, SB1 ↓ from $\overline{SCK}$ ↑	t_{KSB}	t_{KCY3}			ns	
$\overline{SCK}$ ↓ from SB0, SB1 ↓	t_{SBK}	t_{KCY3}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY3}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY3}			ns	

Note: R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

Serial Interface, SBI Mode; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 38

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY4}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH4}, t_{KL4}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK4}	100			ns	V_{DD} = 4.5 to 6.0 V
		300			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI4}	$t_{KCY4}/2$			ns	
SB0, SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO4}	0		300	ns	V_{DD} = 4.5 to 6.0 V; R = 1 kΩ, C = 100 pF. (See note)
		0		1000	ns	R = 1 kΩ, C = 100 pF
SB0, SB1 ↓ from $\overline{SCK}$ ↑	t_{KSB}	t_{KCY4}			ns	
$\overline{SCK}$ ↓ from SB0, SB1 ↓	t_{SBK}	t_{KCY4}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY4}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY4}			ns	

Note: R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

3d

Serial Interface, 2-Wire, I/O Mode; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 39

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY5}	1600			ns	V_{DD} = 4.5 to 6.0 V
		3800			ns	
$\overline{SCK}$ high-level width	t_{KH5}	t_{KCY5} /2 − 50			ns	
$\overline{SCK}$ low-level width	t_{KL5}	t_{KCY5} /2 − 50			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK5}	300			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI5}	600			ns	
SB0, SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO5}	0		250	ns	V_{DD} = 4.5 to 6.0 V R = 1 kΩ, C = 100 pF (See note)
		0		1000	ns	R = 1 kΩ, C = 100 pF (See note)

Note: R and C are load resistance and load capacitance of the $\overline{SCK0}$, SB0, and SB1 output lines.

Serial Interface, 2-Wire, I/O Mode; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 39

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY6}	1600			ns	V_{DD} = 4.5 to 6.0 V
		3800			ns	
$\overline{SCK}$ high-level width	t_{KH6}	650			ns	
$\overline{SCK}$ low-level width	t_{KL6}	800			ns	
SB0, SB1 setup time to $\overline{SCK}$ ↑	t_{SIK6}	100			ns	
SB0, SB1 hold time from $\overline{SCK}$ ↑	t_{KSI6}	t_{KCY6}/2			ns	
SB0, SB1 output delay time from $\overline{SCK}$ ↓	t_{KSO6}	0		300	ns	V_{DD} = 4.5 to 6.0 V R = 1kΩ, C = 100 pF (See note)
		0		1000	ns	

Note: R and C are load resistance and load capacitance of the $\overline{SCK0}$, SB0, and SB1 output lines.

Serial Interface, I²C Bus

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 40

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
SCL input clock frequency	f_{SCL}	0		100	kHz	
Bus release time before start of transfer	t_{BUF}	4.7			μs	
Start condition hold time	t_{HDSTA}	4.0			μs	
SCL low-level time	t_{LOW}	4.7			μs	
SCL high-level time	t_{HIGH}	4.0			μs	
Start condition setup time	t_{SUSTA}	4.7			μs	
Data hold time	t_{HDDAT}	0			μs	SCL fall time: data retention
Data setup time	t_{SUDAT}	250			ns	
SDA, SDA0, SDA1, SCL signal rise time	t_R			1	μs	
SDA, SDA0, SDA1, SCL signal fall time	t_F			300	ns	
Stop condition setup time	t_{SUSTO}	4.7			μs	

Serial Interface, 3-Wire, I/O Mode with Automatic Transmit/Receive Function; Internal $\overline{SCK}$ Output

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 41

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY7}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH7}, t_{KL7}	$t_{KCY7}/2 - 50$			ns	V_{DD} = 4.5 to 6.0 V
		$t_{KCY7}/2 - 150$			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK7}	100			ns	
SI hold time from $\overline{SCK}$ ↑	t_{KSI7}	400			ns	
$\overline{SCK}$ ↓ to SO output delay time	t_{KSO7}			300	ns	V_{DD} = 4.5 to 6.0 V; C = 100 pF (See note)
				1000	ns	C = 100 pF (See note)
STB ↑ from $\overline{SCK}$ ↑	t_{SBD}	400		t_{KCY7}	ns	
Strobe signal high-level width	t_{SBW}	t_{KCY7} −30		t_{KCY7} + 30	ns	
Busy signal set-up time (to busy signal detection timing)	t_{BYS}	100			ns	
Busy signal hold time (from busy signal detection timing)	t_{BYH}	100			ns	
$\overline{SCK}$ ↓ from busy inactive	t_{SPS}			$2t_{KCY7}$	ns	

Note: C is the load capacitance for the SO output line.

Serial Interface, 3-Wire, I/O Mode with Automatic Transmit/Receive Function; External $\overline{SCK}$ Input

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V; refer to figure 41

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY8}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{SCK}$ high- and low-level width	t_{KH8}, t_{KL8}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK8}	100			ns	
SI hold time from $\overline{SCK}$ ↑	t_{KSI8}	400			ns	
$\overline{SCK}$ ↓ to SO output delay time	t_{KSO8}			300	ns	V_{DD} = 4.5 to 6.0 V; C = 100 pF (See note)
				1000	ns	C = 100 pF (See note)
$\overline{SCK}$ ↓ (after STB) from $\overline{SCK}$ ↑	t_{SPS1}	$2t_{KCY8}$			ns	

Note: C is the load capacitance for the SO output line.

3d

Figure 37. Serial Interface Timing; 3-Wire Serial I/O Mode

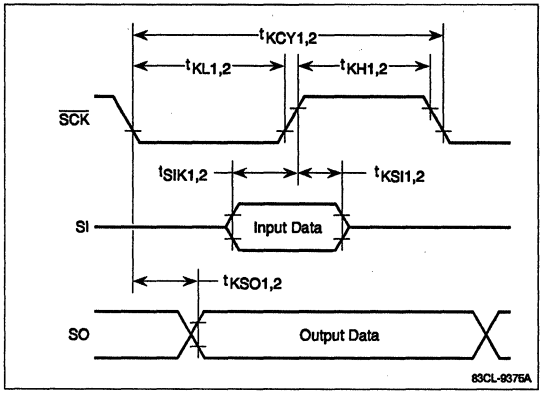


Figure 38. Serial Interface Timing; SBI Mode

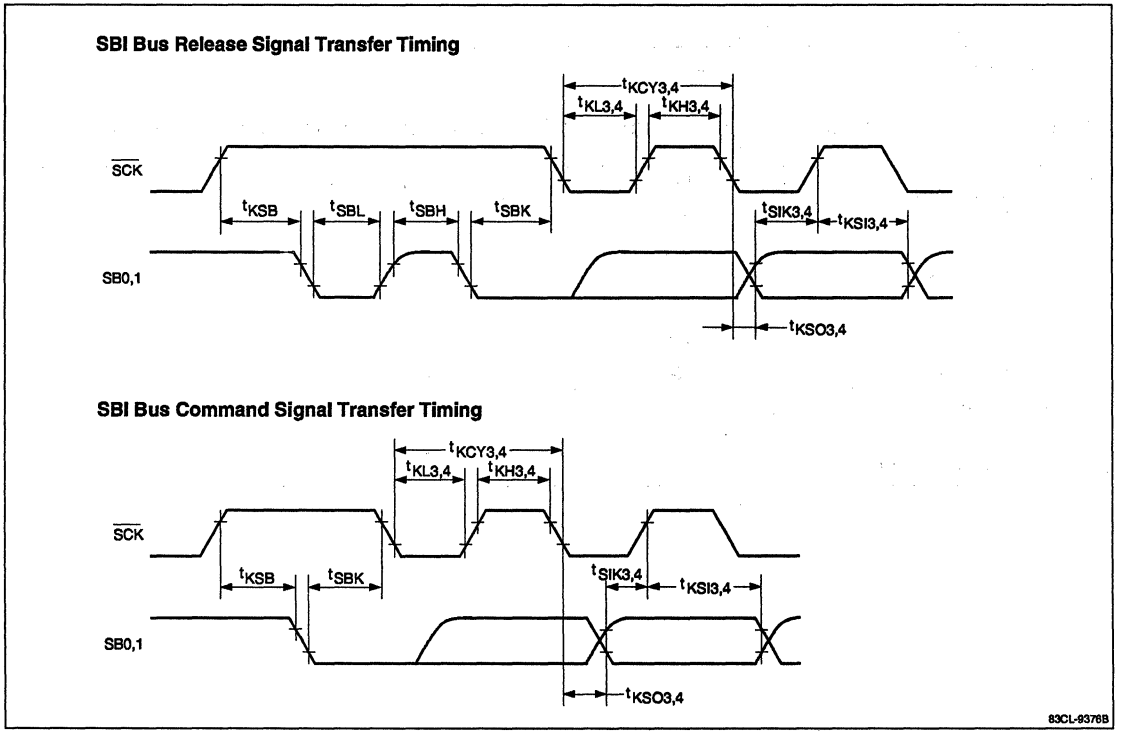


Figure 39. Serial Interface Timing; 2-Wire Serial I/O Mode

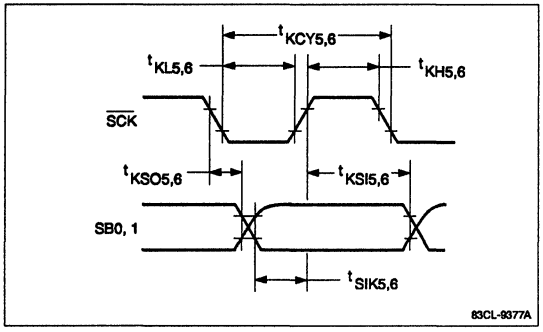


Figure 40. Serial Transfer Timing; I²C Bus Mode

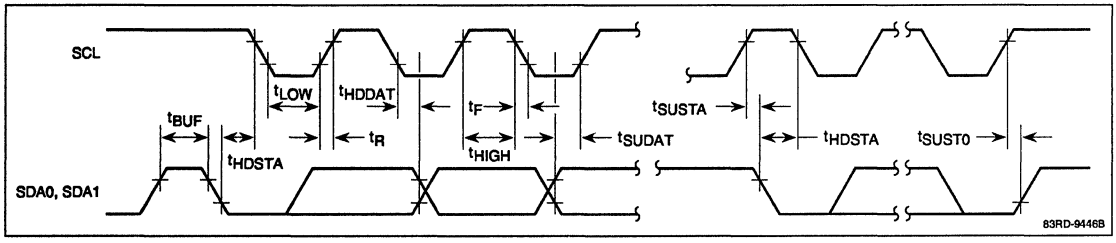
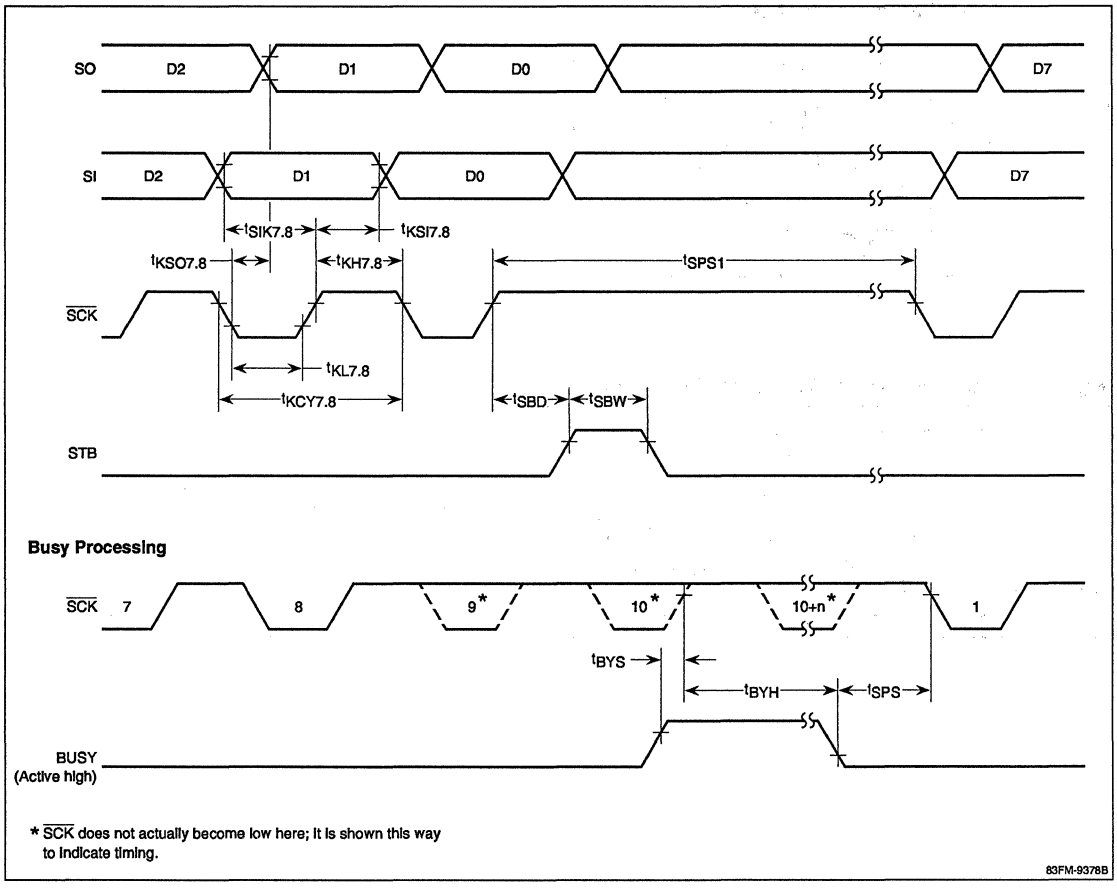


Figure 41. Serial Interface Timing; 3-Wire Serial I/O Mode with Automatic Transmit/Receive Function



A/D Converter

T_A = −40 to +85°C; AV_{DD} = V_{DD} = 2.7 to 6.0 V, AV_{SS} = V_{SS} = 0V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		8	8	8	bit	
Absolute accuracy (See note)				±1.5	LSB	
Conversion time	t_{CONV}	160/f_X			μs	f_X = 4.19 to 8.38 MHz
		80/f_X			μs	f_X = 1 to 4.19 MHz
Sampling time	t_{SAMP}	24/f_X			μs	
Analog input voltage	V_{IAN}	AV_{SS}		AV_{REF}	V	
Reference voltage	AV_{REF}	2.7		AV_{DD}	V	
AV_{REF} current	I_{REF}		0.5	1.5	mA	

Note: Absolute accuracy does not include the quantization error (±1/2 LSB).

Data Memory STOP Mode; Low-Voltage Data Retention

T_A = −40 to +85°C; refer to figure 42

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention supply voltage	V_{DDDR}	2.0		6.0	V	
Data retention power supply current	I_{DDDR}		0.1	10	μA	V_{DDDR} = 2.0 V; subsystem clock stop and feedback resistor disconnected
Release signal set time	t_{SREL}	0			μs	
Oscillation stabilization wait time	t_{WAIT}		$2^{18}/f_X$		ms	Release by $\overline{RESET}$
			(Note 1)		ms	Release by interrupt

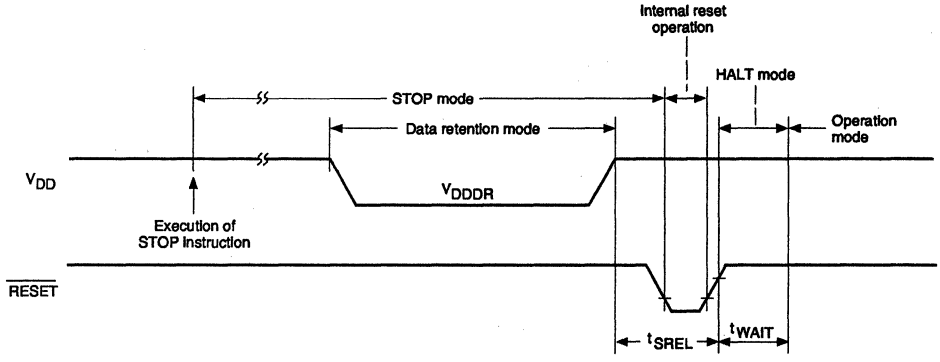
Note:

(1) $2^{13}/f_X$, $2^{15}/f_X$, $2^{16}/f_X$, $2^{17}/f_X$ or $2^{18}/f_X$ can be chosen by using bits 0 to 2 of the oscillation stabilization time select (OSTS) register.

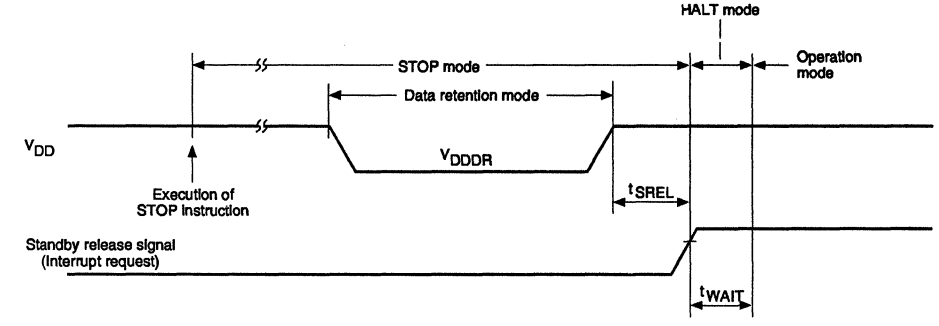
3d

Figure 42. Data Retention Timing

A. STOP mode is released by $\overline{\text{RESET}}$ input



B. STOP mode is released by interrupt signal



83RD-6456B

PROM PROGRAMMING

The PROM in the μPD78P014Y is an OTP or UV EPROM. The 32,768 x 8-bit PROM has the programming characteristics of an NEC μPD27C256A. Table 5 shows the functions of the μPD78P014Y pins in both normal operating and PROM programming mode.

Table 5. Pin Functions During PROM Programming

Function	Normal Operating Mode	Programming Mode
Address input	$P4_0$ - $P4_7$, $P5_0$, $P0_0$, $P5_2$ - $P5_6$	A_0 - A_{14}
Data input	$P3_0$ - $P3_7$	D_0 - D_7
Chip enable/ program pulse	$P6_5/\overline{WR}$	$\overline{CE}$
Output enable	$P6_4/\overline{RD}$	$\overline{OE}$
Program voltage	IC	V_{PP}
Mode voltage	$\overline{RESET}$	Logical 0

PROM Programming Modes

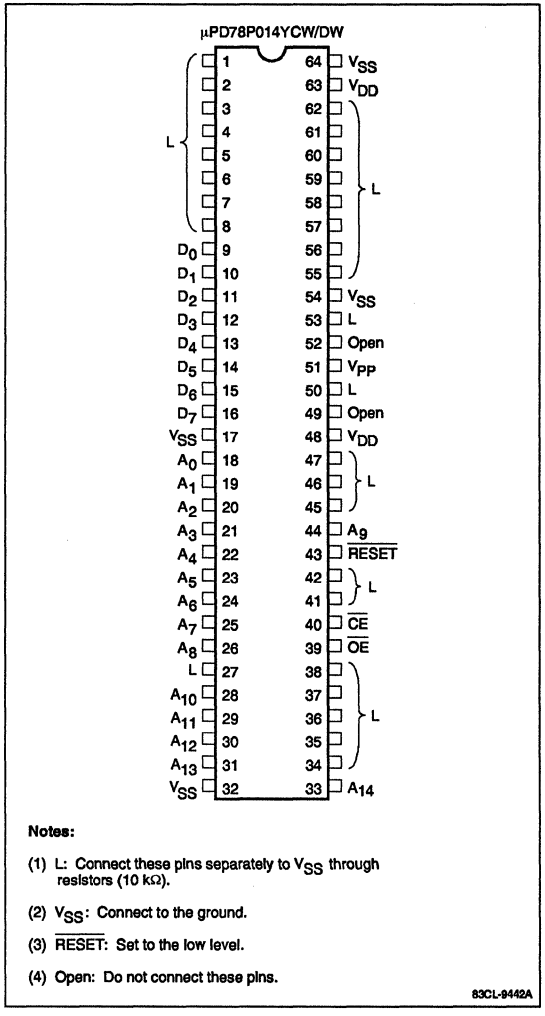
When the $\overline{RESET}$ pin is set low and V_{PP} is set to +5 V or +12.5 V, the μPD78P014Y enters the programming mode of operation. Operation in this mode is determined by the setting of the $\overline{CE}$, $\overline{OE}$, V_{PP} and V_{DD} pins as indicated in Table 6.

3d

Table 6. Programming Operation Modes

Mode	$\overline{RESET}$	V_{PP}	V_{DD}	$\overline{CE}$	$\overline{OE}$	D_0 to D_7
Program write	L	+12.5 V	+6 V	L	H	Data input
Program verify	L	+12.5 V	+6 V	H	L	Data output
Program inhibit	L	+12.5 V	+6 V	H	H	High impedance
Read	L	+5 V	+5 V	L	L	Data output
Output disble	L	+5 V	+5 V	L	H	High impedance
Standby	L	+5 V	+5 V	H	L/H	High impedance

Figure 43. PROM Programming Mode Pin Function; 64-Pin plastic or Ceramic Shrink DIP



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors (10 kΩ).

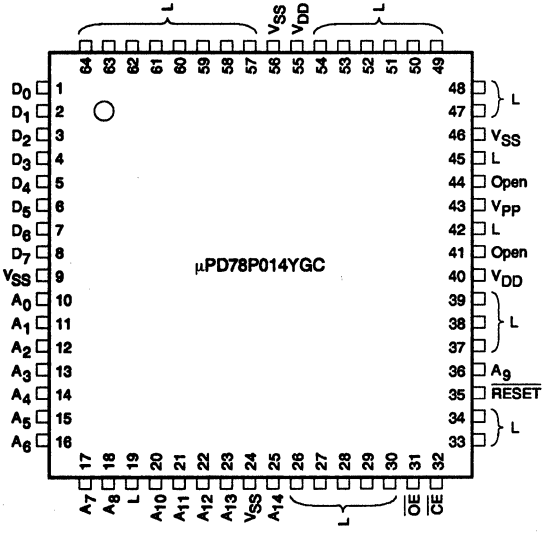
(2) V_{SS}: Connect to the ground.

(3) $\overline{RESET}$: Set to the low level.

(4) Open: Do not connect these pins.

83CL-9442A

Figure 44. PROM Programming Mode Pin Functions; 64-Pin Plastic QFP



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors (10 kΩ).

(2) V_{SS}: Connect to the ground.

(3) $\overline{RESET}$: Set to the low level.

(4) Open: Do not connect these pins.

83CL-9443B

3d

PROM Write Procedure

Data can be written to the PROM by using the following procedure.

(1) Set the pins not used for programming as indicated in figures 43 and 44. Set the RESET pin low and the V_{DD} and V_{PP} pins to +5 V. The $\overline{CE}$ and $\overline{OE}$ pins should be high.

(2) Supply +6.0 V to the V_{DD} pin and +12.5 V to the V_{PP} pin.

(3) Provide the initial address to the A_0 - A_{14} pins.

(4) Provide the write data.

(5) Provide a 1-ms program pulse (active low) to the $\overline{CE}$ pin.

(6) Use the verify mode (pulse $\overline{OE}$ low) to test the data. If data is written correctly, proceed to step 8; if data is not written correctly, repeat steps 4 to 6 up to 25 times. If data is still incorrect, go to step 7.

(7) Classify the PROM as defective and cease write operation.

(8) Perform one additional write with a program pulse width (in ms) equal to three times the number of writes performed in step 5.

(9) Increment the address.

(10) Repeat steps 4-9 until the last adress is programmed.

PROM Read Procedure

The contents of the PROM can be read out of the external data bus (D_0 - D_7) by using the following procedure.

(1) Set the pins not used for programming as indicated in Figures 43 and 44. Set the RESET pin low and the V_{PP} pin and V_{DD} pin to +5 V. The $\overline{CE}$ and $\overline{OE}$ pins should be high.

(2) Input the address of the data to be read to the A_0 - A_{14} pins.

(3) Put an active-low pulse on $\overline{CE}$ and $\overline{OE}$ pins.

(4) Data is output to pins D_0 - D_7.

Program Erasure

The UV EPROM can be erased (all locations FFH) by exposing the window to light having a wavelength shorter than 400 nm, including ultraviolet, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15 W s/cm^2 (ultraviolet ray intensity x exposure time) is required to completely erase the written data. Erasure by an ultraviolet lamp rated at 12,000 μW/cm^2 takes approximately 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

DC Programming Characteristics

T_A = 25 ±5°C, V_{SS} = 0 V.

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH}	V_{IH}	0.7 V_{DDP}		V_{DDP}	V	
Low-level input voltage	V_{IL}	V_{IL}	0		0.3 V_{DDP}	V	
Input leakage current	I_{LIP}	I_{LI}			10	μA	$0 \le V_I \le V_{DDP}$
High-level output voltage	V_{OH1}	V_{OH1}	2.4			V	I_{OH} = −400 μA
	V_{OH2}	V_{OH2}	V_{DD}−0.7			V	I_{OH} = −100 μA
Low-level output voltage	V_{OL}	V_{OL}			0.45	V	I_{OL} = 2.1 mA
Output leakage current	I_{LO}				10	μA	$0 \le V_O \le V_{DDP}$, $\overline{OE}$ = V_{IH}
V_{DDP} power voltage	V_{DDP}	V_{CC}	5.75	6.0	6.25	V	Program memory write mode
			4.5	5.0	5.5	V	Program memory read mode
V_{PP} power voltage	V_{PP}	V_{PP}	12.2	12.5	12.8	V	Program memory write mode
				V_{PP} = V_{DDP}		V	Program memory read mode
V_{DDP} power current	I_{DD}	I_{CC}		5	30	mA	Program memory write mode
				5	30	mA	Program memory read mode $\overline{CE}$ = V_{IL}, V_I = V_{IH}
V_{PP} power current	I_{PP}	I_{PP}		5	30	mA	Program memory write mode $\overline{CE}$ = V_{IL}, $\overline{OE}$ = V_{IH}
				1	100	μA	Program memory read mode

*Corresponding symbols of the μPD27C256A.

AC Programming Characteristics (Write Mode)

T_A = 25 ±5°C, V_{SS} = 0 V, V_{DD} = 6 ±0.25 V, V_{PP} = 12.5 ±0.3 V.

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Conditions
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}	2			μs	
Data input to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}	2			μs	
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}	2			μs	
Address hold time from $\overline{CE}$ ↑	t_{HCA}	t_{AH}	2			μs	
Input data hold time from $\overline{CE}$ ↑	t_{HCID}	t_{DH}	2			μs	
Output data hold time from $\overline{OE}$ ↑	t_{HOOD}	t_{DF}	0		130	ns	
V_{PP} setup time to $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}	1			ms	
V_{DDP} setup time to $\overline{CE}$ ↓	t_{SVDC}	t_{VCS}	1			ms	
Initial program pulse width	t_{WL1}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{WL2}	t_{OPW}	2.85		78.75	ms	
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			150	ns	

* Corresponding symbols of the μPD27C256A.

3d

AC Programming Characteristics (Read Mode)

T_A = 25 ±5°C, V_{SS} = 0 V, V_{DD} = 5 ±0.5 V, V_{PP} = V_{DD}.

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
Address to data output time	t_{DAOD}	t_{ACC}			200	ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}
$\overline{CE}$ ↓ to data output time	t_{DCOD}	t_{CE}			200	ns	$\overline{OE}$ = V_{IL}
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			75	ns	$\overline{CE}$ = V_{IL}
Data hold time from $\overline{OE}$ ↑	t_{HCOD}	t_{DF}	0		60	ns	$\overline{CE}$ = V_{IL}
Data hold time from address	t_{HAOD}	t_{OH}	0			ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}

* Corresponding symbols of the μPD27C256A.

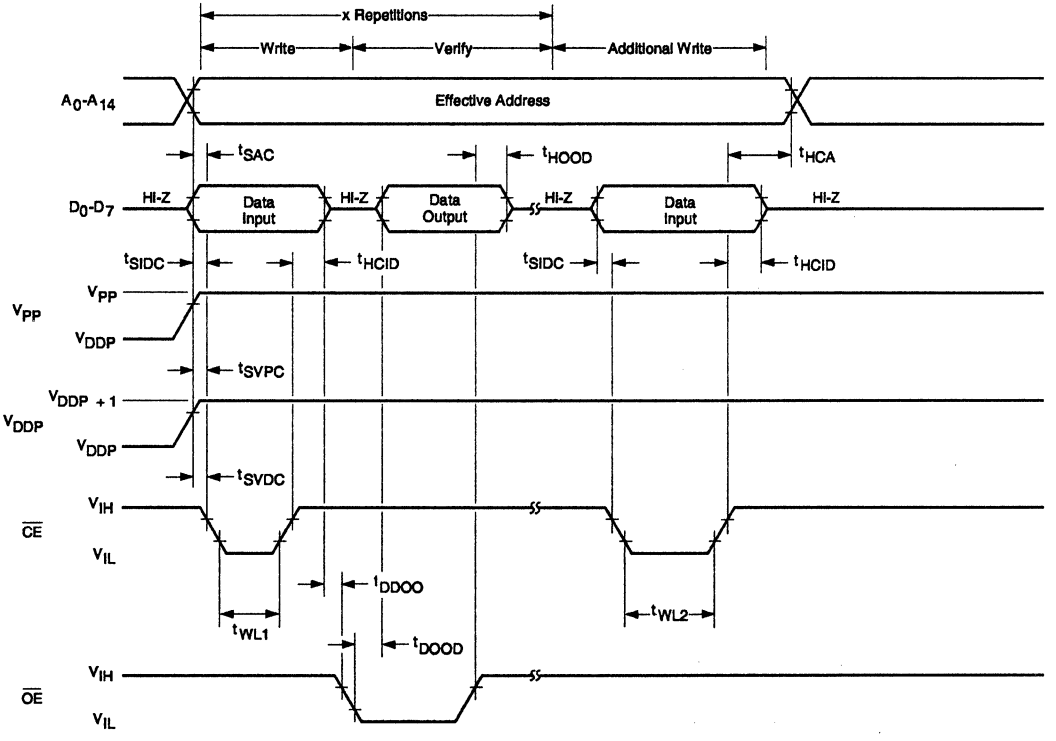
AC Programming Characteristics (PROM Mode)

T_A = 25 ±5°C, V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Condition
PROM mode setup time	t_{SMA}			10	μs	

PROM Timing Diagrams

PROM Write/Verify Mode



Notes:

(1) V_{DDP} must be applied before applying V_{PP}. It should be removed before removing V_{PP}.

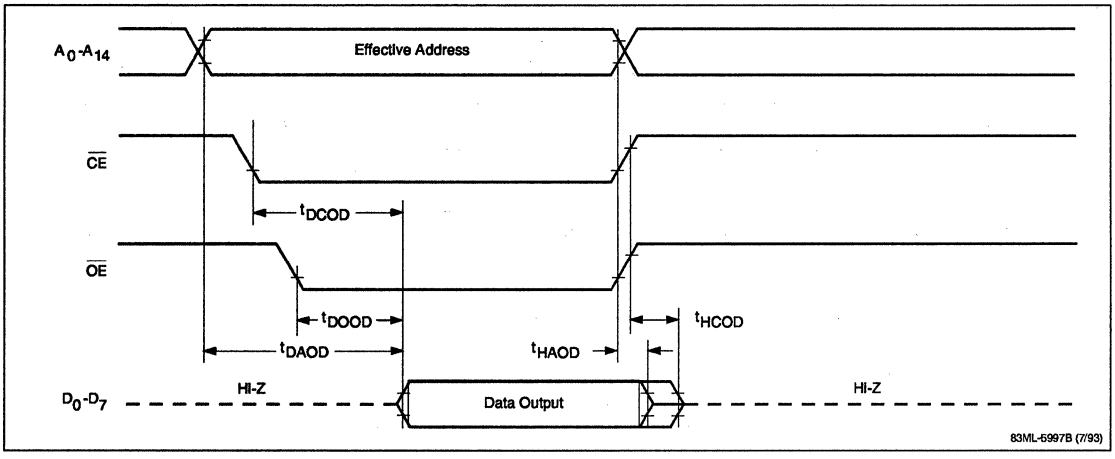
(2) V_{PP} must not exceed +13V, including overshoot.

83YL-9382B (8/93)

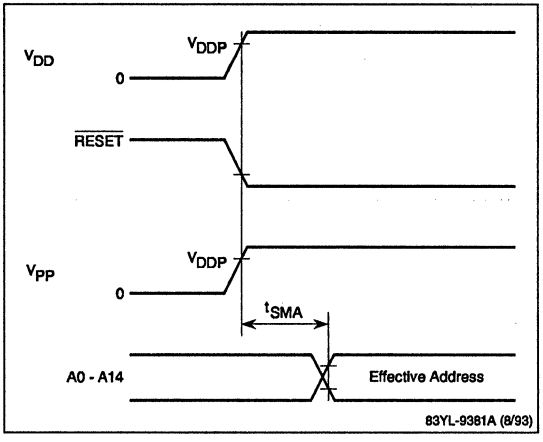
3d

PROM Timing Diagrams (cont)

PROM Read Mode



PROM Mode Setting



NEC
NEC Electronics Inc.

μPD78044 Family (μPD78042/043/044/P044) 8-Bit, K-Series Microcontrollers With FIP Controller/Driver and A/D Converter

Preliminary

September 1993

Description

The μPD78042, μPD78043, μPD78044, and μPD78P044 are members of the K-Series® of microcontrollers featuring a FIP® (VF) controller/driver, A/D converter, 8-bit hardware multiply and divide instructions, bit manipulation instructions, four banks of main registers, an advanced interrupt handling facility, and a powerful set of memory-mapped on-chip peripherals.

Timing is generated by two oscillators. The main oscillator normally drives the CPU and most peripherals. The 32.768-kHz subsystem oscillator provides time keeping when the main oscillator is turned off. Since CMOS power dissipation is proportional to clock rate, the μPD78044 family provides a software selectable instruction cycle time from 0.48 μs to 122 μs. The STOP and HALT modes turn off parts of the microcontroller for additional savings. The data retention mode keeps RAM contents valid down to 2.0 V.

These devices are ideally suited for applications in portable battery-powered equipment, office automation, communications, automotive and consumer electronics, home appliances, and PC peripherals.

K-Series and FIP are registered trademarks of NEC Electronics Inc.

Features

- FIP controller/driver
  - Up to 34 lines of direct-drive, high-voltage output
  - Eight software-controller intensity levels
  - 48 bytes of display RAM
  - Refresh of display without CPU intervention
  - Key scan capability
- Eight-channel, 8-bit A/D converter
- Two-channel serial communications interface
  - 8-bit clock-synchronous interface 0
    - Full-duplex, three-wire mode
    - NEC serial bus interface (SBI) mode
    - Half-duplex, two-wire mode
  - 8-bit clock-synchronous interface 1
    - Full-duplex, three-wire mode
    - Automatic transfer, full-duplex three-wire mode
- Timers; six channels
  - Watchdog timer
  - 16-bit timer/event counter
  - Two 8-bit timer/event counters usable as one 16-bit timer/event counter
  - 6-bit up/down counter for external events
  - Watch (clock) timer
- 68 I/O and bidirectional I/O lines, including high voltage lines for FIP drive
  - Two CMOS input-only lines
  - 27 CMOS bidirectional I/O lines
  - Five n-channel, open-drain I/O lines at 15 V maximum
  - 16 p-channel, open-drain I/O lines at 35 V maximum
  - 18 p-channel, open-drain output lines at 35 V maximum
  - Software or mask selectable pullup or pulldown resistors available on many port lines
- Powerful instruction set
  - 8-bit unsigned multiply and divide
  - 16-bit arithmetic and data transfer instructions
  - 1-bit and 8-bit logic instructions
- Minimum instruction execution times
  - 0.48/0.95/1.91/3.81/7.63 μs using 4.19-MHz main system clock
  - 122 μs using 32.768-kHz subsystem clock
- Memory-mapped on-chip peripherals
  - Special function registers
- Programmable priority, vectored-interrupt controller (two levels)
- Programmable buzzer and clock outputs
- Power-saving and battery back up
  - Variable CPU clock rate
  - STOP mode
  - HALT mode
  - 2-V data retention mode
- CMOS operation; V_{DD} from 2.7 to 6.0 V

Internal High-Capacity ROM and RAM

	78042	78043	78044	78P044
ROM	16K bytes	24K bytes	32K bytes	—
PROM	—	—	—	32K bytes
High-speed RAM	512 bytes	1024 bytes	1024 bytes	1024 bytes
Serial buffer RAM	64 bytes	64 bytes	64 bytes	64 bytes
FIP display RAM	48 bytes	48 bytes	48 bytes	48 bytes

3e

Ordering Information

Part Number	ROM	Package (Package Dwg)
μPD78042GF-xxx-3B9	16K mask ROM	80-pin plastic QFP (P80GF-80-3B9-1)
μPD78043GF-xxx-3B9	24K mask ROM	
μPD78044GF-xxx-3B9	32K mask ROM	
μPD78P044GF-3B9	32K OTP ROM	
μPD78P044KL-S	32K UV EPROM	80-pin ceramic LCC with window (X80KW-80A)

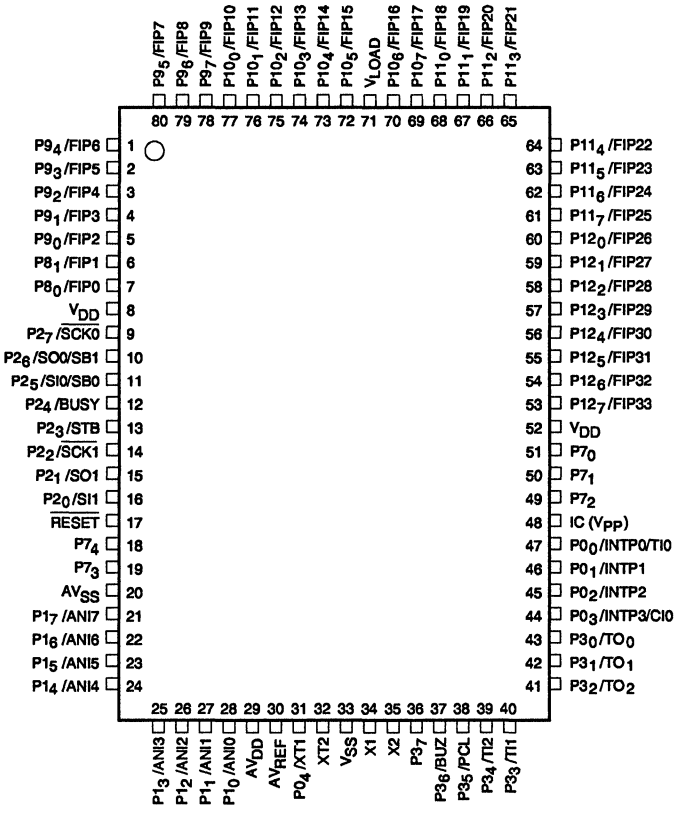
Notes:

(1) xxx indicates ROM code suffix

(2) All devices listed are Standard Quality Grade.

Pin Configurations

80-Pin Plastic QFP or Ceramic LCC With Window



Notes:

(1) Connect IC (Internally connected) pin (V_{PP} on μPD78P044) to V_{SS}.

(2) AV_{DD} should be connected to V_{DD}.

(3) AV_{SS} should be connected to V_{SS}.

83YL-9518B (9/93)

3e

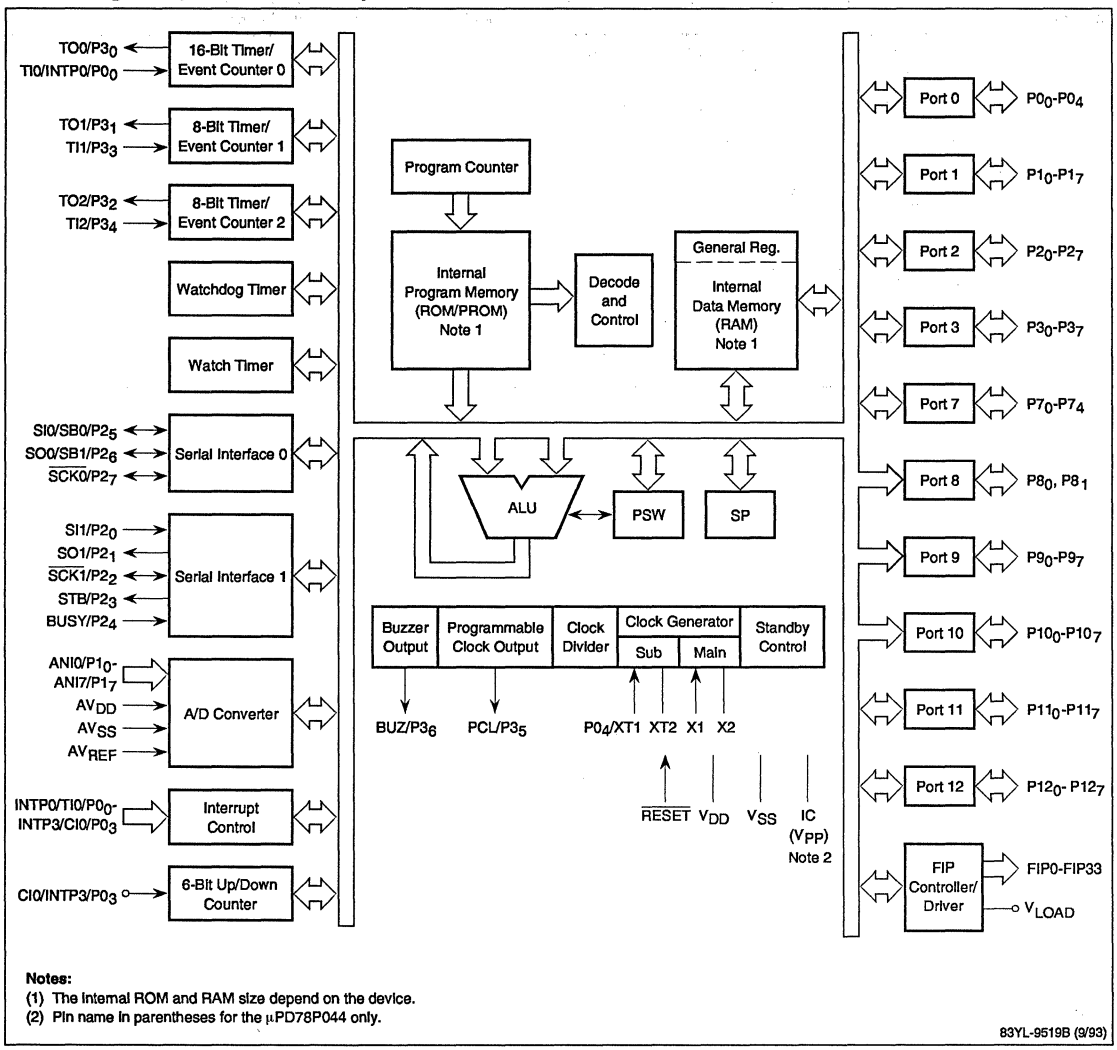
Pin Functions; Normal Operating Mode

Symbol	First Function	Symbol	Alternate Functions
$P0_0$	Port 0; 5-bit, bit-selectable I/O. (Bits 0 and 4 are input only)	INTP0 TI0	External maskable interrupt External count clock input to timer 0
$P0_1$		INTP1	External maskable interrupt
$P0_2$		INTP2	External maskable interrupt
$P0_3$		INTP3 CI0	External maskable interrupt Up/down counter clock input
$P0_4$		XT1	Crystal oscillator or external clock input for subsystem clock
$P1_0$ - $P1_7$	Port 1; 8-bit, bit-selectable I/O port.	ANI0 - ANI7	Analog input to A/D converter
$P2_0$	Port 2; 8-bit, bit-selectable I/O port.	SI1	Serial data input; three-wire serial I/O mode
$P2_1$		SO1	Serial data output; three-wire serial I/O mode
$P2_2$		$\overline{SCK1}$	Serial clock I/O for serial interface 1
$P2_3$		STB	Serial interface; automatic transmit/receive strobe output
$P2_4$		BUSY	Serial interface; automatic transmit/receive busy input
$P2_5$		SI0 SB0	Serial data input; three-wire serial I/O mode Two- or three-wire serial I/O mode
$P2_6$		SO0 SB1	Serial data output; three-wire serial I/O mode Two- or three-wire serial I/O mode
$P2_7$		$\overline{SCK0}$	Serial clock I/O for serial interface 0
$P3_0$	Port 3; 8-bit, bit-selectable I/O port.	TO0	Timer output from timer 0
$P3_1$		TO1	Timer output from timer 1
$P3_2$		TO2	Timer output from timer 2
$P3_3$		TI1	External count clock input to timer 1
$P3_4$		TI2	External count clock input to timer 2
$P3_5$		PCL	Programmable clock output
$P3_6$		BUZ	Programmable buzzer output
$P3_7$		—	
$P7_0$ - $P7_4$	Port 7; 5-bit,bit selectable I/O port. N-channel, open drain.	—	
$P8_0$ - $P8_1$	Port 8; 2-bit, output port. P-channel, open drain.	FIP0, FIP1	FIP digit select outputs
$P9_0$ - $P9_7$	Port 9; 8-bit, output port. P-channel, open drain.	FIP2 - FIP9	FIP digit select outputs
$P10_0$ - $P10_5$	Port 10; 8-bit, output port. P-channel, open drain.	FIP10 - FIP15	FIP digit select or segment outputs
$P10_6$ - $P10_7$		FIP16 - FIP17	FIP segment outputs
$P11_0$ - $P11_7$	Port 11; 8-bit, bit selectable I/O port. P-channel open drain.	FIP18 - FIP25	FIP digit select outputs
$P12_0$ - $P12_7$	Port 12; 8-bit, bit selectable I/O port. P-channel open drain.	FIP26 - FIP33	FIP digit select outputs
V_{LOAD}	FIP controller/driver; pulldown resistor connection.		
$\overline{RESET}$	External system reset input		
X1	Crystal/ceramic resonator connection or external clock input for main system clock		

Pin Functions; Normal Operating Mode (cont)

Symbol	First Function	Symbol	Alternate Functions
X2	Crystal/ceramic resonator connection or inverse of external clock for main system clock		
XT2	Crystal oscillator or left open when not using the subsystem clock		
AV_{REF}	A/D converter reference voltage		
AV_{DD}	A/D converter power supply input		
AV_{SS}	A/D converter ground		
V_{DD}	Power supply input		
V_{PP}	μPD78P044 PROM programming power supply input		
V_{SS}	Power supply ground		
IC	Internal connection		

Block Diagram, μPD78044 Family



Notes:
(1) The internal ROM and RAM size depend on the device.
(2) Pin name in parentheses for the μPD78P044 only.

83YL-9519B (9/93)

FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processing unit (CPU) of the μPD78044 family features 8- and 16-bit arithmetic including an 8- by 8-bit unsigned multiply and a 16- by 8-bit unsigned divide (producing a 16-bit quotient and an 8-bit remainder). The multiply executes in 3.82 μs and the divide in 5.97 μs using the fastest clock cycle with a 4.19-MHz main system clock.

A CALLT vector table and a CALLF area decrease the number of bytes in the call instructions for commonly used subroutines. A one-byte call instruction can access up to 32 subroutines through their addresses contained in the CALLT vector table (40H to 7FH). A two-byte call instruction can access any routine beginning in a specific CALLF area (0800H to 0FFFH).

Internal System Clock Generator

The internal system clocks of the μPD78044 family are derived from the main system or subsystem clock oscillator. See figure 1. The clocks for the watch timer and programmable clock output are derived from the subsystem clock (f_{XT}) or main system clock (f_X). The clocks for all other peripheral hardware are derived from the main system clock.

The CPU clock (ϕ) can be supplied from the main system clock (f_X) or subsystem clock (f_{XT}). Using the processor clock control register (PCC), a CPU clock frequency equal to f_X, $f_X/2$, $f_X/4$, $f_X/8$, $f_X/16$ or the subsystem clock f_{XT} can be selected. The CPU clock selected should be based on the power supply voltage available and the desired power consumption. On power up, the CPU clock defaults to the lowest speed from the main system clock and can be changed while the microcomputer is running.

Since the shortest instruction takes two CPU clocks to execute, the fastest instruction execution time (t_{CY}) of 0.48 μs is achieved with a 4.19-MHz main system clock and a V_{DD} of 4.5 to 6.0 volts. The fastest instruction execution time available across the full voltage range of 2.7 to 6.0 volts is 0.96 μs with a 4.19-MHz main system clock. For the lowest power consumption, the CPU can be operated from the subsystem clock and the fastest instruction execution time is 122 μs at 32.768 kHz.

Memory Space

Program and data memory are mapped into the 64K-byte address space (0000H-FFFFH). See figure 2. The μPD78044 family is optimized for single-chip operation and does not permit external memory.

Internal Program Memory

All devices in the μPD78044 family have internal program memory. The μPD78042, μPD78043, and μPD78044 contain 16K, 24K, and 32K bytes of internal ROM, respectively. The μPD78P044 contains 32K bytes of UV EPROM or one-time programmable ROM. To allow the μPD78P044 to emulate the mask ROM devices, the amount of internal program memory available in the μPD78P044 can be selected using the memory size switching register (IMS).

3e

Internal RAM

Internal RAM comprises three types: high-speed, buffer, and FIP display. The μPD78042 has 624 bytes of internal RAM and the μPD78043/044/P044 have 1136 bytes.

High-speed RAM contains the general register banks and the stack. Unused portions of RAM and unused register bank locations are available for general storage. The μPD78042 has 512 bytes (FD00H-FEFFH) of high-speed RAM; the μPD78043/044/P044 has 1024 bytes (FB00H-FEFFH).

All devices contain 64 bytes (FA80H-FAFFH) of buffer RAM and 48 bytes (FA50H-FA7FH) of FIP display RAM. The buffer area is used for the automatic transfer mode of serial interface 1 or for general storage. The FIP display area is for display data; unused portions are available for general storage.

CPU Control Registers

Program Counter. The program counter is a 16-bit binary counter register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 0001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

Figure 1. Internal System Clock Generator

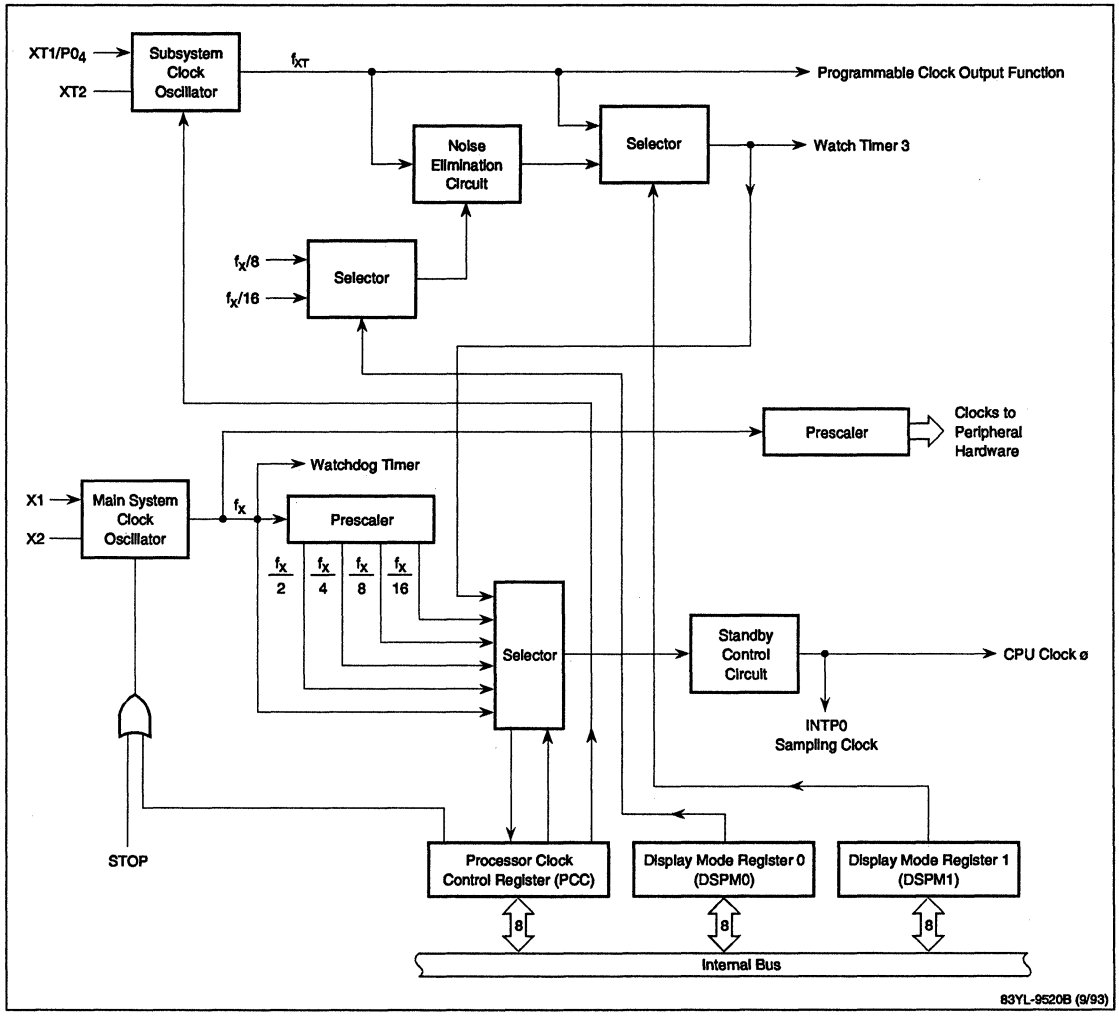
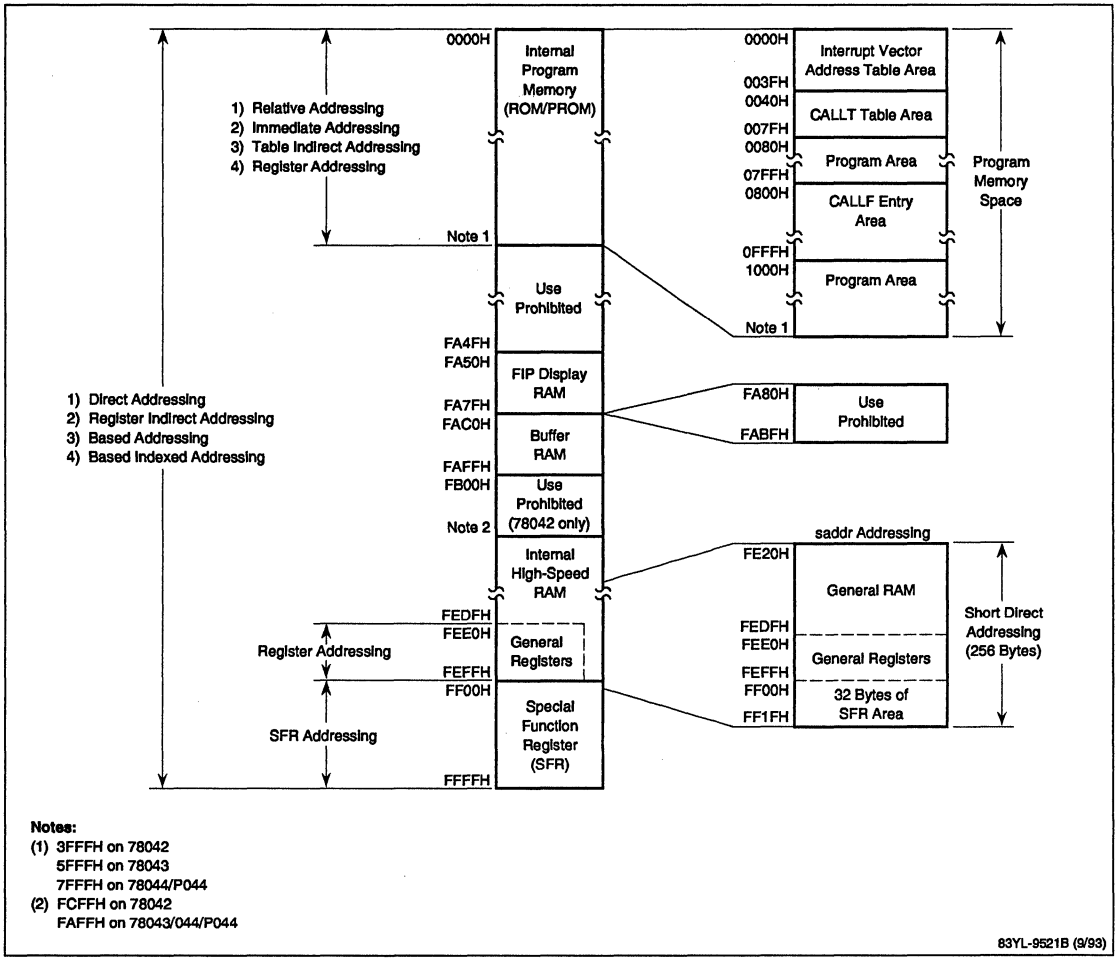


Figure 2. Memory Map



Notes:
(1) 3FFFH on 78042
5FFFH on 78043
7FFFH on 78044/P044
(2) FCFFH on 78042
FAFFH on 78043/044/P044

83YL-9521B (9/93)

3e

Program Status Word. The program status word (PSW) is an 8-bit register containing flags that are set or reset according to the results of an instruction execution. This register can be written to or read from 8 bits at a time. The individual flags can also be manipulated on bit-by-bit basis. The assignment of PSW bits follows:

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

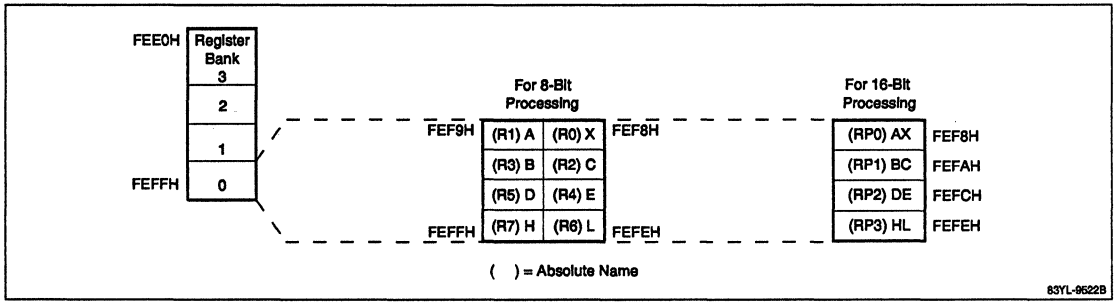
CY	Carry flag
ISP	Inservice (interrupt) priority flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt enable flag

General-Purpose Registers

The general-purpose registers (figure 3) are in four banks at addresses FEE0H to FEFFH in high-speed internal RAM. Each bank comprises eight 8-bit registers, which can be paired as four 16-bit registers. Bits RBS0 and RBS1 in the PSW, set under program control, identify the active register banks at any time.

Eight-bit registers have functional names A, X, B, C, D, E, H, L, and absolute names R0 thru R7; the four 16-bit registers have functional names AX, BC, DE, HL and absolute names RP0 thru RP3. Either the functional or absolute name is acceptable for the operand identifier "r" or "rp" in an instruction.

Figure 3. General-Purpose Registers



Addressing

The program memory addressing (ROM) modes provided are relative, immediate, table indirect, and register addressing. The operand addressing modes provided are implied, register, direct, short direct (saddr), special-function (SFR), register indirect, based, based indexed, and stack addressing.

The SFR addressing and saddr addressing modes use direct addressing and require only 1 byte in the instruction to address RAM. Normally a 65K byte address space requires 2 bytes to address it. One-byte addressing results in faster access times, since the instructions are shorter. SFR addressing addresses the entire 256-byte SFR address space from FF00H to FFFFH. Saddr addressing (see figure 2) addresses the 256-byte address space FE20H to FF1FH. FE20H to FEFFH are composed of 224 bytes of internal high speed RAM; FF00H to FF1FH contain the first 32 bytes in the special function register area.

One-byte addressing is accomplished by using the first byte of the instruction for the opcode (and one operand if register A or AX is used) and the second byte of the instruction as an address (offset) into the 256-byte area. If register A or AX is used, the instructions are 2 bytes long thereby providing fast access times. If immediate data is used, the instruction will be 3 or 4 bytes long depending upon whether the immediate data is a byte or a word. Many 16-bit SFRs are in the space FF00H to FF1FH. Using AX as an operand to these SFRs will provide fast access, since the instructions will only 2 bytes long.

Special Function Registers

The input/output ports, timers, capture and compare registers, and the mode and control registers for the peripherals and the CPU are collectively known as special function registers (table 1). They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by one-byte SFR addressing. FF00H to FF1H can also be accessed using saddr addressing. They are 8 or 16 bits, as required; many of the 8-bit registers are capable of single-bit access as well.

Table 1. Special Function Registers

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF00H	Port 0	P0	R/W	x	x	—	00H
FF01H	Port 1	P1	R/W	x	x	—	00H
FF02H	Port 2	P2	R/W	x	x	—	00H
FF03H	Port 3	P3	R/W	x	x	—	00H
FF07H	Port 7	P7	R/W	x	x	—	00H
FF08H	Port 8	P8	W	x	x	—	00H
FF09H	Port 9	P9	W	x	x	—	00H
FF0AH	Port 10	P10	W	x	x	—	00H
FF0BH	Port 11	P11	R/W	x	x	—	00H
FF0CH	Port 12	P12	R/W	x	x	—	00H
FF10H-FF11H	Compare register 00	CR00	R/W	—	—	x	Undefined
FF12H-FF13H	Capture register 01	CR01	R	—	—	x	Undefined
FF14H-FF15H	16-bit timer register	TM0	R	—	—	x	00H
FF16H	Compare register 10	CR10	R/W	—	x	—	Undefined
FF17H	Compare register 20	CR20	R/W	—	x	—	Undefined
FF18H	8-bit timer register 1	TM1	R	x	x	—	00H
FF19H	8-bit timer register 2	TM2	R	x	x	—	00H
FF18H-FF19H	16-bit timer register	TMS	R	—	—	x	0000H
FF1AH	Serial I/O shift register 0	SIO0	R/W	—	x	—	Undefined
FF1BH	Serial I/O shift register 1	SIO1	R/W	—	x	—	Undefined
FF1FH	A/D conversion result register	ADCR	R	—	x	—	Undefined
FF20H	Port mode register 0	PM0	R/W	x	x	—	1FH
FF21H	Port mode register 1	PM1	R/W	x	x	—	FFH
FF22H	Port mode register 2	PM2	R/W	x	x	—	FFH
FF23H	Port mode register 3	PM3	R/W	x	x	—	FFH
FF27H	Port mode register 7	PM7	R/W	x	x	—	1FH
FF2BH	Port mode register 11	PM11	R/W	x	x	—	FFH
FF2CH	Port mode register 12	PM12	R/W	x	x	—	FFH
FF40H	Timer clock select register 0	TCL0	R/W	x	x	—	00H
FF41H	Timer clock select register 1	TCL1	R/W	—	x	—	00H
FF42H	Timer clock select register 2	TCL2	R/W	—	x	—	00H
FF43H	Timer clock select register 3	TCL3	R/W	—	x	—	88H
FF47H	Sampling clock select register	SCS	R/W	—	x	—	00H
FF48H	16-bit timer mode control register	TMC0	R/W	x	x	—	00H
FF49H	8-bit timer mode control register	TMC1	R/W	x	x	—	00H
FF4AH	Watch timer mode control register	TMC2	R/W	x	x	—	00H
FF4EH	16-bit timer output control register	TOC0	R/W	x	x	—	00H
FF4FH	8-bit timer output control register	TOC1	R/W	x	x	—	00H
FF60H	Serial operating mode register 0	CSIM0	R/W	x	x	—	00H
FF61H	Serial bus interface control register	SBIC	R/W	x	x	—	00H

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF62H	Slave address register	SVA	R/W	—	x	—	Undefined
FF63H	Interrupt timing specify register	SINT	R/W	x	x	—	00H
FF68H	Serial operation mode register 1	CSIM1	R/W	x	x	—	00H
FF69H	Automatic data transmit/receive control register	ADTC	R/W	x	x	—	00H
FF6AH	Automatic data transmit/receive address pointer register	ADTP	R/W	—	x	—	00H
FF6BH	Automatic data transmission/reception interval specification register	ADTI	R/W	x	x	—	00H
FF80H	A/D converter mode register	ADM	R/W	x	x	—	01H
FF84H	A/D converter input select register	ADIS	R/W	—	x	—	00H
FFA0H	Display mode register 0	DSPM0	R/W	(Note 1)	x	—	00H
FFA1H	Display mode register 1	DSPM1	R/W	—	x	—	00H
FFA8H	6-bit up/down counter mode register	UDM	R/W	x	x	—	00H
FFA9H	6-bit up/down counter	UDC	R/W	—	x	—	00H
FFAAH	6-bit up/down counter compare register	UDCC	R/W	—	x	—	00H
FFE0H	Interrupt flag register L	IF0L	R/W	x	x	—	00H
FFE1H	Interrupt flag register H	IF0H	R/W	x	x	—	00H
FFE0H-FFE1H	Interrupt flag register	IF0	R/W	—	—	x	0000H
FFE4H	Interrupt mask flag register L	MK0L	R/W	x	x	—	FFH
FFE5H	Interrupt mask flag register H	MK0H	R/W	x	x	—	FFH
FFE4H-FFE5H	Interrrupt mask flag register	MK0	R/W	—	—	x	FFFFH
FFE8H	Priority order specify flag register L	PR0L	R/W	x	x	—	FFH
FFE9H	Priority order specify flag register H	PR0H	R/W	x	x	—	FFH
FFE8H-FFE9H	Priority order specify flag register	PR0	R/W	—	—	x	FFFFH
FFECH	External interrupt mode register	INTM0	R/W	—	x	—	00H
FFF0H	Memory size switch register (Note 2)	IMS	W	—	x	—	C8H
FFF7H	Pullup resistor option register	PU0	R/W	x	x	—	00H
FFF9H	Watchdog timer mode register	WDTM	R/W	x	x	—	00H
FFFAH	Oscillation stabilization time select register	OSTS	R/W	—	x	—	04H
FFFBH	Processor clock control register	PCC	R/W	x	x	—	04H

Notes:

(1) Bits 0-6 are read/write and bit 7 is read only.

(2) μPD78P044 only.

3e

Input/Output Ports

There are 68 port lines on each device in the μPD78044 family. Table 2 lists the port features and figure 4 shows the circuits at the I/O interfaces.

Table 2. Digital Port Features

Port	Input/Output	Notes	Configuration	Direct Drive	Software Pullup Resistors (Note 1)
0	5-bit I/O	2	Bit selectable	—	Byte selectable, input bits only
1	8-bit I/O	—			
2	8-bit I/O	—			
3	8-bit I/O	3	Bit selectable	LED	
7	5-bit, n-channel I/O	4	Bit selectable	—	Not available
8	2-bit, p-channel output	5, 6	N/A	LED, FIP	
9	8-bit, p-channel output	6, 7	NA	LED, FIP	
10	8-bit, p-channel output	6, 7, 8	N/A	LED, FIP	
11	8-bit, p-channel I/O	7	Bit selectable	FIP	
12	8-bit, p-channel I/O	7	Bit selectable	FIP	

Notes:

(1) Software pullup resistors can be internally connected (only on a port basis) to port bits set to input mode.

(2) $P0_0$ and $P0_4$ are input only and do not have a software pullup resistor.

(3) Mask ROM products: pulldown resistors are selectable per bit with a mask option. Not available on the μPD78P044.

(4) Mask ROM products: pullup resistors are selectable per bit with a mask option. Not available on the μPD78P044.

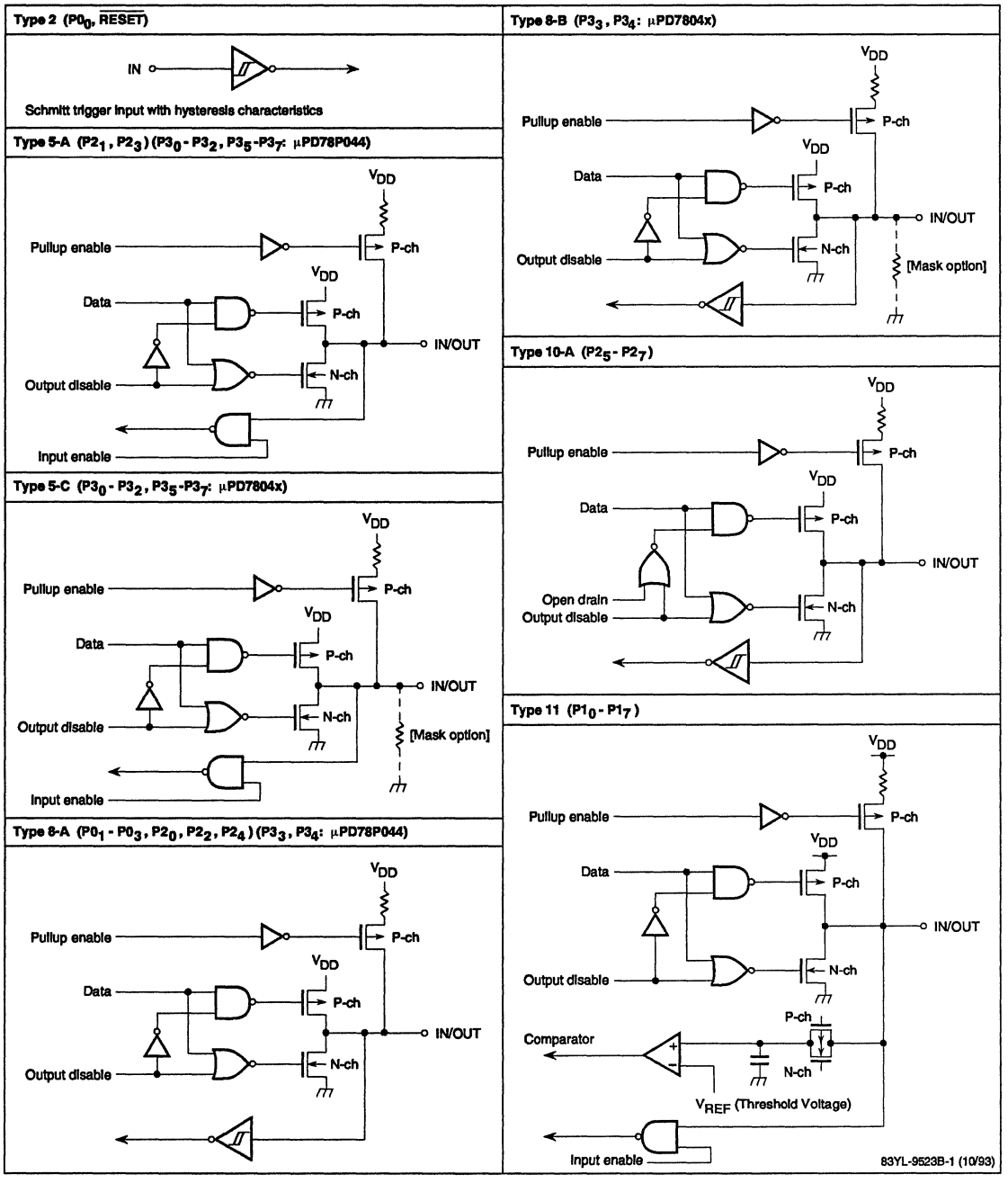
(5) Mask ROM products: pulldown resistors are selectable per bit with a mask option (connection to V_{LOAD} or V_{SS} specifiable as a 2-bit unit).

(6) μPD78P044: pulldown resistors incorporated for all bits (connection to V_{LOAD}).

(7) Mask ROM products: pulldown resistors are selectable per bit with a mask option (connection to V_{LOAD} or V_{SS} specifiable as 4-bit units).

(8) $P10_0$ - $P10_5$; LED, FIP direct drive. $P10_6$ - $P10_7$: FIP direct drive.

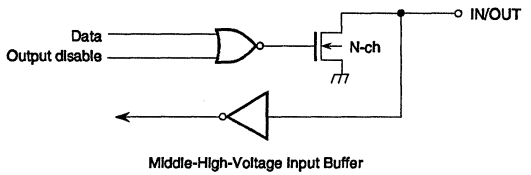
Figure 4. Pin Input/Output Circuits



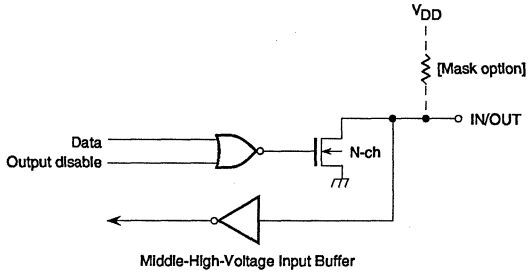
3e

Figure 4. Pin Input/Output Circuits (cont)

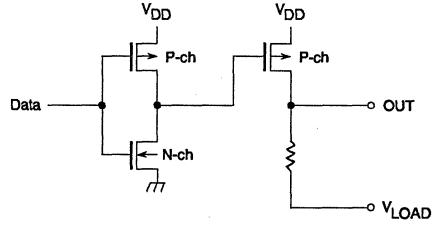
Type 13 ($P7_0$ - $P7_4$: μPD78P044)



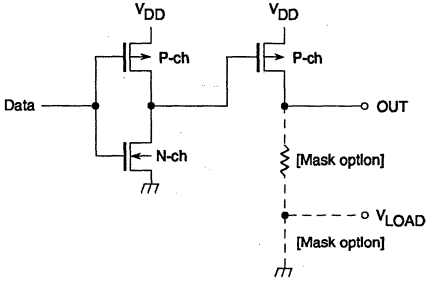
Type 13-A ($P7_0$ - $P7_4$: μPD7804x)



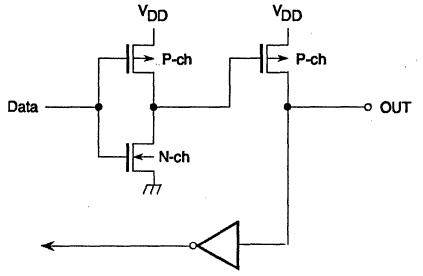
Type 14 ($P8_0$, $P8_1$, $P9_0$ - $P9_7$, $P10_0$ - $P10_7$: μPD78P044)



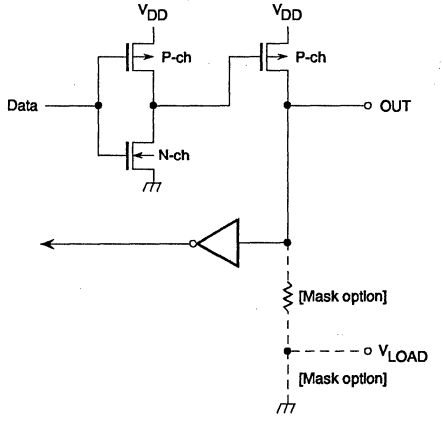
Type 14-A ($P8_0$, $P8_1$, $P9_0$ - $P9_7$, $P10_0$ - $P10_7$: μPD7804x)



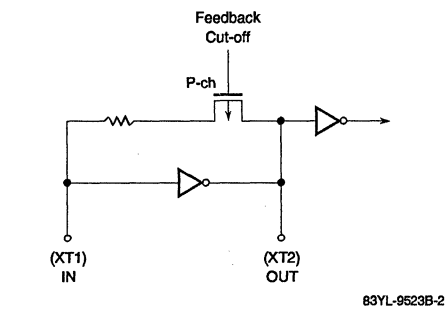
Type 15 ($P11_0$ - $P11_7$, $P12_0$ - $P12_7$: μPD78P044)



Type 15-A ($P11_0$ - $P11_7$, $P12_0$ - $P12_7$: μPD7804x)



Type 16 ($P0_4$, XT2)



83YL-9523B-2 (10/93)

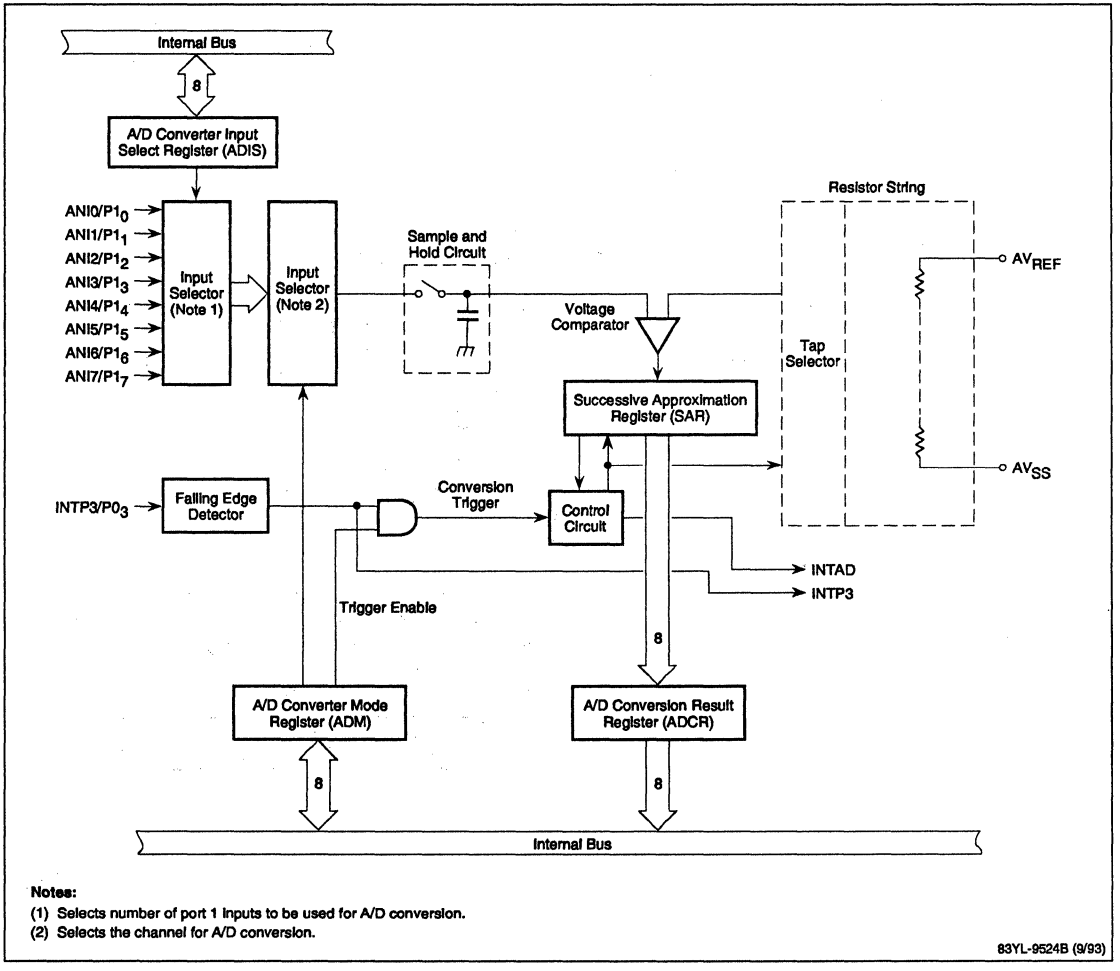
A/D Converter

The μPD78044 family's analog-to-digital converter (figure 5) uses the successive-approximation method for converting one of eight multiplexed analog inputs into 8-bit digital data. The minimum conversion time per input is 38.1 μs at 4.19-MHz operation.

The A/D converter input select register (ADIS) selects the number of inputs that are used in A/D conversion. The remaining inputs are used as ports. The analog input to be converted is selected by programming the A/D converter mode register (ADM). Conversion is started by external interrupt INTP3, or by writing to the ADM register. When conversion is completed, the results are stored in the A/D conversion result register (ADCR) and an INTAD interrupt is generated.

If the A/D converter was started by an external interrupt, it stops after the interrupt is generated. If the A/D converter was started by software, it repeats the conversion until new data is written to the ADM register.

Figure 5. A/D Converter



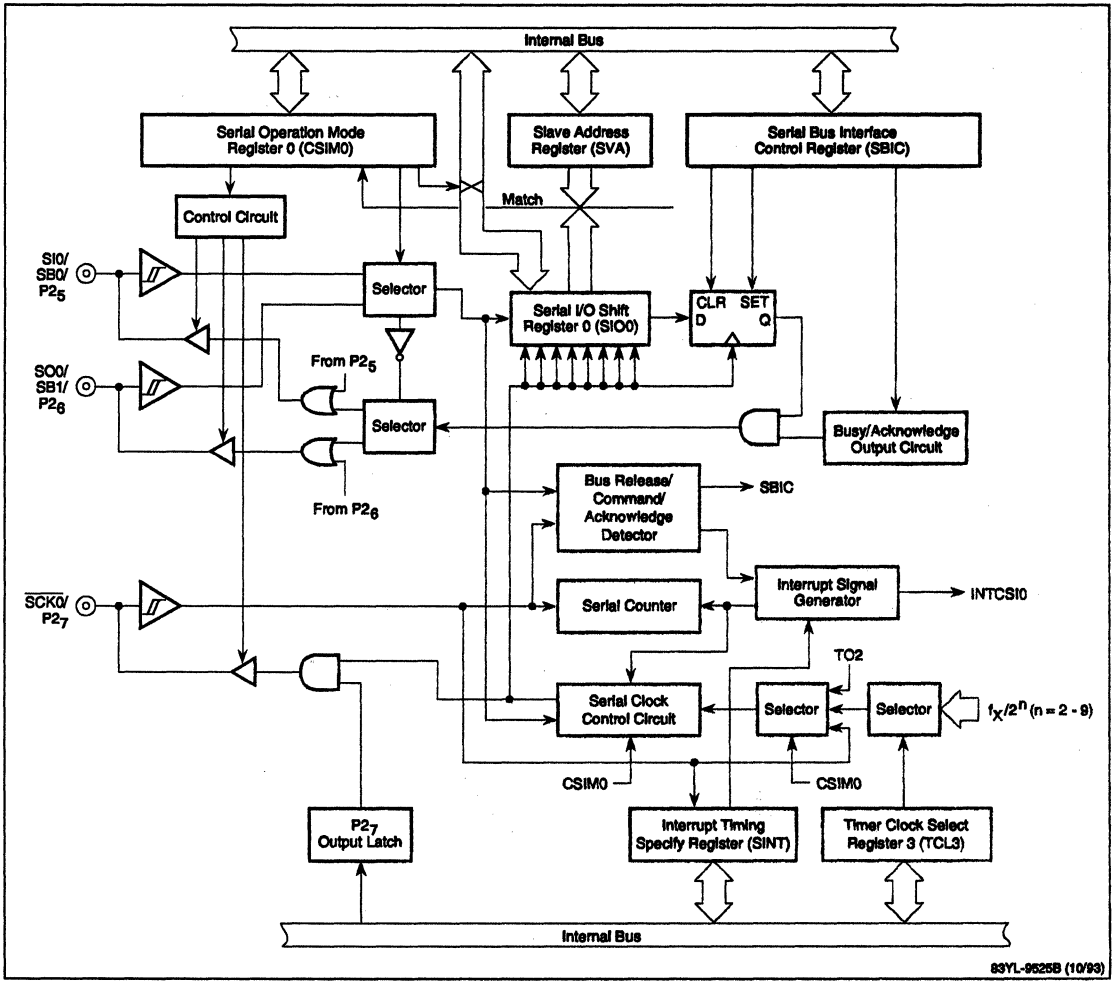
Notes:
(1) Selects number of port 1 inputs to be used for A/D conversion.
(2) Selects the channel for A/D conversion.

83YL-9524B (9/93)

Serial Interface 0

Serial interface 0 is an 8-bit, clock-synchronous interface. It can be operated in three-wire serial I/O mode, NEC serial bus interface (SBI) mode, or two-wire serial I/O mode. The serial clock can be provided from one of eight internal clocks, the output of 8-bit timer register 2, or external clock line $\overline{SCK0}$. See figure 6.

Figure 6. Serial Interface 0

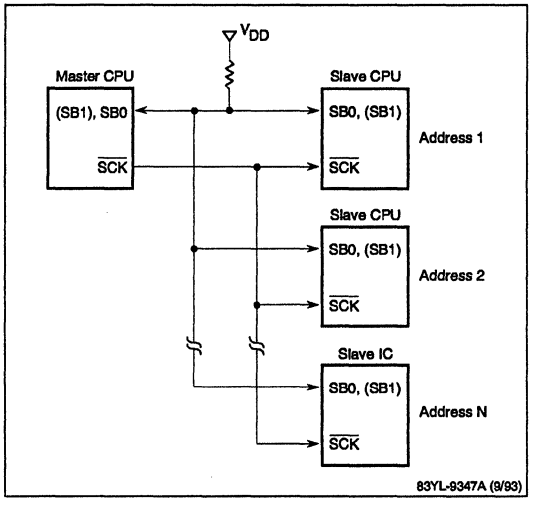


Three-Wire Interface. In the three-wire serial I/O mode, the 8-bit shift register (SIO0) is loaded with a data byte and eight clock pulses are generated. The clock pulse falling edges shift the data byte out to the SO0 line (either MSB or LSB first), while the rising edges shift data in from the SI0 line, providing full-duplex operation. The INTCSI0 interrupt is generated after each 8-bit transfer.

SBI Interface. The NEC SBI mode is a two-wire, high-speed, proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx families. Devices are connected in a master/slave configuration. See figure 7. There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over one of the serial bus lines (SB0 or SB1) using a fixed hardware protocol synchronized with the $\overline{SCK0}$ line.

Each slave device of the μPD78044 family can be programmed to respond in hardware to any one of 256 addresses set in its slave address register (SVA). There are also 256 commands and 256 data types. Since all commands are user definable, many software protocols, simple or complex, can be defined. It is even possible to develop commands that change a slave into a master and the previous master into a slave.

Figure 7. SBI Mode Master/Slave Configurtion



Two-Wire Interface. The two-wire serial I/O mode provides half-duplex operation using either the SB0 or SB1 line and the $\overline{SCK0}$ line. Communication format and handshaking can be handled in software by controlling the output levels of the data and clock lines between transfers. For data transmission, when 8-bit shift register (SIO0) is loaded with a data byte and eight clock pulses are generated. The falling edges shift the data byte out either the SB0 or SB1 line, MSB first. In addition, this data byte is also shifted back into SIO0 on the rising pulse edges providing a means of verifying that the transmission was correct.

For data reception, the SIO0 register is preloaded with the value FFH. As this data value is shifted out on the falling edge of the serial clock, it disables the n-channel open-drain driver. This allows the receive data to be driven onto the serial line and shifted into the SIO0 register on the rising edge of the serial clock. The INTCSI0 interrupt is generated after each 8-bit transfer.

3e

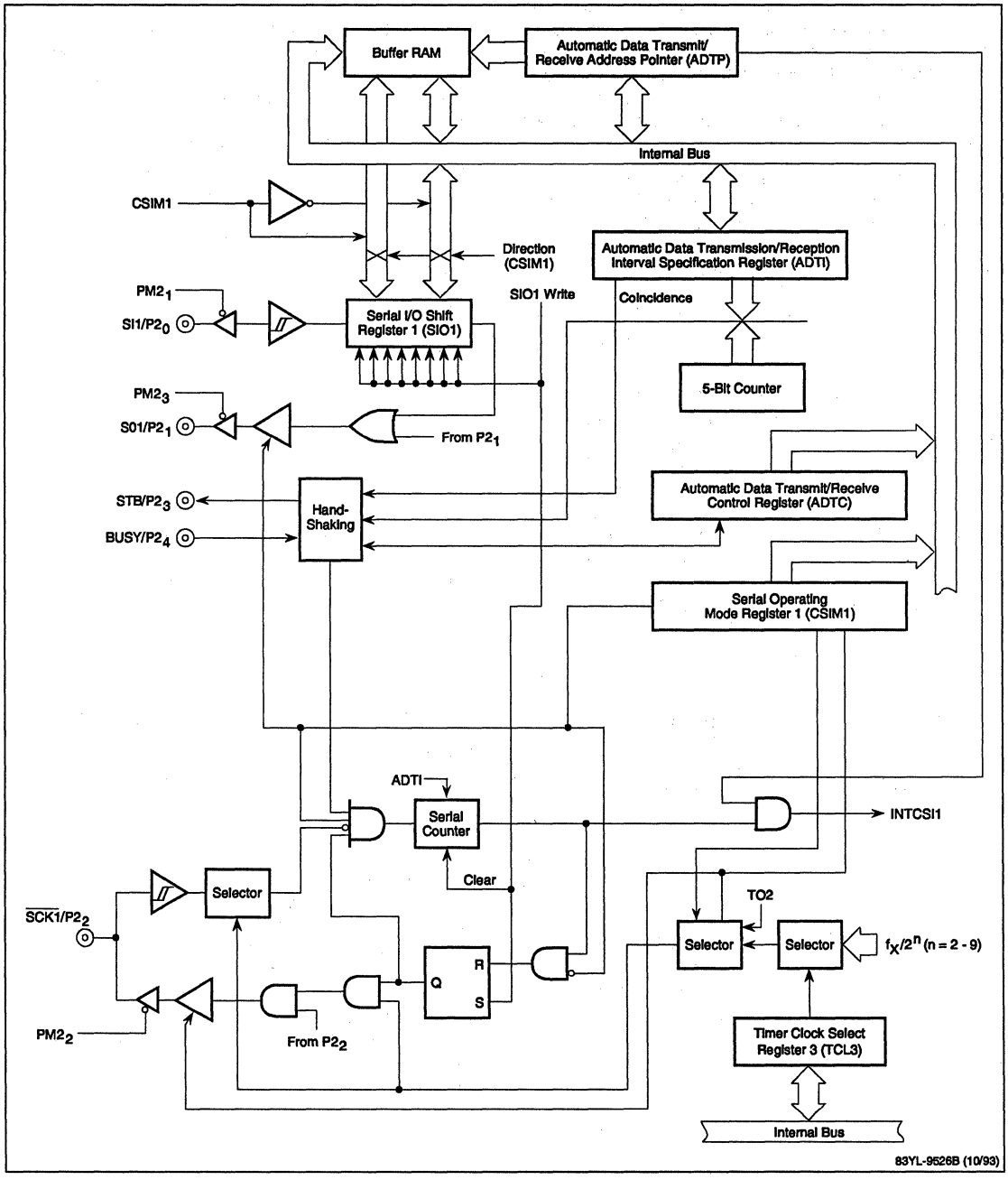
Serial Interface 1

Serial interface 1 is also an 8-bit, clock-synchronous interface (figure 8). It can be operated in either a three-wire serial I/O mode or a three-wire serial I/O mode with automatic transmit/receive. The serial clock can also be provided from one of eight internal clocks (common clock for both interfaces), the output of 8-bit timer register 2, or the external clock line $\overline{SCK1}$.

Three-Wire. In the three-wire serial I/O mode, the 8-bit shift register (SIO1) is loaded with a data byte and eight clock pulses are generated. The clock pulse falling edges shift the data byte out of the SO1 line (MSB or LSB first) while the rising edges shift the data in from the SI1 line, providing full-duplex operation. The INTCSI1 interrupt is generated after each 8-bit transfer.

Three-Wire With Auto Xmt/Rcv. In the three-wire serial I/O mode with automatic transmit/receive, up to 64 data bytes can be transferred with minimal CPU overhead. The data to be transmitted and received is stored in the buffer RAM. Handshaking over the BUSY input line or the strobe (STB) output line, or both, can be selected by the program. Error detection of bit drift due to noise is available for each byte transferred when using the BUSY input line. This automatic transmit/receive mode is ideally suited for transferring data to/from external peripheral devices such as on-screen display (OSD) and LSC controller/driver devices.

Figure 8. Serial Interface 1



While in three-wire serial I/O mode with automatic data transfer, the interface can be operated full duplex or transmit only in single or repetitive operation. In full duplex, a data byte is transferred from the first location in the buffer RAM and shifted out the SO1 line (MSB or LSB first) while the received data is shifted in the SI1 line and stored back in the first buffer location. After the preset number of bytes have been transferred, the INTCSI1 interrupt is generated.

In single-operation transmit mode, the preset number of bytes from the buffer RAM are transmitted out the SO1 line (MSB or LSB first) and the INTCSI1 interrupt is generated after all bytes have been transferred. In repetitive-operation transmit mode, data in the buffer is transmitted repeatedly.

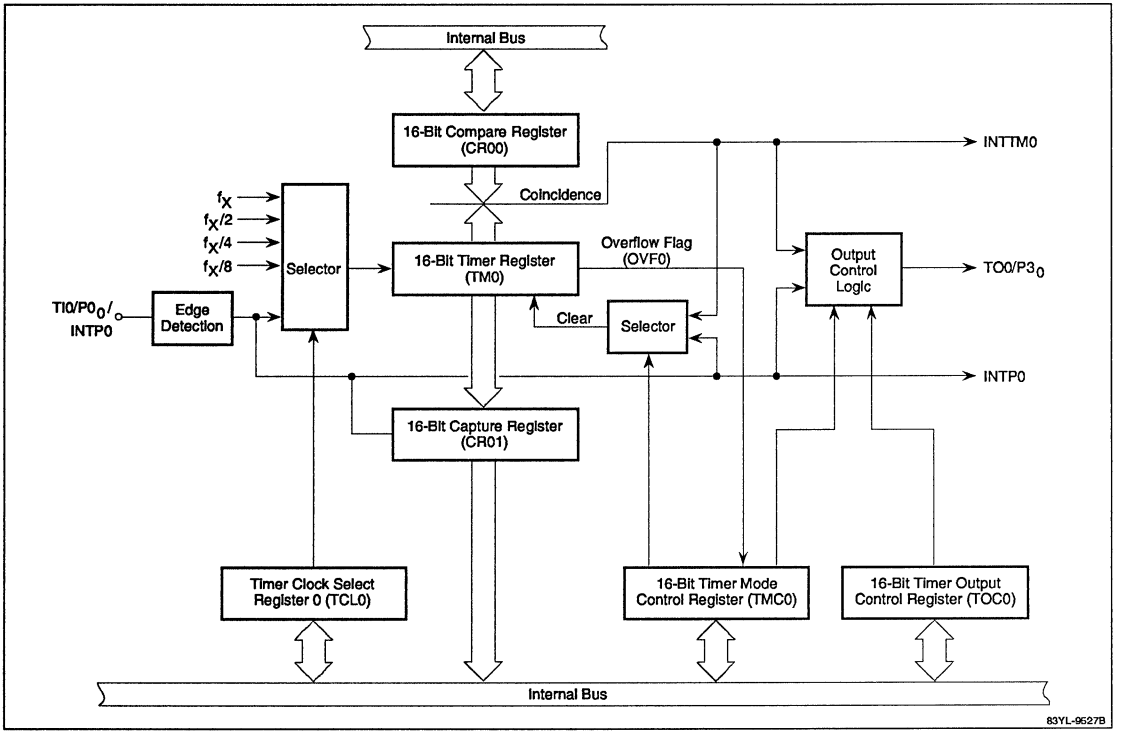
Timers

The μPD78044 family has a 16-bit timer/event counter, two 8-bit timer/event counters (combinable for 16-bit operation), a 6-bit up/down counter, a watch timer, and a watchdog timer. All except the up/down counter can be programmed to count a number of prescaled values of the main system clock. The watch timer can also count the subsystem clock. All timer/event counters and the up/down counter can count external events.

16-Bit Timer/Event Counter 0. Timer/event counter 0 (figure 9) includes a 16-bit counter (TM0), a 16-bit compare register (CR00), a 16-bit capture register (CR01), and a timer output (TO0). Timer 0 can be used (1) as an interval timer, (2) to count external events on the timer input (TI0) pin, (3) to output a programmable square wave or a 14-bit pulse-width modulated output, or (4) to measure pulse widths.

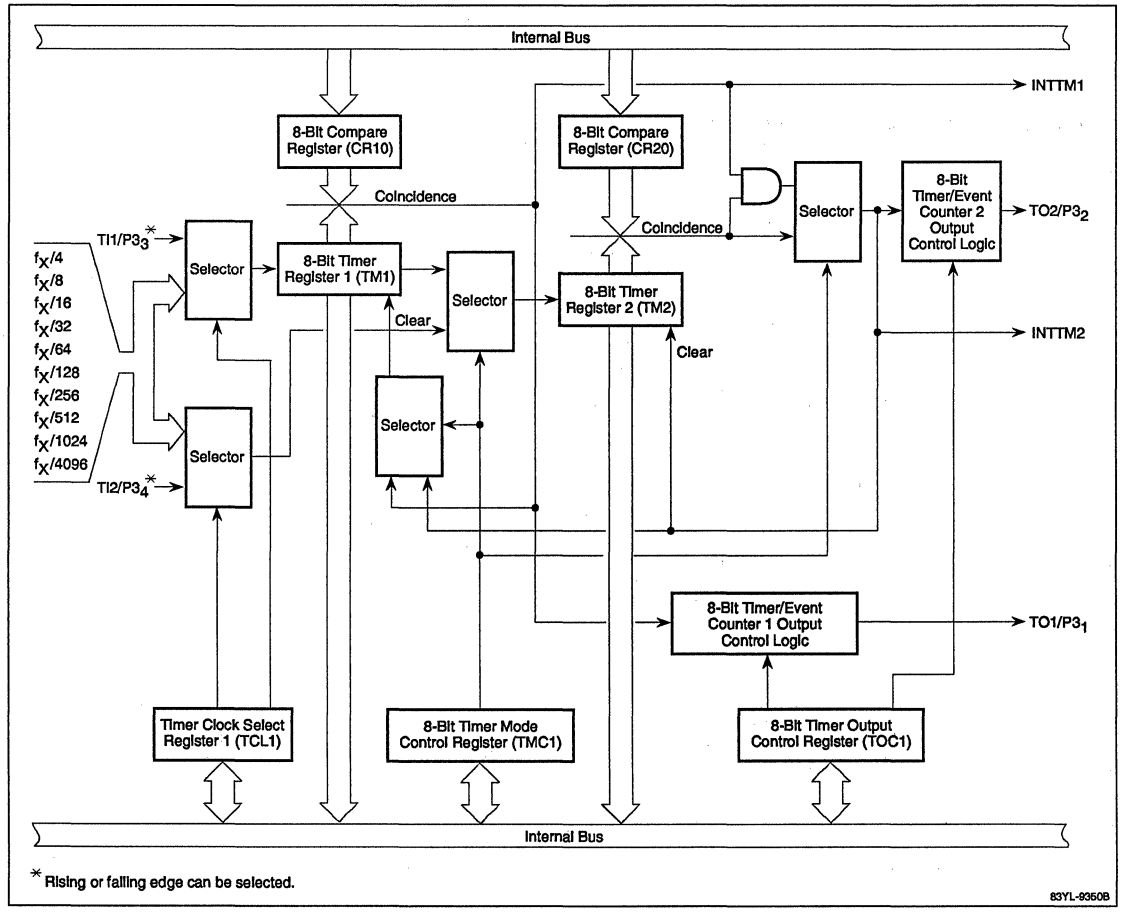
3e

Figure 9. 16-Bit Timer/Event Counter 0



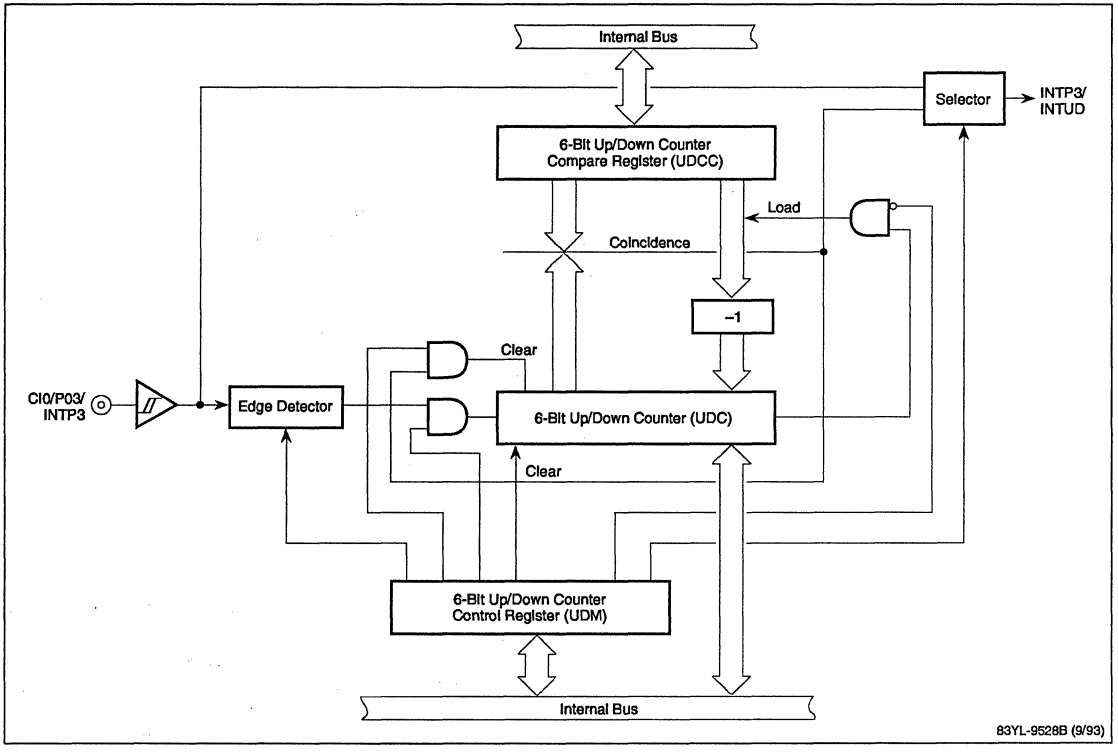
8-Bit Timer/Event Counters 1 and 2. Timer/event counters 1 and 2 (figure 10) each consists of an 8-bit timer register (TM1 or TM2), an 8-bit compare register (CR10 or CR20), and a timer output control logic (TO1 or TO2). The timers are controlled by registers TCL1, TMC1, and TOC1 via five selectors. Timer/event counters 1 and 2 each can be used as an 8-bit interval timer, to count external events on timer input pin TI1 or TI2, or to output a programmable square wave. Also, timers 1 and 2 can be combined as a 16-bit timer/event counter and used as a 16-bit interval timer, to count external events on TI1, or to output a programmable square wave on TO2.

Figure 10. 8-Bit Timer/Event Counters 1 and 2



6-Bit Up/Down Counter. The up/down counter (figure 11) includes counter UDC and compare register UDCC. It counts external events (up or down) on the counter input pin (CI0) and generates an interrupt (INTUD) when the count matches the compare register. The counter is loaded with −1 (down-count mode) or cleared (up-count mode) upon interrupt generation.

Figure 11. 6-Bit Up/Down Counter



Watch Timer 3. Watch timer 3 (figure 12) is a 5-bit timer that can be used as a time source to keep track of time of day, to release the STOP or HALT mode at regular intervals, or to initiate any other task that must be performed at regular intervals. When driven by the subsystem clock, the watch timer continues to operate in the STOP mode.

The watch timer can function as both a watch timer and an interval timer simultaneously. In watch timing, interrupt request INTWT (not a vectored interrupt) can be generated using the main system clock every 0.5 or 1.0 second or by using the subsystem clock every 0.5 or 0.25 second.

In interval timing, vectored interrupt request INTTM3 is generated at preselected time intervals. With a 4.19-MHz main system clock, the following time intervals can be selected: 978 μs; 1.96, 3.91, 7.82, 15.6, or 31.3 ms. With a 32.768-kHz subsystem clock, the following time intervals can be selected: 488 or 978 μs; 1.96, 3.91, 7.82, or 15.6 ms.

Watchdog Timer. The watchdog timer (figure 13) can also perform interval timing. As a watchdog timer, it protects against inadvertent program run-away. It can be selected to generate a nonmaskable interrupt (INTWDT), which vectors to address 0004H, or to generate an internal reset signal, which vectors to the restart address 0000H if the timer is not cleared by the program before it overflows. Eight program-selectable intervals based on the main system clock are available. With a 4.19-MHz main system clock, they are 0.489, 0.978, 1.96, 3.91, 7.82, 15.6, 31.3, and 125 ms. Once initialized and started, the timer cannot change modes and can be stopped only by an external reset.

In interval timing, maskable interrupts (INTWDT), which vector to address 0004H, are generated repeatedly at a preset interval. The time intervals available are the same as in the watchdog timer mode.

Programmable Clock Output

The μPD78044 family has a programmable clock output (PCL) that can be used as carrier output for remote controlled transmissions or as clock output for peripheral devices. The main system clock (f_X) divided by 8, 16, 32, 64, 128, or 256 or the subsystem clock (f_{XT}) can be output on the PCL pin. With a 4.19-MHz main system clock, the following frequencies are available: 524, 262, 131, 65.5, 32.7, and 16.4 kHz. With a 32.768-kHz subsystem clock, 32.768 kHz is also available. See figure 14.

Buzzer Output

The μPD78044 family also has a programmable buzzer output (BUZ). The buzzer output frequency can be programmed to equal the main system clock (f_X) divided by 1024, 2048, or 4096. With a 4.19-MHz main system clock, the buzzer can be set to 4.1, 2.0, or 1.0 kHz. See figure 15.

FIP Controller/Driver

The μPD78044 family can directly drive up to 34 FIP (fluorescent indicator panel) display output lines of which 9 to 24 segments and 2 to 16 digits can be selected through software. The number of digits is selected by display mode register 0 (DSPM0) and the number of segments by display mode register 1 (DSPM1). If an attempt is made to select a total of more than 34 digit and segment outputs, the digit selection will take priority. Any unused pins can be used as outputs or I/O depending on the type.

There are 48 bytes of display data RAM mapped from FA50H to FA7FH. Each display memory bit corresponds to a specific display element. Any bits not used for FIP display can be used for general purpose.

Segment and digit signal output is automatically controlled by a DMA operation from the FIP controller to the display data RAM. The display cycle period is $1024/f_X$ or $2048/f_X$. Register DSPM1 selects the display cycle and one of eight intensity levels. The on-chip circuitry controls the intensity level by varying the driving signal pulse width.

The on-chip circuitry has been designed to allow easy interface to a keyboard. At the end of the display cycle, vectored interrupt INTKS is generated and the controller outputs key scan data (ports 11 and 12) on segment output pins FIP18 to FIP33. Flag KSF indicates key scan or display timing to the key scan software routine.

Pulldown resistors for all FIP lines are mask-selected and can be connected to V_{LOAD} or V_{SS}. Pulldown resistors for FIP0 to FIP17 lines are incorporated in the μPD78P044 and are connected to V_{LOAD}.

Figure 12. Watch Timer 3

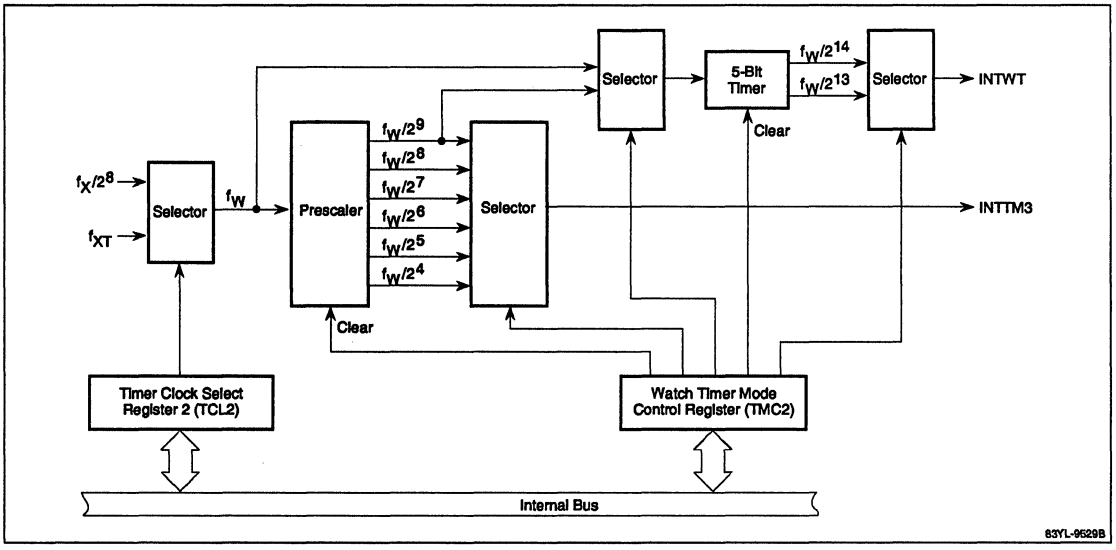
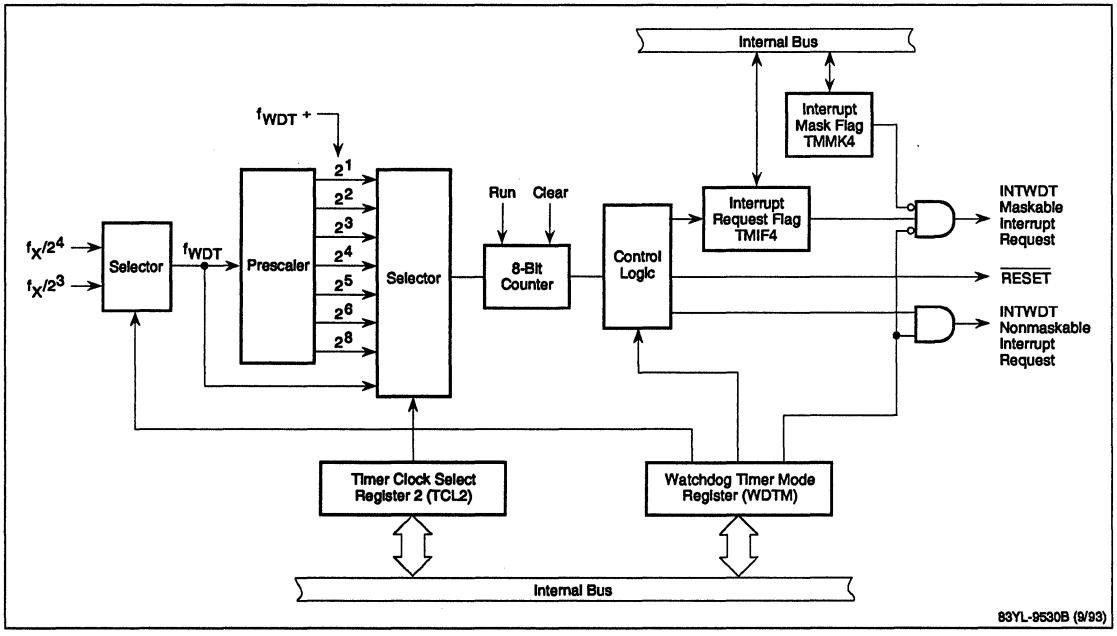


Figure 13. Watchdog Timer



3e

Figure 14. Programmable Clock Output

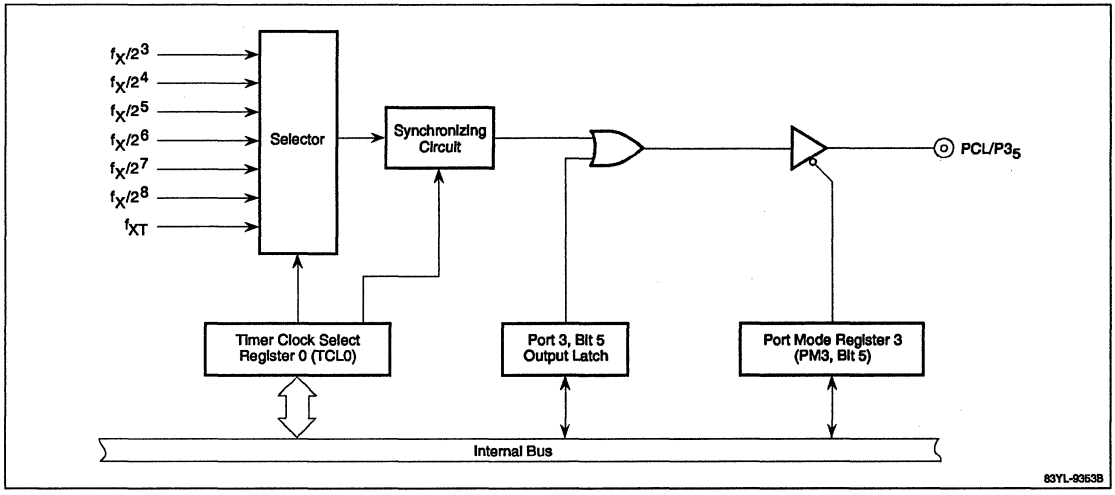


Figure 15. Buzzer Output

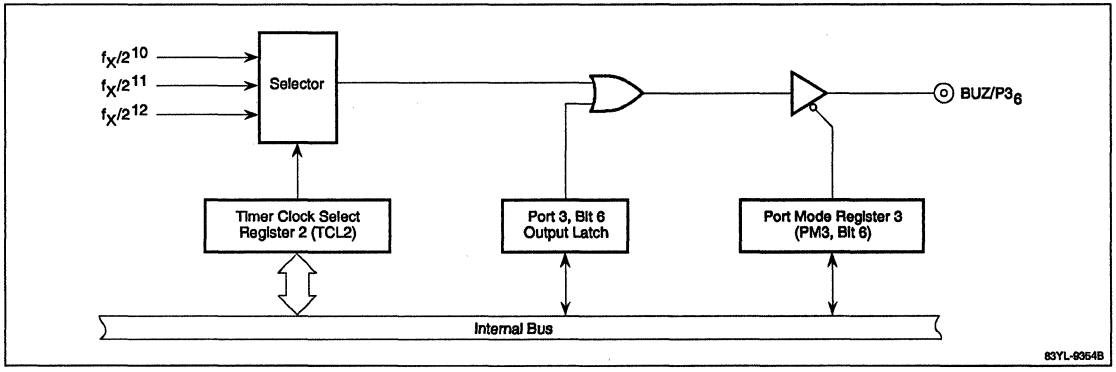
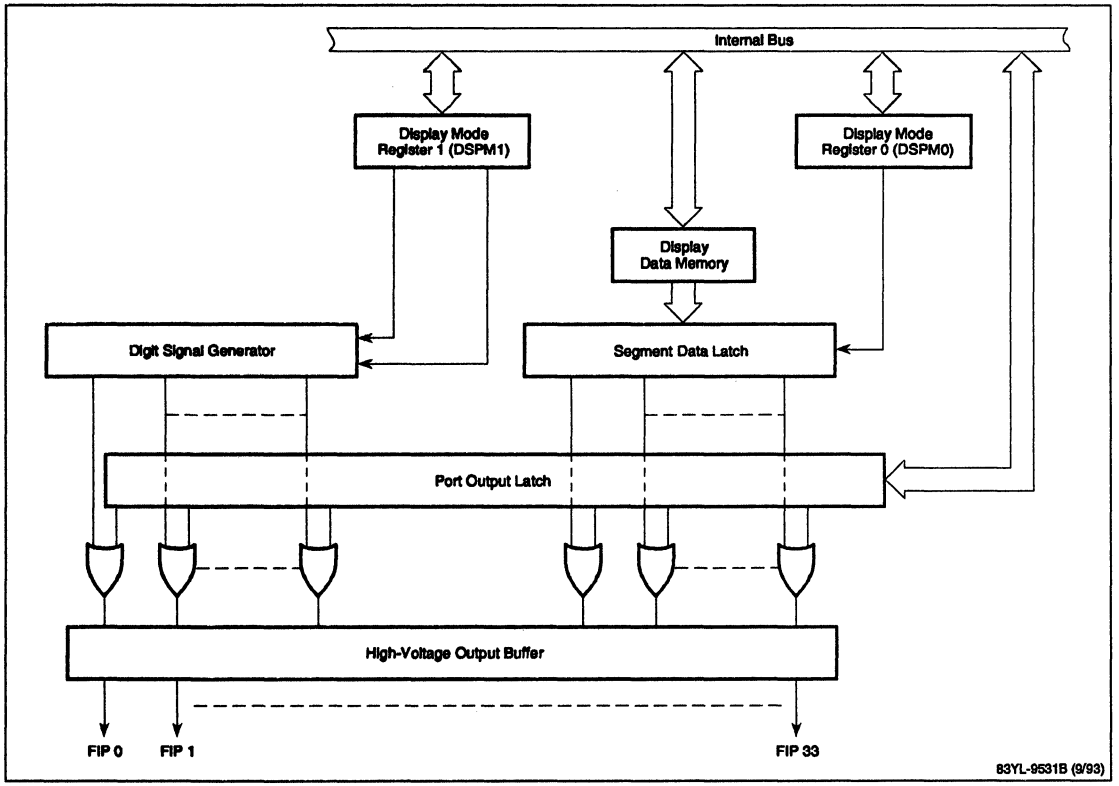


Figure 16. FIP Controller/Driver



3e

Interrupts

The μPD78044 family has 14 maskable hardware interrupt sources: four are external, nine are internal, and one (INTP3/INTUD) can be set for either external or internal. Thirteen of them cause a vectored interrupt; one testable input-only generates an interrupt request. All 14 maskable interrupts can be used to release the HALT mode except INTP0 (when SCS = 0) and INTKS; all except INTKS and INTP0 can release the STOP mode.

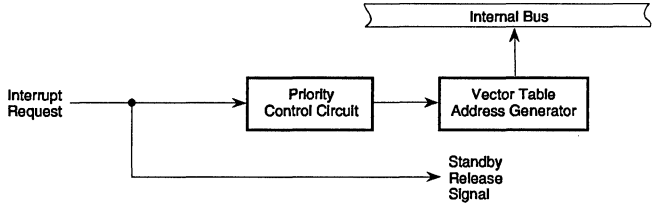
In addition, there is one nonmaskable interrupt from the watchdog timer, one software interrrupt, and a RESET interrupt. The watchdog timer overflow interrupt (interrupt vector table address 0004H) can be initialized to be a nonmaskable interrupt or the highest default priority maskable interrupt. The software interrupt generated by the BRK instruction is not maskable. See table 3 and figure 17.

Table 3. Interrupt Sources and Vector Addresses

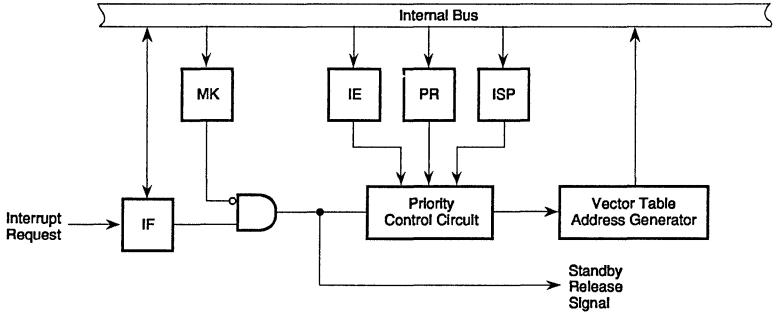
Type of Request	Default Priority	Signal Name	Interupt Source	Location	Vector Address	Interrupt Configuration
Restart	—	RESET	$\overline{\text{RESET}}$ input pin	External	0000H	—
		INTWDT	Watchdog timer overflow (when reset mode selected)	Internal		
Nonmaskable		INTWDT	Watchdog timer overflow (when nonmaskable interrupt selected)	Internal	0004H	A
Maskable	0	INTWDT	Watchdog timer overflow (when interval timer selected)	Internal	0004H	B
	1	INTP0	External interrupt edge detection	External	0006H	C
	2	INTP1	External interrupt edge detection	External	0008H	D
	3	INTP2	External interrupt edge detection	External	000AH	D
	4	INTP3	External interrupt edge detection	External	000CH	D
		INTUD	Up/down counter coincidence signal	Internal		B
	5	INTCSI0	End of clocked serial interface 0 transfer	Internal	000EH	B
	6	INTCSI1	End of clocked serial interface 1 transfer	Internal	0010H	B
	7	INTTM3	Watch timer reference time interval signal	Internal	0012H	B
	8	INTTM0	16-bit timer/event counter coincidence signal	Internal	0014H	B
	9	INTTM1	8-bit timer/event counter 1 coincidence signal	Internal	0016H	B
	10	INTTM2	8-bit timer/event counter 2 coincidence signal	Internal	0018H	B
	11	INTAD	End of A/D conversion	Internal	001AH	B
	12	INTKS	Key scan interrupt generated by FIP controller	Internal	001CH	B
Software		—	BRK instruction	Internal	003EH	E
Test input		INTWT	Clock timer overflow	Internal	—	F

Figure 17. Interrupt Configuration

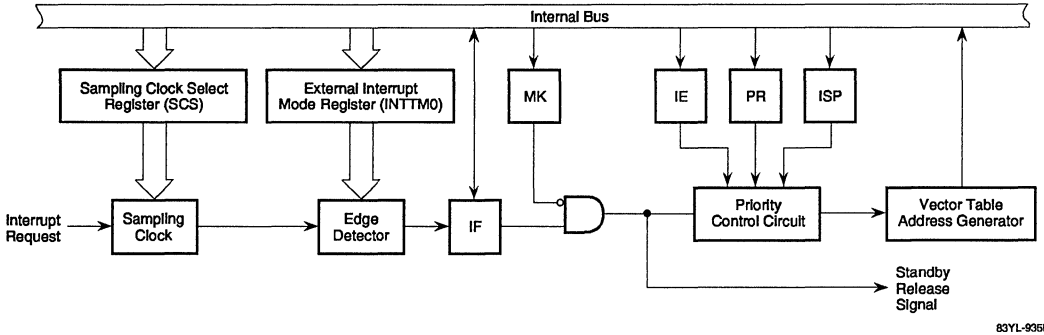
Type A: Internal nonmaskable interrupt



Type B: Internal maskable interrupt



Type C: External maskable interrupt (INTP0)

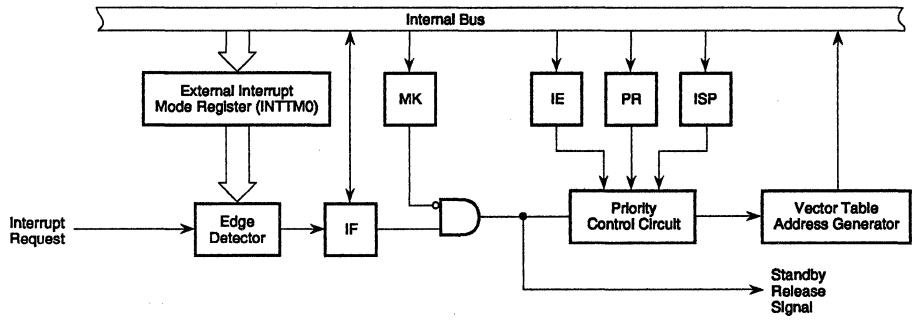


83YL-9356B (1)

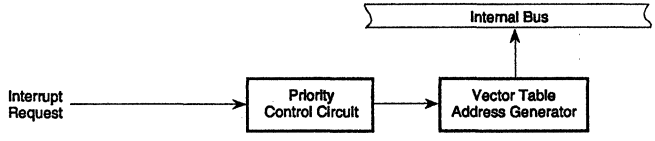
3e

Figure 17. Interrupt Configuration (cont)

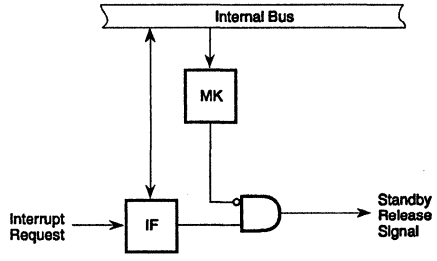
Type D: External maskable interrupt (except INTP0)



Type E: Software Interrupt



Type F: Test Input



Abbreviations:

IF: Interrupt request flag
IE: Interrupt enable flag
ISP: In-service priority flag
MK: Interrupt mask flag
PR: Priority specify flag

83YL-93658 (2)

Interrupt Servicing. The μPD78044 family provides two levels of programmable hardware priority control and services all interrupt requests except the testable interrupt (INTWT) using vectored interrupts. The programmer can choose the priority of servicing each maskable interrupt by the interrupt control registers.

Interrupt Control Registers. The μPD78044 family has three 16-bit interrupt control registers. The interrupt request flag register (IF0) contains an interrupt request flag for each interrupt. The interrupt mask register (MK0) is used to enable or disable any individual interrupt. The priority flag register (PR0) can specify a high or a low priority level for each interrupt except the testable interrupt (INTWT).

Three other 8-bit registers are associated with interrupt processing. The external interrupt mode register (INTM0) selects a rising or falling edge (or both) as the valid edge for external interrupts INTP0, INTP1, and INTP2 (INTP3 is always falling edge). The sampling clock select register (SCS) selects a sampling clock for the noise eliminator circuit on external interrupt INTP0.

The IE and ISP bits of the program status word also control interrrupts. If the IE bit is zero, all maskable interrupts are disabled. The IE bit can be set or cleared by the EI or DI instruction, respectively, or by directly writing to the PSW. The IE bit is cleared each time an interrupt is accepted. The ISP bit is used by hardware to hold the priority level flag of the interrupt being serviced.

Interrupt Priority. If the watchdog timer overflow interrupt (INTWDT) has been initialized to be a nonmaskable interrupt, it has priority over all other interrupts. Two hardware-controlled priority levels are available for all maskable interrupts that generate a vectored interrupt (that is, all except the testable interrupt). Either a high or a low priority level can be assigned by software to each of the maskable interrupts.

Interrupt requests of the same priority or a priority higher than the processor's current priority level are held pending until interrupts in the current service routine are enabled by software or until one instruction has been executed after returning from the current service routine. Interrupt requests of a lower priority are always held pending until one instruction has been executed after returning from the current service routine.

The default priorities in table 3 are fixed by hardware; they are effective only when necessary to choose between two interrupt requests of the same software-assigned priority. For example, after the completion of a high-priority routine, if two interrupts of the same software priority were pending.

The software interrupt, initiated by the BRK instruction, is executed regardless of the procesor's priority and the state of the IE bit. It does not alter the processor's priority.

Vectored Interrupt Servicing. When a vectored interrupt is acknowledged, the program status word and the program counter are saved on the stack, the processor's priority is set to that specified for the interrupt, the IE bit in the PSW is set to zero, and the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (RETB instruction for the software interrupt) reverses the process and the μPD78044 family microcomputer resumes the interrupted routine.

3e

Standby Modes

The HALT, STOP, and data retention modes reduce power consumption when CPU action is not required.

The HALT mode is entered by executing a HALT instruction while the CPU is operating from the main system or subsystem clock. In HALT mode, the CPU clock is stopped while the main system and the subsystem clock continue to run. The HALT mode is released by any unmasked interrupt request (except INTP0 if register SCS = 0 and INTKS), a nonmaskable interrupt request, an unmasked test input, or an external reset pulse.

Power consumption may be further reduced by the STOP mode. The STOP mode is entered by executing a STOP instruction while operating from the main system clock. In STOP mode, the main system clock input pin X1 is internally grounded, stopping both the CPU and the peripheral hardware clock. The STOP mode is released by any unmasked interrupt request except INTP0 and INTKS, a nonmaskable interrupt request, an unmasked test input, or an external reset pulse.

Any peripheral using the main oscillator as a clock source will also be disabled in the STOP mode and interrupts from such a peripheral cannot be used to exit the STOP mode. Table 4 summarizes the HALT and STOP standby modes.

When exiting the STOP mode, a wait time occurs before the CPU begins code execution to allow the main system clock oscillator circuit to stabilize. The oscillation stabilization time is selected by programming the OSTS register with one of five values before entering the STOP mode and ranges from $2^{12}/f_X$ to $2^{17}/f_X$ seconds.

Table 4. Standby Mode Operation Status

Item	HALT Mode	STOP Mode
Setting instruction	HALT instruction	STOP instruction
System clock when setting	Main system or subsystem clock	Main system clock
Clock oscillator	Main system and subsystem clocks can oscillate; CPU clock is stopped.	Subsystem clock can oscillate; CPU and main system clocks are stopped.
CPU	Operation stopped	Operation stopped
Ports	Maintain previous state	Maintain previous state
16-bit timer/event counter	Operational from main system clock	Operation stopped
8-bit timer/event counters	Operational from main system clock or with TI1 and TI2 selected as the count clock	Operational only with TI1 and TI2 as count clock
6-bit up/down counter	Operable	Operable
FIP controller/driver	Inoperable	Inoperable
Watch timer	Operational from main system clock or with f_{XT} as count clock	Operational only with f_{XT} as count clock
Watchdog timer	Operational from main system clock	Operation stopped
Serial interface 0	Operational from main system clock or with external clock	Operational only with external clock
Serial interface 1	Operational from main system clock or with external clock; no automatic transmit/receive mode	Operational only with external clock; no automatic transmit/receive mode
A/D converter	Operational from main system clock	Operation stopped
External interrupts	Operational except for INTP0 when its sampling clock is based on the CPU clock	INTP0 not operational; INTP1 to INTP3 operational

Once in the STOP mode, power consumption can be further minimized by lowering the power supply voltage V_{DD} to 2 volts. This places the device in the data retention mode. The contents of internal RAM and the registers are retained. This mode is released by first raising V_{DD} to the proper operating range and then releasing the STOP mode.

External Reset

The μPD78044 family is reset by taking the $\overline{\text{RESET}}$ pin low or by an overflow of the watchdog timer (if enabled). The $\overline{\text{RESET}}$ input pin is a Schmitt trigger input with hysteresis characteristics to protect against spurious system resets by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low for 10 μs minimum after the power supply reaches its operating voltage.

There is no functional difference between an external reset and an internal reset caused by watchdog timer overflow. In both cases, the main system clock oscillation is stopped and the subsystem clock oscillation continues. During reset, the program counter is loaded with the address in the reset vector (addresses 0000H, 0001H). Once the reset is cleared and the oscillation stabilization time of $2^{17}/f_X$ has elapsed, program execution starts at that address.

NEC
NEC Electronics Inc.

μPD78054 Family (μPD78052/053/054/P054) 8-Bit, K-Series Microcontrollers With UART, A/D and D/A Converters

Preliminary

September 1993

Description

The μPD78052, μPD78053, μPD78054, and μPD78P054 are members of the K-Series® of microcontrollers featuring an A/D and a D/A converter, UART, 8-bit hardware multiply and divide instructions, bit manipulation instructions, four banks of main registers, an advanced interrupt handling facility, and a powerful set of memory-mapped on-chip peripherals.

Timing is generated by two oscillators. The main oscillator normally drives the CPU amd most peripherals. The 32.768-kHz subsystem oscillator provides time keeping when the main oscillator is turned off. Since CMOS power dissipation is proportional to clock rate, the 78054 family provides a software selectable instruction cycle time from 0.40 μs to122 μs. The STOP and HALT modes turn off parts of the microcontroller for additional power savings. The data retention mode keeps RAM contents valid down to 2.0 volts.

These devices are ideally suited for applications in portable battery-powered equipment, office automation, communications, consumer electronics, home appliances, and fitness equipment.

K-Series is a registered trademark of NEC Electronics, Inc.

Features

- Eight-channel 8-bit A/D converter
- Two-channel 8-bit D/A converter
  - Real-time output capability
- Three-channel serial communication interface
  - 8-bit clock-synchronous interface 0
    Full-duplex, three-wire mode
    Half-duplex, two-wire mode
    NEC serial bus interface (SBI) mode
  - 8-bit clock-synchronous interface 1
    Full-duplex, three-wire mode
    Automatic transfer, full-duplex, three-wire mode
  - Serial interface 2
    Full-duplex, three-wire mode
    UART mode
- Timers: five channels
  - Watchdog timer
  - 16-bit timer/event counter
  - Two 8-bit timer/event counters usable as one 16-bit timer/event counter
  - Watch (clock) timer
- 69 I/O lines
  - Two CMOS input-only lines
  - 63 CMOS bidirectional I/O lines
  - One real-time output port operable in one 8-bit or two 4-bit units
  - Four n-channel, open-drain I/O lines at 15 V maximum
  - Software selectable pullup resistors on 63 lines
  - Mask option pullup resistors on four lines available on ROM versions
- External memory expansion
  - 64K bytes total memory space
- Powerful instruction set
  - 8-bit unsigned multiply and divide
  - 16-bit arithmetic and data transfer instructions
  - 1-bit and 8-bit logic instructions
- Minimum instruction execution times:
  - 0.4/0.8/1.6/3.2/6.4 μs program selectable using 5-MHz main system clock
  - 122 μs using 32.768-kHz subsystem clock
- Memory-mapped on-chip peripherals
  - Special function registers
- Programmable priority, vectored-interrupt controller (two levels)
- Programmable buzzer and clock outputs
- Power saving and battery back up
  - Variable CPU clock rate
  - STOP mode
  - HALT mode
  - 2-V data retention mode
- CMOS operation; V_{DD} from 2.7 to 6.0 V

Internal High-Capacity ROM and RAM

	78052	78053	78054	78P054
ROM	16K bytes	24K bytes	32K bytes	—
PROM	—	—	—	32K bytes
High-speed RAM	512 bytes	1024 bytes	1024 bytes	1024 bytes
Serial buffer RAM	32 bytes	32 bytes	32 bytes	32 bytes

3f

Ordering Information

Part Number	ROM	Package	Package Drawing
μPD78052GC-xxx-3B9	16K mask ROM	80-pin plastic QFP	S80GC-65-3B9-1
μPD78053GC-xxx-3B9	24K mask ROM		
μPD78054GC-xxx-3B9	32K mask ROM		
μPD78P054GC-3B9	32K OTP ROM		
μPD78052GK-xxx-BE9	16K mask ROM	80-pin plastic TQFP	P80GK-50-BE9-1
μPD78053GK-xxx-BE9	24K mask ROM		
μPD78054GK-xxx-BE9	32K mask ROM		
μPD78P054GK-BE9	32K OTP ROM		
μPD78P054KK-T	32K UV EPROM	80-pin ceramic LCC w/window	X80KW-65A

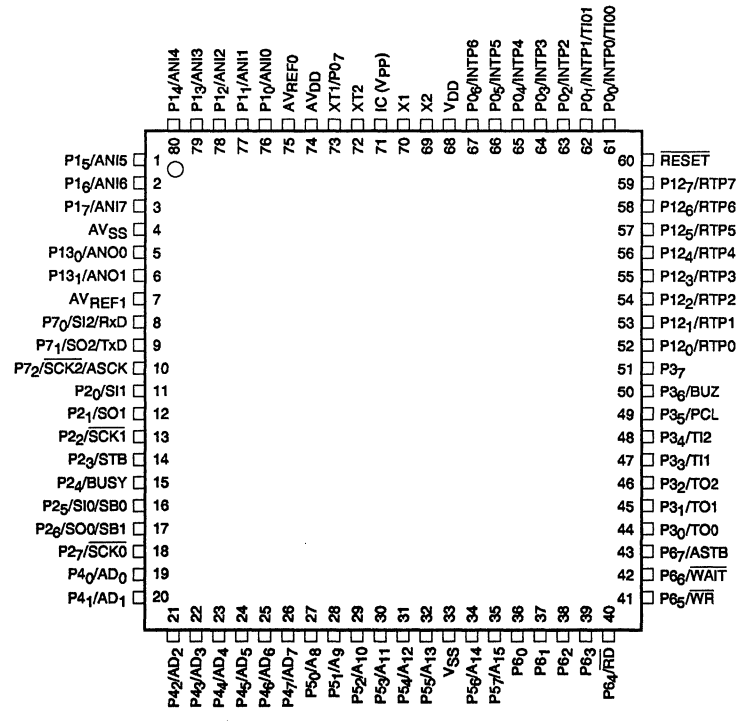
Notes:

(1) xxx indicates ROM code suffix

(2) All devices listed are standard quality grade

Pin Configurations

80-Pin Plastic QFP, Plastic TQFP, or Ceramic LCC



Notes:
(1) Connect IC (internally connected) pin (V PP on μPD78P054) to V SS.
(2) AV_{DD} should be connected to V_{DD}.
(3) AV_{SS} should be connected to V_{SS}.

83RC-9548B (9/93)

3f

Pin Functions; Normal Operating Mode

Symbol	First Function	Symbol	Alternate Function
$P0_0$	Port 0; 8-bit, bit selectable I/O port (Bits 0 and 7 are input only)	INTP0 TI0	External maskable interrupt External count clock input to timer 0 or timer 0 capture trigger to capture registers CR00 and CR01
$P0_1$		INTP1 TI01	External maskable interrupt Timer 0 capture trigger to capture register CR00
$P0_2$ $P0_3$ $P0_4$ $P0_5$ $P0_6$		INTP2 INTP3 INTP4 INTP5 INTP6	External maskable interrupt
$P0_7$		XT1	Crystal oscillator or external clock input for subsystem clock
$P1_0$ - $P1_7$	Port 1; 8-bit, bit-selectable I/O port	ANI0-ANI7	Analog input to A/D converter
$P2_0$	Port 2; 8-bit, bit-selectable I/O port	SI1	Serial data input three-wire serial I/O mode
$P2_1$		SO1	Serial data output three-wire serial I/O mode
$P2_2$		$\overline{SCK1}$	Serial clock I/O for serial interface 1
$P2_3$		STB	Serial interface automatic transmit/receive strobe output
$P2_4$		BUSY	Serial interface automatic transmit/receive busy input
$P2_5$		SI0 SB0	Serial data input three-wire serial I/O mode 2/3-wire serial I/O mode
$P2_6$		SO0 SB1	Serial data output three-wire serial I/O mode 2/3-wire serial I/O mode
$P2_7$		$\overline{SCK0}$	Serial clock I/O for serial interface 0
$P3_0$	Port 3; 8-bit, bit-selectable I/O port	TO0	Timer output from timer 0
$P3_1$		TO1	Timer output from timer 1
$P3_2$		TO2	Timer output from timer 2
$P3_3$		TI1	External count clock input to timer 1
$P3_4$		TI2	External count clock input to timer 2
$P3_5$		PCL	Programmable clock output
$P3_6$		BUZ	Programmable buzzer output
$P3_7$		—	
$P4_0$ - $P4_7$	Port 4; 8-bit I/O port	AD_0 - AD_7	Low-order 8-bit multiplexed address/data bus for external memory
$P5_0$ - $P5_7$	Port 5; 8-bit, bit selectable I/O port	A_8 - A_{15}	High-order 8-bit address bus for external memory
$P6_0$ - $P6_3$	Port 6; 8-bit, bit selectable ($P6_0$ to $P6_3$ n-channel, open-drain I/O with mask option pullup resistors; $P6_4$ - $P6_7$ I/O). See note.	—	
$P6_4$		$\overline{RD}$	External memory read strobe
$P6_5$		$\overline{WR}$	External memory write strobe
$P6_6$		$\overline{WAIT}$	External memory wait signal input
$P6_7$		ASTB	Address strobe used to latch address for external memory

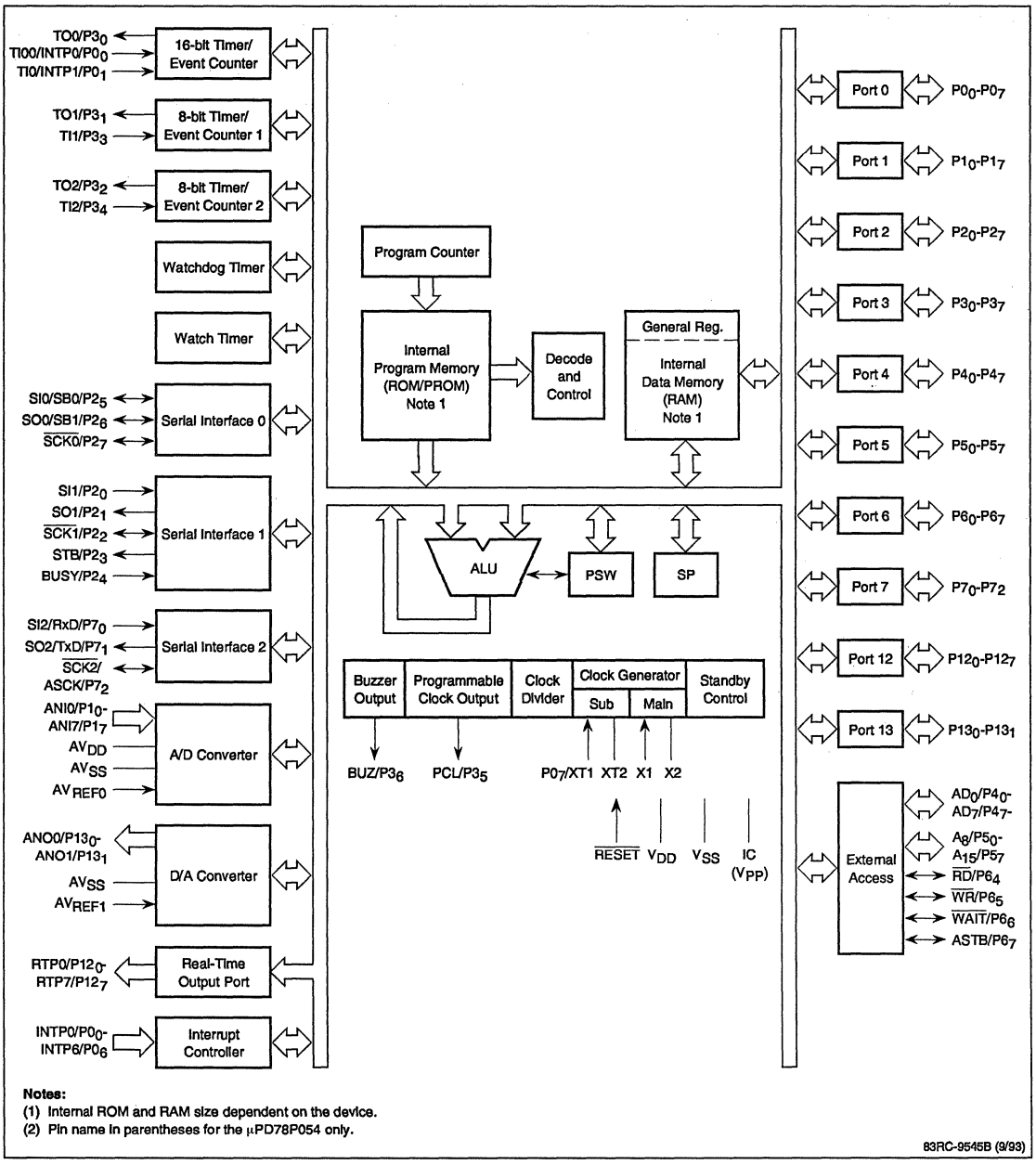
Pin Functions; Normal Operating Mode (cont)

Symbol	First Function	Symbol	Alternate Function
$P7_0$	Port 7; 3-bit , bit-selectable I/O port	SI2 RxD	Serial data input three-wire serial I/O mode Asynchronous serial data input
$P7_1$		SO2 TxD	Serial data output three-wire serial I/O mode Asynchronous serial data output
$P7_2$		$\overline{SCK2}$ ASCK	Serial clock I/O for serial interface 2 Asynchronous serial clock input
$P12_0$- $P12_7$	Port 12; 8-bit selectable I/O port	RTP0- RTP7	Real-time port
$P13_0$, $P13_1$	Port 13; 2-bit selectable I/O port	ANO0, ANO1	Analog output for D/A converter
$\overline{RESET}$	External system reset input		
X1	Crystal/ceramic resonator connection or external clock input for main system clock		
X2	Crystal/ceramic resonator connection or inverse of external clock for main system clock		
XT2	Crystal oscillator or left open when not using the subsystem clock		
AV_{REF0}	A/D converter reference voltage		
AV_{REF1}	D/A converter reference voltage		
AV_{DD}	A/D converter power supply input		
AV_{SS}	A/D and D/A converter ground		
V_{DD}	Power-supply input		
V_{PP}	μPD78P054 PROM programming power-supply input		
V_{SS}	Power-supply ground		
IC	Internal connection		

Note: See table 2 and figure 4 for details.

3f

Block Diagram



Notes:
(1) Internal ROM and RAM size dependent on the device.
(2) Pin name in parentheses for the μPD78P054 only.

83RC-9545B (9/93)

FUNCTIONAL DESCRIPTION

Central Processing Unit

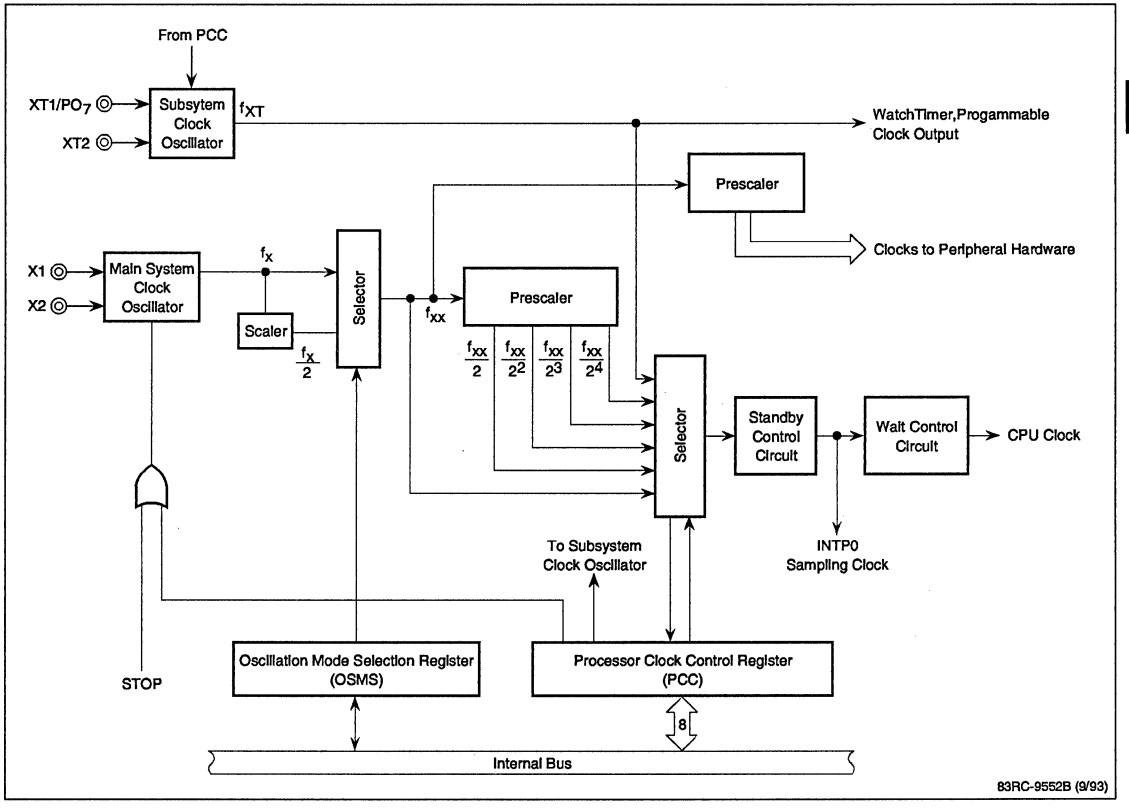
The central processing unit (CPU) of the μPD78054 family features 8- and 16-bit arithmetic including an 8 x 8-bit unsigned multiply and 16 x 8-bit unsigned divide (producing a 16-bit quotient and an 8-bit remainder). The multiply executes in 3.2 μs and the divide in 5 μs using the fastest clock cycle with a main system clock of 5.0 MHz.

A CALLT vector table and a CALLF area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction can access up to 32 subroutines through their addresses contained in the CALLT vector table (40H to 7FH). A 2-byte call instruction can access any routine beginning in a specific CALLF area (0800H to 0FFFH).

Internal System Clock Generator

The internal system clocks of the μPD78054 family are derived from either the main system or the subsystem oscillator. See figure 1. The clocks for the watch timer and programmable clock output are derived from either the subsystem clock (f_{XT}) or the main system clock. The clocks for all other peripheral hardware are derived from the main system clock.

Figure 1. Internal System Clock Generator



3f

The CPU clock (ϕ) can be supplied from either the main system clock (f_X) or the subsystem clock (f_{XT}). A selector, which is controlled by the oscillation mode selection register (OSMS), determines whether the main system clock (f_X) or the scaled main system clock ($f_X/2$) is provided to the prescalar (f_{XX}). Using the processor clock control register (PCC), a CPU clock frequency equal to f_{XX}, $f_{XX}/2$, $f_{XX}/4$, $f_{XX}/8$, $f_{XX}/16$ or the subsystem clock f_{XT} can be selected. The CPU clock selected should be based on the power supply voltage available and the desired power consumption. On power up, the CPU clock defaults to the lowest speed from the main system clock ($f_{XX}/16$ with $f_{XX} = f_X/2$) and can be changed while the microcomputer is running.

Since the shortest instruction takes two CPU clocks to execute, the fastest minimum instruction execution time (t_{CY}) of 0.4 μs is achieved with a main system clock at 5.0 MHz ($f_{XX} = f_X$) and a V_{DD} of 4.5 to 6.0 volts. However, if the watch timer must generate an interrupt every 0.5 or 0.25 seconds, t_{CY} is 0.48 μs at 4.19 MHz with $f_{XX} = f_X$. The fastest minimum instruction execution time available across the full voltage range of 2.7 to 6.0 volts is 0.96 μs with a 4.19 MHz main system clock ($f_{XX} = f_X$). For the lowest power consumption, the CPU can be operated from the subsystem clock and the minimum instruction execution time is 122 μs at 32.768 kHz.

Memory Space

The μPD78054 family has a 64K-byte address space (see figure 2). This address space (0000H-FFFFH) can be used as both program and data memory.

Internal Program Memory

All devices in the μPD78054 family have internal program memory. The μPD78052/053/054 contain 16K, 24K, and 32K bytes of internal ROM, respectively. The μPD78P054 contains 32K bytes of UV EPROM or one time programmable ROM. To allow the μPD78P054 to emulate the mask ROM devices, the amount of internal program memory available in the μPD78P054 can be selected using the memory size switching register (IMS).

Internal RAM

The μPD78052 has 544 bytes and the μPD78053/054/P054 have 1056 bytes of Internal RAM. This Internal RAM consists of two types: high-speed Internal RAM and buffer RAM.

The μPD78052 contains 512 bytes (FD00H to FEFFH) while the μPD78053/054/P054 contain 1024 bytes (FB00H to FEFFH) of high-speed Internal RAM. The high-speed Internal RAM contains the general register banks and the stack. The remainder of the high-speed Internal RAM and any unused register bank locations are available for general storage.

All devices also contain 32 bytes of buffer RAM (FAC0H to FADFH). The buffer RAM is used for the automatic transfer mode of serial interface 1 or for general storage.

To allow the μPD78P054 to emulate the mask ROM devices, the amount of high-speed Internal RAM available in the μPD78P054 can also be selected using the IMS.

External Memory

The μPD78054 family can access 0, 256, 4K, 16K or all available bytes of external memory. The μPD78054 family has an 8-bit wide external data bus and a 16-bit wide external address bus. The low-order 8 bits of the address bus are multiplexed and also provide the 8-bit data bus and are supplied by port 4. The high-order address bits of the 16-bit address bus are taken from port 5 as required. The address latch, read, and write strobes, and the external $\overline{WAIT}$ signal are supplied by port 6.

The memory expansion mode register (MM) controls the size of external memory. It can be programmed to use 0, 4, 6, or 8 bits from port 5 for the high-order address. Any remaining port 5 bits can be used for I/O. The MM register also can be used to specify one additional wait state or the use of the external $\overline{WAIT}$ signal for low-speed external memory or external peripheral devices.

When only internal ROM and RAM are used and no external memory is required, ports 4, 5 and 6 are available as general purpose I/O ports.

CPU Control Registers

Program Counter. The program counter is a 16-bit binary counter register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 0001H.

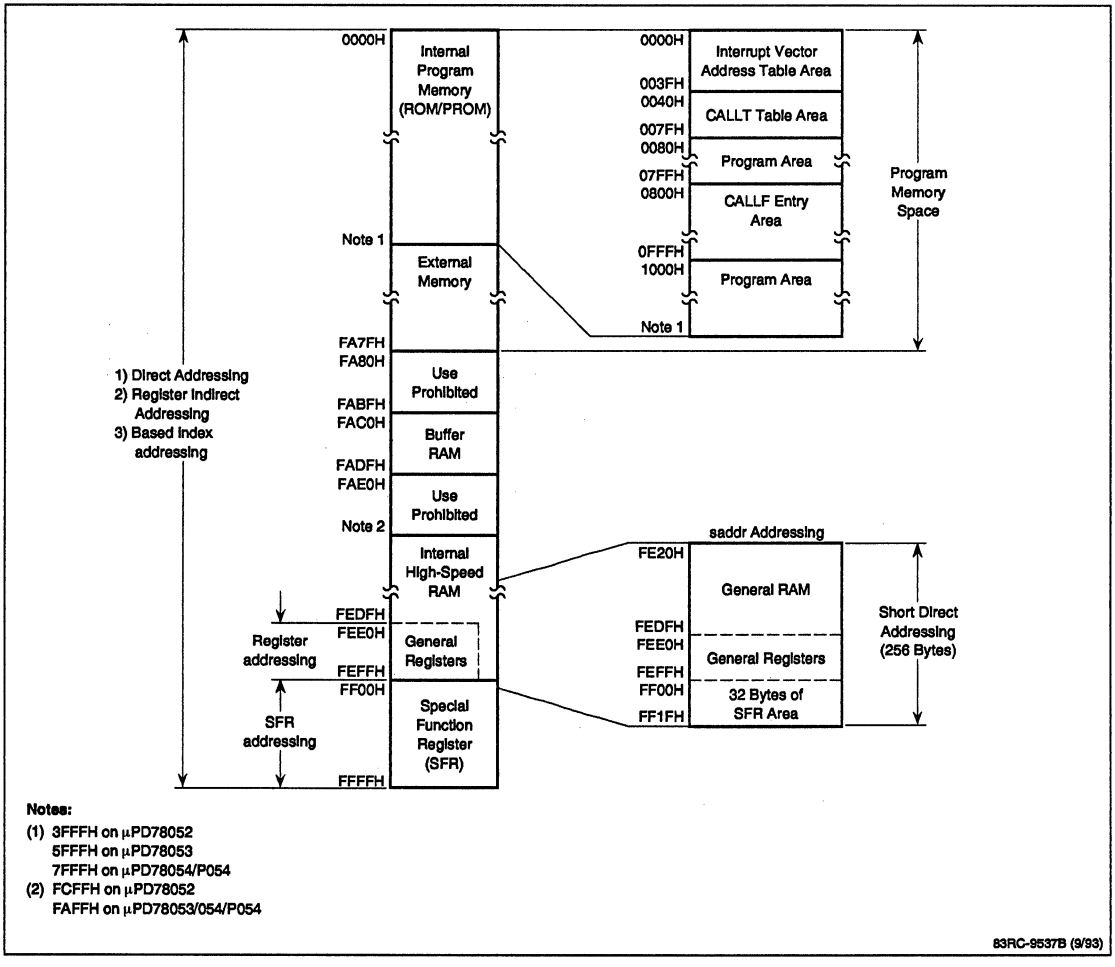
Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

Program Status Word. The program status word (PSW) is an 8-bit register that contains flags that are set or reset depending on the results of an instruction. This register can be written to or read from 8 bits at a time. The individual flags can also be manipulated on a bit-by-bit basis. The assignment of PSW bits follows.

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

CY	Carry flag
ISP	In-service (interrupt) priority flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

Figure 2. Memory Map

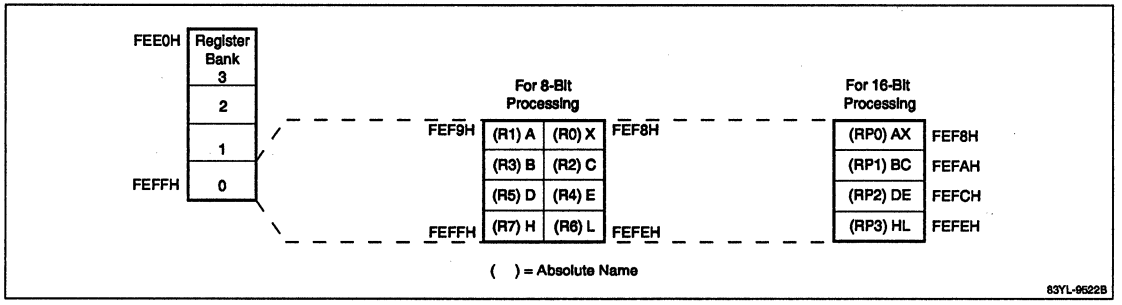


Notes:
(1) 3FFFH on μPD78052
5FFFH on μPD78053
7FFFH on μPD78054/P054
(2) FCFFH on μPD78052
FAFFH on μPD78053/054/P054

83RC-9537B (9/93)

General Registers

Figure 3. General Registers



The general-purpose registers (figure 3) consist of four banks of registers located at addresses FEE0H to FEFFH in Internal RAM. Each bank consists of eight 8-bit general registers that can also be used in pairs to function as four 16-bit registers. Two bits in the PSW (RBS0 and RBS1) specify which of the register banks is active at any time and are set under program control.

Registers have both functional names (A, X, B, C, D, E, H, or L for 8-bit registers and AX, BC, DE, and HL for 16-bit registers) and absolute names (R1, R0, R3, R2, R5, R4, R7, or R6 for 8-bit registers and RP0, RP1, RP2, or RP3 for 16-bit registers). Either the functional or absolute register names can be used in instructions that use the operand identifiers r and rp.

Addressing

The program memory addressing (ROM) modes provided are relative, immediate, table indirect and register addressing. The operand addressing modes provided are implied, register, direct, short direct (saddr), special function (SFR), register indirect, based, based indexed, and stack addressing.

The SFR addressing and saddr addressing modes use direct addressing and require only 1 byte in the instruction to address RAM. Normally a 65K byte address space requires 2 bytes to address it. One-byte addressing results in faster access times, since the instructions are shorter. SFR addressing addresses the entire 256-byte SFR address space from FF00H to FFFFH. Saddr addressing (see figure 2) addresses the 256-byte address space FE20H to FF1FH. FE20H to FEFFH are composed of 224 bytes of internal high speed RAM; FF00H to FF1FH contain the first 32 bytes in the special function register area.

One-byte addressing is accomplished by using the first byte of the instruction for the opcode (and one operand if register A or AX is used) and the second byte of the instruction as an address (offset) into the 256-byte area. If register A or AX is used, the instructions are 2 bytes long, thereby providing fast access times. If immediate data is used, the instruction will be 3 or 4 bytes long depending upon whether the immediate data is a byte or a word. Many 16-bit SFRs are in the space FF00H to FF1FH. Using AX as an operand to these SFRs will provide fast access, since the instructions will be only 2 bytes long.

Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by SFR addressing. FF00H to FF1H can also be accessed using saddr addressing. They are either 8 or 16 bits as required, and many of the 8-bit registers are bit addressable. Table 1 lists the special function registers.

Input/Output Ports

Each device in the μPD78054 family has 69 port lines. Table 2 lists the features of each port and figure 4 shows the structure of each port pin.

Table 1. Special Function Registers

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF00H	Port 0	P0	R/W	x	x	—	00H
FF01H	Port 1	P1	R/W	x	x	—	00H
FF02H	Port 2	P2	R/W	x	x	—	00H
FF03H	Port 3	P3	R/W	x	x	—	00H
FF04H	Port 4	P4	R/W	x	x	—	Undefined
FF05H	Port 5	P5	R/W	x	x	—	Undefined
FF06H	Port 6	P6	R/W	x	x	—	Undefined
FF07H	Port 7	P7	R/W	x	x	—	00H
FF0CH	Port 12	P12	R/W	x	x	—	00H
FF0DH	Port 13	P13	R/W	x	x	—	00H
FF10H-FF11H	Capture/compare register 00	CR00	R/W	—	—	x	Undefined
FF12H-FF13H	Capture/compare register 01	CR01	R/W	—	—	x	Undefined
FF14H-FF15H	16-bit timer register	TM0	R	—	—	x	00H
FF16H	Compare register 10	CR10	R/W	—	x	—	Undefined
FF17H	Compare register 20	CR20	R/W	—	x	—	Undefined
FF18H	8-bit timer register 1	TM1	R	x	x	—	00H
FF19H	8-bit timer register 2	TM2	R	x	x	—	00H
FF18H-FF19H	16-bit timer register	TMS	R	—	—	x	0000H
FF1AH	Serial I/O shift register 0	SIO0	R/W	—	x	—	Undefined
FF1BH	Serial I/O shift register 1	SIO1	R/W	—	x	—	Undefined
FF1FH	A/D conversion result register	ADCR	R	—	x	—	Undefined
FF20H	Port mode register 0	PM0	R/W	x	x	—	FFH
FF21H	Port mode register 1	PM1	R/W	x	x	—	FFH
FF22H	Port mode register 2	PM2	R/W	x	x	—	FFH
FF23H	Port mode register 3	PM3	R/W	x	x	—	FFH
FF25H	Port mode register 5	PM5	R/W	x	x	—	FFH
FF26H	Port mode register 6	PM6	R/W	x	x	—	FFH
FF27H	Port mode register 7	PM7	R/W	x	x	—	FFH
FF2CH	Port mode register 12	PM12	R/W	x	x	—	FFH
FF2DH	Port mode register 13	PM13	R/W	x	x	—	FFH
FF30H	Real-time output port buffer register L	RTBL	R/W	—	x	—	00H
FF31H	Real-time output port buffer register H	RTBH	R/W	—	x	—	00H
FF34H	Real-time output port mode register	RTPM	R/W	x	x	—	00H
FF36H	Real-time output port control register	RTPC	R/W	x	x	—	00H
FF40H	Timer clock select register 0	TCL0	R/W	x	x	—	00H
FF41H	Timer clock select register 1	TCL1	R/W	—	x	—	00H
FF42H	Timer clock select register 2	TCL2	R/W	—	x	—	00H
FF43H	Timer clock select register 3	TCL3	R/W	—	x	—	88H
FF47H	Sampling clock select register	SCS	R/W	—	x	—	00H
FF48H	16-bit timer mode control register	TMC0	R/W	x	x	—	00H

3f

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF49H	8-bit timer mode control register	TMC1	R/W	x	x	—	00H
FF4AH	Watch timer mode control register	TMC2	R/W	x	x	—	00H
FF4CH	Capture/compare control regiter 0	CRC0	R/W	x	x	—	04H
FF4EH	16-bit timer output control register	TOC0	R/W	x	x	—	00H
FF4FH	8-bit timer output control register	TOC1	R/W	x	x	—	00H
FF60H	Serial operating mode register 0	CSIM0	R/W	x	x	—	00H
FF61H	Serial bus interface control register	SBIC	R/W	x	x	—	00H
FF62H	Slave address register	SVA	R/W	—	x	—	Undefined
FF63H	Interrupt timing specify register	SINT	R/W	x	x	—	00H
FF68H	Serial operation mode register 1	CSIM1	R/W	x	x	—	00H
FF69H	Automatic data transmit/receive control register	ADTC	R/W	x	x	—	00H
FF6AH	Automatic data transmit/receive address pointer register	ADTP	R/W	—	x	—	00H
FF6BH	Automatic data transmission/reception interval specification register	ADT1	R/W	x	x	—	00H
FF70H	Asynchronous serial interface mode register	ASIM	R/W	x	x	—	00H
FF71H	Asynchronous serial interface status register	ASIS	R	x	x	—	00H
FF72H	Serial interface operating mode register 2	CSIM2	R/W	x	x	—	00H
FF73H	Baud rate generator control register	BRGC	R/W	—	x	—	00H
FF74H (Note 1)	Transmit shift register Serial I/O shift register Receive buffer register	TXS SI02 RXB	W R/W R	— — —	x x x	— — —	FFH FFH FFH
FF80H	A/D converter mode register	ADM	R/W	x	x	—	01H
FF84H	A/D converter input select register	ADIS	R/W	—	x	—	00H
FF90H	D/A converter data register 0	DACS0	R/W	—	x	—	00H
FF91H	D/A converter data register 1	DACS1	R/W	—	x	—	00H
FF98H	D/A converter mode register	DAM	R/W	x	x	—	00H
FFD0H-FFDFH	External SFR access area (Note 2)	—	R/W	x	x	—	Undefined
FFE0H	Interrupt flag register L	IF0L	R/W	x	x	—	00H
FFE1H	Interrupt flag register H	IF0H	R/W	x	x	—	00H
FFE0H-FFE1H	Interrupt flag register	IF0	R/W	—	—	x	0000H
FFE2H	Interrupt flag register 1L	IF1L	R/W	x	x	—	00H
FFE4H	Interrupt mask flag register L	MK0L	R/W	x	x	—	FFH
FFE5H	Interrupt mask flag register H	MK0H	R/W	x	x	—	FFH
FFE4H-FFE5H	Interrrupt mask flag register	MK0	R/W	—	—	x	FFFFH
FFE6H	Interrupt mask flag register 1L	MK1L	R/W	x	x	—	FFH
FFE8H	Priority order specify flag register L	PR0L	R/W	x	x	—	FFH
FFE9H	Priority order specify flag register H	PR0H	R/W	x	x	—	FFH
FFE8H-FFE9H	Priority order specify flag register	PR0	R/W	—	—	x	FFFFH
FFEAH	Priority order specify flag register 1L	PR1L	R/W	x	x	—	FFH

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FFECH	External interrupt mode register 0	INTM0	R/W	—	x	—	00H
FFEDH	External interrupt mode register 1	INTM1	R/W	—	x	—	00H
FFF0H	Memory size switch register	IMS	W	—	x	—	(Note 3)
FFF2H	Oscillation mode select register	OSMS	R/W	x	x	—	00H
FFF3H	Pullup resistor option register H	PU0H	R/W	x	x	—	00H
FFF6H	Key return mode register	KRM	R/W	x	x	—	02H
FFF7H	Pullup resistor option register L	PU0L	R/W	x	x	—	00H
FFF8H	Memory expanded mode register	MM	R/W	x	x	—	10H
FFF9H	Watchdog timer mode register	WDTM	R/W	x	x	—	00H
FFFAH	Oscillation stabilization time select register	OSTS	R/W	—	x	—	04H
FFFBH	Processor clock control register	PCC	R/W	x	x	—	04H

Notes:

(1) SIO2 can be used instead of TXS and RXB. SIO2 is not a register; it is another symbol which can be used to reference the TXS and RXB registers. A write to SIO2 causes the CPU to write to the TXS register, and a read from SIO2 causes the CPU to read the RXB register.

(2) The external access area cannot be accessed using SFR addressing. It can only be accessed using main memory addressing.

(3) Value after reset depends on device
μPD78052: 44H
μPD78053: C6H
μPD78054: C8H
μPD78P054: C8H.

3f

Table 2. Digital Port Functions

Port	Operational Features	Configuration	Direct Drive Capability	Software Pullup Resistor Connection (Note 1)
Port 0 (Note 3)	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 1 (Note 2)	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 2	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 3	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 4	8-bit input or output	Byte selectable		Byte selectable, input bits only
Port 5	8-bit input or output	Bit selectable	LED	Byte selectable, input bits only
Port 6	8-bit input or output ($P6_0$ - $P6_3$ n-channel)	Bit selectable	LED ($P6_0$-$P6_3$)	Byte selectable, input bits only $P6_0$ - $P6_3$ - mask option only (Note 4) $P6_4$ - $P6_7$ - software
Port 7	3-bit input or output	Bit selectable		Byte selectable, input bits only
Port 12	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 13	2-bit input or output	Bit selectable		Byte selectable, input bits only

Notes:

(1) Software pullup resistors can be internally connected (only on a port-by-port basis) to port bits set to input mode. Pullup resistors are not connected to port bits set to output mode.

(2) Pullup resistors are automatically disconnected on pins used for A/D converter analog inputs.

(3) $P0_0$ and $P0_7$ are input only and do not have a software pullup resistor. When using $P0_7$ as an input, the feedback resistor for the subsystem clock should be disconnected with bit 6 (FRC) of the processor control register (PCC).

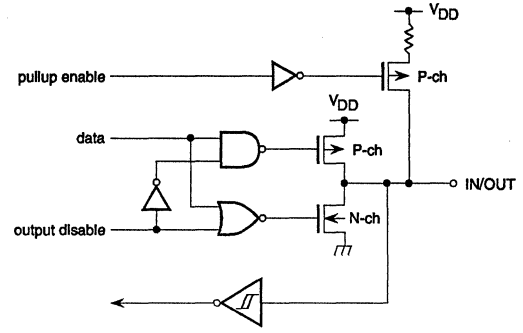
(4) All devices except μPD78P054

Figure 4. Pin Input/Output Circuits

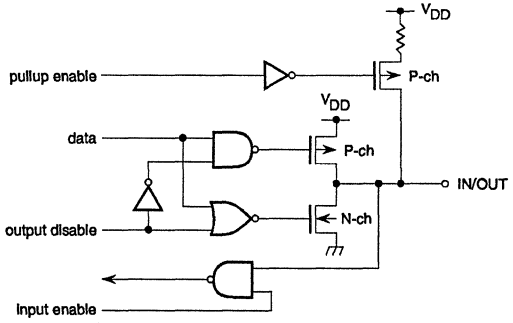
Type 2 ($P0_0$, $\overline{RESET}$)

Schmitt trigger input with hysteresis characteristics

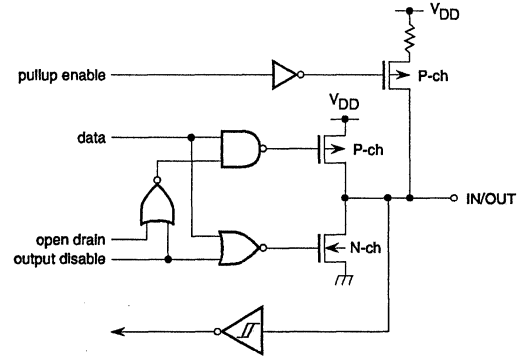
Type 8-A ($P0_1$ - $P0_6$, $P2_0$, $P2_2$, $P2_4$, $P3_3$ - $P3_4$, $P7_0$, $P7_2$)



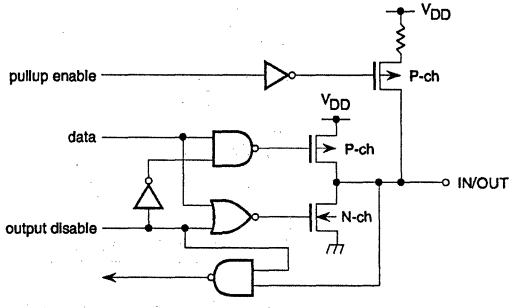
Type 5-A ($P2_1$, $P2_3$, $P3_0$ - $P3_2$, $P3_5$ - $P3_7$, $P5_0$ - $P5_7$, $P6_4$ - $P6_7$, $P7_1$, $P12_0$ - $P12_7$)



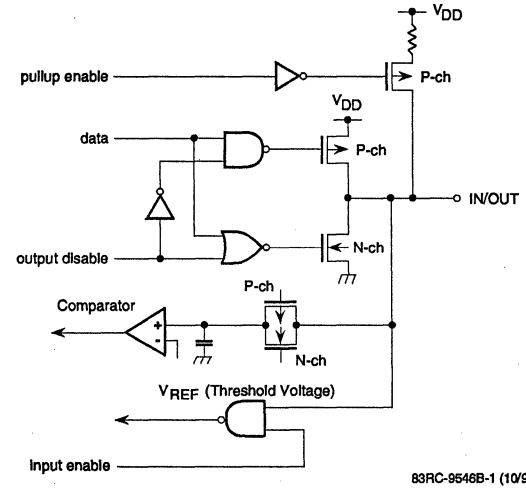
Type 10-A ($P2_5$ - $P2_7$)



Type 5-E ($P4_0$ - $P4_7$)



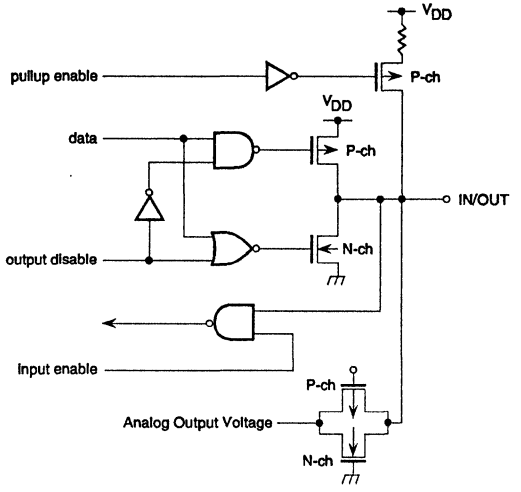
Type 11 ($P1_0$ - $P1_7$)



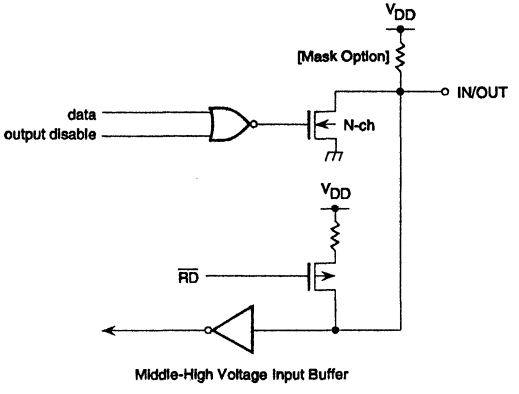
83RC-9546B-1 (10/93)

Figure 4. Pin Input/Output Circuits (cont)

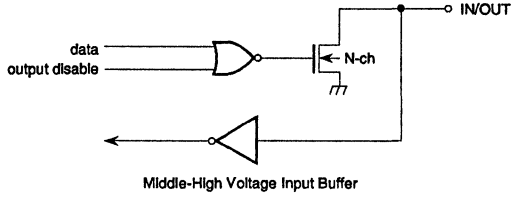
Type 12-A ($P13_0$ - $P13_1$)



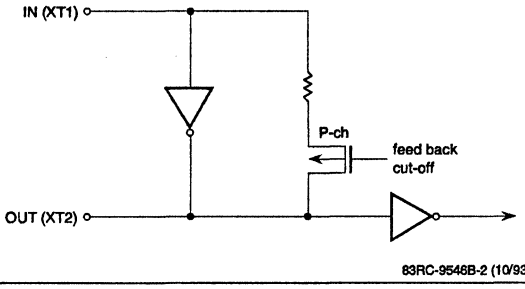
Type 13-B ($P6_0$ - $P6_3$: μPD7805x)



Type 13 ($P6_0$ - $P6_3$: μPD78P054)



Type 16 ($P0_7$, XT2)



83RC-9546B-2 (10/93)

3f

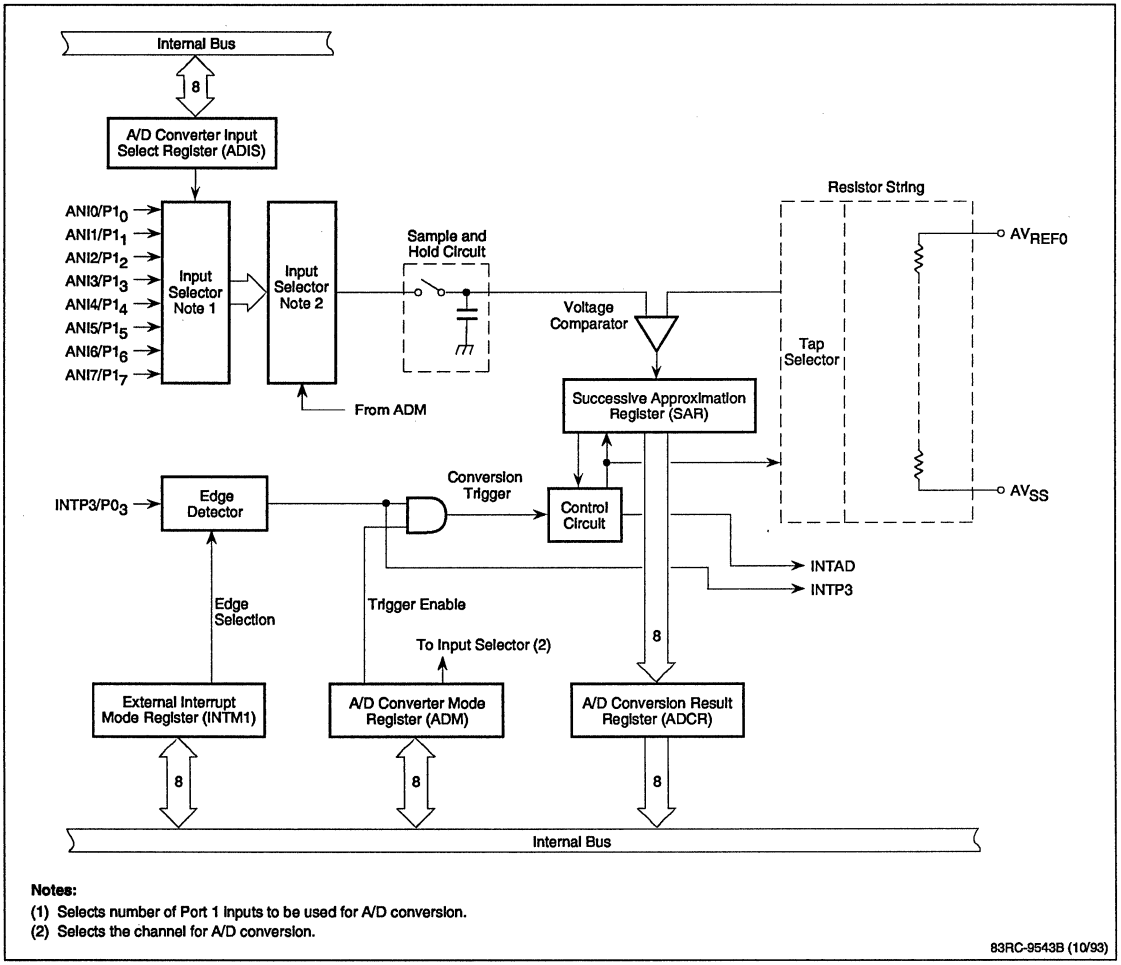
A/D Converter

The μPD78054 family analog-to-digital (A/D) converter (see figure 5) uses the successive-approximation method for converting one of eight multiplexed analog inputs into 8-bit digital data. The conversion time per input is 19.1 μs.

The A/D converter input select register (ADIS) selects the number of inputs that are used in A/D conversion. The remaining inputs are used as ports. The analog input to be converted is selected by programming the A/D converter mode register (ADM). Also, the ADM register is used to select the A/D conversion time. A/D conversion is started by external interrupt INTP3, or by writing to the ADM register. When the conversion is completed, the results are stored in the A/D conversion result register (ADCR) and an INTAD interrupt is generated.

If the A/D converter was started by an external interrupt, the A/D converter stops after the interrupt is generated. If the A/D converter was started by software, the A/D converter repeats the conversion until new data is written to the ADM register.

Figure 5. A/D Converter



Notes:
(1) Selects number of Port 1 inputs to be used for A/D conversion.
(2) Selects the channel for A/D conversion.

83RC-9543B (10/93)

D/A Converter

The μPD78054 family digital-to-analog (D/A) converter (see figure 6) uses an R-2R resistor ladder method for converting the 8-bit digital data to an analog voltage for each of the two independent D/A channels.

The D/A converter mode register (DAM) is used to start and stop D/A conversion as well as selecting "normal mode" or "real-time output mode" for each channel. The 8-bit data to be converted to an analog voltage is written into one of the two D/A conversion value set registers (DACS0 or DACS1). Before D/A conversion is enabled, the respective channel output is in a high-impedance state. The analog output voltage is determined by the following expression where n is 0 or 1 for the respective channel:

$$\text{ANOn (output voltage)} = AV_{REF1} \times \frac{\text{DACSn}}{256} \text{ (volts)}$$

In the normal mode with D/A conversion enabled, the D/A circuit continuously outputs the analog voltage on the ANOn pin representative of the 8-bit value in the DACSn register. When a different 8-bit value is written to the DACSn register, the D/A circuit immediately adjusts the voltage on the ANOn pin to represent the new value. When the D/A conversion operation stops, the output for the respective channel goes to high impedance.

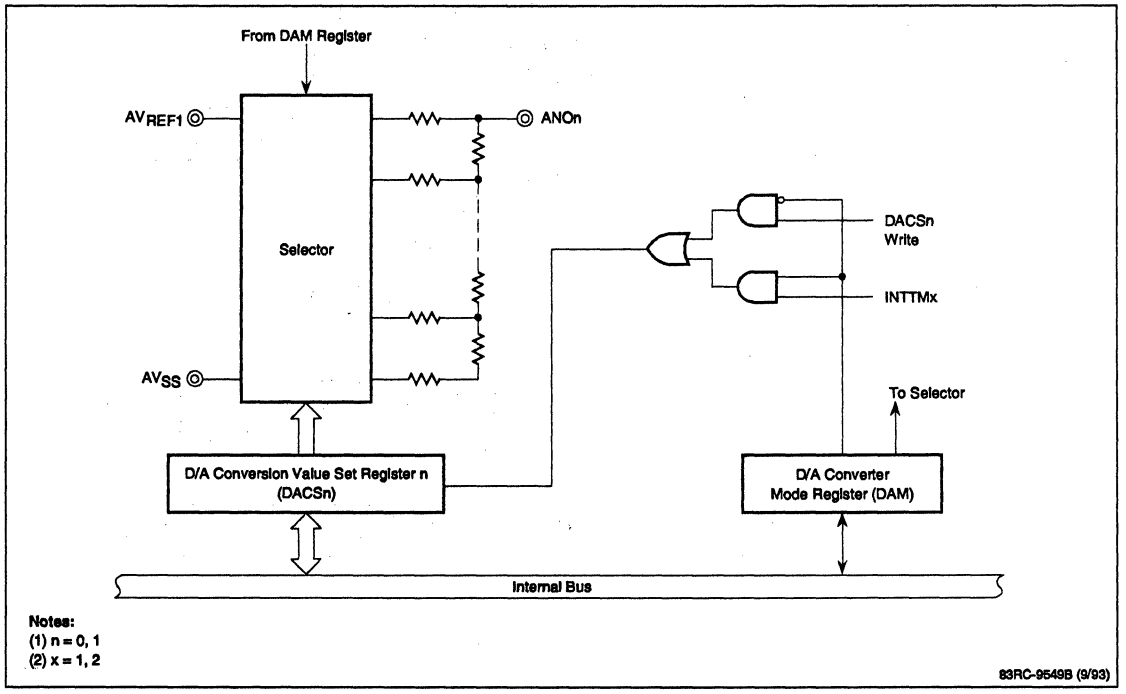
In the real-time output mode when D/A conversion is enabled for a respective channel, the D/A circuit output remains in a high-impedance state. After the interrupt (INTTMx, x = 1 and 2 for D/A channels 0 and 1, respectively) occurs, the D/A circuit outputs the analog voltage on the ANOn pin representative of the 8-bit value in the DACSn register.

When a different 8-bit value is written to the DACSn register, the D/A circuit will adjust the voltage on the ANOn pin to represent the new value after the next interrupt (INTTMx) occurs.

If the data is changed in the DACSn register and the register is read before the next interrupt occurs, the data read will be data previously written before the last interrupt occurred. When the D/A conversion operation stops, the output for the respective channel goes to a high-impedance state.

3f

Figure 6. D/A Converter



Notes:
(1) n = 0, 1
(2) x = 1, 2

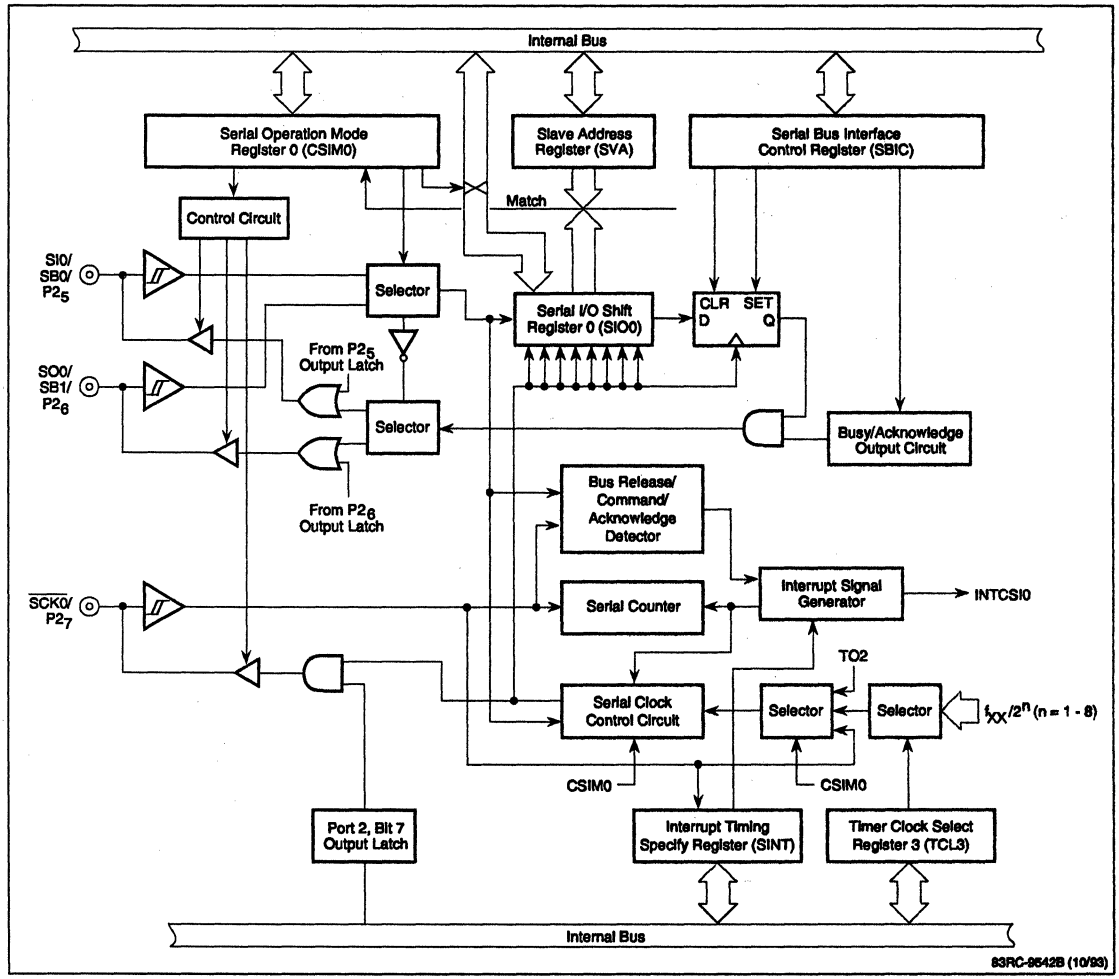
83RC-9549B (9/93)

Serial Interfaces

The μPD78054 family has three independent serial interfaces: serial interface 0, serial interface 1, and serial interface 2.

Serial Interface 0. Serial interface 0 is an 8-bit clock synchronous serial interface (figure 7). It can be operated in either a three-wire serial I/O mode, NEC serial bus interface (SBI) mode, or two-wire serial I/O mode. The serial clock can be provided from one of eight internal clocks, the output of 8-bit timer register 2, or the external clock line $\overline{SCK0}$.

Figure 7. Serial Interface 0



In the three-wire serial I/O mode, the 8-bit shift register (SIO0) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO0 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI0 line providing full-duplex operation. The INTCSI0 interrupt is generated after each 8-bit transfer.

The NEC SBI mode is a two-wire high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are connected in a master/slave configuration (see figure 8). There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over one of the serial bus lines (SB0 or SB1) using a fixed hardware protocol synchronized with the $\overline{SCK0}$ line . Each slave device of the μPD78054 family can be programmed to respond in hardware to any one of 256 addresses set in its slave address register (SVA). There are also 256 commands and 256 data types. Since all commands are user definable, many software protocols, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

The two-wire serial I/O mode provides half-duplex operation using either the SB0 or SB1 line and the $\overline{SCK0}$ line. Communication format and handshaking can be handled in software by controlling the output levels of the data and clock lines between transfers. For data transmission, the 8-bit shift register (SIO0) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of either the SB0 or SB1 line (MSB first). In addition, this byte of data is also shifted back into SIO0 on the rising edge of these pulses providing a way of verifying that the transmission was correct.

For data reception, the SIO0 register is preloaded with the value FFH. As this data value is shifted out on the falling edge of the serial clock, it disables the n-channel open-drain driver. This allows the receive data to be driven on to the serial line and shifted into the SIO0 register on the rising edge of the serial clock. The INTCSI0 interrupt is generated after each 8-bit transfer.

Serial Interface 1. Serial interface 1 is also an 8-bit clock synchronous serial interface (figure 9). It can be operated in either a three-wire serial I/O mode, or three-wire serial I/O mode with automatic transmit/receive. The serial clock can also be provided from one of eight internal clocks (common clock for both interfaces), the output of 8-bit timer register 2, or the external clock line $\overline{SCK1}$.

In the three-wire serial I/O mode, the 8-bit shift register (SIO1) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO1 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI1 line providing full-duplex operation. The INTCSI1 interrupt is generated after each 8-bit transfer.

In the three-wire serial I/O mode with automatic transmit/receive, up to 32 bytes of data can be transferred with minimal CPU overhead. The data to be transmitted and received is stored in the buffer RAM. Handshaking using either the BUSY input line, the strobe (STB) output line, or both, can be selected by the program. Error detection of bit drift due to noise is available for each byte transferred when using the BUSY input line. This automatic transmit/receive mode is ideally suited for transferring data to/from external peripheral devices such as onscreen display (OSD) and LCD controller/driver devices.

3f

While in three-wire serial interface mode with automatic data transfer, the interface can be operated as either a full-duplex interface or a transmit-only interface in single or repetitive operation mode. In the full-duplex mode, a byte of data is transferred from the first location in the buffer RAM and shifted out of the SO1 line (either MSB or LSB first) while the received data is shifted into the SI1 line and stored back in the first buffer location. After the preset number of bytes has been transferred, the INTCSI1 interrupt is generated.

Figure 8. SBI Mode Master/Slave Configuration

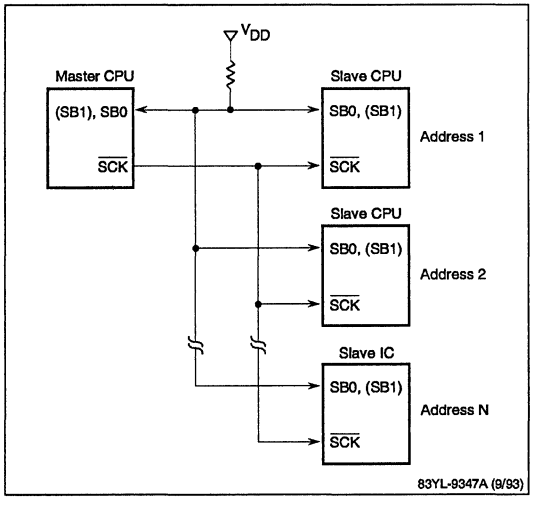
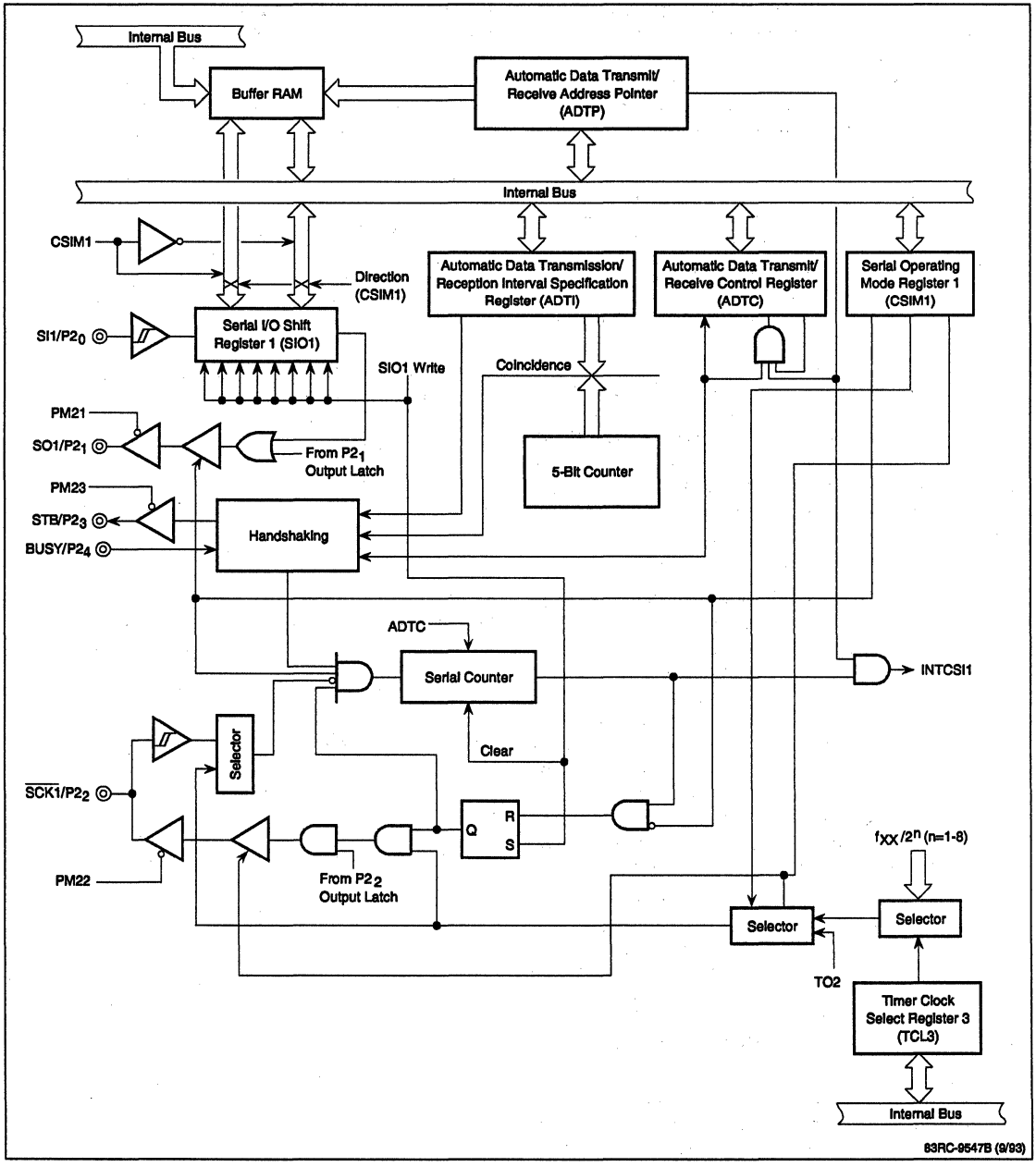


Figure 9. Serial Interface 1



In single-operation transmit mode, the preset number of bytes from the buffer RAM are transmitted out of the SO1 line (either MSB or LSB first) and the INTCSI1 interrupt is generated after all bytes are transferred. In the repetitive operation transmit mode, data in the buffer is transmitted repeatedly.

Serial Interface 2. Serial interface 2 is an 8-bit serial interface (figure 10) that can be operated in either a UART mode or a three-wire serial I/O mode. The internal baud rate generator circuit (figure 11) scales an internal clock to provide standard baud rates from 75 to 38400 bps. For non-standard baud rates, the internal baud rate generator circuit scales an external clock input on the ASCK pin. The output of the baud rate generator circuit is used as the data transmission and sampling clock in the UART mode or the data clock in the three-wire serial I/O mode.

In the UART mode, half or full-duplex operation with various protocols is programmable. The asynchronous serial interface mode register (ASIM) is used to specify the number of stop bits (1 or 2), data character length (7 or 8 bits), parity (none, even, odd, or 0), receive operation control (enable or disable), and transmit operation control (enable or disable). The ASIM register is also used to enable or disable the generation of an interrupt when a reception error occurs and whether an internal or external clock will be supplied to the baud rate generator circuit.

A transmit operation is started by writing a data character to the transmit shift register (TXS) register. The start bit, parity bit, and stop bit(s) are automatically added by the hardware to the data character in the TXS register. The data in the TXS register is shifted out of the TxD line and when the TXS register is empty, a transmission complete interrupt (INTST) is generated.

When the receive operation control is enabled, the RxD line is sampled using the clock specified by the ASIM register. When the RxD line is detected low, sampling starts at the midpoint of each bit. If the first sample yields a low, it is identified as a start bit. The RxD line continues to be sampled at the midpoint of each bit. Reception of one frame of data is complete when the data character bits, parity bit (if being transmitted), and one stop bit are detected after the start bit. Even if the protocol is set for two stop bits, only one stop bit is used for the end of reception detection.

When one frame of data has been received, the received data in the shift register is transferred to the receive buffer (RXB) and a reception complete vectored interrupt (INTSR) is generated even if an error (parity and/or framing) is detected. The data must be read from the RXB register before another frame is received or an overrun error will be generated. If an error occurs, the appropriate flag is set in the asynchronous serial interface status register (ASIS) and a receive error interrupt (INTSER) is generated.

In the three-wire I/O mode, the TXS register is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO2 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI2 into the RXS register line providing full-duplex operation. The INTCSI2 interrupt is generated after each 8-bit transfer.

Timers

The μPD78054 family has one 16-bit timer/event counter, two 8-bit timer/event counters that can be combined for use as a 16-bit timer/event counter, a watch timer and a watchdog timer. All of these can be programmed to count a number of prescaled values of the main system clock. In addition, the watch timer can also count the subsystem clock. All of the timer/event counters can count external events.

16-Bit Timer/Event Counter 0. Timer/event counter 0 (figure 12) consists of a 16-bit counter (TM0), two 16-bit capture registers (CR00, CR01), control registers TMC0, TOC0, and CRC0, clock select register CLC0, and a timer output (TO0). Timer 0 can be used as an interval timer, to count external events on the timer input (TI00) pin, to output a programmable square wave, a 14-bit pulse-width modulated output, a one-shot pulse, or to measure pulse widths.

Figure 10. Serial Interface 2

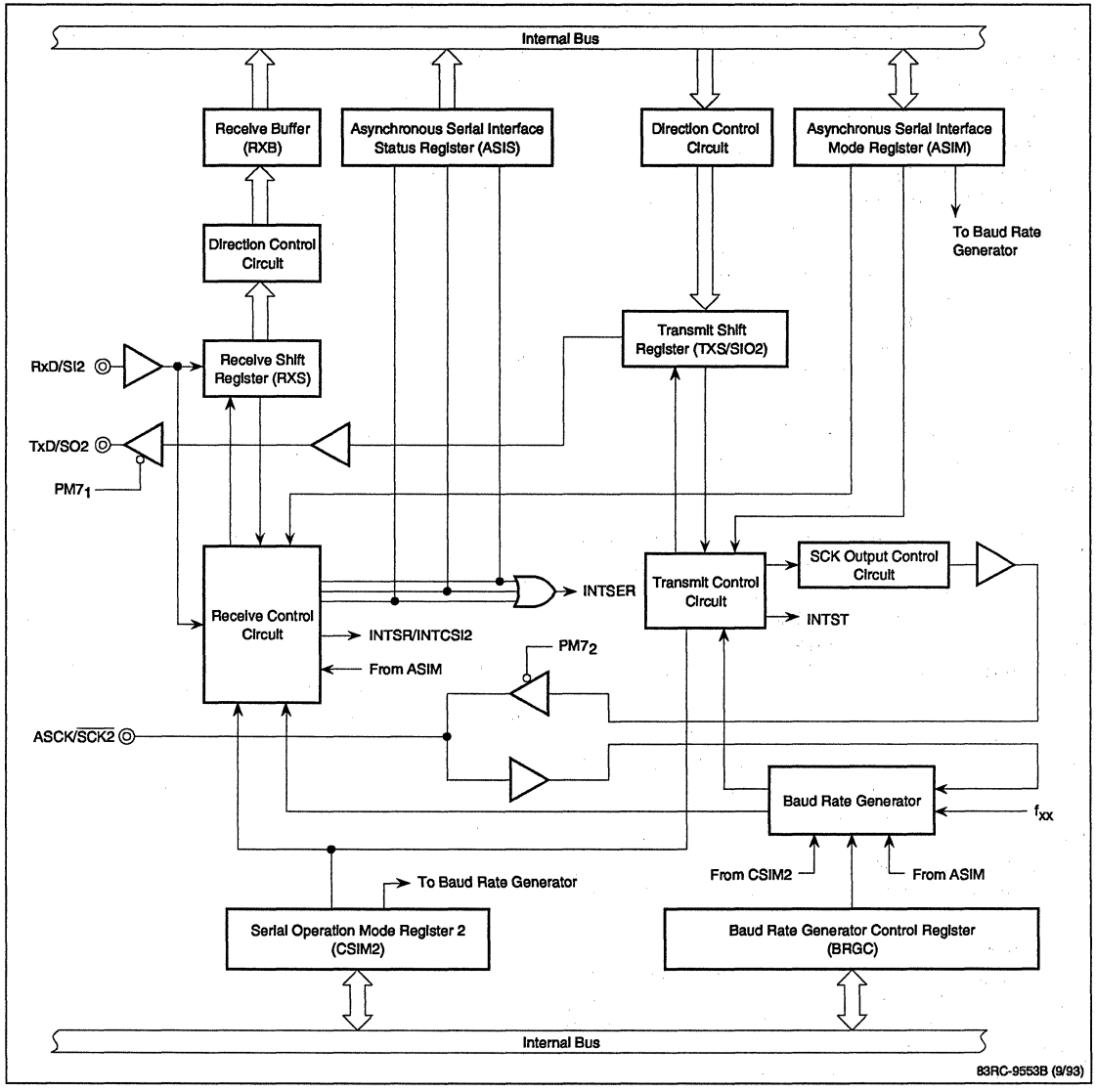
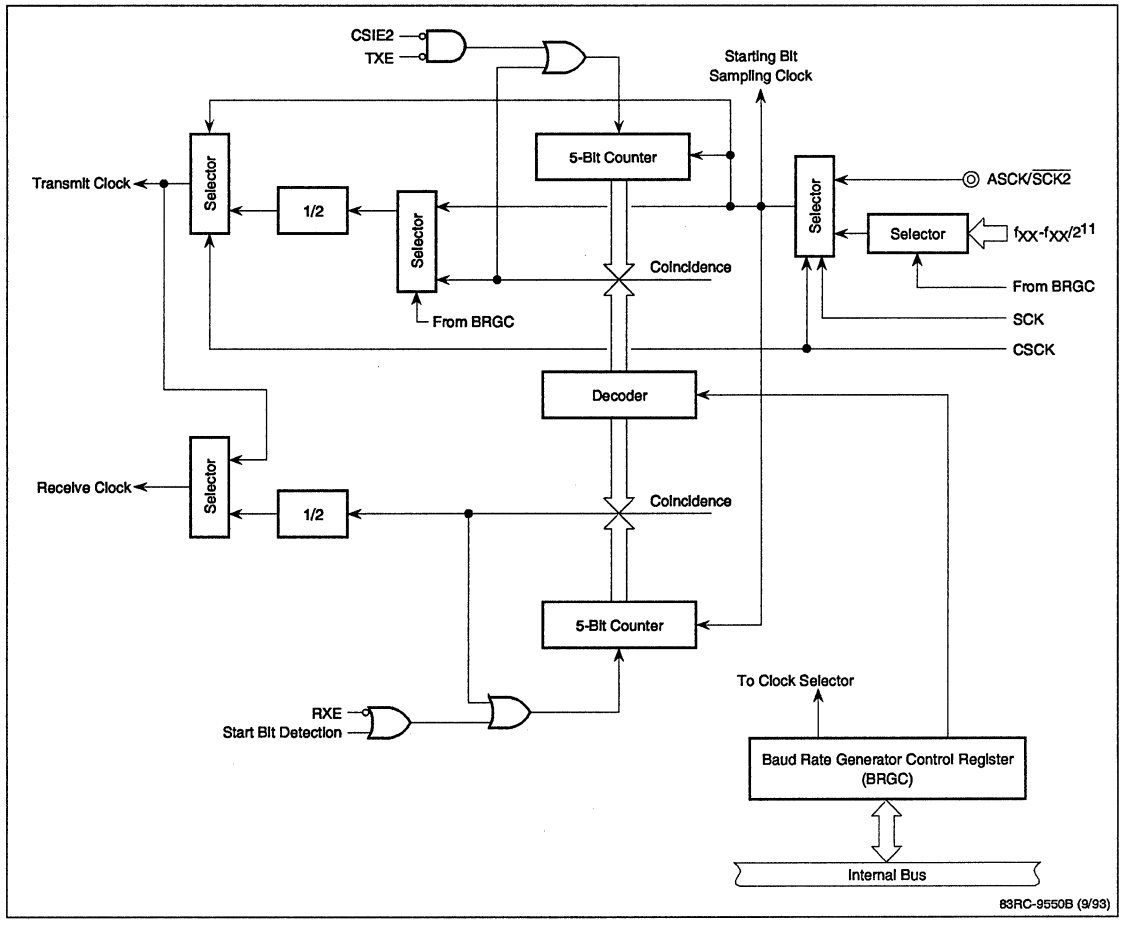
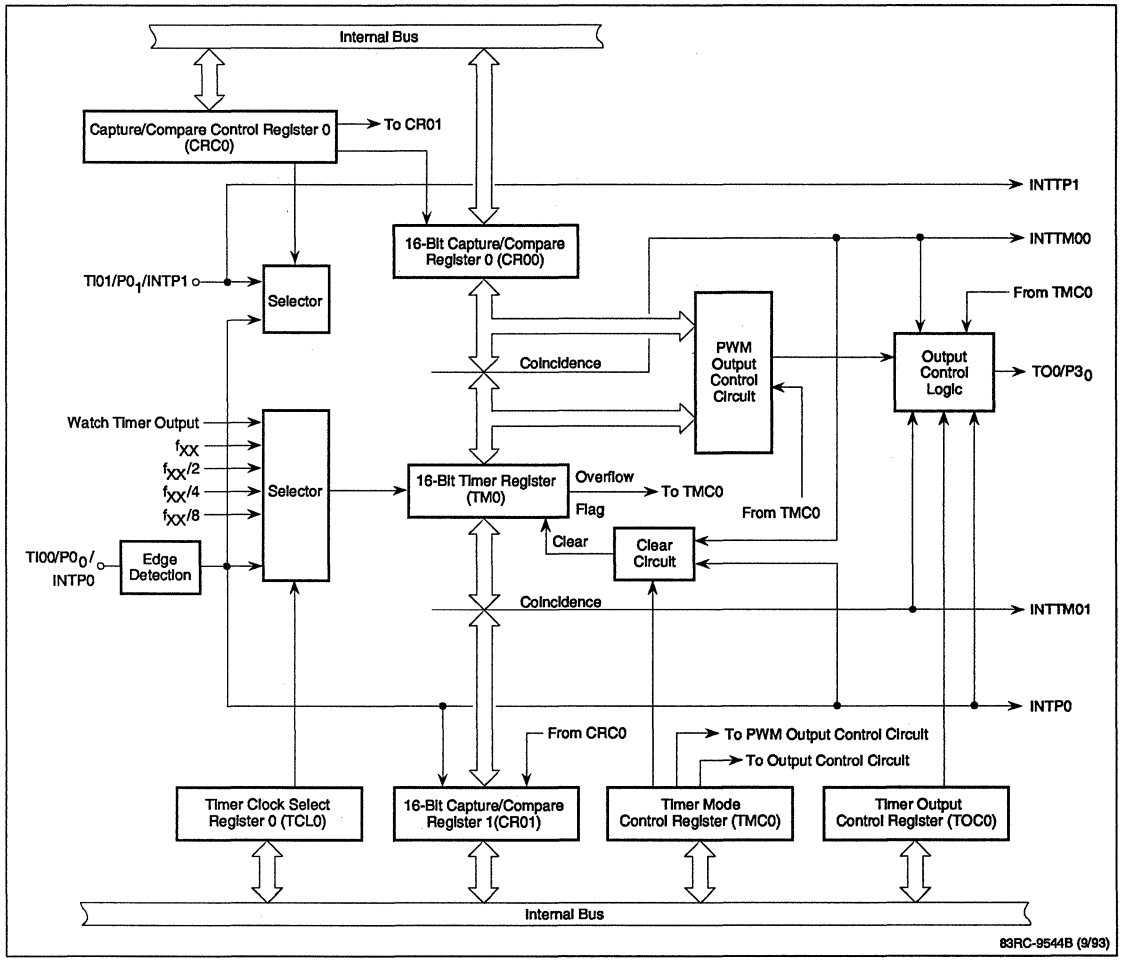


Figure 11. Internal Baud Rate Generator



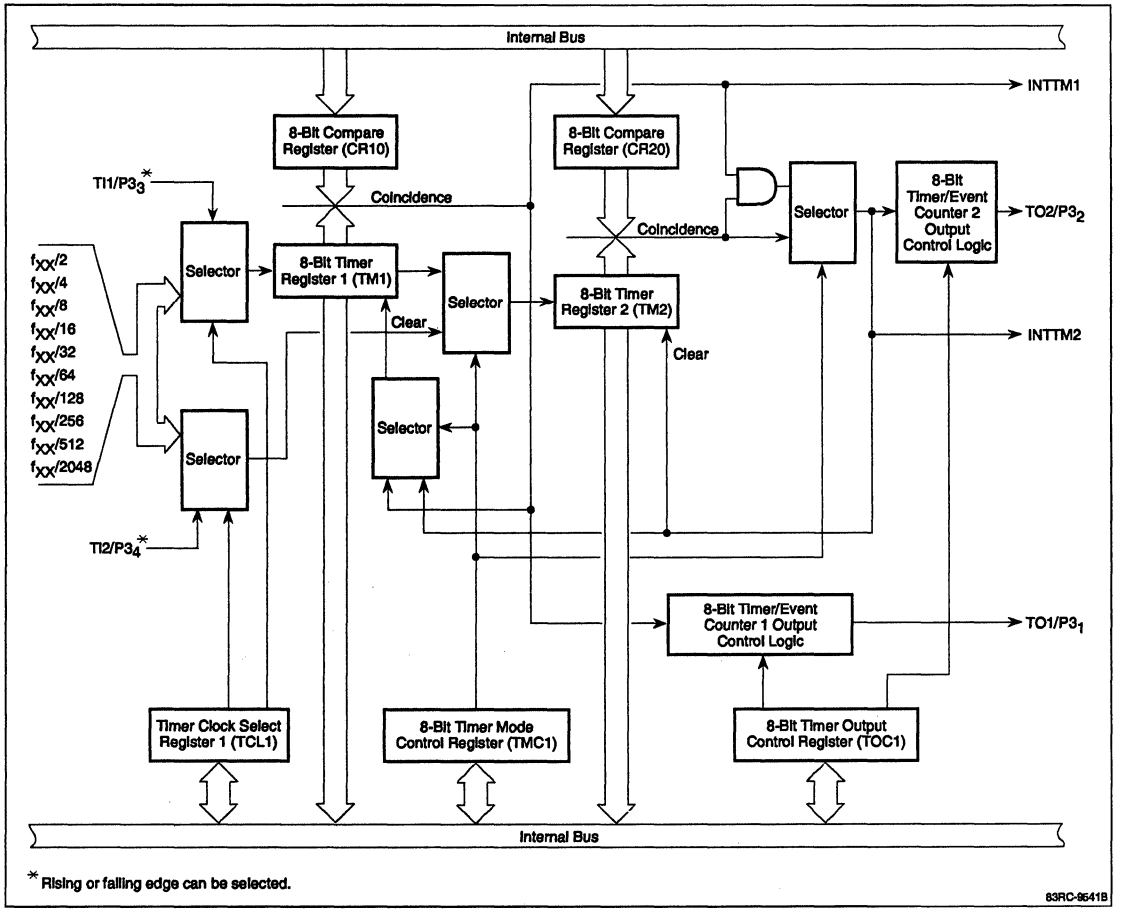
3f

Figure 12. 16-Bit Timer/Event Counter 0



8-Bit Timer/Event Counters 1 and 2. Timer/event counters 1 and 2 (figure 13) each consist of an 8-bit timer (TM1 or TM2), an 8-bit compare register (CR10 or CR20), and a timer output (TO1 or TO2). The timers are controlled by registers TCL1, TMC1, TOC1 via five selectors. Timer/event counters 1 and 2 can each be used as an 8-bit interval timer, to count external events on the timer input pins (TI1 or TI2), or to output a programmable square wave. In addition, timers 1 and 2 also can be combined as a 16-bit timer/event counter and used as a 16-bit interval timer, to count external events on TI1, or to output a programmable square wave on TO2.

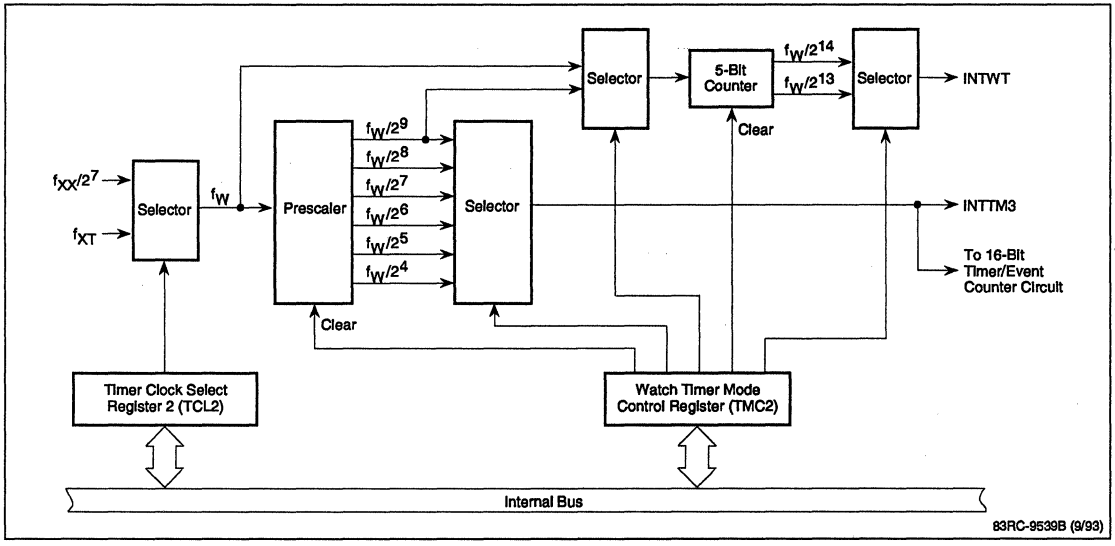
Figure 13. 8-Bit Timer/Event Counters 1 and 2



Watch Timer 3. Watch timer 3 (figure 14) is a 5-bit timer that can be used as a time source to keep track of time of day, to release the STOP or HALT modes at regular intervals, or to initiate any other task that must be performed at regular intervals. When driven by the subsystem clock, the watch timer continues to operate in the STOP mode.

The watch timer can function as both a watch timer and an interval timer simultaneously. When used as a watch timer, interrupt request INTWT can be generated using the main or subsystem clock of 0.5 or 0.25 seconds. When used as an interval timer, vectored interrupt request INTTM3 is generated at preselected time intervals. With a main system clock of 4.19 MHz and $f_{XX} = f_X$ or if using the subsystem clock of 32.768 kHz, the following time intervals can be selected: 489μ s, 978 μs, 1.96 ms, 3.91 ms, 7.82 ms or 15.6 ms.

Figure 14. Watch Timer 3



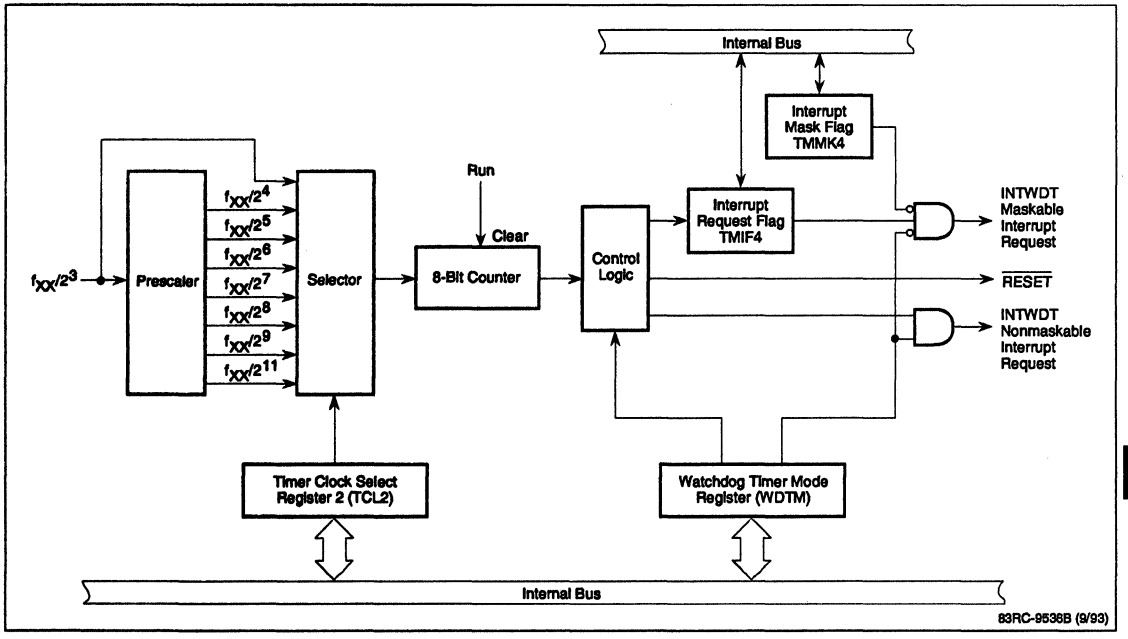
Watchdog Timer. The watchdog timer (figure 15) can be used as either a watchdog timer or an interval timer. When used as a watchdog timer it protects against program runaway. It can be selected to generate a nonmaskable interrupt (INTWDT), which vectors to address 0004H, or to generate an internal reset signal, which vectors to the restart address 0000H if the timer is not cleared by the program before it overflows. Eight program-selectable intervals based on the main system clock are available. With a main system clock of 4.19 MHz and $f_{XX} = f_X$, they are 0.489, 0.978, 1.96, 3.91, 7.82, 15.6, 31.3, and 125 ms. With a main system clock of 4.19 MHz and $f_{XX} = f_X/2$, they are 0.978, 1.96, 3.91, 7.82, 15.6, 31.3, 62.6, and 250 ms. Once the watchdog timer is initialized and started, the timer's mode cannot be changed and the timer can only be stopped by a reset.

When used as an interval timer, maskable interrupts (INTWDT) which vector to address 0004H are generated repeatedly at a preset interval. The time intervals available are the same as in the watchdog timer mode.

Programmable Clock Output

The μPD78054 family has a programmable clock output (PCL) that can be used for carrier output for remote controlled transmissions or as a clock output for peripheral devices. The main system clock (f_{XX}) divided by 1, 2, 4, 8, 16, 32, 64, or 128 or the subsystem clock (f_{XT}) can be output on the PCL pin. See figure 16. If the main system clock is 4.19 MHz and $f_{XX} = f_X$, the following frequencies are available: 4.19 MHz, 2.1 MHz, 1.05 MHz, 524 kHz, 262 kHz, 131 kHz, 65.5 kHz, and 32.7 kHz. With a main system clock of 4.19 MHz and $f_{XX} = f_X/2$, the following frequencies are available: 2.1 MHz, 1.05 MHz, 524 kHz, 262 kHz, 131 kHz, 65.5 kHz, 32.7 kHz and 16.4 kHz. With a subsystem clock of 32.768 kHz, 32.768 kHz is also available.

Figure 15. Watchdog Timer



3f

Figure 16. Programmable Clock Output

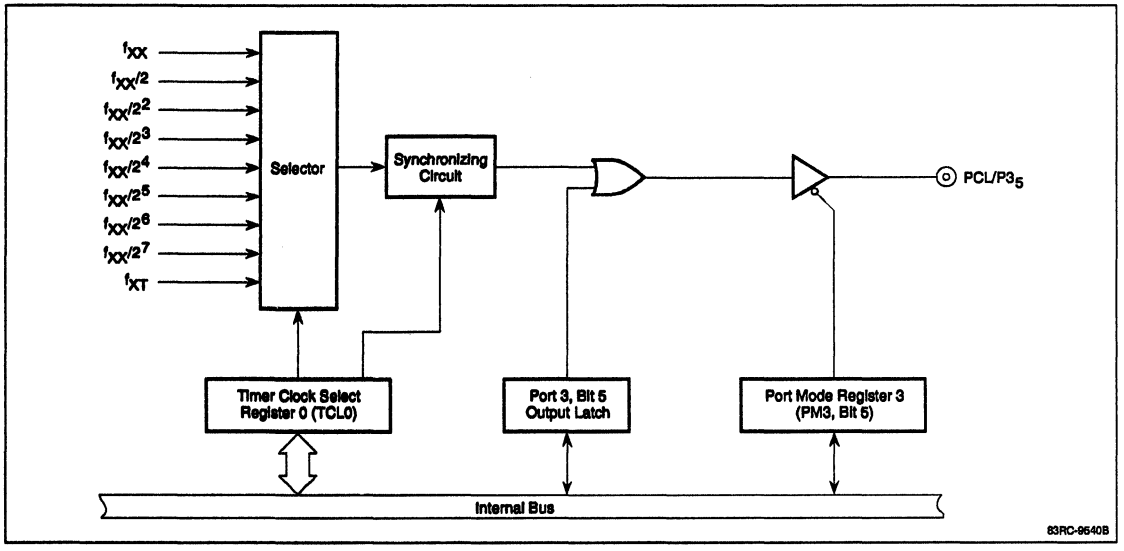


Figure 17. Buzzer Output

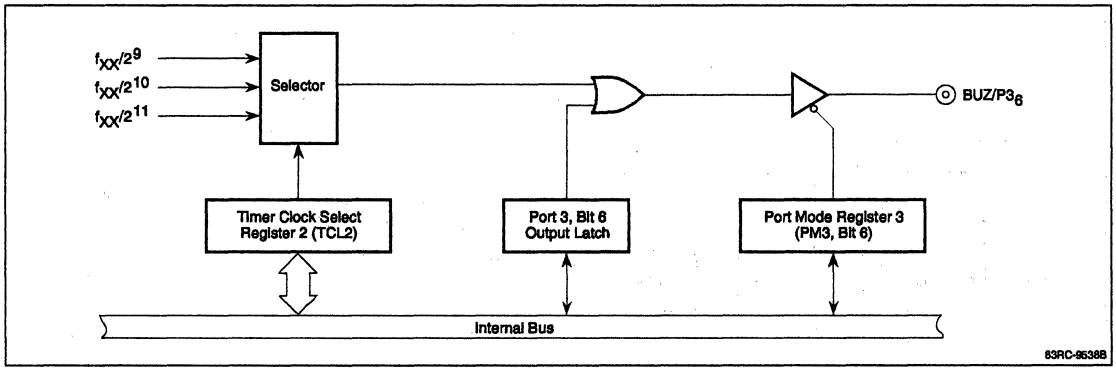
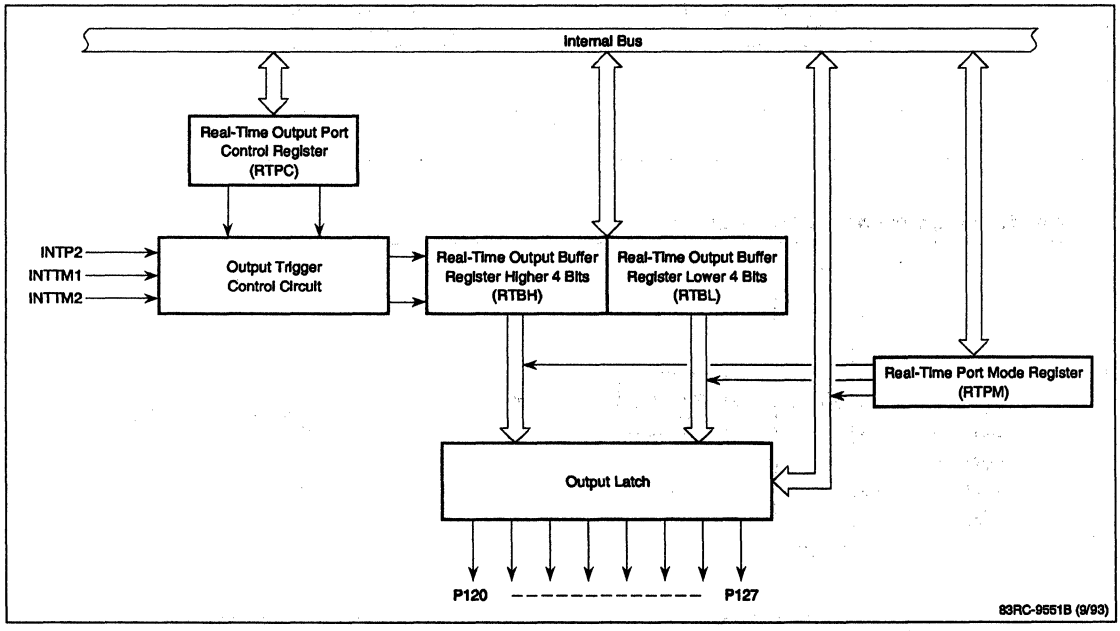


Figure 18. Real-Time Output Port



Buzzer Output

The μPD78054 family also has a programmable buzzer output (BUZ). The buzzer output frequency can be programmed to be equal to the main system clock (f_{XX}) divided by 512, 1024, or 2048. With a main system clock of 4.19 MHz and $f_{XX} = f_X$, the buzzer can be set to 8.2, 4.1 or 2.0 kHz. With a main system clock of 4.19 MHz and $f_{XX} = f_X/2$, the buzzer can be set to 4.1, 2.0, or 1.0 kHz. See figure 17.

Real-Time Output Port

The real-time output port (figure 18) shares pins with port 12. Each bit of port 12 is specified by the real-time output port mode register (RTPM) to be used in the port mode or real-time output port mode. If the real-time output port mode is selected, the real-time output port control register (RTPC) is used to specify the high and low nibbles to be treated separately or together. In the real-time output port mode, the previously written data in the real-time output buffer registers (RTBH, RTBL) is transferred to the output latch simultaneously with the generation of either a timer interrupt (INTTM1, INTTM2) or external interrupt (INTP2).

Interrupts

The μPD78054 family has 21 maskable hardware interrupt sources; 8 are external and 13 are internal. Of these 21 interrupt sources, 19 cause a vectored interrupt while the 2 testable inputs only generate an interrupt request. All of the 21 maskable interrupts can be used to release the HALT mode except INTP0. INTP0 cannot be used to release the STOP mode and cannot release the HALT mode when SCS = 0. In addition, there is one nonmaskable interrupt from the watchdog timer, one software interrupt, and a reset interrupt. The watchdog timer overflow interrupt (interrupt vector table address 0004H) can be initialized to be a nonmaskable interrupt or the highest default priority maskable interrupt. The software interrupt, generated by the BRK instruction, is not maskable. See table 3 and figure 19.

3f

Table 3. Interrupt Sources and Vector Addresses

Type of Request	Default Priority	Signal Name	Interrupt Source	Location	Vector Address	Interrupt* Configuration
Restart	—	RESET	$\overline{\text{RESET}}$ input pin	External	0000H	—
		INTWDT	Watchdog timer overflow (when reset mode selected)	Internal		
Nonmaskable	—	INTWDT	Watchdog timer overflow (when nonmaskable interrupt selected)	Internal	0004H	A
Maskable	0	INTWDT	Watchdog timer overflow (when interval timer selected)	Internal	0004H	B
	1	INTP0	External interrupt edge detection	External	0006H	C
	2	INTP1	External interrupt edge detection	External	0008H	D
	3	INTP2	External interrupt edge detection	External	000AH	D
	4	INTP3	External interrupt edge detection	External	000CH	D
	5	INTP4	External interrupt edge detection	External	000EH	D
	6	INTP5	External interrupt edge detection	External	0010H	D
	7	INTP6	External interrupt edge detection	External	0012H	D
	8	INTCSI0	End of clocked serial interface 0 transfer	Internal	0014H	B
	9	INTCSI1	End of clocked serial interface 1 transfer	Internal	0016H	B
	10	INTSER	Serial interface 2 UART reception error	Internal	0018H	B
	11	INTSR	End of serial interface 2 UART reception	Internal	001AH	B
		INTCSI2	End of serial interface 2 three-wire transfer			
	12	INTST	End of serial interface 2 UART transmission	Internal	001CH	B
	13	INTTM3	Watch timer reference time interval signal	Internal	001EH	B
	14	INTTM00	16-bit timer/event counter capture/compare (CR00) coincidence signal	Internal	0020H	B

Table 3. Interrupt Sources and Vector Addresses (cont)

Type of Request	Default Priority	Signal Name	Interrupt Source	Location	Vector Address	Interrupt* Configuration
Maskable	15	INTTM01	16-bit timer/event counter capture/compare (CR01) coincidence signal	Internal	0022H	B
	16	INTTM1	8-bit timer/event counter 1 coincidence signal	Internal	0024H	B
	17	INTTM2	8-bit timer/event counter 2 coincidence signal	Internal	0026H	B
	18	INTAD	End of A/D conversion	Internal	0028H	B
Software	—	—	BRK instruction	Internal	003EH	E
Test input	—	INTWT	Watch timer overflow	Internal	—	F
	—	INTPT4	Port 4 falling edge detection	External	—	F

* See figure 19

Interrupt Servicing. The μPD78054 family provides two levels of programmable hardware priority control and services all interrupt requests, except the two testable interrupts (INTWT and INTPT4) using vectored interrupts. The programmer can choose the priority of servicing each maskable interrupt by using the interrupt control registers.

Interrupt Control Registers. The μPD78054 family has three 3-byte interrupt control registers. The interrupt request flag registers (IF0L, IF0H, and IF1L) contain an interrupt request flag for each interrupt. The interrupt mask registers (MK0L, MK0H, and MK1L) are used to enable or disable any individual interrupt. The priority flag registers (PR0L, PR0H, and PR1L) can be used to specify a high or a low priority level for each interrupt except the two testable interrupts (INTWT and INTPT4).

Five other 8-bit registers are associated with interrupt processing. The key return mode register (KRM) contains the KRIF interrupt request flag associated with falling-edge detection on port 4 and the KRMK mask flag used to enable or disable clearing of the standby mode if a falling edge is detected on port 4. The external interrupt mode registers (INTM0 and INTM1) are used to select a rising, falling, or both edges as the valid edge for each of the external interrupts INTP0 to INTP6. The sampling clock select register (SCS) is used to select a sampling clock for the noise eliminator circuit on external interrupt INTP0.

The IE and the ISP bit of the program status word are also used to control interrupts. If the IE bit is 0, all maskable interrupts are disabled. The IE bit can be set or cleared using the EI and DI instructions, respectively, or by directly writing to the PSW. The IE bit is cleared each time an interrupt is accepted. The ISP bit is used by hardware to hold the priority level flag of the interrupt being serviced.

Interrupt Priority. If the watchdog timer overflow interrupt (INTWDT) has been initialized to be a nonmaskable interrupt, it has priority over all other interrupts. Two hardware-controlled priority levels are available for all maskable interrupts that generate a vectored interrupt (i.e., all except the two testable interrupts). Either a high or a low priority level can be assigned by software to each of the maskable interrupts.

Interrupt requests of the same priority or a priority higher than the processor's current priority level are held pending until interrupts in the current service routine are enabled by software or until one instruction has been executed after returning from the current service routine. Interrupt requests of a lower priority are always held pending until one instruction has been executed after returning from the current service routine.

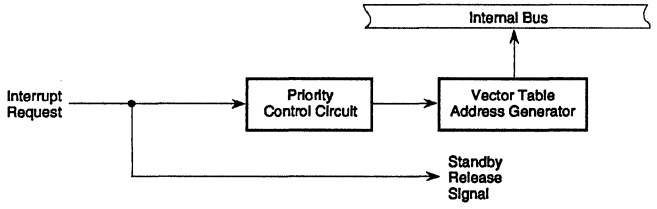
The default priorities listed in table 3 are fixed by hardware and are effective only when it is necessary to choose between two interrupt requests of the same software-assigned priority. For example, the default priorities would be used after the completion of a high-priority routine, if two interrupts of the same software priority were pending.

The software interrupt, initiated by the BRK instruction, is executed regardless of the processor's priority level and the state of the IE bit. It does not alter the processor's priority level.

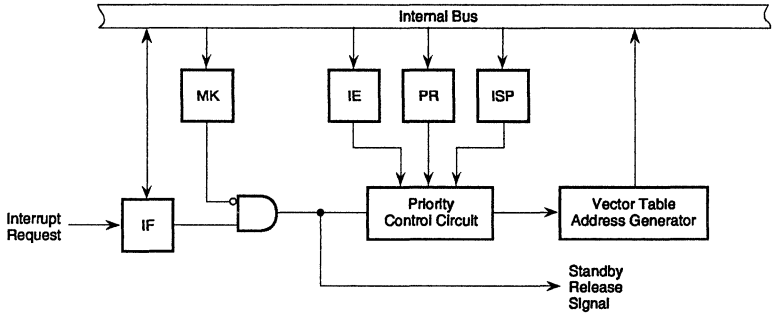
Vectored Interrupt Servicing. When a vectored interrupt is acknowledged, the program status word and the program counter are saved on the stack, the processor's priority is set to that specified for the interrupt, the IE bit in the PSW is set to zero, and the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (RETB instruction for the software interrupt) reverses the process and the μPD78054 family microcomputer resumes the interrupted routine.

Figure 19. Interrupt Configurations

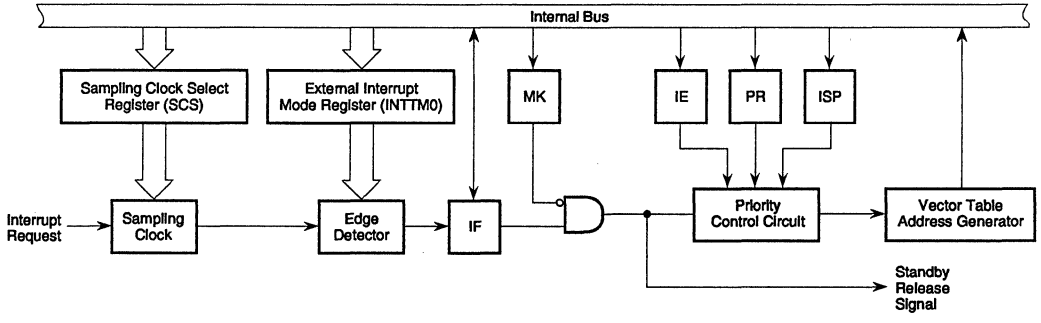
Type A: Internal nonmaskable interrupt



Type B: Internal maskable interrupt



Type C: External maskable interrupt (INTP0)

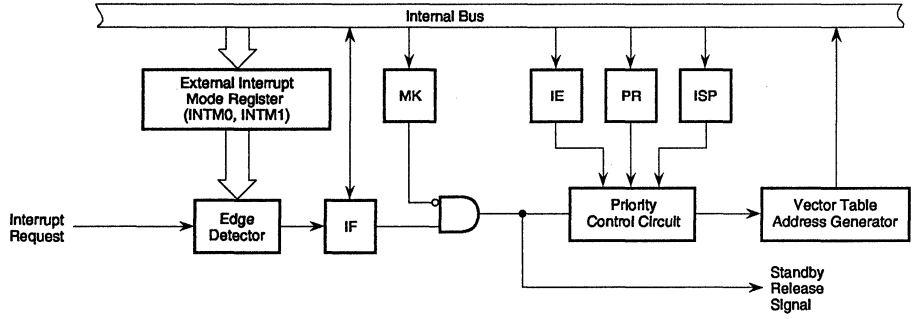


83RC-9591B-1

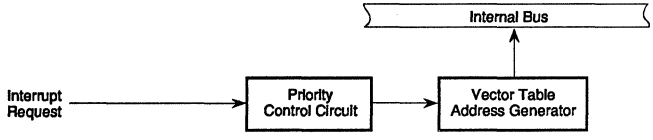
3f

Figure 19. Interrupt Configurations (cont)

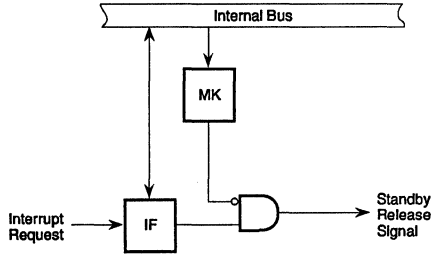
Type D: External maskable interrupt (except INTP0)



Type E: Software Interrupt



Type F: Test Input



Abbreviations:

IF: Interrupt request flag
IE: Interrupt enable flag
ISP: In-service priority flag
MK: Interrupt mask flag
PR: Priority specify flag

83RC-9591B-2

Standby Modes

HALT, STOP, and data retention modes are provided to reduce power consumption when CPU action is not required.

The HALT mode is entered by executing a HALT instruction while the CPU is operating from the main system or subsystem clock. In HALT mode, the CPU clock is stopped while the main system and the subsystem clock continue to run. The HALT mode is released by any unmasked interrupt request, a nonmaskable interrupt request, an unmasked test input, or an external reset pulse.

Power consumption may be further reduced by using the STOP mode. The STOP mode is entered by executing a STOP instruction while operating from the main system clock. In STOP mode, the main system clock input pin X1 is internally grounded stopping both the CPU and the peripheral hardware clock. The STOP mode is released by any unmasked interrupt request (except INTP0 if register SCS = 0), a nonmaskable interrupt request, an unmasked test input, or an external reset pulse. Any peripheral using the main oscillator as a clock source will also be disabled in the STOP mode and interrupts from such a peripheral can not be used to exit the STOP mode. Table 4 summarizes both the HALT and STOP standby modes.

When exiting the STOP mode, a wait time occurs before the CPU begins code execution to allow the main system clock oscillator circuit to stabilize. The oscillation stabilization time is selected by programming the OSTS register with one of five values before entering the STOP mode; the values range from 0.8 msec to 52.4 msec at f_X = 5 MHz.

Table 4. Standby Mode Operation Status

Item	HALT Mode	STOP Mode
Setting instruction	HALT instruction	STOP instruction
System clock when setting	Main system or subsystem clock	Main system clock
Clock oscillator	Main system and subsystem clocks can oscillate; CPU clock is stopped.	Subsystem clock can oscillate; CPU clock and main system clock are stopped.
CPU	Operation stopped	Operation stopped
Ports	Maintain previous state	Maintain previous state
16-bit timer/event counter	Operational from main system clock, or with watch timer output, or TI00 selected as the count clock	Operational only with watch timer output or TI00 selected as count clock.
8-bit timer/event counters	Operational from main system clock or with TI1 and TI2 selected as the count clock	Operational only with TI1 and TI2 selected as count clock
Watch timer	Operational from main system clock or with f_{XT} as count clock	Operational only with f_{XT} as count clock
Watchdog timer	Operational from main system clock	Operation stopped
Serial interface 0	Operational from main system clock or with external clock	Operational only with external clock
Serial interface 1	Operational from main system clock; no automatic transmit/ receive mode	Operational only with external clock; no automatic transmit/ receive mode
Serial interface 2	Operational from main system clock or with external clock	Operation stopped
A/D converter	Operational from main system clock	Operation stopped
D/A converter	Operational	Operational
Real-time output port	Operational	Operational with external trigger or when TI1 and TI2 count clocks are selected
External interrupts	Operational except for INTP0 when its sampling clock is based on the CPU clock	INTP0 not operational; INTP1 to INTP6 operational

3f

Once in the STOP mode, power consumption can be further minimized by lowering the power supply voltage V_{DD} to 2 volts. This places the device in the data retention mode. The contents of Internal RAM and the registers are retained. This mode is released by first raising V_{DD} to the proper operating range and then releasing the STOP mode.

External Reset

The μPD78054 family is reset by taking the $\overline{\text{RESET}}$ pin low or by an overflow of the watchdog timer (if enabled). The $\overline{\text{RESET}}$ input pin is a schmitt-trigger input with hysteresis characteristics to protect against spurious system resets caused by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low for a minimum of 10 μs after the power supply reaches its operating voltage.

There is no functional difference between an external reset and an internal reset caused by the overflow of the watchdog timer. In both cases, the main system clock oscillation is stopped and the subsystem clock oscillation continues. During reset, the program counter is loaded with the address contained in the reset vector (addresses 0000H, 0001H). Once the reset is cleared and the oscillation stabilization time of $2^{16}/f_X$ has elapsed, program execution starts at that address.

NEC
NEC Electronics Inc.

μPD78064 Family (μPD78062/063/064/P064) 8-Bit, K-Series Microcontrollers With LCD Controller/Driver, UART, and A/D Converter

Preliminary

September 1993

Description

The μPD78062, μPD78063, μPD78064, and μPD78P064 are members of the K-Series® of microcontrollers featuring an LCD controller/driver, A/D converter, UART, 8-bit hardware multiply and divide instructions, bit manipulation instructions, four banks of main registers, an advanced interrupt handling facility, and a powerful set of memory-mapped on-chip peripherals.

Timing is generated by two oscillators. The main oscillator normally drives the CPU and most peripherals. The 32.768-kHz subsystem oscillator provides time keeping when the main oscillator is turned off. Since CMOS power dissipation is proportional to clock rate, the 78064 provides a software selectable instruction cycle time from 0.40 μs to 122 μs. The STOP and HALT modes turn off parts of the microcontroller for additional power saving. The data retention mode permits RAM contents valid down to 2 volts.

These devices are ideally suited for applications in portable battery-power equipment, office automation, communications, consumer electronics, home appliances, exercise and fitness equipment.

K-Series is a registered trademark of NEC Electronics, Inc.

Features

- LCD controller/driver for up to 160 segments
  - 40 segment lines
  - 4 common lines
  - Static, 1/2 or 1/3 bias
  - LCD resistor ladder available on ROM version
- Eight-channel 8-bit A/D converter
- Two-channel serial communication interface
  - 8-bit clock synchronous interface 0
    Full duplex, three-wire mode
    NEC serial bus interface (SBI) mode
    Half-duplex, two-wire mode
  - Serial interface 2
    Full-duplex, three-wire mode
    UART mode
- Timers: five channels
  - Watchdog timer
  - 16-bit timer/event counter with two 16-bit capture and compare registers
  - Two 8-bit timer/event counters usable as one 16-bit timer/event counter
  - Watch (clock) timer
- 57 I/O lines
  - Two CMOS input-only lines
  - 55 CMOS bidirectional I/O lines with software selectable pullup resistors
- Powerful instruction set
  - 8-bit unsigned multiply and divide
  - 16-bit arithmetic and data transfer instructions
  - 1-bit and 8-bit logic instructions
- Minimum instruction execution times:
  - 0.4/0.8/1.6/3.2/6.4/12.8 μs using 5-MHz main system clock
  - 122 μs selectable using 32.768-kHz subsystem clock
- Memory-mapped on-chip peripherals
  - Special function registers
- Programmable priority, vectored-interrupt controller (two levels)
- Programmable buzzer and clock outputs
- Power saving and battery back up
  - Variable CPU clock rate
  - HALT mode
  - STOP mode
  - 2-V data retention mode
- CMOS operation; V_{DD} from 2.7 to 6.0 V

3g

Internal High-Capacity ROM and RAM

	78062	78063	78064	78P064
ROM	16K bytes	24K bytes	32K bytes	—
PROM	—	—	—	32K bytes
High-speed RAM	512 bytes	1024 bytes	1024 bytes	1024 bytes
LCD display RAM	40 nibbles	40 nibbles	40 nibbles	40 nibbles

Ordering Information

Part Number	ROM	Package	Package Drawing
μPD78062GC-xxx-7EA	16K mask ROM	100-pin plastic QFP (14 x 14 mm)	P100GC-65-7EA
μPD78063GC-xxx-7EA	24K mask ROM		
μPD78064GC-xxx-7EA	32K mask ROM		
μPD78P064GC-7EA	32K OTP ROM		
μPD78062GF-xxx-3BA	16K mask ROM	100-pin plastic QFP (14 x 20 mm)	P100GF-65-3BA
μPD78063GF-xxx-3BA	24K mask ROM		
μPD78064GF-xxx-3BA	32K mask ROM		
μPD78P064GF-3BA	32K OTP ROM		
μPD78P064KL-T (Note 3)	32K UV EPROM	100-pin ceramic LCC w/window (14 x 20 mm)	

Notes:

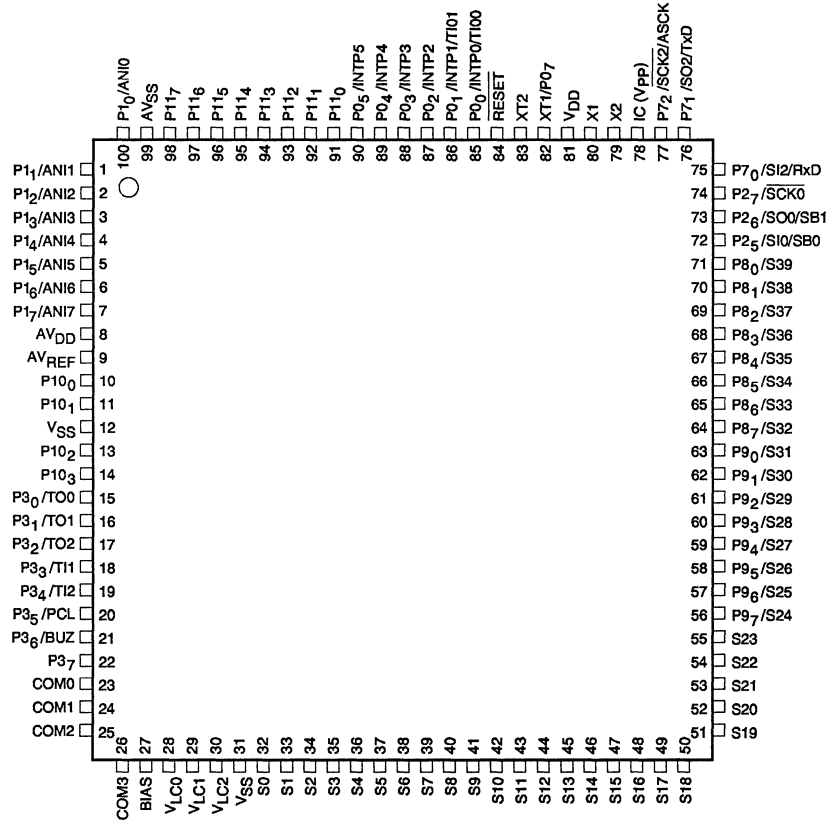
(1) xxx indicates ROM code suffix

(2) All devices listed are standard quality grade

(3) Under development

Pin Configurations

100-Pin Plastic QFP (14 x 14 mm)



Notes:

(1) Connect IC (Internally connected) pin (V_{PP} on μPD78P064) to V_{SS}

(2) AV_{DD} should be connected to V_{DD}

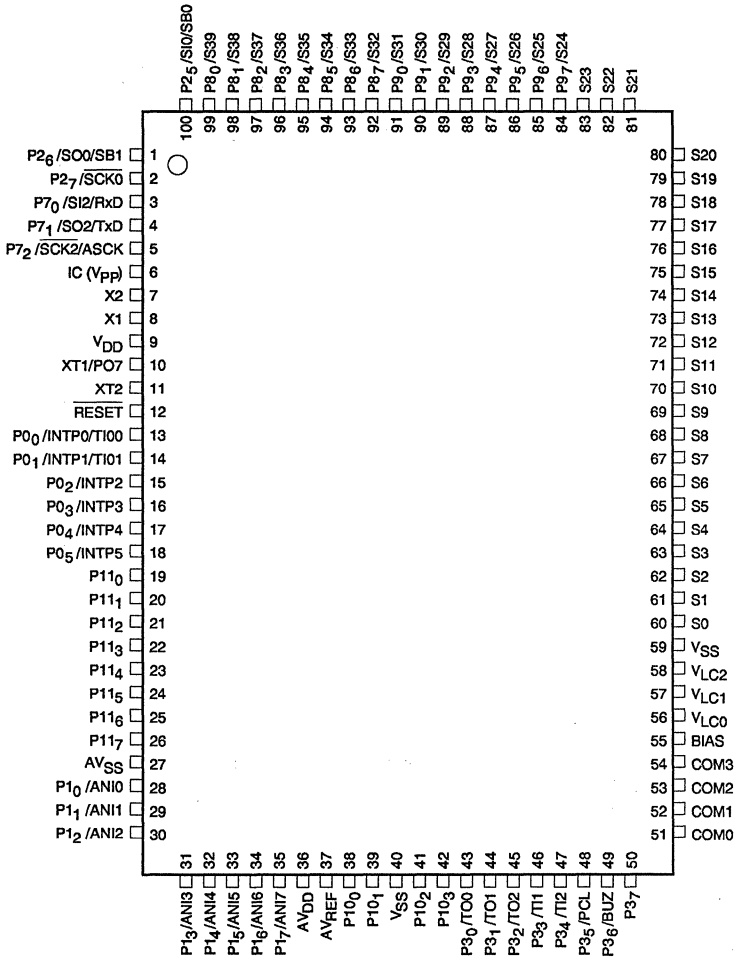
(3) AV_{SS} should be connected to V_{SS}

83YL-9694B

3g

Pin Configurations (cont)

100-Pin Plastic QFP or Ceramic LCC With Window (14 x 20 mm)



Notes:

(1) Connect IC (internally connected) pin (V_{PP} on μPD78P064) to V_{SS}

(2) AV_{DD} should be connected to V_{DD}

(3) AV_{SS} should be connected to V_{SS}

63YL-9696B

Pin Functions; Normal Operating Mode

Symbol	First Function	Symbol	Alternate Function
$P0_0$	Port 0; 7-bit, bit selectable I/O port (Bits 0 and 7 are input only)	INTP0 TI0	External maskable interrupt External count clock input to timer 0 or timer 0 capture trigger to capture registers CR00 and CR01
$P0_1$		INTP1 TI01	External maskable interrupt Timer 0 capture trigger to capture register CR00
$P0_2$ $P0_3$ $P0_4$ $P0_5$		INTP2 INTP3 INTP4 INTP5	External maskable interrupt
$P0_7$		XT1	Crystal oscillator or external clock input for subsystem clock
$P1_0$ - $P1_7$	Port 1; 8-bit, bit-selectable I/O port	ANI0 - ANI7	Analog input to A/D converter
$P2_5$	Port 2; 3-bit, bit-selectable I/O port	SI0 SB0	Serial data input, three-wire serial I/O mode 2/3-wire serial I/O mode
$P2_6$		SO0 SB1	Serial data output, three-wire serial I/O mode 2/3-wire serial I/O mode
$P2_7$		$\overline{SCK0}$	Serial clock I/O for serial interface 0
$P3_0$	Port 3; 8-bit, bit-selectable I/O port	TO0	Timer output from timer 0
$P3_1$		TO1	Timer output from timer 1
$P3_2$		TO2	Timer output from timer 2
$P3_3$		TI1	External count clock input to timer 1
$P3_4$		TI2	External count clock input to timer 2
$P3_5$		PCL	Programmable clock output
$P3_6$		BUZ	Programmable buzzer output
$P3_7$		—	
$P7_0$	Port 7; 3-bit, bit-selectable I/O port	SI2 RxD	Serial data input, three-wire serial I/O mode Asynchronous serial data input
$P7_1$		SO2 TxD	Serial data output, three-wire serial I/O mode Asynchronous serial data output
$P7_2$		$\overline{SCK2}$ ASCK	Serial clock I/O for serial interface 2 Asynchronous serial clock input
$P8_0$ - $P8_7$	Port 8; 8-bit, bit-selectable I/O port	S39 - S32	LCD controller/driver segment signal output
$P9_0$ - $P9_7$	Port 9; 8-bit, bit-selectable I/O port	S31 - S24	LCD controller/driver segment signal output
$P10_0$ - $P10_3$	Port 10; 4-bit, bit-selectable I/O port		
$P11_0$ - $P11_7$	Port 11; 8-bit, bit-selectable I/O port		
S0 - S23	LCD controller/driver segment signal output		
COM0 - COM3	LCD controller/driver common signal output		
V_{LC0} - V_{LC2}	LCD drive voltage input pins		
BIAS	LCD drive power supply output		
$\overline{RESET}$	External system reset input		
X1	Crystal/ceramic reaonator connection or external clock input for main system clock		
X2	Crystal/ceramic resonator connection or inverse of external clock for main system clock		
XT2	Crystal oscillator or left open when not using the subsystem clock		

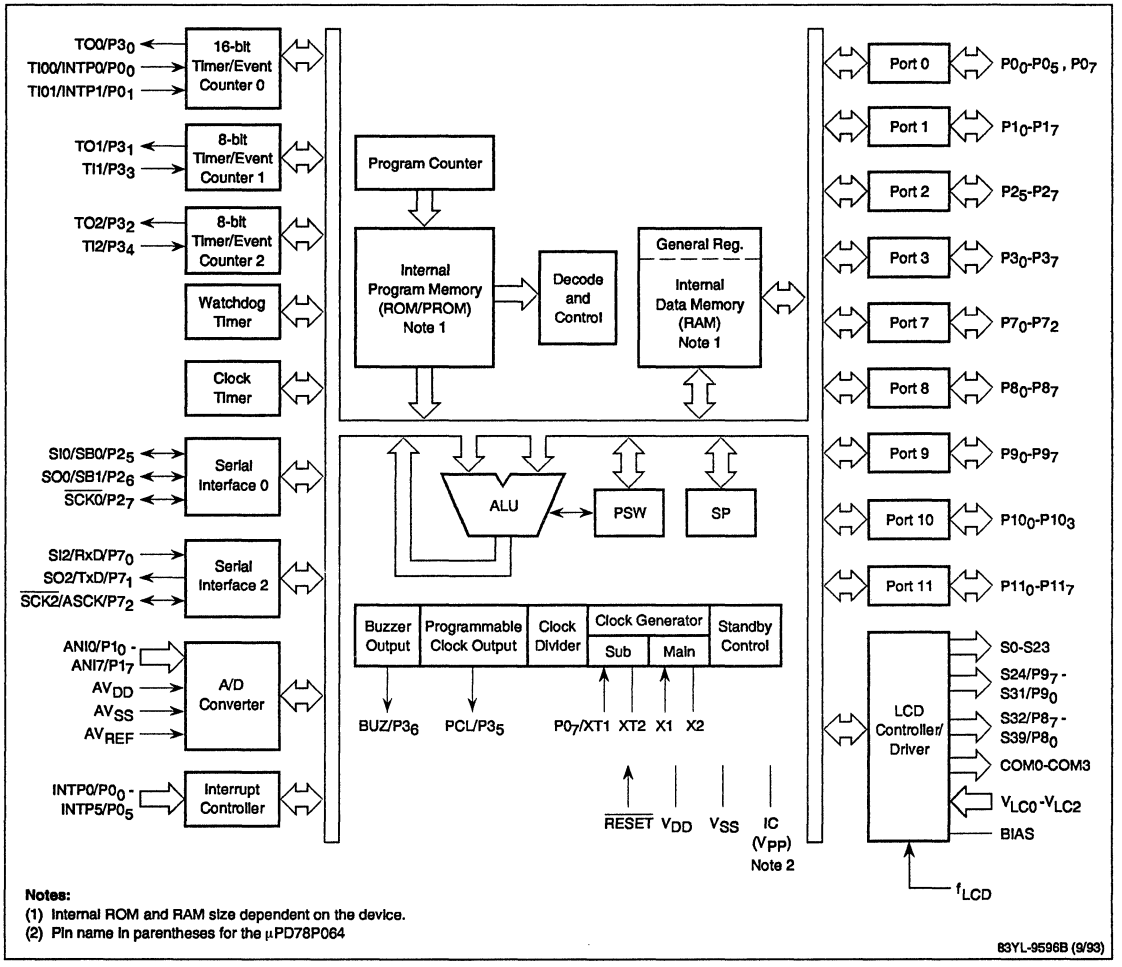
3g

Pin Functions; Normal Operating Mode (cont)

Symbol	First Function	Symbol	Alternate Function
AV_{REF}	A/D converter reference voltage		
AV_{DD}	A/D converter power supply input		
AV_{SS}	A/D converter ground		
V_{DD}	Power-supply input		
V_{PP}	μPD78P064 PROM programming power-supply input		
V_{SS}	Power-supply ground		
IC	Internal connection		

Note: See table 2 and figure 4 for details.

Block Diagram



Notes:
(1) Internal ROM and RAM size dependent on the device.
(2) Pin name in parentheses for the μPD78P064

83YL-9596B (9/93)

3g

FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processing unit (CPU) of the μPD78064 family features 8- and 16-bit arithmetic including an 8 x 8-bit unsigned multiply and 16 x 8-bit unsigned divide (producing a 16-bit quotient and an 8-bit remainder). The multiply executes in 3.2 μs and the divide in 5 μs using the fastest clock cycle with a main system clock of 5.0 MHz.

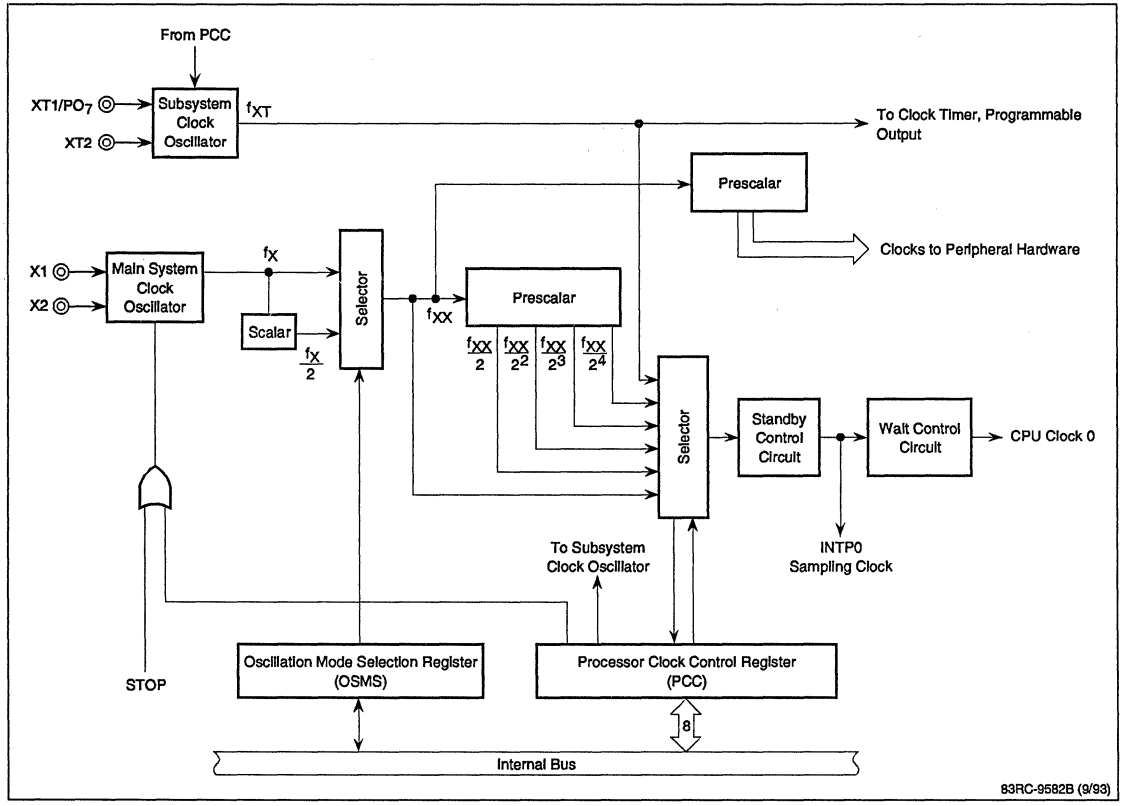
A CALLT vector table and a CALLF area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction can access up to 32 subroutines through their addresses contained in the CALLT vector table (40H to 7FH). A 2-byte call instruction can access any routine beginning in a specific CALLF area (0800H to 0FFFH).

Internal System Clock Generator

The internal system clocks of the μPD78064 family are derived from either the main system or the subsystem oscillator. See figure 1. The clocks for the clock timer and programmable clock output are derived from either the subsystem clock (f_{XT}) or the main system clock. The clocks for all other peripheral hardware are derived from the main system clock.

The CPU clock (ϕ) can be supplied from either the main system clock (f_X) or the subsystem clock (f_{XT}). A selector, which is controlled by the oscillation mode selection register (OSMS), determines whether the main system clock (f_X) or the scaled main system clock ($f_X/2$) is provided to the prescalar (f_{XX}). Using the processor clock control register (PCC), a CPU clock frequency equal to f_{XX}, $f_{XX}/2$, $f_{XX}/4$, $f_{XX}/8$, $f_{XX}/16$ or the subsystem

Figure 1. Internal System Clock Generator



clock f_{XT} can be selected. The CPU clock selected should be based on the power supply voltage available and the desired power consumption. On power up, the CPU clock defaults to the lowest speed from the main system clock ($f_{XX}/16$ with $f_{XX} = f_X/2$) and can be changed while the microcomputer is running.

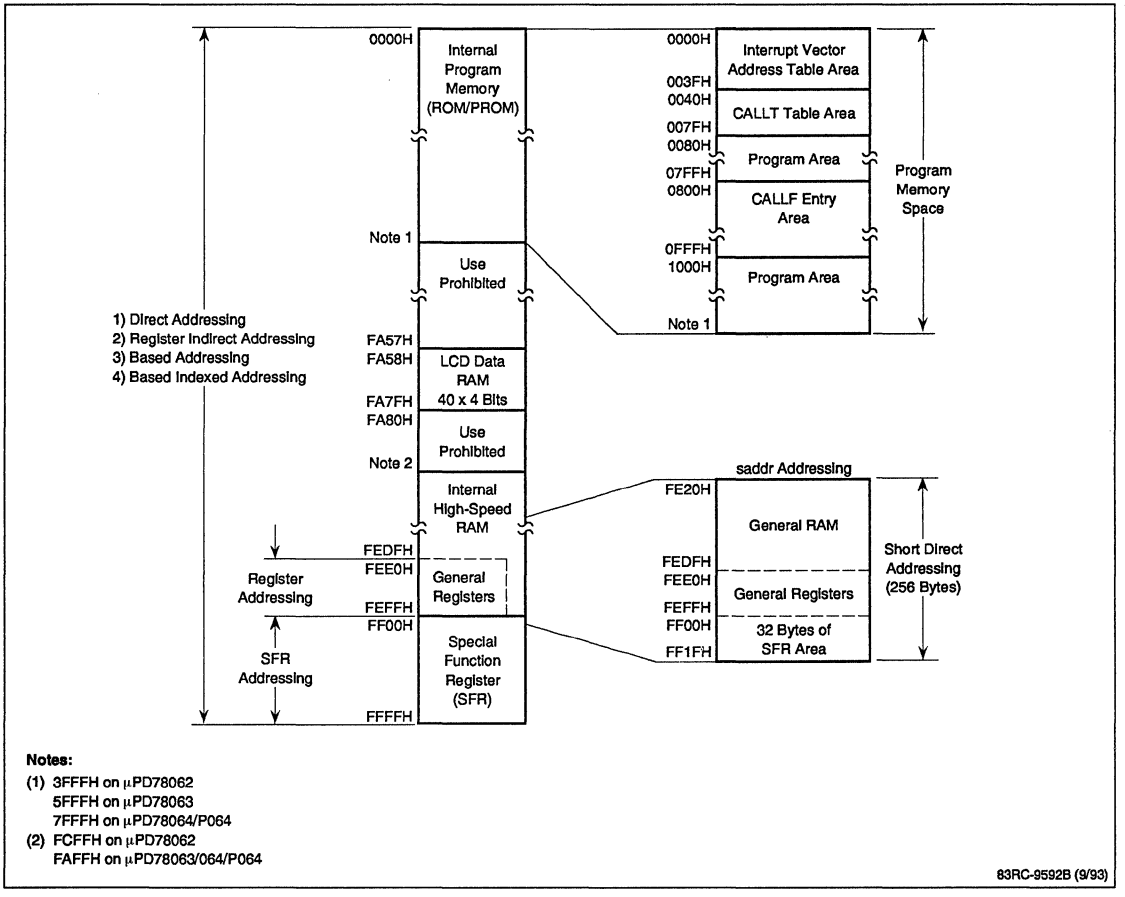
Since the shortest instruction takes two CPU clocks to execute, the fastest minimum instruction execution time (t_{CY}) of 0.4 μs is achieved with a main system clock at 5.0 MHz ($f_{XX} = f_X$) and a V_{DD} of 4.5 to 6.0 volts. However, if the clock timer must generate an interrupt every 0.5 or 0.25 seconds, t_{CY} is 0.48 μs at 4.19 MHz with $f_{XX} = f_X$. The fastest minimum instruction execution time available across the full voltage range of 2.7 to 6.0 volts is 0.96 μs with a 4.19-MHz main system clock ($f_{XX} = f_X$). For the lowest power consumption, the CPU can be operated from the subsystem clock and the minimum instruction execution time is 122 μs at 32.768 kHz.

Memory Space

The μPD78064 family's program and data memory are mapped into the 64K byte address space (0000H-FFFFH). See figure 2. The μPD78064 family is optimized for single-chip operation and does not permit external memory.

3g

Figure 2. Memory Map



Notes:

(1) 3FFFH on μPD78062
5FFFH on μPD78063
7FFFH on μPD78064/P064

(2) FCFFH on μPD78062
FAFFH on μPD78063/064/P064

83RC-9592B (9/93)

Internal Program Memory

All devices in the μPD78064 family have internal program memory. The μPD78062/063/064 contain 16K, 24K, and 32K bytes of internal ROM, respectively. The μPD78P064 contains 32K bytes of UV EPROM or one time programmable ROM. To allow the μPD78P064 to emulate the mask ROM devices, the amount of internal program memory available in the μPD78P064 can be selected using the memory size switching register (IMS).

Internal RAM

The μPD78062 has 512 bytes and the μPD78063/064/P064 have 1024 bytes of Internal RAM. This Internal RAM consists of two types: high-speed Internal RAM and LCD data RAM.

The μPD78062 contains 512 bytes (FD00H to FEFFH) while the μPD78063/064/P064 contain 1024 bytes (FB00H to FEFFH) of high-speed Internal RAM. The high-speed Internal RAM contains the general register banks and the stack. The remainder of the high-speed Internal RAM and any unused register bank locations are available for general storage.

All devices also contain 40 x 4 bits of LCD data RAM (FA58H to FA7FH). The LCD data RAM is used for the display data and/or the unused portion for general storage.

To allow the μPD78P064 to emulate the mask ROM devices, the amount of ROM and high-speed Internal RAM available in the μPD78P064 can also be selected using the IMS.

CPU Control Registers

Program Counter. The program counter is a 16-bit binary counter register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 0001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

Program Status Word. The program status word (PSW) is an 8-bit register that contains flags that are set or reset depending on the results of an instruction. This register can be written to or read from 8 bits at a time. The individual flags can also be manipulated on a bit-by-bit basis. The assignment of PSW bits follows.

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

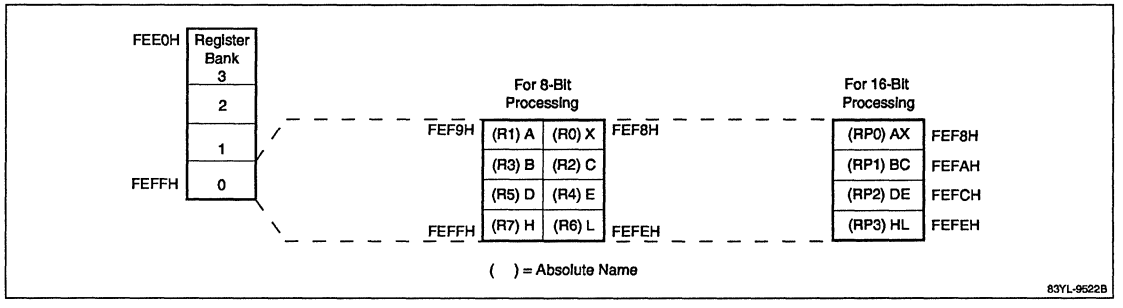
CY	Carry flag
ISP	In-service (interrupt) priority flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

General Registers

The general-purpose registers (figure 3) consist of four banks of registers located at addresses FEE0H to FEFFH in Internal RAM. Each bank consists of eight 8-bit general registers that can also be used in pairs to function as four 16-bit registers. Two bits in the PSW (RBS0 and RBS1) specify which of the register banks is active at any time and are set under program control.

Registers have both functional names (like A, X, B, C, D, E, H, or L for 8-bit registers and AX, BC, DE, and HL for 16-bit registers) and absolute names (like R1, R0, R3, R2, R5, R4, R6, or R7 for 8-bit registers and RP0, RP1, RP2, or RP3 for 16-bit registers). Either the functional or absolute register names can be used in instructions that use the operand identifiers r and rp.

Figure 3. General Registers



Addressing

The program memory addressing (ROM) modes provided are relative, immediate, table indirect, and register addressing. The operand addressing modes provided are implied, register, direct, short direct (saddr), special-function (SFR), register indirect, based, based indexed, and stack addressing.

The SFR addressing and saddr addressing modes use direct addressing and require only 1 byte in the instruction to address RAM. Normally a 65K byte address space requires 2 bytes to address it. One-byte addressing results in faster access times, since the instructions are shorter. SFR addressing addresses the entire 256-byte SFR address space FF00H to FFFFH. Saddr addressing (see figure 2) addresses the 256-byte address space FE20H to FF1FH. FE20H to FEFFH are composed of 224 bytes of internal high speed RAM; FF00H to FF1FH contain the first 32 bytes in the special function register area.

One-byte addressing is accomplished by using the first byte of the instruction for the opcode (and one operand if register A or AX is used) and the second byte of the instruction as an address (offset) into the 256-byte area. If register A or AX is used, the instructions are 2 bytes long thereby providing fast access times. If immediate data is used, the instruction will be 3 or 4 bytes long depending upon whether the immediate data is a byte or a word. Many 16-bit SFRs are in the space FF00H to FF1FH. Using AX as an operand to these SFRs will provide fast access, since the instruction will be only 2 bytes long.

3g

Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by SFR addressing. FF00H to FF1FH can also be accessed using saddr addressing. They are either 8 or 16 bits as required, and many of the 8-bit registers are bit addressable. Table 1 lists the special function registers.

Input/Output Ports

Each device in the μPD78064 family has 57 port lines. Table 2 lists the features of each port and figure 4 shows the structure of each port pin.

Table 1. Special Function Registers

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF00H	Port 0	P0	R/W	x	x	—	00H
FF01H	Port 1	P1	R/W	x	x	—	00H
FF02H	Port 2	P2	R/W	x	x	—	00H
FF03H	Port 3	P3	R/W	x	x	—	00H
FF07H	Port 7	P7	R/W	x	x	—	00H
FF08H	Port 8	P8	R/W	x	x	—	00H
FF09H	Port 9	P9	R/W	x	x	—	00H
FF0AH	Port 10	P10	R/W	x	x	—	00H
FF0BH	Port 11	P11	R/W	x	x	—	00H
FF10H-FF11H	Capture/compare register 00	CR00	R/W	—	—	x	Undefined
FF12H-FF13H	Capture/compare register 01	CR01	R/W	—	—	x	Undefined
FF14H-FF15H	16-bit timer register	TM0	R	—	—	x	00H
FF16H	Compare register 10	CR10	R/W	—	x	—	Undefined
FF17H	Compare register 20	CR20	R/W	—	x	—	Undefined
FF18H	8-bit timer register 1	TM1	R	x	x	—	00H
FF19H	8-bit timer register 2	TM2	R	x	x	—	00H
FF18H-FF19H	16-bit timer register	TMS	R	—	—	x	0000H
FF1AH	Serial I/O shift register 0	SIO0	R/W	—	x	—	Undefined
FF1FH	A/D conversion result register	ADCR	R	—	x	—	Undefined
FF20H	Port mode register 0	PM0	R/W	x	x	—	FFH
FF21H	Port mode register 1	PM1	R/W	x	x	—	FFH
FF22H	Port mode register 2	PM2	R/W	x	x	—	FFH
FF23H	Port mode register 3	PM3	R/W	x	x	—	FFH
FF27H	Port mode register 7	PM7	R/W	x	x	—	FFH
FF28H	Port mode register 8	PM8	R/W	x	x	—	FFH
FF29H	Port mode register 9	PM9	R/W	x	x	—	FFH
FF2AH	Port mode register 10	PM10	R/W	x	x	—	FFH
FF2BH	Port mode register 11	PM11	R/W	x	x	—	FFH
FF40H	Timer clock select register 0	TCL0	R/W	x	x	—	00H
FF41H	Timer clock select register 1	TCL1	R/W	—	x	—	00H
FF42H	Timer clock select register 2	TCL2	R/W	—	x	—	00H
FF43H	Timer clock select register 3	TCL3	R/W	—	x	—	88H
FF47H	Sampling clock select register	SCS	R/W	—	x	—	00H
FF48H	16-bit timer mode control register	TMC0	R/W	x	x	—	00H
FF49H	8-bit timer mode control register	TMC1	R/W	x	x	—	00H
FF4AH	Clock timer mode control register	TMC2	R/W	x	x	—	00H
FF4CH	Capture/compare control register 0	CRC0	R/W	x	x	—	04H
FF4EH	16-bit timer output control register	TOC0	R/W	x	x	—	00H
FF4FH	8-bit timer output control register	TOC1	R/W	x	x	—	00H
FF60H	Serial operating mode register 0	CSIM0	R/W	x	x	—	00H

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF61H	Serial bus interface control register	SBIC	R/W	x	x	—	00H
FF62H	Slave address register	SVA	R/W	—	x	—	Undefined
FF63H	Interrupt timing specify register	SINT	R/W	x	x	—	00H
FF70H	Asynchronous serial interface mode register	ASIM	R/W	x	x	—	00H
FF71H	Asynchronous serial interface status register	ASIS	R	x	x	—	00H
FF72H	Serial interface operating mode register 2	CSIM2	R/W	x	x	—	00H
FF73H	Baud rate generator control register	BRGC	R/W	—	x	—	00H
FF74H	Transmit shift register	TXS	W	—	x	—	FFH
(Note 1)	Serial I/O shift register	SIO2	R/W	—	x	—	FFH
	Receive buffer register	RXB	R	—	x	—	FFH
FF80H	A/D converter mode register	ADM	R/W	x	x	—	01H
FF84H	A/D converter input select register	ADIS	R/W	—	x	—	00H
FFB0H	LCD display mode register	LCDM	R/W	x	x	—	00H
FFB2H	LCD display control register	LCDC	R/W	x	x	—	00H
FFB8H	Key return mode register	KRM	R/W	x	x	—	02H
FFE0H	Interrupt flag register L	IF0L	R/W	x	x	—	00H
FFE1H	Interrupt flag register H	IF0H	R/W	x	x	—	00H
FFE0H-FFE1H	Interrupt flag register	IF0	R/W	—	—	x	0000H
FFE2H	Interrupt request flag register 1L	IF1L	R/W	x	x	—	00H
FFE4H	Interrupt mask flag register L	MK0L	R/W	x	x	—	FFH
FFE5H	Interrupt mask flag register H	MK0H	R/W	x	x	—	FFH
FFE4H-FFE5H	Interrrupt mask flag register	MK0	R/W	—	—	x	FFFFH
FFE6H	Interrupt mask flag register 1L	MK1L	R/W	x	x	—	FFH
FFE8H	Priority order specify flag register L	PR0L	R/W	x	x	—	FFH
FFE9H	Priority order specify flag register H	PR0H	R/W	x	x	—	FFH
FFE8H-FFE9H	Priority order specify flag register	PR0	R/W	—	—	x	FFFFH
FFEAH	Priority order specify flag register 1L	PR1L	R/W	x	x	—	FFH
FFECH	External interrupt mode register 0	INTM0	R/W	—	x	—	00H
FFEDH	External interrupt mode register 1	INTM1	R/W	—	x	—	00H
FFF0H	Memory size switch register	IMS	W	—	x	—	(Note 2)
FFF2H	Oscillation mode select register	OSMS	W	x	x	—	00H
FFF3H	Pullup resistor option register H	PU0H	R/W	x	x	—	00H
FFF7H	Pullup resistor option register L	PU0L	R/W	x	x	—	00H
FFF9H	Watchdog timer mode register	WDTM	R/W	x	x	—	00H

3g

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FFFAH	Oscillation stabilization time select register	OSTS	R/W	—	x	—	04H
FFFBH	Processor clock control register	PCC	R/W	x	x	—	04H

Notes:

(1) SI02 can be used instead of TXS and RXB. SI02 is not a register; it is another symbol that can be used to reference the TXS and RXB registers. A write to SI02 causes the CPU to write to the TXS register, and a read from SI02 causes the CPU to read the RXB register.

(2) Value after reset depends on device
μPD78062: 44H
μPD78063: C6H
μPD78064: C8H
μPD78P064: C8H

Table 2. Digital Port Functions

Port	Operational Features	Configuration	Direct Drive Capability	Software Pullup Resistor Connection (Note 1)
Port 0 (Note 2)	7-bit input or output	Bit selectable		Byte selectable, input bits only
Port 1 (Note 3)	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 2	3-bit input or output	Bit selectable		Byte selectable, input bits only
Port 3	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 7	3-bit input or output	Bit selectable		Byte selectable, input bits only
Port 8 (Note 4)	8-bit input or output	Bit selectable	LCD	Byte selectable, input bits only
Port 9 (Note 4)	8-bit input or output	Bit selectable	LCD	Byte selectable, input bits only
Port 10	4-bit input or output	Bit selectable	LED	Byte selectable, input bits only
Port 11	8-bit input or output	Bit selectable		Byte selectable, input bits only

Notes:

(1) Software pullup resistors can be internally connected (only on a port-by-port basis) to port bits set to input mode. Pullup resistors are not connected to port bits set to output mode.

(2) $P0_0$ and $P0_7$ are input only and do not have a software pullup resistor. When using $P0_7$ as an input, the feedback resistor for the subsystem clock should be disconnected with bit 6 (FRC) of the processor control register (PCC).

(3) Pullup resistors are automatically disconnected on pins used for A/D converter analog inputs.

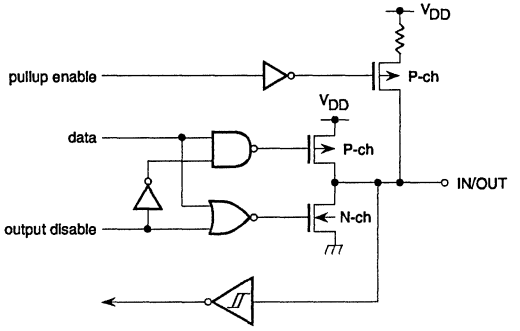
(4) Specified in units of two as either port pins or segment pins.

Figure 4. Pin Input/Output Circuits

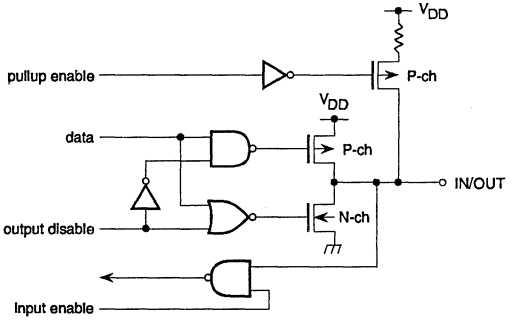
Type 2 ($P0_0$, $\overline{RESET}$)

Schmitt trigger input with hysteresis characteristics

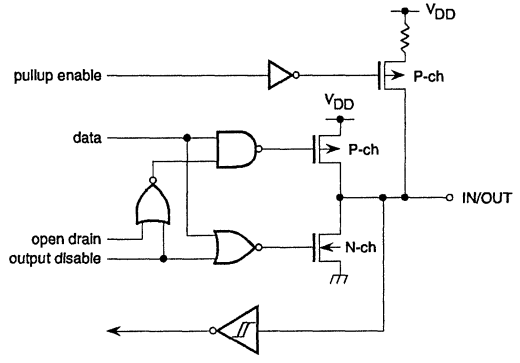
Type 8-A ($P0_1$ - $P0_5$, $P3_3$, $P3_4$, $P7_0$, $P7_2$)



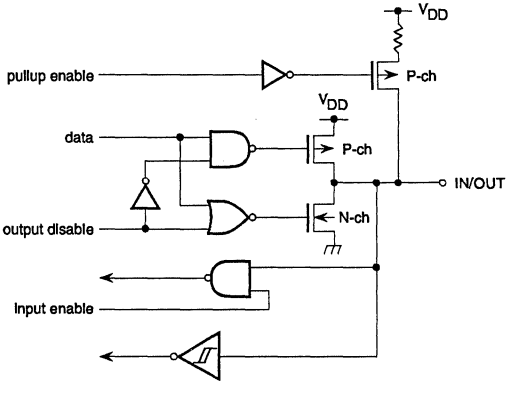
Type 5-A ($P3_0$- $P3_2$, $P3_5$- $P3_7$, $P7_1$, $P10_0$- $P10_3$)



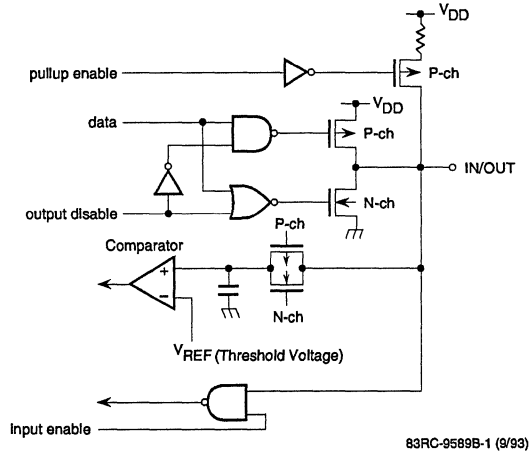
Type 10-A ($P2_5$- $P2_7$)



Type 5-D ($P11_0$- $P11_7$)



Type 11 ($P1_0$- $P1_7$)

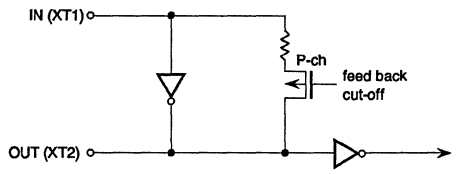


83RC-9589B-1 (9/93)

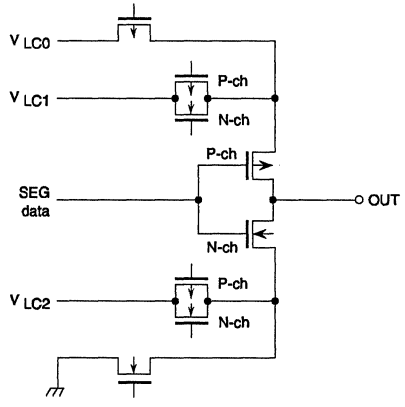
3g

Figure 4. Pin Input/Output Circuits (cont)

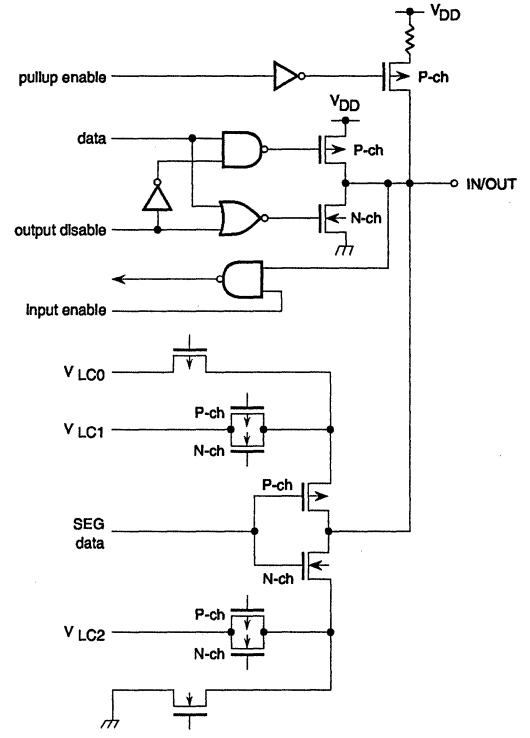
Type 16 (P07, XT2)



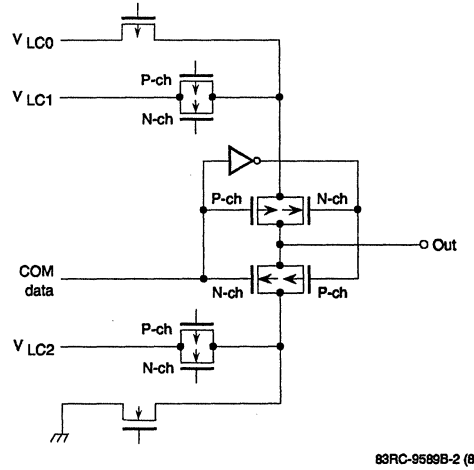
Type 17 (S0 - S23)



Type 17-A (P80 - P87, P90 - P97)



Type 18 (COM0 - COM3)



83RC-9589B-2 (8/93)

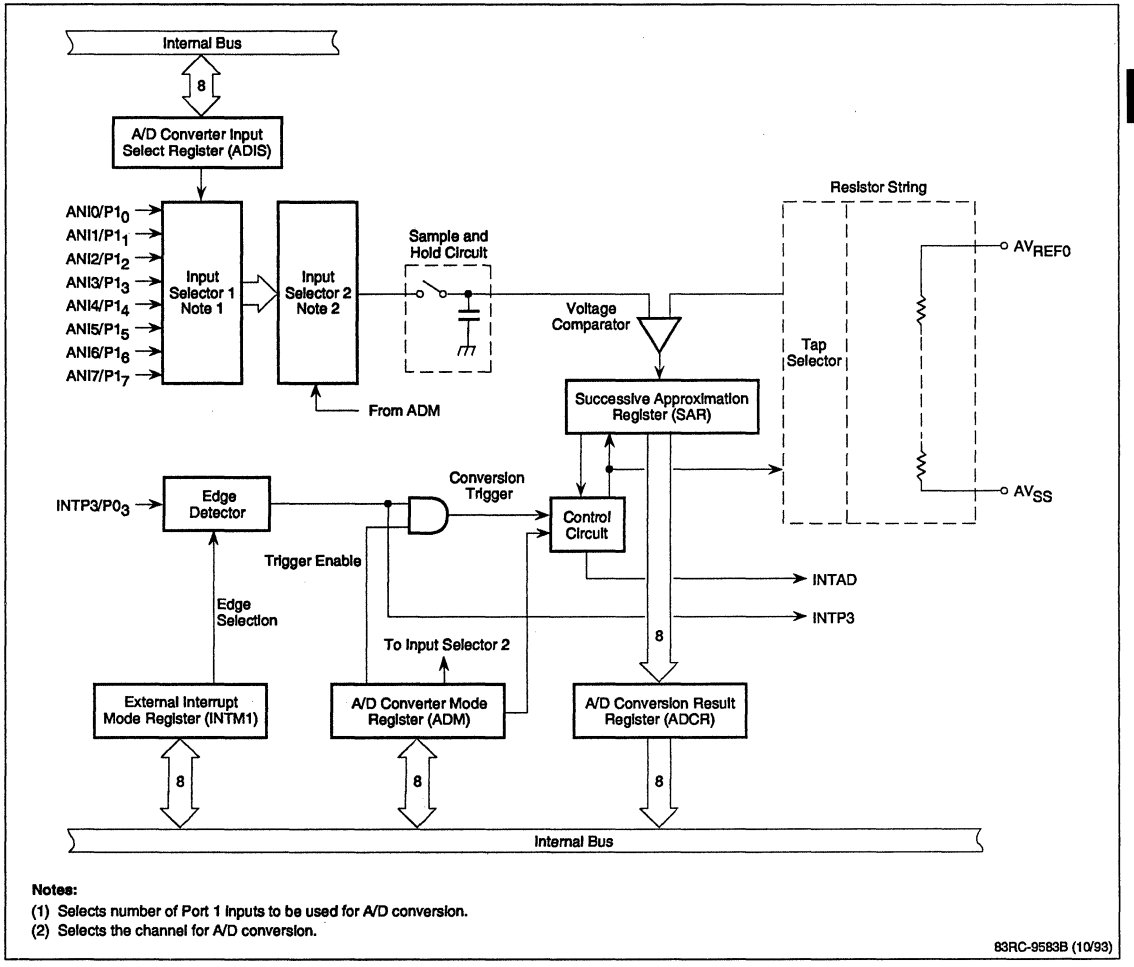
A/D Converter

The μPD78064 family analog-to-digital (A/D) converter (see figure 5) uses the successive-approximation method for converting one of eight multiplexed analog inputs into 8-bit digital data. The minimum conversion time per input is 19.1 μs.

The A/D converter input select register (ADIS) selects the number of inputs that are used in A/D conversion. The remaining inputs are used as ports. The analog input to be converted is selected by programming the A/D converter mode register (ADM). A/D conversion is started by external interrupt INTP3 or by writing to the ADM. When the conversion is completed, the results are stored in the A/D conversion result register (ADCR) and an INTAD interrupt is generated.

If the A/D converter was started by an external interrupt, the A/D converter stops after the interrupt is generated. If the A/D converter was started by software, the A/D converter repeats the conversion until new data is written to the ADM register.

Figure 5. A/D Converter



Notes:
(1) Selects number of Port 1 inputs to be used for A/D conversion.
(2) Selects the channel for A/D conversion.

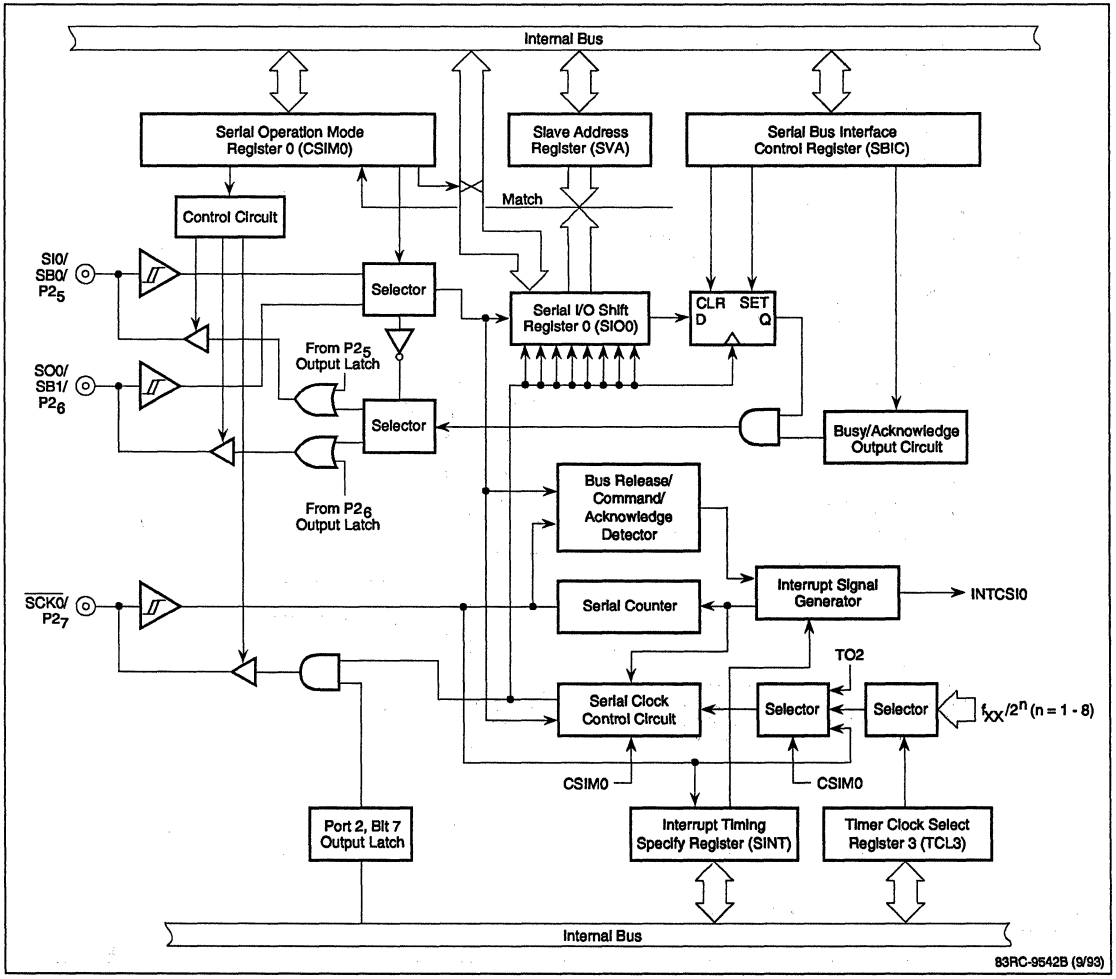
83RC-9583B (10/93)

Serial Interfaces

The μPD78064 family has two independent serial interfaces: serial interface 0 and serial interface 2.

Serial Interface 0. Serial interface 0 is an 8-bit clock synchronous serial interface (figure 6). It can be operated in either a three-wire serial I/O mode, NEC serial bus interface (SBI) mode, or two-wire serial I/O mode. The serial clock can be provided from one of eight internal clocks, the output of 8-bit timer register 2, or the external clock line $\overline{SCK0}$.

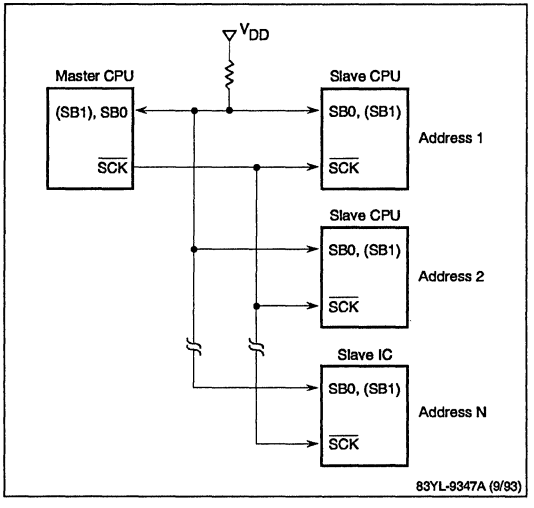
Figure 6. Serial Interface 0



In the three-wire serial I/O mode, the 8-bit shift register (SIO0) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO0 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI0 line providing full-duplex operation. The INTCSI0 interrupt is generated after each 8-bit transfer.

The NEC SBI mode is a two-wire high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are connected in a master/slave configuration (see figure 7). There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over one of the serial bus lines (SB0 or SB1) using a fixed hardware protocol synchronized with the $\overline{SCK0}$ line . Each slave device of the μPD78064 family can be programmed to respond in hardware to any one of 256 addresses set in its slave address register (SVA). There are also 256 commands and 256 data types. Since all commands are user definable, many software protocols, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

Figure 7. SBI Mode Master/Slave Configuration



The two-wire serial I/O mode provides half-duplex operation using either the SB0 or SB1 line and the $\overline{SCK0}$ line. Communication format and handshaking can be handled in software by controlling the output levels of the data and clock lines between transfers. For data transmission, when the 8-bit shift register (SIO0) is loaded with a byte of data, eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of either the SB0 or SB1 line (MSB first). In addition, this byte of data is also shifted back into SIO0 on the rising edge of these pulses providing a means of verifying that the transmission was correct.

For data reception, the SIO0 register is preloaded with the value FFH. As this data value is shifted out on the falling edge of the serial clock, it disables the n-channel open-drain driver. This allows the receive data to be driven on to the serial line and shifted into the SIO0 register on the rising edge of the serial clock. The INTCSI0 interrupt is generated after each 8-bit transfer.

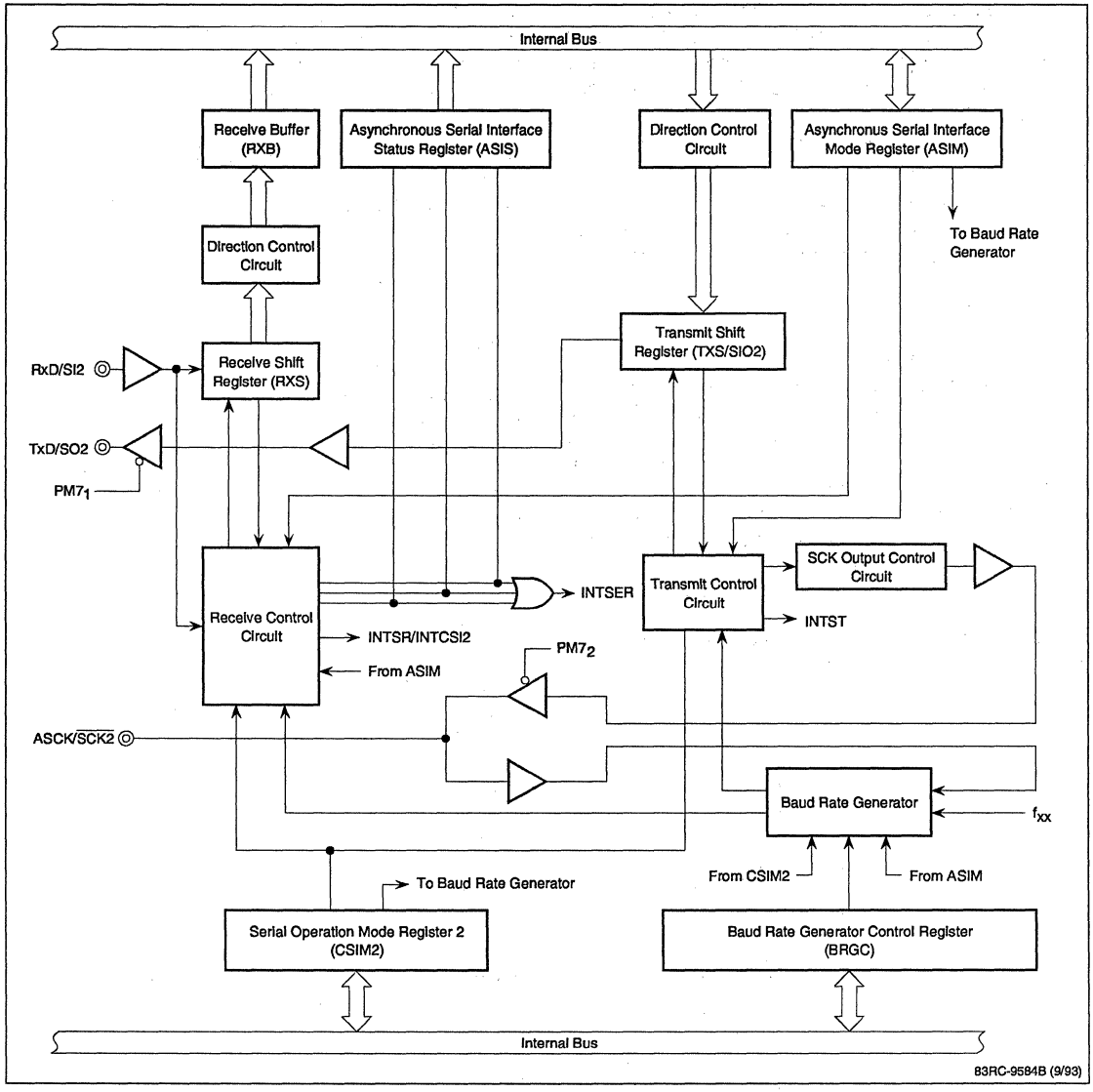
Serial Interface 2. Serial interface 2 is an 8-bit serial interface (figure 8) that can be operated in either a UART mode or a three-wire serial I/O mode. The internal baud rate generator circuit (figure 9) scales an internal clock to provide standard baud rates from 75 to 38400 bps. For non-standard baud rates, the internal baud rate generator circuit scales an external clock input on the ASCK pin. The output of the baud rate generator circuit is used as the data transmission and sampling clock in the UART mode or the data clock in the three-wire serial I/O mode.

In the UART mode, half or full-duplex operation with various protocols is programmable. The asynchronous serial interface mode register (ASIM) is used to specify the number of stop bits (1 or 2), data character length (7 or 8 bits), parity (none, even, or odd), receive operation control (enable or disable), and transmit operation control (enable or disable). The ASIM register is also used to enable or disable the generation of an interrupt when a reception error occurs and if an internal or external clock will be supplied to the baud rate generator circuit.

A transmit operation is started by writing a data character to the transmit shift register (TXS) register. The start bit, parity bit, and stop bits are automatically added by the hardware to the data character in the TXS register. The data in the TXS register is shifted out of the TxD line and when the TXS register is empty, a transmission complete interrupt (INTST) is generated.

When the receive operation control is enabled, the RXD line is sampled using the clock specified by the ASIM register. When the RxD line is detected low, sampling starts at the midpoint of each bit. If the first sample yields a low, it is identified as a start bit. The RxD line continues to be sampled at the midpoint of each bit. Reception of one frame of data is complete when the data character bits, parity bit (if being transmitted), and one stop bit are detected after the start bit. Even if

Figure 8. Serial Interface 2

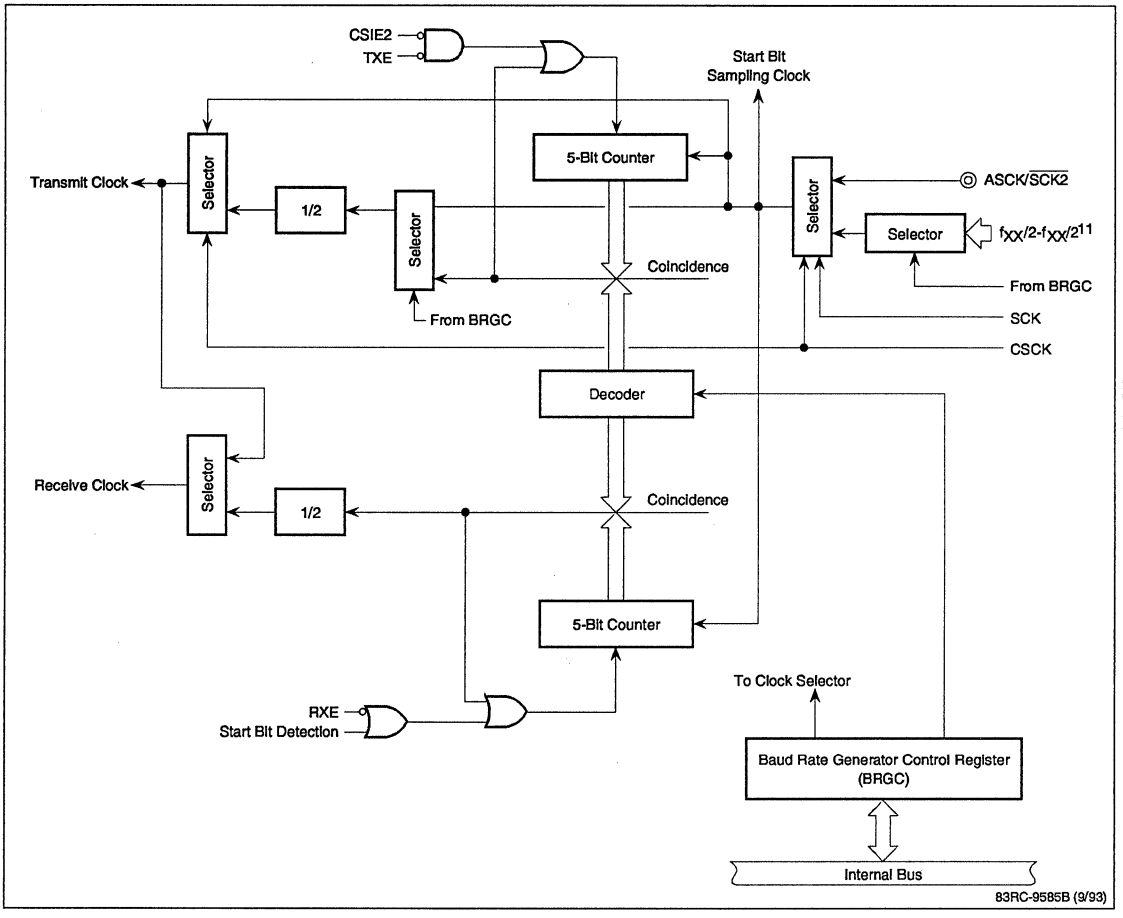


the protocol is set for two stop bits, only one stop bit is used for the end of reception detection.

When one frame of data has been received, the received data in the shift register is transferred to the receive buffer (RXB) and a reception complete vectored interrupt (INTSR) is generated even if an error (parity and/or framing) is detected. The data must be read from the RXB register before another frame is received or an overrun error will be generated. If an error occurs, the appropriate flag is set in the asynchronous serial interface status register (ASIS) and a receive error interrupt (INTSER) is generated.

In the three-wire serial I/O mode, the TXS register is loaded with a byte of data and eight clock pulses are

Figure 9. Internal Baud Rate Generator



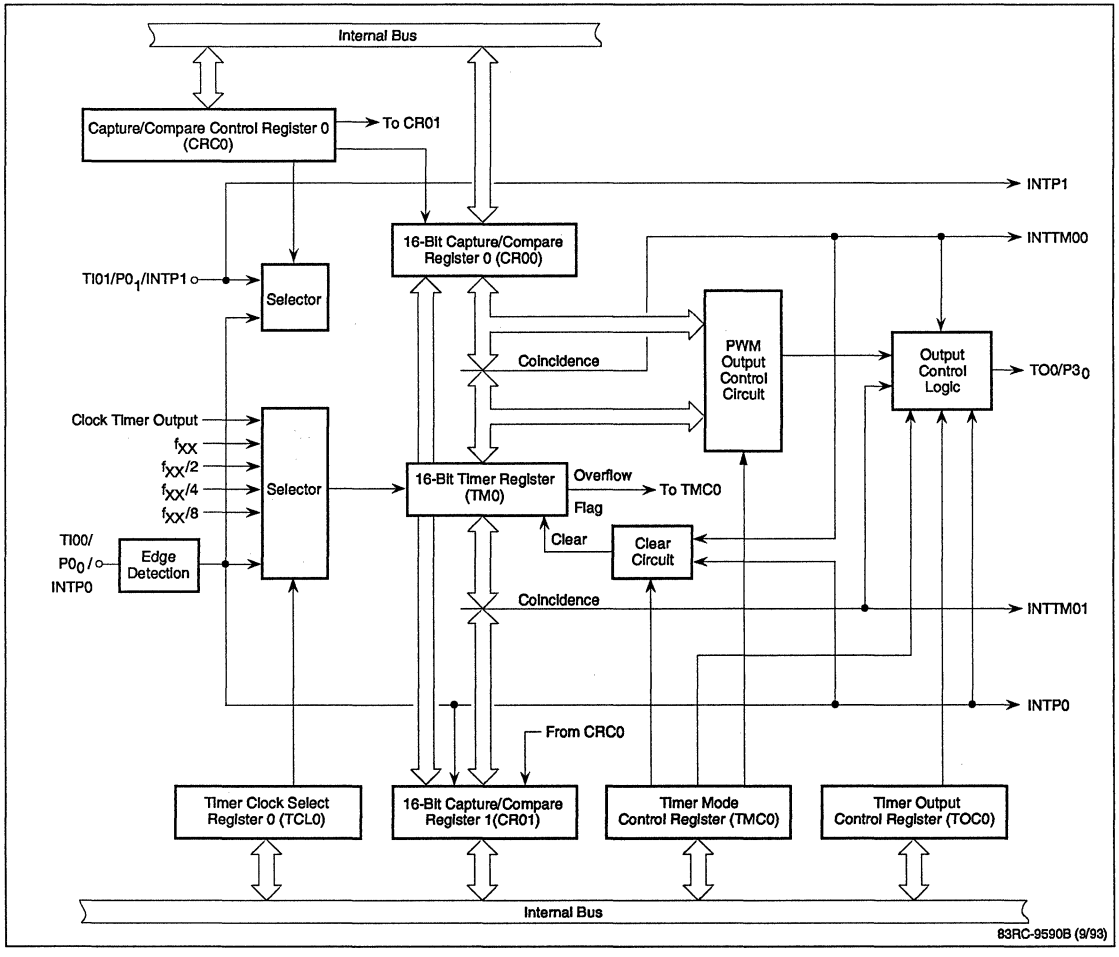
generated. The falling edge of these eight pulses shifts the byte of data out of the SO2 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI2 line providing full-duplex operation. The INTCSI2 interrupt is generated after each 8-bit transfer.

Timers

The μPD78064 family has one 16-bit timer/event counter, two 8-bit timer/event counters that can be combined for use as a 16-bit timer/event counter, a clock timer, and a watchdog timer. All of these can be programmed to count a number of prescaled values of the main system clock. In addition, the clock timer can also count the subsystem clock. All of the timer/event counters can count external events.

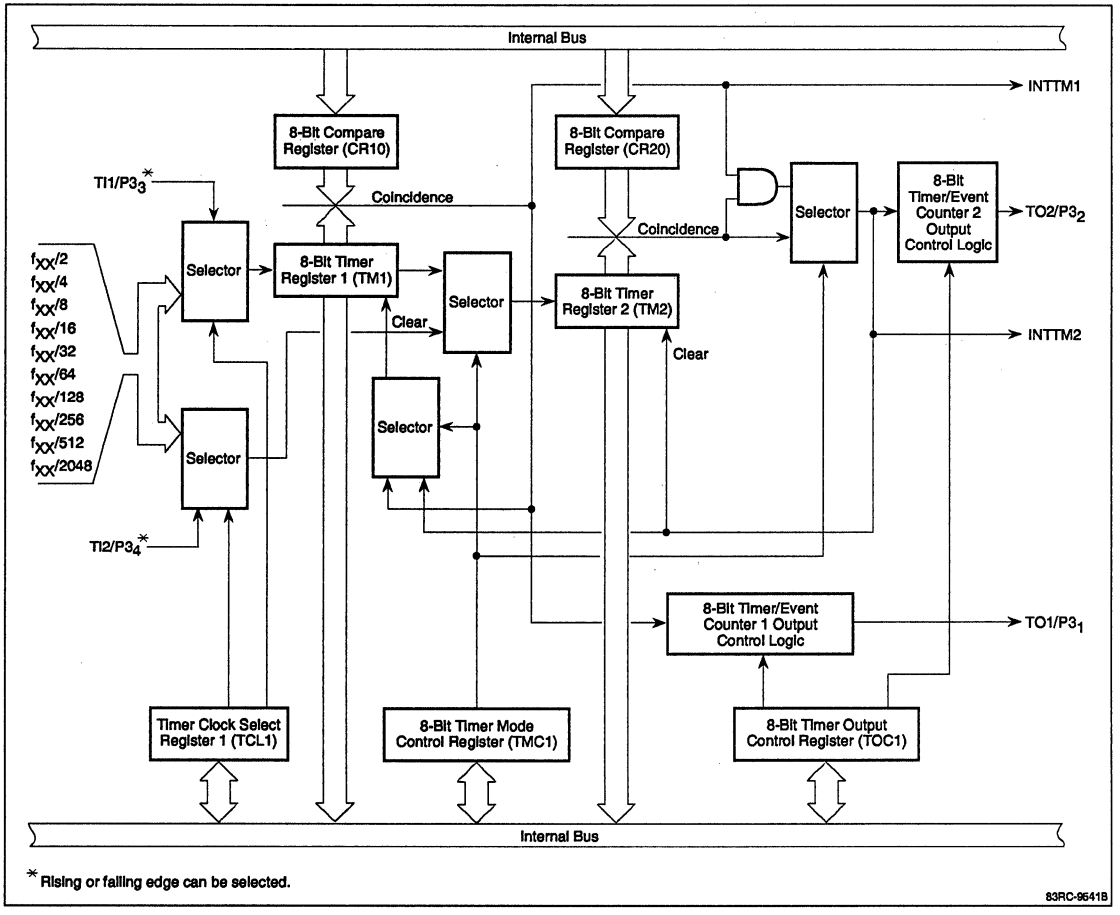
16-Bit Timer/Event Counter 0. Timer/event counter 0 (figure 10) consists of a 16-bit counter (TM0), two 16-bit capture/compare registers (CR00, CR01), control registers (TMC0, TOC0, and CRC0), clock select register (CLC0), and a timer output control circuit (TO0). Timer 0 can be used as an interval timer, to count external events on the timer input (TI00) pin, to output a programmable square wave, a 14-bit pulse-width modulated output, a one-shot pulse output, or to measure pulse widths.

Figure 10. 16-Bit Timer/Event Counter 0



8-Bit Timer/Event Counters 1 and 2. Timer/event counters 1 and 2 (figure 11) each consist of an 8-bit timer (TM1 or TM2), an 8-bit compare register (CR10 or CR20), and timer output control logic (TO1 or TO2). The timers are controlled by registers TCL1, TMC1, and TOC1 via five selectors. Timer/event counters 1 and 2 can each be used as an 8-bit interval timer, to count external events on the timer input pins (TI1 or TI2), or to output a programmable square wave. In addition, timers 1 and 2 also can be combined as a 16-bit timer/event counter and used as a 16-bit interval timer, to count external events on TI1, or to output a programmable square wave on TO2.

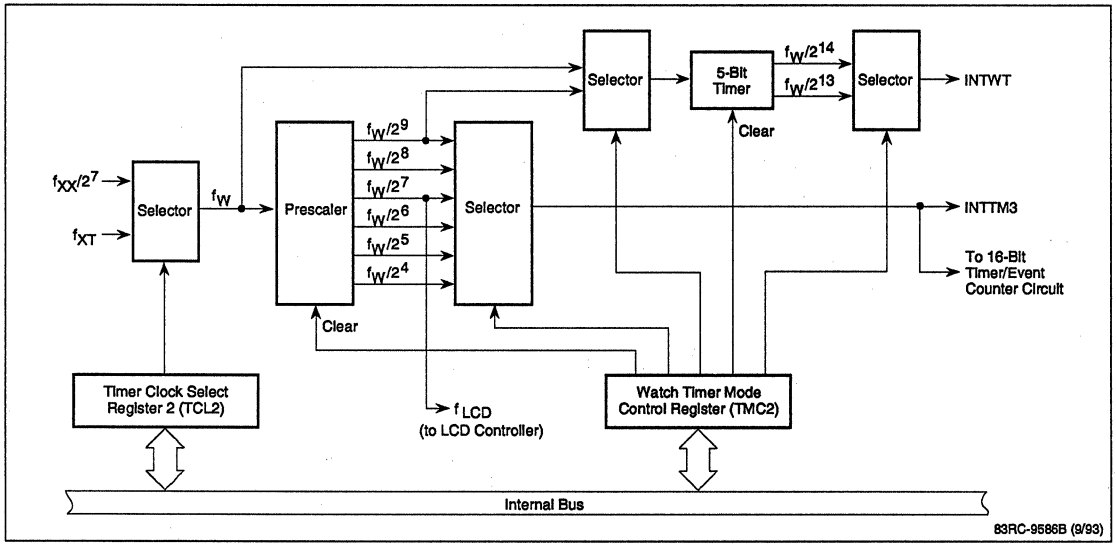
Figure 11. 8-Bit Timer/Event Counters 1 and 2



Clock Timer 3. Clock timer 3 (figure 12) is a 5-bit timer that can be used as a time source to keep track of time of day, to release the STOP or HALT modes at regular intervals, or to initiate any other task that must be performed at regular intervals. When driven by the subsystem clock, the clock timer continues to operate in the STOP mode.

The clock timer can function as both a clock timer and interval timer simultaneously. When used as a clock timer, interrupt request INTWT (not a vectored interrupt) can be generated using a main or subsystem clock every 0.5 or 0.25 seconds. When used as an interval timer, vectored interrupt request INTTM3 is generated at preselected time intervals. With a main system clock of 4.19 MHz and $f_{XX} = f_X$ or if using the subsystem clock of 32.768 kHz, the following time intervals can be selected: 489 μs, 978 μs, 1.96 ms, 3.91 ms, 7.82 ms or 15.6 ms.

Figure 12. Clock Timer 3



Watchdog Timer. The watchdog timer (figure 13) can be used as either a watchdog timer or an interval timer. When used as a watchdog timer it protects against program run-away. It can be selected to generate a nonmaskable interrupt (INTWDT), which vectors to address 0004H, or to generate an internal reset signal, which vectors to the restart address 0000H if the timer is not cleared by the program before it overflows. Eight program-selectable intervals based on the main system clock are available. With a main system clock of 4.19 MHz and $f_{XX} = f_X$, they are 0.489, 0.978, 1.96, 3.91, 7.82, 15.6, 31.3, and 125 ms. With a main system clock of 4.19 MHz and $f_{XX} = f_X/2$, they are 0.978, 1.96, 3.91, 7.82, 15.6, 31.3, 62.6, and 250 ms. Once the watchdog timer is initialized and started, the timer's mode cannot be changed and the timer can only be stopped by reset.

When used as an interval timer, maskable interrupts (INTWDT), which vector to address 0004H, are generated repeatedly at a preset interval. The time intervals available are the same as in the watchdog timer mode.

Programmable Clock Output

The μPD78064 family has a programmable clock output (PCL) that can be used for carrier output for remote-controlled transmissions or as a clock output for peripheral devices. The main system clock (f_{XX}) divided by 1, 2, 4, 8, 16, 32, 64, or 128 or the subsystem clock (f_{XT}) can be output on the PCL pin. If the main system clock is 4.19 MHz and $f_{XX} = f_X$, the following frequencies are available: 4.19 MHz, 2.1 MHz, 1.05 MHz, 524 kHz, 262 kHz, 131 kHz, 65.5 kHz, and 32.7 kHz. With a main system clock of 4.19 MHz and $f_{XX} = f_x/2$, the following frequencies are available: 2.1 MHz, 1.05 MHz, 524 kHz, 262 kHz, 131 kHz, 65.5 kHz, 32.7 kHz and 16.4 kHz. With a subsystem clock of 32.768 kHz, 32.768 kHz is also available. See figure 14.

3g

Figure 13. Watchdog Timer

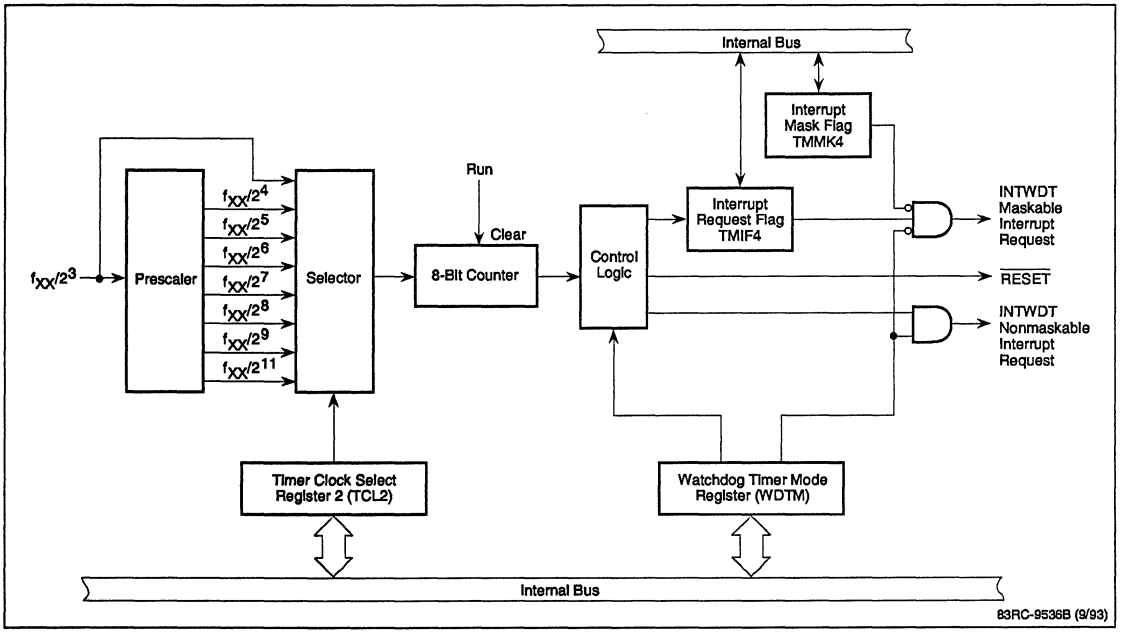
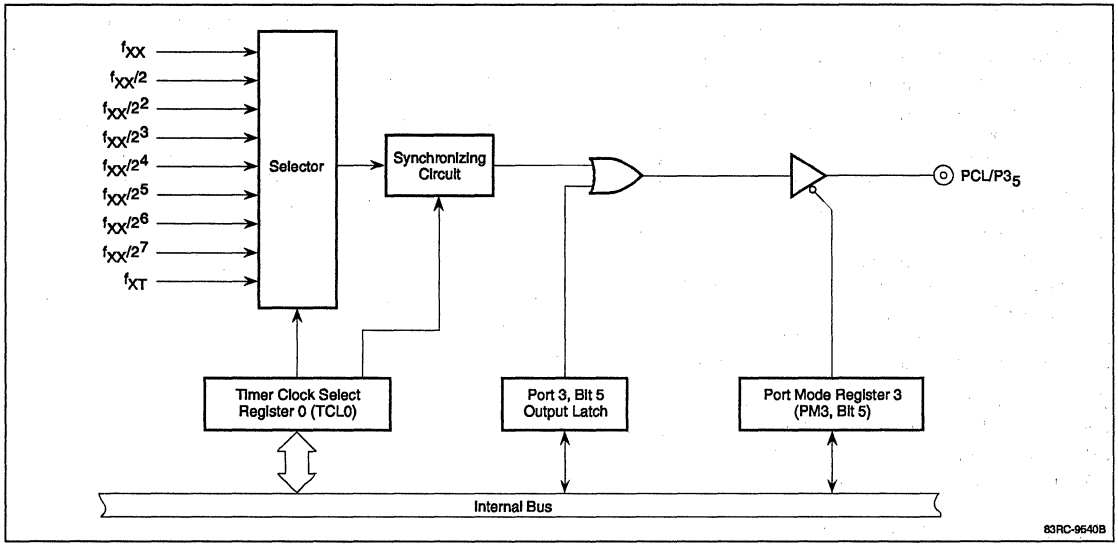


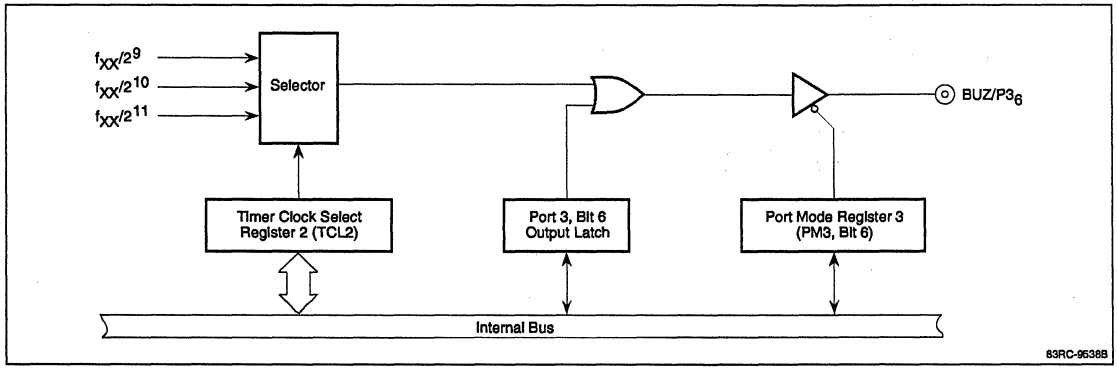
Figure 14. Programmable Clock Output



Buzzer Output

The μPD78064 family also has a programmable buzzer output (BUZ). The buzzer output frequency can be programmed to be equal to the main system clock (f_{XX}) divided by 512, 1024, or 2048. With a main system clock of 4.19 MHz and $f_{XX} = f_X$, the buzzer can be set to 8.2, 4.1, or 2.0 kHz. With a main system clock of 4.19 MHz and $f_{XX} = f_X/2$, the buzzer can be set to 4.1, 2.0, or 1.0 kHz. See figure 15.

Figure 15. Buzzer Output



LCD Controller/Driver

The liquid crystal display (LCD) controller/driver (figure 16) has 4 common plus 40 segment lines and can be programmed to operate in any of four modes. It can operate in the static mode (drive 40 segments), the duplexed mode (drive 80 segments), the triplexed mode (drive 120 segments), or quadruplexed mode (drive 160 segments). The duplexed mode uses 1/2 bias, the triplexed mode can use either 1/2 or 1/3 bias, and the quadruplexed mode uses 1/3 bias.

The LCD controller automatically refreshes the LCD by taking data from the LCD data RAM and uses display data multiplexers, segment drivers S0-S39, and common drivers COM0-COM3 to drive the LCD. The clock timer provides a clock signal (f_{LCD}) that is derived from the clock timer (figure 17). Using the LCD display mode register (LCDM), the main LCD clock (LCDCL) is selected as f_{LCD} (or prescaled values $f_{LCD}/2$, $f_{LCD}/4$, or $f_{LCD}/8$) yielding frame frequencies of 64, 128, 256, or 512 Hz with a main system clock of 4.19 MHz or a subsystem clock of 32.768 kHz.

Figure 16. LCD Controller/Driver

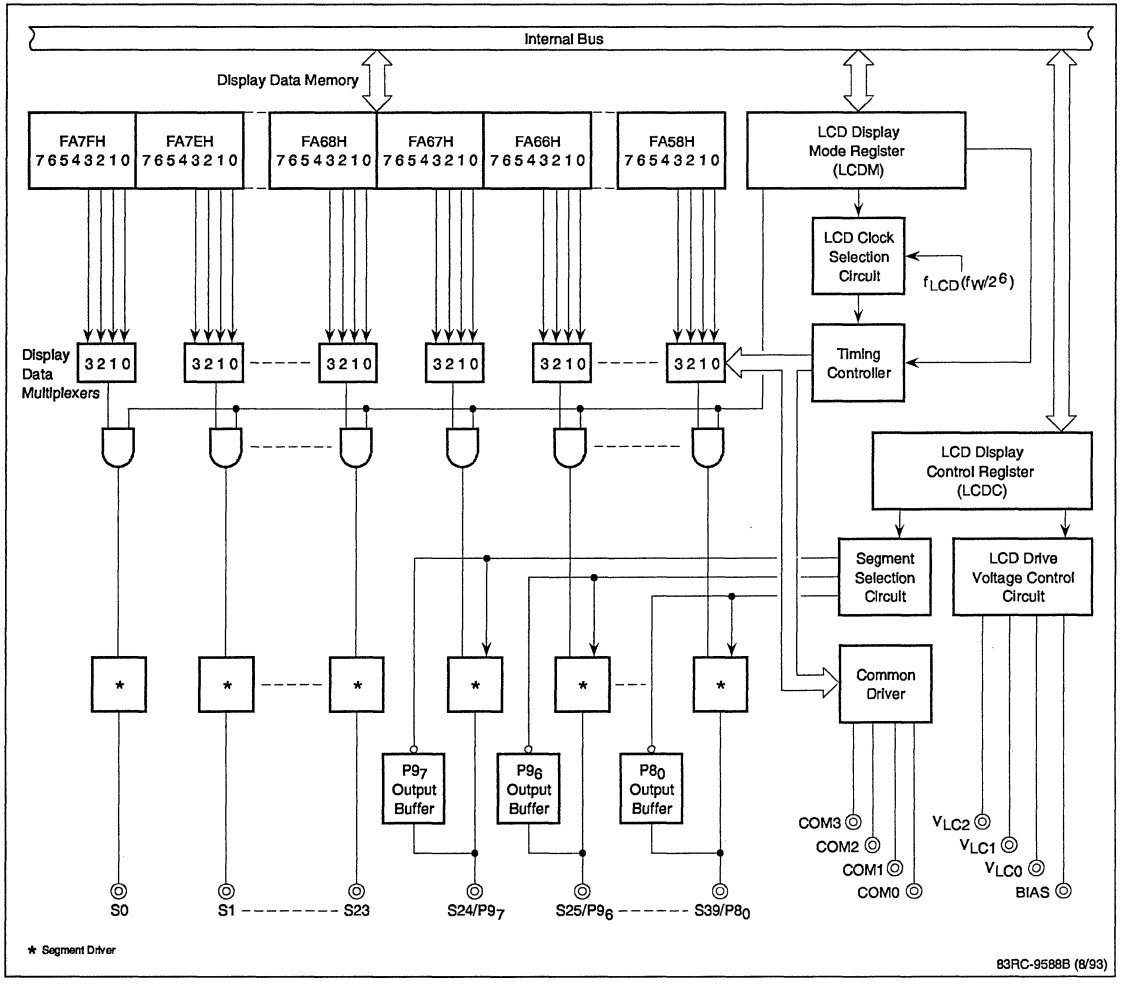
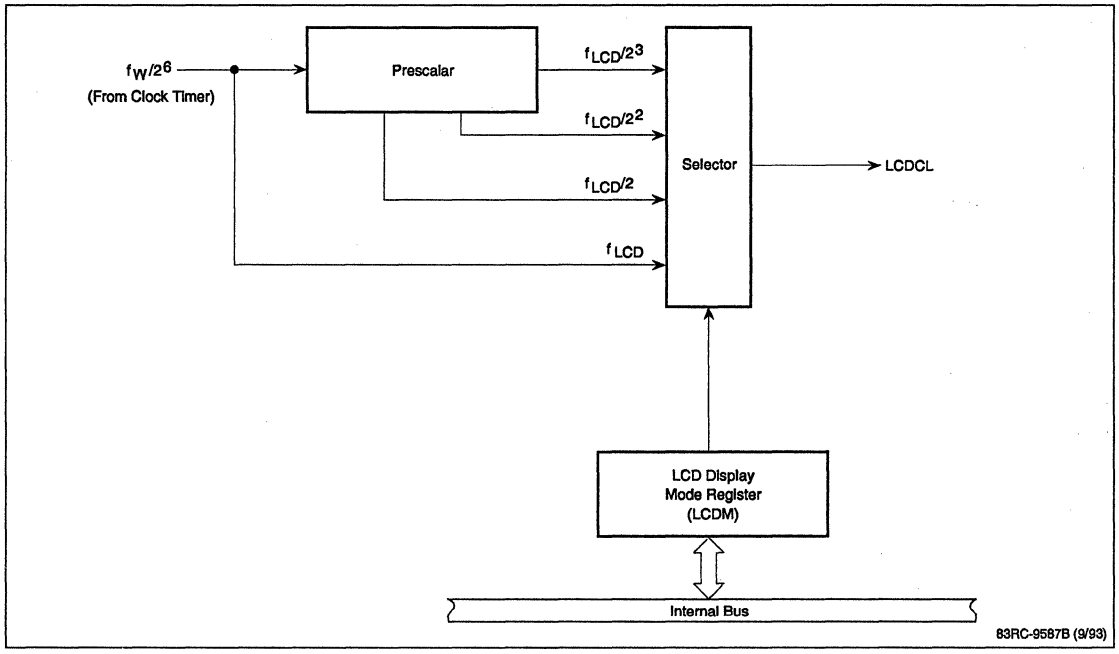


Figure 17. LCD Clock Selection Circuit



The LCDON bit in the LCDM register is used to enable or disable (blank) the display. The LCDM0-LCDM2 bits in the LCDM register are used to select the bias method and LCDM4-LCDM6 are used to set the frame frequency. The LCD controller/driver can operate in the STOP mode as long as the clock timer is driven by the subsystem clock.

Drive levels can be set internally by ordering the resistor ladder mask option on the μPD7806x mask ROM devices; otherwise, external resistors can be connected to pins V_{LC0}-V_{LC2} and the BIAS pin. The BIAS pin can be used to control the contrast of the LCD.

The LCDC register is used to select $P8_0$/S39 to $P9_7$/S24 pin functions and the LCD power supply. The pins $P8_0$/S39 to $P9_7$/S24 are selected in groups of two to be port pins or segment pins S24 to S39. The LCD power supply selections are: (a) LCD power not supplied by the μPD78064 device, (b) LCD drive power supplied from V_{DD}, or (c) LCD drive power supplied from the BIAS pin.

Interrupts

The μPD78064 family has 19 maskable hardware interrupt sources; 7 are external and 12 are internal. Of these 19 interrupt sources, 17 cause a vectored interrupt while the 2 testable inputs only generate an interrupt request. All 19 maskable interrupts can be used to release the HALT mode except INTP0. INTP0 cannot be used to release the STOP mode and cannot release the HALT mode when register SCS = 0. In addition, there is one nonmaskable interrupt from the watchdog timer, one software interrupt, and a reset interrupt. The watchdog timer overflow interrupt (interrupt vector table address 0004H) can be initialized to be a nonmaskable interrupt or the highest default priority maskable interrupt. The software interrupt, generated by the BRK instruction, is not maskable. See table 3 and figure 18.

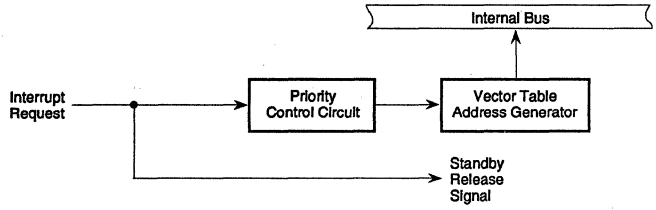
Table 3. Interrupt Sources and Vector Addresses

Type of Request	Default Priority	Signal Name	Interrupt Source	Location	Vector Address	Interrupt* Configuration
Restart	—	RESET	$\overline{\text{RESET}}$ input pin	External	0000H	
	—	INTWDT	Watchdog timer overflow (when reset mode selected)	Internal	—	
Nonmaskable	—	INTWDT	Watchdog timer overflow (when nonmaskable interrupt selected)	Internal	0004H	A
Maskable	0	INTWDT	Watchdog timer overflow (when interval timer selected)	Internal	0004H	B
	1	INTP0	External interrupt edge detection	External	0006H	C
	2	INTP1	External interrupt edge detection	External	0008H	D
	3	INTP2	External interrupt edge detection	External	000AH	D
	4	INTP3	External interrupt edge detection	External	000CH	D
	5	INTP4	External interrupt edge detection	External	000EH	D
	6	INTP5	External interrupt edge detection	External	0010H	D
	7	INTCSI0	End of clocked serial interface 0 transfer	Internal	0014H	B
	8	INTSER	Serial interface 2 UART reception error	Internal	0018H	B
	9	INTSR	End of serial interface 2 UART reception	Internal	001AH	B
		INTCSI2	End of serial interface 2 three-wire transfer			
	10	INTST	End of serial interface 2 UART transmission	Internal	001CH	B
	11	INTTM3	Clock timer reference time interval signal	Internal	001EH	B
	12	INTTM00	16-bit timer/event counter capture/compare (CR00) coincidence signal	Internal	0020H	B
	13	INTTM01	16-bit timer/event counter capture/compare (CR01) coincidence signal	Internal	0022H	B
	14	INTTM1	8-bit timer/event counter 1 coincidence signal	Internal	0024H	B
	15	INTTM2	8-bit timer/event counter 2 coincidence signal	Internal	0026H	B
	16	INTAD	End of A/D conversion	Internal	0028H	B
Software	—	—	BRK instruction	Internal	003EH	E
Test input	—	INTWT	Clock timer overflow	Internal	—	F
	—	INTPT11	Port 11 falling edge detection	External	—	F

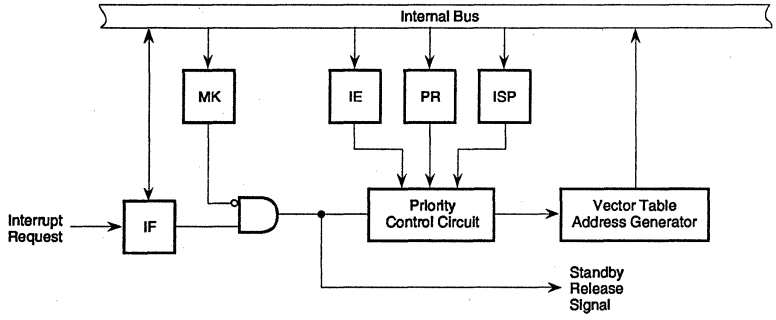
* See figure 18.

Figure 18. Interrupt Configurations

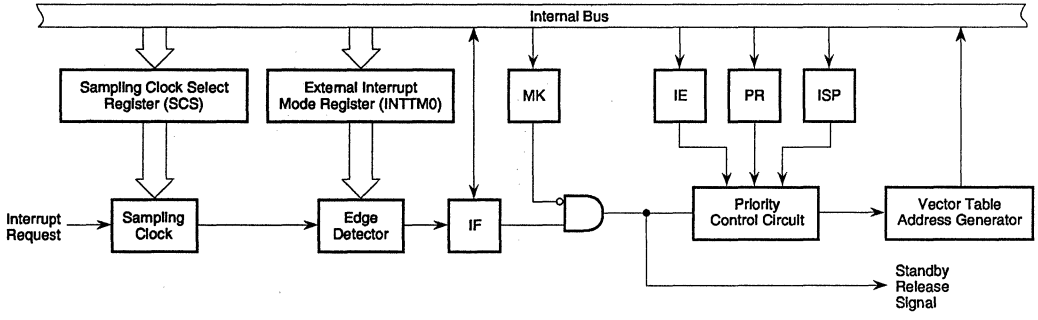
Type A: Internal nonmaskable interrupt



Type B: Internal maskable interrupt



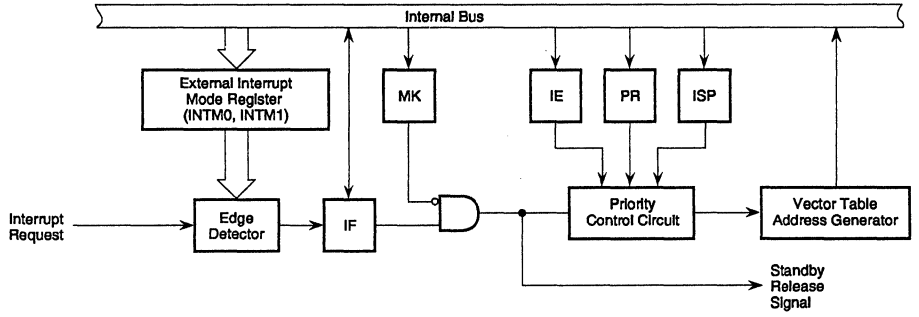
Type C: External maskable interrupt (INTP0)



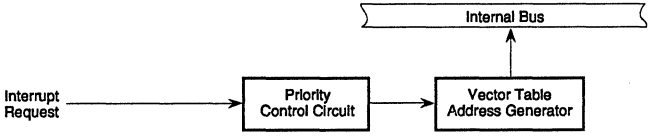
83RC-9591B-1

Figure 18. Interrupt Configurations (cont)

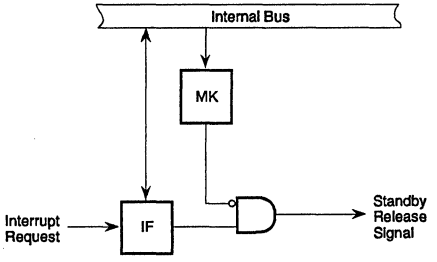
Type D: External maskable interrupt (except INTP0)



Type E: Software interrupt



Type F: Test input



Abbreviations:

IF: Interrupt request flag
IE: Interrupt enable flag
ISP: In-service priority flag
MK: Interrupt mask flag
PR: Priority specify flag

83RC-9691B-2

3g

Interrupt Servicing. The μPD78064 family provides two levels of programmable hardware priority control and services all interrupt requests except the two testable interrupts (INTWT and INTPT11) using vectrored interrupts. The programmer can choose the priority of servicing each maskable interrupt by using the interrupt control registers.

Interrupt Control Registers. The μPD78064 family has three 3-byte interrupt control registers. The interrupt request flag registers (IF0L, IF0H, and IF1L) contain an interrupt request flag for each interrupt. The interrupt mask registers (MK0L, MK0H, and MK1L) are used to enable or disable any individual interrupt. The priority flag registers (PR0L, PR0H, and PR1L) can be used to specify a high or a low priority level for each interrupt except the two testable interrupts (INTWT and INTPT11).

Five other 8-bit registers are associated with interrupt processing. The key return mode register (KRM) contains the KRIF interrupt request flag associated with falling-edge detection on port 11 and the KRMK mask flag used to enable or disable clearing of the standby mode if a falling edge is detected on 8 ($P11_0$ - $P11_7$), 6($P11_2$ - $P11_7$), 4($P11_4$ - $P11_7$), or 1($P11_7$) bit (s) of port 11 as determined by the KRM2 and KRM3 bits. The external interrupt mode registers (INTM0 and INTM1) are used to select a rising, falling, or both edges as the valid edge for each of the external interrupts INTP0 to INTP5. The sampling clock select register (SCS) is used to select a sampling clock for the noise eliminator circuit on external interrupt INTP0.

The IE and the ISP bit of the program status word are also used to control interrupts. If the IE bit is 0, all maskable interrupts are disabled. The IE bit can be set or cleared using the EI and DI instructions, respectively, or by directly writing to the PSW. The IE bit is cleared each time an interrupt is accepted. The ISP bit is used by hardware to hold the priority level flag of the interrupt being serviced.

Interrupt Priority. If the watchdog timer overflow interrupt (INTWDT) has been initialized to be a nonmaskable interrupt, it has priority over all other interrupts. Two hardware-controlled priority levels are available for all maskable interrupts that generate a vectored interrupt (i.e., all except the testable interrupt). Either a high or a low priority level can be assigned by software to each of the maskable interrupts. Interrupt requests of the same priority or a priority higher than the processor's current priority level are held pending until interrupts in the current service routine are enabled by software or until one instruction has been executed after returning from the current service routine. Interrupt requests of a lower priority are always held pending until one instruction has been executed after returning from the current service routine.

The default priorities listed in table 3 are fixed by hardware and are effective only when it is necessary to choose between two interrupt requests of the same software-assigned priority. For example, the default priorities would be used after the completion of a high-priority routine, if two interrupts of the same software priority were pending.

The software interrupt, initiated by the BRK instruction, is executed regardless of the processor's priority level and the state of the IE bit. It does not alter the processor's priority level.

Vectored Interrupt Servicing. When a vectored interrupt is acknowledged, the program status word and the program counter are saved on the stack, the processor's priority is set to that specified for the interrupt, the IE bit in the PSW is set to zero, and the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (RETB instruction for the software interrupt) reverses the process and the μPD78064 family microcomputer resumes the interrupted routine.

Standby Modes

HALT, STOP, and data retention modes are provided to reduce power consumption when CPU action is not required.

The HALT mode is entered by executing a HALT instruction while the CPU is operating from the main system or subsystem clock. In HALT mode, the CPU clock is stopped while the main system and the subsystem clock continue to run. The HALT mode is released by any unmasked interrupt request (except INTP0 if register SCS = 0), a nonmaskable interrupt request, an unmasked test input, or an external reset pulse.

Power consumption may be further reduced by using the STOP mode. The STOP mode is entered by executing a STOP instruction while operating from the main system clock. In STOP mode, the main system clock input pin X1 is internally grounded stopping both the CPU and the peripheral hardware clock. The STOP mode is released by any unmasked interrupt request except INTP0, a nonmaskable interrupt request, an unmasked test input, or an external reset pulse. Any peripheral using the main oscillator as a clock source will also be disabled in the STOP mode and interrupts

from such a peripheral can not be used to exit the STOP mode. Table 4 summarizes both the HALT and STOP standby modes.

When exiting the STOP mode, a wait time occurs before the CPU begins code execution to allow the main system clock oscillator circuit to stabilize. The oscillation stabilization time is selected by programming the OSTS register with one of five values before entering the STOP mode; the values range from 0.8 msec to 52.4 msec at f_X = 5 MHz.

Once in the STOP mode, power consumption can be further minimized by lowering the power supply voltage V_{DD} to 2 volts. This places the device in the data retention mode. The contents of Internal RAM and the registers are retained. This mode is released by first raising V_{DD} to the proper operating range and then releasing the STOP mode.

External Reset

The μPD78064 family is reset by taking the $\overline{\text{RESET}}$ pin low or by an overflow of the watchdog timer (if enabled). The $\overline{\text{RESET}}$ input pin is a schmitt-trigger input with hysteresis characteristics to protect against spurious system resets caused by noise.

There is no functional difference between an external reset and an internal reset caused by the overflow of the watchdog timer. In both cases, the main system clock oscillation is stopped and the subsystem clock oscillation continues. During reset, the program counter is loaded with the address contained in the reset vector (addresses 0000H, 0001H). Once the reset is cleared and the oscillation stabilization time of $2^{16}/f_X$ has elapsed, program execution starts at that address.

Table 4. Standby Mode Operation Status

Item	HALT Mode	STOP Mode
Setting instruction	HALT instruction	STOP instruction
System clock when setting	Main system or subsystem clock	Main system clock
Clock oscillator	Main system and subsystem clocks can oscillate; CPU clock is stopped.	Subsystem clock can oscillate; CPU clock and main system clock are stopped.
CPU	Operation stopped	Operation stopped
Ports	Maintain previous state	Maintain previous state
16-bit timer/event counter	Operational from main system clock, or with clock timer output, or TI00 selected as the count clock	Operational only with clock timer output or TI00 selected as count clock.
8-bit timer/event counters	Operational from main system clock or with TI1 and TI2 selected as the count clock	Operational only with TI1 and TI2 selected as count clock
Clock timer	Operational from main system clock or with f_{XT} as count clock	Operational only with f_{XT} as count clock
Watchdog timer	Operational from main system clock	Operation stopped
Serial interface 0	Operational from main system clock or with external clock	Operational only with external clock
Serial interface 2	Operational from main system clock or with external clock	Operation stopped
LCD controller/ driver	Operational from main system clock or when f_{XT} is selected as count clock to clock timer	Operational when f_{XT} is selected as count clock to clock timer
A/D converter	Operational from main system clock	Operation stopped
External interrupts	Operational except for INTP0 when its sampling clock is based on the CPU clock	INTP0 not operational; INTP1 to INTP5 operational

3g

μPD78K0 Product Line

The μPD78K0 product line's instruction set features both 8- and 16-bit data transfer and arithmetic instructions, 8-bit logic instructions, and single-bit manipulation instructions. Multiply and divide instructions are also included except as noted in the μPD78K0 Instruction Set table. Branch instructions exist to test individual bits in the program status word, the 8-bit accumulator, the special function registers, and in the short address (saddr) portion of memory. Instructions range in length from 1 to 4 bytes, depending on the instruction and addressing mode.

This programming reference contains the following three tables for the μPD78K0 product line: (1) instruction set, (2) special function registers, and (3) interrupt vectors and test inputs.

Operands and Operations

Refer to the following tables for the definitions of symbols in the operand and operation columns of the Instruction Set table.

Specify operands in accordance with the rules of operand representation; for details, refer to the assembler specifications. If two or more items are available in the description method, select one.

Uppercase letters, such as "A" or "PSW," are key symbols and must be written as shown in the Registers, Flags, and Symbols table. See the Registers, Flags and Symbols table for the list of key symbols. Lowercase letters, such as "sfr" or "mem" are not key symbols and an absolute value or label must be substituted by the user when writing the instruction. For example, "MOV A, sfr" may be written as "MOV A, P0." When the symbols +, –, #, !, $, and [ ] are used as a prefix of a word, the symbol remains while lower case letters are replaced by a value. For example, "ADD A, #byte" may be written as "ADD A, #0AFH," or "BR $addr16" may be written as "BR $LOOP1."

Symbols r and rp can be described using the functional name or absolute name.

Operands

Symbol	Definitions
#	Immediate data
!	Absolute address
$	Relative address
[ ]	Indirect addressing
r	Register Functional name: X, A, C, B, E, D, L, H Absolute name: R0 to R7
rp	Register pair Functional name: AX, BC, DE, HL Absolute name: RP0 to RP3
sfr	Special Function Registers (8-bit). See the table "Special Function Registers (μPD78K0 Product Line)." This table shows the SFRs included in each device family, access units usable with each SFR, and the reserved symbol and address of each SFR.
sfrp	Special Function Register Pair (16-bit). See the table "Special Function Registers (μPD78K0 Product Line)." This table shows the SFRs included in each device family, access units usable with each SFR, and the reserved symbol and address of each SFR.
saddr	Memory address addressed by means of short direct addressing: FE20H-FF1FH immediate data or label.
saddrp	Memory address addressed by means of short direct addressing pair: FE20H-FF1EH immediate data or label (even address only)
addr16	16-bit address: 0000H-FFFFH immediate data or label (even address only)
addr11	11-bit address: 0800H-0FFFH immediate data or label
addr5	5-bit address: 40H-7EH immediate data or label (even address only)
word	16-bit data: 16-bit immediate data or label
byte	8-bit data: 8-bit immediate data or label
bit	3-bit immediate data (bit position in byte) or label
RBn	Register bank: RB0-RB3

Registers, Flags, and Symbols

Symbol	Definitions
A	A register; 8-bit accumulator
X	X register
B	B register
C	C register
D	D register
E	E register
H	H register
L	L register
R0-R7	Registers 0 to 7 (absolute names)

Registers, Flags, and Symbols (cont)

Symbol	Definitions
AX	Register pair (AX); 16-bit accumulator
BC	Register pair (BC)
DE	Register pair (DE)
HL	Register pair (HL)
RP0- RP3	Register pairs 0 to 3 (absolute names)
PC	Program counter
SP	Stack pointer
PSW	Program status word
CY	Carry flag
AC	Auxiliary carry flag
Z	Zero flag
RBS1- RBS0	Register bank select flags
IE	Interrupt enable flag
jdisp8	Signed 2's complement data (8 bits) indicating relative address distance between first address of next instruction and branch destination address
( )	Memory contents indicated by address or register contents in ( )
xxH	Hexadecimal number
x_H, x_L	Higher 8 bits and lower 8 bits of 16-bit register pair
∧	Logical product (AND)
∨	Logical SUM (OR)
∀	Exclusive logical sum (exclusive OR)
—	Inverted data

Flag Column Indicators

Symbol	Action
Blank	No change
0	Cleared to 0
1	Set to 1
x	Set or cleared depending on the result
R	Value previously saved is restored

Instruction Set (μPD78K0 Product Line)

Mnemonic	Operand	Operation	Bytes	Flags Z	Flags AC	Flags CY
8-Bit Data Transfer						
MOV	r, #byte	r ← byte	2			
	saddr, #byte	(saddr) ← byte	3			
	sfr, #byte	sfr ← byte	3			
	A, r (Note 1)	A ← r	1			
	r, A (Note 1)	r ← A	1			
	A, saddr	A ← (saddr)	2			
	saddr, A	(saddr) ← A	2			
	A, sfr	A ← sfr	2			
	sfr, A	sfr ← A	2			
	A, !addr16	A ← (addr16)	3			
	!addr16, A	(addr16) ← A	3			
	PSW, #byte	PSW ← byte	3	x	x	x
	A, PSW	A ← PSW	2			
	PSW, A	PSW ← A	2	x	x	x
	A, [DE]	A ← (DE)	1			
	[DE], A	(DE) ← A	1			
	A, [HL]	A ← (HL)	1			
	[HL], A	(HL) ← A	1			
	A, [HL + byte]	A ← (HL + byte)	2			
	[HL + byte], A	(HL + byte) ← A	2			
	A, [HL + B]	A ← (HL + B)	1			
	[HL + B], A	(HL + B) ← A	1			
	A, [HL + C]	A ← (HL + C)	1			
	[HL + C], A	(HL + C) ← A	1			
XCH	A, r (Note 1)	A ↔ r	1			
	A, saddr	A ↔ (saddr)	2			
	A, sfr	A ↔ sfr	2			
	A, !addr16	A ↔ (addr16)	3			
	A, [DE]	A ↔ (DE)	1			
	A, [HL]	A ↔ (HL)	1			
	A, [HL + byte]	A ↔ (HL + byte)	2			
	A, [HL + B]	A ↔ (HL + B)	2			
	A, [HL + C]	A ↔ (HL + C)	2			

Instruction Set (μPD78K0 Product Line) (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
16-Bit Data Transfer						
MOVW	rp, #word	rp ← word	3			
	saddrp, #word	(saddrp) ← word	4			
	sfrp, #word	sfrp ← word	4			
	AX, saddrp	AX ← (saddrp)	2			
	saddrp, AX	(saddrp) ← AX	2			
	AX, sfrp	AX ← sfrp	2			
	sfrp, AX	sfrp ← AX	2			
	AX, rp (Note 2)	AX ← rp	1			
	rp, AX (Note 2)	rp ← AX	1			
	AX, !addr16	AX ← (addr16)	3			
	!addr16, AX	(addr16) ← AX	3			
XCHW	AX, rp (Note 2)	AX ↔ rp	1			
8-Bit Operations						
ADD	A, #byte	A, CY ← A + byte	2	x	x	x
	saddr, #byte	(saddr), CY ← (saddr) + byte	3	x	x	x
	A, r (Note 3)	A, CY ← A + r	2	x	x	x
	r, A (Note 3)	r, CY ← r + A	2	x	x	x
	A, saddr	A, CY ← A + (saddr)	2	x	x	x
	A, !addr16	A, CY ← A + (addr16)	3	x	x	x
	A, [HL]	A, CY ← A + (HL)	1	x	x	x
	A, [HL + byte]	A, CY ← A + (HL + byte)	2	x	x	x
	A, [HL + B]	A, CY ← A + (HL + B)	2	x	x	x
	A, [HL + C]	A, CY ← A + (HL + C)	2	x	x	x
ADDC	A, #byte	A, CY ← A + byte + CY	2	x	x	x
	saddr, #byte	(saddr), CY ← (saddr) + byte + CY	3	x	x	x
	A, r (Note 3)	A, CY ← A + r + CY	2	x	x	x
	r, A (Note 3)	r, CY ← r + A + CY	2	x	x	x
	A, saddr	A, CY ← A + (saddr) + CY	2	x	x	x
	A, !addr16	A, CY ← A + (addr16) + CY	3	x	x	x
	A, [HL]	A, CY ← A + (HL) + CY	1	x	x	x
	A, [HL + byte]	A, CY ← A + (HL + byte) + CY	2	x	x	x
	A, [HL + B]	A, CY ← A + (HL + B) + CY	2	x	x	x
	A, [HL + C]	A, CY ← A + (HL + C) + CY	2	x	x	x

Instruction Set (μPD78K0 Product Line) (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
8-Bit Operations (cont)						
SUB	A, #byte	A, CY ← A – byte	2	x	x	x
	saddr, #byte	(saddr), CY ← (saddr) – byte	3	x	x	x
	A, r (Note 3)	A, CY ← A – r	2	x	x	x
	r, A (Note 3)	r, CY ← r – A	2	x	x	x
	A, saddr	A, CY ← A – (saddr)	2	x	x	x
	A, !addr16	A, CY ← A – (addr16)	3	x	x	x
	A, [HL]	A, CY ← A – (HL)	1	x	x	x
	A, [HL + byte]	A, CY ← A – (HL + byte)	2	x	x	x
	A, [HL + B]	A, CY ← A – (HL + B)	2	x	x	x
	A, [HL + C]	A, CY ← A – (HL + C)	2	x	x	x
SUBC	A, #byte	A, CY ← A – byte – CY	2	x	x	x
	saddr, #byte	(saddr), CY ← (saddr) – byte – CY	3	x	x	x
	A, r (Note 3)	A, CY ← A – r – CY	2	x	x	x
	r, A (Note 3)	r, CY ← r – A – CY	2	x	x	x
	A, saddr	A, CY ← A – (saddr) – CY	2	x	x	x
	A, !addr16	A, CY ← A – (addr16) – CY	3	x	x	x
	A, [HL]	A, CY ← A – (HL) – CY	1	x	x	x
	A, [HL + byte]	A, CY ← A – (HL + byte) – CY	2	x	x	x
	A, [HL + B]	A, CY ← A – (HL + B) – CY	2	x	x	x
	A, [HL + C]	A, CY ← A – (HL + C) – CY	2	x	x	x
AND	A, #byte	A ← A ∧ byte	2	x		
	saddr, #byte	(saddr) ← (saddr) ∧ byte	3	x		
	A, r (Note 3)	A ← A ∧ r	2	x		
	r, A (Note 3)	r ← r ∧ A	2	x		
	A, saddr	A ← A ∧ (saddr)	2	x		
	A, !addr16	A ← A ∧ (addr16)	3	x		
	A, [HL]	A ← A ∧ (HL)	1	x		
	A, [HL + byte]	A ← A ∧ (HL + byte)	2	x		
	A, [HL + B]	A ← A ∧ (HL + B)	2	x		
	A, [HL + C]	A ← A ∧ (HL + C)	2	x		

Instruction Set (μPD78K0 Product Line) (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
8-Bit Operations (cont)						
OR	A, #byte	A ← A ∨ byte	2	x		
	saddr, #byte	(saddr) ← (saddr) ∨ byte	3	x		
	A, r (Note 3)	A ← A ∨ r	2	x		
	r, A (Note 3)	r ← r ∨ A	2	x		
	A, saddr	A ← A ∨ (saddr)	2	x		
	A, !addr16	A ← A ∨ (addr16)	3	x		
	A, [HL]	A ← A ∨ (HL)	1	x		
	A, [HL + byte]	A ← A ∨ (HL + byte)	2	x		
	A, [HL + B]	A ← A ∨ (HL + B)	2	x		
	A, [HL + C]	A ← A ∨ (HL + C)	2	x		
XOR	A, #byte	A ← A ∀ byte	2	x		
	saddr, #byte	(saddr) ← (saddr) ∀ byte	3	x		
	A, r (Note 3)	A ← A ∀ r	2	x		
	r, A (Note 3)	r ← r ∀ A	2	x		
	A, saddr	A ← A ∀ (saddr)	2	x		
	A, !addr16	A ← A ∀ (addr16)	3	x		
	A, [HL]	A ← A ∀ (HL)	1	x		
	A, [HL + byte]	A ← A ∀ (HL + byte)	2	x		
	A, [HL + B]	A ← A ∀ (HL + B)	2	x		
	A, [HL + C]	A ← A ∀ (HL + C)	2	x		
CMP	A, #byte	A – byte	2	x	x	x
	saddr, #byte	(saddr) – byte	3	x	x	x
	A, r (Note 3)	A – r	2	x	x	x
	r, A (Note 3)	r – A	2	x	x	x
	A, saddr	A – (saddr)	2	x	x	x
	A, !addr16	A – (addr16)	3	x	x	x
	A, [HL]	A – (HL)	1	x	x	x
	A, [HL + byte]	A – (HL + byte)	2	x	x	x
	A, [HL + B]	A – (HL + B)	2	x	x	x
	A, [HL + C]	A – (HL + C)	2	x	x	x
16-Bit Operations						
ADDW	AX, #word	AX, CY ← AX + word	3	x	x	x
SUBW	AX, #word	AX, CY ← AX – word	3	x	x	x
CMPW	AX, #word	AX – word	3	x	x	x
Multiplication/Division (Note 4)						
MULU	X	AX ← A x X (Note 4)	2			
DIVUW	C	AX(quotient), C (remainder) ← AX ÷ C (Note 4)	2			

Instruction Set (μPD78K0 Product Line) (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	Flags AC	Flags CY
Increment/Decrement						
INC	r	r ← r + 1	1	x	x	
	saddr	(saddr) ← (saddr) + 1	2	x	x	
DEC	r	r ← r − 1	1	x	x	
	saddr	(saddr) ← (saddr) − 1	2	x	x	
INCW	rp	rp ← rp + 1	1			
DECW	rp	rp ← rp − 1	1			
Rotate						
ROR	A, 1	(CY, A_7 ← A_0, A_{m-1} ← A_m) x 1	1			x
ROL	A, 1	(CY, A_0 ← A_7, A_{m+1} ← A_m) x 1	1			x
RORC	A, 1	(CY ← A_0, A_7 ← CY, A_{m-1} ← A_m) x 1	1			x
ROLC	A, 1	(CY ← A_7, A_0 ← CY, A_{m+1} ← A_m) x 1	1			x
ROR4	[HL]	A_{3-0} ← $(HL)_{3-0}$, $(HL)_{7-4}$ ← A_{3-0}, $(HL)_{3-0}$ ← $(HL)_{7-4}$	2			
ROL4	[HL]	A_{3-0} ← $(HL)_{7-4}$, $(HL)_{3-0}$ ← A_{3-0}, $(HL)_{7-4}$ ← $(HL)_{3-0}$	2			
BCD Adjustment						
ADJBA		Decimal adjust accumulator after addition	2	x	x	x
ADJBS		Decimal adjust acccumulator after subtraction	2	x	x	x
Bit Manipulation						
MOV1	CY, saddr.bit	CY ← (saddr.bit)	3			x
	CY, sfr.bit	CY ← sfr.bit	3			x
	CY, A.bit	CY ← A.bit	2			x
	CY, PSW.bit	CY ← PSW.bit	3			x
	CY, [HL].bit	CY ← (HL).bit	2			x
	saddr.bit, CY	(saddr.bit) ← CY	3			
	sfr.bit, CY	sfr.bit ← CY	3			
	A.bit, CY	A.bit ← CY	2			
	PSW.bit, CY	PSW.bit ← CY	3	x	x	
	[HL].bit, CY	(HL).bit ← CY	2			
AND1	CY, saddr.bit	CY ← CY ∧ (saddr.bit)	3			x
	CY, sfr.bit	CY ← CY ∧ sfr.bit	3			x
	CY, A.bit	CY ← CY ∧ A.bit	2			x
	CY, PSW.bit	CY ← CY ∧ PSW.bit	3			x
	CY, [HL].bit	CY ← CY ∧ (HL).bit	2			x
OR1	CY, saddr.bit	CY ← CY ∨ (saddr.bit)	3			x
	CY, sfr.bit	CY ← CY ∨ sfr.bit	3			x
	CY, A.bit	CY ← CY ∨ A.bit	2			x
	CY, PSW.bit	CY ← CY ∨ PSW.bit	3			x
	CY, [HL].bit	CY ← CY ∨ (HL).bit	2			x

Instruction Set (μPD78K0 Product Line) (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
Bit Manipulation (cont)						
XOR1	CY, saddr.bit	CY ← CY ∀ (saddr.bit)	3			x
	CY, sfr.bit	CY ← CY ∀ sfr.bit	3			x
	CY, A.bit	CY ← CY ∀ A.bit	2			x
	CY, PSW.bit	CY ← CY ∀ PSW.bit	3			x
	CY, [HL].bit	CY ← CY ∀ (HL).bit	2			x
SET1	saddr.bit	(saddr.bit) ← 1	2			
	sfr.bit	sfr.bit ← 1	3			
	A.bit	A.bit ← 1	2			
	PSW.bit	PSW.bit ← 1	2	x	x	x
	[HL].bit	(HL).bit ← 1	2			
	CY	CY ← 1	1			1
CLR1	saddr.bit	(saddr.bit) ← 0	2			
	sfr.bit	sfr.bit ← 0	3			
	A.bit	A.bit ← 0	2			
	PSW.bit	PSW.bit ← 0	2	x	x	x
	[HL].bit	(HL).bit ← 0	2			
	CY	CY ← 0	1			0
NOT1	CY	CY ← $\overline{CY}$	1			x
Call/Return						
CALL	!addr16	(SP– 1) ← $(PC + 3)_H$, (SP – 2) ← $(PC + 3)_L$, PC ← addr16, SP ← SP – 2	3			
CALLF	!addr11	(SP – 1) ← $(PC + 2)_H$, (SP – 2) ← $(PC + 2)_L$, PC_{15-11} ← 00001, PC_{10-0} ← addr11, SP ← SP – 2	2			
CALLT	[addr5]	(SP – 1) ← $(PC + 1)_H$, (SP – 2) ← $(PC + 1)_L$, PC_H ← (00000000, addr5 + 1), PC_L ← (00000000, addr5), SP← SP – 2	1			
BRK		(SP – 1) ← PSW, (SP – 2) ← $(PC + 1)_H$, (SP – 3) ← $(PC + 1)_L$, PC_H ← (003FH), PC_L ← (003EH), SP ← SP – 3, IE ← 0	1			
RET		PC_L ← (SP), PC_H ← (SP + 1), SP ← SP + 2	1			
RET1		PC_L ← (SP), PC_H ← (SP + 1), PSW ← (SP + 2), SP ← SP + 3	1	R	R	R
RETB		PC_L ← (SP), PC_H ← (SP + 1), PSW ←(SP + 2), SP ← SP + 3	1	R	R	R
Stack Manipulation						
PUSH	PSW	(SP – 1) ← PSW, SP ← SP – 1	1			
	rp	(SP – 1) ← rp_H, (SP – 2) ← rp_L, SP ← SP – 2	1			
POP	PSW	PSW ← (SP), SP ← SP + 1	1	R	R	R
	rp	rp_L ← (SP), rp_H ← (SP +1), SP ← SP + 2	1			

Instruction Set (μPD78K0 Product Line) (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
Stack Manipulation (cont)						
MOVW	SP, #word	SP ← word	4			
	SP, AX	SP ← AX	2			
	AX, SP	AX ← SP	2			
Unconditional Branch						
BR	!addr16	PC ← addr16	3			
	$addr16	PC ← PC + 2 + jdisp8	2			
	AX	PC_H ← A, PC_L ← X	2			
Conditional Branch						
BC	$addr16	PC ← PC + 2 + jdisp8 if CY = 1	2			
BNC	$addr16	PC ← PC + 2 + jdisp8 if CY = 0	2			
BZ	$addr16	PC ← PC + 2 + jdisp8 if Z = 1	2			
BNZ	$addr16	PC ← PC + 2 + jdisp8 if Z = 0	2			
BT	saddr.bit, $addr16	PC ← PC + 3 + jdisp8 if (saddr.bit) = 1	3			
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 1	4			
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 1	3			
	PSW.bit, $addr16	PC ← PC + 3 + jdisp8 if PSW.bit = 1	3			
	[HL].bit, $addr16	PC ← PC + 3 + jdisp8 if (HL).bit = 1	3			
BF	saddr.bit, $addr16	PC ← PC + 4 + jdisp8 if (saddr.bit) = 0	4			
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 0	4			
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 0	3			
	PSW.bit, $addr16	PC ← PC + 4 + jdisp8 if PSW.bit = 0	4			
	[HL].bit, $addr16	PC ← PC + 3 + jdisp8 if (HL).bit = 0	3			
BTCLR	saddr.bit, $addr16	PC ← PC + 4 + jdisp8 if (saddr.bit) = 1 then reset (saddr.bit)	4			
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit	4			
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 1 then reset A.bit	3			
	PSW.bit, $addr16	PC ← PC + 4 + jdisp8 if PSW.bit = 1 then reset PSW.bit	4	x	x	x
	[HL].bit, $addr16	PC ← PC + 3 + jdisp8 if (HL).bit = 1 then reset (HL).bit	3			
DBNZ	B, $addr16	B ← B − 1, then PC ← PC + 2 + jdisp8 if B ≠ 0	2			
	C, $addr16	C ← C − 1, then PC ← PC + 2 + jdisp8 if C ≠ 0	2			
	saddr, $addr16	(saddr) ← (saddr) − 1, then PC ← PC + 3 + jdisp8 if (saddr) ≠ 0	3			

Instruction Set (μPD78K0 Product Line) (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
CPU Control						
SEL	RBn	RBS1-0 ← n, n = 0-3	2			
NOP		No Operation	1			
EI		IE ← 1 (Enable Interrupt)	2			
DI		IE ← 0 (Disable Interrupt)	2			
HALT		Set HALT mode	2			
STOP		Set STOP mode	2			

Notes:

(1) Except r = A. The chip will not operate normally with r = A.

(2) Only for rp = BC, DE, or HL

(3) When r = A and the operand A, A is assembled, the operand format r, A is selected by the assembler and generates an operand A, A. The operand A, A will execute correctly in the chip.

(4) These instructions not available in μPD78002 and μPD78002Y families.

Special Function Registers (μPD78K0 Product Line)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits)			State	Family						
				1	8	16	After Reset	02	02Y	14	14Y	44	54	64
FF00H	Port 0	P0	R/W	x	x	—	00H	x	x	x	x	x	x	x
FF01H	Port 1	P1	R/W	x	x	—	00H	x	x	x	x	x	x	x
FF02H	Port 2	P2	R/W	x	x	—	00H	x	x	x	x	x	x	x
FF03H	Port 3	P3	R/W	x	x	—	00H	x	x	x	x	x	x	x
FF04H	Port 4	P4	R/W	x	x	—	Undefined	x	x	x	x		x	
FF05H	Port 5	P5	R/W	x	x	—	Undefined	x	x	x	x		x	
FF06H	Port 6	P6	R/W	x	x	—	Undefined	x	x	x	x		x	
FF07H	Port 7	P7	R/W	x	x	—	00H					x	x	x
FF08H	Port 8 (78044) Port 8 (78064)	P8	W R/W	x x	x x	—	00H 00H					x		 x
FF09H	Port 9 (78044) Port 9 (78064)	P9	W R/W	x x	x x	—	00H 00H					x		 x
FF0AH	Port 10 (78044) Port 10 (78064)	P10	W R/W	x x	x x	—	00H 00H					x		 x
FF0BH	Port 11	P11	R/W	x	x	—	00H					x		x
FF0CH	Port 12	P12	R/W	x	x	—	00H					x	x	
FF0DH	Port 13	P13	R/W	x	x	—	00H						x	
FF10H FF11H	Compare reg 00 (78014, 14Y, 44) Capture/compare register 00 (78054, 64)	CR00 CR00	R/W R/W	— —	— —	x x	Undefined Undefined			x	x	x	 x	 x
FF12H FF13H	Compare reg 01 (014, 014Y, 044) Capture/compare register 01 (78054,064)	CR01 CR01	R R/W	— —	— —	x x	Undefined Undefined			x	x	x	 x	 x
FF14H FF15H	16-bit timer register	TM0	R	—	—	x	00H			x	x	x	x	x
FF16H	Compare register 10	CR10	R/W	—	x	—	Undefined	x	x	x	x	x	x	x
FF17H	Compare register 20	CR20	R/W	—	x	—	Undefined	x	x	x	x	x	x	x
FF18H FF19H FF18H FF19H	8-bit timer register 1 8-bit timer register 2 16-bit timer register	TM1 TM2 TMS	R R R	x x —	x x —	— — x	00H 00H 0000H	x x x	x x x	x x x	x x x	x x x	x x x	x x x
FF1AH	Serial I/O shift register 0	SIO0	R/W	—	x	—	Undefined	x	x	x	x	x	x	x
FF1BH	Serial I/O shift register 1	SIO1	R/W	—	x	—	Undefined			x	x	x	x	
FF1FH	A/D conversion result register	ADCR	R	—	x	—	Undefined			x	x	x	x	x
FF20H	Port mode register 0	PM0	R/W	x	x	—	1FH FFH	x	x	x	x	x	 x	 x
FF21H	Port mode register 1	PM1	R/W	x	x	—	FFH	x	x	x	x	x	x	x
FF22H	Port mode register 2	PM2	R/W	x	x	—	FFH	x	x	x	x	x	x	x
FF23H	Port mode register 3	PM3	R/W	x	x	—	FFH	x	x	x	x	x	x	x
FF25H	Port mode register 5	PM5	R/W	x	x	—	FFH	x	x	x	x		x	
FF26H	Port mode register 6	PM6	R/W	x	x	—	FFH	x	x	x	x		x	

Special Function Registers (μPD78K0 Product Line) (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits)			State After Reset	Family						
				1	8	16		02	02Y	14	14Y	44	54	64
FF27H	Port mode register 7	PM7	R/W	x	x	—	1FH (44) FFH (54/64)					x	x	x
FF28H	Port mode register 8	PM8	R/W	x	x	—	FFH							x
FF29H	Port mode register 9	PM9	R/W	x	x	—	FFH							x
FF2AH	Port mode register 10	PM10	R/W	x	x	—	FFH							x
FF2BH	Port mode register 11	PM11	R/W	x	x	—	FFH					x		x
FF2CH	Port mode register 12	PM12	R/W	x	x	—	FFH					x	x	
FF2DH	Port mode register 13	PM13	R/W	x	x	—	FFH						x	
FF30H	Real-time buffer register L	RTBL	R/W	—	x	—	00H						x	
FF31H	Real-time buffer register H	RTBH	R/W	—	x	—	00H						x	
FF34H	Real-time output port mode register	RTPM	R/W	x	x	—	00H						x	
FF36H	Real-time output port control register	RTPC	R/W	x	x	—	00H						x	
FF40H	Timer clock select register 0	TCL0	R/W	x	x	—	00H	x	x	x	x	x	x	x
FF41H	Timer clock select register 1	TCL1	R/W	—	x	—	00H	x	x	x	x	x	x	x
FF42H	Timer clock select register 2	TCL2	R/W	—	x	—	00H	x	x	x	x	x	x	x
FF43H	Timer clock select register 3	TCL3	R/W	—	x	—	88H	x	x	x	x	x	x	x
FF47H	Sampling clock select register	SCS	R/W	—	x	—	00H	x	x	x	x	x	x	x
FF48H	16-bit timer mode control register	TMC0	R/W	x	x	—	00H			x	x	x	x	x
FF49H	8-bit timer mode control register	TMC1	R/W	x	x	—	00H	x	x	x	x	x	x	x
FF4AH	Watch (clock) timer mode control register	TMC2	R/W	x	x	—	00H	x	x	x	x	x	x	x
FF4CH	Capture/compare control register 0	CRC0	R/W	x	x	—	04H						x	x
FF4EH	16-bit timer output control register	TOC0	R/W	x	x	—	00H			x	x	x	x	x
FF4FH	8-bit timer output control register	TOC1	R/W	x	x	—	00H	x	x	x	x	x	x	x
FF60H	Serial operating mode register 0	CSIM0	R/W	x	x	—	00H	x	x	x	x	x	x	x
FF61H	Serial bus interface control register	SBIC	R/W	x	x	—	00H	x	x	x	x	x	x	x
FF62H	Slave address register	SVA	R/W	—	x	—	Undefined	x	x	x	x	x	x	x
FF63H	Interrupt timing specify register	SINT	R/W	x	x	—	00H	x	x	x	x	x	x	x
FF68H	Serial operation mode register 1	CSIM1	R/W	x	x	—	00H			x	x	x	x	
FF69H	Automatic data transmit/ receive control register	ADTC	R/W	x	x	—	00H			x	x	x	x	
FF6AH	Automatic data transmit/ receive address pointer register	ADTP	R/W	—	x	—	00H			x	x	x	x	

Special Function Registers (μPD78K0 Product Line) (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits)			State After Reset	Family						
				1	8	16		02	02Y	14	14Y	44	54	64
FF6BH	Automatic data transmit/ receive interval specification register	ADTI	R/W	x	x	—	00H					x	x	
FF70H	Asynchronous serial interface mode register	ASIM	R/W	x	x	—	00H						x	x
FF71H	Asynchronous serial interface status register	ASIS	R	x	x	—	00H						x	x
FF72H	Serial operating mode register 2	CSIM2	R/W	x	x	—	00H						x	x
FF73H	Baud rate generator control register	BRGC	R/W	—	x	—	00H						x	x
FF74H	Transmit shift register Receive buffer register Serial I/O shift register	TXS RXB SIO2 (Note 1)	W R R/W	— — —	x x x	— — —	FFH FFH FFH						x x x	x x x
FF80H	A/D converter mode register	ADM	R/W	x	x	—	01H			x	x	x	x	x
FF84H	A/D converter input select register	ADIS	R/W	—	x	—	00H			x	x	x	x	x
FF90H	D/A conversion value register 0	DACS0	R/W	—	x	—	00H						x	
FF91H	D/A conversion value register 1	DACS1	R/W	—	x	—	00H						x	
FF98H	D/A converter mode register	DAM	R/W	x	x	—	00H						x	
FFA0H	Display mode register 0	DSPM0	R/W	x (Note 2)	x (Note 2)	—	00H					x		
FFA1H	Display mode register 1	DSPM1	R/W	—	x	—	00H					x		
FFA8H	6-bit up/down counter mode register	UDM	R/W	x	x	—	00H					x		
FFA9H	6-bit up/down counter	UDC	R/W	—	x	—	00H					x		
FFAAH	6-bit up/down counter compare register	UDCC	R/W	—	x	—	00H					x		
FFB0H	LCD display mode register	LCDM	R/W	x	x	—	00H							x
FFB2H	LCD display control register	LCDC	R/W	x	x	—	00H							x
FFB8H	Key return mode register	KRM	R/W	x	x	—	02H							x
FFD0H FFDFH	External SFR access area (Note 3)	—	R/W	x	x	—	Undefined	x	x	x	x		x	
FFE0H FFE1H FFE0H FFE1H	Interrupt flag register L Interrupt flag register H Interrupt flag register	IFOL IF0H IFO	R/W R/W R/W	x x —	x x —	— — x	00H 00H 0000H	x x x	x x x	x x x	x x x	x x x	x x x	x x x
FFE2H	Interrupt request flag register 1L	IF1L	R/W	x	x	—	00H						x	x

3h

Special Function Registers (μPD78K0 Product Line) (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits)			State After Reset	Family						
				1	8	16		02	02Y	14	14Y	44	54	64
FFE4H	Interrupt mask flag register L	MKOL	R/W	x	x	—	FFH	x	x	x	x	x	x	x
FFE5H	Interrupt mask flag register H	MK0H	R/W	x	x	—	FFH	x	x	x	x	x	x	x
FFE4H FFE5H	Interrupt mask flag register	MKO	R/W	—	—	x	FFFFH	x	x	x	x	x	x	x
FFE6H	Interrupt mask flag register 1L	MK1L	R/W	x	x	—	FFH						x	x
FFE8H	Priority order specify flag register L	PR0L	R/W	x	x	—	FFH	x	x	x	x	x	x	x
FFE9H	Priority order specify flag register H	PR0H	R/W	x	x	—	FFH	x	x	x	x	x	x	x
FFE8H FFE9H	Priority order specify flag register	PR0	R/W	—	—	x	FFFFH	x	x	x	x	x	x	x
FFEAH	Priority specification flag register 1L	PR1L	R/W	x	x	—	FFH						x	x
FFECH	External interrupt mode register	INTM0	R/W	—	x	—	00H	x	x	x	x	x	x	x
FFEDH	External interrupt mode register 1	INTM1	R/W	—	x	—	00H						x	x
FFF0H	Memory size switch register	IMS	W	—	x	—	(Note 4)			x	x	x	x	x
FFF2H	Oscillation mode select register	OSMS	W	x	x	—	00H						x	x
FFF3H	Pullup resistor option register H	PUOH	R/W	x	x	—	00H						x	x
FFF6H	Key return mode register	KRM	R/W	x	x	—	02H	x	x	x	x		x	
FFF7H	Pullup resistor option register	PUO PUOL	R/W R/W	x x	x x	—	00H 00H	x	x	x	x	x	 x	 x
FFF8H	Memory expanded mode register	MM	R/W	x	x	—	10H	x	x	x	x		x	
FFF9H	Watchdog timer mode register	WDTM	R/W	x	x	—	00H	x	x	x	x	x	x	x
FFFAH	Oscillation stabilization time select register	OSTS	R/W	—	x	—	04H	x	x	x	x	x	x	x
FFFBH	Processor clock control register	PCC	R/W	x	x	—	04H	x	x	x	x	x	x	x

Notes:

(1) SIO2 can be used instead of TXS and RXB. SIO2 is not a register; it is another symbol that can be used to reference the TXS and RXB registers. A write to SIO2 causes the CPU to write to the TXS register, and a read from SIO2 causes the CPU to read the RXB register.

(2) Bit 7 only can be manipulated as a read only bit.

(3) The external access area cannot be accessed using SFR addressing. It can only be accessed using main memory addressing.

(4) The value in the memory size switch register (IMS) after reset depends upon the ROM size. See User's Manual for the value. This register is available in all mask ROM, OTP, and EPROM devices.

Interrupt Vectors and Test Inputs (μPD78K0 Product Line)

Type of Request	Interrupt Source	Signal Name	Request Flag	Mask Flag	Priority Flag	Location	Interrupt Vector Address for Listed Functions				
							78002 78002Y	78014 78014Y	78044	78054	78064
Restart	RESET input or watchdog timer overflow when reset mode selected	RESET INTWDT	— —	— —	— —	External Internal	0000 0000	0000 0000	0000 0000	0000 0000	0000 0000
Non maskable	Watchdog timer overflow when NMI mode selected	INTWDT	TMIF4	—	—	Internal	0004	0004	0004	0004	0004
Maskable (Note 1)	Watchdog timer overflow when interval timer mode selected	INTWDT	TMIF4	TMMK4	TMPR4	Internal	0004	0004	0004	0004	0004
	External interrupt edge detection	INTP0	PIF0	PMK0	PPR0	External	0006	0006	0006	0006	0006
	External interrupt edge detection	INTP1	PIF1	PMK1	PPR1	External	0008	0008	0008	0008	0008
	External interrupt edge detection	INTP2	PIF2	PMK2	PPR2	External	000A	000A	000A	000A	000A
	External interrupt edge detection or up/down coincidence signal	INTP3 INTUD	PIF3 PIF3	PMK3 PMK3	PPR3 PPR3	External Internal	000C —	000C —	000C 000C	000C —	000C —
	External interrupt edge detection	INTP4	PIF4	PMK4	PPR4	External	—	—	—	000E	000E
	External interrupt edge detection	INTP5	PIF5	PMK5	PPR5	External	—	—	—	0010	0010
	External interrupt edge detection	INTP6	PIF6	PMK6	PPR6	External	—	—	—	0012	—
	End of clocked serial interface 0 transfer	INTCSI0	CSIIF0	CSIMK0	CSIPR0	Internal	000E	000E	000E	0014	0014
	End of clocked serial interface 1 transfer	INTCSI1	CSIIF1	CSIMK1	CSIPR1	Internal	—	0010	0010	0016	—
	Serial interface 2 UART reception	INTSER	SERIF	SERMK	SERPR	Internal	—	—	—	0018	0018
	End of serial interface 2 UART reception	INTSR	SRIF	SRMK	SRPR	Internal	—	—	—	001A	001A
	End of serial interface 2 three-wire transfer	INTCSI2	SRIF	SRMK	SRPR	Internal	—	—	—	001A	001A
	End of serial interface 2 UART transmission	INTST	STIF	STMK	STPR	Internal	—	—	—	001C	001C
	Watch (Clock) timer reference time interval signal	INTTM3	TMIF3	TMMK3	TMPR3	Internal	0012	0012	0012	001E	001E

3h

Interrupt Vectors and Test Inputs (μPD78K0 Product Line) (cont)

Type of Request	Interrupt Source	Signal Name	Request Flag	Mask Flag	Priority Flag	Location	Interrupt Vector Address for Listed Functions				
							78002 78002Y	78014 78014Y	78044	78054	78064
Maskable (Note 1) (cont)	16-bit timer/event counter coincidence signal	INTTM0	TMIF0	TMMK0	TMPR0	Internal	—	0014	0014	—	—
	16-bit timer/event counter capture/ compare (CR00)	INTTM00	TMIF00	TMMK00	TMPR00	Internal	—	—	—	0020	0020
	16-bit timer/event counter capture/ compare (CR01)	INTTM01	TMIF01	TMMK01	TMPR01	Internal	—	—	—	0022	0022
	8-bit timer/event counter 1 coincidence signal	INTTM1	TMIF1	TMMK1	TMPR1	Internal	0016	0016	0016	0024	0024
	8-bit timer/event counter 2 coincidence signal	INTTM2	TMIF2	TMMK2	TMPR2	Internal	0018	0018	0018	0026	0026
	End of A/D conversion	INTAD	ADIF	ADMK	ADPR	Internal	—	001A	001A	0028	0028
	Key scan interrupt generated by FIP controller	INTKS	KSIF	KSMK	KSPR	Internal	—	—	001C	—	—
Software	BRK instruction	—	—	—	—	Internal	003E	003E	003E	003E	003E
Test signals (Note 2)	Watch (Clock) timer overflow	INTWT	WTIF	WTMK	—	Internal	test	test	test	test	test
	Port 4 falling edge detect	INTPT4	KRIF	KRMK	—	External	test	test	—	test	—
	Port 11 falling edge detect	INTPT11	KRIF	KRMK	—	External	—	—	—	—	test

Notes:

(1) Maskable interrupts are shown in order of descending default priority.

(2) Test signals do not generate vectored interrupts but may release standby modes. Code following a STOP or HALT operation should perform a test on the flag to determine if a test input released the standby mode.

Section 4
μPD78K2 Product Line
8-Bit, K-Series Microcontrollers

NEC
NEC Electronics Inc.

μPD78214 Family (μPD78212/213/214/P214) 8-Bit, K-Series Microcontrollers With A/D Converter, Real-Time Output Ports

July 1993

Description

The μPD78212, μPD78213, μPD78214, and μPD78P214 are members of the K-Series® of microcontrollers and are designed for real-time embedded control applications. These 8-bit, single-chip microcontrollers have a minimum instruction time of 333 ns at 12 MHz (500 ns for the μPD78213). They feature 8-bit hardware multiply and divide instructions, four banks of main registers, an advanced interrupt handling facility, a powerful set of memory mapped on-chip peripherals, and the ability to address up to 1M bytes of external data memory. On-board memory includes 384 or 512 bytes of RAM, 8K or 16K bytes of mask ROM, or 16K bytes of UV EPROM or one-time programmable (OTP) ROM.

The advanced interrupt handling facility provides two levels of programmable hardware priority control and two separate methods of servicing interrupt requests: vectored and macro service. The macro service facility reduces the overhead involved in servicing peripheral interrupts by transferring data between the memory-mapped special function registers (SFRs) and memory without the use of time consuming interrupt service routines. In addition, the macro service facility can be initialized to automatically alter timer compare register values or to repeatedly output a prespecified pattern at a fixed or variable rate. By using macro service to control the real-time output ports, the μPD78214 family can easily and accurately drive two independent stepper motors.

The combination of the macro service facility, four banks of main registers, extended data memory address space, and powerful on-chip peripherals makes these devices ideal for applications in office automation, communication, HVAC, and industrial control.

Features

- Complete single-chip microcontroller
  - 8-bit ALU
  - Program memory (ROM)
    - μPD78213: ROMless
    - μPD78212: 8K bytes
    - μPD78214/P214: 16K bytes
  - Data memory (RAM)
    - μPD78212: 384 bytes
    - μPD78213/214/P214: 512 bytes
- Powerful instruction set
  - 8-bit unsigned multiply and divide
  - 16-bit arithmetic instructions
  - 1-bit and 8-bit logic instructions
- Minimum instruction time
  - 333 ns at 12 MHz (μPD78212/214/P214)
  - 500 ns at 12 MHz (μPD78213)
- Memory expansion
  - 8085 bus-compatible
  - 64K program address space
  - 1M data address space
- Large I/O capacity
  - Up to 54 I/O port lines on μPD78212/214/P214
  - Up to 36 I/O port lines on μPD78213
  - Software programmable pullup resistors
- Memory-mapped on-chip peripherals (special function registers)
- Timer/counter unit
  - 16-bit timer 0:
    - Two 16-bit compare registers
    - One 16-bit capture register
    - One external interrupt/capture line
  - 8-bit timer 1:
    - One 8-bit compare register
    - One 8-bit capture/compare register
    - One external interrupt/capture line
  - 8-bit timer/counter 2:
    - Two 8-bit compare registers
    - One 8-bit capture register
    - One external interrupt/capture line
    - One external event counter line
  - 8-bit timer 3:
    - One 8-bit compare register
- Four 8-bit precision timer-controlled pulse-width modulated (PWM) output lines
- Two 4-bit (or one 8-bit) real-time output ports
- Eight-channel 8-bit A/D converter
- Programmable priority interrupt controller (two levels)
- Two methods of interrupt service
  - Vectored interrupts
  - Macro service mode with choice of three different types

4a

K-Series is a registered trademark of NEC Electronics, Inc.

Features (cont)

- Two-channel serial communication interface
  - — Asynchronous serial interface (UART)
    Dedicated baud rate generator
  - — Clock-synchronized interface
    Full-duplex, three-wire mode
    NEC serial bus interface (SBI) mode
- Refresh output for pseudostatic RAM
- STOP and HALT standby functions
- 5-volt CMOS technology

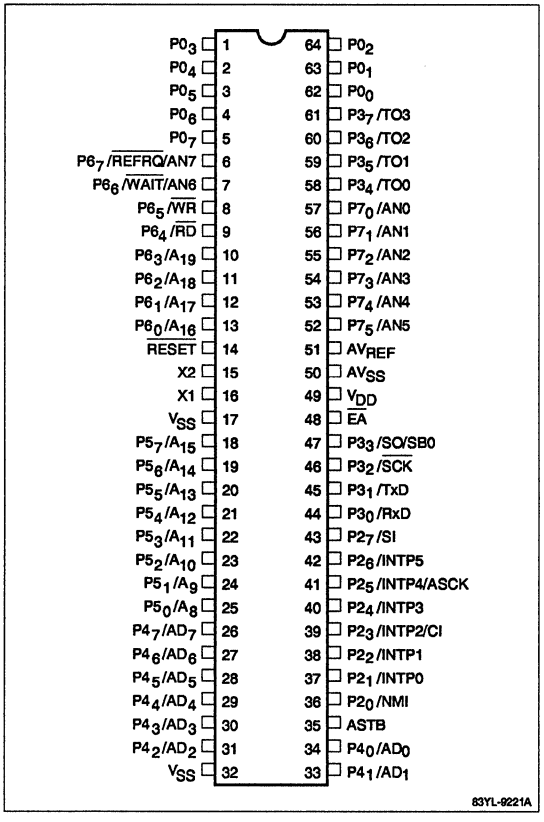
Ordering Information

Part Number	ROM	Package	Package Drawing
μPD78212CW-xxx	8K mask ROM	64-pin plastic shrink DIP	P64C-70-750A, C
μPD78213CW	ROMless		
μPD78214CW-xxx	16K mask ROM		
μPD78P214CW	16K OTP ROM		
μPD78212GC-xxx	8K mask ROM	64-pin plastic QFP	P64GC-80-AB8-2
μPD78213GC	ROMless		
μPD78214GC-xxx	16K mask ROM		
μPD78P214GC	16K OTP ROM		
μPD78212GJ-xxx	8K mask ROM	74-pin plastic QFP	S74GJ-100-5BJ-1
μPD78213GJ	ROMless		
μPD78214GJ-xxx	16K mask ROM		
μPD78P214GJ	16K OTP ROM		
μPD78213G36	ROMless	64-pin plastic QUIP	P64GQ-100-36
μPD78214Gxxx36	16K mask ROM		
μPD78P214GQ	16K OTP ROM		
μPD78213L	ROMless	68-pin PLCC	P68L-50A1-1
μPD78214L-xxx	16K mask ROM		
μPD78P214L	16K OTP ROM		
μPD78P214DW	16K UV EPROM	64-pin shrink cerdip w/window	P64DW-70-750A1

xxx indicates ROM code suffix

Pin Configurations

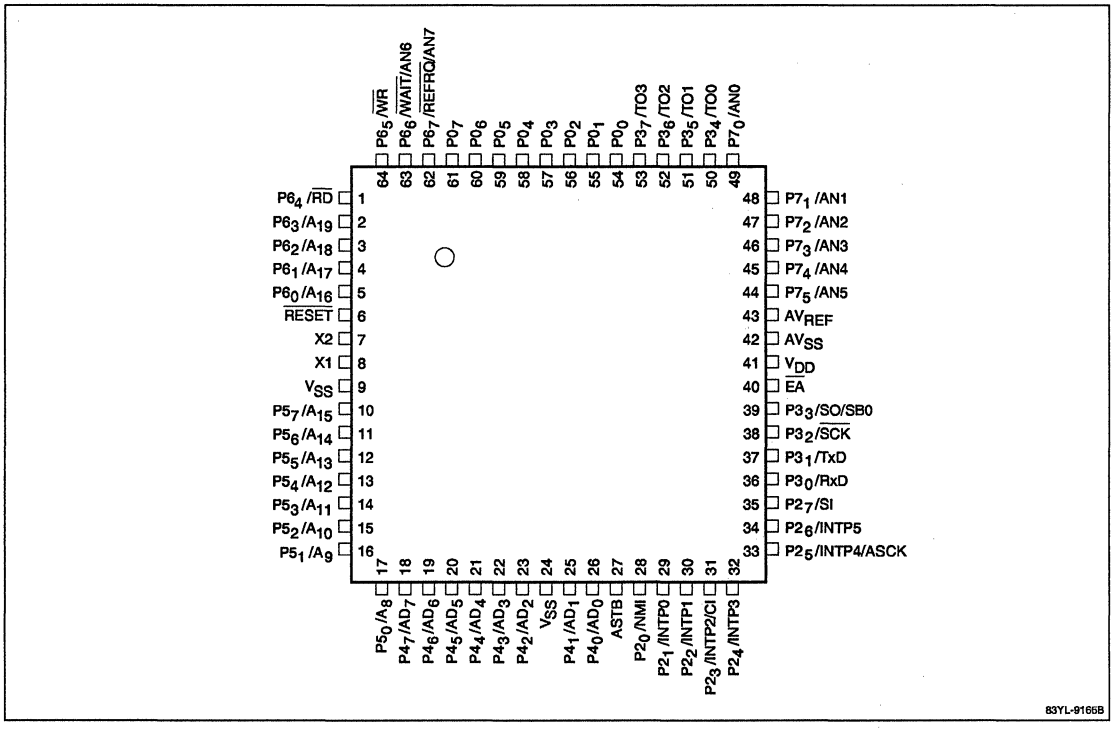
64-Pin Shrink DIP (Plastic or Ceramic) or 64-Pin Plastic QUIP



4a

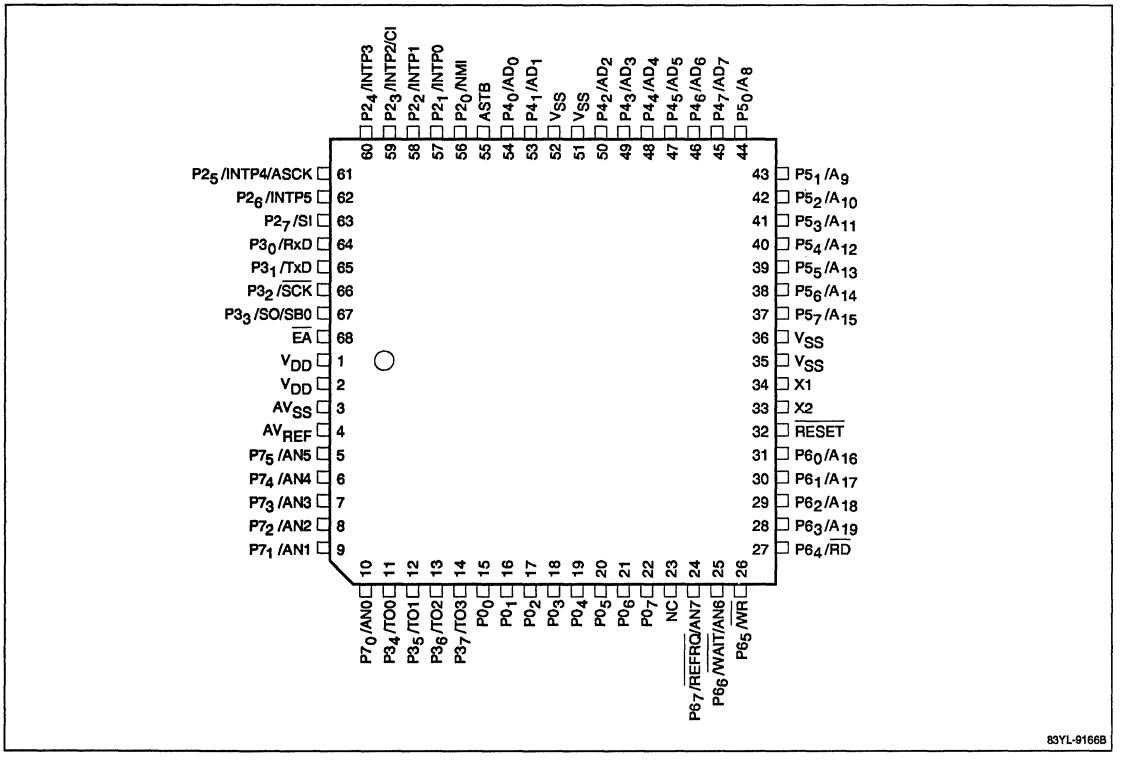
Pin Configurations (cont)

64-Pin Plastic QFP



Pin Configurations (cont)

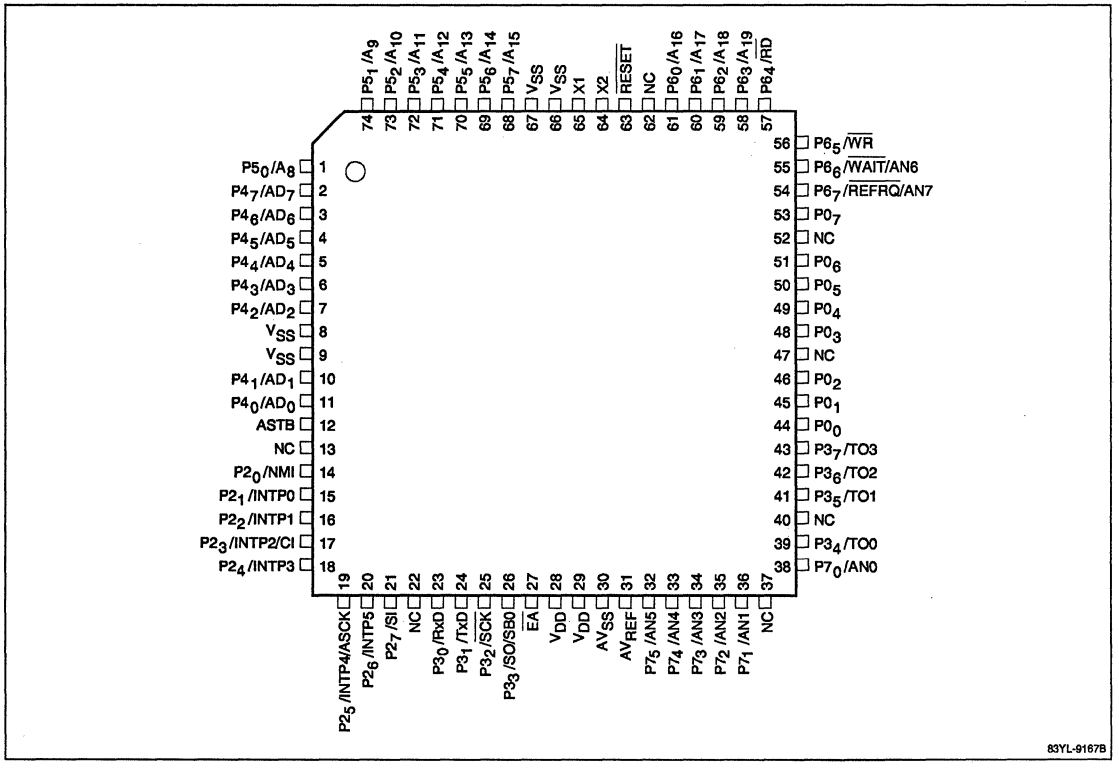
68-Pin PLCC



4a

Pin Configurations (cont)

74-Pin Plastic QFP

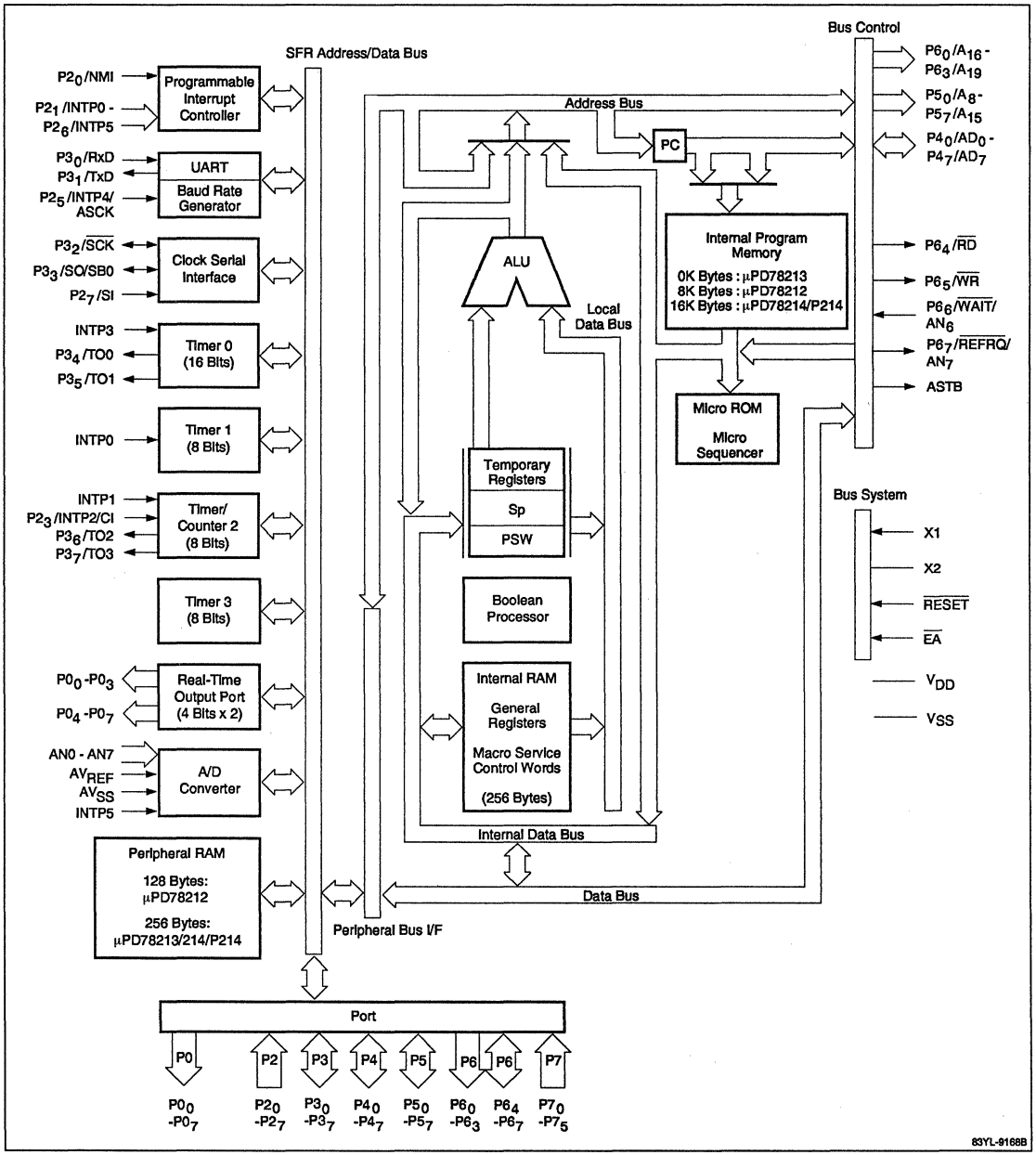


Pin Functions; Normal Operating Mode

Symbol	First Function	Symbol	Second Function
$P0_0$ - $P0_7$	Port 0; 8-bit tristate output port/real time output port		
$P2_0$	Port 2; 8-bit input port	NMI	External nonmaskable interrupt
$P2_1$		INTP0	Maskable external interrupts
$P2_2$		INTP1	
$P2_3$		INTP2	Maskable external interrupt
		CI	External clock input to timer/counter 2
$P2_4$		INTP3	Maskable external interrupt
$P2_5$		INTP4	Maskable external interrupt
		ASCK	Asynchronous serial clock input
$P2_6$		INTP5	Maskable external interrupt
$P2_7$		SI	Serial data input for three-wire serial I/O mode
$P3_0$	Port 3; 8-bit, bit-selectable tristate input/output port	RxD	Asynchronous serial receive data input
$P3_1$		TxD	Asynchronous serial transmit data output
$P3_2$		$\overline{SCK}$	Serial shift clock input/output
$P3_3$		SO	Serial data output for three-wire serial I/O mode
		SB0	I/O bus for NEC serial bus interface (SBI)
$P3_4$ - $P3_7$		TO0 - TO3	Timers T0 to T3 outputs
$P4_0$ - $P4_7$	Port 4; 8-bit tristate input/output port	AD_0 - AD_7	Low-order 8-bit multiplexed address/data bus
$P5_0$ - $P5_7$	Port 5; 8-bit, bit-selectable tristate input/output port	A_8 - A_{15}	High-order 8-bit address bus
$P6_0$ - $P6_3$	Port 6; 4-bit output port	A_{16} - A_{19}	Extended memory address bus
$P6_4$	Port 6; 4-bit, bit-selectable tristate input/output port	$\overline{RD}$	External memory read strobe output
$P6_5$		$\overline{WR}$	External memory write strobe output
$P6_6$		$\overline{WAIT}$	External memory wait signal input
		AN6	Analog voltage input to A/D converter
$P6_7$		$\overline{REFRQ}$	Refresh pulse output used by external pseudostatic memory
		AN7	Analog voltage input to A/D converter
$P7_0$ - $P7_5$	Port 7; 6-bit input port	AN0 - AN5	Analog voltage inputs to A/D converter
ASTB	Address strobe output used to latch the low-order 8 address for external memory		
$\overline{RESET}$	External system reset input		
$\overline{EA}$	Internal ROM or external memory control signal input. Low-level input selects external memory. High-level input selects internal ROM. A low-level input on a μPD78214 places the device in ROMless mode and external memory is accessed.		
X1	Crystal/ceramic resonator connection or external clock input		
X2	Crystal/ceramic resonator connection or inverse of external clock		
AV_{REF}	A/D converter reference voltage		
AV_{SS}	A/D converter ground		
V_{DD}	+5 volt power supply input		
V_{SS}	Power supply ground		
NC	No connection		

4a

Block Diagram



FUNCTIONAL DESCRIPTION

Central Processing Unit (CPU)

The μPD78214 family CPU features 8- and 16-bit arithmetic including an 8 x 8-bit unsigned multiply and 16 x 8-bit unsigned divide (producing a 16-bit quotient and an 8-bit remainder). The multiply executes in 3.67 μs and the divide in 12.36 μs at 12 MHz (4.00 and 12.69 μs respectively for μPD78213).

A CALLT vector table and a CALLF program area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction (CALLT) can access up to 32 subroutines through the addresses contained in the CALLT vector table. A 2-byte call instruction (CALLF) can access any routine beginning at a specific address in the CALLF area.

The internal system clock (f_{CLK}) is generated by dividing the oscillator frequency by two. Therefore, at the maximum oscillator frequency of 12 MHz, the internal system clock is 6 MHz. The minimum instruction execution time for an instruction fetched from internal ROM is 333 ns (500 ns when fetched from external memory).

Memory Space

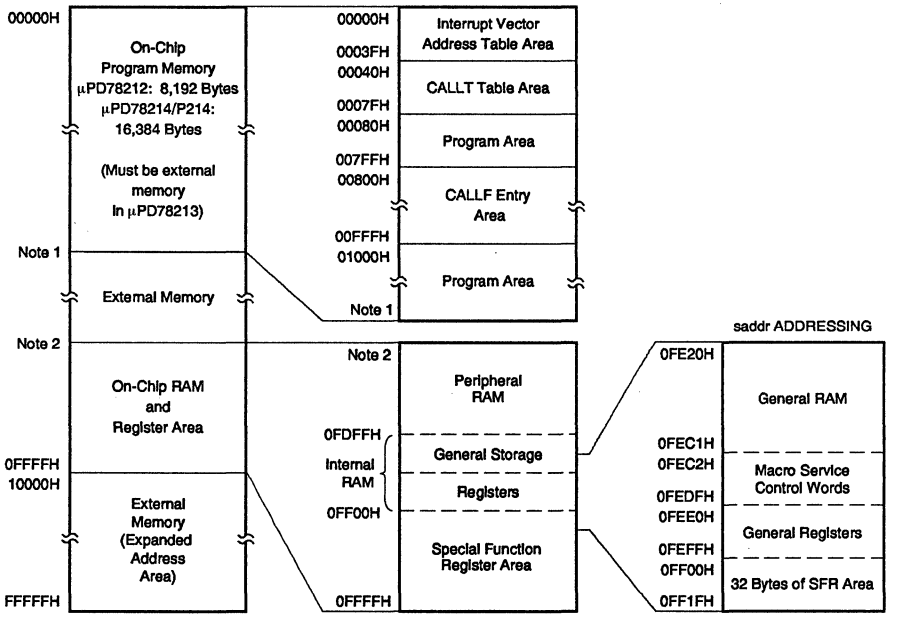
The μPD78214 family has a 1M byte address space (see figure 1). The first 64K bytes of this address space (00000H-0FFFFH) can be used as both program and data memory. The remaining 960K bytes of this address space (10000H-FFFFFH) can only be used as data memory and is known as expanded memory.

External Memory

The μPD78214 family has an 8-bit wide external data bus and a 16-bit wide external address bus (20-bit wide if expanded memory is enabled). The low-order 8 bits of the address bus are multiplexed to provide the 8-bit data bus and are supplied by I/O port 4. The high-order

4a

Figure 1. Memory Map



address bits of the 16-bit address bus are taken from port 5. If expanded memory is enabled, the expanded address nibble is provided by $P6_0$ to $P6_3$. Address latch, read, and write strobes are also provided.

The memory expansion mode register (MM) is used to enable external memory, to specify up to two additional wait states or the use of the $\overline{WAIT}$ input pin for the first 64K bytes of memory, and to enable the high-speed internal ROM fetch. Ports 4, 5, and 6 are available as general purpose I/O ports when only internal ROM is used and no external program or data space is required.

Expanded Data Memory

The MM register is also used to enable the external expanded data memory space, addresses 10000H to FFFFFH. When the expanded data memory is enabled, the entire 1M byte address space is divided into 16 banks of 64K bytes each. The low-order 4-bits of the P6 or the PM6 registers are used as bank selection registers to supply the address information to A_{16} to A_{19}. Data can easily be transferred from one memory bank to another by using the appropriate instructions. Address lines A_{16} to A_{19} are only active when an instruction that uses expanded addressing is being executed. A programmable wait control register (PW) allows the programmer to specify up to two additional wait states or the use of the $\overline{WAIT}$ input pin for expanded data memory space.

On-Chip RAM

The μPD78213/214/P214 have a total of 512 bytes of on-chip RAM (384 bytes in the μPD78212). The upper 256-byte area (FE00H-FEFFH) features high-speed access and is known as "Internal RAM." The remainder (FD00H-FDFFH and FD80H-FDFFH in the μPD78212) is accessed at the same speed as external memory and is known as "Peripheral RAM." The general register banks and the macro service control words are stored in Internal RAM. The remainder of Internal RAM and any unused register bank locations are available for general storage.

On-Chip Program Memory

The μPD78212 and μPD78214 contain 8K and 16K bytes of internal ROM respectively. The μPD78P214 contains 16K bytes of UV EPROM or one-time programmable ROM. Instructions from on-chip program memory can be fetched at high speed or at the same rate as from external memory. The μPD78213 does not have on-chip program memory.

CPU Control Registers

Program Counter. The program counter is a 16-bit binary counter register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 00001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

Program Status Word. The program status word (PSW) is an 8-bit register that contains flags that are set or reset depending on the results of an instruction. This register can be written to or read from 8 bits at a time. The individual flags can also be manipulated on a bit-by-bit basis. The assignment of PSW bits follows.

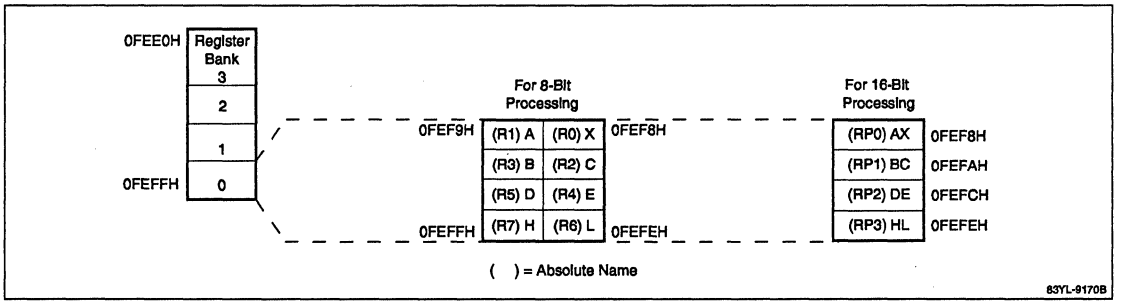
7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

CY	Carry flag
ISP	Interrupt priority status flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

General Registers

The general-purpose registers (figure 2) consist of four banks of registers located at addresses FEE0H to FEFFH in Internal RAM. Each bank consists of eight 8-bit general registers that can also be used in pairs to function as four 16-bit registers. Two bits in the PSW (RBS0 and RBS1) specify which of the register banks is active. The bits are set under program control. Registers have both functional names (like A, X, B, C for 8-bit registers and AX, BC for 16-bit registers) and absolute names (like R1, R0, R3, R2 for 8-bit registers and RP0, RP1, etc. for 16-bit registers). Each instruction determines whether a register is referred to by its functional or absolute name and whether it is 8 or 16 bits.

Figure 2. General Registers



Addressing

The μPD78214 family features 1-byte addressing of both the special function registers and the portion of on-chip RAM from FE20H to FEFFH. The 1-byte sfr addressing accesses the entire SFR area, while the 1-byte saddr addressing access the first 32 bytes of the SFR area and 224 bytes of Internal RAM. The 16-bit SFRs and words of memory in these areas can be addressed by 1-byte saddrp addressing, which is valid for even addresses only. Since many instructions use 1-byte addressing, access to these locations is almost as fast and versatile as access to the general registers. There are seven addressing modes for data in main memory: direct, register, register indirect with autoincrement and decrement, saddr, SFR, based, and indexed. There are also both 8-bit and 16-bit immediate operands.

Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and the CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by 1-byte sfr addressing. They are either 8 or 16 bits as required, and many of the 8-bit registers are capable of single-bit access as well. Locations FFD0H through FFDFH are known as the external SFR area. Registers in external circuitry interfaced and mapped to these addresses can be addressed with sfr addressing. Table 1 is a list of the special function registers.

Table 1. Special Function Registers

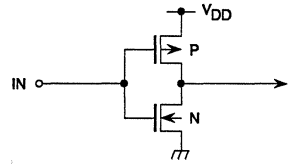
Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
0FF00H	Port 0	P0	R/W	x	x	—	Undefined
0FF02H	Port 2	P2	R	x	x	—	Undefined
0FF03H	Port 3	P3	R/W	x	x	—	Undefined
0FF04H	Port 4	P4	R/W	x	x	—	Undefined
0FF05H	Port 5	P5	R/W	x	x	—	Undefined
0FF06H	Port 6	P6	R/W	x	x	—	x0H
0FF07H	Port 7	P7	R	x	x	—	Undefined
0FF0AH	Port 0 buffer register (low)	P0L	R/W	x	x	—	Undefined
0FF0BH	Port 0 buffer register (high)	P0H	R/W	x	x	—	Undefined
0FF0CH	Real-time output port control register	RTPC	R/W	x	x	—	00H
0FF10H-0FF11H	16-bit compare register 0 (16-bit timer 0)	CR00	R/W	—	—	x	Undefined
0FF12H-0FF13H	16-bit compare register (16-bit timer 0)	CR01	R/W	—	—	x	Undefined
0FF14H	8-bit compare register (8-bit timer 1)	CR10	R/W	—	x	—	Undefined
0FF15H	8-bit compare register (8-bit timer/counter 2)	CR20	R/W	—	x	—	Undefined
0FF16H	8-bit compare register (8-bit timer/counter 2)	CR21	R/W	—	x	—	Undefined
0FF17H	8-bit compare register (8-bit timer 3)	CR30	R/W	—	x	—	Undefined
0FF18H-0FF19H	16-bit capture register (16-bit timer 0)	CR02	R	—	—	x	Undefined
0FF1AH	8-bit capture register (8-bit timer/counter 2)	CR22	R	—	x	—	Undefined
0FF1CH	8-bit capture/compare register (8-bit timer 1)	CR11	R/W	—	x	—	Undefined
0FF20H	Port 0 mode register	PM0	W	—	x	—	FFH
0FF23H	Port 3 mode register	PM3	W	—	x	—	FFH
0FF25H	Port 5 mode register	PM5	W	—	x	—	FFH
0FF26H	Port 6 mode register	PM6	R/W	x	x	—	FxH
0FF30H	Capture/compare control register 0	CRC0	W	—	x	—	10H
0FF31H	Timer output control register	TOC	W	—	x	—	00H
0FF32H	Capture/compare control register 1	CRC1	W	—	x	—	00H
0FF34H	Capture/compare control register 2	CRC2	W	—	x	—	00H
0FF40H	Pullup resistor option register	PUO	R/W	x	x	—	00H
0FF43H	Port 3 mode control register	PMC3	R/W	x	x	—	00H
0FF50H-0FF51H	16-bit timer register 0	TM0	R	—	—	x	0000H
0FF52H	8-bit timer register 1	TM1	R	—	x	—	00H
0FF54H	8-bit timer register 2	TM2	R	—	x	—	00H
0FF56H	8-bit timer register 3	TM3	R	—	x	—	00H
0FF5CH	Prescaler mode register 0	PRM0	W	—	x	—	00H
0FF5DH	Timer control register 0	TMC0	R/W	—	x	—	00H
0FF5EH	Prescaler mode register 1	PRM1	W	—	x	—	00H
0FF5FH	Timer control register 1	TMC1	R/W	—	x	—	00H
0FF68H	A/D converter mode register	ADM	R/W	x	x	—	00H
0FF6AH	A/D conversion result register	ADCR	R	—	x	—	Undefined
0FF80H	Clocked serial interface mode register	CSIM	R/W	x	x	—	00H

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
0FF82H	Serial bus interface control register	SBIC	R/W	x	x	—	00H
0FF86H	Serial shift register	SIO	R/W	—	x	—	Undefined
0FF88H	Asynchronous serial interface mode register	ASIM	R/W	x	x	—	80H
0FF8AH	Asynchronous serial interface status register	ASIS	R	x	x	—	00H
0FF8CH	Serial receive buffer: UART	RxB	R	—	x	—	Undefined
0FF8EH	Serial transmit shift register: UART	TxS	W	—	x	—	Undefined
0FF90H	Baud rate generator control register	BRGC	W	—	x	—	00H
0FFC0H	Standby control register	STBC	R/W	—	x	—	0000x000B
0FFC4H	Memory expansion mode register	MM	R/W	x	x	—	20H
0FFC5H	Programmable wait control register	PW	R/W	x	x	—	80H
0FFC6H	Refresh mode register	RFM	R/W	x	x	—	00H
0FFD0H-0FFDFH	External SFR area	—	R/W	x	x	—	Undefined
0FFE0H	Interrupt request flag register L	IF0L	R/W	x	x	—	00H
0FFE1H	Interrupt request flag register H	IF0H	R/W	x	x	—	00H
0FFE0H-0FFE1H	Interrupt request flag register	IF0	R/W	—	—	x	0000H
0FFE4H	Interrupt mask flag register L	MK0L	R/W	x	x	—	FFH
0FFE5H	Interrupt mask flag register H	MK0H	R/W	x	x	—	FFH
0FFE4H-0FFE5H	Interrupt mask flag register	MK0	R/W	—	—	x	FFFFH
0FFE8H	Priority specification flag register L	PR0L	R/W	x	x	—	FFH
0FFE9H	Priority specification flag register H	PR0H	R/W	x	x	—	FFH
0FFE8H-0FFE9H	Priority specification flag register	PR0	R/W	—	—	x	FFFFH
0FFECH	Interrupt service mode specification flag register L	ISM0L	R/W	x	x	—	00H
0FFEDH	Interrupt service mode specification flag register H	ISM0H	R/W	x	x	—	00H
0FFECH-0FFEDH	Interrupt service mode specification flag register	ISM0	R/W	—	—	x	00H
0FFF4H	External interrupt mode register 0	INTM0	R/W	x	x	—	00H
0FFF5H	External interrupt mode register 1	INTM1	R/W	x	x	—	00H
0FFF8H	Interrupt status register	IST	R/W	x	x	—	00H

Figure 3. Pin I/O Circuits

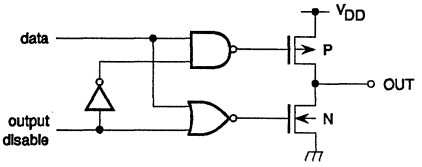
Type 1 ($\overline{EA}$)



Type 2 ($P2_0$, P_1, $\overline{RESET}$)

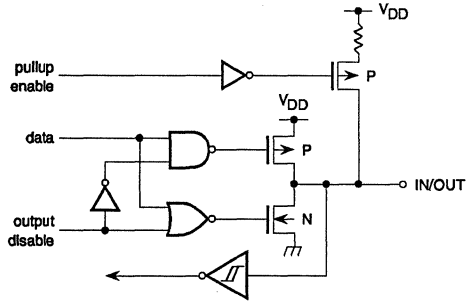
Schmitt trigger input with hysteresis characteristic.

Type 4 (P0, $P6_0$ - $P6_3$, ASTB)

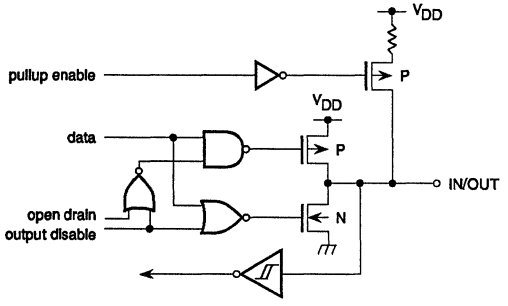


Push-pull output where the output can be placed in high-impedance (both P and N channels are turned off).

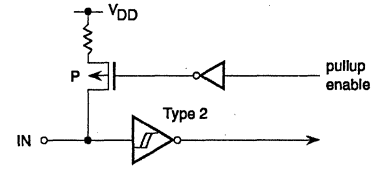
Type 8-A ($P3_2$)



Type 10-A ($P3_3$)

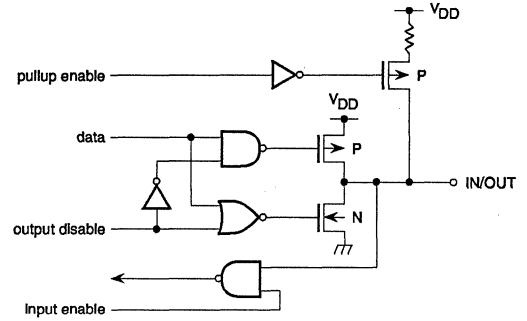


Type 2-A ($P2_2$ - $P2_7$)

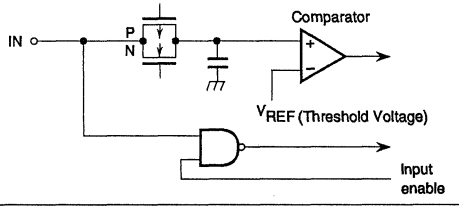


Schmitt trigger input with hysteresis characteristic.

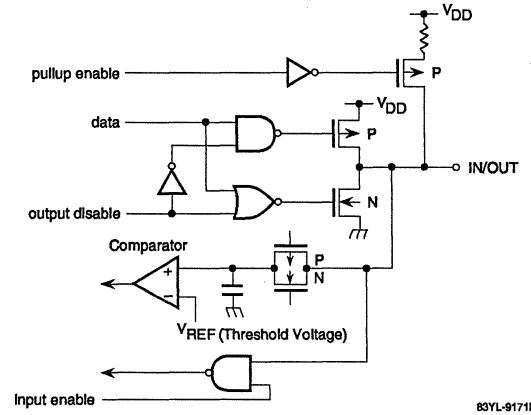
Type 5-A ($P3_0$, $P3_1$, $P3_4$ - $P3_7$, P4, P5, $P6_4$, $P6_5$)



Type 9 (P7)



Type 11 ($P6_6$ - $P6_7$)



83YL-9171B

Input/Output Ports

There are up to 54 port lines on the μPD78212/214/P214 and up to 36 port lines on the μPD78213. (Ports 4, 5, and two bits of port 6 are not available on the μPD78213 since the μPD78213 must always use external memory.) Table 2 lists the features of each port and figure 3 shows the structure of each port pin. The pin levels of all port 2, 3, and 7 pins can always be read or tested regardless of the dual pin function.

Table 2. Digital Port Functions

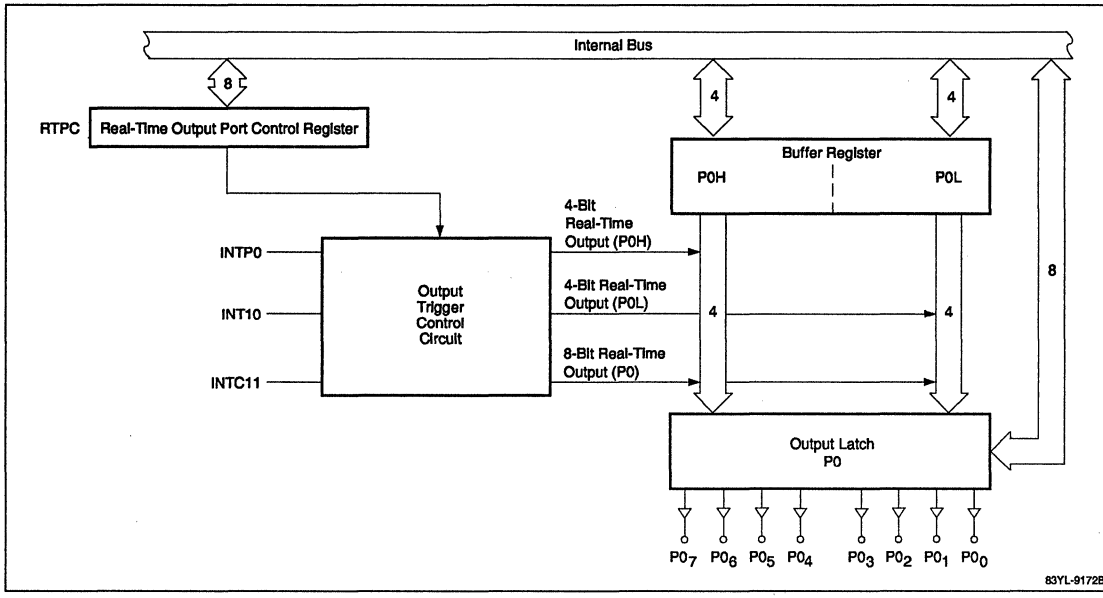
Port	Operational Features	Configuration	Direct Drive Capability	Software Pullup Resistor Connection
Port 0	8-bit high impedance output		Transistor	
Port 2	8-bit Schmitt trigger input			In 6-bit unit ($P2_2$ - $P2_7$)
Port 3	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 4	8-bit input or output	Byte selectable	LED	Byte selectable
Port 5	8-bit input or output	Byte selectable	LED	Byte selectable, input bits only
Port 6	4-bit output (bits 0 to 3) 4-bit input or output (bits 4 to 7)	Bit selectable		In 4-bit unit, input bits only
Port 7	6-bit input			

Note: Software pullup resistors can be internally connected only on a port-by-port basis to port bits set to input mode. Pullup resistors are not connected to port bits set to output mode.

Real-time Output Port

The real-time output port (RTPC) shares pins with port 0. It can be used as two independent 4-bit real-time output ports or one 8-bit real-time output port. In the real-time output mode, data stored beforehand in the buffer registers, P0H and P0L, is transferred immediately to the output latch of P0 on the occurrence of a timer 1 interrupt (INTC10 or INTC11) or external interrupt (INTP0) (see figure 4). By using the real-time output port with the macro service function, port 0 can be used to output preprogrammed patterns at preprogrammed variable time intervals. In this mode, two independent stepper motors can accurately be driven at a fixed or variable rate.

Figure 4. Real-Time Output Port



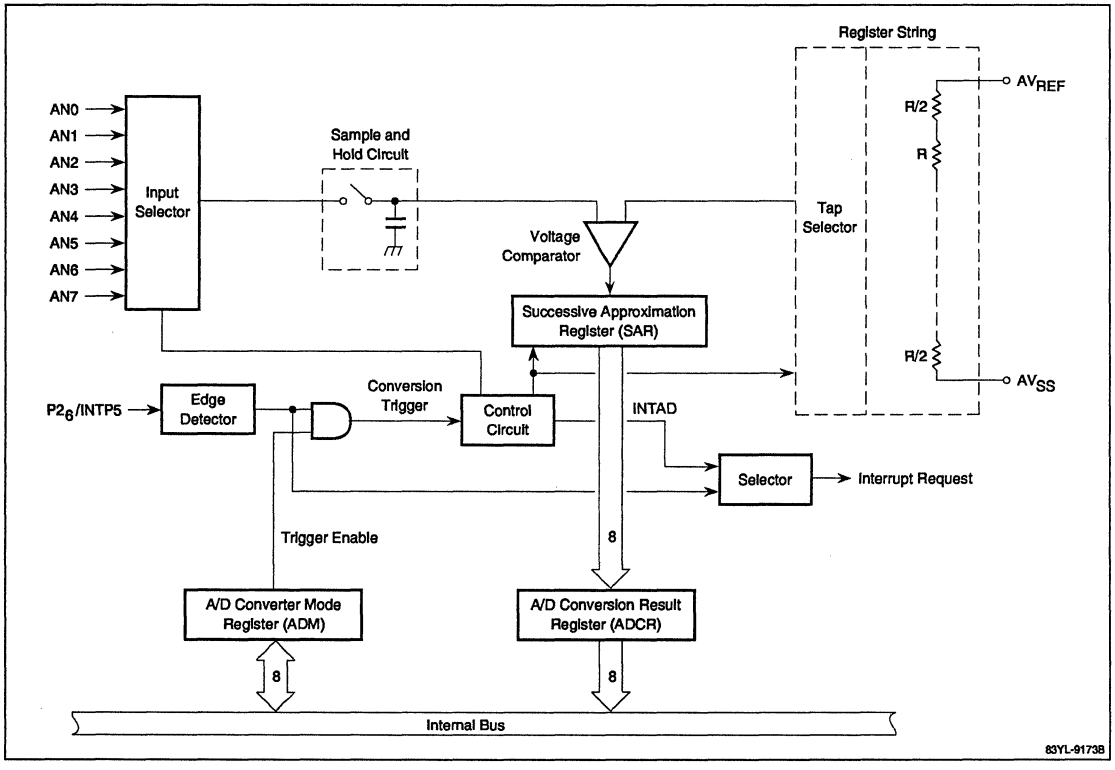
Analog-to-Digital (A/D) Converter

The μPD78214 family A/D converter (see figure 5) uses the successive-approximation method for converting up to eight multiplexed analog inputs into 8-bit digital data. The conversion time per input is 30 μs at 12 MHz operation. A/D conversion can be started by an external interrupt, INTP5, or under software control.

The A/D converter can operate in either scan mode or select mode. In scan mode, from one to eight sequential inputs can be programmed for conversion. The A/D data, stores it in the A/D conversion result (ADCR) register, and generates an interrupt (INTAD). This converted data can be easily transferred to memory by using the macro service function.

In select mode, only one of the eight A/D inputs can be selected for conversion. The ADCR register is continually updated and can be read at any time. If the A/D converter is started by an external interrupt, an INTAD interrupt occurs at the completion of each conversion. If the A/D converter is started by software, no interrupts are generated.

Figure 5. A/D Converter



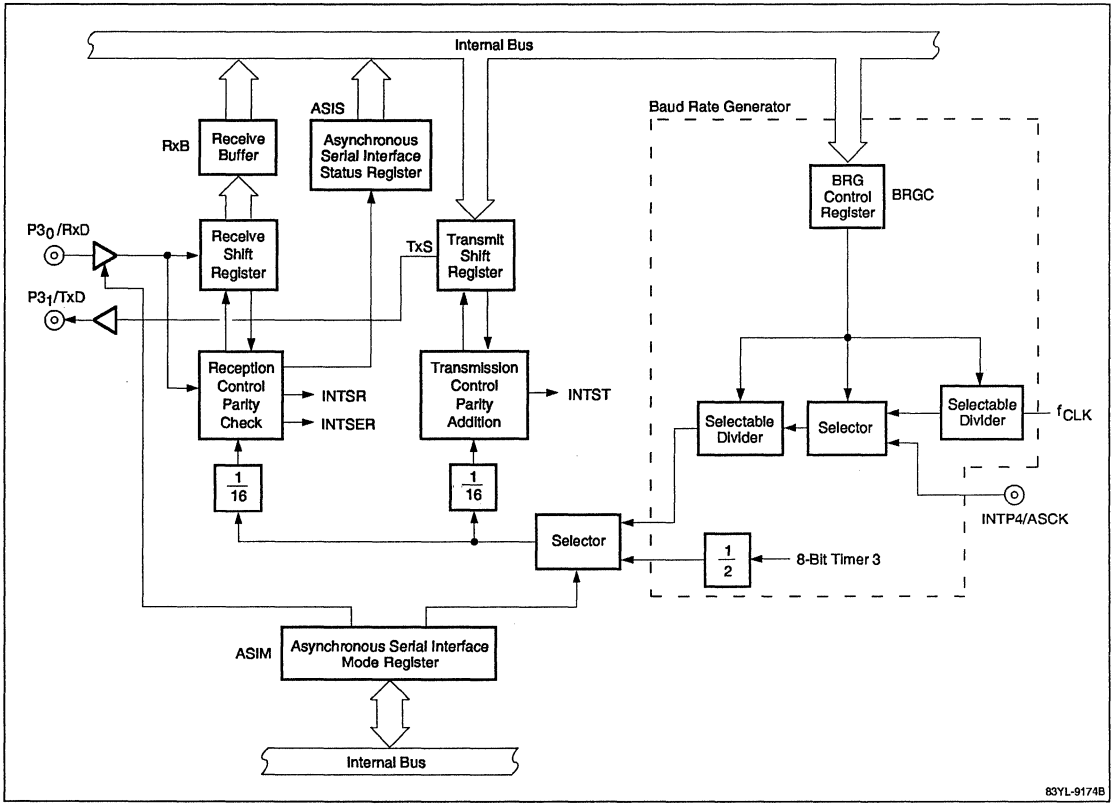
4a

Serial Interface

The μPD78214 family has two independent serial interfaces. The first is a standard UART. The UART (figure 6) permits full-duplex operation and can be programmed for 7- or 8-bits of data after the start bit, followed by one or two stop bits. Odd, even, zero or no parity can also be selected. The serial clock for the UART can be provided by an on-chip baud rate generator or timer 3.

By using either the internal system clock or an external clock input into the ASCK pin, the baud rate generator is capable of generating all of the commonly used baud rates. The UART generates three interrupts: INTST (transmission complete), INTSR (reception complete), and INTSER (reception error).

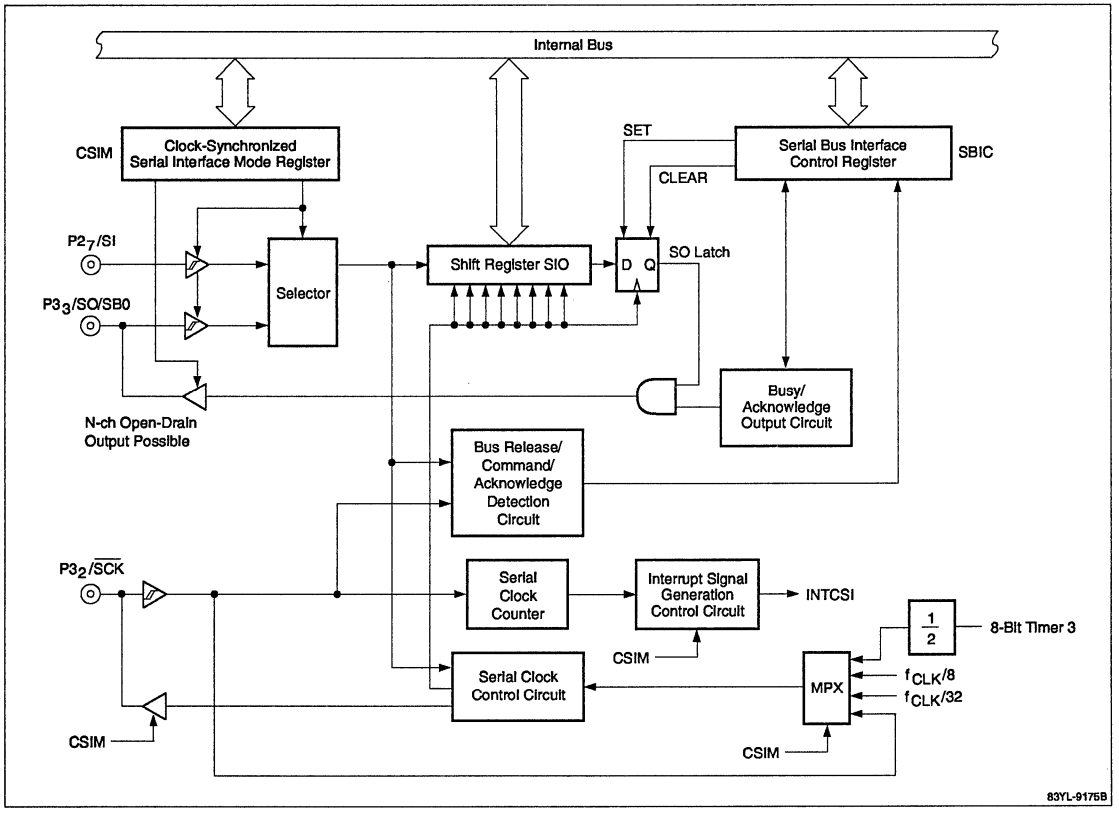
Figure 6. Asynchronous Serial Interface



The second interface is an 8-bit clock-synchronized serial interface (figure 7). It can be operated in either a three-wire serial I/O mode or NEC serial bus interface (SBI) mode.

In the three-wire serial I/O mode, the 8-bit shift register (SIO) is loaded with a byte of data and eight clock pulses are generated. These eight pulses shift the byte of data out of the SO line (MSB first) and in from the SI line providing full-duplex operation. This interface can also be set to receive or to transmit data only. The INTCSI interrupt is generated after each 8-bit transfer. One of three internal clocks or an external clock clocks the data.

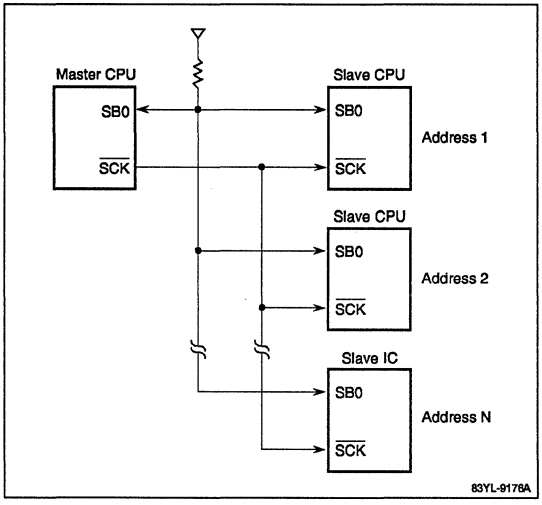
Figure 7. Clock-Synchronized Serial Interface



4a

The NEC SBI mode is a two-wire high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are connected in a master/slave configuration (see figure 8). There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over the serial bus line (SB0) using a fixed hardware protocol synchronized with the $\overline{SCK}$ line. Each slave μPD78214 family device can be programmed in software to respond to any one of 256 addresses. There are also 256 commands and 256 data types. Since all commands are user definable, any software protocol, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

Figure 8. SBI Mode Master/Slave Configuration



Timers

The μPD78214 family has one 16-bit timer and three 8-bit timers. The 16-bit timer counts the internal system clock ($f_{CLK}/8$) while the three 8-bit timers can be programmed to count a number of prescaled values of the internal system clock. One of the 8-bit timers can also count external events.

Timer 0 consists of a 16-bit timer (TM0), two 16-bit compare registers (CR00 and CR01), and a 16-bit capture register (CR02). Timer 0 can be used as two interval timers, to output a programmable square wave or two pulse-width modulated signals, or to measure pulse widths (see figure 9).

Timer 1 consists of an 8-bit timer (TM1), 8-bit compare register (CR10), and 8-bit capture/compare register (CR11). Timer 1 can be used as two interval timers or to measure pulse widths. In addition, it can be used to generate the output trigger for the real-time output port (see figure 10).

Timer/counter 2 consists of an 8-bit timer (TM2), two 8-bit compare registers (CR20 and CR21), and an 8-bit capture register (CR22). Timer/counter 2 can also be used as two interval timers, to output a programmable square wave or two pulse-width modulated signals, or to measure pulse widths. In addition, it can be used to count external events sensed on the CI line or as a one-shot timer (see figure 11).

Timer 3 consists of an 8-bit timer (TM3) and an 8-bit compare register (CR30). Timer 3 can be used as an interval timer or as a clock for the clock-synchronized serial interface (see figure 12).

Figure 9. 16-Bit Timer 0

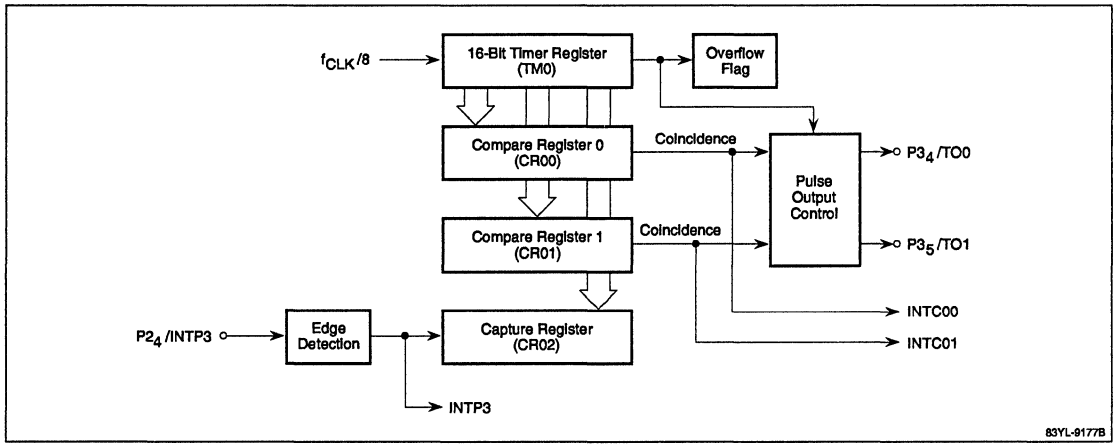
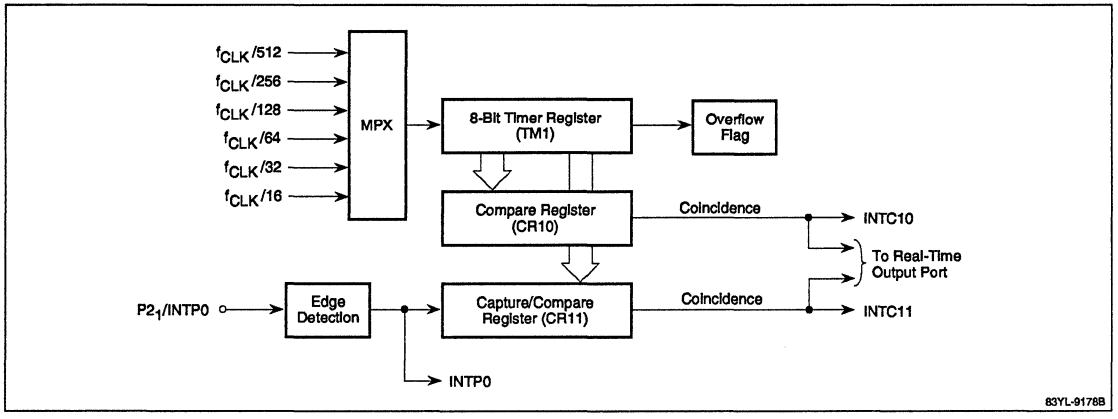


Figure 10. 8-Bit Timer 1



4a

Figure 11. 8-Bit Timer/Counter 2

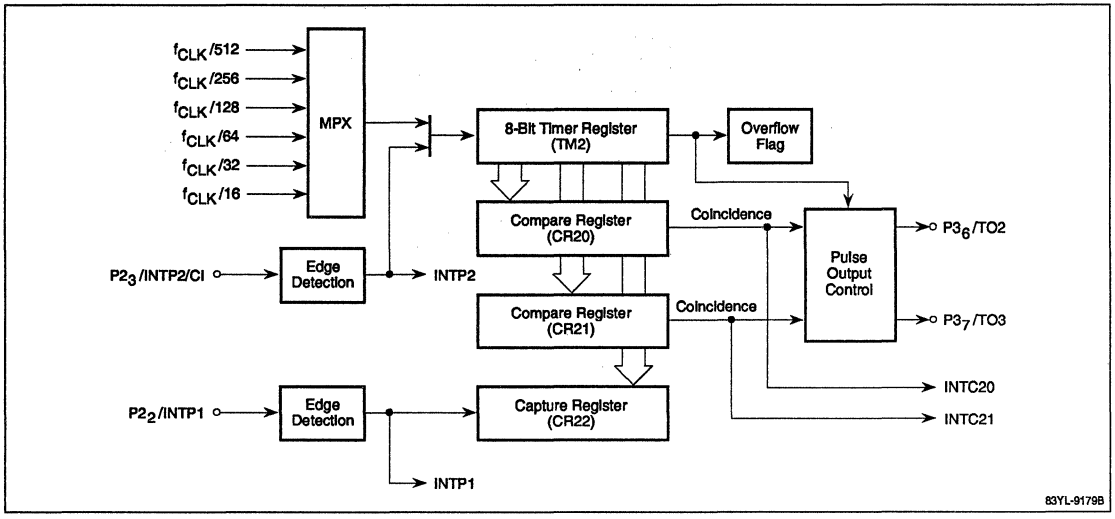
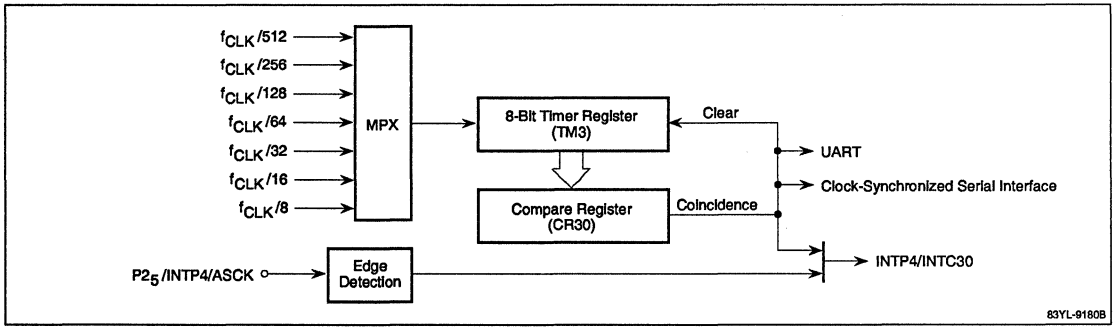


Figure 12. 8-Bit Timer 3



Interrupts

The μPD78214 family has 18 maskable hardware interrupt sources; 6 are external and 12 are internal. Since there are only 16 interrupt vectors and sets of control flags, 2 of the 6 external maskable interrupts, INTP4 and INTP5, share interrupt vectors and control flags with INTC30 and INTAD respectively. The active interrupt source for each shared vector must be chosen by the program. In addition, there is one nonmaskable interrupt and one software interrupt. The software interrupt, generated by the BRK instruction, is not maskable (see table 3).

Table 3. Interrupt Sources and Vector Addresses

Interrupt Request Type	Default Priority	Interrupt Request Generation Source	Macro Service Type	Vector Table Address
Software	None	BRK instruction execution	—	003EH
Nonmaskable	None	NMI (pin input edge detection)	—	0002H
Maskable	0	INTP0 (pin input edge detection)	A, B	0006H
	1	INTP1 (pin input edge detection)	A, B	0008H
	2	INTP2 (pin input edge detection)	A, B	000AH
	3	INTP3 (pin input edge detection)	B	000CH
	4	INTC00 (TM0-CR00 coincidence signal generation)	B	0014H
	5	INTC01 (TM0-CR01 coincidence signal generation)	B	0016H
	6	INTC10 (TM1-CR10 coincidence signal generation)	A, B, C	0018H
	7	INTC11 (TM1-CR11 coincidence signal generation)	A, B, C	001AH
	8	INTC21 (TM2-CR21 coincidence signal generation)	A, B	001CH
	9	INTP4 (pin input edge detection)	B	000EH
		INTC30 (TM3-CR30 coincidence signal generation)	A, B	
	10	INTP5 (pin input edge detection)	B	0010H
		INTAD (end of A/D conversion)	A, B	
	11	INTC20 (TM2-CR20 coincidence signal generation)	A, B	0012H
	12	INTSER (generation of asynchronous serial interface receive error)	—	0020H
	13	INTSR (end of asynchronous serial interface reception)	A, B	0022H
	14	INTST (end of asynchronous serial interface transmission)	A, B	0024H
	15	INTCSI (end of clocked serial interface transmission)	A, B	0026H

Interrupt Servicing. The μPD78214 family provides two levels of programmable hardware priority control and two different methods of handling maskable interrupt requests: standard vectoring and macro service. The programmer can choose the priority and mode of servicing each maskable interrupt by using the interrupt control registers.

Interrupt Control Registers. The μPD78214 family has four 16-bit interrupt control registers. Each bit in each register is dedicated to one of the 16 active maskable interrupt sources. The interrupt request flag register (IF0) contains an interrupt request flag for each interrupt. The interrupt mask register (MK0) is used to enable or disable any interrupt. The interrupt service mode register (ISM0) specifies whether an interrupt is processed by vectoring or macro service. The priority flag register (PR0) can be used to specify a high or a low priority level for each interrupt.

Two other 8-bit registers are associated with interrupt processing. The interrupt status register (IST) indicates if a nonmaskable interrupt request on the NMI pin is being processed and can be used to allow nesting of nonmaskable interrupt requests. The IE and the ISP bits of the program status word are also used to control interrupts. If the IE bit is zero, all maskable interrupts, but not macro service, are disabled. The IE bit can be set or cleared using the EI and DI instructions, respectively, or by directly writing to the PSW. The IE bit is cleared each time an interrupt is accepted. The ISP bit is used by hardware to hold the priority level flag of the interrupt being serviced.

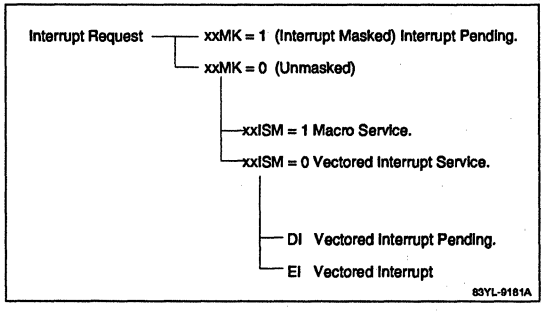
Interrupt Priority. The nonmaskable interrupt (NMI) has priority over all other interrupts. Two hardware controlled priority levels are available for the maskable interrupts. Either a high or a low priority level can be assigned by software to each of the maskable interrupts. Interrupt requests of a priority higher than the processor's current priority level are accepted; requests of the same or lower priority are held pending until the processor's priority state is lowered by program control within the current service routine or by a return instruction from the current service routine.

Interrupt requests programmed to be handled by macro service have priority over all vectored interrupt service regardless of the assigned priority level, and macro service requests are accepted even when the interrupt enable bit in the PSW is set to the disable state (see figure 13).

The default priorities listed in table 3 are fixed by hardware and are effective only when it is necessary to choose between two interrupt requests of the same software-assigned priority. For example, the default priorities would be used after the completion of a high priority routine if two interrupts of the same priority routine were pending.

The software interrupt, initiated by the BRK instruction, is executed regardless of the processor's priority level and the state of the IE bit. It does not alter the processor's priority level.

Figure 13. Interrupt Service Sequence



Vectored Interrupt. When vectored interrupt is specified for a given interrupt request, (1) the program status word and the program counter are saved on the stack, (2) the processor's priority is set to that specified for the interrupt, (3) the IE bit in the PSW is set to zero, and (4) the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (RETB instruction for the software interrupt) reverses the process and the μPD78214 family device resumes the interrupted routine.

Macro Service

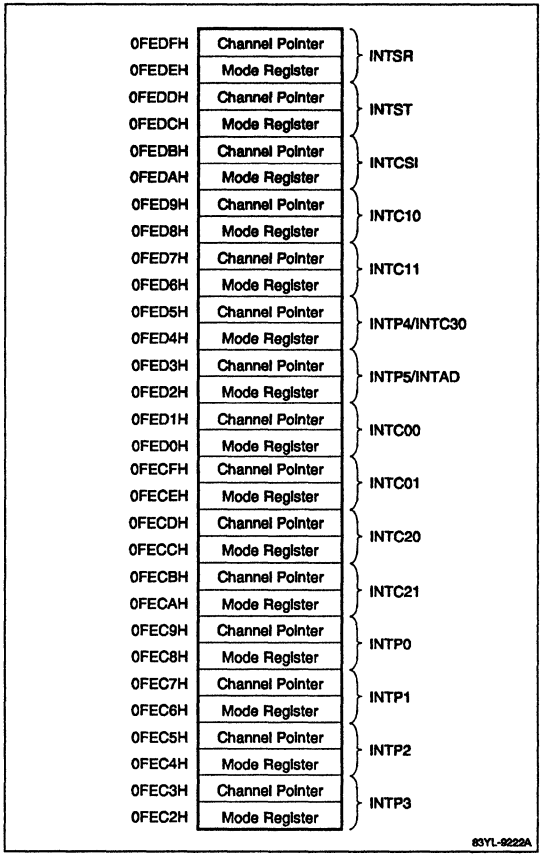
When macro service is specified for a given interrupt, the macro service hardware temporarily stops the executing program and begins to transfer data between the special function register area and the memory space. One byte is transferred each interrupt. When the data transfer is complete, control is returned to the executing program, providing a completely transparent method of interrupt service. Macro service significantly improves response time and makes it unnecessary to save any registers.

For each request on the interrupt line, one operation is performed, and an 8-bit counter is decremented. When the counter reaches zero, a vectored interrupt service routine is entered according to the specified priority.

Macro service is provided for all of the maskable interrupt requests except INTSER, the asynchronous serial interface receive error interrupt request. Each interrupt request has a dedicated macro service control word stored in Internal RAM (see figure 14). The function to be performed is specified in the control word.

The μPD78214 family provides three different types of macro service transfers.

Figure 14. Macro Service Control Word Map



Macro Service Type A. A byte of data is transferred in either direction between a special function register, preassigned for each interrupt request, and a buffer in internal RAM (FExx). The preassigned SFRs for the 12 interrupt requests that support macro service Type A transfers are listed in table 4.

Table 4. Macro Service Type A Interrupts and Assigned SFRs

Interrupt Request	Source/Destination SFR
INTC10: TM1-CR10 coincidence	CR10: Timer 1 8-bit compare register
INTC11: TM1-CR11 coincidence	CR11: Timer 1 8-bit capture/ compare register
INTC20: TM2-CR20 coincidence	CR20: Timer 2 8-bit compare register
INTC21: TM2-CR21 coincidence	CR21: Timer 2 8-bit compare register
INTC30: TM3-CR30 coincidence	CR30: Timer 3 8-bit compare register
INTSR: End of asynchronous serial interface reception	RxB: Serial receive buffer
INTST: End of asynchronous serial interface transmission	TxS: Serial transmit shift register
INTCSI: End of clocked serial interface transmission	SIO: Serial shift register
INTAD: End of A/D conversion	ADCR: A/D conversion result register
INTP0: External interrupt pin $P0_1$	CR11: Timer 1 8-bit capture/ compare register
INTP1: External interrupt pin $P0_2$	CR22: Timer 2 8-bit capture register
INTP2: External interrupt pin $P0_3$	TM2: Timer 2 8-bit timer register

Macro Service Type B. A byte of data is transferred in either direction between any specified special function register and a buffer anywhere in the 64K byte address space. Macro service Type B transfers can be initiated by any maskable interrupt except INTSER.

Macro Service Type C. A byte of data is transferred from a buffer anywhere in the 64K byte address space to one of the 8-bit compare registers of timer 1. At the same time, a second byte of data is transferred from a buffer anywhere in the 64K byte address space to the real-time output port buffer. Macro service Type C transfers can be initiated by INTC10 with data transferred to CR10 and P0L or P0H, or by INTC11 with data transferred to CR11 and P0L or P0H.

In addition, the macro service Type C transfer can be initialized to automatically alter timer compare register values or to repeatedly output a prespecified pattern at a fixed or variable rate. By using macro service Type C transfers to control the real-time output ports, the μPD78214 family can easily and accurately drive two independent stepper motors.

Refresh

The refresh signal is used with any pseudostatic RAM equivalent of the NEC μPD428128. The refresh cycle can be set to one of four intervals: 16, 32, 64, or 128/f_{CLK} (2.6, 5.3, 10.7, and 21.3 μs at 12 MHz). The refresh cycle is timed to follow a read or write operation to avoid interference with external memory access cycles.

Standby Modes

HALT and STOP modes are provided to reduce power consumption when CPU action is not required. In HALT mode, the CPU is stopped but the system clock continues to run. The HALT mode is released by any unmasked interrupt, an external NMI, or an external reset pulse. In STOP mode, both the CPU and the system clock are stopped, further minimizing the power consumption. The STOP mode is released by either an external reset pulse or an external NMI. The HALT and STOP modes are entered by programming the standby control register (STBC). This register is a protected location and can be written to only by a special instruction. If the third and fourth bytes of the instruction are not complements of each other, the data is not written and the next instruction is executed.

External Reset

The μPD78214 family is reset by taking the $\overline{RESET}$ pin low. The $\overline{RESET}$ input pin contains a noise filter to protect against spurious system resets caused by noise. On power-up, the $\overline{RESET}$ pin must remain low until the power supply reaches its operating voltage and the oscillator has stabilized. During reset, the program counter is loaded with the address contained in the reset vector table (address 0000H, 0001H); program execution starts at that address upon the $\overline{RESET}$ pin going high. While $\overline{RESET}$ is low, all external lines except V_{SS}, V_{DD}, AV_{SS}, AV_{REF}, X1, and X2 are in the high impedance state.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = +25°C

Operating voltage, V_{DD}	−0.5 to +7.0 V
AV_{REF}	−0.5 to V_{DD} + 0.5 V
AV_{SS}	−0.5 to +0.5 V
Input voltage, V_{I1}	−0.5 to V_{DD} + 0.5 V
V_{I2} (Note 1)	−0.5 to AV_{REF} + 0.5 V
V_{I3} (Note 2 for μPD78P214)	−0.5 to +13.5 V
Output voltage, V_O	−0.5 to V_{DD} + 0.5 V
Low-level output current, I_{OL}	
per pin	15 mA
total, all output pins	100 mA
High-level output current, I_{OH}	
per pin	−10 mA
total, all output pins	−50 mA
Operating temperature, T_{OPT}	−40 to +85°C
Storage temperature, T_{STG}	−65 to +150°C

Notes:

(1) Pins $P7_0$/AN0 - $P7_5$/AN5, $P6_6/\overline{WAIT}$/AN6, and $P6_7/\overline{REFRQ}$/AN7 when the pin is used as the A/D converter input or is selected by bits AN10-AN12 of the ADM register when the A/D converter is not in operation. However, V_{I1} absolute maximum ratings should also be satisfied.

(2) $P2_0$/NMI, $\overline{EA}/V_{PP}$, and $P2_1$/INTP0/A_9 pins in the PROM programming mode

Exposure to absolute maximum ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC characteristics.

Operating Conditions

Oscillation Frequency	T_A	V_{DD}
f_{XX} = 4 to 12 MHz	−40 to +85°C	+5 V ±10%

Capacitance

T_A = +25°C; V_{DD} = V_{SS} = 0 V

Item	Symbol	Max	Unit	Conditions
Input capacitance	C_I	20	pF	f = 1 MHz; pins not used for measurement are at 0 V
Output capacitance	C_O	20	pF	
Input/output capacitance	C_{IO}	20	pF	

DC Characteristics

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V

Item	Symbol	Min	Typ	Max	Unit	Conditions
Low-level input voltage	V_{IL}	0		0.8	V	
High-level input voltage	V_{IH1}	2.2		V_{DD}	V	Except the specified pins (Notes 1, 2)
	V_{IH2}	2.2		AV_{REF}	V	Specified pins (Note 1)
	V_{IH3}	0.8 V_{DD}		V_{DD}	V	Specified pins (Note 2)
Low-level output voltage	V_{OL1}			0.45	V	I_{OL} = 2.0 mA
	V_{OL2}			1.0	V	I_{OL} = 8.0 mA (Note 3)
High-level output voltage	V_{OH1}	V_{DD} − 1.0			V	I_{OH} = −1.0 mA
	V_{OH2}	V_{DD} − 0.5			V	I_{OH} = −100 μA
	V_{OH3}	2.0			V	I_{OH} = −5.0 mA (Note 4)
X1 low-level input current	I_{IL}			−100	μA	0 ≤ V_I ≤ V_{IL}
X1 high-level input current	I_{IH}			100	μA	V_{IH3} ≤ V_I ≤ V_{DD}
Input leakage current	I_{LI}			±10	μA	0V ≤ V_I ≤ V_{DD}
Output leakage current	I_{LO}			±10	μA	0V ≤ V_O ≤ V_{DD}
AV_{REF} current	AI_{REF}		1.5	5.0	mA	Operating mode, f_{XX} = 12 MHz
V_{DD} power supply current	I_{DD1}		20	40	mA	Operating mode, f_{XX} = 12 MHz
	I_{DD2}		7	20	mA	HALT mode, f_{XX} = 12 MHz
Data retention voltage	V_{DDDR}	2.5		5.5	V	STOP mode
Data retention current	I_{DDDR}		2	20	μA	STOP mode; V_{DDDR} = 2.5 V
			5	50	μA	STOP mode; V_{DDDR} = 5 V ±10%
Pullup resistor	R_L	15	40	80	kΩ	V_I = 0 V

Notes:

(1) Pins $P7_0$/AN0 - $P7_5$/AN5, $P6_6$/$\overline{WAIT}$/AN6, and $P6_7$/$\overline{REFRQ}$/AN7 when the pin is used as the A/D converter input or is selected by bits AN10 - AN12 of the ADM register when the A/D converter is not in operation.

(2) X1, X2, $\overline{RESET}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2/CI, $P2_4$/INTP3, $P2_5$/INTP4/ASCK, $P2_6$/INTP5, $P2_7$/SI, $P3_2$/SCK, $P3_3$/SO/SB0, and $\overline{EA}$ pins.

(3) Pins $P4_0$/AD_0 - $P4_7$/AD_7, and $P5_0$/A_8 - $P5_7$/A_{15}.

(4) Pins $P0_0$ - $P0_7$.

4a

AC Characteristics—Read/Write Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF

Item	Symbol	Calculation Formula (2, 3)	Min	Max	Unit	Conditions
X1 input clock cycle time	t_{CYX}	—	82	250	ns	
Address setup time to ASTB ↓	t_{SAST}	t_{CYX} − 30	52		ns	
Address hold time from ASTB ↓ (Note 1)	t_{HSTA}	—	25		ns	
Address hold time from $\overline{RD}$↑	t_{HRA}	—	30		ns	
Address hold time from $\overline{WR}$ ↑	t_{HWA}	—	30		ns	
Address to $\overline{RD}$ ↓ delay time	t_{DAR}	$2t_{CYX}$ − 35	129		ns	
Address float time to $\overline{RD}$ ↓	t_{FAR}	$t_{CYX}/2$ − 30	11		ns	
Address to data input time	t_{DAID}	$(4+2n)t_{CYX}$ − 100		228	ns	No wait states
ASTB ↓ to data input time	t_{DSTID}	$(3+2n)t_{CYX}$ − 65		181	ns	No wait states
$\overline{RD}$ ↓ to data input time	t_{DRID}	$(2+2n)t_{CYX}$ − 64		100	ns	No wait states
ASTB ↓ to $\overline{RD}$ ↓ delay time	t_{DSTR}	t_{CYX} − 30	52		ns	
Data hold time from $\overline{RD}$ ↑	t_{HRID}	—	0		ns	
$\overline{RD}$ ↑ to address active time	t_{DRA}	$2t_{CYX}$ − 40	124		ns	
$\overline{RD}$ ↑ to ASTB ↑ delay time	t_{DRST}	$2t_{CYX}$ − 40	124		ns	
$\overline{RD}$ low-level width	t_{WRL}	$(2+2n)t_{CYX}$ − 40	124		ns	No wait states
ASTB high-level width	t_{WSTH}	t_{CYX} − 30	52		ns	
Address to $\overline{WR}$ ↓ delay time	t_{DAW}	$2t_{CYX}$ − 35	129		ns	
$\overline{ASTB}$ ↓ to data output time	t_{DSTOD}	t_{CYX} + 60		142	ns	
$\overline{WR}$ ↓ to data output time	t_{DWOD}	—		60	ns	
ASTB ↓ to $\overline{WR}$ ↓ delay time	t_{DSTW1}	t_{CYX} − 30	52		ns	
	t_{DSTW2}	$2t_{CYX}$ − 35	129		ns	Refresh mode
Data setup time to $\overline{WR}$ ↑	t_{SODWR}	$(3+2n)t_{CYX}$ − 100	146		ns	No wait states
Data setup time to $\overline{WR}$ ↓	t_{SODWF}	t_{CYX} − 60	22		ns	Refresh mode
Data hold time from $\overline{WR}$ ↑ (Note 1)	t_{HWOD}	—	20		ns	
$\overline{WR}$ ↑ to ASTB ↑ delay time	t_{DWST}	t_{CYX} − 40	42		ns	
$\overline{WR}$ low-level width	t_{WWL1}	$(3+2n)t_{CYX}$ − 50	196		ns	No wait states
	t_{WWL2}	$(2+2n)t_{CYX}$ − 50	114		ns	Refresh mode; No wait states
Address to $\overline{WAIT}$ ↓ input time	t_{DAWT}	$3t_{CYX}$ − 100		146	ns	
ASTB ↓ to $\overline{WAIT}$ ↓ input time	t_{DSTWT}	$2t_{CYX}$ − 80		84	ns	
$\overline{WAIT}$ hold time from ASTB ↓	t_{HSTWT}	$2Xt_{CYX}$ + 10	174		ns	One external wait state
ASTB ↓ to $\overline{WAIT}$ ↑ delay time	t_{DSTWTH}	$2(1+X)t_{CYX}$ − 55		273	ns	One external wait state
$\overline{RD}$ ↓ to $\overline{WAIT}$ input time	t_{DRWTL}	t_{CYX} − 60		22	ns	
$\overline{WAIT}$ hold time from $\overline{RD}$ ↓	t_{HRWT}	$(2X-1)t_{CYX}$ + 5	87		ns	One external wait state
$\overline{RD}$ ↓ to $\overline{WAIT}$ ↑ delay time	t_{DRWTH}	$(2X+1)t_{CYX}$ − 60		186	ns	One external wait state
$\overline{WAIT}$ ↑ to data input time	t_{DWTID}	t_{CYX} − 20		62	ns	
$\overline{WAIT}$ ↑ to $\overline{WR}$ ↑ delay time	t_{DWTW}	$2t_{CYX}$ − 10	154		ns	
$\overline{WAIT}$ ↑ to $\overline{RD}$ ↑ delay time	t_{DWTR}	t_{CYX} − 10	72		ns	
$\overline{WR}$ ↓ to $\overline{WAIT}$ input time	t_{DWWTL}	t_{CYX} − 60		22	ns	Refresh disabled

AC Characteristics—Read/Write Operation (cont)

Item	Symbol	Calculation Formula (2, 3)	Min	Max	Unit	Conditions
$\overline{WAIT}$ hold time from $\overline{WR}$ ↓	t_{HWWT1}	$(2X-1)t_{CYX}+5$	87		ns	One external wait state; refresh disabled
	t_{HWWT2}	$2(X-1)t_{CYX}+5$	5		ns	One external wait state; refresh enabled
$\overline{WR}$ ↓ to $\overline{WAIT}$ ↑ delay time	t_{DWWTH1}	$(2X+1)t_{CYX}-60$		186	ns	One external wait state; refresh disabled
	t_{DWWTH2}	$2Xt_{CYX}-60$		104	ns	One external wait state; refresh enabled
$\overline{RD}$ ↑ to $\overline{REFRQ}$ ↓ delay time	t_{DRRFQ}	$2t_{CYX}-10$	154		ns	
$\overline{WR}$ ↑ to $\overline{REFRQ}$ ↓ delay time	t_{DWRFQ}	$t_{CYX}-10$	72		ns	
$\overline{REFRQ}$ low-level width	t_{WRFQL}	$2t_{CYX}-44$	120		ns	
$\overline{REFRQ}$ ↑ to ASTB ↑ delay time	t_{DRFQST}	$4t_{CYX}-48$	280		ns	

Notes:

(1) The hold time includes the time during which V_{OH} and V_{OL} are retained under the following load conditions: C_L = 100 pF and R_L = 2 kΩ

(2) n indicates the number of internal wait states.

(3) x indicates the number of external wait states (1, 2, 3, ...)

4a

Serial Port Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF

Item	Symbol	Min	Max	Unit	Conditions
Serial clock cycle time	t_{CYSK}	1.0		μs	External clock input
		1.3		μs	Internal clock/16 output
		5.3		μs	Internal clock/64 output
Serial clock low-level width	t_{WSKL}	420		ns	External clock input
		556		ns	Internal clock/16 output
		2.5		μs	Internal clock/64 output
Serial clock high-level width	t_{WSKH}	420		ns	External clock input
		556		ns	Internal clock/16 output
		2.5		μs	Internal clock/64 output
SI, SB0 setup time to $\overline{SCK}$ ↑	t_{SSSK}	150		ns	
SI, SB0 hold time from $\overline{SCK}$ ↑	t_{HSSK}	400		ns	
SO/SB0 output delay time from $\overline{SCK}$ ↓	t_{DSBSK1}	0	300	ns	CMOS push-pull output (3-line serial I/O mode)
	t_{DSBSK2}	0	800	ns	Open-drain output (SBI mode), R_L = 1 kΩ
SB0 high, hold time from $\overline{SCK}$ ↑	t_{HSBSK}	4		t_{CYX}	SBI mode
SB0 low, setup time to $\overline{SCK}$ ↓	t_{SSBSK}	4		t_{CYX}	SBI mode
SB0 low-level width	t_{WSBL}	4		t_{CYX}	
SB0 high-level width	t_{WSBH}	4		t_{CYX}	

A/D Converter Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = AV_{SS} = 0 V

Item	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		8			Bit	
Full-scale error (Note 1)				0.4	%	AV_{REF} = 4.0 V to V_{DD}; T_A = −10 to +70°C
				0.8	%	AV_{REF} = 3.4 V to V_{DD}; T_A = −10 to +70°C
				0.8	%	AV_{REF} = 4.0 V to V_{DD}
Quantization error				±1/2	LSB	
Conversion time	t_{CONV}	360			t_{CYX}	83 ns ≤ t_{CYX} ≤ 125 ns (Note 2)
		240			t_{CYX}	125 ns ≤ t_{CYX} ≤ 250 ns (Note 3)
Sampling time	t_{SAMP}	72			t_{CYX}	83 ns ≤ t_{CYX} ≤ 125 ns (Note 2)
		48			t_{CYX}	125 ns ≤ t_{CYX} ≤ 250 ns (Note 3)
Analog input voltage	V_{IAN}	−0.3		AV_{REF} + 0.3	V	
Analog input impedance	R_{AN}		1000		MΩ	
Analog reference voltage	AV_{REF}	3.4		V_{DD}	V	
AV_{REF} current	AI_{REF}		1.5	5.0	mA	Operating mode, f_{XX} = 12 MHz
			0.2	1.5	mA	STOP mode

Note:

(1) Quantization errror is not included. Unit is defined as percent of full-scale value.

(2) FR bit of ADM register is 0.

(3) FR bit of ADM registers is 1.

Interrupt Timing Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V

Item	Symbol	Min	Max	Unit	Conditions
NMI low-level width	t_{WNIL}	10		μs	
NMI high-level width	t_{WNIH}	10		μs	
INTP0-INTP5 low-level width	t_{WITL}	24		t_{CYX}	
INTP0-INTP5 high-level width	t_{WITH}	24		t_{CYX}	
$\overline{\text{RESET}}$ low-level width	t_{WRSL}	10		μs	
$\overline{\text{RESET}}$ high-level width	t_{WRSH}	10		μs	

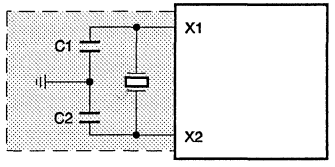
Data Retention Characteristics

T_A = −40 to +85°C

Item	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.5		5.5	V	STOP mode
Data retention current	I_{DDDR}		2	20	μA	V_{DDDR} = 2.5 V
			5	50	μA	V_{DDDR} = 5 V ±10%
V_{DD} rise time	t_{RVD}	200			μs	
V_{DD} fall time	t_{FVD}	200			μs	
V_{DD} retention time (from STOP mode setting)	t_{HVD}	0			ms	
STOP release signal input time	t_{DREL}	0			ms	
Oscillation stabilization wait time	t_{WAIT}	30			ms	Crystal resonator
		5			ms	Ceramic resonator
Low-level input voltage	V_{IL}	0		0.1 V_{DDDR}	V	Specified pins (Note 1)
High-level input voltage	V_{IH}	0.9 V_{DDDR}		V_{DDDR}	V	Specified pins (Note 1)

Note: $\overline{\text{RESET}}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2/CI, $P2_4$/INTP3, $P2_5$/INTP4/ASCK, $P2_6$/INTP5, $P2_7$/SI, $P3_2$/$\overline{\text{SCK}}$, $P3_3$/SO/SB0, and $\overline{\text{EA}}$ pins.

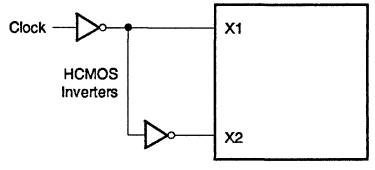
Recommended Resonator Circuit



Ceramic or crystal resonator frequency f_{XX} = 4 to 12 MHz
External oscillation circuit should be as close to the X1 and X2 pins as possible
Do not place other signal lines in the shaded area

83YL-9189A

Recommended External Clock Circuit



Clock frequency f_{XX} = 4 to 12 MHz
In STOP mode, X1 is internally shorted to V_{SS} to prevent leakage current. Therefore, STOP mode is not available when using this external clock circuit.

83YL-9190A

External Clock Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V

Item	Symbol	Min	Max	Unit	Conditions
X1 input low-level width	t_{WXL}	30	130	ns	
X1 input high-level width	t_{WXH}	30	130	ns	
X1 input rise time	t_{XR}	0	30	ns	
X1 input fall time	t_{XF}	0	30	ns	
X1 input clock cycle time	t_{CYX}	82	250	ns	

Recommended Ceramic Resonators

Manufacturer	Frequency (MHz)	Part Number	C1 (pF)	C2 (pF)
Murata Mfg.	12	CSA12.0MT	30	30
		CST12.0MTW (1)	None (2)	None (2)
	4	CSA4.00MG040	100	100
		CST4.00MG040	None (2)	None (2)
Kyocera Corp.	12	KBR12.0M	33	33

Notes:

(1) Recommended for μPD78212/213/214 only.

(2) C1 and C2 are contained in the resonator.

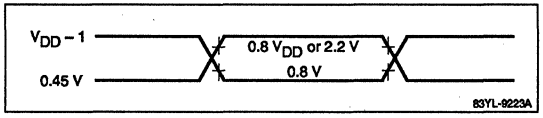
Recommended Crystal Resonators

Manufacturer	Frequency (MHz)	Part Number	C1 (pF)	C2 (pF)
Kinseki	12	HC-49/U	18	18

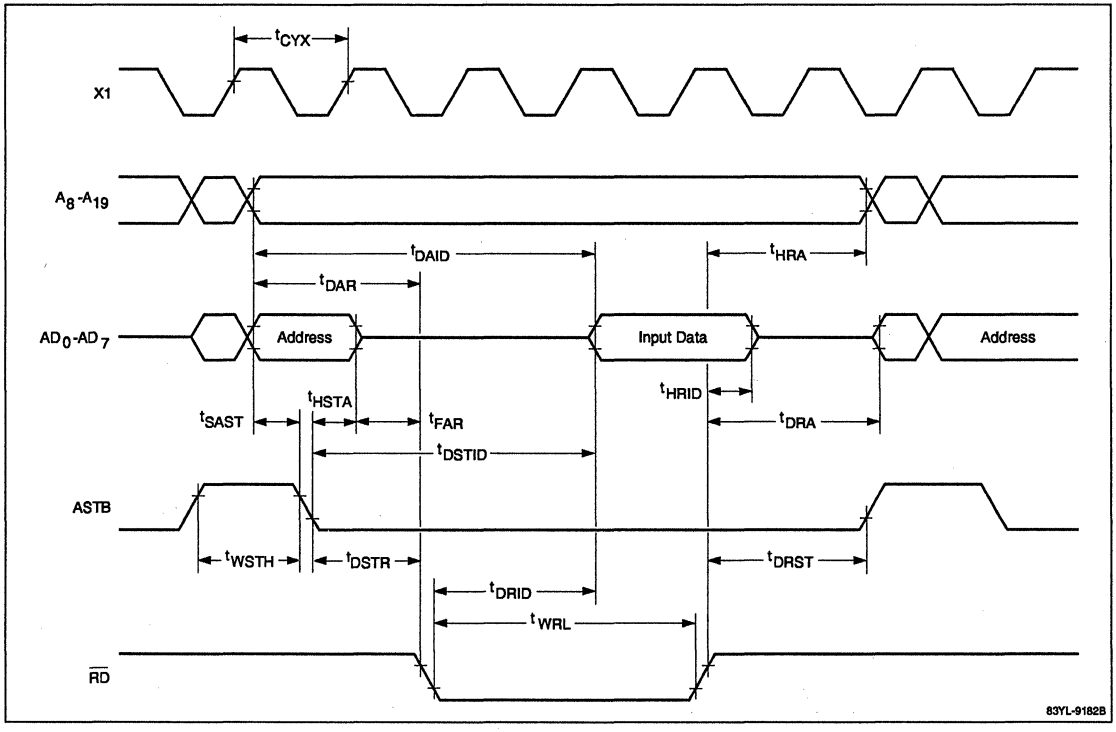
4a

Timing Waveforms

Voltage Thresholds for AC Timing Measurements

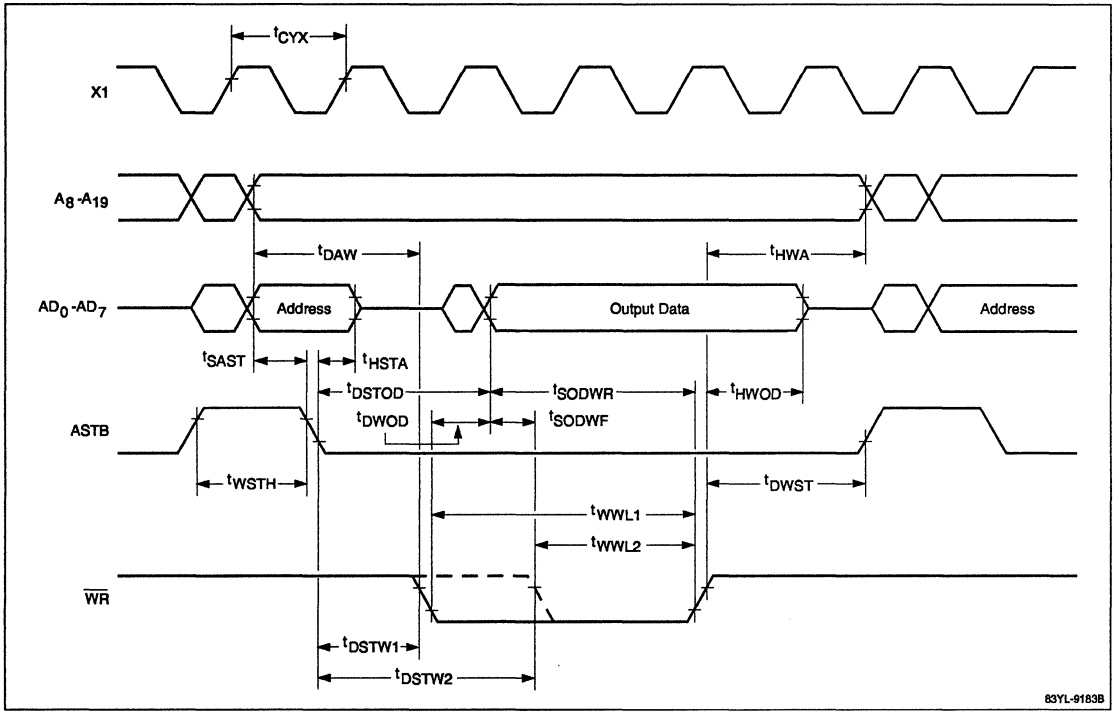


Read Operation



Timing Waveforms (cont)

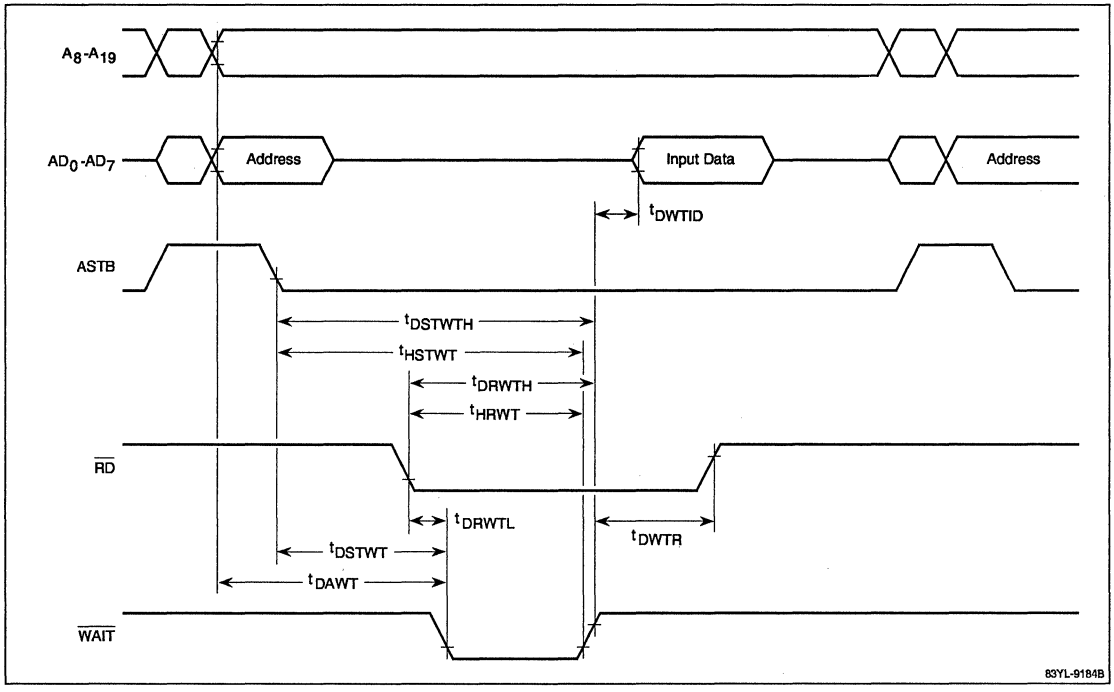
Write Operation



4a

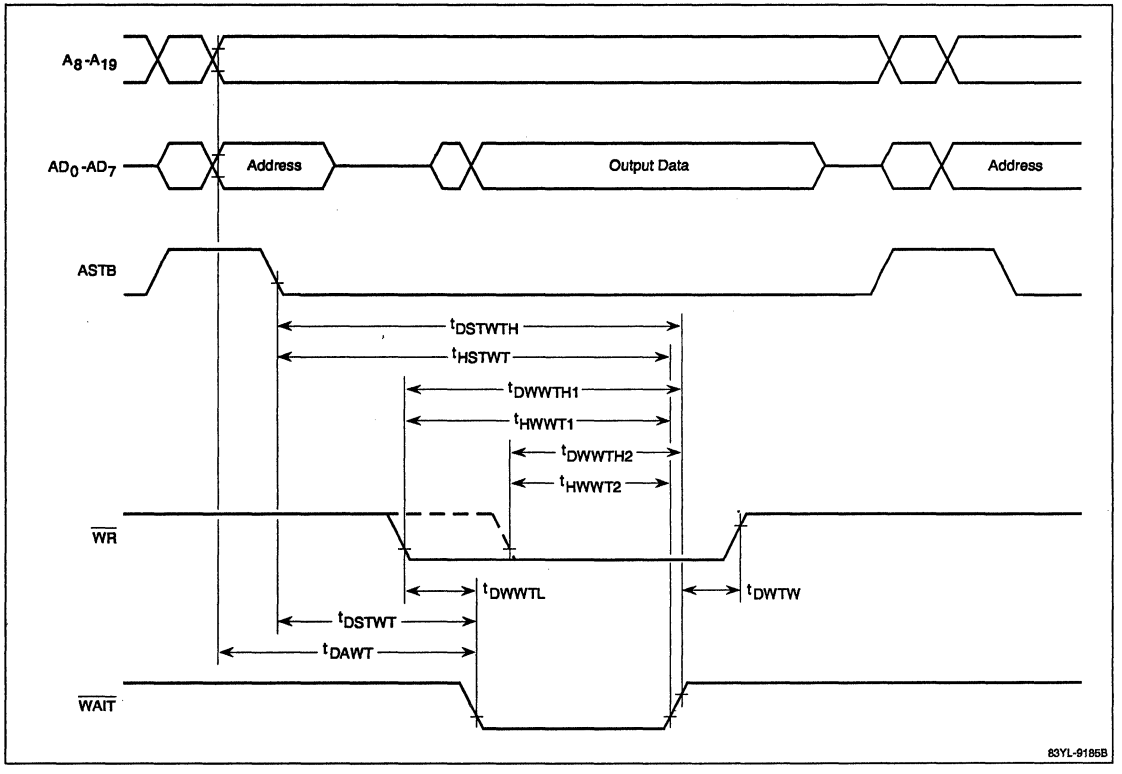
Timing Waveforms (cont)

External $\overline{WAIT}$ Signal Input (Read Operation)



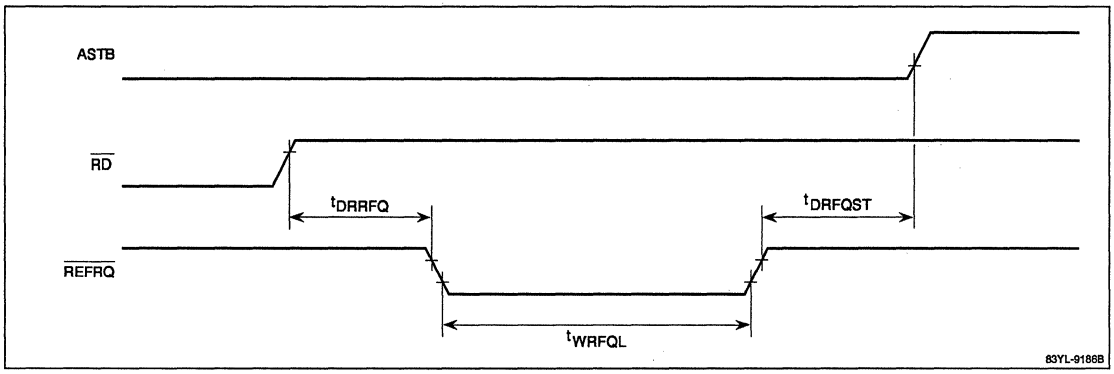
Timing Waveforms (cont)

External $\overline{WAIT}$ Signal Input (Write Operation)

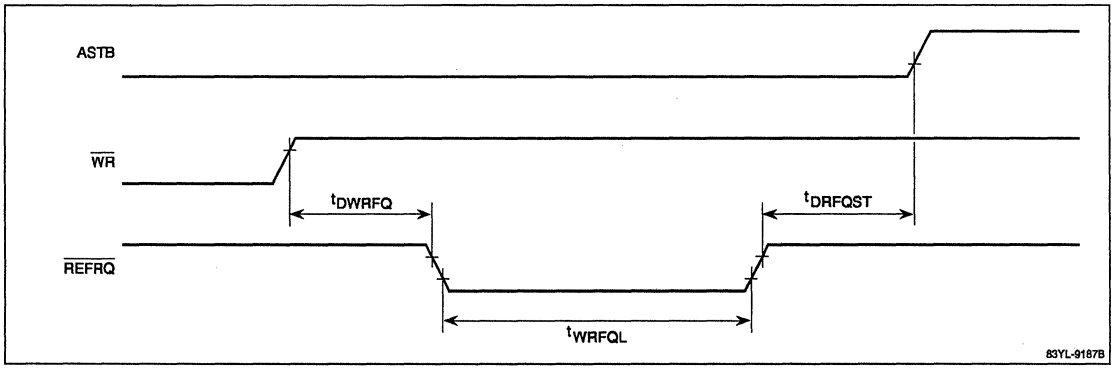


Timing Waveforms (cont)

Refresh After Read



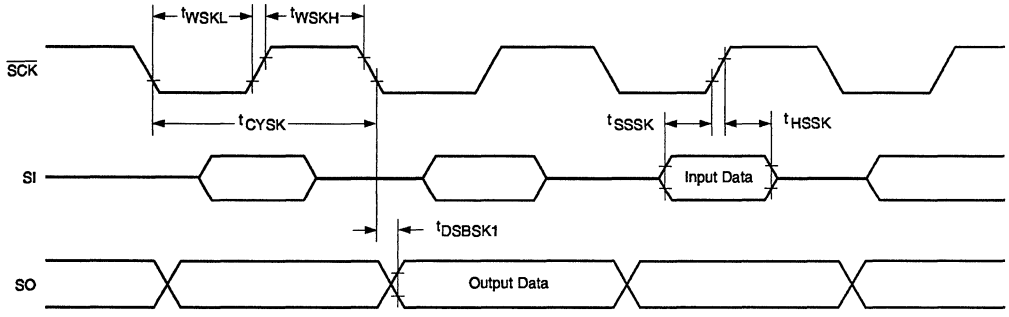
Refresh After Write



Timing Waveforms (cont)

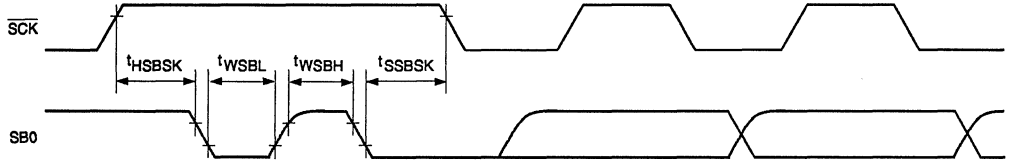
Serial Operation

Three-Line Serial I/O Timing

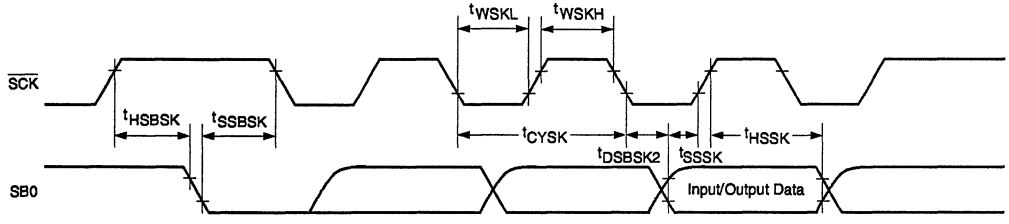


SBI Mode

Bus Release Signal Transfer Timing



Command Signal Transfer Timing

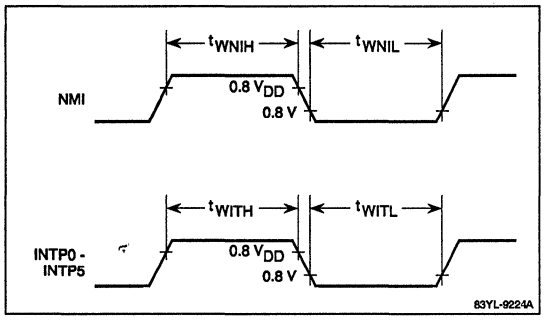


83YL-9188B

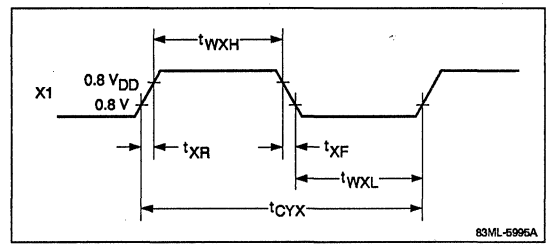
4a

Timing Waveforms (cont)

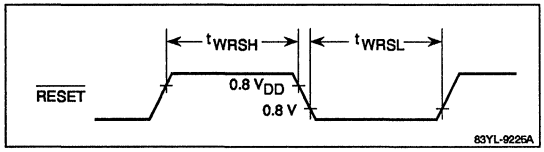
Interrupt Input



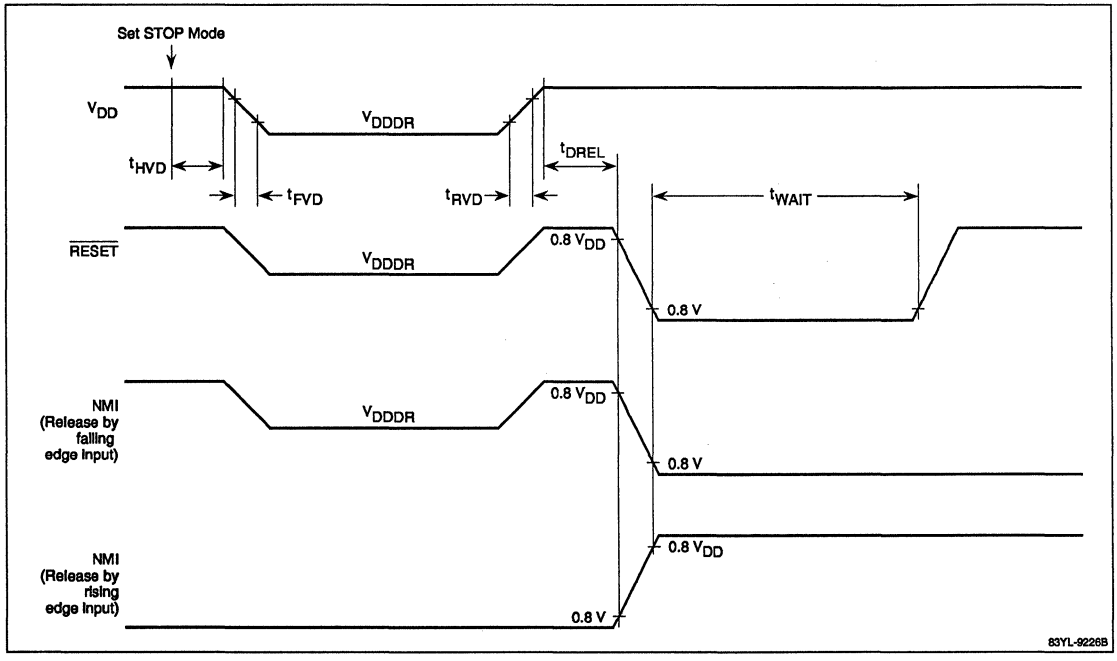
External Clock



Reset Input



Data Retention Characteristics



μPD78P214 PROGRAMMING

In the μPD78P214, the mask ROM of μPD78214 is replaced by a one-time programmable ROM (OTP ROM) or a reprogrammable, ultraviolet erasable ROM (UV EPROM). The ROM is 16,384 x 8 bits and can be programmed using a general-purpose PROM writer with a μPD27C256A programming mode.

The PA-78P214CW/GC/GJ/GQ/L are the socket adaptors used for configuring the μPD78P214 to fit a standard PROM socket.

Refer to tables 5 and 6 and figures 15 through 18 for special information applicable to PROM programming.

Table 5. Pin Functions During PROM Programming

Pin	Pin*	Function
$P0_0$ - $P0_7$	A_0-A_7	Address input pins for PROM operations
$P5_0/A_8$	A_8	Address input pin for PROM operations
$P2_1$/INTP0	A_9	Address input pin for PROM operations
$P5_2/A_{10}$ - $P5_6/A_{14}$	A_{10} - A_{14}	Address input pins for PROM operations
$P4_0/AD_0$ - $P4_7/AD_7$	D_0 - D_7	Data pins for PROM operations
$P6_5/\overline{WR}$	$\overline{CE}$	Strobes data into the PROM
$P6_4/\overline{RD}$	$\overline{OE}$	Enables a data read from the PROM
$P2_0$/NMI	NMI	PROM programming mode is entered by applying +12.5 volts to this pin
$\overline{RESET}$	$\overline{RESET}$	PROM programming mode requires applying a low voltage to this pin
$\overline{EA}$	V_{PP}	High voltage applied to this pin for program write/verify
V_{DD}	V_{DD}	Positive power supply pin
V_{SS}	V_{SS}	Ground

* Pin name in PROM programming mode.

4a

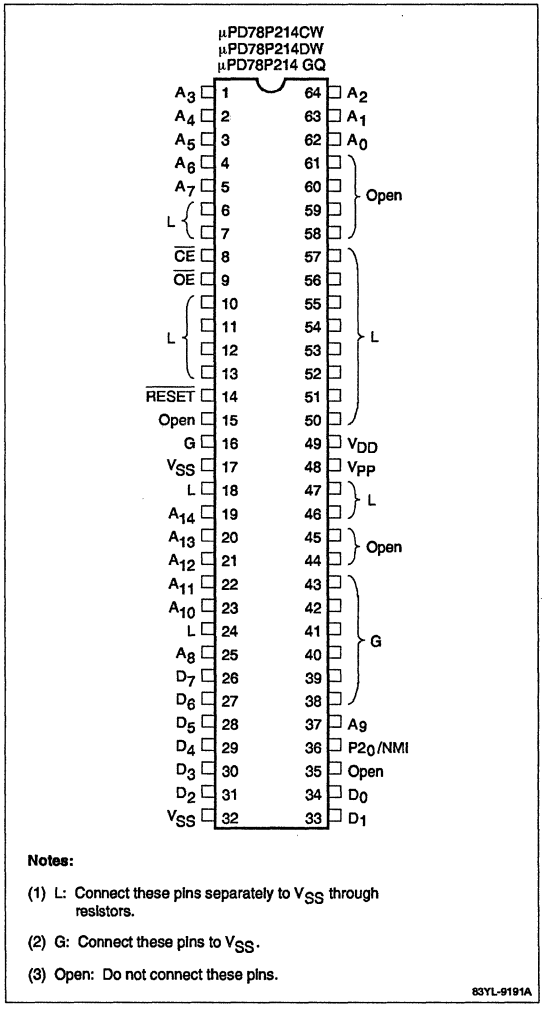
Table 6. Summary of Operation Modes for PROM Programming

Mode	NMI	$\overline{RESET}$	$\overline{CE}$	$\overline{OE}$	V_{PP}	V_{DD}	D_0 - D_7
Program write	+12.5 V	L	L	H	+12.5 V	+6 V	Data input
Program verify	+12.5 V	L	H	L	+12.5 V	+6 V	Data output
Program inhibit	+12.5 V	L	H	H	+12.5 V	+6 V	High Z
Read out	+12.5 V	L	L	L	+5 V	+5 V	Data output
Output disable	+12.5 V	L	L	H	+5 V	+5 V	High Z
Standby	+12.5 V	L	H	L/H	+5 V	+5 V	High Z

Note: When +12.5 V is applied to V_{PP} and +6 V to V_{DD}, both $\overline{CE}$ and $\overline{OE}$ cannot be set to low level (L) simultaneously.

Pin Functions in μPD78P214 PROM Programming Mode

Figure 15. 64-Pin Plastic and Ceramic Shrink DIP 64-Pin Plastic QUIP



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors.

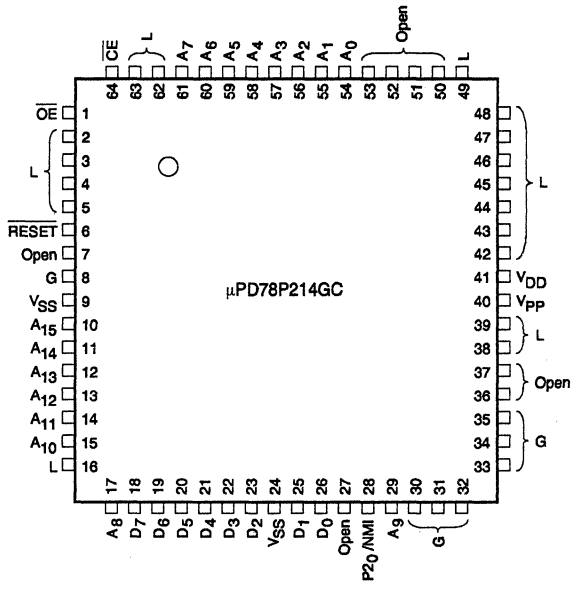
(2) G: Connect these pins to V_{SS}.

(3) Open: Do not connect these pins.

83YL-9191A

Pin Functions in μPD78P214 PROM Programming Mode (cont)

Figure 16. 64-Pin Plastic QFP



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors.

(2) G: Connect these pins to V_{SS}.

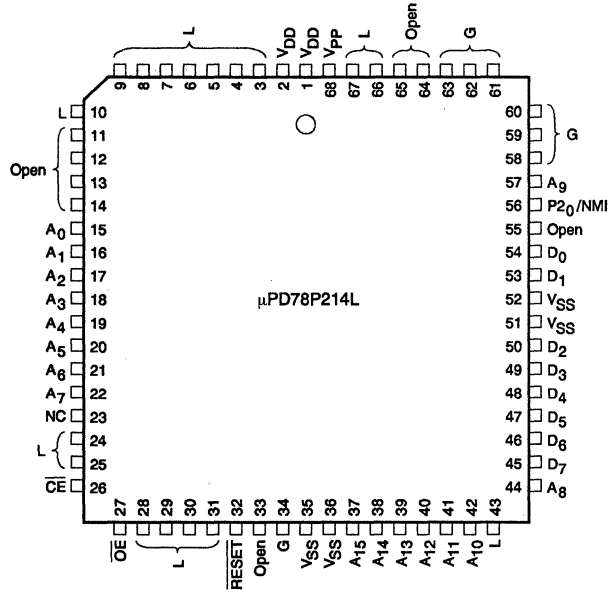
(3) Open: Do not connect these pins.

83YL-9192B

4a

Pin Functions in μPD78P214 PROM Programming Mode (cont)

Figure 17. 68-Pin PLCC



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors.

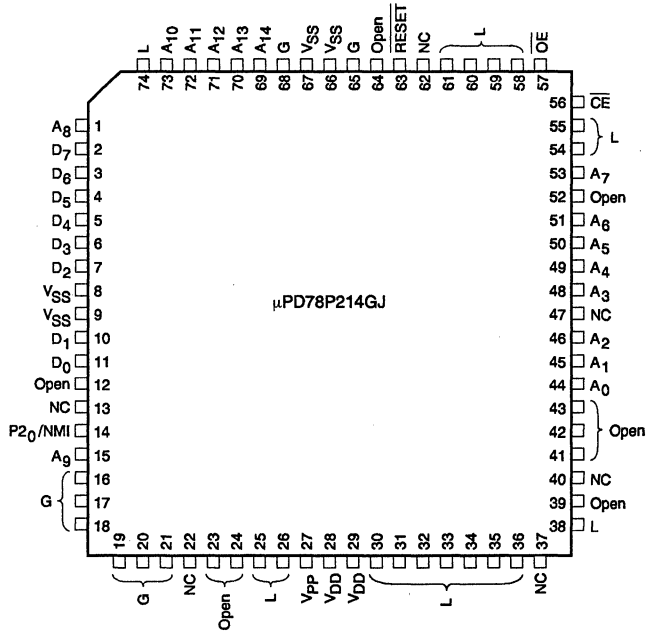
(2) G: Connect these pins to V_{SS}.

(3) Open: Do not connect these pins.

83YL-9193B

Pin Functions in μPD78P214 PROM Programming Mode (cont)

Figure 18. 74-Pin Plastic QFP



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors.

(2) G: Connect these pins to V_{SS}.

(3) Open: Do not connect these pins.

83YL-9194B

4a

PROM Write Procedure

(1) Set the pins not used for programming as indicated in figures 15 through 18. Connect the $\overline{RESET}$ pin to a low level, the V_{DD} and V_{PP} pins to +5 V, and apply +12.5 V to the NMI pin. The $\overline{CE}$ and $\overline{OE}$ pins should be high.

(2) Apply +6 V to the V_{DD} pin and +12.5 V to the V_{PP} pin.

(3) Provide the initial address to the A_0 to A_{14} pins.

(4) Provide write data.

(5) Provide 1-ms program pulse (active low) to the $\overline{CE}$ pin.

(6) This data is now verified with a pulse (active low) to the $\overline{OE}$ pin. If the data has been written, proceed to step 8; if not, repeat steps 4 to 6. If the data cannot be correctly written after 25 attempts, go to step 7.

(7) Classify as defective and stop write operation.

(8) Provide write data and supply program pulse (for additional writing) for 3 ms times the number of writes performed in step 5.

(9) Increment the address.

(10) Repeat steps 4 to 9 until the end address. NEC reserves address 4000H for future functional extension. If a PROM writer cannot specify a final programming address, FFH must be written in address 4000H.

PROM Read Procedure

(1) Set the pins not used for programming as indicated in figures 15 through 18. Fix the $\overline{RESET}$ pin to a low level, the V_{DD} and V_{PP} pins to +5 V, and apply +12.5 V to the NMI pin. The $\overline{CE}$ and $\overline{OE}$ pins should be high.

(2) Input the address of the data to be read to pins A_0 - A_{14}.

(3) Read mode is entered with a pulse (active low) on both the $\overline{CE}$ and $\overline{OE}$ pins.

(4) Data is output to the D_0 to D_7 pins.

EPROM Erasure

Data in an EPROM is erased by exposing the quartz window in the ceramic package to light having a wavelength shorter than 400 nm, including ultraviolet rays, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15 Ws/cm^2 (ultraviolet ray intensity x exposure time) is required to completely erase written data. Erasure by an ultraviolet lamp rated at 12 mW/cm^2 takes approximately 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

DC Programming Characteristics

T_A = 25 ±5°C; V_{IP} = 12.5 ±0.5 V applied to NMI pin; V_{SS} = 0 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH}	V_{IH}	2.4		V_{DDP}+0.3	V	
Low-level input voltage	V_{IL}	V_{IL}	−0.3		0.8	V	
Input leakage current	I_{LIP}	I_{LI}			10	μA	$0 \le V_I \le V_{DDP}$
High-level output voltage	V_{OH1}	V_{OH1}	2.4			V	I_{OH} = −400 μA
	V_{OH2}	V_{OH2}	V_{DD} − 0.7			V	I_{OH} = −100 μA
Low-level output voltage	V_{OL}	V_{OL}			0.45	V	I_{OL} = 2.1 mA
Output leakage current	I_{LO}				10	μA	$0 \le V_O \le V_{DDP}$, $\overline{OE}$ = V_{IH}
NMI pin high-voltage input current	I_{IP}				±10	μA	
V_{DDP} power voltage	V_{DDP}	V_{CC}	5.75	6.0	6.25	V	Program memory write mode
			4.5	5.0	5.5	V	Program memory read mode
V_{PP} power voltage	V_{PP}	V_{PP}	12.2	12.5	12.8	V	Program memory write mode
				V_{PP} = V_{DDP}		V	Program memory read mode
V_{DDP} power current	I_{DD}	I_{CC}		5	30	mA	Program memory write mode
				5	30	mA	Program memory read mode $\overline{CE}$ = V_{IL}, V_I = V_{IH}
V_{PP} power current	I_{PP}	I_{PP}		5	30	mA	Program memory write mode $\overline{CE}$ = V_{IL}, $\overline{OE}$ = V_{IH}
				1	100	μA	Program memory read mode

* Corresponding symbols of the μPD27C256A.

AC Programming Characteristics (Write Mode)

T_A = 25 ±5°C; V_{IP} = 12.5 ±0.5 V applied to NMI pin; V_{SS} = 0 V; V_{DD} = 6 ±0.25 V; V_{PP} = 12.5 ±0.3 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Conditions
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}	2			μs	
Data input to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}	2			μs	
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}	2			μs	
Address hold time from $\overline{CE}$ ↑	t_{HCA}	t_{AH}	2			μs	
Input data hold time from $\overline{CE}$ ↑	t_{HCID}	t_{DH}	2			μs	
Output data hold time from $\overline{OE}$ ↑	t_{HOOD}	t_{DF}	0		130	ns	
V_{PP} setup time to $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}	1			ms	
V_{DDP} setup time to $\overline{CE}$ ↓	t_{SVDC}	t_{VCS}	1			ms	
Initial program pulse width	t_{WL1}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{WL2}	t_{OPW}	2.85		78.75	ms	
NMI high-voltage input setup time to $\overline{CE}$ ↓	t_{SPC}		2			μs	
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			150	ns	

* Corresponding symbols of the μPD27C256A.

4a

AC Programming Characteristics (Read Mode)

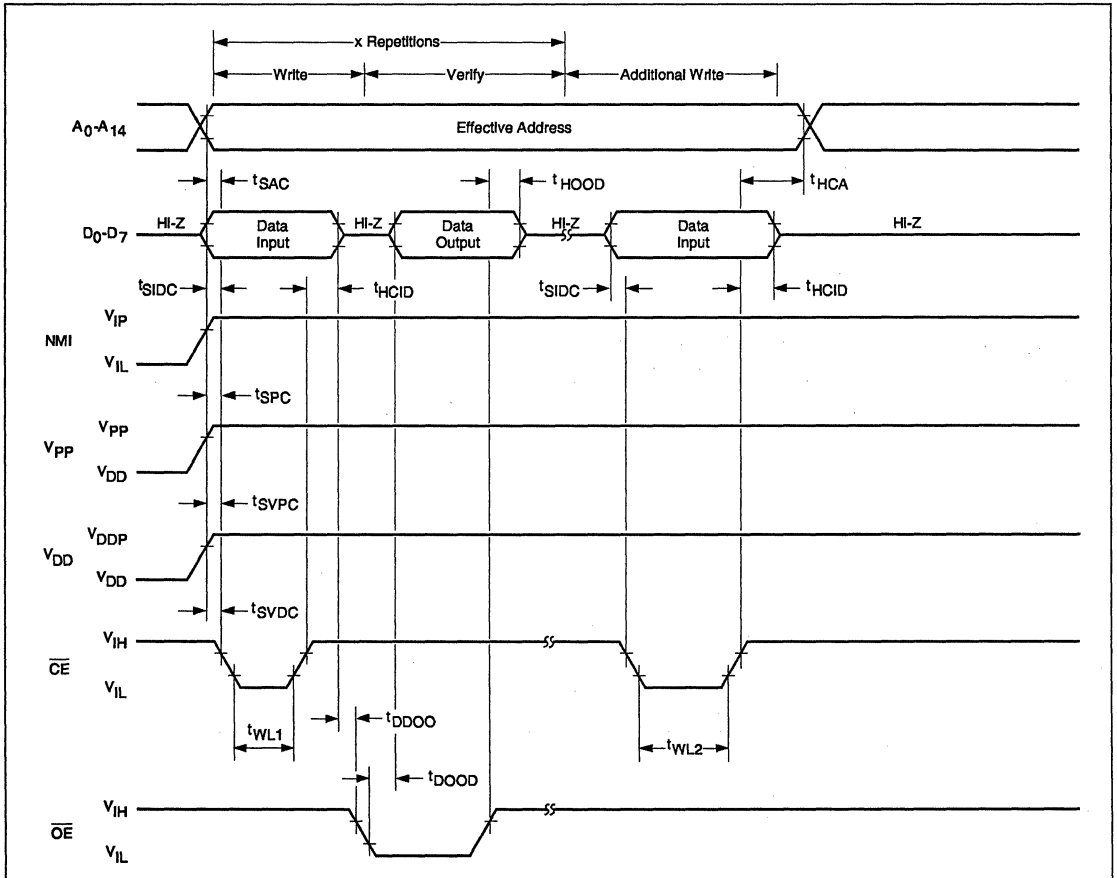
T_A = 25 ±5°C; V_{IP} = 12.5 ±0.5 V applied to NMI pin; V_{SS} = 0 V; V_{DD} = 5 ±0.5 V; V_{PP} = V_{DDP}

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
Address to data output time	t_{DAOD}	t_{ACC}			200	ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}
$\overline{CE}$ ↓ to data output time	t_{DCOD}	t_{CE}			200	ns	$\overline{OE}$ = V_{IL}
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			75	ns	$\overline{CE}$ = V_{IL}
Data hold time from $\overline{OE}$ ↑	t_{HCOD}	t_{DF}	0		60	ns	$\overline{CE}$ = V_{IL}
Data hold time from address	t_{HAOD}	t_{OH}	0			ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}

* Corresponding symbols of the μPD27C256A.

PROM Timing Diagrams

PROM Write/Verify Mode

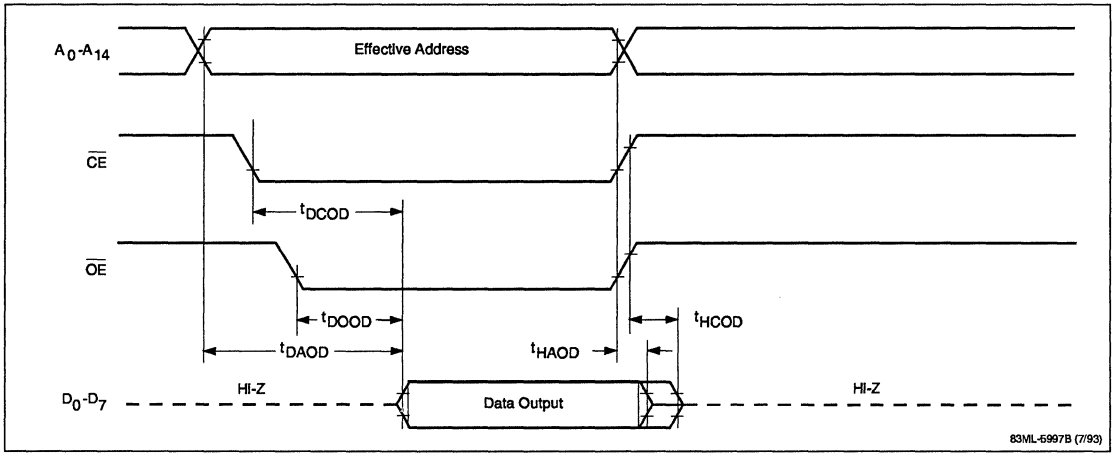


Notes:
(1) V_{DD} must be applied before applying V_{PP}. It should be removed after removing V_{PP}.
(2) V_{PP} must not exceed +13V, including overshoot.

83ML-5996B

PROM Timing Diagrams (cont)

PROM Read Mode



4a

NEC
NEC Electronics Inc.

μPD78218A Family (μPD78217A/218A/P218A) 8-Bit, K-Series Microcontrollers With A/D Converter, Real-Time Output Ports

July 1993

Description

The μPD78217A, μPD78218A, and μPD78P218A are members of the K-Series® of microcontrollers and are designed for real-time embedded control applications. The μPD78218A family is pin compatible with the μPD78214 family and offers increased internal memory with enhanced timer and macro service facilities. These 8-bit, single-chip microcontrollers have a minimum instruction time of 333 ns at 12 MHz (500 ns for the μPD78217A). They feature 8-bit hardware multiply and divide instructions, four banks of main registers, an advanced interrupt handling facility, a powerful set of memory mapped on-chip peripherals, and the ability to address up to 1M bytes of external data memory. On board memory includes 1024 bytes of RAM, 32K bytes of mask ROM, or 32K bytes of UV EPROM or one-time programmable (OTP) ROM.

The advanced interrupt handling facility provides two levels of programmable hardware priority control and two separate methods of servicing interrupt requests: vectored and macro service. The macro service facility reduces the overhead involved in servicing peripheral interrupts by transferring data between the memory-mapped special function registers (SFRs) and memory without the use of time consuming interrupt service routines. In addition, the macro service facility can be initialized to automatically alter timer compare register values or to repeatedly output a prespecified pattern at a fixed or variable rate. By using macro service to control the real-time output ports, the μPD78218A family can easily and accurately drive two independent stepper motors.

The combination of the macro service facility, four banks of main registers, extended data memory address space, and powerful on-chip peripherals makes these devices ideal for applications in office automation, communication, HVAC, and industrial control.

Features

- Complete single-chip microcontroller
  - 8-bit ALU
  - Program memory (ROM)
    μPD78217A: ROMless
    μPD78218A/P218A: 32K bytes
  - Data memory (RAM): 1024 bytes
- Pin compatible with μPD78214 family
- Powerful instruction set
  - 8-bit unsigned multiply and divide
  - 16-bit arithmetic instructions
  - 1-bit and 8-bit logic instructions
- Minimum instruction time
  - 333 ns at 12 MHz (μPD78218A/P218A)
  - 500 ns at 12 MHz (μPD78217A)
- Memory expansion
  - 8085 bus-compatible
  - 64K program address space
  - 1M data address space
- Large I/O capacity
  - Up to 54 I/O port lines on μPD78218A/P218A
  - Up to 36 I/O port lines on μPD78217A
  - Software programmable pullup resistors
- Memory-mapped on-chip peripherals (special function registers)
- Timer/counter unit
  - 16-bit timer 0:
    Two 16-bit compare registers
    One 16-bit capture register
    One external interrupt/capture line
  - 8-bit timer 1:
    One 8-bit compare register
    One 8-bit capture/compare register
    One external interrupt/capture line
  - 8-bit timer/counter 2:
    Two 8-bit compare registers
    One 8-bit capture register
    One external interrupt/capture line
    One external event counter line
  - 8-bit timer 3:
    One 8-bit compare register
- Four 8-bit precision timer-controlled pulse-width modulated (PWM) output lines
- Two 4-bit (or one 8-bit) real-time output ports
- Eight-channel 8-bit A/D converter
- Programmable priority interrupt controller (two levels)
- Two methods of interrupt service
  - Vectored interrupts
  - Macro service mode with choice of three different types

K-series is a registered trademark of NEC Electronics, Inc.

Features (cont)

- Two-channel serial communication interface
  - — Asynchronous serial interface (UART)
    Dedicated baud rate generator
  - — Clock-synchronized interface
    Full-duplex, three-wire mode
    NEC serial bus interface (SBI) mode
- Refresh output for pseudostatic RAM
- STOP and HALT standby functions
- 5-volt CMOS technology

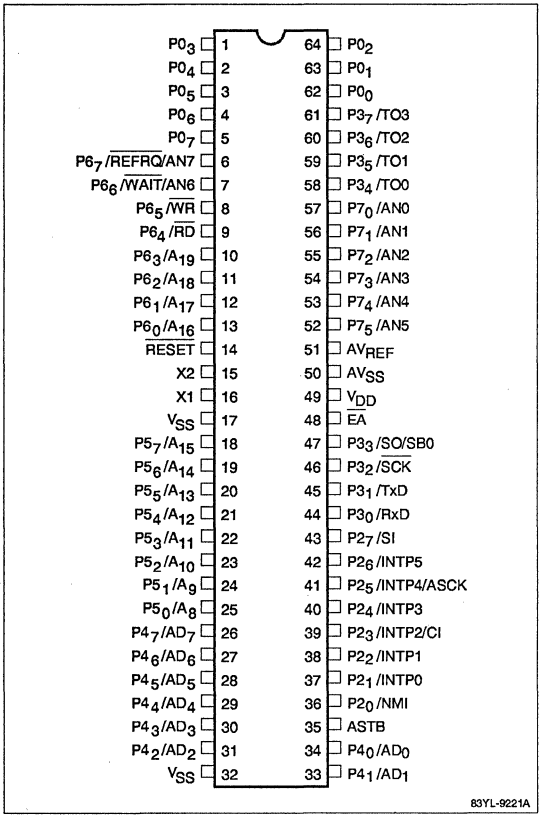
Ordering Information

Part Number	Package	Package Drawing	ROM
μPD78217ACW	64-pin plastic shrink DIP	(P64C-70-750A, C)	ROMless
μPD78217AGC	64-pin plastic QFP	(P64GC-80-AB8-2)	
μPD78218ACW-xxx	64-pin plastic shrink DIP	(P64C-70-750A, C)	32K mask ROM
μPD78218AGC-xxx	64-pin plastic QFP	(P64GC-80-AB8-2)	
μPD78P218ACW	64-pin plastic shrink DIP	(P64C-70-750A, C)	32K OTP ROM
μPD78P218AGC	64-pin plastic QFP	(P64GC-80-AB8-2)	
μPD78P218ADW	64-pin shrink cerdip w/ window	(P64DW-70-750A1)	32K UV EPROM

Note: xxx indicates ROM code suffix

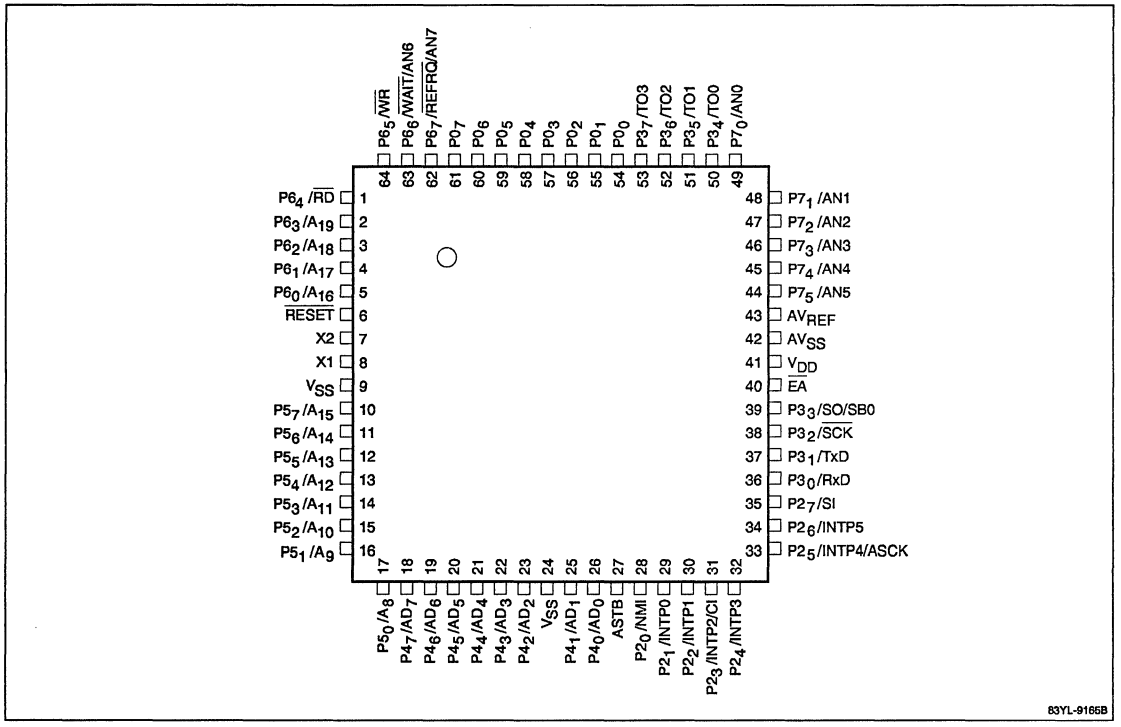
Pin Configurations

64-Pin Shrink DIP (Plastic or Ceramic)



Pin Configurations (cont)

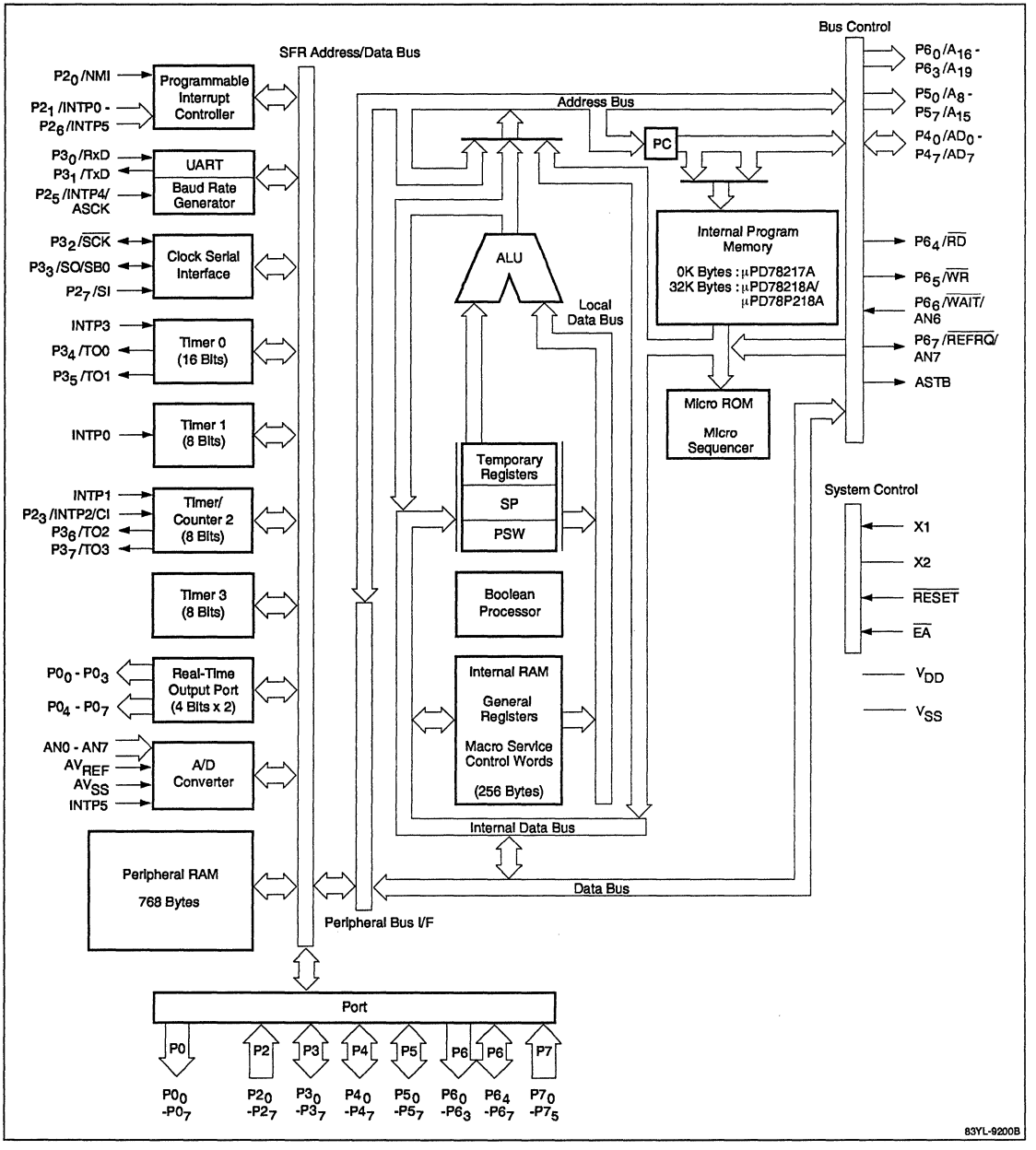
64-Pin Plastic QFP



Pin Functions; Normal Operating Mode

Symbol	First Function	Symbol	Second Function
$P0_0$ - $P0_7$	Port 0; 8-bit tristate output port/real time output port		
$P2_0$	Port 2; 8-bit input port	NMI	External nonmaskable interrupt
$P2_1$		INTP0	Maskable external interrupts
$P2_2$		INTP1	
$P2_3$		INTP2	Maskable external interrupt
		CI	External clock input to timer/counter 2
$P2_4$		INTP3	Maskable external interrupt
$P2_5$		INTP4	Maskable external interrupt
		ASCK	Asynchronous serial clock input
$P2_6$		INTP5	Maskable external interrupt
$P2_7$		SI	Serial data input for three-wire serial I/O mode
$P3_0$	Port 3; 8-bit, bit-selectable tristate input/output port	RxD	Asynchronous serial receive data input
$P3_1$		TxD	Asynchronous serial transmit data output
$P3_2$		$\overline{SCK}$	Serial shift clock input/output
$P3_3$		SO	Serial data output for three-wire serial I/O mode
		SBO	I/O bus for NEC serial bus interface (SBI)
$P3_4$ - $P3_7$		TO0 - TO3	Timers T0 to T3 outputs
$P4_0$ - $P4_7$	Port 4; 8-bit tristate input/output port	AD_0 - AD_7	Low-order 8-bit multiplexed address/data bus
$P5_0$ - $P5_7$	Port 5; 8-bit, bit-selectable tristate input/output port	A_8 - A_{15}	High-order 8-bit address bus
$P6_0$ - $P6_3$	Port 6; 4-bit output port	A_{16} - A_{19}	Extended memory address bus
$P6_4$	Port 6; 4-bit, bit-selectable tristate input/output port	$\overline{RD}$	External memory read strobe
$P6_5$		$\overline{WR}$	External memory write strobe
$P6_6$		$\overline{WAIT}$	External memory wait signal input
		AN6	Analog voltage input to A/D converter
$P6_7$		$\overline{REFRQ}$	Refresh pulse output used by external pseudostatic memory
		AN7	Analog voltage input to A/D converter
$P7_0$ - $P7_5$	Port 7; 6-bit input port	AN0 - AN5	Analog voltage inputs to A/D converter
ASTB	Address strobe output used to latch the low-order 8 address for external memory		
$\overline{RESET}$	External system reset input		
$\overline{EA}$	Internal ROM or external memory control signal input. Low-level input selects external memory. High-level input selects internal ROM. A low-level input on a μPD78218A or μPD78P218A places the device in ROMless mode and external memory is accessed.		
X1	Crystal/ceramic resonator connection or external clock input		
X2	Crystal/ceramic resonator connection or inverse of external clock		
AV_{REF}	A/D converter reference voltage		
AV_{SS}	A/D converter ground		
V_{DD}	+5 volt power supply input		
V_{SS}	Power supply ground		
NC	No connection		

Block Diagram



4b

μPD78218A and μPD78214 Differences

The μPD78218A family is a pin compatible enhanced version of the μPD78214 family. Some of the enhancements include a larger internal program ROM and data RAM memory space, an improved 16-bit timer 0, and an enhanced macro service facility. Table 1 highlights the differences between the μPD78218A and μPD78214 families.

Table 1. Differences Between the μPD78218A and μPD78214 Families

Item	μPD78218A Family	μPD78214 Family
Maximum on-chip ROM	32K bytes	16K bytes
Maximum on-chip RAM	1024 bytes	512 bytes
16-bit timer	Software-triggered one-shot pulse output	Not available
Macro service counter	8/16-bit selectable (except Type A transfers)	8-bit only
Type C macro service pointer, MPT and MPD	Increments full 16 bits	Increments only lower 8 bits
Macro service execution times	Execution times differ; refer to hardware user's manual	Execution times differ; refer to hardware user's manual
PUSH PSW instruction	Execution times differ; refer to software user's manual	Execution times differ; refer to software user's manual
Oscillation stabilization time when exiting STOP mode	Time equivalent to NMI active pulse width plus 16 bits of dedicated counter or 15 bits of dedicated counter	Time equivalent to NMI active pulse width plus 16 bits of dedicated counter
A/D converter reference voltage	3.6 V to V_{DD}	3.4 V to V_{DD}
Programmable device operating voltage	5 V ±0.3 V	5 V ±10%
Package	64-pin plastic shrink DIP	64-pin plastic shrink DIP
	64-pin plastic QFP	64-pin plastic QFP
	64-pin shrink cerdip w/window	64-pin shrink cerdip w/window
		64-pin plastic QUIP
		68-pin PLCC
		74-pin plastic QFP

FUNCTIONAL DESCRIPTION

Central Processing Unit (CPU)

The μPD78218A family CPU features 8- and 16-bit arithmetic including an 8 x 8-bit unsigned multiply and 16 x 8-bit unsigned divide (producing a 16-bit quotient and an 8-bit remainder). The multiply executes in 3.67 μs and the divide in 12.36 μs at 12 MHz (4.00 and 12.69 μs respectively for μPD78217A).

A CALLT vector table and a CALLF program area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction (CALLT) can access up to 32 subroutines through the addresses contained in the CALLT vector table. A 2-byte call instruction (CALLF) can access any routine beginning at a specific address in the CALLF area.

The internal system clock (f_{CLK}) is generated by dividing the oscillator frequency by two. Therefore, at the maximum oscillator frequency of 12 MHz, the internal system clock is 6 MHz. The minimum instruction execution time for an instruction fetched from internal ROM is 333 ns (500 ns when fetched from external memory).

Memory Space

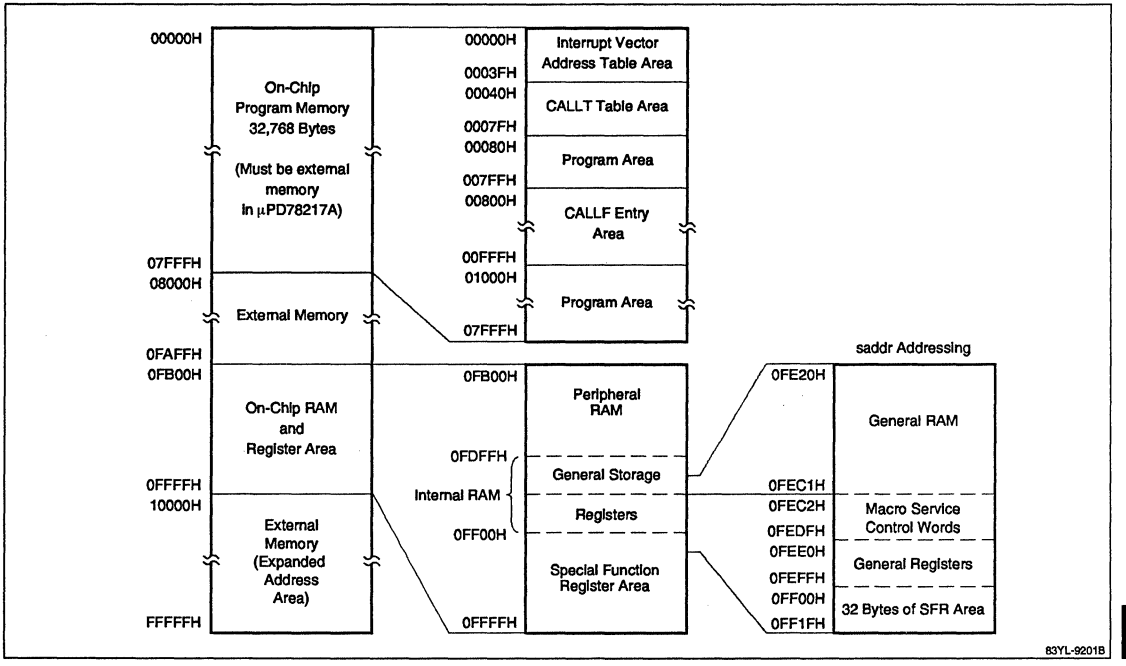
The μPD78218A family has a 1M byte address space (see figure 1). The first 64K bytes of this address space (00000H-0FFFFH) can be used as both program and data memory. The remaining 960K bytes of this address space (10000H-FFFFFH) can only be used as data memory and is known as expanded memory.

External Memory

The μPD78218A family has an 8-bit wide external data bus and a 16-bit wide external address bus (20-bit wide if expanded memory is enabled). The low-order 8 bits of the address bus are multiplexed to provide the 8-bit data bus and are supplied by I/O port 4. The high-order address bits of the 16-bit address bus are taken from port 5. If expanded memory is enabled, the expanded address nibble is provided by $P6_0$ to $P6_3$. Address latch, read, and write strobes are also provided.

The memory expansion mode register (MM) is used to enable external memory, to specify up to two additional wait states or the use of the $\overline{WAIT}$ input pin for the first 64K bytes of memory, and to enable the high-speed internal ROM fetch. Ports 4, 5, and 6 are available as general purpose I/O ports when only internal ROM is used and no external program or data space is required.

Figure 1. Memory Map



Expanded Data Memory

The MM register is also used to enable the external expanded data memory space, addresses 10000H to FFFFFH. When the expanded data memory is enabled, the entire 1M byte address space is divided into 16 banks of 64K bytes each. The low-order 4-bits of the P6 or the PM6 registers are used as bank selection registers to supply the address information to A_{16} to A_{19}. Data can easily be transferred from one memory bank to another by using the appropriate instructions. Address lines A_{16} to A_{19} are only active when an instruction that uses expanded addressing is being executed. A programmable wait control register (PW) allows the programmer to specify up to two additional wait states or the use of the $\overline{WAIT}$ input pin for expanded data memory space.

On-Chip RAM

The μPD78218A family has a total of 1024 bytes of on-chip RAM. The upper 256-byte area (FE00H-FEFFH) features high-speed access and is known as "Internal RAM." The remaining 768 bytes (FB00H-FDFFH) are accessed at the same speed as external memory and are known as "Peripheral RAM." The general register banks and the macro service control words are stored in Internal RAM. The remainder of Internal RAM and any unused register bank locations are available for general storage.

On-Chip Program Memory

The μPD78218A contains 32K bytes of internal ROM respectively. The μPD78P218A contains 32K bytes of UV EPROM or one-time programmable ROM. Instructions from on-chip program memory can be fetched at high speed or at the same rate as from external memory. The μPD78217A does not have on-chip program memory.

CPU Control Registers

Program Counter. The program counter is a 16-bit binary counter register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 00001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the

stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

Program Status Word. The program status word (PSW) is an 8-bit register that contains flags that are set or reset depending on the results of an instruction. This register can be written to or read from 8 bits at a time. The individual flags can also be manipulated on a bit-by-bit basis. The assignment of PSW bits follows.

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

CY	Carry flag
ISP	Interrupt priority status flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

General Registers

The general-purpose registers (figure 2) consist of four banks of registers located at addresses FEE0H to FEFFH in Internal RAM. Each bank consists of eight 8-bit general registers that can also be used in pairs to function as four 16-bit registers. Two bits in the PSW (RBS0 and RBS1) specify which of the register banks is active. The bits are set under program control. Registers have both functional names (like A, X, B, C for 8-bit registers and AX, BC for 16-bit registers) and absolute names (like R1, R0, R3, R2 for 8-bit registers and RP0, RP1 for 16-bit registers). Each instruction determines whether a register is referred to by its functional or absolute name and whether it is 8 or 16 bits.

Addressing

The μPD78218A family features 1-byte addressing of both the special function registers and the portion of on-chip RAM from FE20H to FEFFH. The 1-byte sfr addressing accesses the entire SFR area, while the 1-byte saddr addressing accesses the first 32 bytes of the SFR area and 224 bytes of Internal RAM. The 16-bit SFRs and words of memory in these areas can be addressed by 1-byte saddrp addressing, which is valid for even addresses only. Since many instructions use 1-byte addressing, access to these locations is almost as fast and versatile as access to the general registers. There are seven addressing modes for data in main memory: direct, register, register indirect with autoincrement and decrement, saddr, SFR, based, and indexed. There are also both 8-bit and 16-bit immediate operands.

Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and the CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by 1-byte sfr addressing. They are either 8 or 16 bits as required, and many of the 8-bit registers are capable of single-bit access as well. Locations FFD0H through FFDFH are known as the external SFR area. Registers in external circuitry interfaced and mapped to these addresses can be addressed with sfr addressing. Table 2 is a list of the special function registers.

Figure 2. General Registers

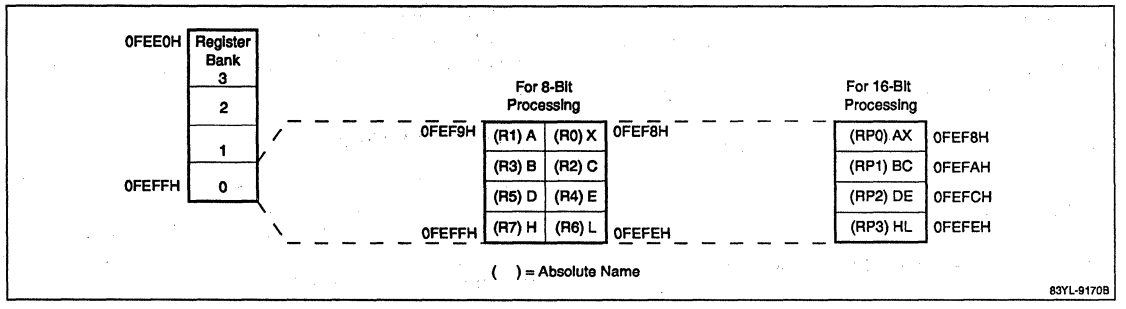


Table 2. Special Function Registers

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
0FF00H	Port 0	P0	R/W	x	x	—	Undefined
0FF02H	Port 2	P2	R	x	x	—	Undefined
0FF03H	Port 3	P3	R/W	x	x	—	Undefined
0FF04H	Port 4	P4	R/W	x	x	—	Undefined
0FF05H	Port 5	P5	R/W	x	x	—	Undefined
0FF06H	Port 6	P6	R/W	x	x	—	x0H
0FF07H	Port 7	P7	R	x	x	—	Undefined
0FF0AH	Port 0 buffer register (low)	P0L	R/W	x	x	—	Undefined
0FF0BH	Port 0 buffer register (high)	P0H	R/W	x	x	—	Undefined
0FF0CH	Real-time output port control register	RTPC	R/W	x	x	—	00H
0FF10H-0FF11H	16-bit compare register 0 (16-bit timer 0)	CR00	R/W	—	—	x	Undefined
0FF12H-0FF13H	16-bit compare register (16-bit timer 0)	CR01	R/W	—	—	x	Undefined
0FF14H	8-bit compare register (8-bit timer 1)	CR10	R/W	—	x	—	Undefined
0FF15H	8-bit compare register (8-bit timer/counter 2)	CR20	R/W	—	x	—	Undefined
0FF16H	8-bit compare register (8-bit timer/counter 2)	CR21	R/W	—	x	—	Undefined
0FF17H	8-bit compare register (8-bit timer 3)	CR30	R/W	—	x	—	Undefined
0FF18H-0FF19H	16-bit capture register (16-bit timer 0)	CR02	R	—	—	x	Undefined
0FF1AH	8-bit capture register (8-bit timer/counter 2)	CR22	R	—	x	—	Undefined
0FF1CH	8-bit capture/compare register (8-bit timer 1)	CR11	R/W	—	x	—	Undefined
0FF20H	Port 0 mode register	PM0	W	—	x	—	FFH
0FF23H	Port 3 mode register	PM3	W	—	x	—	FFH
0FF25H	Port 5 mode register	PM5	W	—	x	—	FFH
0FF26H	Port 6 mode register	PM6	R/W	x	x	—	FxH
0FF30H	Capture/compare control register 0	CRC0	W	—	x	—	10H
0FF31H	Timer output control register	TOC	W	—	x	—	00H
0FF32H	Capture/compare control register 1	CRC1	W	—	x	—	00H
0FF34H	Capture/compare control register 2	CRC2	W	—	x	—	00H
0FF40H	Pullup resistor option register	PUO	R/W	x	x	—	00H
0FF43H	Port 3 mode control register	PMC3	R/W	x	x	—	00H
0FF50H-0FF51H	16-bit timer register 0	TM0	R	—	—	x	0000H
0FF52H	8-bit timer register 1	TM1	R	—	x	—	00H
0FF54H	8-bit timer register 2	TM2	R	—	x	—	00H
0FF56H	8-bit timer register 3	TM3	R	—	x	—	00H
0FF5CH	Prescaler mode register 0	PRM0	W	—	x	—	00H
0FF5DH	Timer control register 0	TMC0	R/W	—	x	—	00H
0FF5EH	Prescaler mode register 1	PRM1	W	—	x	—	00H
0FF5FH	Timer control register 1	TMC1	R/W	—	x	—	00H
0FF68H	A/D converter mode register	ADM	R/W	x	x	—	00H
0FF6AH	A/D conversion result register	ADCR	R	—	x	—	Undefined
0FF7DH	One-shot pulse output control register	OSPC	R/W	x	x	—	00H

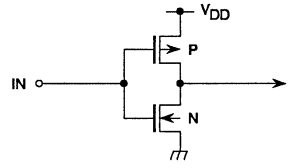
4b

Table 2. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
0FF80H	Clocked serial interface mode register	CSIM	R/W	x	x	—	00H
0FF82H	Serial bus interface control register	SBIC	R/W	x	x	—	00H
0FF86H	Serial shift register	SIO	R/W	—	x	—	Undefined
0FF88H	Asynchronous serial interface mode register	ASIM	R/W	x	x	—	80H
0FF8AH	Asynchronous serial interface status register	ASIS	R	x	x	—	00H
0FF8CH	Serial receive buffer: UART	RxB	R	—	x	—	Undefined
0FF8EH	Serial transmit shift register: UART	TxS	W	—	x	—	Undefined
0FF90H	Baud rate generator control register	BRGC	W	—	x	—	00H
0FFC0H	Standby control register	STBC	R/W	—	x	—	0000x000B
0FFC4H	Memory expansion mode register	MM	R/W	x	x	—	20H
0FFC5H	Programmable wait control register	PW	R/W	x	x	—	80H
0FFC6H	Refresh mode register	RFM	R/W	x	x	—	00H
0FFD0H-0FFDFH	External SFR area	—	R/W	x	x	—	Undefined
0FFE0H	Interrupt request flag register L	IF0L	R/W	x	x	—	00H
0FFE1H	Interrupt request flag register H	IF0H	R/W	x	x	—	00H
0FFE0H-0FFE1H	Interrupt request flag register	IF0	R/W	—	—	x	000H
0FFE4H	Interrupt mask flag register L	MK0L	R/W	x	x	—	FFH
0FFE5H	Interrupt mask flag register H	MK0H	R/W	x	x	—	FFH
0FFE4H-0FFE5H	Interrupt mask flag register	MK0	R/W	—	—	x	FFFH
0FFE8H	Priority specification flag register L	PR0L	R/W	x	x	—	FFH
0FFE9H	Priority specification flag register H	PR0H	R/W	x	x	—	FFH
0FFE8H-0FFE9H	Priority specification flag register	PR0	R/W	—	—	x	FFFH
0FFECH	Interrupt service mode specification flag register L	ISM0L	R/W	x	x	—	00H
0FFEDH	Interrupt service mode specification flag register H	ISM0H	R/W	x	x	—	00H
0FFECH-0FFEDH	Interrupt service mode specification flag register	ISM0	R/W	—	—	x	000H
0FFF4H	External interrupt mode register 0	INTM0	R/W	x	x	—	00H
0FFF5H	External interrupt mode register 1	INTM1	R/W	x	x	—	00H
0FFF8H	Interrupt status register	IST	R/W	x	x	—	00H

Figure 3. Pin I/O Circuits

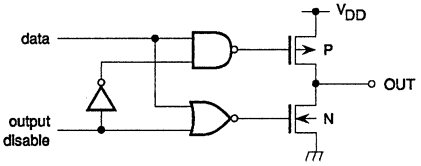
Type 1 ($\overline{EA}$)



Type 2 ($P2_0$, $P1_1$, $\overline{RESET}$)

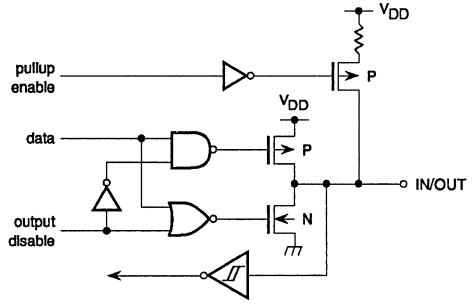
Schmitt trigger input with hysteresis characteristic.

Type 4 (P0, $P6_0$ - $P6_3$, ASTB)

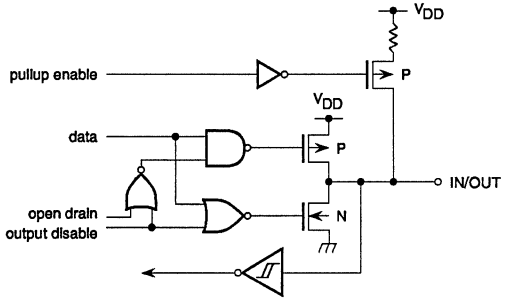


Push-pull output where the output can be placed in high-impedance (both P and N channels are turned off).

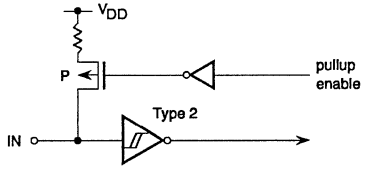
Type 8-A ($P3_2$)



Type 10-A ($P3_3$)

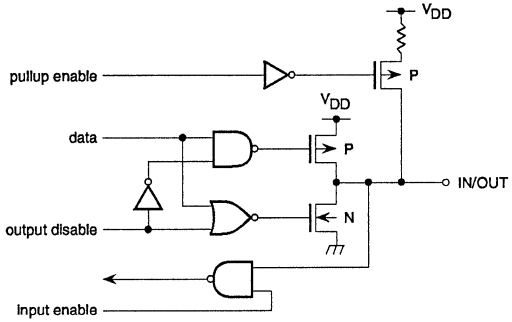


Type 2-A ($P2_2$ - $P2_7$)

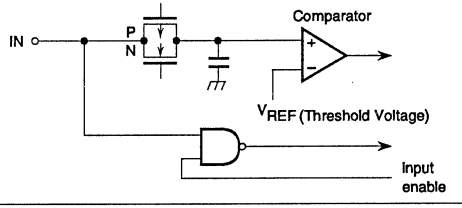


Schmitt trigger input with hysteresis characteristic.

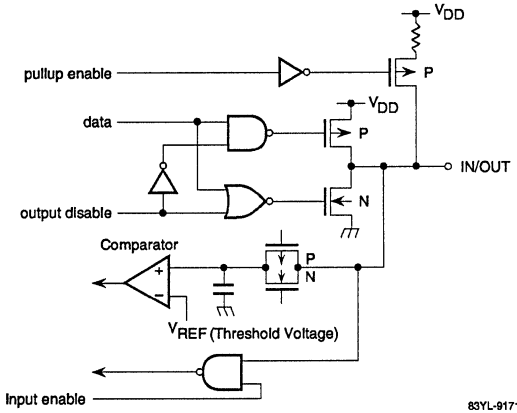
Type 5-A ($P3_0$, $P3_1$, $P3_4$ - $P3_7$, P4, P5, $P6_4$, $P6_5$)



Type 9 (P7)



Type 11 ($P6_6$ - $P6_7$)



83YL-9171B

4b

Input/Output Ports

There are up to 54 port lines on the μPD78218A/P218A and up to 36 port lines on the μPD78217A. (Ports 4, 5, and two bits of port 6 are not available on the μPD78217A since the μPD78217A must always use external memory.) Table 3 lists the features of each port and figure 3 shows the structure of each port pin. The pin levels of all port 2, 3, and 7 pins can always be read or tested regardless of the dual pin function.

Real-time Output Port

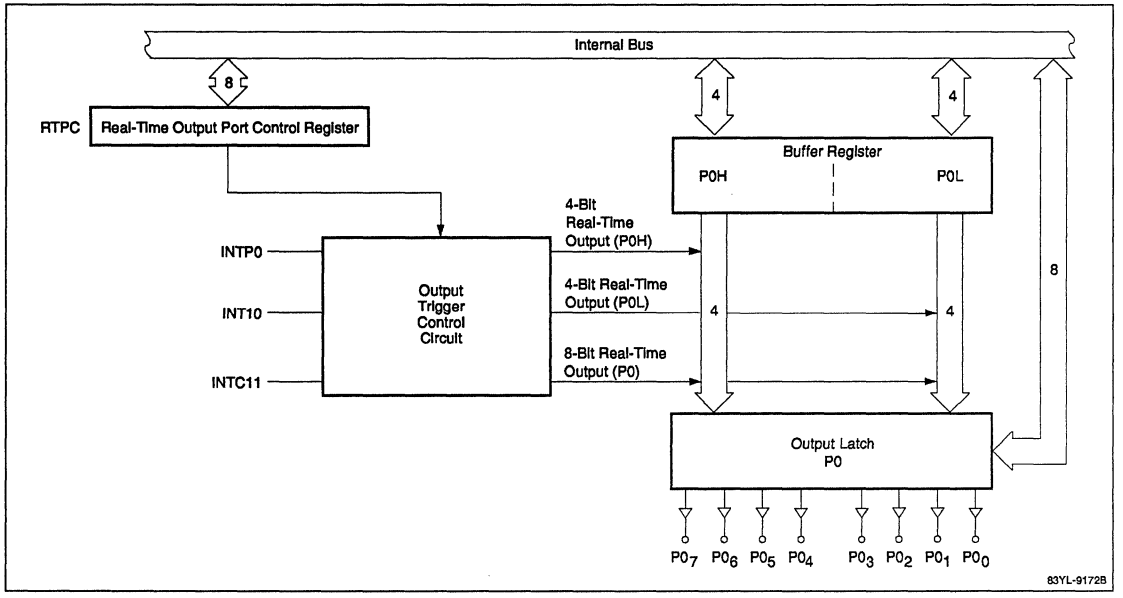
The real-time output port (RTPC) shares pins with port 0. It can be used as two independent 4-bit real-time output ports or one 8-bit real-time output port. In the real-time output mode, data stored beforehand in the buffer registers, P0H and P0L, is transferred immediately to the output latch of P0 on the occurrence of a timer 1 interrupt (INTC10 or INTC11) or external interrupt (INTP0) (see figure 4). By using the real-time output port with the macro service function, port 0 can be used to output preprogrammed patterns at preprogrammed variable time intervals. In this mode, two independent stepper motors can accurately be driven at a fixed or variable rate.

Table 3. Digital Port Functions

Port	Operational Features	Configuration	Direct Drive Capability	Software Pullup Resistor Connection
Port 0	8-bit high impedance output		Transistor	
Port 2	8-bit Schmitt trigger input			In 6-bit units ($P2_2$ - $P2_7$)
Port 3	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 4	8-bit input or output	Byte selectable	LED	Byte selectable
Port 5	8-bit input or output	Byte selectable	LED	Byte selectable, input bits only
Port 6	4-bit output (bits 0 to 3) 4-bit input or output (bits 4 to 7)	Bit selectable		In 4-bit unit, input bits only
Port 7	6-bit input			

Note: Software pullup resistors can be internally connected only on a port-by-port basis to port bits set to input mode. Pullup resistors are not connected to port bits set to output mode.

Figure 4. Real-Time Output Port



4b

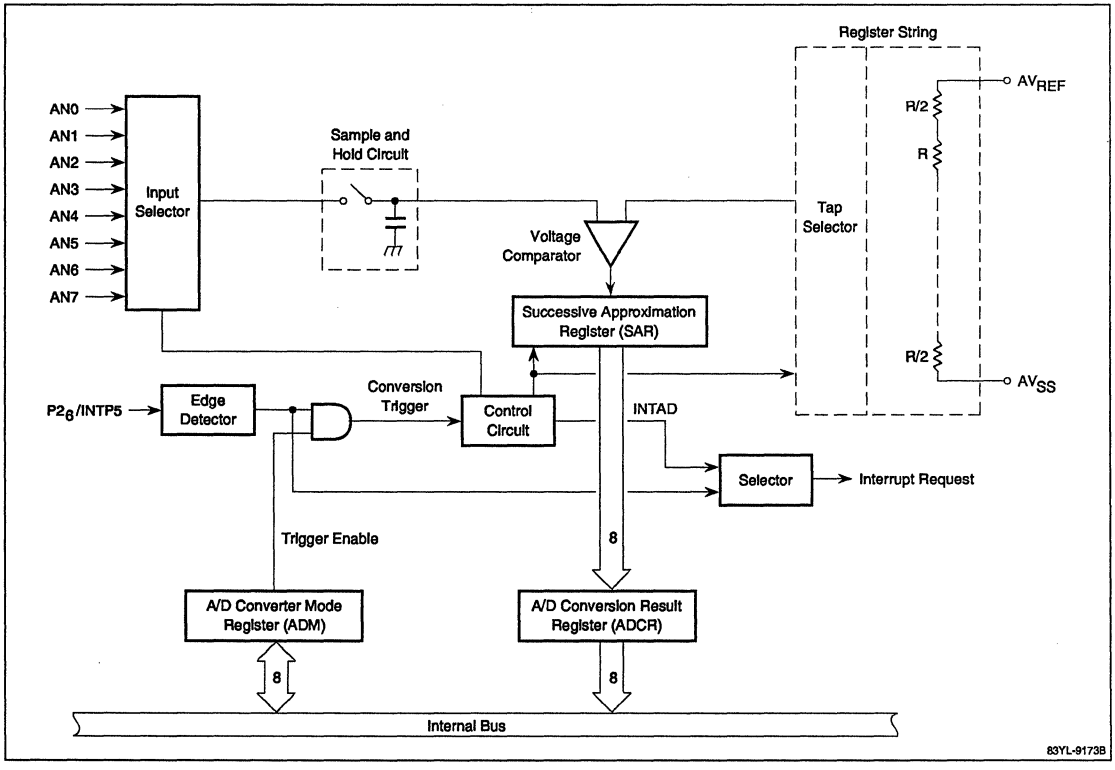
Analog-to-Digital (A/D) Converter

The μPD78218A family A/D converter (see figure 5) uses the successive-approximation method for converting up to eight multiplexed analog inputs into 8-bit digital data. The conversion time per input is 30 μs at 12 MHz operation. A/D conversion can be started by an external interrupt, INTP5, or under software control.

The A/D converter can operate in either scan mode or select mode. In scan mode, from one to eight sequential inputs can be programmed for conversion. The A/D converter selects each input in order, converts the data, stores it in the A/D conversion result (ADCR) register, and generates an interrupt (INTAD). This converted data can be easily transferred to memory by using the macro service function.

In select mode, only one of the eight A/D inputs can be selected for conversion. The ADCR register is continually updated and can be read at any time. If the A/D converter is started by an external interrupt, an INTAD interrupt occurs at the completion of each conversion. If the A/D converter is started by software, no interrupts are generated.

Figure 5. A/D Converter



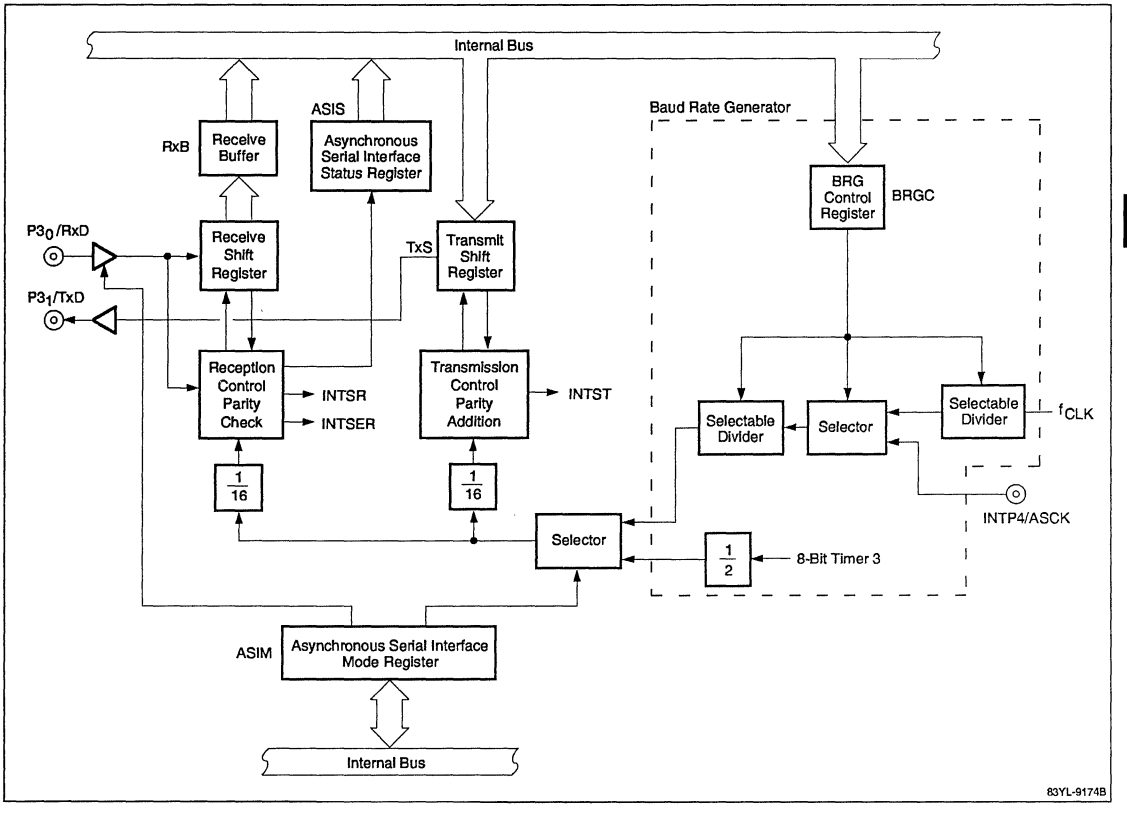
Serial Interface

The μPD78218A family has two independent serial interfaces. The first is a standard UART. The UART (figure 6) permits full-duplex operation and can be programmed for 7- or 8-bits of data after the start bit, followed by one or two stop bits. Odd, even, zero or no parity can also be selected. The serial clock for the UART can be provided by an on-chip baud rate generator or timer 3. By using either the internal system clock or an external clock input into the ASCK pin, the baud rate generator is capable of generating all of the commonly used baud rates. The UART generates three interrupts: INTST (transmission complete), INTSR (reception complete), and INTSER (reception error).

The second interface is an 8-bit clock-synchronized serial interface (figure 7). It can be operated in either a three-wire serial I/O mode or NEC serial bus interface (SBI) mode.

In the three-wire serial I/O mode, the 8-bit shift register (SIO) is loaded with a byte of data and eight clock pulses are generated. These eight pulses shift the byte of data out of the SO line (MSB first) and in from the SI line providing full-duplex operation. This interface can also be set to receive or to transmit data only. The INTCSI interrupt is generated after each 8-bit transfer. One of three internal clocks or an external clock clocks the data.

Figure 6. Asynchronous Serial Interface



4b

Figure 7. Clock-Synchronized Serial Interface

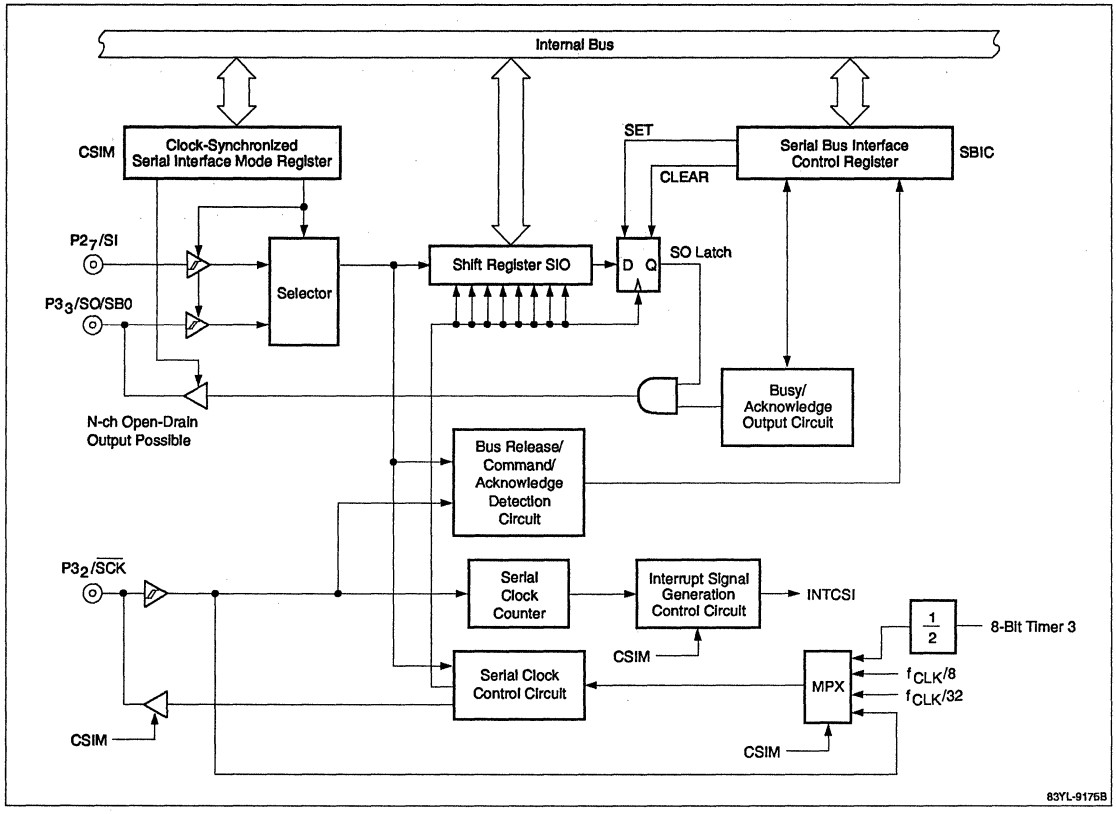
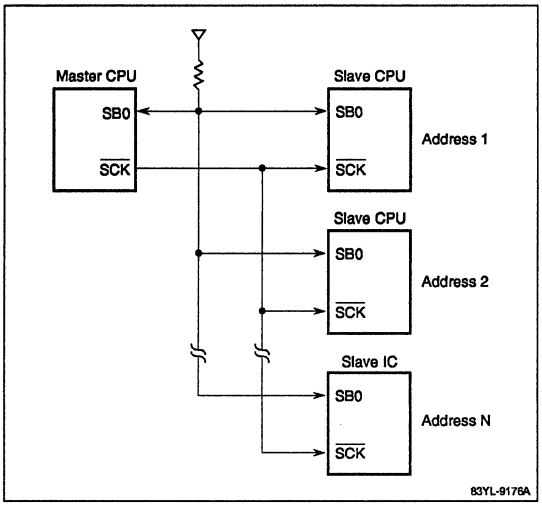


Figure 8. SBI Mode Master/Slave Configuration



The NEC SBI mode is a two-wire high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are connected in a master/slave configuration (see figure 8). There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over the serial bus line (SB0) using a fixed hardware protocol synchronized with the $\overline{SCK}$ line. Each slave μPD78218A family device can be programmed in software to respond to any one of 256 addresses. There are also 256 commands and 256 data types. Since all commands are user definable, any software protocol, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

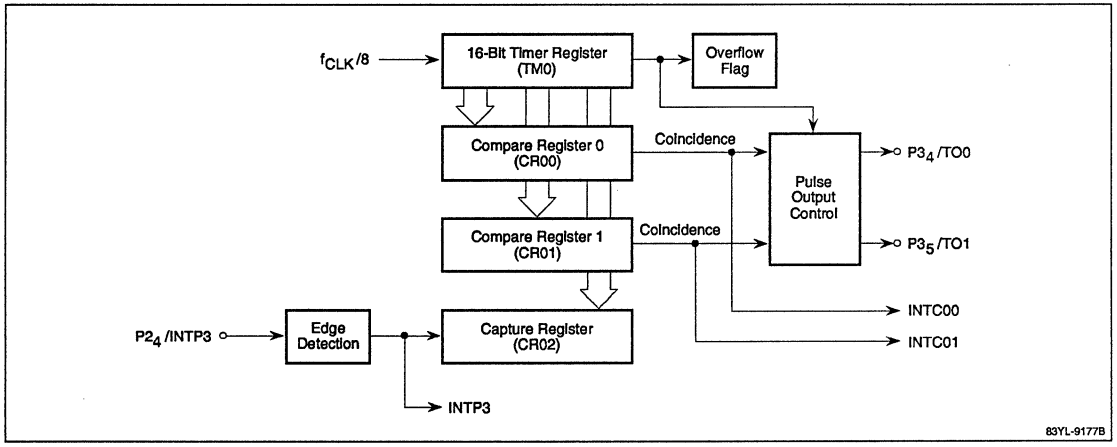
Timers

The μPD78218A family has one 16-bit timer and three 8-bit timers. The 16-bit timer counts the internal system clock ($f_{CLK}/8$) while the three 8-bit timers can be programmed to count a number of prescaled values of the internal system clock. One of the 8-bit timers can also count external events.

Timer 0 consists of a 16-bit timer (TM0), two 16-bit compare registers (CR00 and CR01), and a 16-bit capture register (CR02). Timer 0 can be used as two interval timers, to output a programmable square wave or two pulse-width modulated signals, to measure pulse widths, or to generate a software-triggered one-shot output pulse (see figure 9).

4b

Figure 9. 16-Bit Timer 0



Timer 1 consists of an 8-bit timer (TM1), 8-bit compare register (CR10), and 8-bit capture/compare register (CR11). Timer 1 can be used as two interval timers or to measure pulse widths. In addition, it can be used to generate the output trigger for the real-time output port (see figure 10).

Timer/counter 2 consists of an 8-bit timer (TM2), two 8-bit compare registers (CR20 and CR21), and an 8-bit capture register (CR22). Timer/counter 2 can also be used as two interval timers, to output a programmable square wave or two pulse-width modulated signals, or to measure pulse widths. In addition, it can be used to count external events sensed on the CI line or as a one-shot timer (see figure 11).

Timer 3 consists of an 8-bit timer (TM3) and an 8-bit compare register (CR30). Timer 3 can be used as an interval timer or as a clock for the clock-synchronized serial interface (see figure 12).

Figure 10. 8-Bit Timer 1

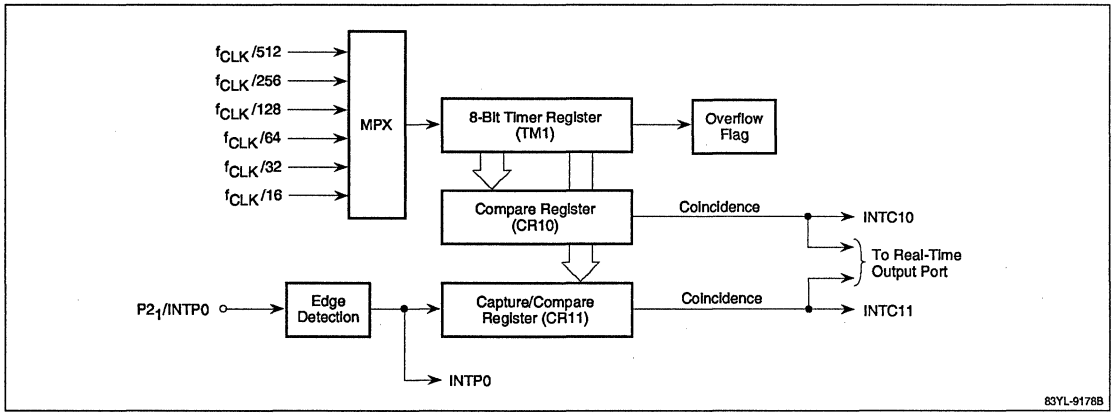


Figure 11. 8-Bit Timer/Counter 2

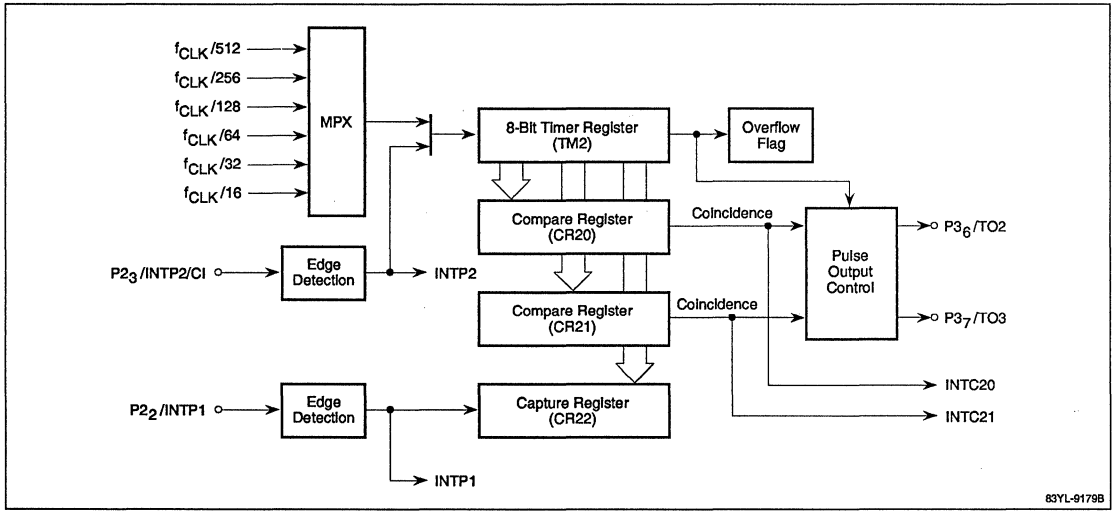
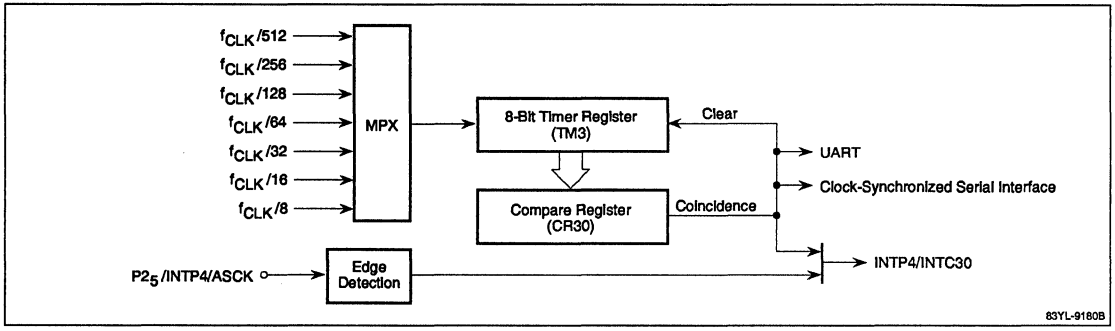


Figure 12. 8-Bit Timer 3



Interrupts

The μPD78218A family has 18 maskable hardware interrupt sources; 6 are external and 12 are internal. Since there are only 16 interrupt vectors and sets of control flags, 2 of the 6 external maskable interrupts, INTP4 and INTP5, share interrupt vectors and control flags with INTC30 and INTAD respectively. The active interrupt source for each shared vector must be chosen by the program. In addition, there is one nonmaskable interrupt and one software interrupt. The software interrupt, generated by the BRK instruction, is not maskable (see table 4).

Table 4. Interrupt Sources and Vector Addresses

Interrupt Request Type	Default Priority	Interrupt Request Generation Source	Macro Service Type	Vector Table Address
Software	None	BRK instruction execution	—	003EH
Nonmaskable	None	NMI (pin input edge detection)	—	0002H
Maskable	0	INTP0 (pin input edge detection)	A, B	0006H
	1	INTP1 (pin input edge detection)	A, B	0008H
	2	INTP2 (pin input edge detection)	A, B	000AH
	3	INTP3 (pin input edge detection)	B	000CH
	4	INTC00 (TM0-CR00 coincidence signal generation)	B	0014H
	5	INTC01 (TM0-CR01 coincidence signal generation)	B	0016H
	6	INTC10 (TM1-CR10 coincidence signal generation)	A, B, C	0018H
	7	INTC11 (TM1-CR11 coincidence signal generation)	A, B, C	001AH
	8	INTC21 (TM2-CR21 coincidence signal generation)	A, B	001CH
	9	INTP4 (pin input edge detection)	B	000EH
		INTC30 (TM3-CR30 coincidence signal generation)	A, B	
	10	INTP5 (pin input edge detection)	B	0010H
		INTAD (end of A/D conversion)	A, B	
	11	INTC20 (TM2-CR20 coincidence signal generation)	A, B	0012H
	12	INTSER (generation of asynchronous serial interface receive error)	—	0020H
	13	INTSR (end of asynchronous serial interface reception)	A, B	0022H
	14	INTST (end of asynchronous serial interface transmission)	A, B	0024H
	15	INTCSI (end of clocked serial interface transmission)	A, B	0026H

Interrupt Servicing. The μPD78218A family provides two levels of programmable hardware priority control and two different methods of handling maskable interrupt requests: standard vectoring and macro service. The programmer can choose the priority and mode of servicing each maskable interrupt by using the interrupt control registers.

Interrupt Control Registers. The μPD78218A family has four 16-bit interrupt control registers. Each bit in each register is dedicated to one of the 16 active maskable interrupt sources. The interrupt request flag register (IF0) contains an interrupt request flag for each interrupt. The interrupt mask register (MK0) is used to enable or disable any interrupt. The interrupt service mode register (ISM0) specifies whether an interrupt is processed by vectoring or macro service. The priority flag register (PR0) can be used to specify a high or a low priority level for each interrupt.

Two other 8-bit registers are associated with interrupt processing. The interrupt status register (IST) indicates if a nonmaskable interrupt request on the NMI pin is being processed and can be used to allow nesting of nonmaskable interrupt requests. The IE and the ISP bits of the program status word are also used to control interrupts. If the IE bit is zero, all maskable interrupts, but not macro service, are disabled. The IE bit can be set or cleared using the EI and DI instructions, respectively, or by directly writing to the PSW. The IE bit is cleared each time an interrupt is accepted. The ISP bit is used by hardware to hold the priority level flag of the interrupt being serviced.

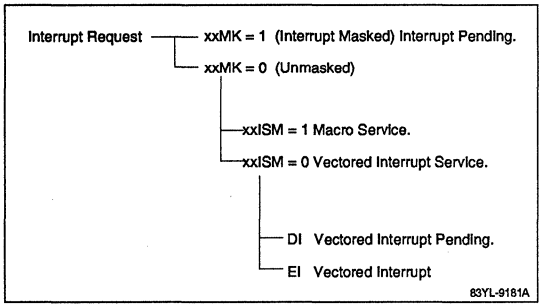
Interrupt Priority. The nonmaskable interrupt (NMI) has priority over all other interrupts. Two hardware controlled priority levels are available for the maskable interrupts. Either a high or a low priority level can be assigned by software to each of the maskable interrupts. Interrupt requests of a priority higher than the processor's current priority level are accepted; requests of the same or lower priority are held pending until the processor's priority state is lowered by program control within the current service routine or by a return instruction from the current service routine.

Interrupt requests programmed to be handled by macro service have priority over all vectored interrupt service regardless of the assigned priority level, and macro service requests are accepted even when the interrupt enable bit in the PSW is set to the disable state (see figure 13).

The default priorities listed in table 4 are fixed by hardware and are effective only when it is necessary to choose between two interrupt requests of the same software assigned priority. For example, the default priorities would be used after the completion of a high priority routine, if two interrupts of the same priority routine were pending.

The software interrupt, initiated by the BRK instruction, is executed regardless of the processor's priority level and the state of the IE bit. It does not alter the processor's priority level.

Figure 13. Interrupt Service Sequence



Vectored Interrupt. When vectored interrupt is specified for a given interrupt request, (1) the program status word and the program counter are saved on the stack, (2) the processor's priority is set to that specified for the interrupt, (3) the IE bit in the PSW is set to zero, and (4) the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (RETB instruction for the software interrupt) reverses the process and the μPD78218A family device resumes the interrupted routine.

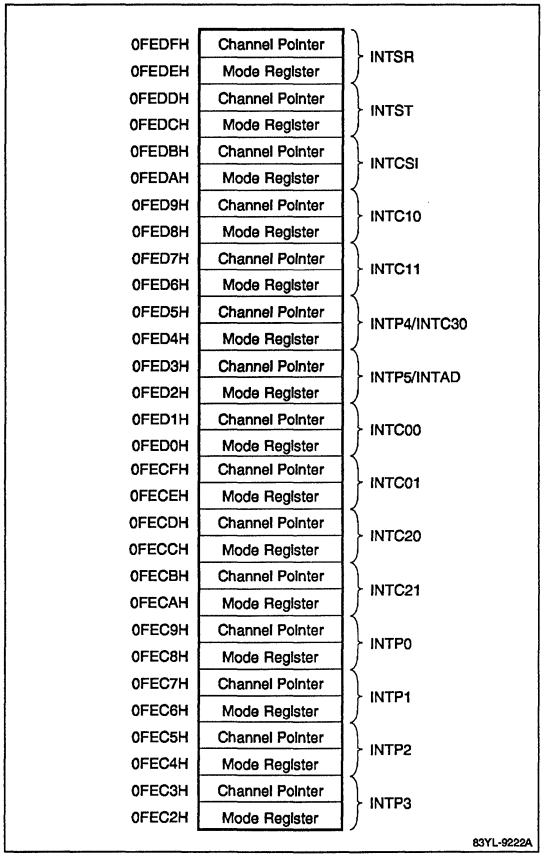
Macro Service

When macro service is specified for a given interrupt, the macro service hardware temporarily stops the executing program and begins to transfer data between the special function register area and the memory space. One byte is transferred each interrupt. When the data transfer is complete, control is returned to the executing program, providing a completely transparent method of interrupt service. Macro service significantly improves response time and makes it unnecessary to save any registers.

For each request on the interrupt line, one operation is performed, and an 8- or 16-bit counter is decremented. When the counter reaches zero, a vectored interrupt service routine is entered according to the specified priority.

Macro service is provided for all of the maskable interrupt requests except INTSER, the asynchronous serial interface receive error interrupt request. Each interrupt request has a dedicated macro service control word stored in Internal RAM (see figure 14). The function to be performed is specified in the control word.

Figure 14. Macro Service Control Word Map



The μPD78218A family provides three different types of macro service transfers:

Macro Service Type A. A byte of data is transferred in either direction between a special function register, preassigned for each interrupt request, and a buffer in Internal RAM (FExx). Only the 8-bit macro service counter is available for Type A transfers. The preassigned SFRs for the 12 interrupt requests that support macro service Type A transfers are listed in table 5.

Table 5. Macro Service Type A Interrupts and Assigned SFRs

Interrupt Request	Source/Destination SFR
INTC10: TM1-CR10 coincidence	CR10: Timer 1 8-bit compare register
INTC11: TM1-CR11 coincidence	CR11: Timer 1 8-bit capture/ compare register
INTC20: TM2-CR20 coincidence	CR20: Timer 2 8-bit compare register
INTC21: TM2-CR21 coincidence	CR21: Timer 2 8-bit compare register
INTC30: TM3-CR30 coincidence	CR30: Timer 3 8-bit compare register
INTSR: End of asynchronous serial interface reception	RxB: Serial receive buffer
INTST: End of asynchronous serial interface transmission	TxS: Serial transmit shift register
INTCSI: End of clocked serial interface transmission	SIO: Serial shift register
INTAD: End of A/D conversion	ADCR: A/D conversion result register
INTP0: External interrupt pin $P0_1$	CR11: Timer 1 8-bit capture/ compare register
INTP1: External interrupt pin $P0_2$	CR22: Timer 2 8-bit capture register
INTP2: External interrupt pin $P0_3$	TM2: Timer 2 8-bit timer register

Macro Service Type B. A byte of data is transferred in either direction between any specified special function register and a buffer anywhere in the 64K byte address space. The macro service counter can be programmed either to be an 8- or 16-bit counter. Macro service Type B transfers can be initiated by any maskable interrupt except INTSER.

Macro Service Type C. A byte of data is transferred from a buffer anywhere in the 64K byte address space to one of the 8-bit compare registers of timer 1. At the same time, a second byte of data is transferred from a buffer anywhere in the 64K byte address space to the real-time output port buffer. The macro service counter can be programmed either to be an 8- or 16-bit counter. Macro service Type C transfers can be initiated by INTC10 with data transferred to CR10 and P0L or P0H, or by INTC11 with data transferred to CR11 and P0L or P0H.

In addition, the macro service Type C transfer can be initialized to automatically alter timer compare register values or to repeatedly output a prespecified pattern at a fixed or variable rate. By using macro service Type C transfers to control the real-time output ports,

the μPD78218A family can easily and accurately drive two independent stepper motors.

Refresh

The refresh signal is used with any pseudostatic RAM equivalent of the NEC μPD428128. The refresh cycle can be set to one of four intervals: 16, 32, 64, or $128/f_{CLK}$ (2.6, 5.3, 10.7, and 21.3 μs at 12 MHz). The refresh cycle is timed to follow a read or write operation to avoid interference with external memory access cycles.

Standby Modes

HALT and STOP modes are provided to reduce power consumption when CPU action is not required. In HALT mode, the CPU is stopped but the system clock continues to run. The HALT mode is released by any unmasked interrupt, an external NMI, or an external reset pulse. In STOP mode, both the CPU and the system clock are stopped, further minimizing the power consumption. The STOP mode is released by either an external reset pulse or an external NMI. The HALT and STOP modes are entered by programming the standby control register (STBC). This register is a protected location and can be written to only by a special instruction. If the third and fourth bytes of the instruction are not complements of each other, the data is not written and the next instruction is executed.

External Reset

The μPD78218A family is reset by taking the $\overline{\text{RESET}}$ pin low. The $\overline{\text{RESET}}$ input pin contains a noise filter to protect against spurious system resets caused by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low until the power supply reaches its operating voltage and the oscillator has stabilized. During reset the program counter is loaded with the address contained in the reset vector table (address 0000H, 0001H); program execution starts at that address upon the $\overline{\text{RESET}}$ pin going high. While $\overline{\text{RESET}}$ is low, all external lines except V_{SS}, V_{DD}, AV_{SS}, AV_{REF}, X1, and X2 are in the high impedance state.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = +25°C

Operating voltage, V_{DD}	−0.5 to +7.0 V
AV_{REF}	−0.5 to V_{DD} + 0.5 V
AV_{SS}	−0.5 to + 0.5 V
Input voltage, V_{I1}	−0.5 to V_{DD} + 0.5 V
V_{I2} (Note 1)	−0.5 to AV_{REF} + 0.5 V
V_{I3} (Note 2 for μPD78P218A)	−0.5 to +13.5 V
Output voltage, V_O	−0.5 to V_{DD} + 0.5 V
Low-level output current, I_{OL}	
per pin	15 mA
total, all output pins	100 mA
High-level output current, I_{OH}	
per pin	−10 mA
total, all output pins	−50 mA
Operating temperature, T_{OPT}	−40 to +85°C
Storage temperature, T_{STG}	−65 to +150°C

Notes:

(1) Pins $P7_0$/AN0 - $P7_5$/AN5, $P6_6/\overline{WAIT}$/AN6, and $P6_7/\overline{REFRQ}$/AN7 when used as the A/D converter input pins. However, the absolute maximum rating of V_{I1} must also be satisfied.

(2) $P2_0$/NMI, $\overline{EA}/V_{PP}$, and $P2_1$/INTP0/A_9 pins in the PROM programming mode

Exposure to absolute maximum ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC characteristics.

Operating Conditions

Oscillation Frequency, f_{XX}	T_A	V_{DD}
4 to 12 MHz	−40 to +85°C	+5 V ±10% (μPD78217A/218A); +5 V ±0.3 V (μPD78P218A)

Capacitance

T_A = +25°C; V_{DD} = V_{SS} = 0 V

Item	Symbol	Max	Unit	Conditions
Input capacitance	C_I	20	pF	f = 1 MHz; pins not used for measurement are at 0 V
Output capacitance	C_O	20	pF	
Input/output capacitance	C_{IO}	20	pF	

DC Characteristics

T_A = −40 to +85°C; V_{SS} = 0 V; V_{DD} = +5 V ±10% (V_{DD} = +5 V ±0.3 V for μPD78P218A)

Item	Symbol	Min	Typ	Max	Unit	Conditions
Low-level input voltage	V_{IL}	0		0.8	V	
High-level input voltage	V_{IH1}	2.2		V_{DD}	V	Except the specified pins (Notes 1, 2)
	V_{IH2}	2.2		AV_{REF}	V	Specified pins (Note 1)
	V_{IH3}	0.8 V_{DD}		V_{DD}	V	Specified pins (Note 2)
Low-level output voltage	V_{OL1}			0.45	V	I_{OL} = 2.0 mA
	V_{OL2}			1.0	V	I_{OL} = 8.0 mA (Note 3)
High-level output voltage	V_{OH1}	V_{DD} − 1.0			V	I_{OH} = −1.0 mA
	V_{OH2}	V_{DD} − 0.5			V	I_{OH} = −100 μA
	V_{OH3}	2.0			V	I_{OH} = −5.0 mA (Note 4)
X1 low-level input current	I_{IL}			−100	μA	0V ≤ V_I ≤ V_{IL}
X1 high-level input current	I_{IH}			100	μA	V_{IH3} ≤ V_I ≤ V_{DD}
Input leakage current	I_{LI}			±10	μA	0V ≤ V_I ≤ V_{DD}
Output leakage current	I_{LO}			±10	μA	0V ≤ V_O ≤ V_{DD}
AV_{REF} current	AI_{REF}		1.5	5.0	mA	Operating mode, f_{XX} = 12 MHz
V_{DD} power supply current	I_{DD1}		20	40	mA	Operating mode, f_{XX} = 12 MHz
	I_{DD2}		7	20	mA	HALT mode, f_{XX} = 12 MHz

4b

DC Characteristics (cont)

Item	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.5		5.5	V	STOP mode
Data retention current	I_{DDDR}		2	20	μA	STOP mode; V_{DDDR} = 2.5 V
			5	50	μA	STOP mode; V_{DDDR} = 5 V ±10%
Pullup resistor	R_L	15	40	80	kΩ	V_I = 0 V

Notes:

(1) Pins $P7_0$/AN0 - $P7_5$/AN5, $P6_6$/$\overline{WAIT}$/AN6, and $P6_7$/$\overline{REFRQ}$/AN7 when the pin is used as an A/D converter input pin.

(2) X1, X2, $\overline{RESET}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1,$P2_3$/$\overline{INTP2}$/CI, $P2_4$/INTP3, $P2_5$/$\overline{INTP4}$/ASCK, $P2_6$/INTP5, $P2_7$/SI, $P3_2$/SCK, $P3_3$/SO/SB0, and EA pins.

(3) Pins $P4_0$/AD_0 - $P4_7$/AD_7, and $P5_0$/A_8 - $P5_7$/A_{15}.

(4) Pins $P0_0$ - $P0_7$.

AC Characteristics—Read/Write Operation

T_A = −40 to +85°C; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF; V_{DD} = +5 V ±10% (V_{DD} = +5 V ±0.3 V for μPD78P218A)

Item	Symbol	Calculation Formula (2, 3)	Min	Max	Unit	Conditions
X1 input clock cycle time	t_{CYX}	—	82	250	ns	
Address setup time to ASTB ↓	t_{SAST}	t_{CYX} − 30	52		ns	
Address hold time from ASTB ↓ (Note 1)	t_{HSTA}	—	25		ns	
Address hold time from $\overline{RD}$↑	t_{HRA}	—	30		ns	
Address hold time from $\overline{WR}$ ↑	t_{HWA}	—	30		ns	
Address to $\overline{RD}$ ↓ delay time	t_{DAR}	$2t_{CYX}$ − 35	129		ns	
Address float time to $\overline{RD}$ ↓	t_{FAR}	$t_{CYX}/2$ − 30	11		ns	
Address to data input time	t_{DAID}	(4+2n) t_{CYX} − 100		228	ns	No wait states
ASTB ↓ to data input time	t_{DSTID}	(3+2n)t_{CYX} − 65		181	ns	No wait states
$\overline{RD}$ ↓ to data input time	t_{DRID}	(2+2n)t_{CYX} − 64		100	ns	No wait states
ASTB ↓ to $\overline{RD}$ ↓ delay time	t_{DSTR}	t_{CYX} − 30	52		ns	
Data hold time from $\overline{RD}$ ↑	t_{HRID}	—	0		ns	
$\overline{RD}$ ↑ to address active time	t_{DRA}	$2t_{CYX}$ − 40	124		ns	
$\overline{RD}$ ↑ to ASTB ↑ delay time	t_{DRST}	$2t_{CYX}$ − 40	124		ns	
$\overline{RD}$ low-level width	t_{WRL}	(2+2n)t_{CYX} − 40	124		ns	No wait states
ASTB high-level width	t_{WSTH}	t_{CYX} − 30	52		ns	
Address to $\overline{WR}$ ↓ delay time	t_{DAW}	$2t_{CYX}$ − 35	129		ns	
$\overline{ASTB}$ ↓ to data output time	t_{DSTOD}	t_{CYX} + 60		142	ns	
$\overline{WR}$ ↓ to data output time	t_{DWOD}	—		60	ns	
ASTB ↓ to $\overline{WR}$ ↓ delay time	t_{DSTW1}	t_{CYX} − 30	52		ns	
	t_{DSTW2}	$2t_{CYX}$ − 35	129		ns	Refresh mode
Data setup time to $\overline{WR}$ ↑	t_{SODWR}	(3+2n)t_{CYX} − 100	146		ns	No wait states
Data setup time to $\overline{WR}$ ↓	t_{SODWF}	t_{CYX} − 60	22		ns	Refresh mode
Data hold time from $\overline{WR}$ ↑ (Note 1)	t_{HWOD}	—	20		ns	
$\overline{WR}$ ↑ to ASTB ↑ delay time	t_{DWST}	t_{CYX} − 40	42		ns	
$\overline{WR}$ low-level width	t_{WWL1}	(3+2n)t_{CYX} − 50	196		ns	No wait states
	t_{WWL2}	(2+2n)t_{CYX} − 50	114		ns	Refresh mode; No wait states
Address to $\overline{WAIT}$ ↓ input time	t_{DAWT}	$3t_{CYX}$ − 100		146	ns	

AC Characteristics—Read/Write Operation (cont)

Item	Symbol	Calculation Formula (2, 3)	Min	Max	Unit	Conditions
ASTB ↓ to $\overline{WAIT}$ ↓ input time	t_{DSTWT}	$2t_{CYX} - 80$		84	ns	
$\overline{WAIT}$ hold time from ASTB ↓	t_{HSTWT}	$2Xt_{CYX} + 10$	174		ns	One external wait state
ASTB ↓ to $\overline{WAIT}$ ↑ delay time	t_{DSTWTH}	$2(1+X)t_{CYX} - 55$		273	ns	One external wait state
$\overline{RD}$ ↓ to $\overline{WAIT}$ ↓ input time	t_{DRWTL}	$t_{CYX} - 60$		22	ns	
$\overline{WAIT}$ hold time from $\overline{RD}$ ↓	t_{HRWT}	$(2X-1)t_{CYX} + 5$	87		ns	One external wait state
$\overline{RD}$ ↓ to $\overline{WAIT}$ ↑ delay time	t_{DRWTH}	$(2X+1)t_{CYX} - 60$		186	ns	One external wait state
$\overline{WAIT}$ ↑ to data input time	t_{DWTID}	$t_{CYX} - 20$		62	ns	
$\overline{WAIT}$ ↑ to $\overline{WR}$ ↑ delay time	t_{DWTW}	$2t_{CYX} - 10$	154		ns	
$\overline{WAIT}$ ↑ to $\overline{RD}$ ↑ delay time	t_{DWTR}	$t_{CYX} - 10$	72		ns	
$\overline{WR}$ ↓ to $\overline{WAIT}$ input time	t_{DWWTL}	$t_{CYX} - 60$		22	ns	Refresh disabled
$\overline{WAIT}$ hold time from $\overline{WR}$ ↓	t_{HWWT1}	$(2X-1)t_{CYX} + 5$	87		ns	One external wait state; refresh disabled
	t_{HWWT2}	$2(X-1)t_{CYX} + 5$	5		ns	One external wait state; refresh enabled
$\overline{WR}$ ↓ to $\overline{WAIT}$ ↑ delay time	t_{DWWTH1}	$(2X+1)t_{CYX} - 60$		186	ns	One external wait state; refresh disabled
	t_{DWWTH2}	$2Xt_{CYX} - 60$		104	ns	One external wait state; refresh enabled
$\overline{RD}$ ↑ to $\overline{REFRQ}$ ↓ delay time	t_{DRRFQ}	$2t_{CYX} - 10$	154		ns	
$\overline{WR}$ ↑ to $\overline{REFRQ}$ ↓ delay time	t_{DWRFQ}	$t_{CYX} - 10$	72		ns	
$\overline{REFRQ}$ low-level width	t_{WRFQL}	$2t_{CYX} - 44$	120		ns	
$\overline{REFRQ}$ ↑ to ASTB ↑ delay time	t_{DRFQST}	$4t_{CYX} - 48$	280		ns	

Notes:

(1) The hold time includes the time during which V_{OH} and V_{OL} are retained under the following load conditions: C_L = 100 pF and R_L = 2 kΩ

(2) n indicates the number of internal wait states.

(3) X indicates the number of external wait states (1, 2, 3, ...)

Serial Port Operation

T_A = −40 to +85°C; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF; V_{DD} = +5 V ±10% (V_{DD} = +5 V ±0.3 V for μPD78P218A)

Item	Symbol	Min	Max	Unit	Conditions
Serial clock cycle time	t_{CYSK}	1.0		μs	External clock input
		1.3		μs	Internal clock/16 output
		5.3		μs	Internal clock/64 output
Serial clock low-level width	t_{WSKL}	420		ns	External clock input
		556		ns	Internal clock/16 output
		2.5		μs	Internal clock/64 output
Serial clock high-level width	t_{WSKH}	420		ns	External clock input
		556		ns	Internal clock/16 output
		2.5		μs	Internal clock/64 output
SI, SB0 setup time to $\overline{SCK}$ ↑	t_{SSSK}	150		ns	
SI, SB0 hold time from $\overline{SCK}$ ↑	t_{HSSK}	400		ns	

Serial Port Operation (cont)

Item	Symbol	Min	Max	Unit	Conditions
SO/SB0 output delay time from $\overline{SCK}$ ↓	t_{DSBSK1}	0	300	ns	CMOS push-pull output (3-line serial I/O mode)
	t_{DSBSK2}	0	800	ns	Open-drain output (SBI mode), R_L = 1 kΩ
SB0 high, hold time from $\overline{SCK}$ ↑	t_{HSBSK}	4		t_{CYX}	SBI mode
SB0 low, setup time to $\overline{SCK}$ ↓	t_{SSBSK}	4		t_{CYX}	SBI mode
SB0 low-level width	t_{WSBL}	4		t_{CYX}	
SB0 high-level width	t_{WSBH}	4		t_{CYX}	

A/D Converter Operation

T_A = −40 to +85°C; V_{SS} = AV_{SS} = 0 V; V_{DD} = +5 V ±10% (V_{DD} = +5 V ±0.3 V for μPD78P218A)

Item	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		8			Bit	
Full-scale error *				0.4	%	AV_{REF} = 4.0 V to V_{DD}; T_A = −10 to +70°C
				0.8	%	AV_{REF} = 3.6 V to V_{DD}; T_A = −10 to +70°C
				0.8	%	AV_{REF} = 4.0 V to V_{DD}
Quantization error				±1/2	LSB	
Conversion time	t_{CONV}	360			t_{CYX}	82 ns ≤ t_{CYX} < 125 ns (FR bit of ADM register is 0)
		240			t_{CYX}	125 ns ≤ t_{CYX} < 250 ns (FR bit of ADM register is 1)
Sampling time	t_{SAMP}	72			t_{CYX}	82 ns ≤ t_{CYX} < 125 ns (FR bit of ADM register is 0)
		48			t_{CYX}	125 ns ≤ t_{CYX} < 250 ns (FR bit of ADM register is 1)
Analog input voltage	V_{IAN}	−0.3		AV_{REF} + 0.3	V	
Analog input impedance	R_{AN}		1000		MΩ	
Analog reference voltage	AV_{REF}	3.6		V_{DD}	V	
AV_{REF} current	AI_{REF}		1.5	5.0	mA	Operating mode, f_{XX} = 12 MHz
			0.2	1.5	mA	STOP mode

* Quantization errror is not included. Unit is defined as percent of full-scale value.

Interrupt Timing Operation

T_A = −40 to +85°C; V_{SS} = 0 V; V_{DD} = +5 V ±10% (V_{DD} = +5 V ±0.3 V for μPD78P218A)

Item	Symbol	Min	Max	Unit	Conditions
NMI low-level width	t_{WNIL}	10		μs	
NMI high-level width	t_{WNIH}	10		μs	
INTP0-INTP5 low-level width	t_{WITL}	24		t_{CYX}	
INTP0-INTP5 high-level width	t_{WITH}	24		t_{CYX}	
$\overline{RESET}$ low-level width	t_{WRSL}	10		μs	
$\overline{RESET}$ high-level width	t_{WRSH}	10		μs	

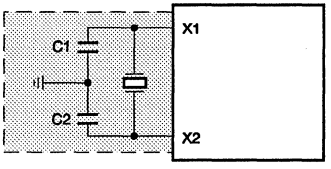
Data Retention Characteristics

T_A = −40 to +85°C

Item	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.5		5.5	V	STOP mode
Data retention current	I_{DDDR}		2	20	μA	V_{DDDR} = 2.5 V
			5	50	μA	V_{DDDR} = 5 V ±10%
V_{DD} rise time	t_{RVD}	200			μs	
V_{DD} fall time	t_{FVD}	200			μs	
V_{DD} retention time (from STOP mode setting)	t_{HVD}	0			ms	
STOP release signal input time	t_{DREL}	0			ms	
Oscillation stabilization wait time	t_{WAIT}	30			ms	Crystal resonator
		5			ms	Ceramic resonator
Low-level input voltage	V_{IL}	0		0.1 V_{DDDR}	V	Specified pins *
High-level input voltage	V_{IH}	0.9 V_{DDDR}		V_{DDDR}	V	Specified pins *

* $\overline{RESET}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2/CI, $P2_4$/INTP3, $P2_5$/INTP4/ASCK, $P2_6$//ASCK, $P2_6$/INTP5, $P2_7$/SI, $P3_2$/$\overline{SCK}$, $P3_3$/SO/SB0, and $\overline{EA}$ pins.

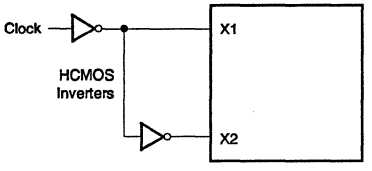
Recommended Resonator Circuit



Ceramic or crystal resonator frequency f_{XX} = 4 to 12 MHz
External oscillation circuit should be as close to the X1 and X2 pins as possible
Do not place other signal lines in the shaded area

83YL-9189A

Recommended External Clock Circuit



Clock frequency f_{XX} = 4 to 12 MHz
In STOP mode, X1 is internally shorted to V_{SS} to prevent leakage current. Therefore, STOP mode is not available when using this external clock circuit.

83YL-9190A

Recommended Ceramic Resonators (μPD78217A/218A only)

Manufacturer	Frequency (MHz)	Part Number	C1 (pF)	C2 (pF)
Murata Mfg.	12	CSA12.0MTZ	30	30
		CST12.0MTW	None*	None*

*C1 and C2 are contained in the resonator.

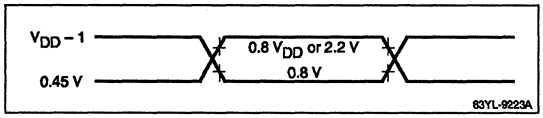
External Clock Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V

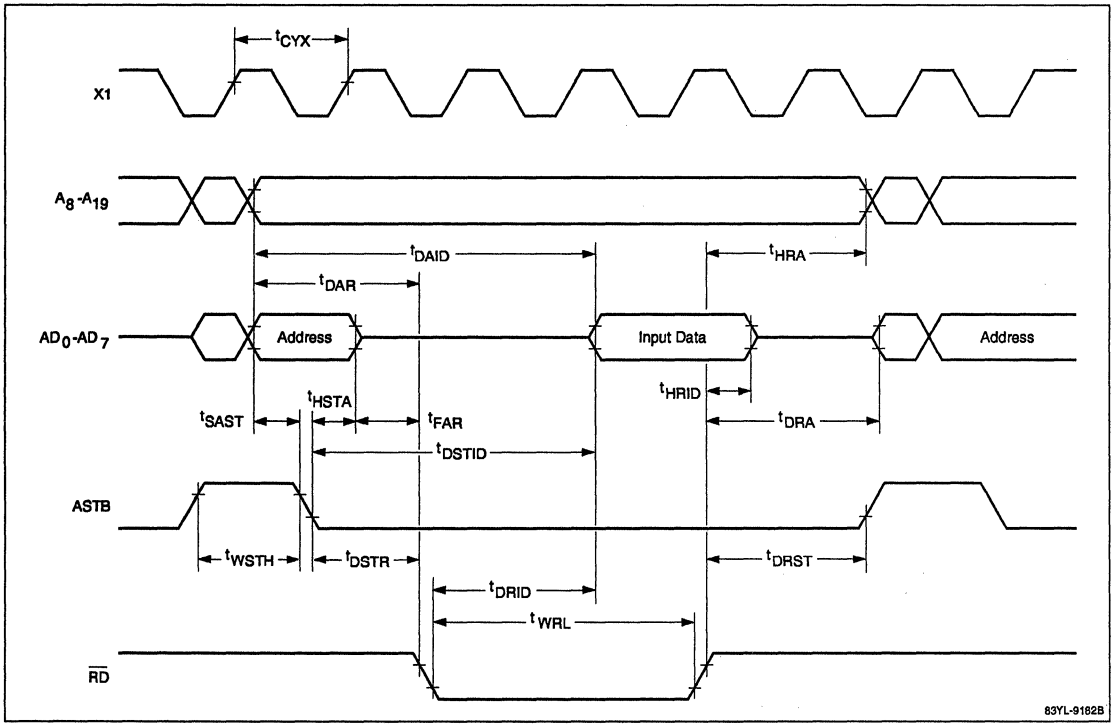
Item	Symbol	Min	Max	Unit	Conditions
X1 input low-level width	t_{WXL}	30	130	ns	
X1 input high-level width	t_{WXH}	30	130	ns	
X1 input rise time	t_{XR}	0	30	ns	
X1 input fall time	t_{XF}	0	30	ns	
X1 input clock cycle time	t_{CYX}	82	250	ns	

Timing Waveforms

Voltage Thresholds for AC Timing Measurements

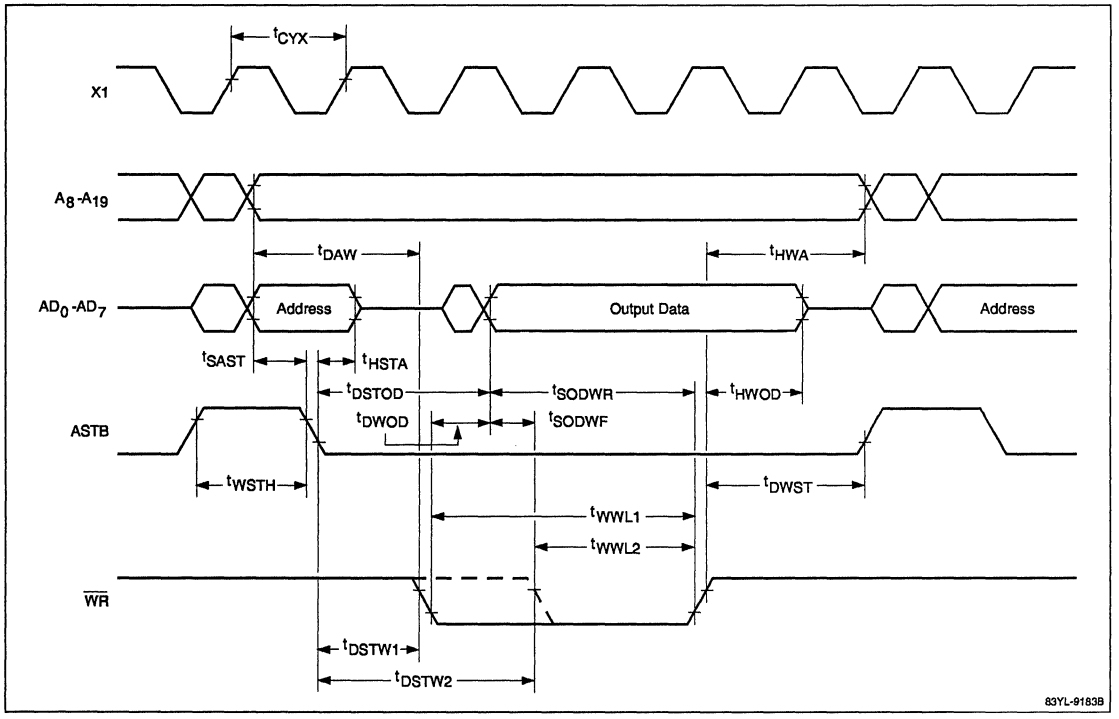


Read Operation



Timing Waveforms (cont)

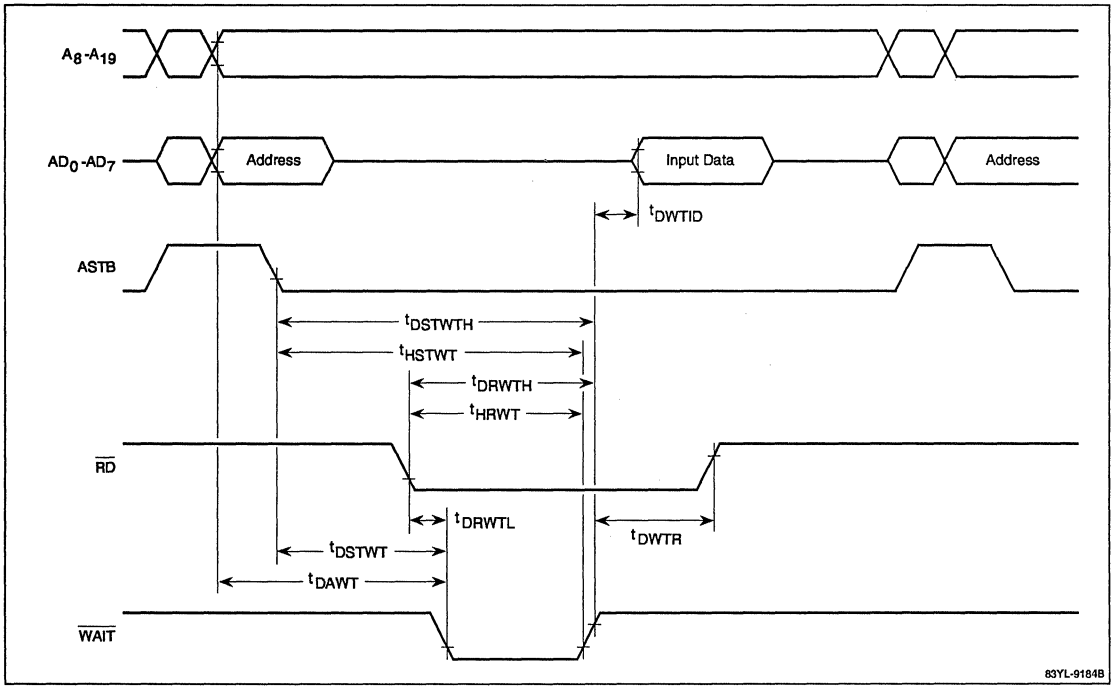
Write Operation



4b

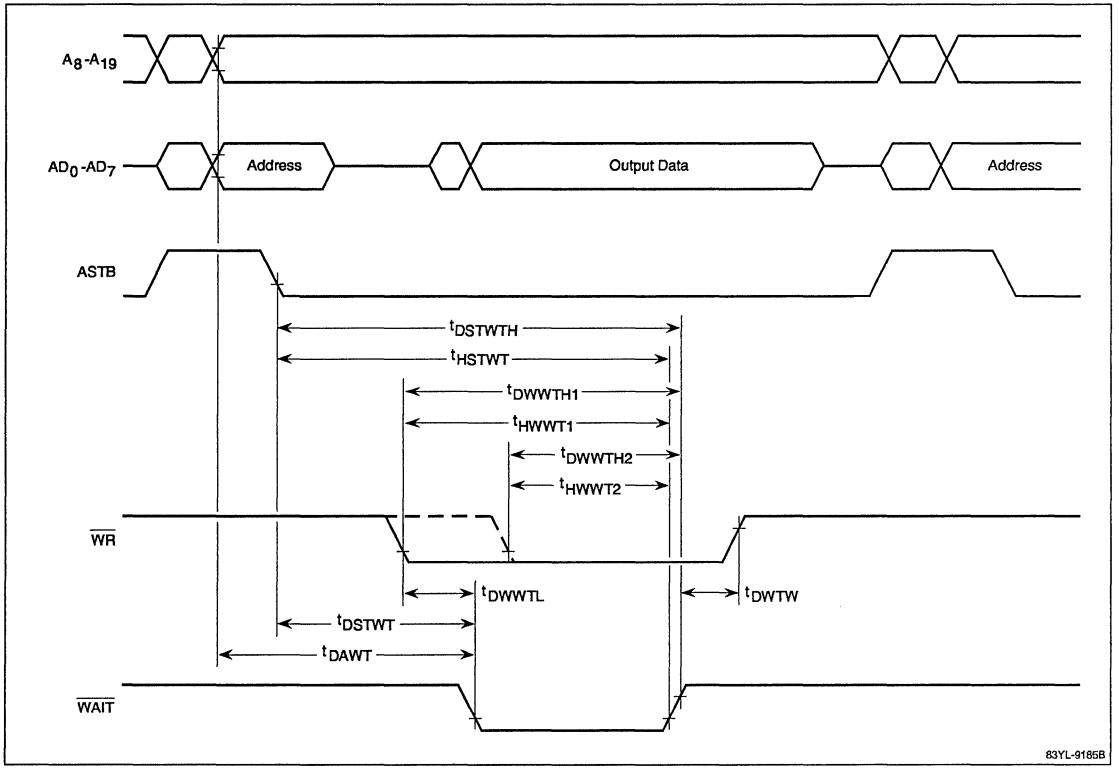
Timing Waveforms (cont)

External $\overline{WAIT}$ Signal Input (Read Operation)



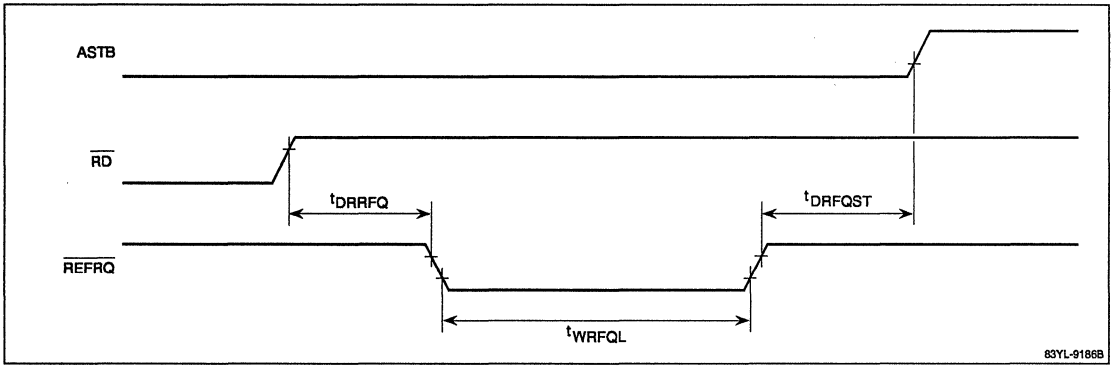
Timing Waveforms (cont)

External $\overline{WAIT}$ Signal Input (Write Operation)

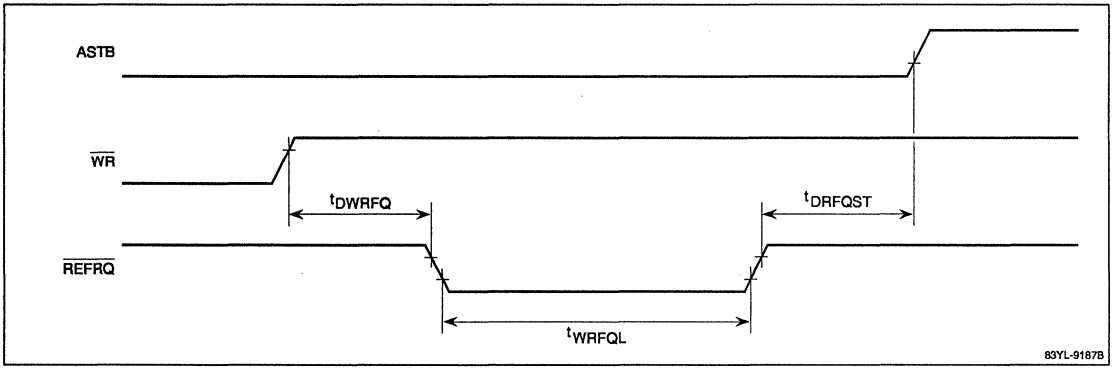


Timing Waveforms (cont)

Refresh After Read



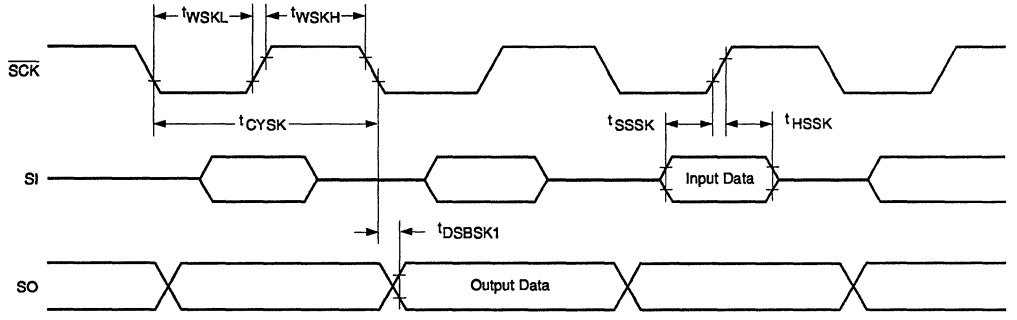
Refresh After Write



Timing Waveforms (cont)

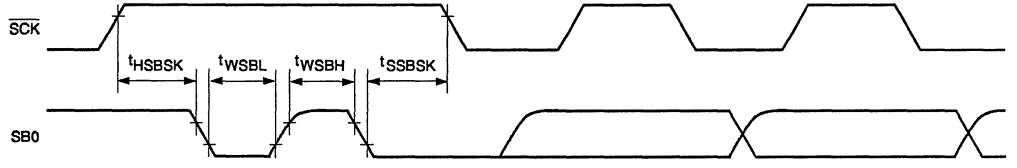
Serial Operation

Three-Line Serial I/O Timing

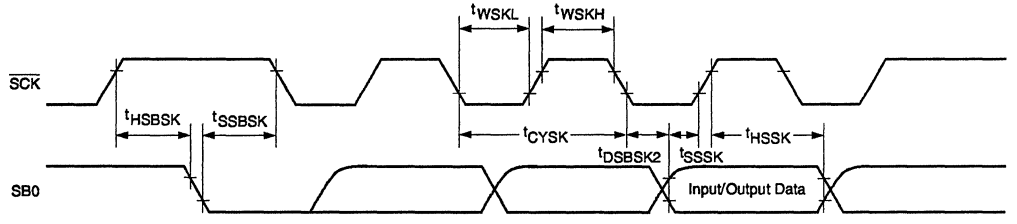


SBI Mode

Bus Release Signal Transfer Timing



Command Signal Transfer Timing

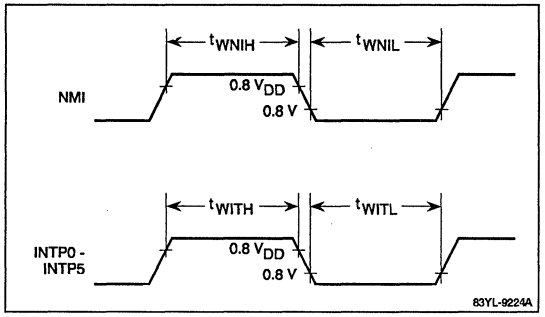


83YL-9188B

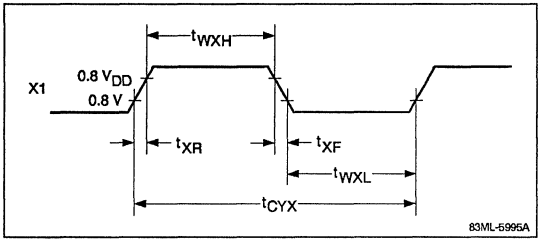
4b

Timing Waveforms (cont)

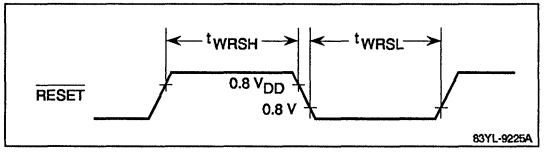
Interrupt Input



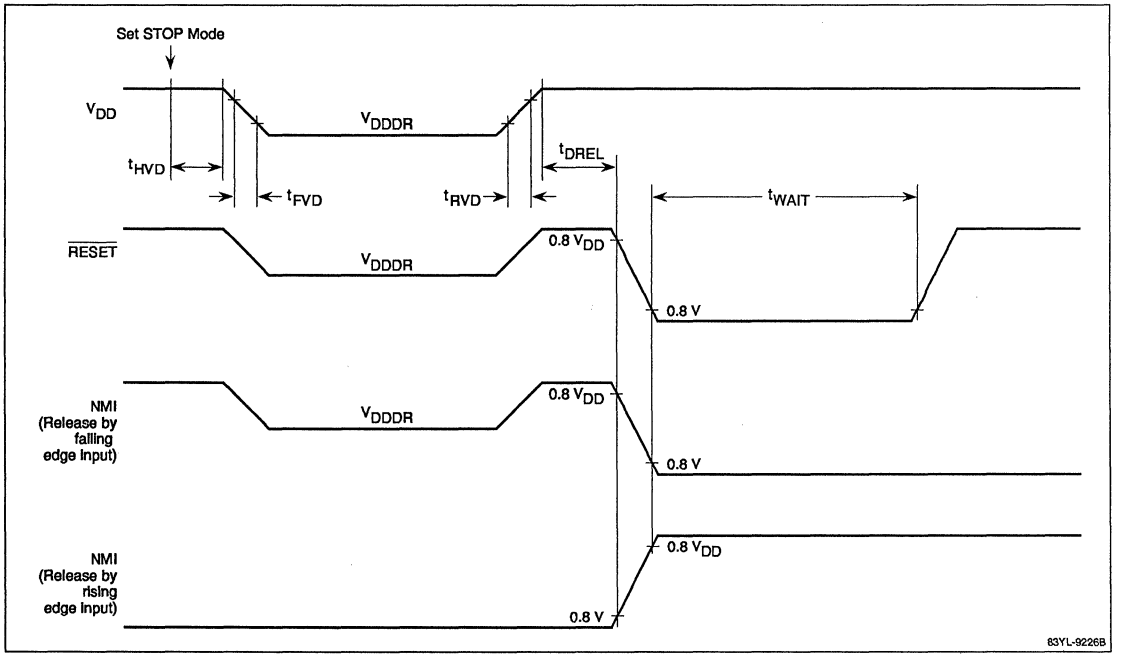
External Clock



Reset Input



Data Retention Characteristics



μPD78P218A PROGRAMMING

In the μPD78P218A, the mask ROM of μPD78218A is replaced by a one-time programmable ROM (OTP ROM) or a reprogrammable, ultraviolet erasable ROM (UV EPROM). The ROM is 32,768 x 8 bits and can be programmed using a general-purpose PROM writer with a μPD27C256A programming mode.

The PA-78P214CW/GC are the socket adaptors used for configuring the μPD78P218A to fit a standard PROM socket.

Refer to tables 6 and 7 and figures 15 and 16 for special information applicable to PROM programming.

Table 6. Pin Functions During PROM Programming

Pin	Pin*	Function
$P0_0$ - $P0_7$	A_0 - A_7	Address input pins for PROM operations
$P5_0/A_8$	A_8	Address input pin for PROM operations
$P2_1$/INTP0	A_9	Address input pin for PROM operations
$P5_2/A_{10}$ - $P5_6/A_{14}$	A_{10} - A_{14}	Address input pins for PROM operations
$P4_0/AD_0$ - $P4_7/AD_7$	D_0 - D_7	Data pins for PROM operations
$P6_5/\overline{WR}$	$\overline{CE}$	Strobes data into the PROM
$P6_4/\overline{RD}$	$\overline{OE}$	Enables a data read from the PROM
$P2_0$/NMI	NMI	PROM programming mode is entered by applying +12.5 volts to this pin
$\overline{RESET}$	$\overline{RESET}$	PROM programming mode requires applying a low voltage to this pin
$\overline{EA}$	V_{PP}	High voltage applied to this pin for program write/verify
V_{DD}	V_{DD}	Positive power supply pin
V_{SS}	V_{SS}	Ground

*Pin name in PROM programming mode.

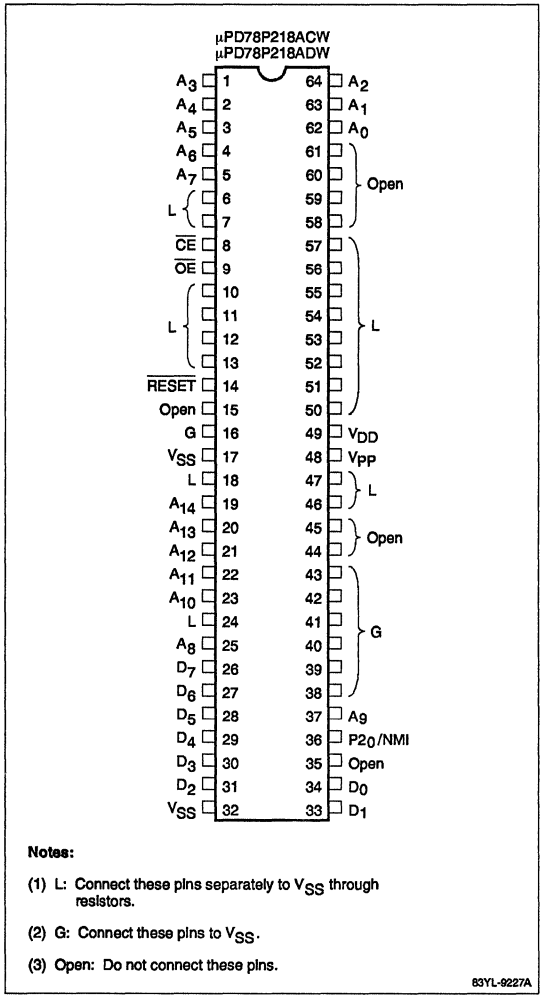
4b

Table 7. Summary of Operation Modes for PROM Programming

Mode	NMI	$\overline{RESET}$	$\overline{CE}$	$\overline{OE}$	V_{PP}	V_{DD}	D_0 - D_7
Program write	+12.5 V	L	L	H	+12.5 V	+6 V	Data input
Program verify	+12.5 V	L	H	L	+12.5 V	+6 V	Data output
Program inhibit	+12.5 V	L	H	H	+12.5 V	+6 V	High Z
Read out	+12.5 V	L	L	L	+5 V	+5 V	Data output
Output disable	+12.5 V	L	L	H	+5 V	+5 V	High Z
Standby	+12.5 V	L	H	L/H	+5 V	+5 V	High Z

Note: When +12.5 V is applied to V_{PP} and +6 V to V_{DD}, both $\overline{CE}$ and $\overline{OE}$ cannot be set to low level (L) simultaneously.

Figure 15. Pin Functions in μPD78P218A PROM Programming Mode; 64-Pin Plastic and Ceramic Shrink DIP, 64-Pin Plastic QUIP



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors.

(2) G: Connect these pins to V_{SS}.

(3) Open: Do not connect these pins.

83YL-9227A

PROM Write Procedure

(1) Set the pins not used for programming as indicated in figures 15 and 16. Connect the $\overline{RESET}$ pin to a low level, the V_{DD} and V_{PP} pins to +5 V, and apply +12.5 V to the NMI pin. The $\overline{CE}$ and $\overline{OE}$ pins should be high.

(2) Apply +6 V to the V_{DD} pin and +12.5 V to the V_{PP} pin.

(3) Provide the initial address to the A_0 to A_{14} pins.

(4) Provide write data.

(5) Provide 1-ms program pulse (active low) to the $\overline{CE}$ pin.

(6) This data is now verified with a pulse (active low) to the $\overline{OE}$ pin. If the data has been written, proceed to step 8; if not, repeat steps 4 to 6. If the data cannot be correctly written after 25 attempts, go to step 7.

(7) Classify as defective and stop write operation.

(8) Provide write data and supply program pulse (for additional writing) for 3 ms times the number of writes performed in step 5.

(9) Increment the address.

(10) Repeat steps 4 to 9 until the end address.

PROM Read Procedure

(1) Set the pins not used for programming as indicated in figures 15 and 16. Fix the $\overline{RESET}$ pin to a low level, the V_{DD} and V_{PP} pins to +5 V, and apply +12.5 V to the NMI pin. The $\overline{CE}$ and $\overline{OE}$ pins should be high.

(2) Input the address of the data to be read to pins A_0-A_{14}.

(3) Read mode is entered with a pulse (active low) on both the $\overline{CE}$ and $\overline{OE}$ pins.

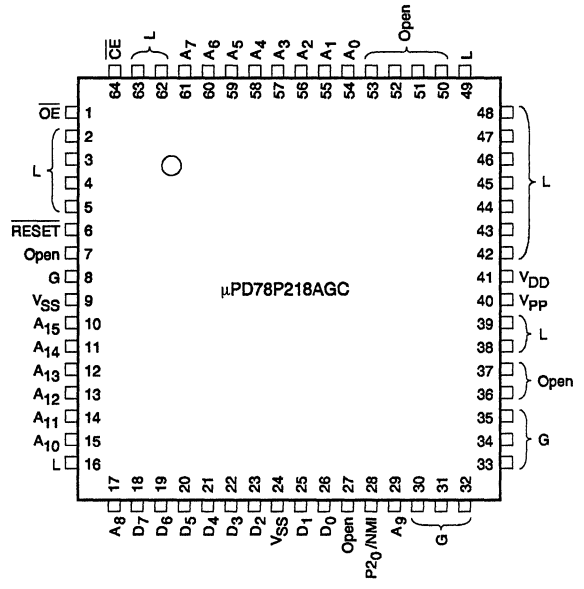
(4) Data is output to the D_0 to D_7 Pins.

EPROM Erasure

Data in an EPROM is erased by exposing the quartz window in the ceramic package to light having a wavelength shorter than 400 nm, including ultraviolet rays, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15 Ws/cm^2 (ultraviolet ray intensity x exposure time) is required to completely erase written data. Erasure by an ultraviolet lamp rated at 12 mW/cm^2 takes approximately 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

Figure 16. Pin Functions in μPD78P218A PROM Programming Mode; 64-Pin Plastic QFP



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors.

(2) G: Connect these pins to V_{SS}.

(3) Open: Do not connect these pins.

83YL-9228B

4b

DC Programming Characteristics

T_A = 25 ±5°C; V_{IP} = 12.5 ±0.5 V applied to NMI pin; V_{SS} = 0 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH}	V_{IH}	2.4		V_{DDP}+0.3	V	
Low-level input voltage	V_{IL}	V_{IL}	−0.3		0.8	V	
Input leakage current	I_{LIP}	I_{LI}			10	μA	$0 \le V_I \le V_{DDP}$
High-level output voltage	V_{OH1}	V_{OH1}	2.4			V	I_{OH} = −400 μA
	V_{OH2}	V_{OH2}	V_{DD}−0.7			V	I_{OH} = −100 μA
Low-level output voltage	V_{OL}	V_{OL}			0.45	V	I_{OL} = 2.1 mA
Output leakage current	I_{LO}				10	μA	$0 \le V_O \le V_{DDP}$, $\overline{OE}$ = V_{IH}
NMI pin high-voltage input current	I_{IP}				±10	μA	
V_{DDP} power voltage	V_{DDP}	V_{CC}	5.75	6.0	6.25	V	Program memory write mode
			4.5	5.0	5.5	V	Program memory read mode
V_{PP} power voltage	V_{PP}	V_{PP}	12.2	12.5	12.8	V	Program memory write mode
				V_{PP} = V_{DDP}		V	Program memory read mode

DC Programming Characteristics (cont)

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
V_{DDP} power current	I_{DDP}	I_{CC}		5	30	mA	Program memory write mode
				5	30	mA	Program memory read mode $\overline{CE} = V_{IL}$, $V_I = V_{IH}$
V_{PP} power current	I_{PP}	I_{PP}		5	30	mA	Program memory write mode $\overline{CE} = V_{IL}$, $\overline{OE} = V_{IH}$
				1	100	μA	Program memory read mode

*Corresponding symbols of the μPD27C256A.

AC Programming Characteristics (Write Mode)

T_A = 25 ±5°C; V_{IP} = 12.5 ±0.5 V applied to NMI pin; V_{SS} = 0 V; V_{DD} = 6 ±0.25 V; V_{PP} = 12.5 ±0.3 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Conditions
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}	2			μs	
Data input to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}	2			μs	
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}	2			μs	
Address hold time from $\overline{CE}$ ↑	t_{HCA}	t_{AH}	2			μs	
Input data hold time from $\overline{CE}$ ↑	t_{HCID}	t_{DH}	2			μs	
Output data hold time from $\overline{OE}$ ↑	t_{HOOD}	t_{DF}	0		130	ns	
V_{PP} setup time to $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}	1			ms	
V_{DDP} setup time to $\overline{CE}$ ↓	t_{SVDC}	t_{VCS}	1			ms	
Initial program pulse width	t_{WL1}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{WL2}	t_{OPW}	2.85		78.75	ms	
NMI high-voltage input setup time to $\overline{CE}$ ↓	t_{SPC}		2			μs	
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			150	ns	

*Corresponding symbols of the μPD27C256A.

AC Programming Characteristics (Read Mode)

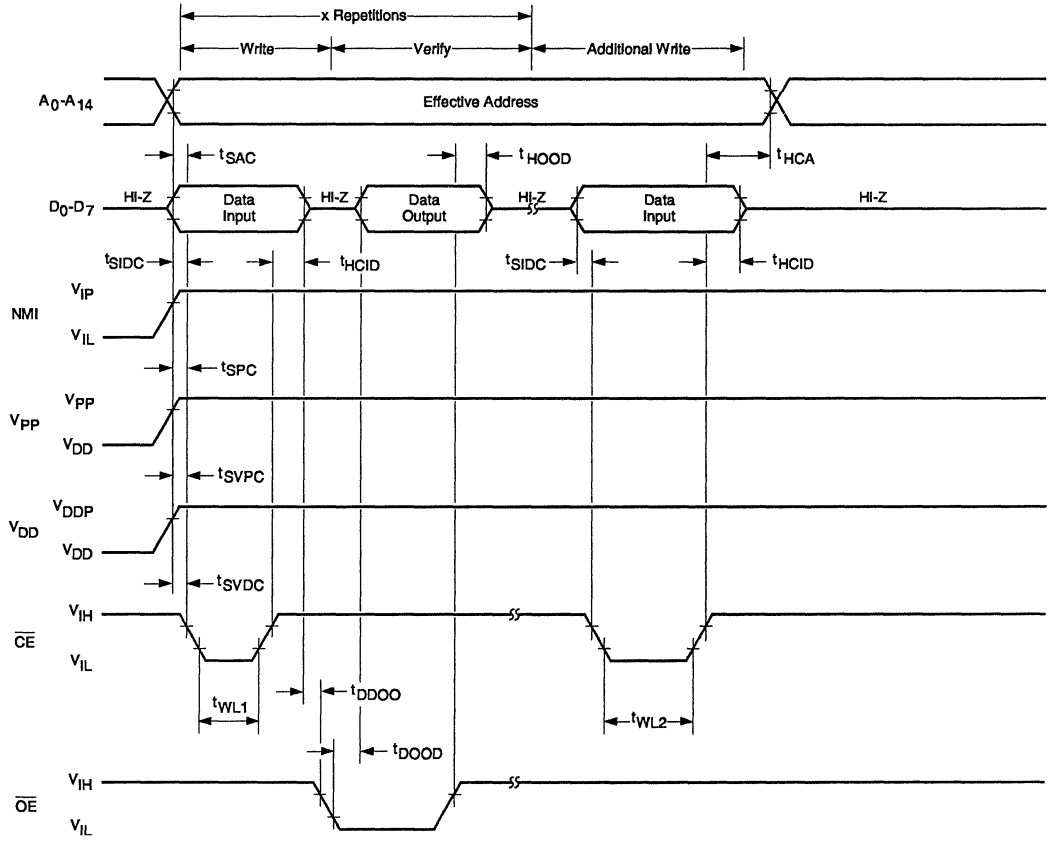
T_A = 25 ±5°C; V_{IP} = 12.5 ±0.5 V applied to NMI pin; V_{SS} = 0 V; V_{DD} = 5 ±0.5 V; V_{PP} = V_{DDP}

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
Address to data output time	t_{DAOD}	t_{ACC}			200	ns	$\overline{CE} = \overline{OE} = V_{IL}$
$\overline{CE}$ ↓ to data output time	t_{DCOD}	t_{CE}			200	ns	$\overline{OE} = V_{IL}$
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			75	ns	$\overline{CE} = V_{IL}$
Data hold time from $\overline{OE}$ ↑	t_{HCOD}	t_{DF}	0		60	ns	$\overline{CE} = V_{IL}$
Data hold time from address	t_{HAOD}	t_{OH}	0			ns	$\overline{CE} = \overline{OE} = V_{IL}$

* Corresponding symbols of the μPD27C256A.

PROM Timing Diagrams

PROM Write/Verifying Mode

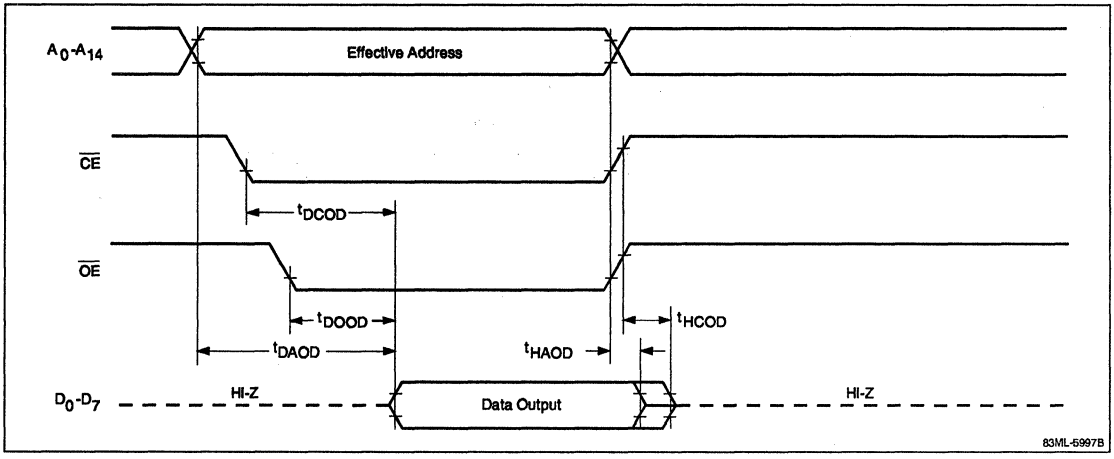


Notes:
(1) V_{DD} must be applied before applying V_{PP}. It should be removed after removing V_{PP}.
(2) V_{PP} must not exceed +13V, including overshoot.

83ML-5996B

4b

PROM Read Mode



NEC
NEC Electronics Inc.

μPD78224 Family (μPD78220/224/P224) 8-Bit, K-Series Microcontrollers With Analog Comparators, Real-Time Output Ports

August 1993

Description

The μPD78220, μPD78224, and μPD78P224 are members of the K-Series® of microcontrollers. These 8-bit, single-chip microcontrollers contain extended addressing capabilities for up to 1M byte of external memory. The devices also integrate sophisticated analog and digital peripherals as well as two low-power standby modes that make them ideal for low-power/battery backup applications.

The μPD78224 family focuses on embedded control with features such as hardware multiply and divide, two levels of interrupt response, four banks of main registers for multitasking, and macro service for processor-independent peripheral and memory DMA. Augmenting this high-performance core are advanced components; for example, eight analog voltage comparators, two independent serial interfaces, several counter/timers for PWM outputs, and a real-time output port. On board memory includes 640 bytes of RAM and 16K bytes of mask ROM or OTP ROM.

The macro service routine allows data to be transferred between any combination of memory and peripherals independent of the current program execution. The four banks of processor registers allow simplified context switching to be performed. Both features, combined with powerful on-chip peripherals, make the μPD78224 family ideal for a wide variety of embedded control applications.

Features

- Complete single-chip microcontroller
  - 8-bit ALU
  - 16K ROM
  - 640 bytes RAM
  - Both 1-bit and 8-bit logic
- Instruction prefetch queue
- Hardware multiply and divide
- Memory expansion
  - 8085 bus-compatible
  - 64K program address space
  - 1M data address space
- Large I/O capacity: up to 71 I/O port lines
- Extensive timer/counter functions
  - One 16-bit timer/counter/event counter
  - Two 8-bit timer/counter/event counters
- Four timer-controlled PWM channels
- Two 4-bit real-time output ports
- Extensive interrupt handler
  - Vectored interrupt handling
  - Programmable priority
  - Macro service mode
- Two independent serial ports
- Refresh output for pseudostatic RAM
- On-chip clock generator
  - 12-MHz maximum CPU clock frequency
  - 0.33-μs instruction cycle
- CMOS silicon gate technology
- 5-volt power supply

K-Series is a registered trademark of NEC Electronics,Inc.

Ordering Information

Part Number	ROM	Package (Dwg)
μPD78220L	ROMless	84-pin PLCC (P84L-50A3-1)
μPD78224L-xxx	16K mask ROM	
μPD78P224L	16K OTP ROM	
μPD78220GJ-5BG	ROMless	94-pin plastic QFP (S94GJ-80-5BG-1)
μPD78224GJ-xxx-5BG	16K mask ROM	
μPD78P224GJ-5BG	16K OTP ROM	

4c

Pin Identification

Symbol	Function
$P0_0$ - $P0_7$	Output port 0
$P1_0$ - $P1_7$	I/O port 1
$P2_0$/NMI	Input port 2/Nonmaskable interrupt input
$P2_1$ - $P2_2$/INTP0 - INTP1	Input port 2/Ext interrupt input/timer trigger
$P2_3$/INTP2/CI	Input port 2/Ext interrupt input/Clock input
$P2_4$/INTP3	Input port 2/Ext interrupt input/timer trigger
$P2_5$/INTP4	Input port 2/Ext interrupt input
$P2_6$/INTP5	Input port 2/Ext interrupt input
$P2_7$/INTP6/SI	Input port 2/Ext interrupt input/Serial input
$P3_0$/RxD	I/O port 3/Serial receive input
$P3_1$/ TxD	I/O port 3/Serial transmit output
$P3_2$/$\overline{SCK}$	I/O port 3/Serial clock input/output
$P3_3$/SO/SB0	I/O port 3/Serial output/Serial bus I/O
$P3_4$ - $P3_7$/TO0 - TO3	I/O port 3/Timer output
$P4_0$ - $P4_7$/AD_0 - AD_7	I/O port 4/Lower address byte/data bus
$P5_0$ - $P5_7$/A_8 - A_{15}	I/O port 5/Upper address byte

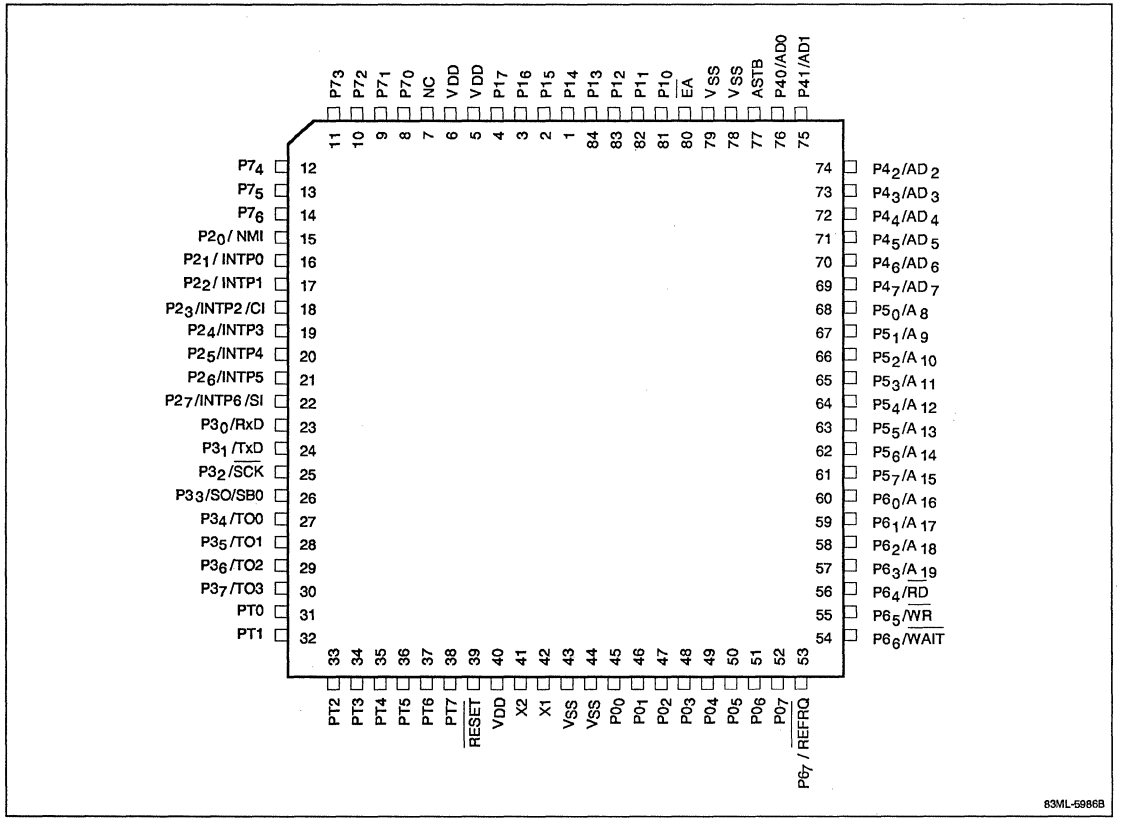
Pin Identification

Symbol	Function
$P6_0$ - $P6_3/A_{16}$ - A_{19}	Output port 6/Extended address nibble
$P6_4/\overline{RD}$	I/O port 6/Read strobe output
$P6_5/\overline{WR}$	I/O port 6/Write strobe output
$P6_6/\overline{WAIT}$	I/O port 6/Wait input
$P6_7/\overline{REFREQ}$	I/O port 6/Rerresh output
$P7_0$ - $P7_6$	I/O port 7
PT0 - PT7	Port T analog inputs to voltage comparators

Symbol	Function
ASTB	Address strobe output
$\overline{RESET}$	External reset input
$\overline{EA}$	External memory access control input
X1, X2	External crystal or external clock input
V_{DD}	Positive power supply input
V_{SS}	Power return; normallly ground
NC	No connection
IC	Internal connection; connect to V_{SS}

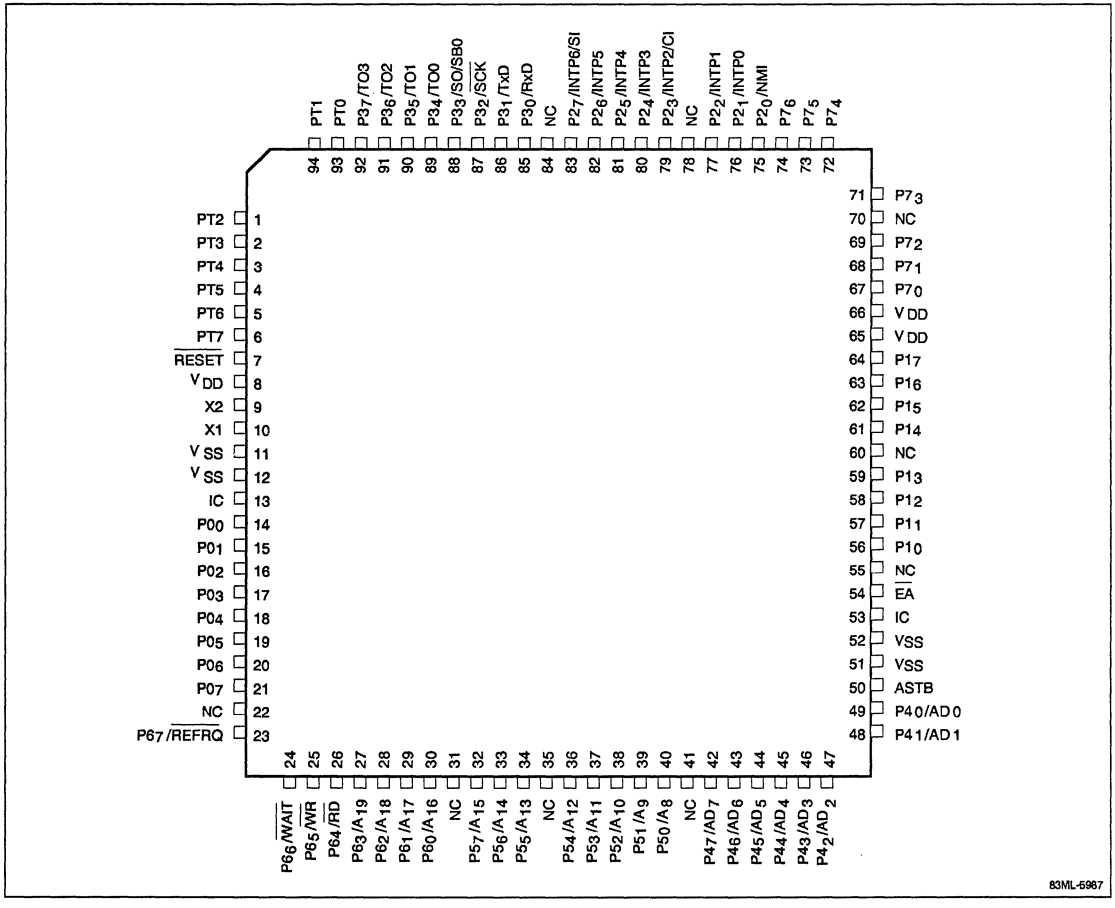
Pin Configurations

84-Pin PLCC



Pin Configurations (cont)

94-Pin Plastic QFP



4c

Pin Functions

$P0_0$ - $P0_7$. Port 0 is an 8-bit, tristate output port. Port 0 can also be configured as two 4-bit, real-time (timer-controlled) output ports.

$P1_0$ - $P1_7$. Port 1 is an 8-bit bidirectional tristate port. Bits are individually programmable as input/output. Each pin is capable of driving an LED directly (8 mA).

$P2_0$ - $P2_7$. Port 2 is an 8-bit input port.

NMI. Nonmaskable interrupt input.

INTP0 - INTP6. External interrupt inputs. INTP0, INTP1, and INTP3 are timer capture trigger inputs.

CI. External clock input to the timer.

SI. Serial data input for three-line serial I/O mode.

$P3_0$ - $P3_7$. Port 3 is an 8-bit tristate I/O port, each bit programmable as input/output.

RxD. Receive serial data input.

TxD. Transmit serial data output.

$\overline{SCK}$. Serial shift clock output/input.

SO. Serial data output for three-line serial I/O mode.

SB0. I/O bus for the clocked serial interface.

TO0 - TO3. Timer flip-flop outputs

$P4_0$ - $P4_7$. Port 4 is an 8-bit, bidirectional tristate port.

AD_0 - AD_7. Multiplexed address/data bus used with external memory or expanded I/O.

$P5_0$ - $P5_7$. Port 5 is an 8-bit, tristate output port.

A_8 - A_{15}. Upper-order address bus used with external memory or expanded I/O.

$P6_0$ - $P6_3$. Pins $P6_0$ - $P6_3$ of port 6 are outputs.

A_{16} - A_{19}. Extended-order address bus used with external memory.

$P6_4$ - $P6_7$. Pins $P6_4$ - $P6_7$ of port 6 are individually programmable tristate input/output pins.

$\overline{RD}$. Read strobe output used by external memory (or data registers) to place data on the I/O bus during a read operation.

$\overline{WR}$. Write strobe output used by external memory (or data registers) to latch data from the I/O bus during a write operation.

$\overline{WAIT}$. Wait signal input.

$\overline{REFRQ}$. Refresh pulse output used by external pseudostatic memory.

$P7_0$ - $P7_6$. Port 7 has seven individually programmable tristate I/O pins.

PT0 - PT7. Port T is an eight-line input port. The analog voltage on each line is compared continuously with a programmable threshold voltage.

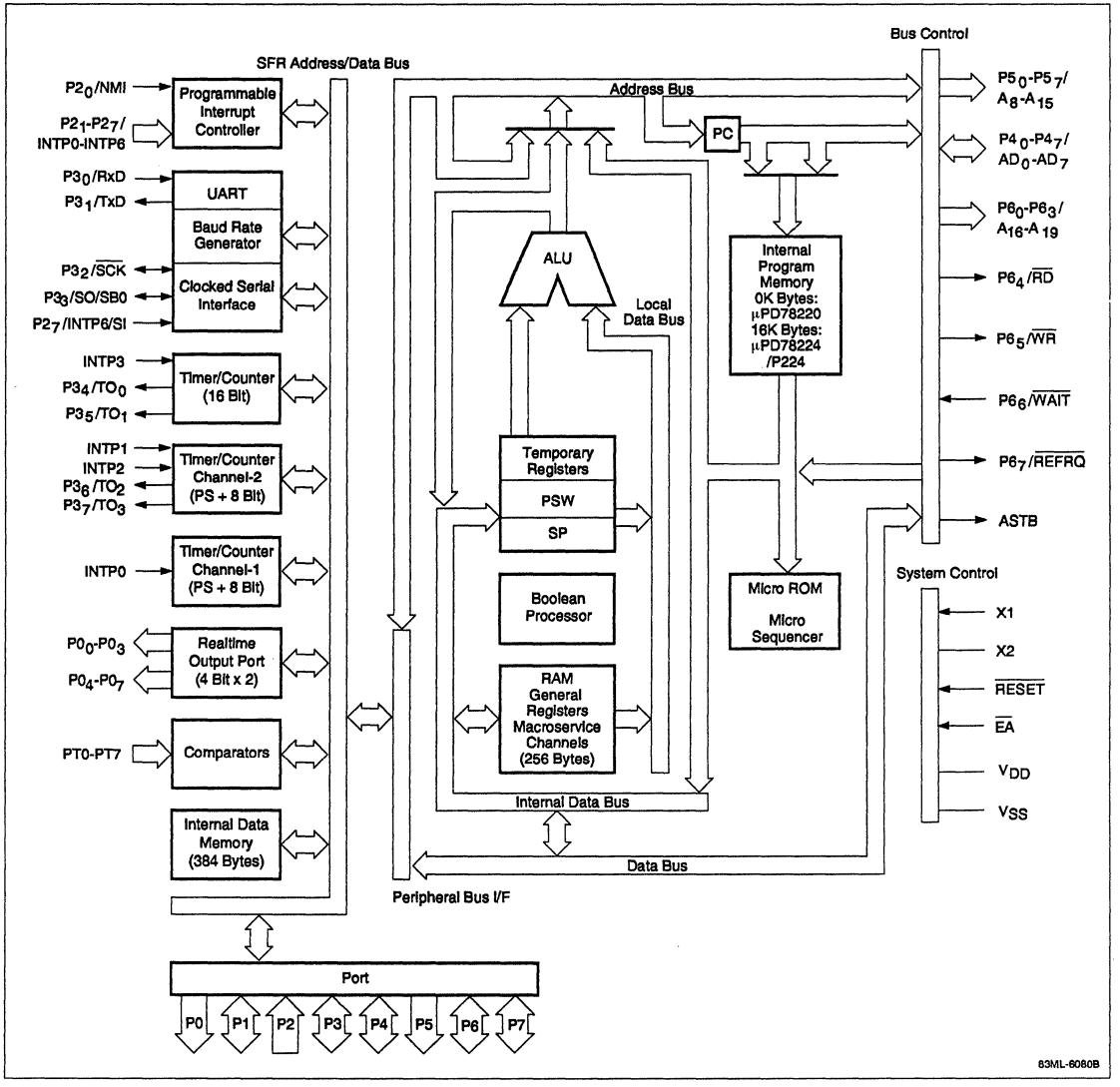
ASTB. Address strobe output used by external circuitry to latch the low-order 8 address bits during the first part of a read or write cycle.

$\overline{RESET}$. A low level on this external reset input sets all registers to their specified reset values. This pin, together with $P2_0$/NMI, sets the μPD78P224 in the PROM programming mode.

$\overline{EA}$. Control signal input that selects external memory or internal ROM as the program memory. When $\overline{EA}$ is low, ROMless mode is initiated and external memory is accessed.

X1, X2. For frequency control of the internal clock oscillator, a crystal is connected to X1 and X2. If the clock is supplied by an external source, the clock signal is connected to X1 and the inverted clock signal is connected to X2.

Block Diagram



4c

FUNCTIONAL DESCRIPTION

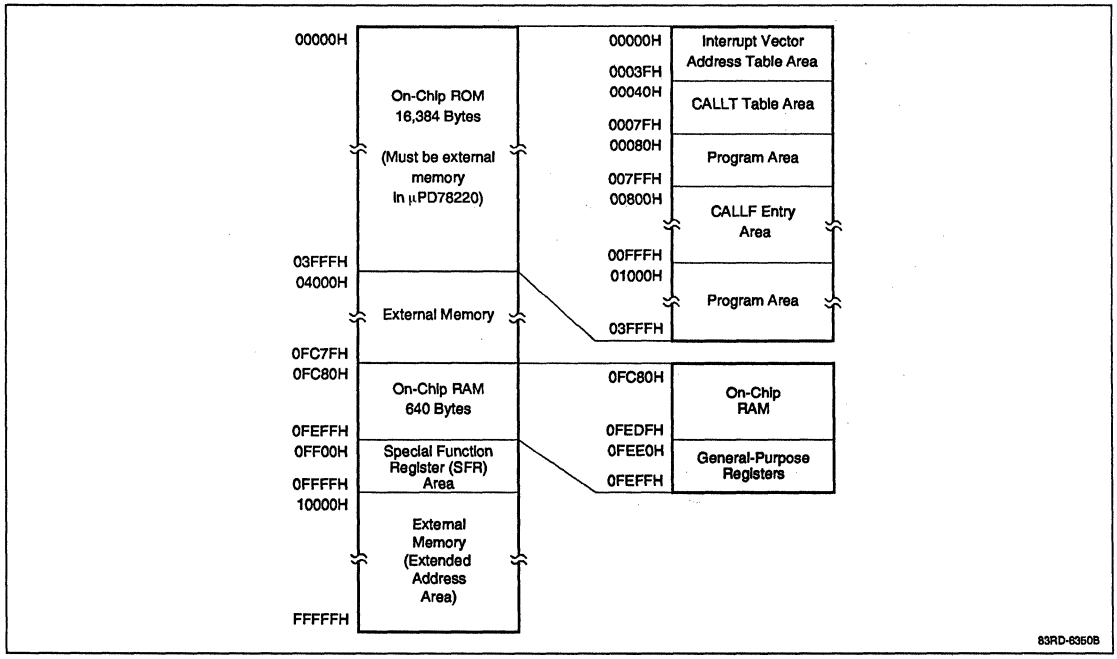
Timing

The maximum clock frequency is 12 MHz. The clock is derived from an external crystal or an external oscillator. The internal processor clock is two-phase and the machine states are executed at a rate of 6 MHz. The shortest instructions require two states (333 ns). The CPU contains a one-byte instruction prefetch. This allows a subsequent instruction to be fetched during the execution of an instruction that does not reference memory.

Memory Map

The μPD78224 family has 1M bytes of address space. This address space is partitioned into 64K bytes of program memory starting at address 00000H. (See figure 1). The remainder of the 1M bytes can be accessed as data memory space.

Figure 1. Memory Map



External memory is supported by I/O port 4, an 8-bit multiplexed address/data bus. The memory mapping register controls the size of external memory as well as the number of added wait states. The upper address byte is derived from port 5, and the extended address nibble is derived from port 6.

The μPD78224 has on-chip mask ROM occupying the space from 00000H to 03FFFH. When the ROM is used and no other program or data space is required, ports 4, 5, and 6 are available as additional I/O ports.

General-Purpose Registers

The general-purpose registers are mapped into specific addresses in data memory. They are made up of four banks, each bank consisting of eight 8-bit or four 16-bit registers. The register bank used is specified by a CPU instruction. This can be checked by reading RBS0 and RBS1 in the program status word (PSW). The general-purpose register configuration is shown in figure 2.

Special Registers

There are three different special registers. The first is a 16-bit binary counter that holds the next program address to be executed and is named the program counter. The stack pointer is the second special 16-bit register. The stack pointer holds the address of the stack area (a last in, first out system). The third special register is an 8-bit program status word. This register contains various flags that are set or preset depending on the results of instruction execution. The program status word format is as follows:

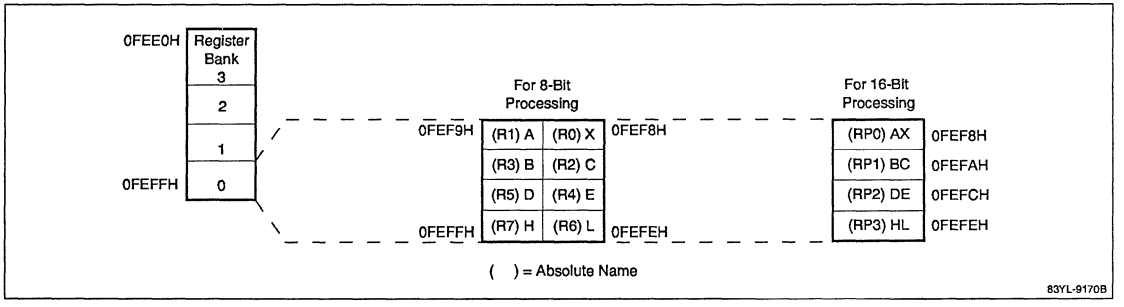
7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

CY	Carry flag
ISP	Interrupt priority status flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

Special Function Registers

These registers are assigned to special functions such as the mode and control registers for on-chip peripheral hardware. They are mapped into the 256-byte memory space from 0FF00H to 0FFFFH. Table 1 is a list of special function registers.

Figure 2. Register Mapping



4c

Table 1. Special Function Registers

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
0FF00H	Port 0	P0	R/W	x	x	—	Undefined
0FF01H	Port 1	P1	R/W	x	x	—	Undefined
0FF02H	Port 2	P2	R	x	x	—	Undefined
0FF03H	Port 3	P3	R/W	x	x	—	Undefined
0FF04H	Port 4	P4	R/W	x	x	—	Undefined
0FF05H	Port 5	P5	R/W	x	x	—	Undefined
0FF06H	Port 6	P6	R/W	x	x	—	x0H
0FF07H	Port 7	P7	R/W	x	x	—	Undefined
0FF0AH	Port 0 buffer register (low)	P0L	R/W	x	x	—	Undefined
0FF0BH	Port 0 buffer register (high)	P0H	R/W	x	x	—	Undefined
0FF0CH	Real-time output port control register	RTPC	R/W	x	x	—	00H
0FF10H-FF11H	16-bit compare register 0 (16-bit timer/counter)	CR00	R/W	—	—	x	Undefined
0FF12H-FF13H	16-bit compare register 1 (16-bit timer/counter)	CR01	R/W	—	—	x	Undefined
0FF14H	8-bit compare register (8-bit timer/counter 1)	CR10	R/W	—	x	—	Undefined
0FF15H	8-bit compare register (8-bit timer/counter 2)	CR20	R/W	—	x	—	Undefined
0FF16H	8-bit compare register (8-bit timer/counter 2)	CR21	R/W	—	x	—	Undefined
0FF17H	BRG 8-bit compare register	CR30	R/W	—	x	—	Undefined
0FF18H-FF19H	16-bit capture register (16-bit timer/counter)	CR02	R	—	—	x	Undefined
0FF1AH	8-bit capture register (8-bit timer/counter 2)	CR22	R	—	x	—	Undefined
0FF1CH	8-bit capture/compare register (8-bit timer/counter 1)	CR11	R/W	—	x	—	Undefined
0FF20H	Port 0 mode register	PM0	W	—	x	—	FFH
0FF21H	Port 1 mode register	PM1	W	—	x	—	FFH
0FF23H	Port 3 mode register	PM3	W	—	x	—	FFH
0FF25H	Port 5 mode register	PM5	W	—	x	—	FFH
0FF26H	Port 6 mode register	PM6	R/W	—	x	—	FFH
0FF27H	Port 7 mode register	PM7	W	—	x	—	7FH
0FF30H	Capture/compare control register 0	CRC0	W	—	x	—	10H
0FF31H	Timer output control register	TOC	W	—	x	—	00H
0FF32H	Capture/compare control register 1	CRC1	W	—	x	—	00H
0FF34H	Capture/compare control register 2	CRC2	W	—	x	—	00H
0FF43H	Port 3 mode control register	PMC3	R/W	x	x	—	00H
0FF50H-FF51H	16-bit timer register 0	TM0	R	—	—	x	0000H
0FF52H	8-bit timer register 1: CH -1	TM1	R	—	x	—	00H
0FF54H	8-bit timer register 2: CH-2	TM2	R	—	x	—	00H
0FF56H	BRG 8-bit timer register	TM3	R	—	x	—	00H
0FF5CH	Prescaler mode register 0	PRM0	W	—	x	—	00H
0FF5DH	Timer control register 0	TMC0	R/W	—	x	—	00H

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol		R/W	Access Units (Bits) 1	8	16	State After Reset
0FF5EH	Prescaler mode register 1	PRM1		W	—	x	—	00H
0FF5FH	Timer control register 1	TMC1		R/W	—	x	—	00H
0FF6EH	Port T mode register	PMT		R/W	x	x	—	00H
0FF6FH	Port T	PT		R	x	x	—	Undefined
0FF80H	Clocked serial interface mode register	CSIM		R/W	x	x	—	00H
0FF82H	Serial bus interface control register	SBIC		R/W	x	x	—	00H
0FF86H	Serial shift register	SIO		R/W	—	x	—	Undefined
0FF88H	Asynchronous serial interface mode register	ASIM		R/W	x	x	—	80H
0FF8AH	Asynchronous serial interface status register	ASIS		R	x	x	—	00H
0FF8CH	Serial receive buffer: UART	RxB		R	—	x	—	Undefined
0FF8EH	Serial transmit shift register: UART	TxS		W	—	x	—	Undefined
0FFC0H	Standby control register	STBC		R/W	—	x	—	0000x000B
0FFC4H	Memory expansion mode register	MM		R/W	x	x	—	20H
0FFC5H	Programmable wait control register	PW		R/W	x	x	—	80H
0FFC6H	Refresh mode register	RFM		R/W	x	x	—	00H
0FFE0H	Interrupt request flag register L	IF0L	IF0	R/W	x	x	x	Undefined
0FFE1H	Interrupt request flag register H	IF0H		R/W	x	x	—	Undefined
0FFE4H	Interrupt mask flag register L	MK0L	MK0	R/W	x	x	x	FFFFH
0FFE5H	Interrupt mask flag register H	MK0H		R/W	x	x	—	FFFFH
0FFE8H	Priority specification flag register L	PR0L	PR0	R/W	x	x	x	FFFFH
0FFE9H	Priority specification flag register H	PR0H		R/W	x	x	—	FFFFH
0FFECH	Interrupt service mode speification flag register L	ISM0L	ISM0	R/W	x	x	x	0000H
0FFEDH	Interrupt service mode specification flag register H	ISM0H		R/W	x	x	—	0000H
0FFF4H	External interrupt mode register 0	INTM0		R/W	x	x	—	00H
0FFF5H	External interrupt mode register 1	INTM1		R/W	x	x	—	00H
0FFF8H	Interrupt status register	IST		R/W	x	x	—	00H

4c

Input/Output Ports

Functions of ports P0 - P7 and port PT are explained below. All ports are 8 bits wide except P7, which is 7 bits wide.

Port	Function
P0	8-bit output port or two 4-bit real time output ports
P1	Bit programmable for input or output; large current capacity
P2	Input
P3	Bit programmable for input or output
P4	Input or output
P5	Output
$P6_0$ - $P6_3$	Output
$P6_4$ - $P6_7$	Bit programmable for input or output
P7	Bit programmable for input or output
PT	Inputs to eight voltage comparators

Real-time Output Port

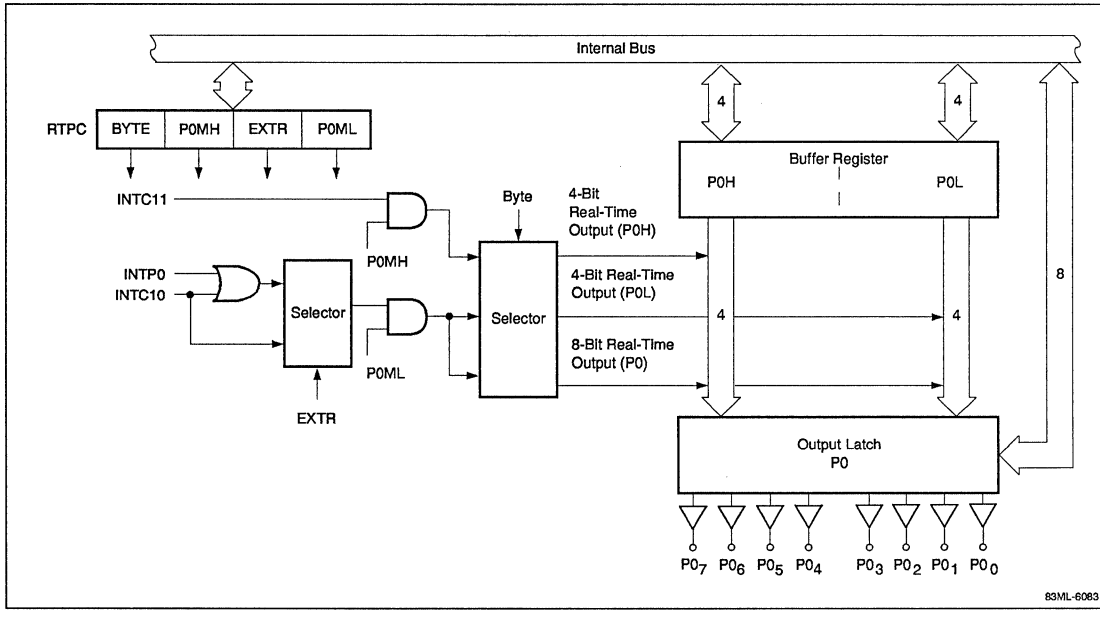
The real-time output port (figure 3) shares pins with port 0. The high and low nibbles may be treated separately or together. In the real-time output function, data stored beforehand in the buffer register is transferred to the output latch simultaneously with the generation of either a timer interrupt or external interrupt. Using the real-time output function in conjunction with the macro service function enables port 0 to output preprogrammed patterns at preprogrammed variable time intervals.

Port T

As shown in figure 4, the analog input voltage on each line of port T is compared with a programmable threshold voltage. The comparator output is 1 if the input voltage is higher than the threshold, or 0 if it is lower.

Four bits from the PTM regster are decoded to set the threshold voltage at one of 15 steps: V_{DD} x 1/16 through V_{DD} x 15/16. Each comparator operates continously as follows.

Figure 3. Real-Time Output Port



(1) Threshold voltage is set by writing the PTM register.

(2) As each comparison is completed, the result is latched in port T and the next comparison begins.

(3) Unless the PTM regiser is written, the threshold voltage is not changed.

Two bits from the PTM register specify the connection of pullup resistors in 4-bit units. When PTM is set to 00H, the resistor ladder is released and threshold voltage is not supplied to the comparators. This can be done in the standby mode to eliminate unnecessary current drain.

Serial Interface

The μPD78224 family has two independent serial interfaces.

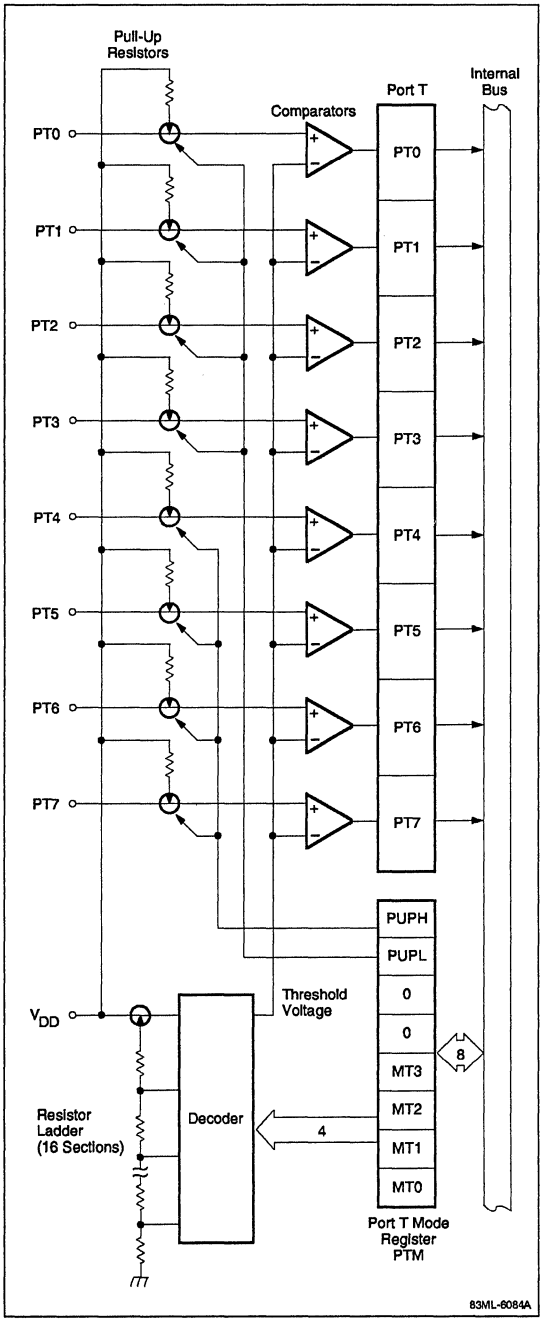
- Asynchronous serial interface (UART) (figure 5)
- Clock-synchronized serial interface (figure 6)

A universal asynchronous receiver transmitter (UART) is used as an asynchronous serial interface. This interface transfers one byte of data following a start bit. The μPD78224 contains a baud rate generator. This allows data to be transferred over a wide range of transfer rates.

The clock-synchronized serial interface has two different modes of operation:

- Three-line serial I/O mode.
  In this mode, data 8 bits long is transferred along three lines: a serial clock (SCK) line and two serial bus lines lines (SO and SI). This mode is convenient when a μPD78224 device is connected to peripheral I/Os and display controllers that have the conventional clock-synchronized serial interface.
- Serial bus interface mode (SBI)
  In this mode, the μPD78224 family can communicate data with several devices using the serial clock (SCK) and the serial data bus (SBO) lines. This mode conforms to NEC's serial bus format. In SBI mode, addresses that select a device to communicate with, commands that direct the device, and actual data are output to the serial data bus. A handshake line, which was required for connecting several devices in the conventional clock-synchronized serial interface, is not needed.

Figure 4. Comparator Port T



4c

Figure 5. Asynchronous Serial Interface

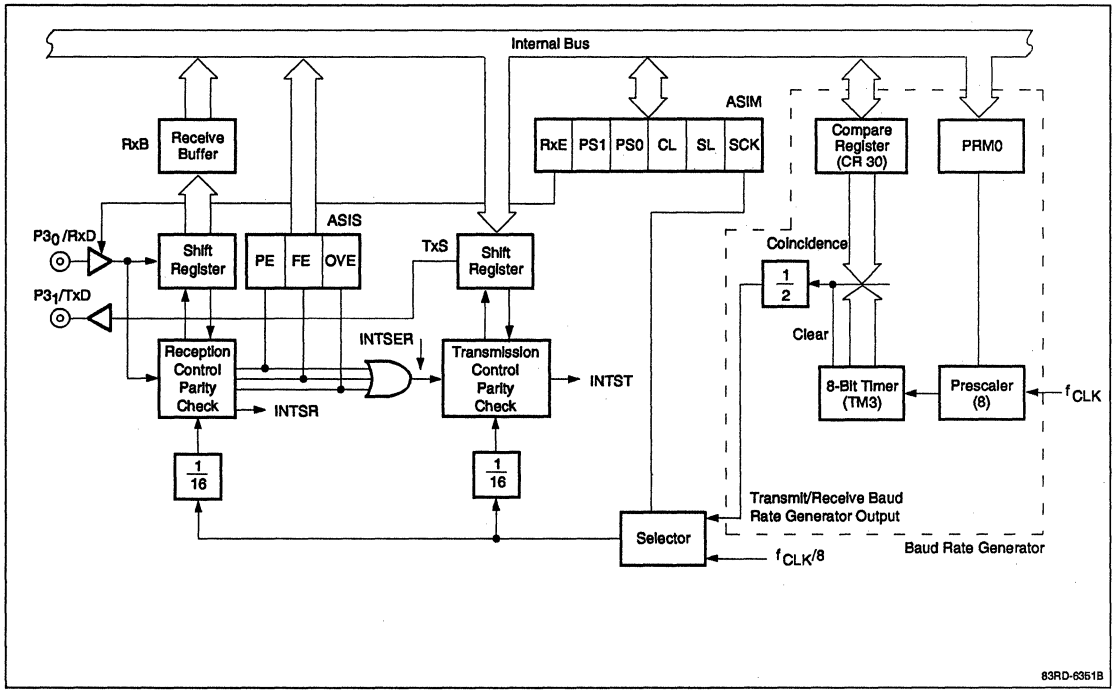
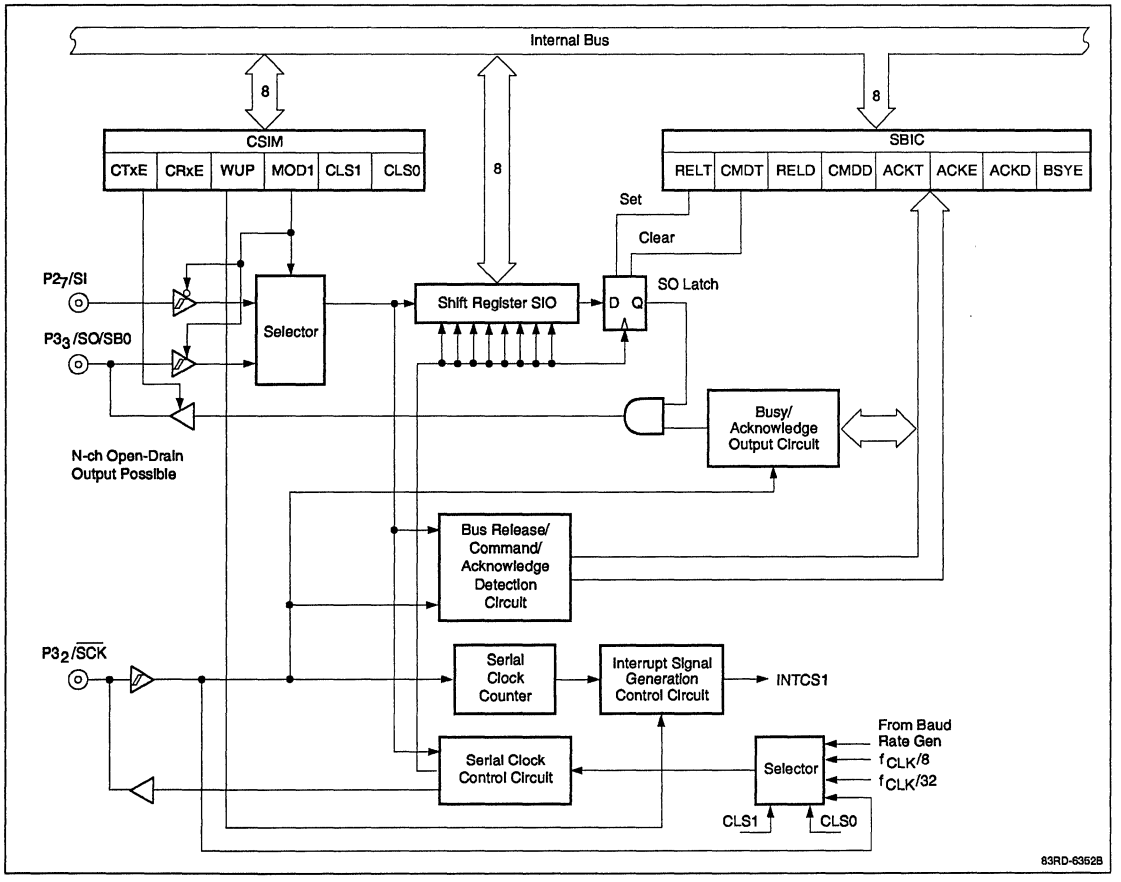


Figure 6. Clock-Synchronized Serial Interface



4c

Timer/Counters

The μPD78224 family has three timer/counters: one 16-bit and two 8-bit. The 16-bit timer/counter (figure 7) has the basic functionality of an interval timer, a programmable square wave output, and a pulse-width measurer. These functions can provide a digital delayed one-shot output, a pulse-width modulated output, and a cycle measurer.

The two 8-bit timer/counters can provide the basic functions of an interval timer and a pulse-width measurer. Timer/counter 1 can also be used as a timer for output trigger generation for the real-time output port. Timer/counter 2 can also provide an external event counter, a one-shot timer, a programmable square-wave output, a pulse-width modulated output, and a cycle measurer. See figures 8 and 9.

Figure 7. 16-Bit Timer/Counter

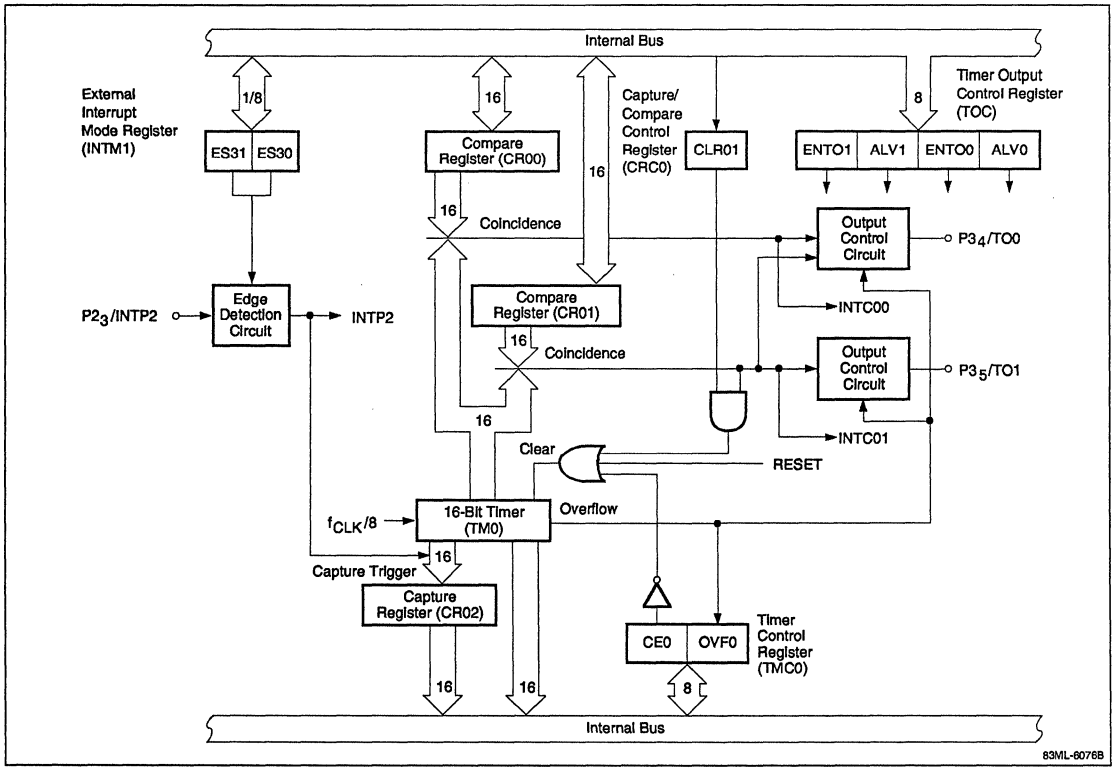
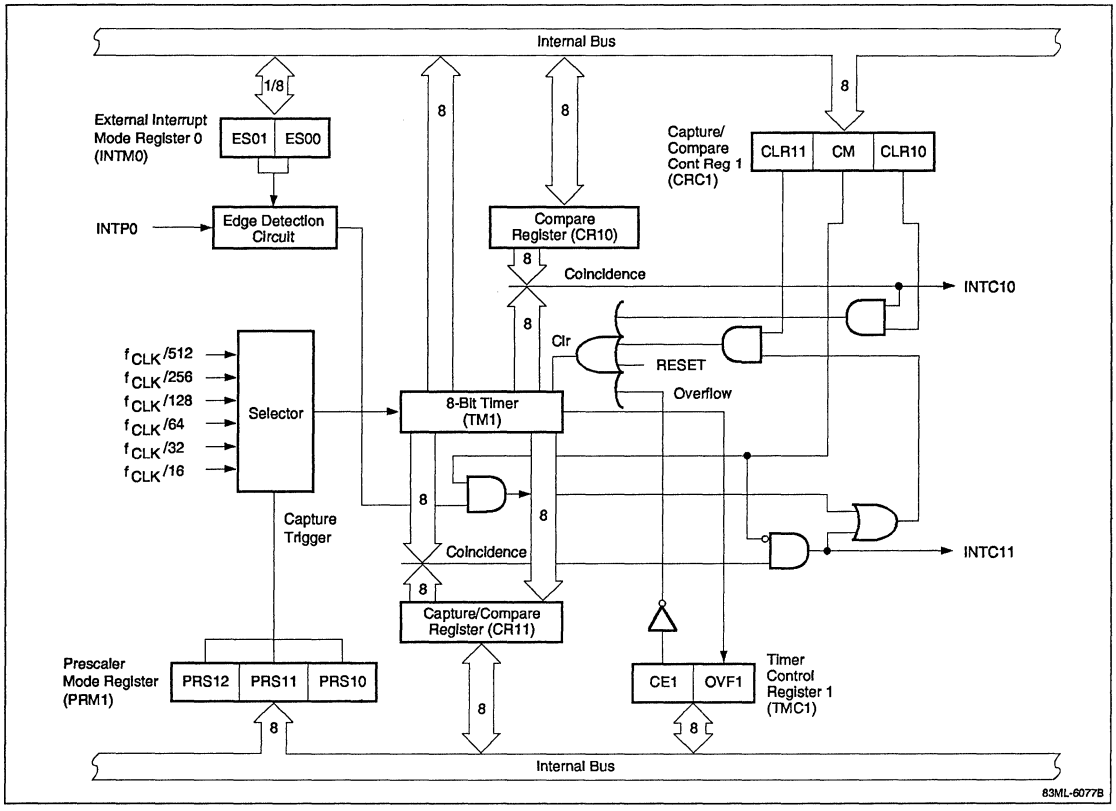
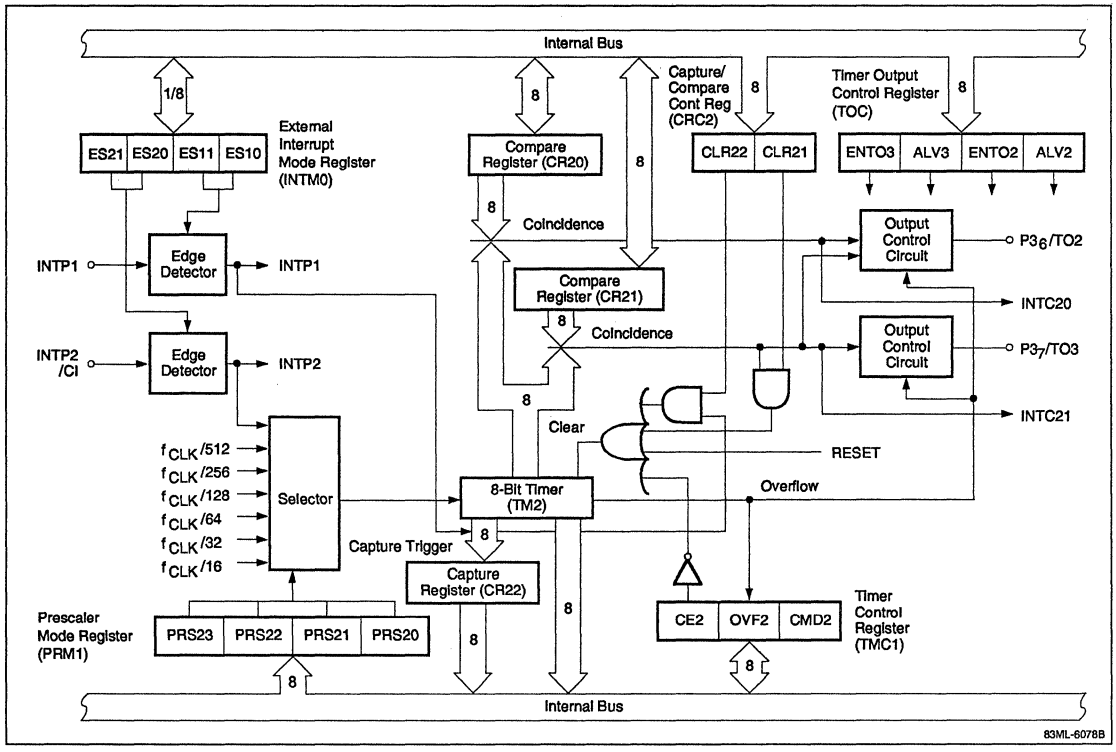


Figure 8. 8-Bit Timer/Counter 1



4c

Figure 9. 8-Bit Timer/Counter 2



Interrupts

There are 18 interrupt request sources; each source is allocated a location in the vector table. (See table 2). There is one software interrupt request and one of the remaining 17 interrupts is nonmaskable. The software interrupt and the nonmaskable interrupt are unconditionally received even in the DI state. These two interrupts possess the maximum priority. The maskable interrupt requests are subject to mask control by the setting of the interrupt mask flag.

There are default priorities associated with each maskable interrupt and these can be assigned to either of two programmable priority levels. Interrupts may be serviced by the vectored interrupt method where a branch to a desired service program is executed. Interrupts may also be handled by the macro service function where a preassigned process is performed without program intervention.

Table 2. Interrupt Sources and Vector Addresses

Interrupt Request Type	Default Priority	Interrupt Request Source	Macro Service Handling	Vector Table Address
Software	None	BRK instruction execution	—	003EH
Nonmaskable	None	NMI (pin input edge detection)	—	0002H
Maskable	0	INTP0 (pin input edge detection)	—	0006H
	1	INTP1 (pin input edge detection)	—	0008H
	2	INTP2 (pin input edge detection)	—	000AH
	3	INTP3 (pin input edge detection)	—	000CH
	4	INTC00 (TM0-CR00 coincidence signal generation)	—	0014H
	5	INTC01 (TM0-CR01 coincidence signal generation)	—	0016H
	6	INTC10 (TM1-CR10 coincidence signal generation)	Yes	0018H
	7	INTC11 (TM1-CR11 coincidence signal generation)	Yes	001AH
	8	INTC21 (TM2-CR21 coincidence signal generation)	—	001CH
	9	INTP4 (pin input edge detection)	Yes	000EH
	10	INTP5 (pin input edge detection)	—	0010H
	11	INTP6 (pin input edge detection)	—	0012H
	12	INTSER (generation of asynchronous serial interface receive error)	—	0020H
	13	INTSR (end of asynchronous serial interface reception)	Yes	0022H
	14	INTST (end of asynchronous serial interface transmission)	Yes	0024H
	15	INTCSI (end of clocked serial interface transfer)	Yes	0026H

4c

Macro Service

When macro service function can be programmed to transfer data from a special function register to memory or from memory to a special function register. Transfer events are triggered by interrupt requests and take place without software intervention. There are six interrupt requests where macro servicing can be executed. The macro service function is controlled by the macro service mode register and the macro service channel pointer. The macro service mode register assigns the macro servicing mode and the macro service channel pointer indicates the address of the memory location pointers. The location of each register and its corresponding interrupt is shown in figure 10.

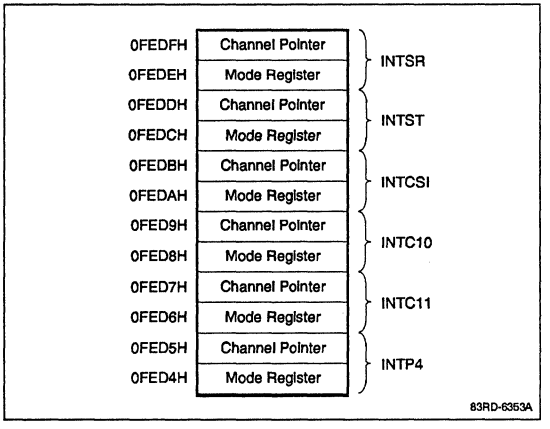
Refresh

The refresh signal is used with a pseudostatic RAM. The refresh cycle can be set to one of four intervals ranging from 2.6 to 21.3 μs. The refresh is timed to follow a read or write operation to avoid interference.

Standby Modes

HALT and STOP functions reduce system power consumption. In HALT mode, the CPU stops and the system clock continues to run. A release of the HALT mode is initiated by an unmasked interrupt request, an NMI, or a $\overline{\text{RESET}}$ input. In the STOP mode, the CPU and system clock are both stopped, reducing the power consumption even further. The STOP mode is released by an NMI input or a $\overline{\text{RESET}}$ input.

Figure 10. Macro Service Control Word Map



ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = +25°C

Operating voltage, V_{DD}	−0.5 to +7.0 V
Input voltage, V_I	−0.5 to V_{DD} + 0.5 V
Output voltage, V_O	−0.5 to V_{DD} + 0.5 V
Low-level output current, I_{OL}	
per pin	30 mA (peak), 15 mA (mean)
total, all output pins	150 mA (peak), 100 mA (mean)
High-level output current, I_{OH}	
per pin	−2 mA
total, all output pins	−50 mA
Operating temperature, T_{OPT}	−40 to +85°C
Storage temperature, T_{STG}	−65 to +150°C

Exposure to absolute maximum ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC characteristics.

Operating Conditions

Oscillation Frequency	T_A	V_{DD}
f_{XX} = 4 to 12 MHz	−40 to +85°C	+5 V ±5%
	−10 to +70°C	+5 V ±10%

Capacitance

T_A = +25°C; V_{DD} = V_{SS} = 0 V

Item	Symbol	Max	Unit	Conditions
Input capacitance	C_I	20	pF	f = 1 MHz; pins not used for measurement are at 0 V
Output capacitance	C_O	20	pF	
Input/output capacitance	C_{IO}	20	pF	

DC Characteristics

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V

Item	Symbol	Min	Typ	Max	Unit	Conditions
Low-level input voltage	V_{IL}	0		0.8	V	Except PT pins
High-level input voltage	V_{IH1}	2.2		V_{DD}	V	Except PT pins and pins in Note 1
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	Pins in Note 1
Low-level output voltage	V_{OL1}			0.45	V	I_{OL} = 2.0 mA
	V_{OL2}			1.0	V	I_{OL} = 8.0 mA (Port PI pins)
High-level output voltage	V_{OH1}	V_{DD} − 1.0			V	I_{OH} = −1.0 mA
	V_{OH2}	V_{DD} − 0.5			V	I_{OH} = −100 μA
Input leakage current	I_{LI}			±10	μA	V_I = 0 to V_{DD}
Output leakage current	I_{LO}			±10	μA	V_O = 0 to V_{DD}
Pullup current	I_{IPT}		−150	−400	μA	V_I = 0 V; PT pins
V_{DD} power supply current	I_{DD1}		16	40	mA	Operating mode, f_{XX} = 12 MHz
	I_{DD2}		7	20	mA	HALT mode, f_{XX} = 12 MHz
Data retention voltage	V_{DDDR}	2.5		5.5	V	STOP mode
Data retention current	I_{DDDR}		2	20	μA	STOP mode; V_{DDDR} = 2.5 V
			5	50	μA	STOP mode; V_{DDDR} = 5 V ±10%

Notes:

(1) X1, X2, $\overline{\text{RESET}}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/$\underline{\text{INTP}}$2/CI, $P2_4$/INTP3, $P2_5$/$\underline{\text{INTP}}$4, $P2_6$/INTP5, $P2_7$/INTP6/SI, $P3_2$/SCK, $P3_3$/SO/SB0, and $\overline{\text{EA}}$ pins.

Read/Write Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF

Item	Symbol	Min	Max	Unit	Conditions
X1 input clock cycle time	t_{CYX}	82	250	ns	
Address setup time to ASTB ↓	t_{SAST}	52		ns	
Address hold time from ASTB ↓ (Note 2)	t_{HSTA}	25		ns	R_L = 5 kΩ, C_L = 50 pF
Address to $\overline{RD}$ ↓ delay time	t_{DAR}	129		ns	
Address float time from $\overline{RD}$ ↓	t_{FAR}	11		ns	
Address to data input time	t_{DAID}		228	ns	
ASTB ↓ to data input time	t_{DSTID}		181	ns	
$\overline{RD}$ ↓ to data input time	t_{DRID}		99	ns	
ASTB ↓ to $\overline{RD}$ ↓ delay time	t_{DSTR}	52		ns	
Data hold time from $\overline{RD}$ ↑	t_{HRID}	0		ns	
$\overline{RD}$ ↑ to address active time	t_{DRA}	124		ns	
$\overline{RD}$ ↑ to ASTB ↑ delay time	t_{DRST}	124		ns	
$\overline{RD}$ low-level width	t_{WRL}	124		ns	No wait states
ASTB high-level width	t_{WSTH}	52		ns	
Address to $\overline{WR}$ ↓ delay time	t_{DAW}	129		ns	
$\overline{ASTB}$ ↓ to data output time	t_{DSTOD}		142	ns	
$\overline{WR}$ ↓ to data output time	t_{DWOD}		60	ns	
ASTB ↓ to $\overline{WR}$ ↓ delay time	t_{DSTW1}	52		ns	
	t_{DSTW2}	129		ns	Refresh mode
Data setup time to $\overline{WR}$ ↑	t_{SODWR}	146		ns	
Data setup time to $\overline{WR}$ ↓ (Note 1)	t_{SODWF}	22		ns	Refresh mode
Data hold time from $\overline{WR}$ ↑ (Note 2)	t_{HWOD}	20		ns	
$\overline{WR}$ ↑ to ASTB ↑ delay time	t_{DWST}	42		ns	
$\overline{WR}$ low-level width	t_{WWL1}	196		ns	
	t_{WWL2}	114		ns	Refresh mode
Address to $\overline{WAIT}$ ↓ input time	t_{DAWT}		146	ns	
ASTB ↓ to $\overline{WAIT}$ ↓ input time	t_{DSTWT}		84	ns	
$\overline{WAIT}$ hold time from X1 ↓	t_{HWTX}	0		ns	
$\overline{WAIT}$ setup time to X1 ↑	t_{SWTX}*	0		ns	

Notes:

(1) When accessing a pseudostatic RAM (μPD4168, etc.) that clocks in data at the falling edge of $\overline{WR}$, use t_{SODWF} instead of t_{SODWR} as the data setup time.

(2) The hold time includes the time during which V_{OH} and V_{OL} are retained under the following load conditions: C_L = 100 pF and R_L = 2 kΩ.

4c

Serial Port Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF

Item	Symbol	Min	Max	Unit	Conditions
Serial clock cycle time	t_{CYSK}	1.0		μs	External clock input
		1.3		μs	Internal clock/16 output
		5.3		μs	Internal clock/64 output
Serial clock low-level width	t_{WSKL}	420		ns	External clock input
		556		ns	Internal clock/16 output
		2.5		μs	Internal clock/64 output
Serial clock high-level width	t_{WSKH}	420		ns	External clock input
		556		ns	Internal clock/16 output
		2.5		μs	Internal clock/64 output
SI, SB0 setup time to $\overline{SCK}$ ↑	t_{SSSK}	150		ns	
SI, SB0 hold time from $\overline{SCK}$ ↓	t_{HSSK}	400		ns	
SO/SB0 output delay time from $\overline{SCK}$ ↓	t_{DSBSK1}	0	300	ns	CMOS push-pull output (3-line serial I/O mode)
	t_{DSBSK2}	0	800	ns	Open-drain output (SBI mode), R_L = 1 kΩ
SB0 high, hold time from $\overline{SCK}$ ↑	t_{HSBSK}	4		t_{CYX}	SBI mode
SB0 low, setup time to $\overline{SCK}$ ↓	t_{SSBSK}	4		t_{CYX}	SBI mode
SB0 low-level width	t_{WSBL}	4		t_{CYX}	
SB0 high-level width	t_{WSBH}	4		t_{CYX}	
RxD setup time to $\overline{SCK}$ ↑	t_{SRXSK}	80		ns	
RxD hold time after $\overline{SCK}$ ↑	t_{HSKRX}	80		ns	
$\overline{SCK}$ ↓ to TxD delay time	t_{DSKTX}		210	ns	

Comparator Port Operation

Item	Symbol	Min	Max	Unit	Conditions
Comparison accuracy	V_{ACOMP}		100	mV	—
			100	mV	μPD78P224
Comparison time	t_{COMP}	128	256	t_{CYX}	
Sampling time	t_{SAMP}	62		t_{CYX}	
PT input voltage	V_{IPT}	0	V_{DD}	V	

Interrupt Timing Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V

Item	Symbol	Min	Max	Unit	Conditions
NMI low-level width	t_{WNIL}	10		μs	
NMI high-level width	t_{WNIH}	10		μs	
INTP0-INTP6 low-level width	t_{WITL}	24		t_{CYX}	
INTP0-INTP6 high-level width	t_{WITH}	24		t_{CYX}	
$\overline{RESET}$ low-level width	t_{WRSL}	10		μs	
$\overline{RESET}$ high-level width	t_{WRSH}	10		μs	

Data Retention Characteristics

Item	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.5		5.5	V	STOP mode
Data retention current	I_{DDDR}		2	20	μA	V_{DDDR} = 2.5 V
			5	50	μA	V_{DDDR} = 5 V ±10%
V_{DD} rise time	t_{RVD}	200			μs	
V_{DD} fall time	t_{FVD}	200			μs	

Data Retention Characteristics (cont)

Item	Symbol	Min	Typ	Max	Unit	Conditions
V_{DD} retention time (from STOP mode setting)	t_{HVD}	0			ms	
STOP release signal input time	t_{DREL}	0			ms	
Oscillation stabilization wait time	t_{WAIT}	30			ms	Crystal resonator
		5			ms	Ceramic resonator
Low-level input voltage	V_{IL}	0		0.1 V_{DDDR}	V	Specified pins (Note)
High-level input voltage	V_{IH}	0.9 V_{DDDR}		V_{DDDR}	V	Specified pins (Note)

Note: $\overline{RESET}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2/CI, $P2_4$/INTP3, $P2_5$/$\overline{INTP4}$, $P2_6$/INTP5, $P2_7$/INTP6/SI, $P3_2$/$\overline{SCK}$, $P3_3$/SO/SB0, and $\overline{EA}$ pins.

4c

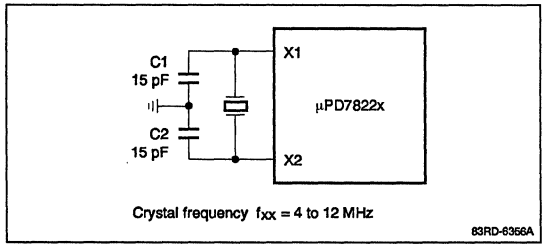
Timing Dependent on t_{CYX}

Item	Symbol	Calculation Formula (1, 2)	Min/Max	12 MHz	Unit
X1 input clock cycle time	t_{CYX}		Min	82	ns
Address setup time to ASTB ↓	t_{SAST}	$t_{CYX} - 30$	Min	52	ns
Address to $\overline{RD}$ ↓ delay time	t_{DAR}	$2t_{CYX} - 35$	Min	129	ns
Address float time from $\overline{RD}$ ↓	t_{FAR}	$t_{CYX}/2 - 30$	Min	11	ns
Address to data input time	t_{DAID}	$(4+2n)t_{CYX} - 100$	Max	228	ns
ASTB ↓ to data input time	t_{DSTID}	$(3+2n)t_{CYX} - 65$	Max	181	ns
$\overline{RD}$ ↓ to data input time	t_{DRID}	$(2+2n)t_{CYX} - 65$	Max	99	ns
ASTB ↓ to $\overline{RD}$ ↓ delay time	t_{DSTR}	$t_{CYX} - 30$	Min	52	ns
$\overline{RD}$ ↑ to address active time	t_{DRA}	$2t_{CYX} - 40$	Min	124	ns
$\overline{RD}$ ↑ to ASTB ↑ delay time	t_{DRST}	$2t_{CYX} - 40$	Min	124	ns
$\overline{RD}$ low-level width	t_{WRL}	$(2+2n)t_{CYX} - 40$	Min	124	ns
ASTB high-level width	t_{WSTH}	$t_{CYX} - 30$	Min	52	ns
Address to $\overline{WR}$ ↓ delay time	t_{DAW}	$2t_{CYX} - 35$	Min	129	ns
$\overline{ASTB}$ ↓ to data output time	t_{DSTOD}	$t_{CYX} + 60$	Max	142	ns
ASTB ↓ to $\overline{WR}$ ↓ delay time	t_{DSTW1}	$t_{CYX} - 30$	Min	52	ns
	t_{DSTW2}	$2t_{CYX} - 35$ (refresh mode)	Min	129	ns
Data setup time to $\overline{WR}$ ↑	t_{SODWR}	$(3+2n)t_{CYX} - 100$	Min	146	ns
Data setup time to $\overline{WR}$ ↓	t_{SODWF}	$t_{CYX} - 60$ (refresh mode)	Min	22	ns
$\overline{WR}$ ↑ to ASTB ↑ delay time	t_{DWST}	$t_{CYX} - 40$	Min	42	ns
$\overline{WR}$ low-level width	t_{WWL1}	$(3+2n)t_{CYX} - 50$	Min	196	ns
	t_{WWL2}	$(2+2n)t_{CYX} - 50$ (refresh mode)	Min	114	ns
Address to $\overline{WAIT}$ ↓ input time	t_{DAWT}	$3t_{CYX} - 100$	Max	146	ns
ASTB ↓ to $\overline{WAIT}$ ↓ input time	t_{DSTWT}	$2t_{CYX} - 80$	Max	84	ns

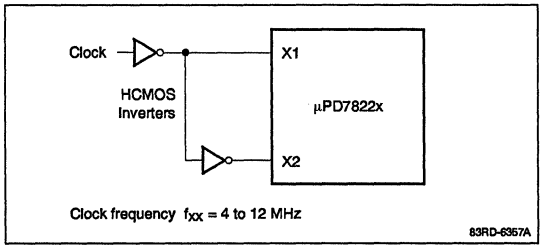
Note:

(1) n indicates the number of internal wait states.

Recommended Oscillator Circuit



Recommended External Clock Circuit



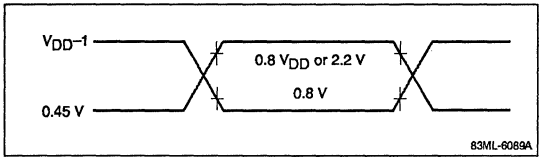
External Clock Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V

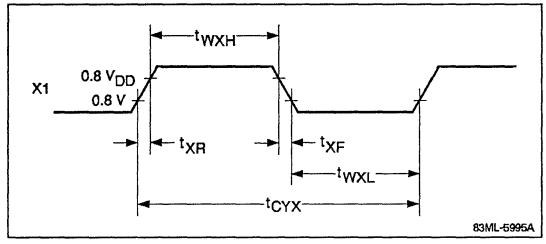
Item	Symbol	Min	Max	Unit	Conditions
X1 input low-level width	t_{WXL}	30	130	ns	
X1 input high-level width	t_{WXH}	30	130	ns	
X1 input rise time	t_{XR}	0	30	ns	
X1 input fall time	t_{XF}	0	30	ns	
X1 input clock cycle time	t_{CYX}	82	250	ns	

Timing Waveforms

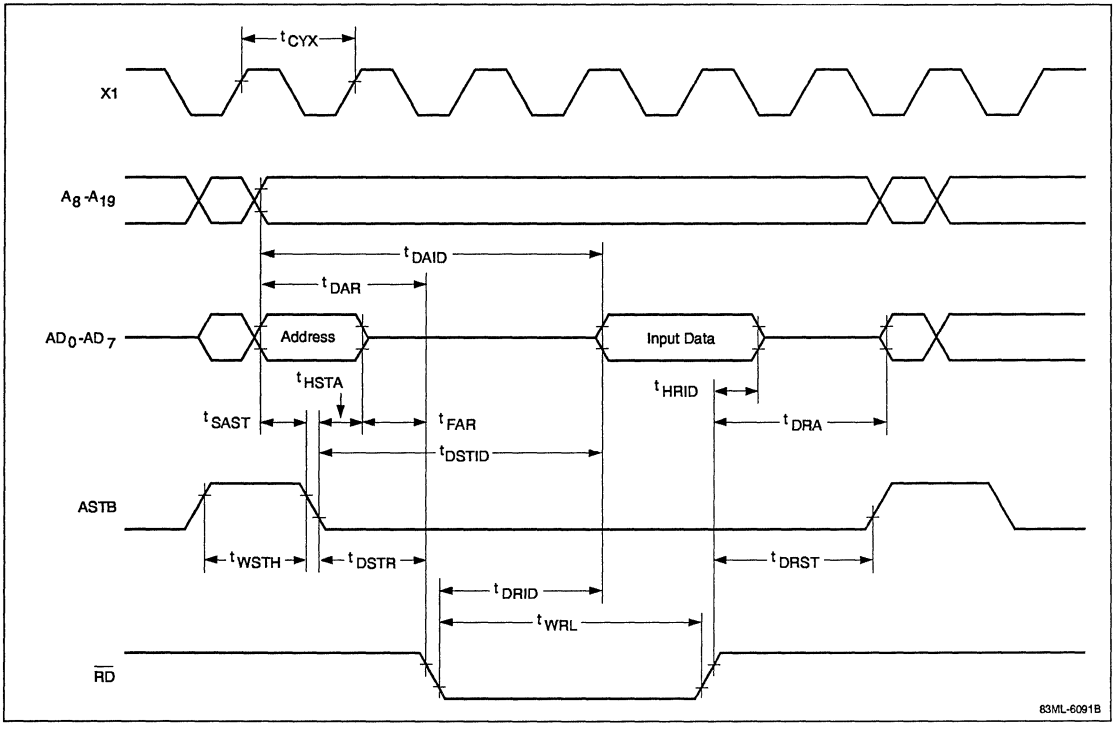
Voltage Thresholds for Timing Measurements



External Clock

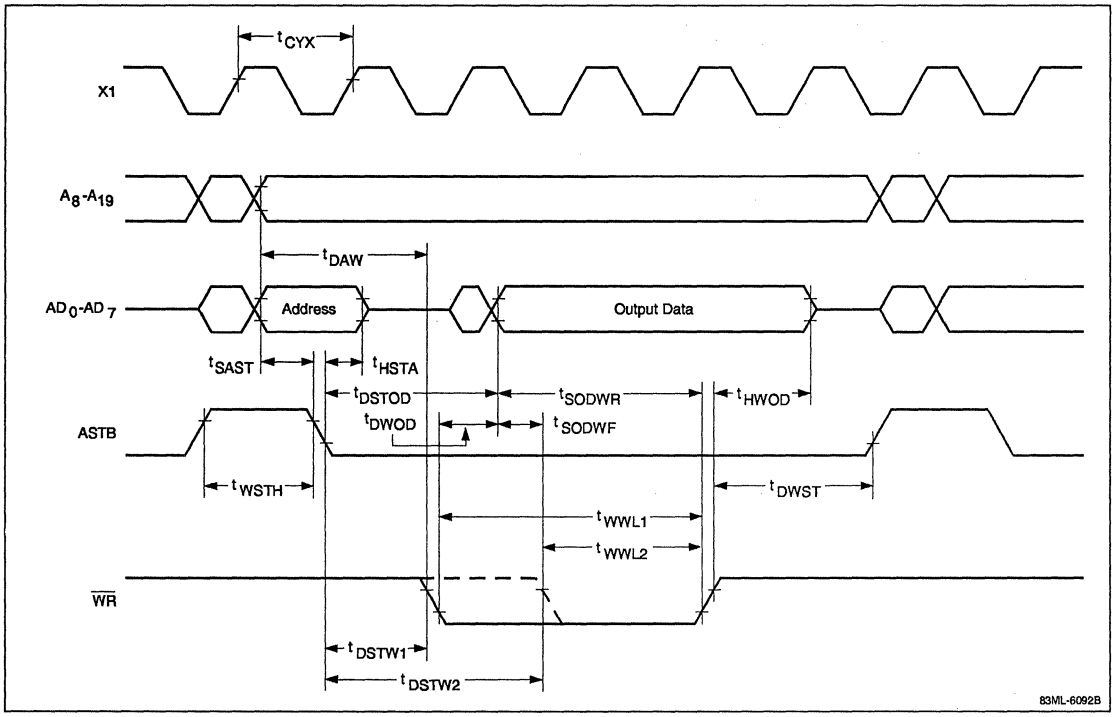


Read Operation

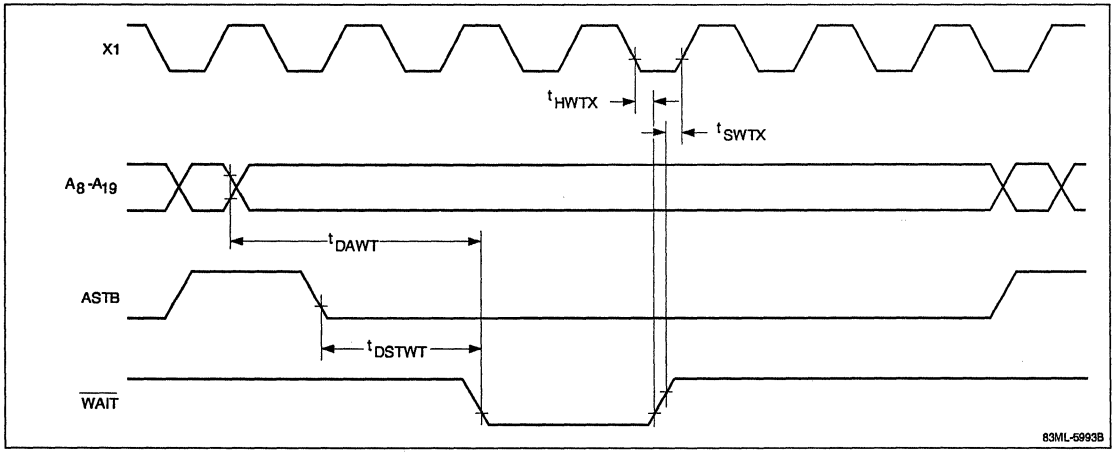


4c

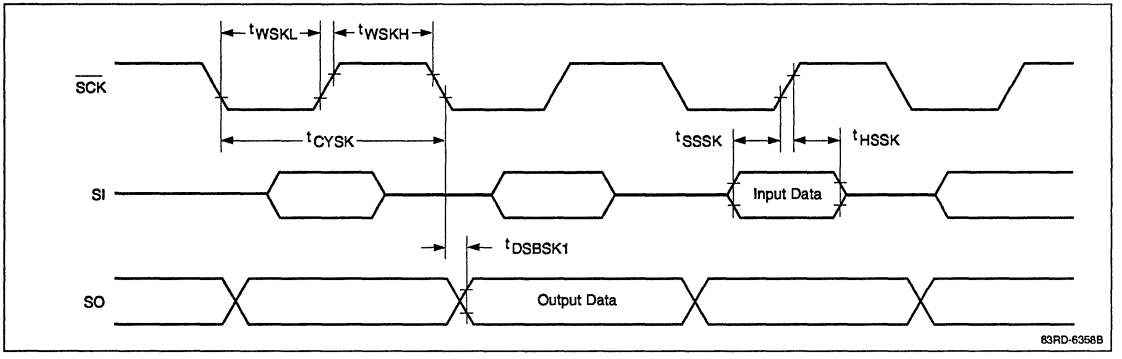
Write Operation



External WAIT Input

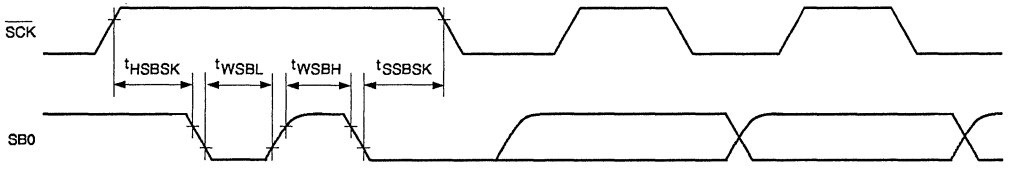


Clock-Synchronized Serial Interface; Three-Line I/O Mode

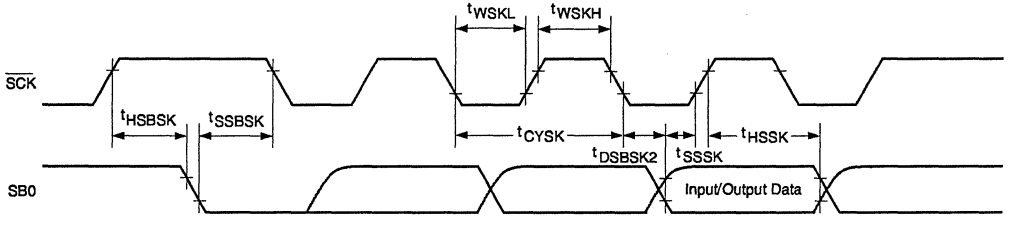


Clock-Synchronized Serial Interface; SBI Mode

Bus Release Signal Transfer Timing



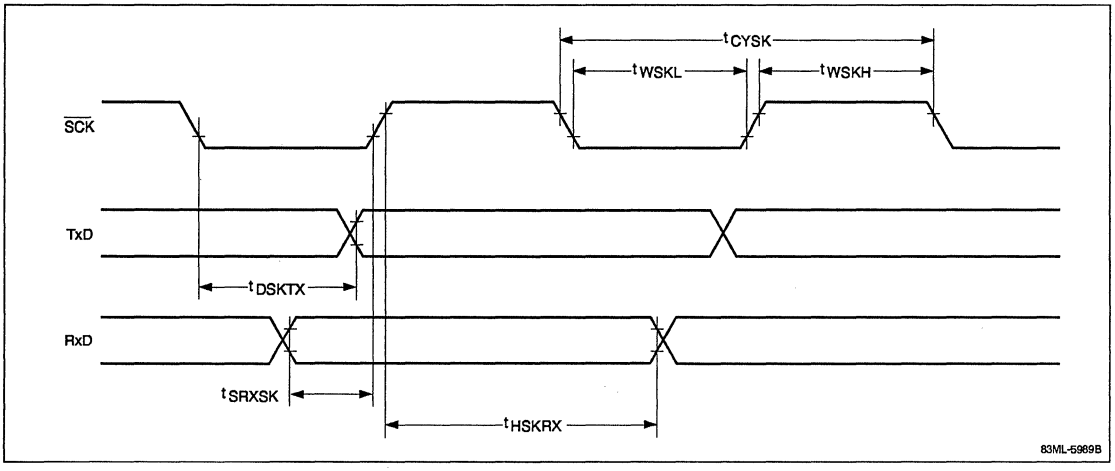
Command Signal Transfer Timing



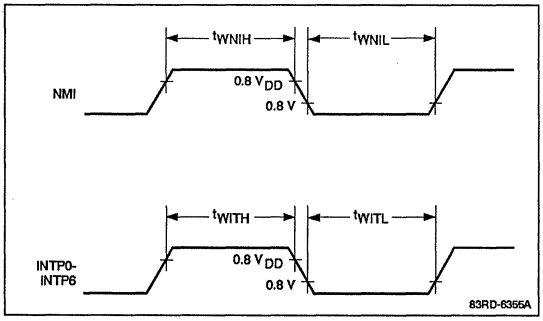
83RD-6354B

4c

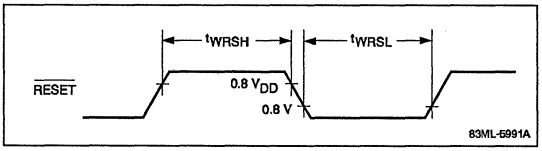
Asynchronous Mode



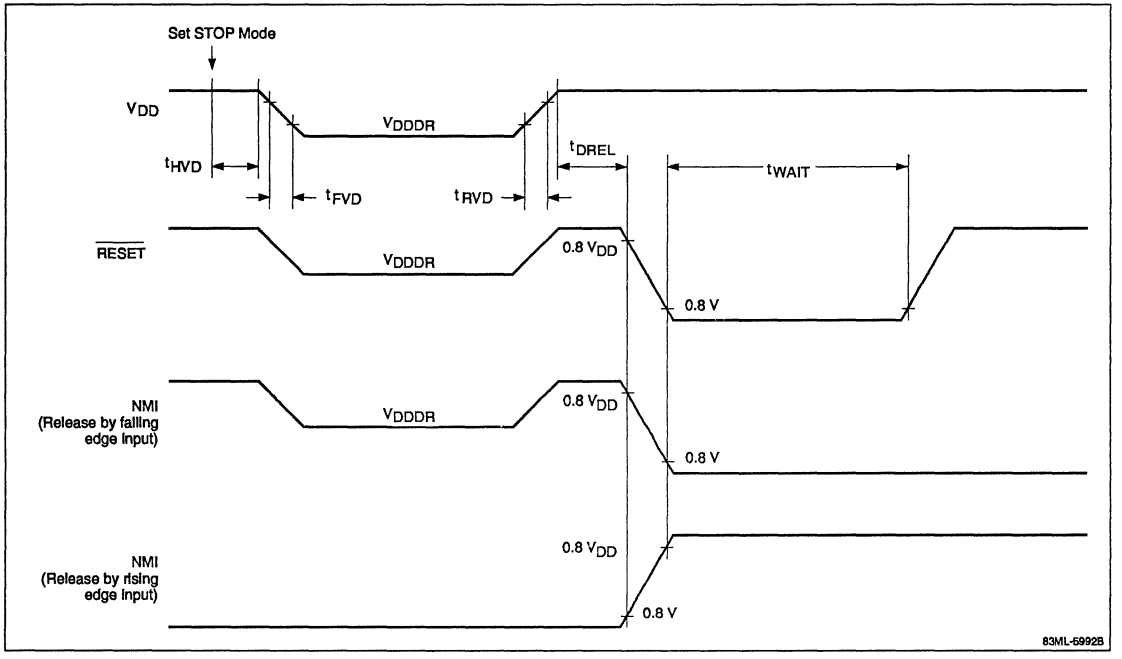
Interrupt Input



Reset Input



Data Retention Characteristics



4c

μPD78P224 PROGRAMMING

In the μPD78P224, the mask ROM of μPD78224 is replaced by a one-time programmable ROM (OTP ROM). The ROM is 16,384 x 8 bits and can be programmed using a general-purpose PROM writer with a μPD27C256A programming mode.

The PA-78P224GJ/L are the socket adaptors used for configuring the μPD78P224 to fit a standard PROM socket.

Refer to tables 3 through 6 and the PROM timing diagrams for special information applicable to PROM programming.

Table 3. Pin Functions During PROM Programming

Pin	Pin*	Function
$P0_0$ - $P0_7$	A_0 - A_7	Input pins for PROM write/verify operations
$P5_0/A_8$	A_8	Input pin for PROM write/verify operations
$P2_1$/INTP0	A_9	Input pin for PROM write/verify operations

Table 3. Pin Functions During PROM Programming (cont)

Pin	Pin*	Function
$P5_2/A_{10}$- $P5_6/A_{14}$	A_{10} - A_{14}	Input pins for PROM write/verify operations
$P4_0/AD_0$ - $P4_7/AD_7$	D_0 - D_7	Data pins for PROM operations
$P6_5/\overline{WR}$	$\overline{CE}$	Strobes data into the PROM
$P6_4/\overline{RD}$	$\overline{OE}$	Enables a data read from the PROM
$P2_0$/NMI	NMI	PROM programming mode is entered by applying a high voltage to this pin
$\overline{RESET}$	$\overline{RESET}$	PROM programming mode requires applying a low voltage to this pin
$\overline{EA}$	V_{PP}	High voltage applied to this pin for program write/verify
V_{DD}	V_{DD}	Positive power supply pin
V_{SS}	V_{SS}	Ground

* Pin name in PROM programming mode.

Table 4. Summary of Operation Modes for PROM Programming

Mode	NMI	$\overline{RESET}$	$\overline{CE}$	$\overline{OE}$	V_{PP}	V_{DD}	D_0 - D_7
Program write	+12.5 V	L	L	H	+12.5 V	+6 V	Data input
Program verify	+12.5 V	L	H	L	+12.5 V	+6 V	Data output
Program inhibit	+12.5 V	L	H	H	+12.5 V	+6 V	High Z
Read out	+12.5 V	L	L	L	+5 V	+5 V	Data output
Output disable	+12.5 V	L	L	H	+5 V	+5 V	High Z
Standby	+12.5 V	L	H	L/H	+5 V	+5 V	High Z

Note: When +12.5 V is applied to V_{PP} and +6 V to V_{DD}, both $\overline{CE}$ and $\overline{OE}$ cannot be set to low level (L) simultaneously.

DC Programming Characteristics

T_A = 25 ±5°C; V_{IP} = 12.5 ±0.5 V applied to NMI pin; V_{SS} = 0 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH}	V_{IH}	2.4		V_{DDP}+0.3	V	
Low-level input voltage	V_{IL}	V_{IL}	−0.3		0.8	V	
Input leakage current	I_{LIP}	I_{LI}			10	μA	$0 \le V_I \le V_{DDP}$
High-level output voltage	V_{OH1}	V_{OH1}	2.4			V	I_{OH} = −400 μA
	V_{OH2}	V_{OH2}	V_{DD} − 0.7			V	I_{OH} = −100 μA
Low-level output voltage	V_{OL}	V_{OL}			0.45	V	I_{OL} = 2.1 mA
Output leakage current	I_{LO}				10	μA	$0 \le V_O \le V_{DDP}$, $\overline{OE}$ = V_{IH}
NMI pin high-voltage input current	I_{IP}				±10	μA	
V_{DDP} power voltage	V_{DDP}	V_{CC}	5.75	6.0	6.25	V	Program memory write mode
			4.5	5.0	5.5	V	Program memory read mode
V_{PP} power voltage	V_{PP}	V_{PP}	12.2	12.5	12.8	V	Program memory write mode
				V_{PP} = V_{DDP}		V	Program memory read mode
V_{DDP} power current	I_{DD}	I_{CC}		5	30	mA	Program memory write mode
				5	30	mA	Program memory read mode $\overline{CE}$ = V_{IL}, V_I = V_{IH}
V_{PP} power current	I_{PP}	I_{PP}		5	30	mA	Program memory write mode $\overline{CE}$ = V_{IL}, $\overline{OE}$ = V_{IH}
				1	100	μA	Program memory read mode

* Corresponding symbols of the μPD27C256A.

AC Programming Characteristics (Write Mode)

T_A = 25 ±5°C; V_{IP} = 12.5 ±0.5 V applied to NMI pin; V_{SS} = 0 V; V_{DD} = 6 ±0.25 V; V_{PP} = 12.5 ±0.3 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Conditions
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}	2			μs	
Data input to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}	2			μs	
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}	2			μs	
Address hold time from $\overline{CE}$ ↑	t_{HCA}	t_{AH}	2			μs	
Input data hold time from $\overline{CE}$ ↑	t_{HCID}	t_{DH}	2			μs	
Output data hold time from $\overline{OE}$ ↑	t_{HOOD}	t_{DF}	0		130	ns	
V_{PP} setup time to $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}	1			ms	
V_{DDP} setup time to $\overline{CE}$ ↓	t_{SVDC}	t_{VCS}	1			ms	
Initial program pulse width	t_{WL1}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{WL2}	t_{OPW}	2.85		78.75	ms	
NMI high-voltage input setup time to $\overline{CE}$ ↓	t_{SPC}		2			μs	
Address to data output time	t_{DAOD}	t_{ACC}			200	ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}
$\overline{CE}$ ↓ to data output time	t_{DCOD}	t_{CE}			200	ns	$\overline{OE}$ = V_{IL}
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			75	ns	$\overline{CE}$ = V_{IL}
Data hold time from $\overline{OE}$ ↑	t_{HCOD}	t_{DF}	0		60	ns	$\overline{CE}$ = V_{IL}
Data hold time from address	t_{HAOD}	t_{OH}	0			ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}

* Corresponding symbols of the μPD27C256A.

4c

PROM Write Procedure

(1) Connect the $\overline{\text{RESET}}$ pin to a low level, and apply +12.5 V to the NMI pin.

(2) Apply +6 V to the V_{DD} pin and +12.5 V to the V_{PP} pin.

(3) Provide the initial address.

(4) Provide write data.

(5) Provide 1-ms program pulse (active low) to the $\overline{\text{CE}}$ pin.

(6) This data is now verified with a pulse (active low) to the $\overline{\text{OE}}$ pin. If the data has been written, proceed to step 8; if not, repeat steps 4 to 6. If the data cannot be correctly written after 25 attempts, go to step 7.

(7) Classify as defective and stop write operation.

(8) Provide write data and supply program pulse (for additional writing) for 3 ms times the number of writes performed in steps 5.

(9) Increment the address.

(10) Repeat steps 4 to 9 until the end address.

PROM Read Procedure

(1) Fix the $\overline{\text{RESET}}$ pin to a low level, and apply +12.5 V to the NMI pin.

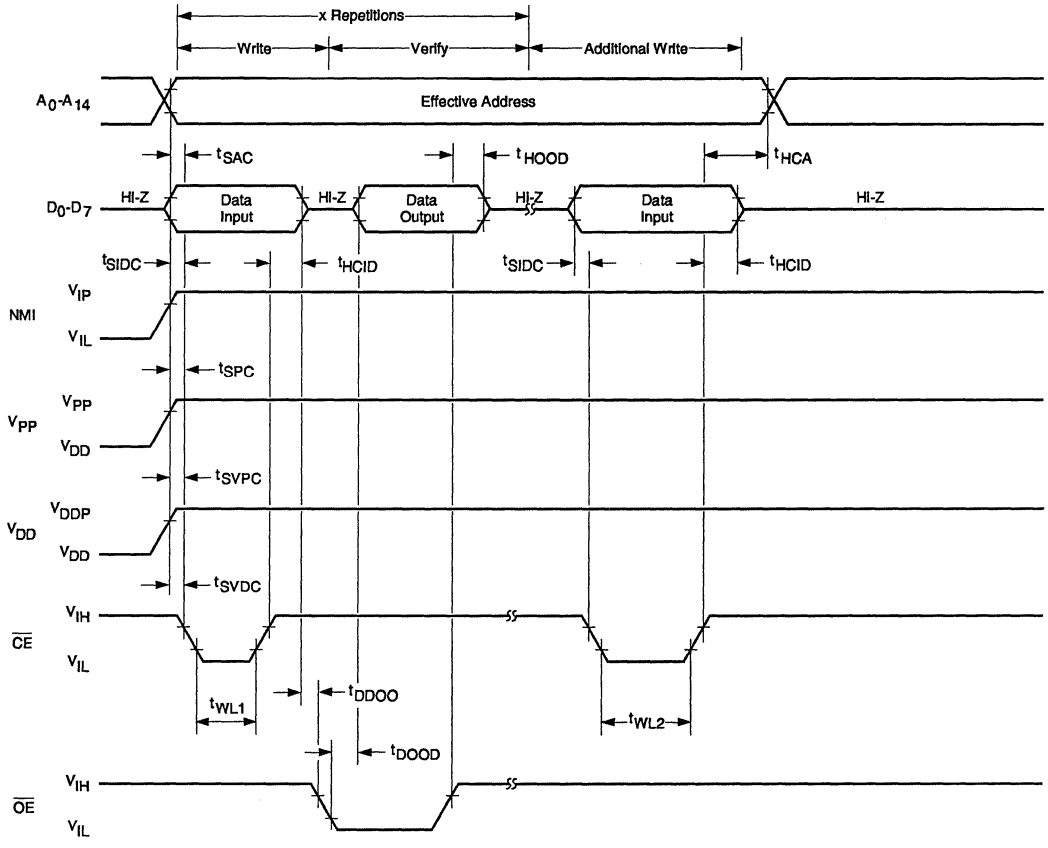
(2) Input the address of the data to be read to pins A_0 - A_{14}.

(3) Read mode is entered with a pulse (active low) on both the $\overline{\text{CE}}$ and $\overline{\text{OE}}$ pins.

(4) Data is output to the D_0 to D_7 Pins.

PROM Timing Diagrams

PROM Write Mode



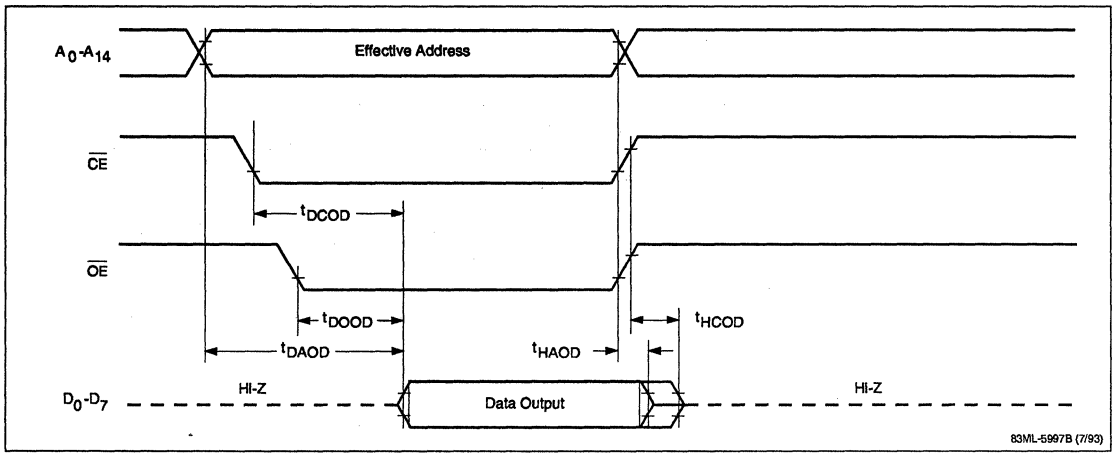
Notes:
(1) V_{DD} must be applied before applying V_{PP}. It should be removed after removing V_{PP}.
(2) V_{PP} must not exceed +13V, including overshoot.

83ML-5996B

4c

PROM Timing Diagrams

PROM Read Mode



NEC
NEC Electronics Inc.

μPD78238 Family (μPD78233/234/237/238/P238) 8-Bit, K-Series Microcontrollers With A/D and D/A Converters, Real-Time Output Ports

June 1993

Description

The μPD78233, μPD78234, μPD78237, μPD78238, and μPD78P238 are members of the K-Series® of microcontrollers and are designed for real-time embedded control applications. These 8-bit, single-chip microcontrollers have a minimum instruction time of 333 ns at 12 MHz (500 ns for the μPD78233/237). They feature 8-bit hardware multiply and divide instructions, four banks of main registers, an advanced interrupt handling facility, a powerful set of memory mapped on-chip peripherals, and the ability to address up to 1M bytes of external data memory. On board memory includes 640 or 1024 bytes of RAM, 16K or 32K bytes of mask ROM, or 32K bytes of UV EPROM or one-time programmable (OTP) ROM.

The advanced interrupt handling facility provides two levels of programmable hardware priority control and two separate methods of servicing interrupt requests: vectored and macro service. The macro service facility reduces the overhead involved in servicing peripheral interrupts by transferring data between the memory-mapped special function registers (SFRs) and memory without the use of time consuming interrupt service routines. In addition, the macro service facility can be initialized to automatically alter timer compare register values or to repeatedly output a prespecified pattern at a fixed or variable rate. By using macro service to control the real-time output ports, the μPD78238 family can easily and accurately drive two independent stepper motors.

The combination of the macro service facility, four banks of main registers, extended data memory address space, and powerful on-chip peripherals makes these devices ideal for applications in office automation, communication, HVAC, and industrial control.

Features

- Complete single-chip microcontroller
  - 8-bit ALU
  - Program memory (ROM)
    - μPD78233/237: ROMless
    - μPD78234: 16K bytes
    - μPD78238/P238: 32K bytes
  - Data memory (RAM)
    - μPD78233/234: 640 bytes
    - μPD78237/238/P238: 1024 bytes
- Powerful instruction set
  - 8-bit unsigned multiply and divide
  - 16-bit arithmetic instructions
  - 1-bit and 8-bit logic instructions
- Minimum instruction time
  - 333 ns at 12 MHz (μPD78234/238/P238)
  - 500 ns at 12 MHz (μPD78233/237)
- Memory expansion
  - 8085 bus-compatible
  - 64K program address space
  - 1M data address space
- Large I/O capacity
  - Up to 64 I/O port lines on μPD78234/238/P238
  - Up to 46 I/O port lines on μPD78233/237
  - Software programmable pullup resistors
- Memory-mapped on-chip peripherals (special function registers)
- Timer/counter unit
  - 16-bit timer 0:
    - Two 16-bit compare registers
    - One 16-bit capture register
    - One external interrupt/capture line
  - 8-bit timer 1:
    - One 8-bit compare register
    - One 8-bit capture/compare register
    - One external interrupt/capture line
  - 8-bit timer/counter 2:
    - Two 8-bit compare registers
    - One 8-bit capture register
    - One external interrupt/capture line
    - One external event counter line
  - 8-bit timer 3:
    - One 8-bit compare register
- Pulse-width modulated (PWM) outputs
  - Two 12-bit precision hardware controlled
  - Four 8-bit precision timer controlled
- Two 4-bit (or one 8-bit) real-time output ports
- Eight-channel 8-bit A/D converter
- Two-channel 8-bit D/A converter
- Programmable priority interrupt controller (two levels)
- Two methods of interrupt service
  - Vectored interrupts
  - Macro service mode with choice of three different types

K-series is a registered trademark of NEC Electronics, Inc.

Features (cont)

- Two-channel serial communication interface
  - — Asynchronous serial interface (UART)
    Dedicated baud rate generator
  - — Clock-synchronized interface
    Full-duplex, three-wire mode
    NEC serial bus interface (SBI) mode
- Refresh output for pseudostatic RAM
- STOP and HALT standby functions
- 5-volt CMOS technology

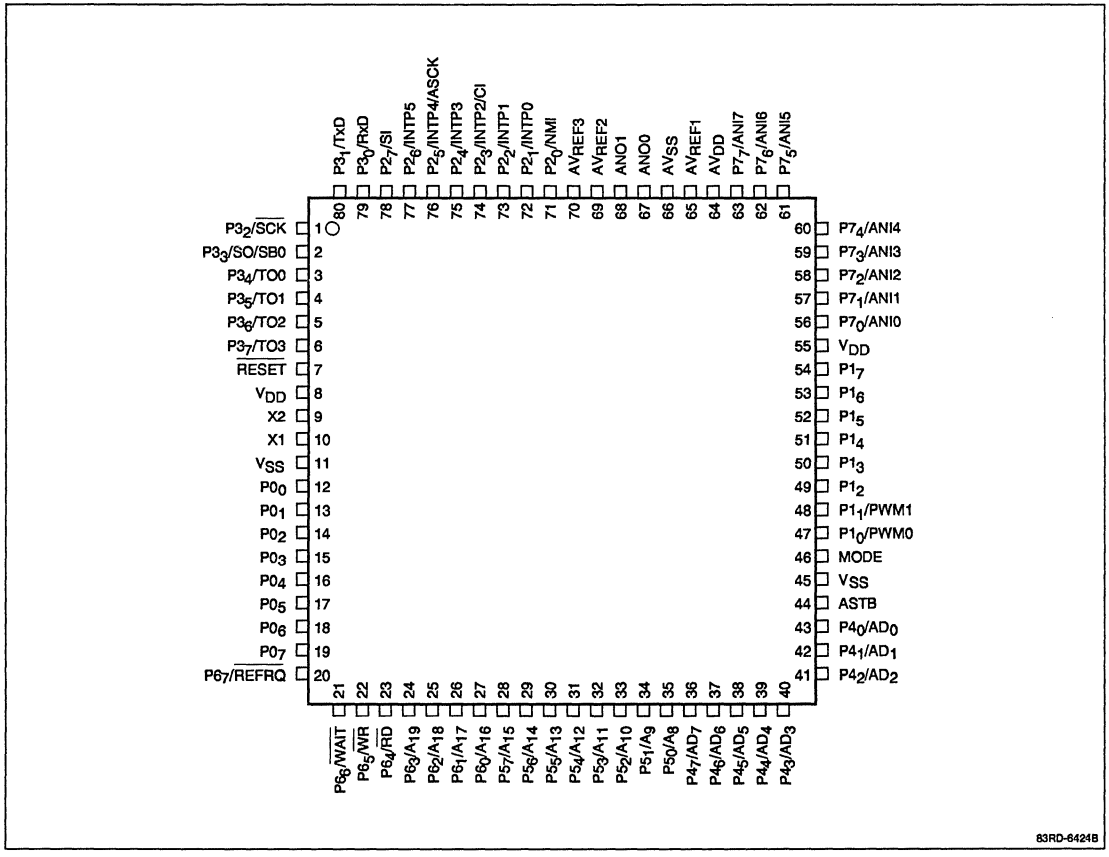
Ordering Information

Part Number	ROM	Package	Package Drawing
μPD78233GC	ROMless	80-pin plastic QFP	S80GC-65-3B9-1
μPD78234GC-xxx	16K mask ROM		
μPD78237GC	ROMless		
μPD78238GC-xxx	32K mask ROM		
μPD78P238GC	32K OTP ROM		
μPD78233GJ	ROMless	94-pin plastic QFP	S94GJ-80-5BG-1
μPD78234GJ-xxx	16K mask ROM		
μPD78237GJ	ROMless		
μPD78238GJ-xxx	32K mask ROM		
μPD78P238GJ	32K OTP ROM		
μPD78233LQ	ROMless	84-pin PLCC	P84L-50A3-1
μPD78234LQ-xxx	16K mask ROM		
μPD78237LQ	ROMless		
μPD78238LQ-xxx	32K mask ROM		
μPD78P238LQ	32K OTP ROM		
μPD78P238KF	32K UV EPROM	94-pin ceramic LCC with window	X94KW-80A

Note: xxx indicates ROM code suffix.

Pin Configurations

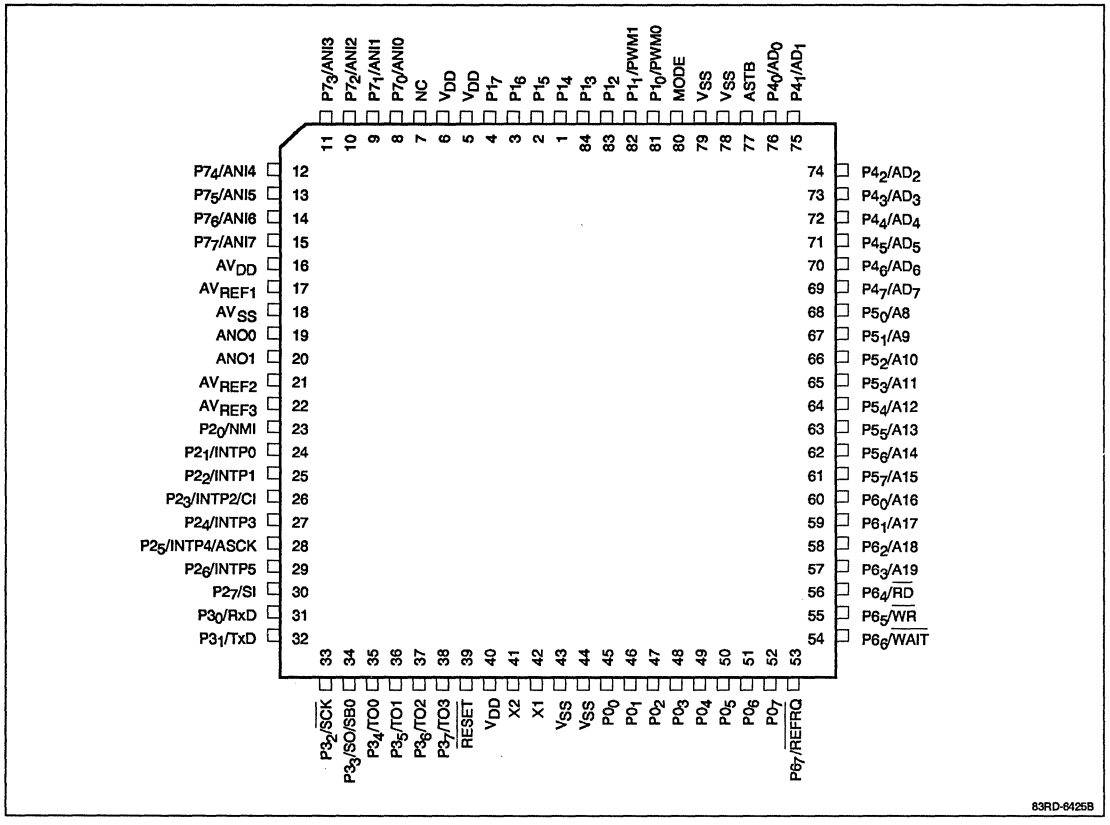
80-Pin Plastic QFP



83RD-6424B

Pin Configurations (cont)

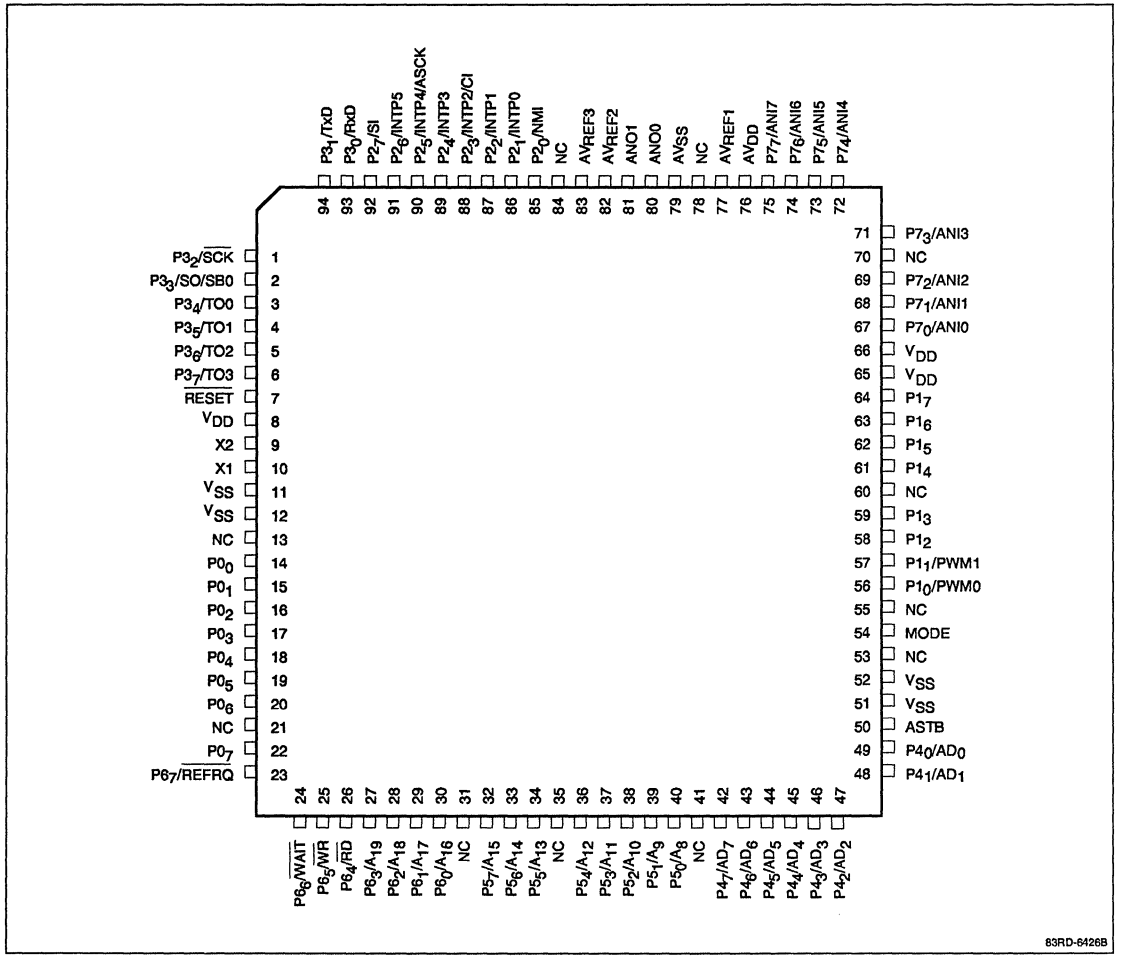
84-Pin PLCC (Plastic Leaded Chip Carrier)



83RD-6426B

Pin Configurations (cont)

94-Pin Plastic QFP and Ceramic LCC with Window



83RD-6426B

Pin Functions; Normal Operating Mode

Symbol	First Function	Symbol	Second Function
$P0_0$ - $P0_7$	Port 0; 8-bit tristate output port/real time output port		
$P1_0$ $P1_1$	Port 1; 8-bit, bit-selectable tristate input/output port	PWM0 PWM1	Pulse-width modulated outputs
$P1_2$ - $P1_7$		—	
$P2_0$	Port 2; 8-bit input port	NMI	External nonmaskable interrupt
$P2_1$ $P2_2$		INTP0 INTP1	Maskable external interrupts
$P2_3$		INTP2 CI	Maskable external interrupt External clock input to timer/counter 2
$P2_4$		INTP3	Maskable external interrupt
$P2_5$		INTP4 ASCK	Maskable external interrupt Asynchronous serial clock input
$P2_6$		INTP5	Maskable external interrupt
$P2_7$		SI	Serial data input for three-wire serial I/O mode
$P3_0$	Port 3; 8-bit, bit-selectable tristate input/output port	RxD	Asynchronous serial receive data input
$P3_1$		TxD	Asynchronous serial transmit data input
$P3_2$		$\overline{SCK}$	Serial shift clock input/output
$P3_3$		SO SBO	Serial data output for three-wire serial I/O mode I/O bus for NEC serial bus interface (SBI)
$P3_4$ - $P3_7$		TO0 - TO3	Timers T0 to T3 outputs
$P4_0$ - $P4_7$	Port 4; 8-bit tristate input/output port	AD_0 - AD_7	Low-order 8-bit multiplexed address/data bus
$P5_0$ - $P5_7$	Port 5; 8-bit, bit-selectable tristate input/output port	A_8 - A_{15}	High-order 8-bit address bus
$P6_0$ - $P6_3$	Port 6; 4-bit output port	A_{16} - A_{19}	Extended memory address bus
$P6_4$	Port 6; 4-bit, bit-selectable tristate input/output port	$\overline{RD}$	External memory read strobe
$P6_5$		$\overline{WR}$	External memory write strobe
$P6_6$		$\overline{WAIT}$	External memory wait signal input
$P6_7$		$\overline{REFRQ}$	Refresh pulse output used by external pseudostatic memory
$P7_0$ - $P7_7$	Port 7; 8-bit input port	ANI0 - ANI7	Analog voltage input to A/D converter
ANO0, ANO1	Analog voltage output from D/A converter		
ASTB	Address strobe output used to latch the low-order 8 address for external memory		
$\overline{RESET}$	External system reset input		
MODE	Internal ROM or external memory control signal input. A low-level input selects internal ROM. A high-level input selects external memory. The μPD78234/238 can be used in ROMless mode by setting the MODE pin high. However, the μPD78P238 cannot be used in ROMless mode and its MODE pin must only be set low.		
X1	Crystal/ceramic resonator connection or external clock input		
X2	Crystal/ceramic resonator connection or inverse of external clock		
AV_{REF1}	A/D converter reference voltage		

Pin Functions; Normal Operating Mode (cont)

Symbol	First Function	Symbol	Second Function
AV_{REF2}, AV_{REF3}	D/A converter reference voltages		
AV_{DD}	A/D converter power supply		
AV_{SS}	A/D converter ground		
V_{DD}	+5 volt power supply input		
V_{SS}	Power supply ground		
NC	No connection		

FUNCTIONAL DESCRIPTION

Central Processing Unit (CPU)

The μPD78238 family CPU features 8- and 16-bit arithmetic including an 8 x 8-bit unsigned multiply and 16 x 8-bit unsigned divide (producing a 16-bit quotient and an 8-bit remainder). The multiply executes in 3.67 μs and the divide in 12.36 μs at 12 MHz (4.00 and 12.69 μs respectively for μPD78233/237).

A CALLT vector table and a CALLF program area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction (CALLT) can access up to 32 subroutines through the addresses contained in the CALLT vector table. A 2-byte call instruction (CALLF) can access any routine beginning at a specific address in the CALLF area.

The internal system clock (f_{CLK}) is generated by dividing the oscillator frequency by two. Therefore, at the maximum oscillator frequency of 12 MHz, the internal system clock is 6 MHz. The minimum instruction execution time for an instruction fetched from internal ROM is 333 ns (500 ns when fetched from external memory).

Memory Space

The μPD78238 family has a 1M byte address space (see figure 1). The first 64K bytes of this address space (00000H-0FFFFH) can be used as both program and data memory. The remaining 960K bytes of this address space (10000H-FFFFFH) can only be used as data memory and is known as expanded memory.

Block Diagram

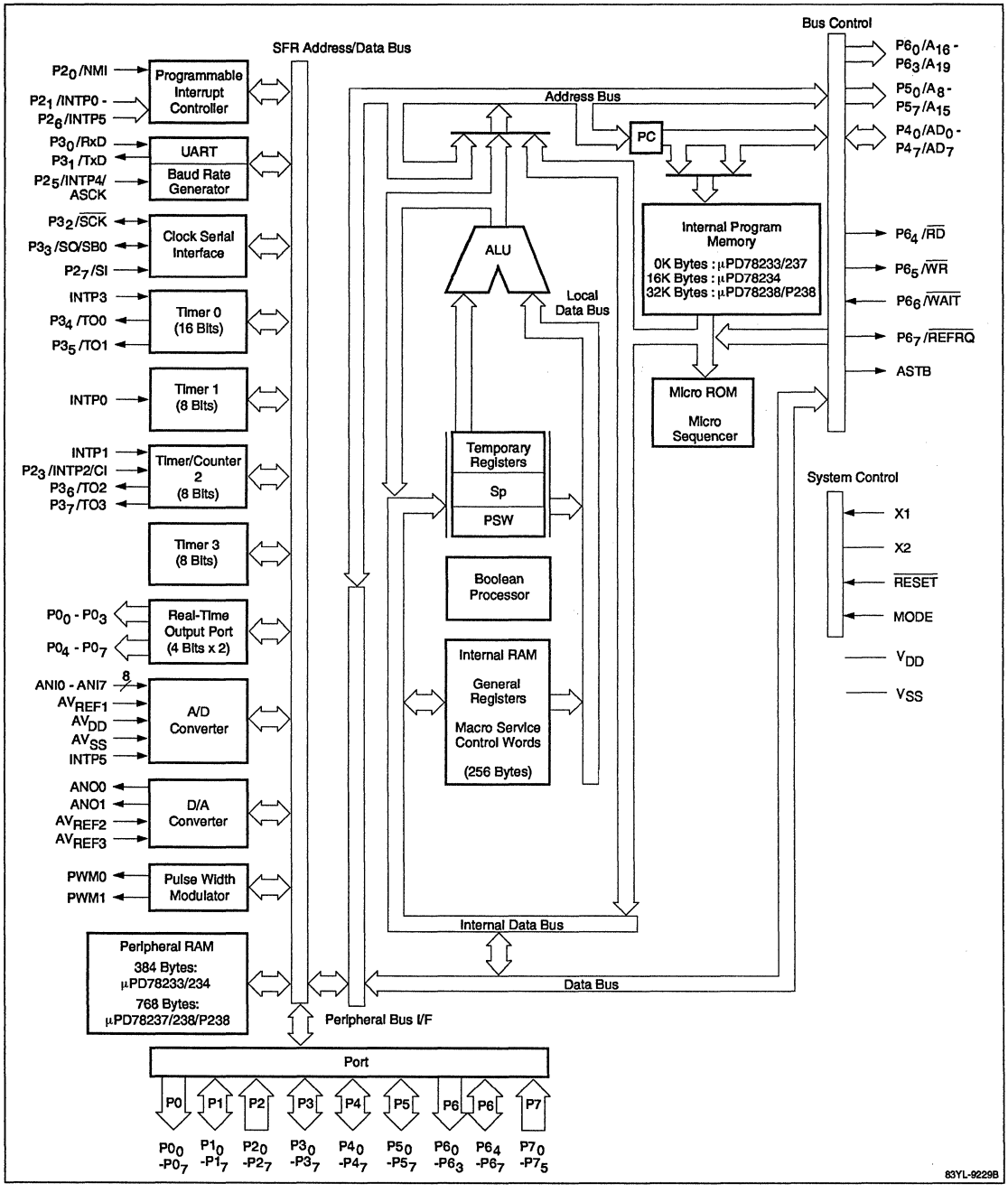
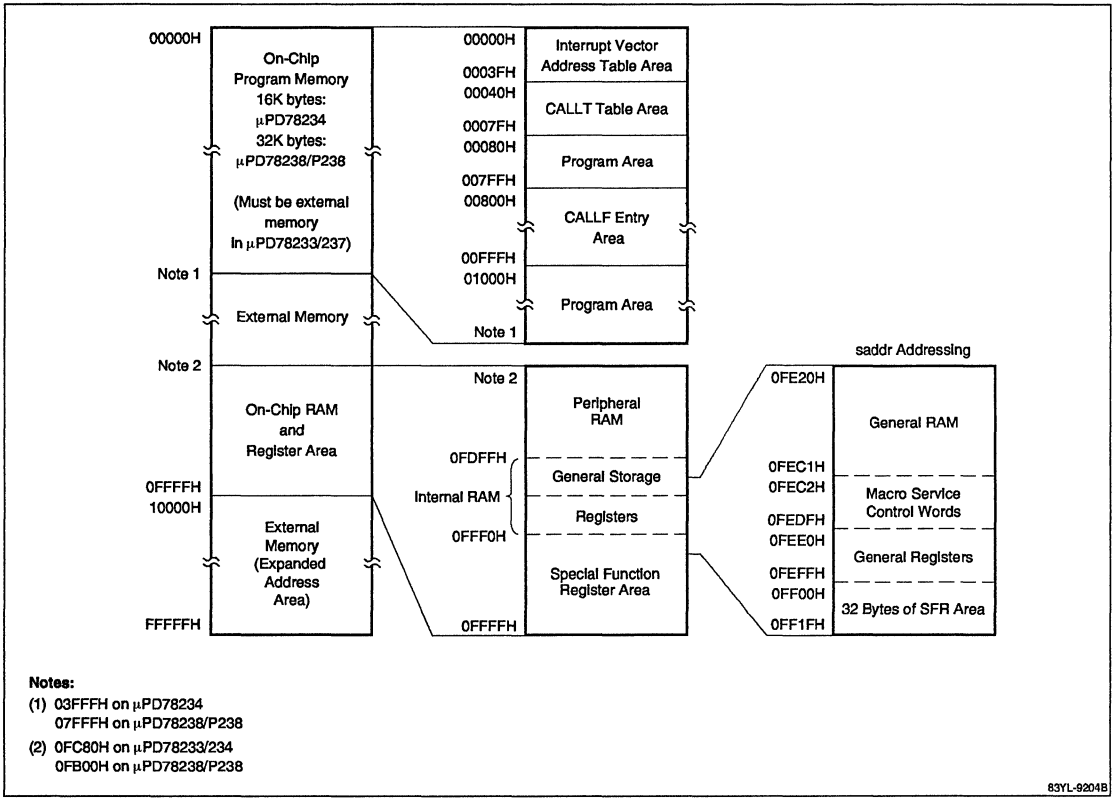


Figure 1. Memory Map



External Memory

The μPD78238 family has an 8-bit wide external data bus and a 16-bit wide external address bus (20-bit wide if expanded memory is enabled). The low-order 8 bits of the address bus are multiplexed to provide the 8-bit data bus and are supplied by I/O port 4. The high-order address bits of the 16-bit address bus are taken from port 5. If expanded memory is enabled, the expanded address nibble is provided by $P6_0$ to $P6_3$. Address latch, read, and write strobes are also provided.

The memory expansion mode register (MM) is used to enable external memory, to specify up to two additional wait states or the use of the $\overline{WAIT}$ input pin for the first 64K bytes of memory, and to enable the high-speed internal ROM fetch. Ports 4, 5, and 6 are available as general purpose I/O ports when only internal ROM is used and no external program or data space is required.

Expanded Data Memory

The MM register is also used to enable the external expanded data memory space, addresses 10000H to FFFFFH. When the expanded data memory is enabled, the entire 1M byte address space is divided into 16 banks of 64K bytes each. The low-order 4-bits of the P6 or the PM6 registers are used as bank selection registers to supply the address information to A_{16} to A_{19}. Data can easily be transferred from one memory bank to another by using the appropriate instructions. Address lines A_{16} to A_{19} are only active when an instruction that uses expanded addressing is being executed. A programmable wait control register (PW) allows the programmer to specify up to two additional wait states or the use of the $\overline{WAIT}$ input pin for expanded data memory space.

On-Chip RAM

The μPD78237/238 have a total of 1024 bytes of on-chip RAM (640 bytes in the μPD78233/234).

The μPD78P238 also contains 1024 bytes of on-chip RAM. By using the memory size select (IMS) register, the μPD78P238 can be programmed to emulate either a μPD78234 device with 640 bytes of on-chip RAM or a μPD78238 with 1024 bytes. The programming of this register is transparent to the ROM-based device, allowing easy transfer of code to a ROM-based device.

The upper 256-byte area (FE00H-FEFFH) features high-speed access and is known as "Internal RAM." The remainder (FB00H-FDFFH and FC80H-FDFFH in the μPD78233/234) is accessed at the same speed as external memory and is known as "Peripheral RAM." The general register banks and the macro service control words are stored in Internal RAM. The remainder of Internal RAM and any unused register bank locations are available for general storage.

On-Chip Program Memory

The μPD78234/238 contain 16K and 32K bytes of internal ROM respectively. The μPD78P238 contains 32K bytes of UV EPROM or one-time programmable ROM. By using the IMS register, the μPD78P238 can be programmed to emulate a μPD78234 device with 16K bytes of internal PROM or a μPD78238 with 32K bytes. This programming is transparent to the ROM-based device, allowing easy transfer of code to a ROM-based device. Instructions from on-chip program memory can be fetched at high speed or at the same rate as from external memory. The μPD78233 and the μPD78237 do not have on-chip program memory.

CPU Control Registers

Program Counter. The program counter is a 16-bit binary counter register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 0001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

Program Status Word. The program status word (PSW) is an 8-bit register that contains flags that are set or reset depending on the results of an instruction.

This register can be written to or read from 8 bits at a time. The individual flags can also be manipulated on a bit-by-bit basis. The assignment of PSW bits follows.

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

CY	Carry flag
ISP	Interrupt priority status flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

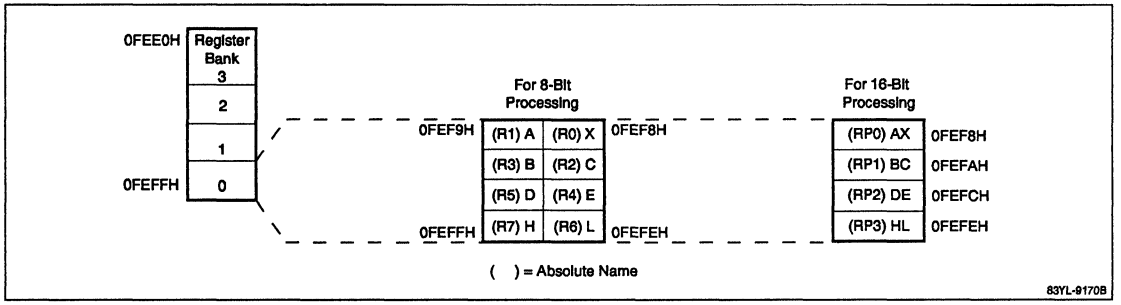
General Registers

The general-purpose registers (figure 2) consist of four banks of registers located at addresses FEE0H to FEFFH in Internal RAM. Each bank consists of eight 8-bit general registers that can also be used in pairs to function as four 16-bit registers. Two bits in the PSW (RBS0 and RBS1) specify which of the register banks is active. The bits are set under program control. Registers have both functional names (like A, X, B, C for 8-bit registers and AX, BC for 16-bit registers) and absolute names (like R1, R0, R3, R2 for 8-bit registers and RP0, RP1 for 16-bit registers). Each instruction determines whether a register is referred to by its functional or absolute name and whether it is 8 or 16 bits.

Addressing

The μPD78238 family features 1-byte addressing of both the special function registers and the portion of on-chip RAM from FE20H to FEFFH. The 1-byte sfr addressing accesses the entire SFR area, while the 1-byte saddr addressing accesses the first 32 bytes of the SFR area and 224 bytes of internal RAM. Sixteen-bit SFRs and words of memory in these areas can be addressed by 1-byte saddrp addressing, which is valid for even addresses only. Since many instructions use 1-byte addressing, access to these locations is almost as fast and versatile as access to the general registers. There are seven addressing modes for data in main memory: direct, register, register indirect with autoincrement and decrement, saddr, SFR, based, and indexed. There are also both 8-bit and 16-bit immediate operands.

Figure 2. General Registers



Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and the CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by 1-byte sfr addressing. They are either 8 or 16 bits as required, and many of the 8-bit registers are capable of single-bit access as well. Locations FFD0H through FFDFH are known as the external SFR area. Registers in external circuitry interfaced and mapped to these addresses can be addressed with SFR addressing. Table 1 is a list of the special function registers.

4d

Table 1. Special Function Registers

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
0FF00H	Port 0	P0	R/W	x	x	—	Undefined
0FF01H	Port 1	P1	R/W	x	x	—	Undefined
0FF02H	Port 2	P2	R	x	x	—	Undefined
0FF03H	Port 3	P3	R/W	x	x	—	Undefined
0FF04H	Port 4	P4	R/W	x	x	—	Undefined
0FF05H	Port 5	P5	R/W	x	x	—	Undefined
0FF06H	Port 6	P6	R/W	x	x	—	x0H
0FF07H	Port 7	P7	R	x	x	—	Undefined
0FF0AH	Port 0 buffer register (low)	P0L	R/W	x	x	—	Undefined
0FF0BH	Port 0 buffer register (high)	P0H	R/W	x	x	—	Undefined
0FF0CH	Real-time output port control register	RTPC	R/W	x	x	—	00H
0FF10H-0FF11H	16-bit compare register 0 (16-bit timer 0)	CR00	R/W	—	—	x	Undefined
0FF12H-0FF13H	16-bit compare register (16-bit timer 0)	CR01	R/W	—	—	x	Undefined
0FF14H	8-bit compare register (8-bit timer 1)	CR10	R/W	—	x	—	Undefined
0FF15H	8-bit compare register (8-bit timer/counter 2)	CR20	R/W	—	x	—	Undefined
0FF16H	8-bit compare register (8-bit timer/counter 2)	CR21	R/W	—	x	—	Undefined
0FF17H	8-bit compare register (8-bit timer 3)	CR30	R/W	—	x	—	Undefined
0FF18H-0FF19H	16-bit capture register (16-bit timer 0)	CR02	R	—	—	x	Undefined
0FF1AH	8-bit capture register (8-bit timer/counter 2)	CR22	R	—	x	—	Undefined
0FF1CH	8-bit capture/compare register (8-bit timer 1)	CR11	R/W	—	x	—	Undefined
0FF20H	Port 0 mode register	PM0	W	—	x	—	FFH
0FF21H	Port 1 mode register	PM1	W	—	x	—	FFH
0FF23H	Port 3 mode register	PM3	W	—	x	—	FFH
0FF25H	Port 5 mode register	PM5	W	—	x	—	FFH
0FF26H	Port 6 mode register	PM6	R/W	x	x	—	FxH
0FF30H	Capture/compare control register 0	CRC0	W	—	x	—	10H
0FF31H	Timer output control register	TOC	W	—	x	—	00H
0FF32H	Capture/compare control register 1	CRC1	W	—	x	—	00H
0FF34H	Capture/compare control register 2	CRC2	W	—	x	—	00H
0FF40H	Pullup resistor option register	PUO	R/W	x	x	—	00H
0FF43H	Port 3 mode control register	PMC3	R/W	x	x	—	00H
0FF50H-0FF51H	16-bit timer register 0	TM0	R	—	—	x	0000H
0FF52H	8-bit timer register 1	TM1	R	—	x	—	00H
0FF54H	8-bit timer register 2	TM2	R	—	x	—	00H
0FF56H	8-bit timer register 3	TM3	R	—	x	—	00H
0FF5CH	Prescaler mode register 0	PRM0	W	—	x	—	00H
0FF5DH	Timer control register 0	TMC0	R/W	—	x	—	00H
0FF5EH	Prescaler mode register 1	PRM1	W	—	x	—	00H
0FF5FH	Timer control register 1	TMC1	R/W	—	x	—	00H
0FF60H	D/A converter value setting register 0	DACS0	R/W	—	x	—	00H

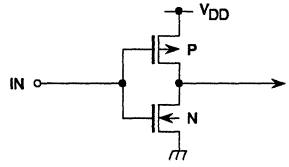
Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
0FF61H	D/A converter value setting register 1	DACS1	R/W	—	x	—	00H
0FF68H	A/D converter mode register	ADM	R/W	x	x	—	00H
0FF6AH	A/D conversion result register	ADCR	R	—	x	—	Undefined
0FF70H	PWM control register	PWMC	R/W	—	x	—	05H
0FF72H-0FF73H	PWM modulo register 0	PWM0	W	—	—	x	Undefined
0FF74H-0FF75H	PWM modulo register 1	PWM1	W	—	—	x	Undefined
0FF7DH	One-shot pulse output control register	OSPC	R/W	x	x	—	00H
0FF80H	Clocked serial interface mode register	CSIM	R/W	x	x	—	00H
0FF82H	Serial bus interface control register	SBIC	R/W	x	x	—	00H
0FF86H	Serial shift register	SIO	R/W	—	x	—	Undefined
0FF88H	Asynchronous serial interface mode register	ASIM	R/W	x	x	—	80H
0FF8AH	Asynchronous serial interface status register	ASIS	R	x	x	—	00H
0FF8CH	Serial receive buffer: UART	RxB	R	—	x	—	Undefined
0FF8EH	Serial transmit shift register: UART	TxS	W	—	x	—	Undefined
0FF90H	Baud rate generator control register	BRGC	W	—	x	—	00H
0FFC0H	Standby control register	STBC	R/W	—	x	—	0000x000B
0FFC4H	Memory expansion mode register	MM	R/W	x	x	—	20H
0FFC5H	Programmable wait control register	PW	R/W	x	x	—	80H
0FFC6H	Refresh mode register	RFM	R/W	x	x	—	00H
0FFCFH	Memory size select register	IMS	W	—	x	—	Undefined
0FFD0H-0FFDFH	External SFR area	—	R/W	x	x	—	Undefined
0FFE0H	Interrupt request flag register L	IF0L	R/W	x	x	—	00H
0FFE1H	Interrupt request flag register H	IF0H	R/W	x	x	—	00H
0FFE0H-0FFE1H	Interrupt request flag register H	IF0	R/W	—	—	x	0000H
0FFE4H	Interrupt mask flag register L	MK0L	R/W	x	x	—	FFH
0FFE5H	Interrupt mask flag register H	MK0H	R/W	x	x	—	FFH
0FFE4H-0FFE5H	Interrupt mask flag register H	MK0H	R/W	—	—	x	FFFH
0FFE8H	Priority specification flag register L	PR0L	R/W	x	x	—	FFH
0FFE9H	Priority specification flag register H	PR0H	R/W	x	x	—	FFH
0FFE8H-0FFE9H	Priority specification flag register H	IF0	R/W	—	—	x	FFFH
0FFECH	Interrupt service mode specification flag register L	ISM0L	R/W	x	x	—	00H
0FFEDH	Interrupt service mode specification flag register H	ISM0H	R/W	x	x	—	00H
0FFECH-0FFEDH	Interrupt service mode specification flag register	ISM0	R/W	—	—	x	00H
0FFF4H	External interrupt mode register 0	INTM0	R/W	x	x	—	00H
0FFF5H	External interrupt mode register 1	INTM1	R/W	x	x	—	00H
0FFF8H	Interrupt status register	IST	R/W	x	x	—	00H

4d

Figure 3. Pin I/O Circuits

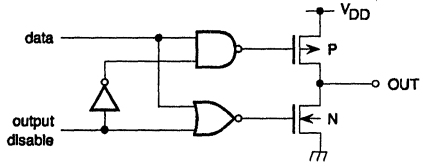
Type 1 (MODE)



Type 2 ($P2_0$, $P2_1$, $\overline{RESET}$)

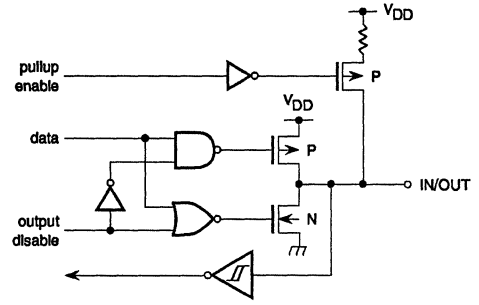
Schmitt trigger input with hysteresis characteristic.

Type 4 (P0, $P6_0$ - $P6_3$, ASTB)

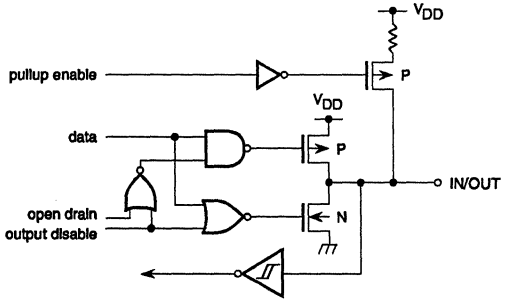


Push-pull output where the output can be placed in high-impedance (both P and N channels are turned off).

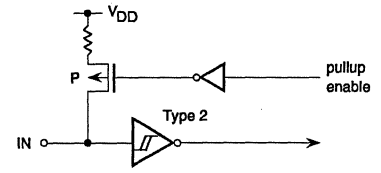
Type 8-A ($P3_2$)



Type 10-A ($P3_3$)

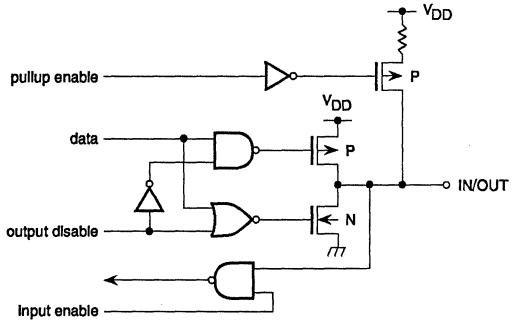


Type 2-A ($P2_2$ - $P2_7$)

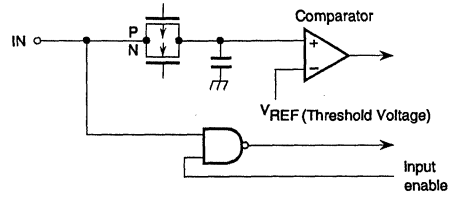


Schmitt trigger input with hysteresis characteristic.

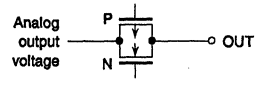
Type 5-A ($P1_0$ - $P1_7$, $P3_0$, $P3_1$, $P3_4$ - $P3_7$, P4, P5, $P6_4$, $P6_7$)



Type 9 (P7)



Type 12 (ANO0 and ANO1)



83YL-9206B

Input/Output Ports

There are up to 64 port lines on the μPD78234/238/P238 and up to 46 port lines on the μPD78233/37. (Ports 4, 5, and two bits of port 6 are not available on the μPD78233/237 since the μPD78233/237 must always use external memory.) Table 2 lists the features of each port and figure 3 shows the structure of each port pin. The pin levels of all port 2, 3, and 7 pins can always be read or tested regardless of the dual pin function.

Table 2. Digital Port Functions

Port	Operational Features	Configuration	Direct Drive Capability	Software Pullup Resistor Connection
Port 0	8-bit high impedance output		Transistor	
Port 1	8-bit input or output	Bit selectable	LED	Byte selectable, input bits only
Port 2	8-bit Schmitt trigger input			In 6-bit unit ($P2_2$-$P2_7$)
Port 3	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 4	8-bit input or output	Byte selectable	LED	Byte selectable
Port 5	8-bit input or output	Byte selectable	LED	Byte selectable, input bits only
Port 6	4-bit output (bits 0 to 3) 4-bit input or output (bits 4 to 7)	Bit selectable		In 4-bit unit, input bits only
Port 7	8-bit input			

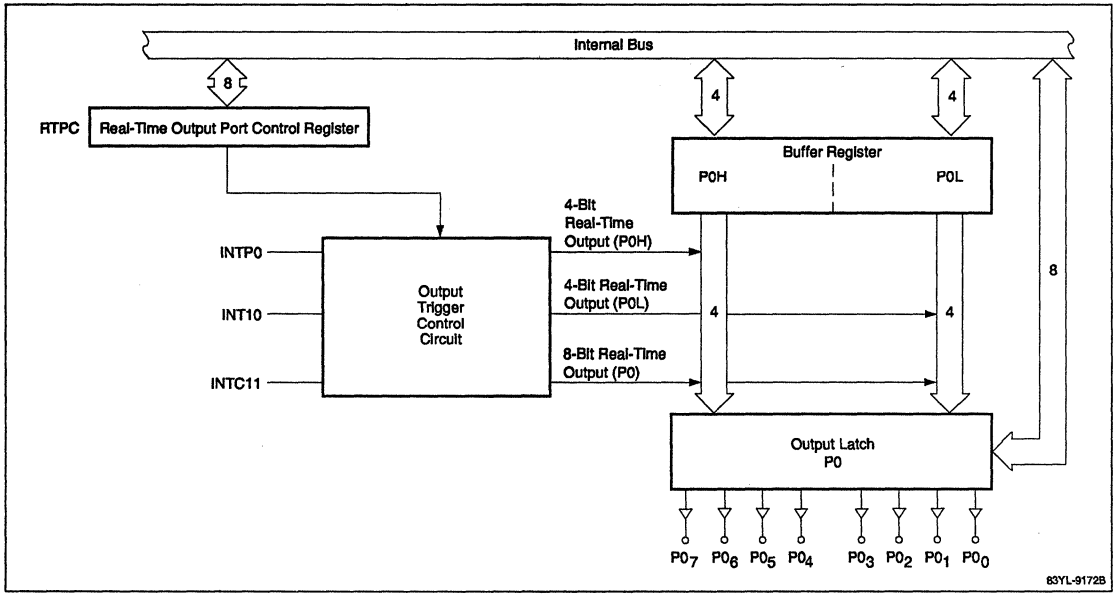
Note:

(1) Software pullup resistors can be internally connected only on a port-by-port bits to port bits set to input mode. Pullup resistors are not connected to port bits set to output mode.

Real-time Output Port

The real-time output port (RTPC) shares pins with port 0. It can be used as two independent 4-bit real-time output ports or one 8-bit real-time output port. In the real-time output mode, data stored beforehand in the buffer registers, P0H and P0L, is transferred immediately to the output latch of P0 on the occurrence of a timer 1 interrupt (INTC10 or INTC11) or external interrupt (INTP0) (see figure 4). By using the real-time output port with the macro service function, port 0 can be used to output preprogrammed patterns at preprogrammed variable time intervals. In this mode, two independent stepper motors can accurately be driven at a fixed or variable rate.

Figure 4. Real-time Output Port



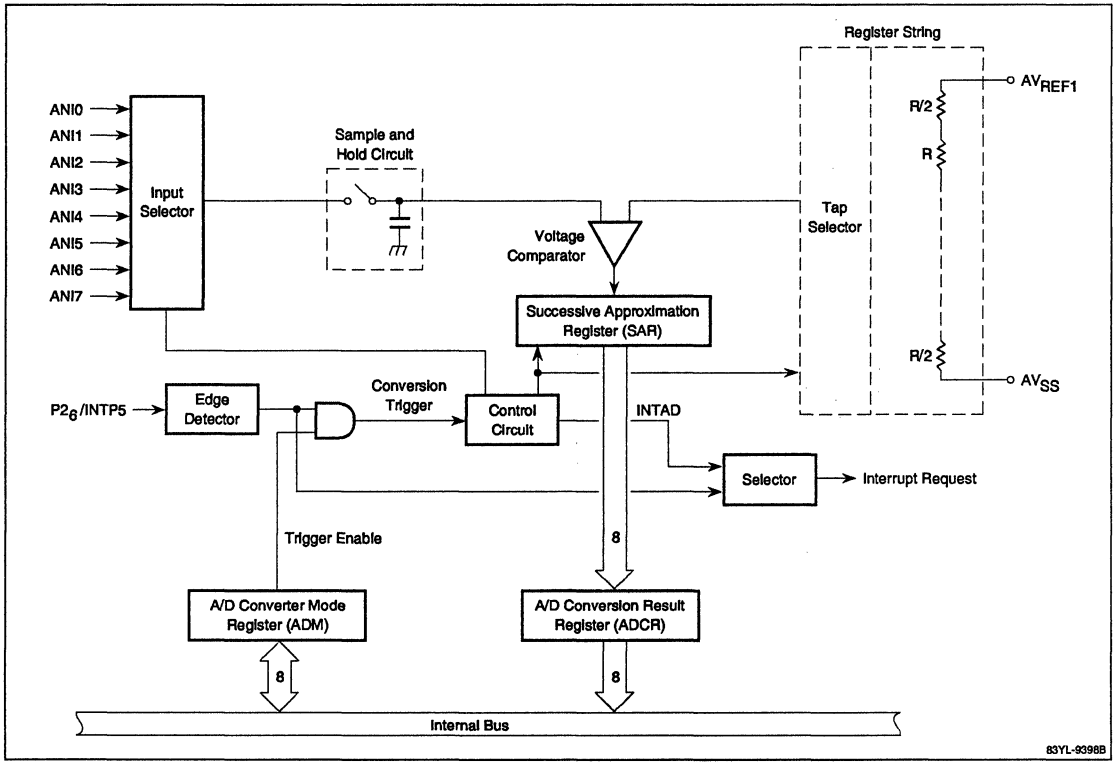
Analog-to-Digital (A/D) Converter

The μPD78238 family A/D converter (see figure 5) uses the successive-approximation method for converting up to eight multiplexed analog inputs into 8-bit digital data. The conversion time per input is 30 μs at 12 MHz operation. A/D conversion can be started by an external interrupt, INTP5, or under software control.

The A/D converter can operate in either scan mode or select mode. In scan mode, from one to eight sequential inputs can be programmed for conversion. The A/D converter selects each input in order, converts the data, stores it in the A/D conversion result (ADCR) register, and generates an interrupt (INTAD). This converted data can be easily transferred to memory by using the macro service function.

In select mode, only one of the eight A/D inputs can be selected for conversion. The ADCR register is continually updated and can be read at any time. If the A/D converter is started by an external interrupt, an INTAD interrupt occurs at the completion of each conversion. If the A/D converter is started by software, no interrupts are generated.

Figure 5. A/D Converter

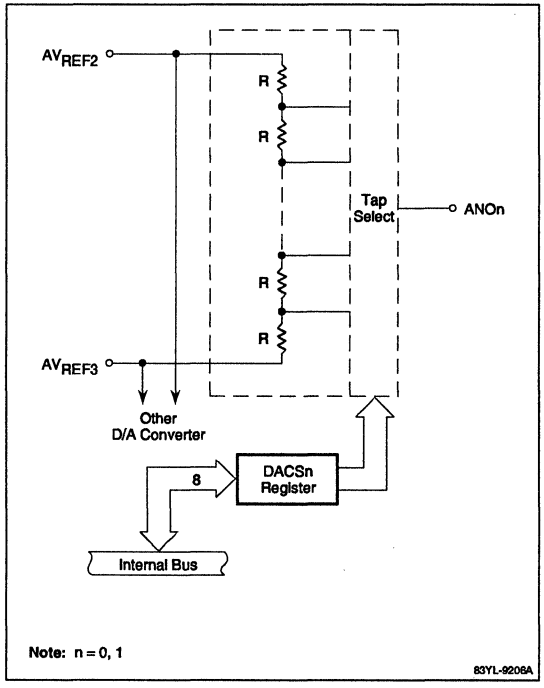


4d

Digital-to-Analog (D/A) Converter

The μPD78238 family has two D/A converters as shown in figure 6. The 8-bit digital data, written to the DACSn register (n= 0, 1), selects one of the 256 taps on a resistor ladder between AV_{REF2} and AV_{REF3}. The selected voltage becomes the analog output at the ANOn pin. The ANOn is a high-impedance output and requires an external buffer to drive a low-impedance load.

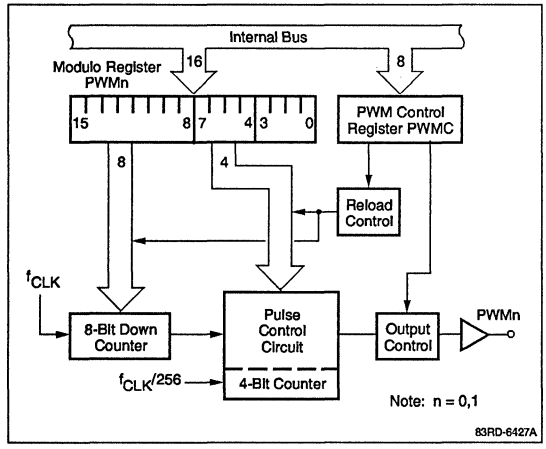
Figure 6. D/A Converter



Hardware Pulse-Width Modulated Outputs

The μPD78238 family has two 12-bit resolution pulse-width modulated (PWM) outputs (see figure 7) with a repetition rate of 23.4 kHz at 12 MHz (f_{CLK} = 6 MHz). The polarity of each output can be selected under program control. The two PWM outputs, PWM0 and PWM1, share pins with port 1, bits 0 and 1 respectively. These outputs are designed for controlling DC motors.

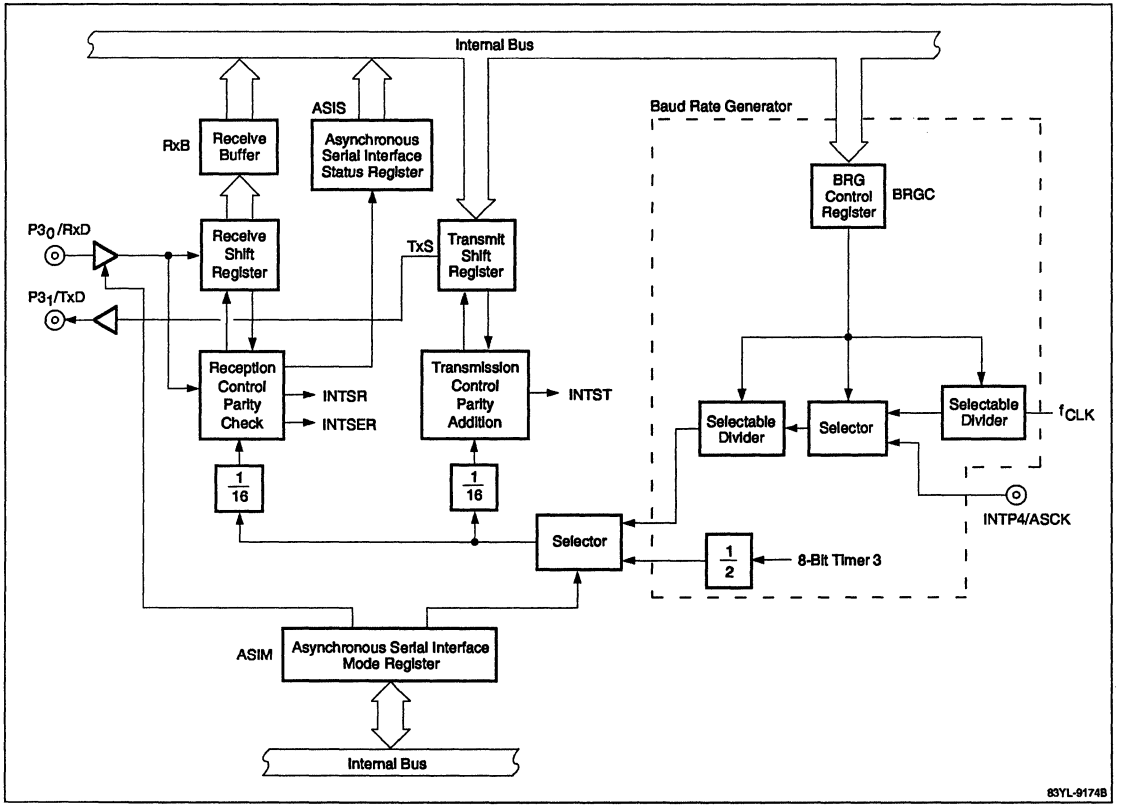
Figure 7. Hardware Pulse-Width Modulator



Serial Interface

The μPD78238/P238 have two independent serial interfaces. The first is a standard UART. The UART (figure 8) permits full-duplex operation and can be programmed for 7- or 8-bits of data after the start bit, followed by one or two stop bits. Odd, even, zero or no parity can also be selected. The serial clock for the UART can be provided by an on-chip baud rate generator or timer 3. By using either the internal system clock or an external clock input into the ASCK pin, the baud rate generator is capable of generating all of the commonly used baud rates. The UART generates three interrupts: INTST (transmission complete), INTSR (reception complete), and INTSER (reception error).

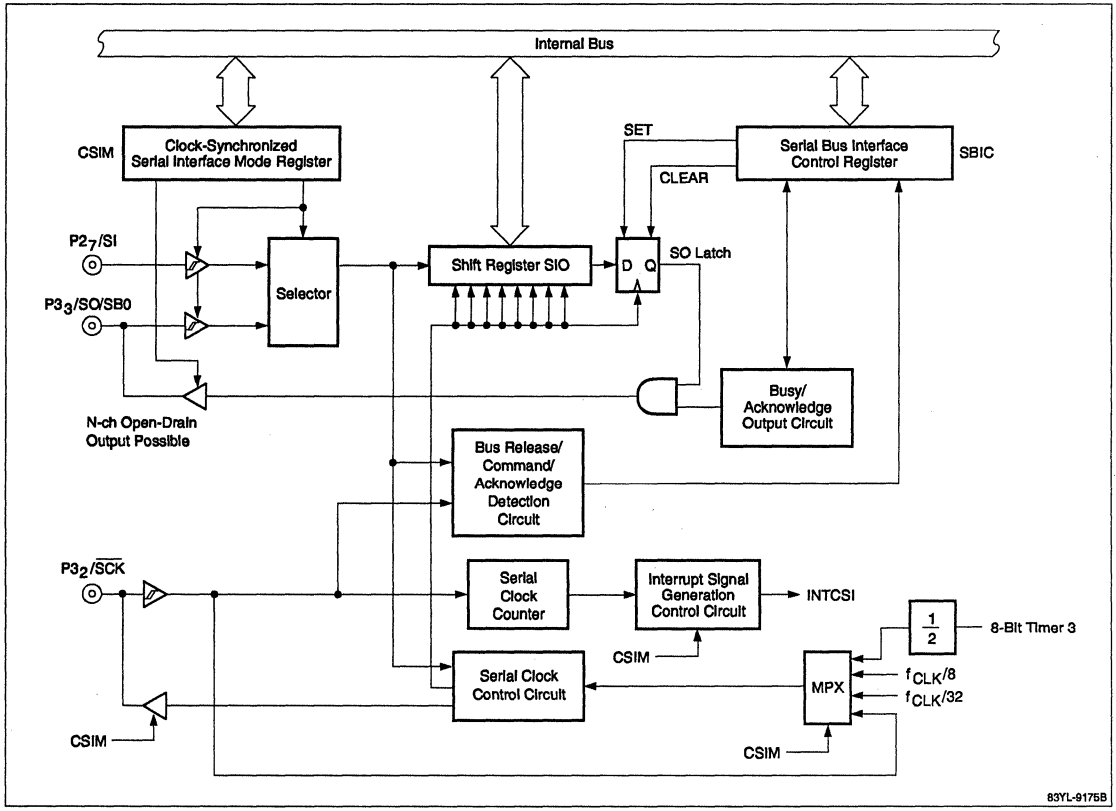
Figure 8. Asynchronous Serial Interface



4d

The second interface is an 8-bit clock-synchronized serial interface (figure 9). It can be operated in either a three-wire serial I/O mode or NEC serial bus interface (SBI) mode.

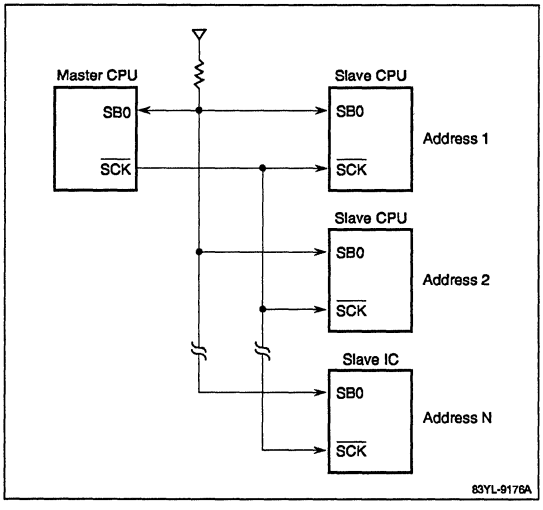
Figure 9. Clock-Synchronized Serial Interface



In the three-wire serial I/O mode, the 8-bit shift register (SIO) is loaded with a byte of data and eight clock pulses are generated. These eight pulses shift the byte of data out of the SO line (MSB first) and in from the SI line providing full-duplex operation. This interface can also be set to receive or to transmit data only. The INTCSI interrupt is generated after each 8-bit transfer. One of three internal clocks or an external clock clocks the data.

The NEC SBI mode is a two-wire high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are connected in a master/slave configuration (see figure 10). There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over the serial bus line (SB0) using a fixed hardware protocol synchronized with the $\overline{SCK}$ line. Each slave μPD78238 family device can be programmed in software to respond to any one of 256 addresses. There are also 256 commands and 256 data types. Since all commands are user definable, any software protocol, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

Figure 10. SBI Mode Master/Slave Configuration



Timers

The μPD78238 family has one 16-bit timer and three 8-bit timers. The 16-bit timer counts the internal system clock ($f_{CLK}/8$) while the three 8-bit timers can be programmed to count a number of prescaled values of the internal system clock. One of the 8-bit timers can also count external events.

Timer 0 consists of a 16-bit timer (TM0), two 16-bit compare registers (CR00 and CR01), and a 16-bit capture register (CR02). Timer 0 can be used as two interval timers, to output a programmable square wave or two pulse-width modulated signals, to measure pulse widths, or to generate a software-triggered one-shot pulse. (see figure 11).

Timer 1 consists of an 8-bit timer (TM1), 8-bit compare register (CR10), and 8-bit capture/compare register (CR11). Timer 1 can be used as two interval timers or to measure pulse widths. In addition, it can be used to generate the output trigger for the real-time output port (see figure 12).

Timer/counter 2 consists of an 8-bit timer (TM2), two 8-bit compare registers (CR20 and CR21), and an 8-bit capture register (CR22). Timer/counter 2 can also be used as two interval timers, to output a programmable square wave or two pulse-width modulated signals, or to measure pulse widths. In addition, it can be used to count external events sensed on the CI line or as a one-shot timer (see figure 13).

Timer 3 consists of an 8-bit timer (TM3) and an 8-bit compare register (CR30). Timer 3 can be used as an interval timer or as a clock for the clock-synchronized serial interface (see figure 14).

Figure 11. 16-Bit Timer 0

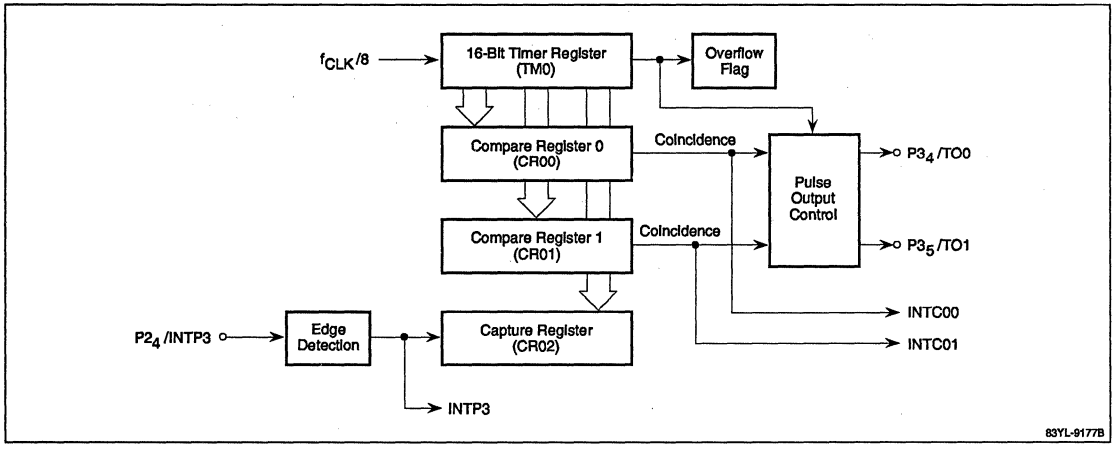


Figure 12. 8-Bit Timer 1

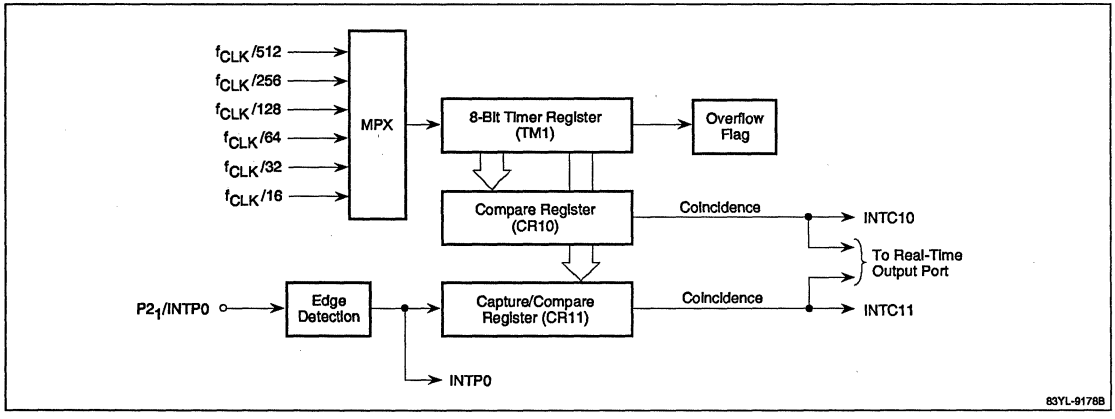


Figure 13. 8-Bit Timer/Counter 2

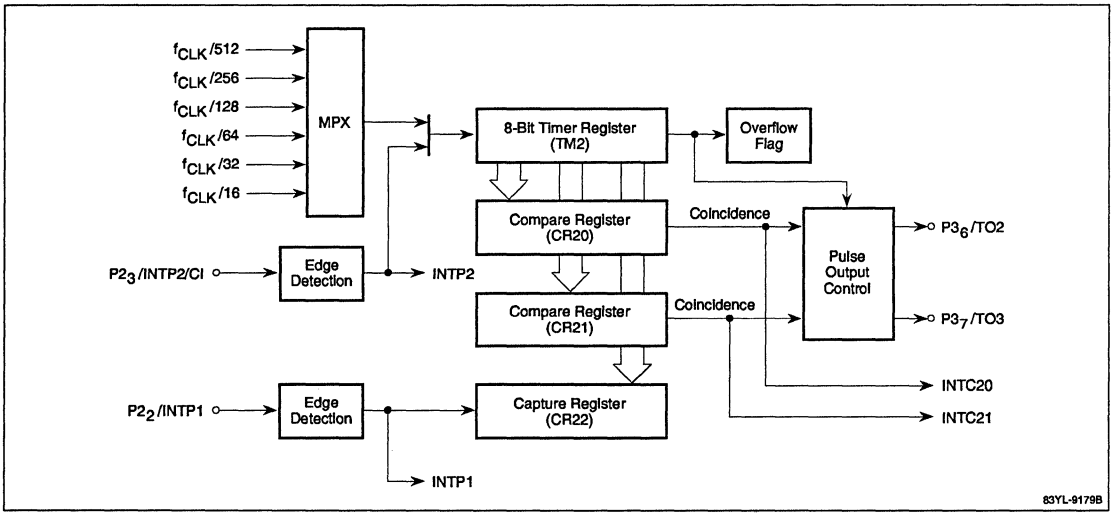
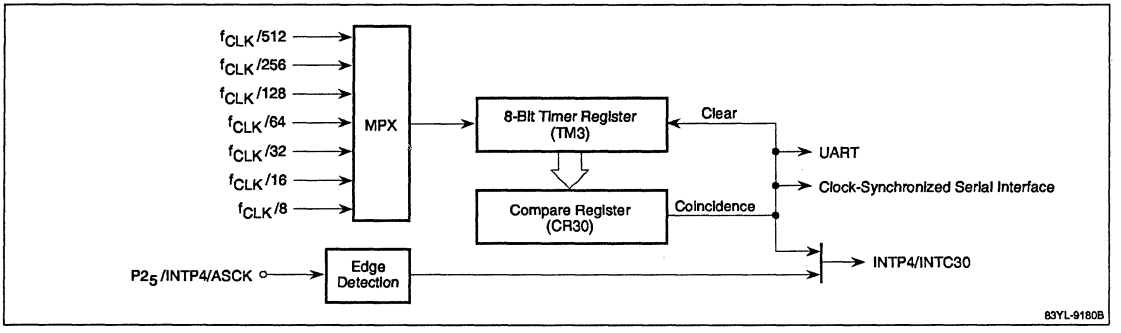


Figure 14. 8-Bit Timer 3



4d

Interrupts

The μPD78238 family has 18 maskable hardware interrupt sources; 6 are external and 12 are internal. Since there are only 16 interrupt vectors and sets of control flags, 2 of the 6 external maskable interrupts, INTP4 and INTP5, share interrupt vectors and control flags with INTC30 and INTAD respectively. The active interrupt source for each shared vector must be chosen by the program. In addition, there is one nonmaskable interrupt and one software interrupt. The software interrupt, generated by the BRK instruction, is not maskable (see table 3).

Interrupt Servicing. The μPD78238 family provides two levels of programmable hardware priority control and two different methods of handling maskable interrupt requests: standard vectoring and macro service. The programmer can choose the priority and mode of servicing each maskable interrupt by using the interrupt control registers.

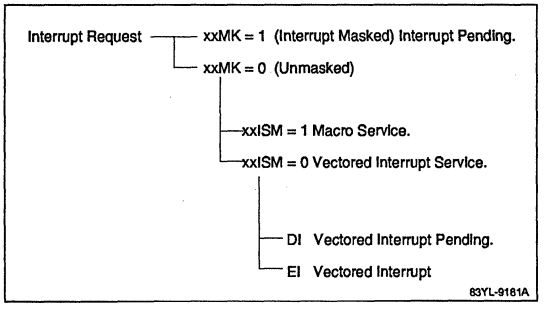
Interrupt Control Registers. The μPD78238 family has four 16-bit interrupt control registers. Each bit in each register is dedicated to one of the 16 active maskable interrupt sources. The interrupt request flag register (IF0) contains an interrupt request flag for each interrupt. The interrupt mask register (MK0) is used to enable or disable any interrupt. The interrupt service mode register (ISM0) specifies whether an interrupt is processed by vectoring or macro service. The priority flag register (PR0) can be used to specify a high or a low priority level for each interrupt.

Two other 8-bit registers are associated with interrupt processing. The interrupt status register (IST) indicates if a nonmaskable interrupt request on the NMI pin is being processed and can be used to allow nesting of nonmaskable interrupt requests. The IE and the ISP bits of the program status word are also used to control interrupts. If the IE bit is zero, all maskable interrupts, but not macro service, are disabled. The IE bit can be set or cleared using the EI and DI instructions, respectively, or by directly writing to the PSW. The IE bit is cleared each time an interrupt is accepted. The ISP bit is used by hardware to hold the priority level flag of the interrupt being serviced.

Interrupt Priority. The nonmaskable interrupt (NMI) has priority over all other interrupts. Two hardware controlled priority levels are available for the maskable interrupts. Either a high or a low priority level can be assigned by software to each of the maskable interrupts. Interrupt requests of a priority higher than the processor's current priority level are accepted; requests of the same or lower priority are held pending until the processor's priority state is lowered by program control within the current service routine or by a return instruction from the current service routine.

Interrupt requests programmed to be handled by macro service have priority over all vectored interrupt service regardless of the assigned priority level, and macro service requests are accepted even when the interrupt enable bit in the PSW is set to the disable state (see figure 15).

Figure 15. Interrupt Service Sequence



The default priorities listed in table 3 are fixed by hardware and are effective only when it is necessary to choose between two interrupt requests of the same software assigned priority. For example, the default priorities would be used after the completion of a high priority routine if two interrupts of the same priority routine were pending.

The software interrupt, initiated by the BRK instruction, is executed regardless of the processor's priority level and the state of the IE bit. It does not alter the processor's priority level.

Table 3. Interrupt Sources and Vector Addresses

Interrupt Request Type	Default Priority	Interrupt Request Generation Source	Macro Service Type	Vector Table Address
Software	None	BRK instruction execution	—	003EH
Nonmaskable	None	NMI (pin input edge detection)	—	0002H
Maskable	0	INTP0 (pin input edge detection)	A, B	0006H
	1	INTP1 (pin input edge detection)	A, B	0008H
	2	INTP2 (pin input edge detection)	A, B	000AH
	3	INTP3 (pin input edge detection)	B	000CH
	4	INTC00 (TM0-CR00 coincidence signal generation)	B	0014H
	5	INTC01 (TM0-CR01 coincidence signal generation)	B	0016H
	6	INTC10 (TM1-CR10 coincidence signal generation)	A, B, C	0018H
	7	INTC11 (TM1-CR11 coincidence signal generation)	A, B, C	001AH
	8	INTC21 (TM2-CR21 coincidence signal generation)	A, B	001CH
	9	INTP4 (pin input edge detection)	B	000EH
		INTC30 (TM3-CR30 coincidence signal generation)	A, B	
	10	INTP5 (pin input edge detection)	B	0010H
		INTAD (end of A/D conversion)	A, B	
	11	INTC20 (TM2-CR20 coincidence signal generation)	A, B	0012H
	12	INTSER (generation of asynchronous serial interface receive error)	—	0020H
	13	INTSR (end of asynchronous serial interface reception)	A, B	0022H
	14	INTST (end of asynchronous serial interface transmission)	A, B	0024H
	15	INTCSI (end of clocked serial interface transmission)	A, B	0026H

Vectored Interrupt. When vectored interrupt is specified for a given interrupt request, (1) the program status word and the program counter are saved on the stack, (2) the processor's priority is set to that specified for the interrupt, (3) the IE bit in the PSW is set to zero, and (4) the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (RETB instruction for the software interrupt) reverses the process and the μPD78238 family device resumes the interrupted routine.

Macro Service

When macro service is specified for a given interrupt, the macro service hardware temporarily stops the executing program and begins to transfer data between the special function register area and the memory space. One byte is transferred each interrupt. When the data transfer is complete, control is returned to the executing program, providing a completely transparent method of interrupt service. Macro service significantly improves response time and makes it unnecessary to save any registers.

For each request on the interrupt line, one operation is performed, and an 8- or 16-bit counter is decremented. When the counter reaches zero, a vectored interrupt service routine is entered according to the specified priority.

Macro service is provided for all of the maskable interrupt requests except INTSER, the asynchronous serial interface receive error interrupt request. Each interrupt request has a dedicated macro service control word stored in Internal RAM (see figure 16). The function to be performed is specified in the control word.

The μPD78238 family provides three different types of macro service transfers:

Macro Service Type A. A byte of data is transferred in either direction between a special function register, preassigned for each interrupt request, and a buffer in internal RAM (FExx). Only the 8-bit macro service counter is available for Type A transfers. The preassigned SFRs for the 12 interrupt requests that support macro service Type A transfers are listed in table 4.

Figure 16. Macro Service Control Word Map

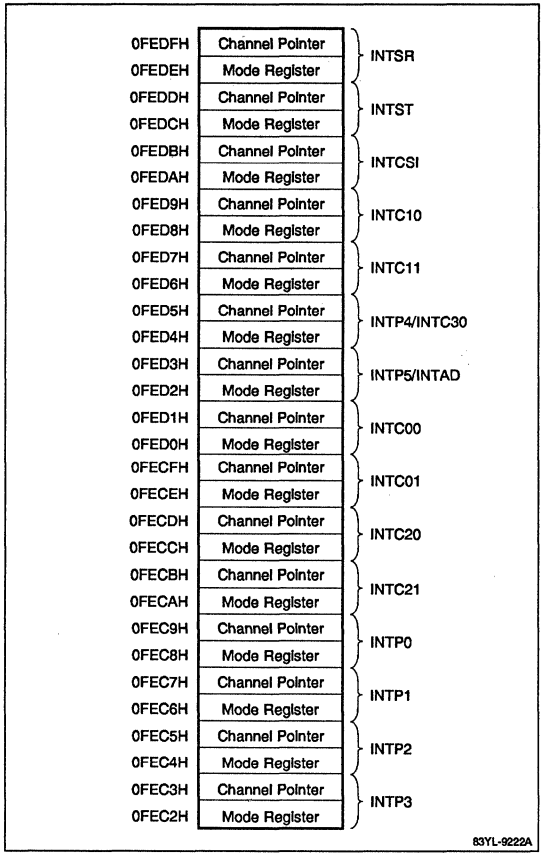


Table 4. Macro Service Type A Interrupts and Assigned SFRs

Interrupt Request	Source/Destination SFR
INTC10: TM1-CR10 coincidence	CR10: Timer 1 8-bit compare register
INTC11: TM1-CR11 coincidence	CR11: Timer 1 8-bit capture/compare register
INTC20: TM2-CR20 coincidence	CR20: Timer 2 8-bit compare register
INTC21: TM2-CR21 coincidence	CR21: Timer 2 8-bit compare register
INTC30: TM3-CR30 coincidence	CR30: Timer 3 8-bit compare register
INTSR: End of asynchronous serial interface reception	RxB: Serial receive buffer
INTST: End of asynchronous serial interface transmission	TxS: Serial transmit shift register
INTCSI: End of clocked serial interface transmission	SIO: Serial shift register
INTAD: End of A/D conversion	ADCR: A/D conversion result register
INTP0: External interrupt pin $P0_1$	CR11: Timer 1 8-bit capture/compare register
INTP1: External interrupt pin $P0_2$	CR22: Timer 2 8-bit capture register
INTP2: External interrupt pin $P0_3$	TM2: Timer 2 8-bit timer register

Macro Service Type B. A byte of data is transferred in either direction between any specified special function register and a buffer anywhere in the 64K byte address space. Macro service Type B transfers can be initiated by any maskable interrupt except INTSER.

Macro Service Type C. A byte of data is transferred from a buffer anywhere in the 64K byte address space to one of the 8-bit compare registers of timer 1. At the same time, a second byte of data is transferred from a buffer anywhere in the 64K byte address space to the real-time output port buffer. The macro service counter can be programmed either to be an 8- or 16-bit counter. Macro service Type C transfers can be initiated by INTC10 with data transferred to CR10 and P0L or P0H, or by INTC11 with data transferred to CR11 and P0L or P0H.

In addition, the macro service Type C transfer can be initialized to automatically alter timer compare register values or to repeatedly output a prespecified pattern at a fixed or variable rate. By using macro service Type C transfers to control the real-time output ports, the μPD78238 family device can easily and accurately drive two independent stepper motors.

Refresh

The refresh signal is used with any pseudostatic RAM equivalent of the NEC μPD428128. The refresh cycle can be set to one of four intervals: 16, 32, 64, or 128/f_{CLK} (2.6, 5.3, 10.7, and 21.3 μs at 12 MHz). The refresh cycle is timed to follow a read or write operation to avoid interference with external memory access cycles.

Standby Modes

HALT and STOP modes are provided to reduce power consumption when CPU action is not required. In HALT mode, the CPU is stopped but the system clock continues to run. The HALT mode is released by any unmasked interrupt, an external NMI, or an external reset pulse. In STOP mode, both the CPU and the system clock are stopped, further minimizing the power consumption. The STOP mode is released by either an external reset pulse or an external NMI. The HALT and STOP modes are entered by programming the standby control register (STBC). This register is a protected location and can be written to only by a special instruction. If the third and fourth bytes of the instruction are not complements of each other, the data is not written and the next instruction is executed.

External Reset

The μPD78238 family is reset by taking the $\overline{\text{RESET}}$ pin low. The $\overline{\text{RESET}}$ input pin contains a noise filter to protect against spurious system resets caused by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low until the power supply reaches its operating voltage and the oscillator has stabilized. During reset, the program counter is loaded with the address contained in the reset vector table (address 0000H- 0001H); program execution starts at that address upon the $\overline{\text{RESET}}$ pin going high. While $\overline{\text{RESET}}$ is low, all external lines except V_{SS}, V_{DD}, AV_{SS}, AV_{REF1}, AV_{REF2}, AV_{REF3}, X1, and X2 are in the high impedance state.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = +25°C

Operating voltage, V_{DD}	−0.5 to +7.0 V
AV_{DD}	AV_{SS} to V_{DD} + 0.5 V
AV_{SS}	−0.5 to + 0.5 V
Input voltage, V_{I1}	−0.5 to V_{DD} + 0.5 V
V_{I2} (Note 1 for μPD78P238)	−0.5 to +13.5 V
Output voltage, V_O	−0.5 to V_{DD} + 0.5 V
Low-level output current, I_{OL}	
Per pin	15 mA
Total, all output pins	100 mA
High-level output current, I_{OH}	
Per pin	−10 mA
Total, all output pins	−50 mA
A/D converter reference input voltage, AV_{REF1}	−0.5 to V_{DD} + 0.3 V
D/A converter reference input voltage, AV_{REF2}	−0.5 to V_{DD} + 0.3 V
AV_{REF3}	−0.5 to V_{DD} + 0.3 V
Operating temperature, T_{OPT}	−40 to +85°C
Storage temperature, T_{STG}	−65 to +150°C

Note:

(1) MODE/V_{PP} and $P2_1$/INTP0/A_9 in programming mode

Exposure to absolute maximum ratings for extended periods may affect device reliability; exceeding the rating could cause permanent damage. The device should be operated within the limits specified under DC and AC chracteristics.

Operating Conditions

Oscillation Frequency	T_A	V_{DD}
f_{XX} = 4 to 12 MHz	−40 to +85°C	+5 V ±10%

Capacitance

T_A = +25°C; V_{DD} = V_{SS} = 0 V.

Item	Symbol	Max	Unit	Conditions
Input capacitance	C_I	20	pF	f = 1 MHz; pins not used for measurement are at 0 V
Output capacitance	C_O	20	pF	
Input/output capacitance	C_{IO}	20	pF	

External Clock Operation

T_A = −40 to +85°C; V_{DD} = +5 V 10%; V_{SS} = 0 V

Item	Symbol	Min	Max	Unit	Conditions
X1 input low-level width	t_{WXL}	30	130	ns	
X1 input high-level width	t_{WXH}	30	130	ns	
X1 input rise time	t_{XR}	0	30	ns	
X1 input fall time	t_{XF}	0	30	ns	
X1 input clock cycle time	t_{CYX}	82	250	ns	

DC Characteristics

T_A = −40 to +85°C; V_{DD} = AV_{DD} = +5 V ±10%; V_{SS} = AV_{SS} = 0 V

Item	Symbol	Min	Typ	Max	Unit	Conditions
Low-level input voltage	V_{IL}	0		0.8	V	
High-level input voltage	V_{IH1}	2.2		V_{DD}	V	Except the specified pins (Note 1)
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	Specified pins (Note 1)
Low-level output voltage	V_{OL1}			0.45	V	I_{OL} = 2.0 mA
	V_{OL2}			1.0	V	I_{OL} = 8.0 mA (Note 2)
High-level output voltage	V_{OH1}	V_{DD} − 1.0			V	I_{OH} = −1.0 mA
	V_{OH2}	V_{DD} − 0.5			V	I_{OH} = −100 μA
	V_{OH3}	2.0			V	I_{OH} = −5.0 mA (Note 3)
X1 low-level input current	I_{IL}			−100	μA	0 V ≤ V_I ≤ V_{IL}
X1 high-level input current	I_{IH}			100	μA	V_{IH2} ≤ V_I ≤ V_{DD}
Input leakage current	I_{LI}			±10	μA	0 V ≤ V_I ≤ V_{DD}
Output leakage current	I_{LO}			±10	μA	0 V ≤ V_O ≤ V_{DD}
V_{DD} power supply current	I_{DD1}		20	40	mA	Operating mode, f_{XX} = 12 MHz
	I_{DD2}		7	20	mA	HALT mode, f_{XX} = 12 MHz
Data retention voltage	V_{DDDR}	2.5		5.5	V	STOP mode

DC Characteristics (cont)

Item	Symbol	Min	Typ	Max	Unit	Conditions
Data retention current	I_{DDDR}			10	μA	STOP mode; V_{DDDR} = 2.5 V
				20	μA	STOP mode; V_{DDDR} = 5 V ±10%
Pullup resistor	R_L	15	40	80	kΩ	V_I = 0 V

Notes:

(1) X1, X2, $\overline{RESET}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1,$P2_3$/$\overline{INTP2}$/CI, $P2_4$/INTP3, $P2_5$/INTP4/ASCK, $P2_6$/INTP5, $P2_7$/SI, $P3_2$/SCK, $P3_3$/SO/SB0, and MODE pins.

(2) Pins $P1_0$ - $P1_7$, $P4_0/AD_0$ - $P4_7/AD_7$, and $P5_0/A_8$ - $P5_7/A_{15}$.

(3) Pins $P0_0$ - $P0_7$.

AC Characteristics—Read/Write Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF

Item	Symbol	Calculation Formula (Note 2, 3)	Min	Max	Unit	Conditions
X1 input clock cycle time	t_{CYX}	—	82	250	ns	
Address setup time to ASTB ↓	t_{SAST}	t_{CYX} − 30	52		ns	
Address hold time from ASTB ↓ (Note 1)	t_{HSTA}	—	25		ns	
Address hold time from $\overline{RD}$↑	t_{HRA}	—	30		ns	
Address hold time from $\overline{WR}$ ↑	t_{HWA}	—	30		ns	
Address to $\overline{RD}$ ↓ delay time	t_{DAR}	$2t_{CYX}$ − 35	129		ns	
Address float time to $\overline{RD}$ ↓	t_{FAR}	$t_{CYX}/2$ − 30	11		ns	
Address to data input time	t_{DAID}	(4+2n) t_{CYX} − 100		228	ns	No wait states
ASTB ↓ to data input time	t_{DSTID}	(3+2n)t_{CYX} − 65		181	ns	No wait states
$\overline{RD}$ ↓ to data input time	t_{DRID}	(2+2n)t_{CYX} − 64		100	ns	No wait states
ASTB ↓ to $\overline{RD}$ ↓ delay time	t_{DSTR}	t_{CYX} − 30	52		ns	
Data hold time from $\overline{RD}$ ↑	t_{HRID}	—	0		ns	
$\overline{RD}$ ↑ to address active time	t_{DRA}	$2t_{CYX}$ − 40	124		ns	
$\overline{RD}$ ↑ to ASTB ↑ delay time	t_{DRST}	$2t_{CYX}$ − 40	124		ns	
$\overline{RD}$ low-level width	t_{WRL}	(2+2n)t_{CYX} − 40	124		ns	No wait states
ASTB high-level width	t_{WSTH}	t_{CYX} − 30	52		ns	
Address to $\overline{WR}$ ↓ delay time	t_{DAW}	$2t_{CYX}$ − 35	129		ns	
$\overline{ASTB}$ ↓ to data output time	t_{DSTOD}	t_{CYX} + 60		142	ns	
$\overline{WR}$ ↓ to data output time	t_{DWOD}	—		60	ns	
ASTB ↓ to $\overline{WR}$ ↓ delay time	t_{DSTW1}	t_{CYX} − 30	52		ns	
	t_{DSTW2}	$2t_{CYX}$ − 35	129		ns	Refresh mode
Data setup time to $\overline{WR}$ ↑	t_{SODWR}	(3+2n)t_{CYX} − 100	146		ns	No wait states
Data setup time to $\overline{WR}$ ↓	t_{SODWF}	t_{CYX} − 60	22		ns	Refresh mode
Data hold time from $\overline{WR}$ ↑ (Note 1)	t_{HWOD}	—	20		ns	
$\overline{WR}$ ↑ to ASTB ↑ delay time	t_{DWST}	t_{CYX} − 40	42		ns	
$\overline{WR}$ low-level width	t_{WWL1}	(3+2n)t_{CYX} − 50	196		ns	No wait states
	t_{WWL2}	(2+2n)t_{CYX} − 50	114		ns	Refresh mode; No wait states
Address to $\overline{WAIT}$ ↓ input time	t_{DAWT}	$3t_{CYX}$ − 100		146	ns	
ASTB ↓ to $\overline{WAIT}$ ↓ input time	t_{DSTWT}	$2t_{CYX}$ − 80		84	ns	
$\overline{WAIT}$ hold time from ASTB ↓	t_{HSTWT}	$2Xt_{CYX}$ + 10	174		ns	One external wait state
ASTB ↓ to $\overline{WAIT}$ ↑ delay time	t_{DSTWTH}	2(1+X)t_{CYX} − 55		273	ns	One external wait state

AC Characteristics—Read/Write Operation (cont)

Item	Symbol	Calculation Formula (Note 2, 3)	Min	Max	Unit	Conditions
$\overline{RD}$ ↓ to $\overline{WAIT}$ input time	t_{DRWTL}	$t_{CYX} - 60$		22	ns	
$\overline{WAIT}$ hold time from $\overline{RD}$ ↓	t_{HRWT}	$(2X-1)t_{CYX} + 5$	87		ns	One external wait state
$\overline{RD}$ ↓ to $\overline{WAIT}$ ↑ delay time	t_{DRWTH}	$(2X+1)t_{CYX} - 60$		186	ns	One external wait state
$\overline{WAIT}$ ↑ to data input time	t_{DWTID}	$t_{CYX} - 20$		62	ns	
$\overline{WAIT}$ ↑ to $\overline{WR}$ ↑ delay time	t_{DWTW}	$2t_{CYX} - 10$	154		ns	
$\overline{WAIT}$ ↑ to $\overline{RD}$ ↑ delay time	t_{DWTR}	$t_{CYX} - 10$	72		ns	
$\overline{WR}$ ↓ to $\overline{WAIT}$ ↓ input time	t_{DWWTL}	$t_{CYX} - 60$		22	ns	Refresh disabled
$\overline{WAIT}$ hold time from $\overline{WR}$ ↓	t_{HWWT1}	$(2X-1)t_{CYX} + 5$	87		ns	One external wait state; refresh disabled
	t_{HWWT2}	$2(X-1)t_{CYX} + 5$	5		ns	One external wait state; refresh enabled
$\overline{WR}$ ↓ to $\overline{WAIT}$ ↑ delay time	t_{DWWTH1}	$(2X+1)t_{CYX} - 60$		186	ns	One external wait state; refresh disabled
	t_{DWWTH2}	$2Xt_{CYX} - 60$		104	ns	One external wait state; refresh enabled
$\overline{RD}$ ↑ to $\overline{REFRQ}$ ↓ delay time	t_{DRRFQ}	$2t_{CYX} - 10$	154		ns	
$\overline{WR}$ ↑ to $\overline{REFRQ}$ ↓ delay time	t_{DWRFQ}	$t_{CYX} - 10$	72		ns	
$\overline{REFRQ}$ low-level width	t_{WRFQL}	$2t_{CYX} - 44$	120		ns	
$\overline{REFRQ}$ ↑ to ASTB ↑ delay time	t_{DRFQST}	$4t_{CYX} - 48$	280		ns	

Notes:

(1) The hold time includes the time during which V_{OH} and V_{OL} are retained under the following load conditions: C_L = 100 pF and R_L = 2 kΩ

(2) n indicates the number of internal wait states.

(3) X indicates the number of external wait states (1, 2, 3, ...)

Serial Port Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF

Item	Symbol	Min	Max	Unit	Conditions
Serial clock cycle time	t_{CYSK}	1.0		μs	External clock input
		1.3		μs	Internal clock/16 output
		5.3		μs	Internal clock/64 output
Serial clock low-level width	t_{WSKL}	420		ns	External clock input
		556		ns	Internal clock/16 output
		2.5		μs	Internal clock/64 output
Serial clock high-level width	t_{WSKH}	420		ns	External clock input
		556		ns	Internal clock/16 output
		2.5		μs	Internal clock/64 output
SI, SB0 setup time to $\overline{SCK}$ ↑	t_{SSSK}	150		ns	
SI, SB0 hold time from $\overline{SCK}$ ↑	t_{HSSK}	400		ns	
SO/SB0 output delay time from $\overline{SCK}$ ↓	t_{DSBSK1}	0	300	ns	CMOS push-pull output (3-line serial I/O mode)
	t_{DSBSK2}	0	800	ns	Open-drain output (SBI mode), R_L = 1 kΩ
SB0 high, hold time from $\overline{SCK}$ ↑	t_{HSBSK}	4		t_{CYX}	SBI mode
SB0 low, setup time to $\overline{SCK}$ ↓	t_{SSBSK}	4		t_{CYX}	SBI mode
SB0 low-level width	t_{WSBL}	4		t_{CYX}	
SB0 high-level width	t_{WSBH}	4		t_{CYX}	

4d

A/D Converter Operation

T_A = −40 to +85°C; V_{DD} = AV_{DD} +5 V ±10%; V_{SS} = AV_{SS} = 0 V

Item	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		8			Bit	
Full-scale error (Note 1)				0.4	%	AV_{REF1} = 4.0 V to AV_{DD}; T_A = −10 to +70°C
				0.8	%	AV_{REF1} = 3.4 V to AV_{DD}
				0.6	%	AV_{REF1} = 4.0 V to AV_{DD}
Quantization error				±1/2	LSB	
Conversion time	t_{CONV}	240			t_{CYX}	82 ns ≤ t_{CYX} ≤ 250 ns
Sampling time	t_{SAMP}	48			t_{CYX}	82 ns ≤ t_{CYX} ≤ 250 ns
Analog input voltage	V_{IAN}	−0.3		AV_{REF1} + 0.3	V	
Analog input impedance	R_{AN}		1000		MΩ	
Analog reference voltage	AV_{REF1}	3.4		AV_{DD}	V	
AV_{REF1} current	AI_{REF1}		1.5	3.0	mA	f_{XX} = 12 MHz
			0.7	1.5	mA	Note 2
AV_{DD} current	AI_{DD1}		1.4	3.0	mA	f_{XX} = 12 MHz
	AI_{DD2}		10	20	μA	Note 3

Notes:

(1) Quantization errror is not included. Unit is defined as percent of full-scale value.

(2) When CS bit of the ADM register is set to 0.

(3) When CS bit of the ADM register is set to 0 in the STOP mode.

D/A Converter Operation

T_A = −40 to +85°C; AV_{REF2} = V_{DD} = +5 V ±10%; AV_{REF3} = V_{SS} = 0 V

Item	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		8			Bit	
Overall Error				0.4	%	Load conditions: 4 MΩ, 30 pF
				0.6	%	Load conditions: 2 MΩ, 30 pF
				0.6	%	AV_{REF2} = 0.75 V_{DD} AV_{REF3} = 0.25 V_{DD} Load conditions: 4 MΩ, 30 pF
				0.8	%	AV_{REF2} = 0.75 V_{DD}; AV_{REF3} = 0.25 V_{DD}; Load conditions: 2 MΩ, 30 pF
Setting time	Undefined			10	μs	Load conditions: 2 MΩ, 30 pF
Analog reference voltage 2	V_{AVREF2}	0.75 V_{DD}		V_{DD}	V	
Analog reference voltage 3	V_{AVREF3}	0	0.25 V_{DD}	V		
Reference power input current 2	AI_{REF2}	0	5	mA		
Reference power input current 3	AI_{REF3}	−5.0	0	mA		
Output resistance	R_O		20		kΩ	DACS0, DACS1 set to 7FH

Interrupt Timing Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V

Item	Symbol	Min	Max	Unit	Conditions
NMI low-level width	t_{WNIL}	10		μs	
NMI high-level width	t_{WNIH}	10		μs	
INTP0-INTP5 low-level width	t_{WITL}	24		t_{CYX}	
INTP0-INTP5 high-level width	t_{WITH}	24		t_{CYX}	
$\overline{\text{RESET}}$ low-level width	t_{WRSL}	10		μs	
$\overline{\text{RESET}}$ high-level width	t_{WRSH}	10		μs	

Data Retention Characteristics

T_A = −40 to +85°C

Item	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.5		5.5	V	STOP mode
Data retention current	I_{DDDR}		2	10	μA	V_{DDDR} = 2.5 V
			5	20	μA	V_{DDDR} = 5 V ±10%
V_{DD} rise time	t_{RVD}	200			μs	
V_{DD} fall time	t_{FVD}	200			μs	
V_{DD} retention time (from STOP mode setting)	t_{HVD}	0			ms	
STOP release signal input time	t_{DREL}	0			ms	
Oscillation stabilization wait time	t_{WAIT}	30			ms	Crystal resonator
		5			ms	Ceramic resonator
Low-level input voltage	V_{IL}	0		0.1 V_{DDDR}	V	Specified pins (Note 1)
High-level input voltage	V_{IH}	0.9 V_{DDDR}		V_{DDDR}	V	Specified pins (Note 1)

Note:

(1) $\overline{\text{RESET}}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2/CI, $P2_4$/INTP3, $P2_5$/INTP4/ASCK, $P2_6$/INTP5, $P2_7$/SI, $P3_2$/SCK, $P3_3$/SO/SB0, and MODE pins.

Recommended Crystal Resonators (μPD78233/234 only)

Manufacturer	Frequency (MHz)	Part Number	C1 (pF)	C2 (pF)
Kinseki	12	HC-49/U	18	18

Recommended Ceramic Resonators (μPD78233/234 only)

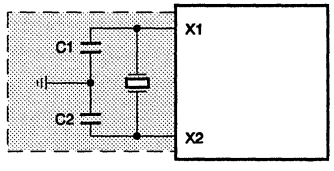
Manufacturer	Frequency (MHz)	Part Number	C1 (pF)	C2 (pF)
Murata mfg.	12	CSA12.0MT	30	30
		CST12.0MTW	None (1)	None (1)
Kyocera Corp.	12	KBR12.0M	33	33

Notes:

(1) C1 and C2 are contained in the resonator.

4d

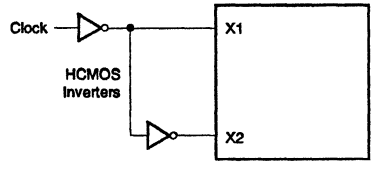
Recommended Resonator Circuit



Ceramic or crystal resonator frequency f_{XX} = 4 to 12 MHz
External oscillation circuit should be as close to the X1 and X2 pins as possible
Do not place other signal lines in the shaded area

83YL-9188A

Recommended External Clock Circuit

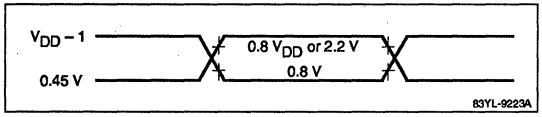


Clock frequency f_{XX} = 4 to 12 MHz
In STOP mode, X1 is internally shorted to V_{SS} to prevent leakage current. Therefore, STOP mode is not available when using this external clock circuit.

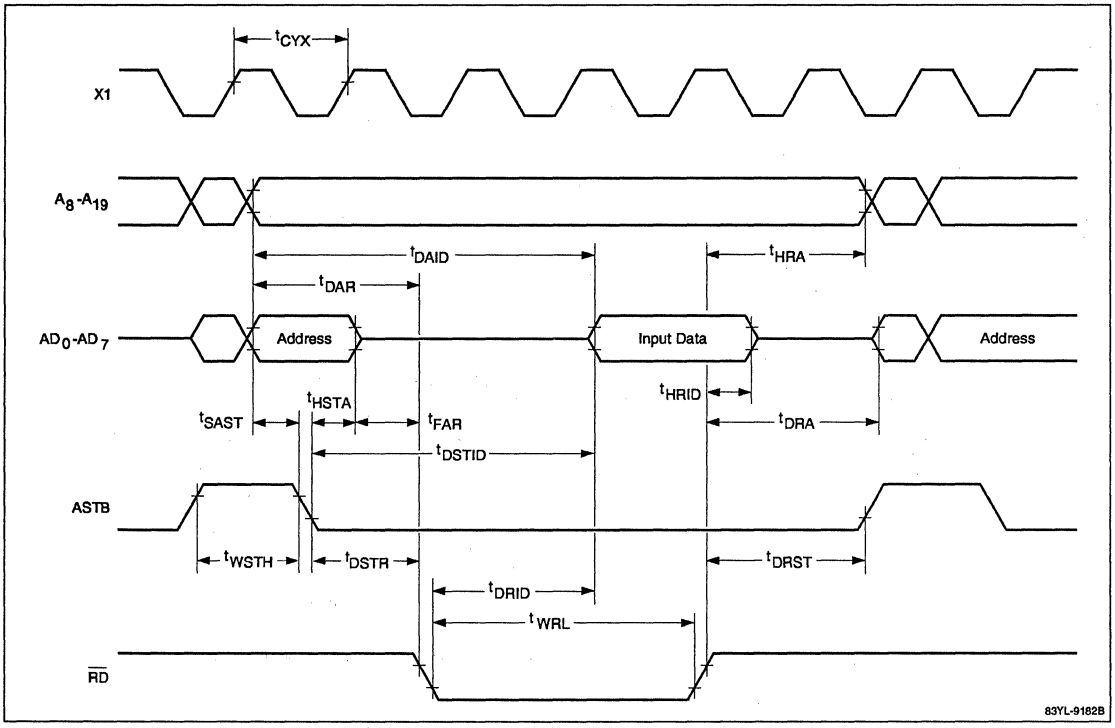
83YL-9190A

Timing Waveforms

Voltage Thresholds for AC Timing Measurements

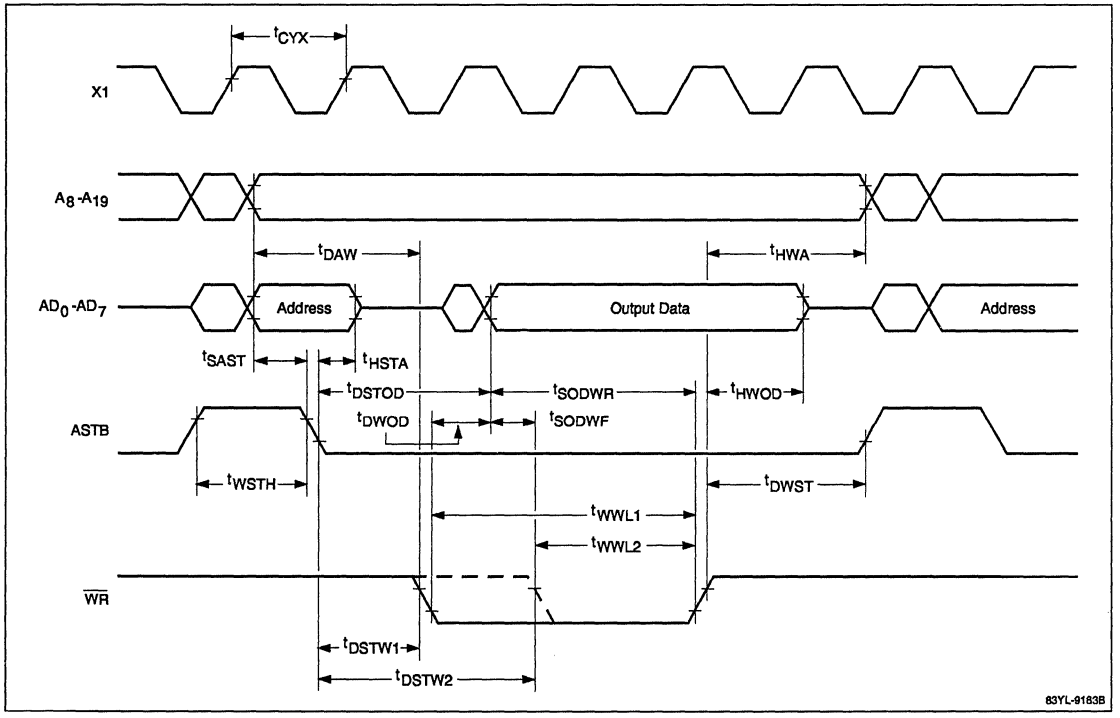


Read Operation



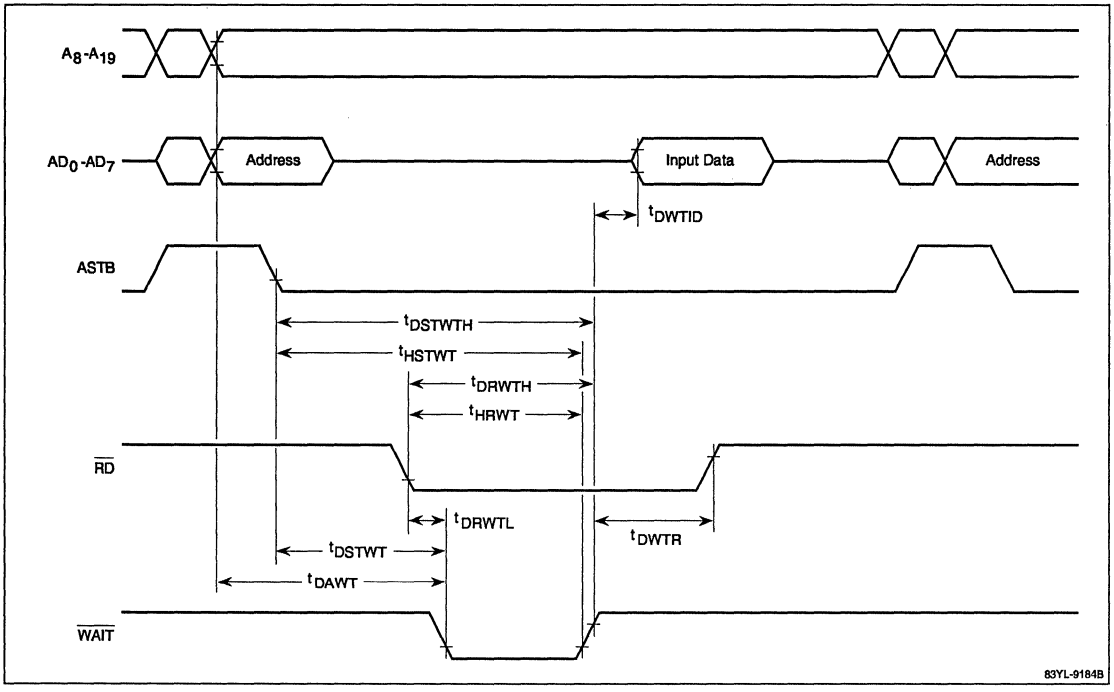
Timing Waveforms (cont)

Write Operation



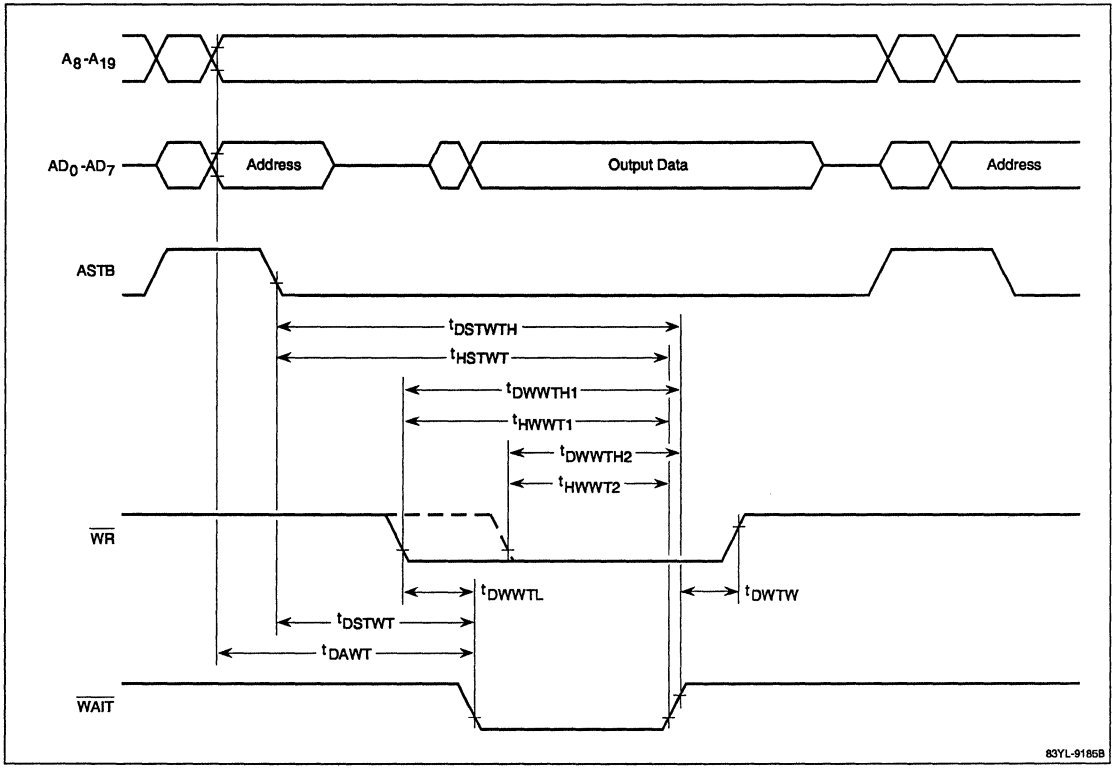
Timing Waveforms (cont)

External $\overline{WAIT}$ Signal Input (Read Operation)



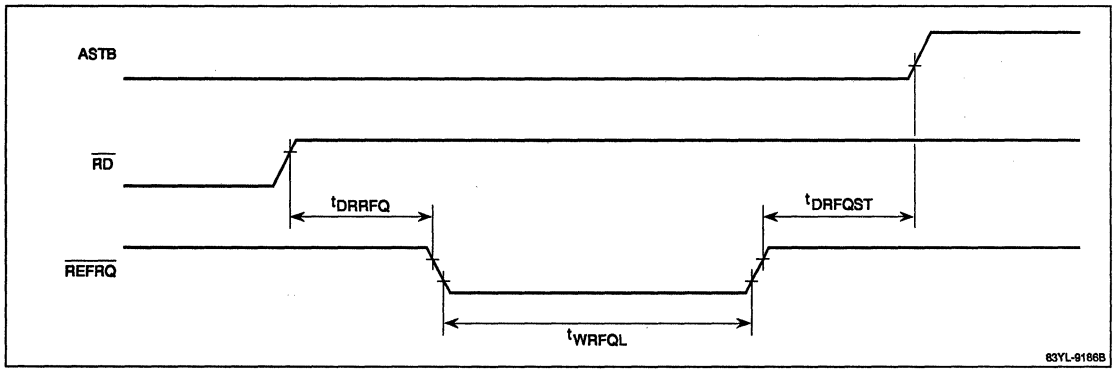
Timing Waveforms (cont)

External $\overline{WAIT}$ Signal Input (Write Operation)

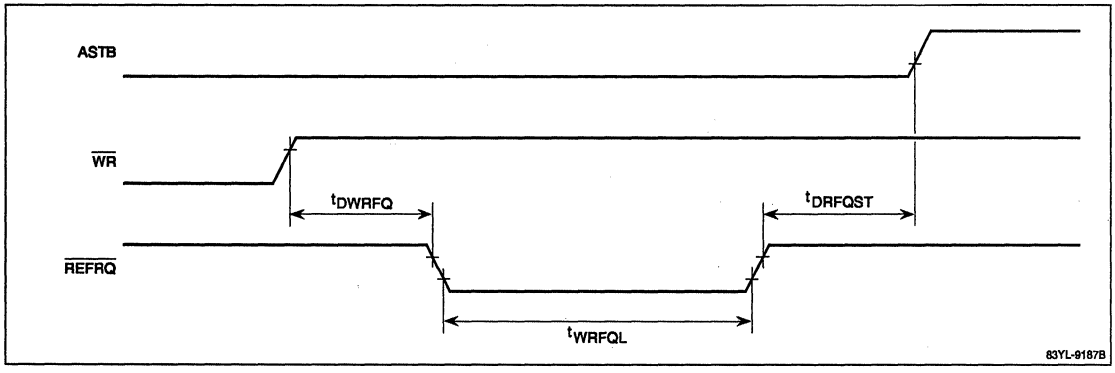


Timing Waveforms (cont)

Refresh After Read



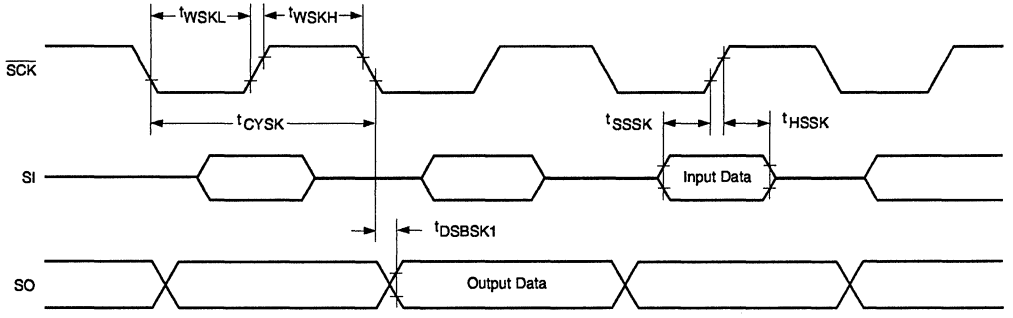
Refresh After Write



Timing Waveforms (cont)

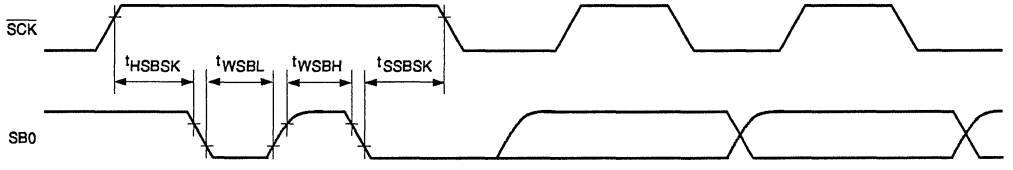
Serial Operation

Three-Line Serial I/O Timing

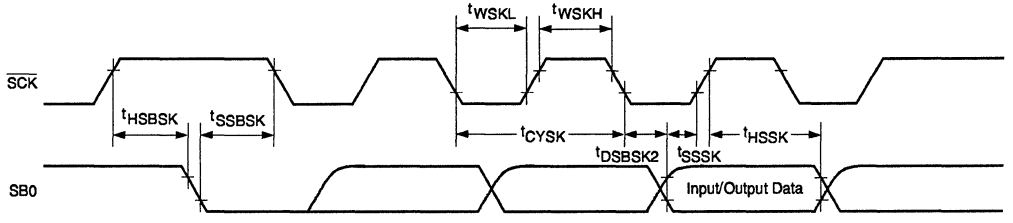


SBI Mode

Bus Release Signal Transfer Timing



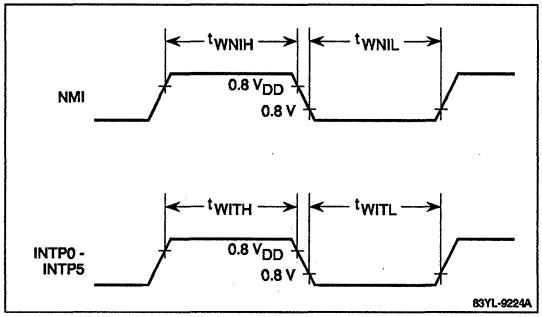
Command Signal Transfer Timing



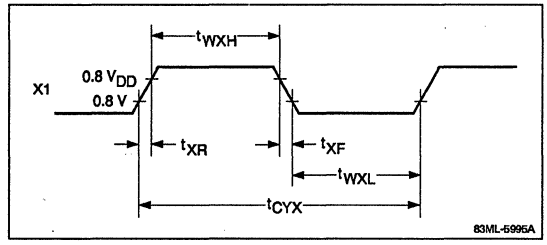
83YL-9168B

4d

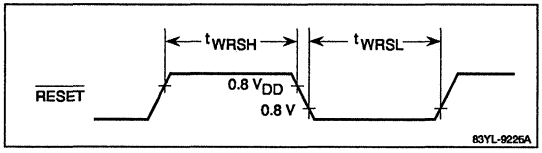
Interrupt Input



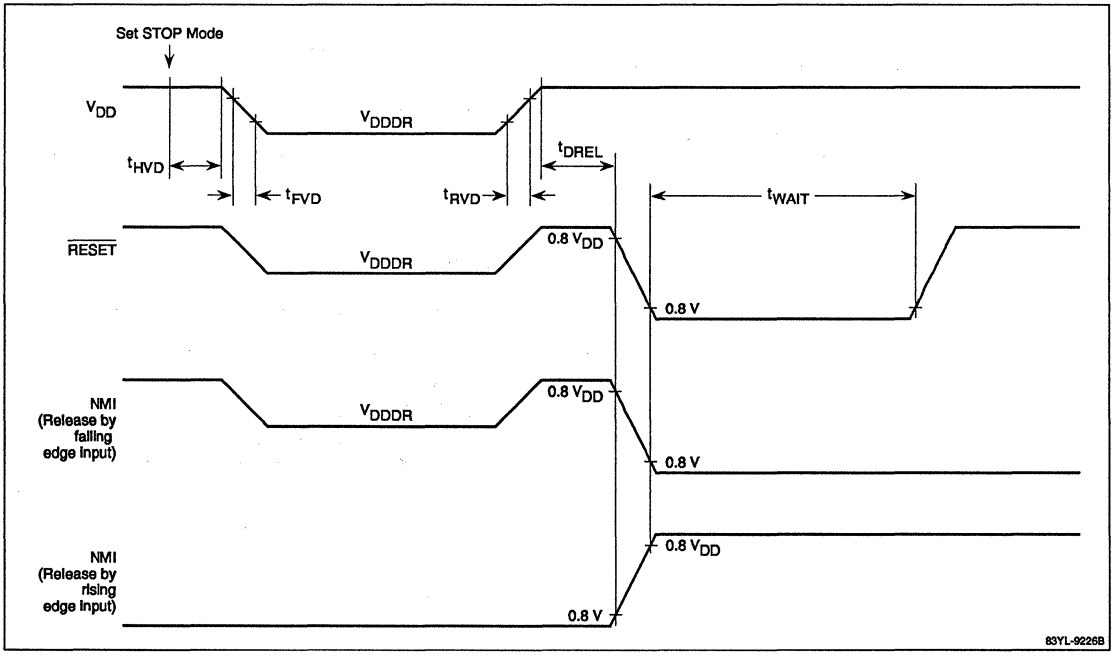
External Clock



Reset Input



Data Retention Characteristics



μPD78P238 PROGRAMMING

In the μPD78P238, the mask ROM of μPD78234 is replaced by a one-time programmable ROM (OTP ROM) or a reprogrammable, ultraviolet erasable ROM (UV EPROM). The ROM is 32K x 8 bits and can be programmed using a general-purpose PROM writer with a μPD27C256A programming mode.

The PA-78P238GC/GJ/LQ/KF are the socket adaptors used for configuring the μPD78P238 to fit a standard PROM socket.

Refer to tables 5 and 6 and figures 17 through 19 for special information applicable to PROM programming.

Table 5. Pin Functions During PROM Programming

Pin	Pin*	Function
$P0_0$ - $P0_7$	A_0 - A_7	Address input pins for PROM operations
$P5_0/A_8$	A_8	Address input pin for PROM operations
$P2_1$/INTP0	A_9	Address input pin for PROM operations
Pin	Pin*	Function

Table 5. Pin Functions During PROM Programming (cont)

Pin	Pin*	Function
Table 5. Pin Functions During PROM Programming		
Pin	Pin*	Function
$P5_2/A_{10}$ - $P5_6/A_{14}$	A_{10} - A_{14}	Address input pins for PROM operations
$P4_0/AD_0$ - $P4_7/AD_7$	D_0 - D_7	Data pins for PROM operations
$P6_5/\overline{WR}$	$\overline{CE}$	Strobes data into the PROM
$P6_4/\overline{RD}$	$\overline{OE}$	Enables a data read from the PROM
$\overline{RESET}$	$\overline{RESET}$	PROM programming mode requires applying a low voltage to this pin
MODE	V_{PP}	High voltage applied to this pin for program write/verify
V_{DD}	V_{DD}	Positive power supply pin
V_{SS}	V_{SS}	Ground

* Pin name in PROM programming mode.

4d

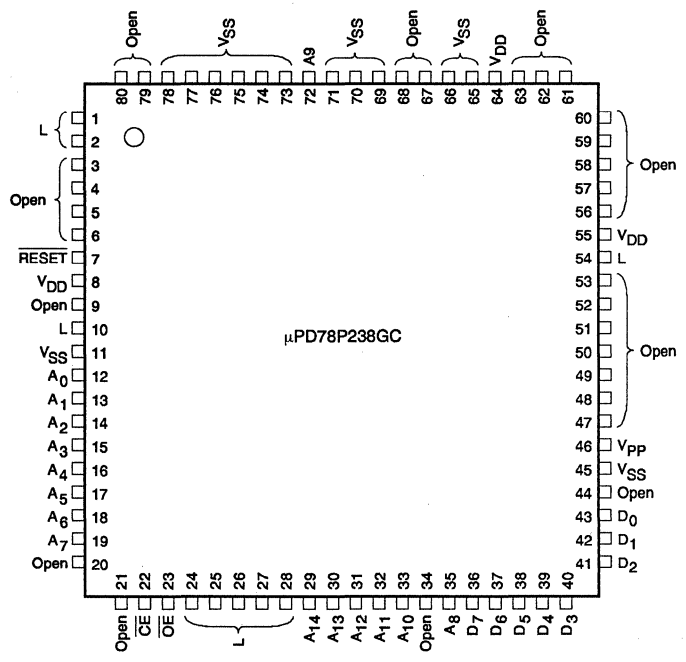
Table 6. Summary of Operation Modes for PROM Programming

Mode	$\overline{RESET}$	$\overline{CE}$	$\overline{OE}$	V_{PP}	V_{DD}	D_0 - D_7
Program write	L	L	H	+12.5 V	+6 V	Data input
Program verify	L	H	L	+12.5 V	+6 V	Data output
Program inhibit	L	H	H	+12.5 V	+6 V	High Z
Read out	L	L	L	+5 V	+5 V	Data output
Output disable	L	L	H	+5 V	+5 V	High Z
Standby	L	H	L/H	+5 V	+5 V	High Z

Note: When +12.5 V is applied to V_{PP} and +6 V to V_{DD}, both $\overline{CE}$ and $\overline{OE}$ cannot be set to low level (L) simultaneously.

Pin Functions in μPD78P238 PROM Programming Mode

Figure 17. 80-Pin Plastic QFP



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors.

(2) V_{SS}: Connect these pins to V_{SS}.

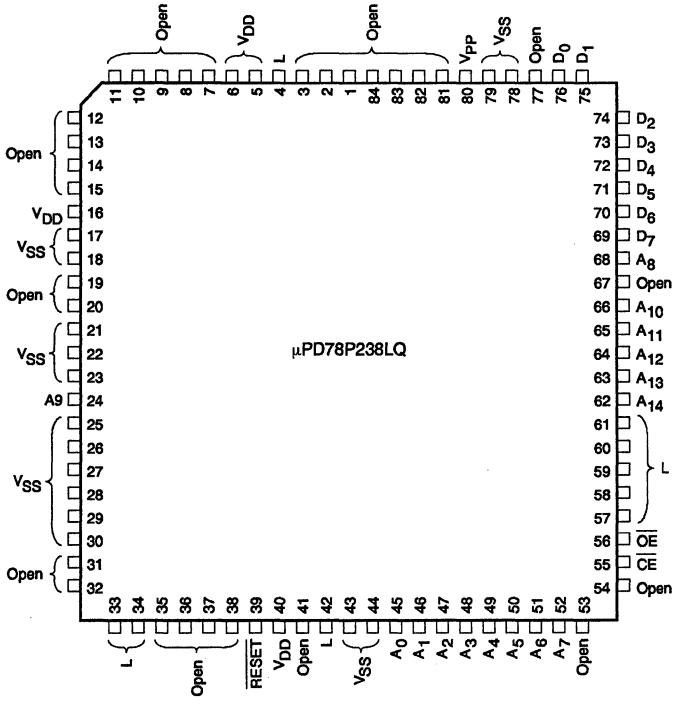
(3) Open: Do not connect these pins.

(4) $\overline{RESET}$: Set to a low level.

83YL-9207B

Pin Functions in μPD78P238 PROM Programming Mode (cont)

Figure 18. 84-Pin PLCC



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors.

(2) V_{SS}: Connect these pins to V_{SS}.

(3) Open: Do not connect these pins.

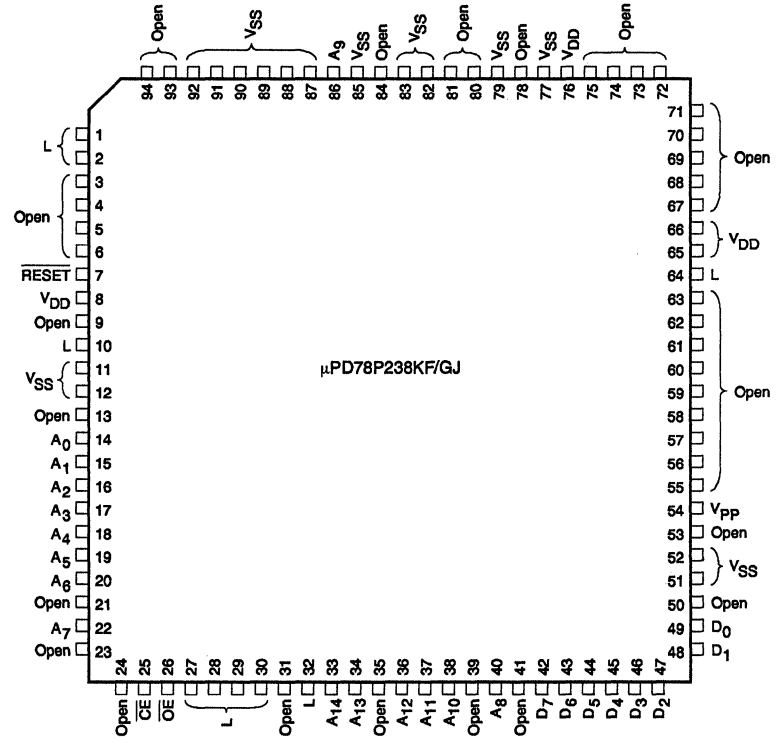
(4) $\overline{RESET}$: Set to a low level.

83YL-9208B

4d

Pin Functions in μPD78P238 PROM Programming Mode (cont)

Figure 19. 94-Pin Plastic QFP 94-Pin Ceramic LCC with Window



Notes:

(1) L: Connect these pins separately to V_{SS} through resistors.

(2) V_{SS}: Connect these pins to V_{SS}.

(3) Open: Do not connect these pins.

(4) $\overline{RESET}$: Set to a low level.

83YL-9209B

PROM Write Procedure

(1) Set the pins not used for programming as indicated in figures 17 through 19. Connect the $\overline{RESET}$ pin to a low level and apply +5 V to the V_{DD} and V_{PP} pin. The $\overline{CE}$ and $\overline{OE}$ pins should be high.

(2) Apply +6 V to the V_{DD} pin and +12.5 V to the V_{PP} pin.

(3) Provide the initial address to the A_0 to A_{14} pins.

(4) Provide write data.

(5) Provide 1-ms program pulse (active low) to the $\overline{CE}$ pin.

(6) This data is now verified with a pulse (active low) to the $\overline{OE}$ pin. If the data has been written, proceed to step 8; if not, repeat steps 4 to 6. If the data cannot be correctly written after 25 attempts, go to step 7.

(7) Classify as defective and stop write operation.

(8) Provide write data and supply program pulse (for additional writing) for 3 ms times the number of writes performed in step 5.

(9) Increment the address.

(10) Repeat steps 4 to 9 until the end address.

PROM Read Procedure

(1) Set the pins not used for programming as indicated in figures 17 through 19. Fix the $\overline{RESET}$ pin to a low level and apply +5 V to the V_{DD} and V_{PP} pin. The $\overline{CE}$ and $\overline{OE}$ pins should be high.

(2) Input the address of the data to be read to pins A_0 - A_{14}.

(3) Read mode is entered with a pulse (active low) on both the $\overline{CE}$ and $\overline{OE}$ pins.

(4) Data is output to the D_0 to D_7 Pins.

EPROM Erasure

Data in an EPROM is erased by exposing the quartz window in the ceramic package to light having a wavelength shorter than 400 nm, including ultraviolet rays, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15 Ws/cm² (ultraviolet ray intensity x exposure time) is required to completely erase written data. Erasure by an ultraviolet lamp rated at 12 mW/cm² takes approximately 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

4d

DC Programming Characteristics

T_A = 25 ±5°C; $V_{PP} \geq 4.5$ V; V_{SS} = 0 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH}	V_{IH}	2.4		V_{DDP}+0.3	V	
Low-level input voltage	V_{IL}	V_{IL}	−0.3		0.8	V	
Input leakage current	I_{LIP}	I_{LI}			10	μA	$0 \leq V_I \leq V_{DDP}$
High-level output voltage	V_{OH1}	V_{OH1}	2.4			V	I_{OH} = −400 μA
	V_{OH2}	V_{OH2}	V_{DD}−0.7			V	I_{OH} = −100 μA
Low-level output voltage	V_{OL}	V_{OL}			0.45	V	I_{OH} = 2.1 mA
Output leakage current	I_{LO}				10	μA	$0 \leq V_O \leq V_{DDP}$, $\overline{OE}$ = V_{IH}
V_{DDP} power voltage	V_{DDP}	V_{CC}	5.75	6.0	6.25	V	Program memory write mode
			4.5	5.0	5.5	V	Program memory read mode
V_{PP} power voltage	V_{PP}	V_{PP}	12.2	12.5	12.8	V	Program memory write mode
				V_{PP} = V_{DDP}		V	Program memory read mode
V_{DDP} power current	I_{DDP}	I_{CC}		5	30	mA	Program memory write mode
				5	30	mA	Program memory read mode $\overline{CE}$ = V_{IL}, V_I = V_{IH}
V_{PP} power current	I_{PP}	I_{PP}		5	30	mA	Program memory write mode $\overline{CE}$ = V_{IL}, $\overline{OE}$ = V_{IH}
				1	100	μA	Program memory read mode

* Corresponding symbols of the μPD27C256A.

AC Programming Characteristics (Write Mode)

T_A = 25 ±5°C; V_{PP} ≥ 4.5 V; V_{SS} = 0 V; V_{DD} = 6 ±0.25 V; V_{PP} = 12.5 ±0.3 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Conditions
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}	2			μs	
Data input to $\overline{OE}$ ↓ delay time	t_{HOID}	t_{OES}	2			μs	
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}	2			μs	
Address hold time from $\overline{CE}$ ↑	t_{HCA}	t_{AH}	2			μs	
Input data hold time from $\overline{CE}$ ↑	t_{HCID}	t_{DH}	2			μs	
Output data hold time from $\overline{OE}$ ↑	t_{HOOD}	t_{DF}	0		130	ns	
V_{PP} setup time to $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}	1			ms	
V_{DDP} setup time to $\overline{CE}$ ↓	t_{SVDC}	t_{VCS}	1			ms	
Initial program pulse width	t_{WL1}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{WL2}	t_{OPW}	2.85		78.75	ms	
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			150	ns	

* Corresponding symbols of the μPD27C256A.

AC Programming Characteristics (Read Mode)

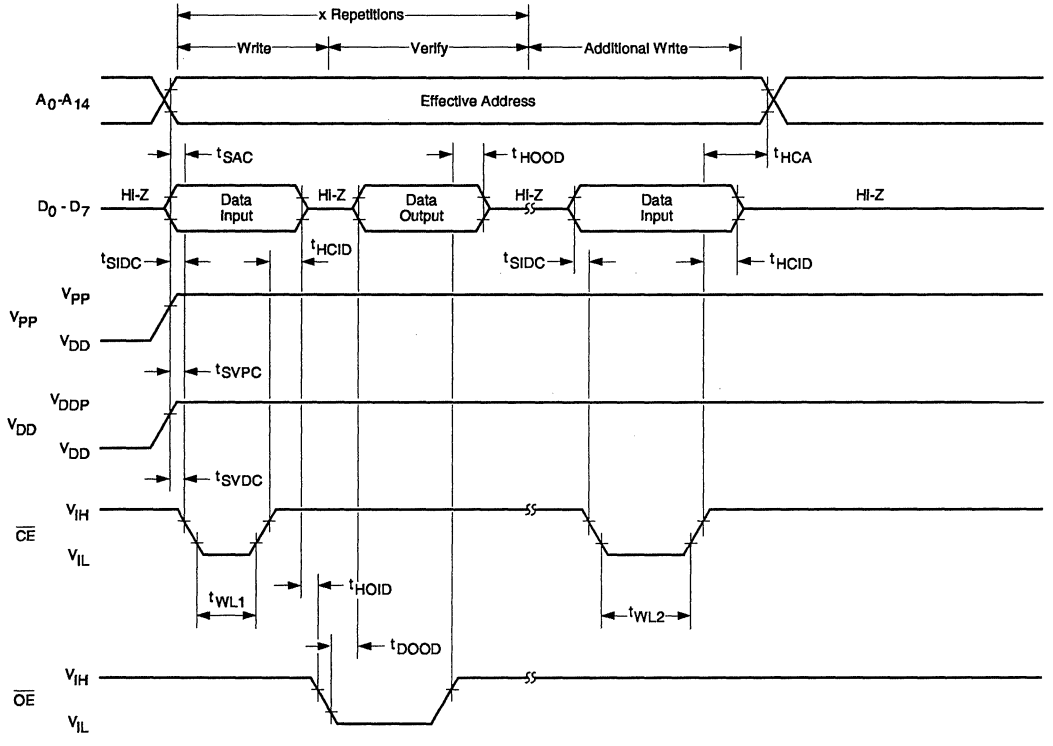
T_A = 25 ±5°C; V_{PP} ≥ 4.5 V; V_{PP} = 5 ±0.5 V; V_{PP} = V_{DDP}; V_{SS} = 0 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Conditions
Address to data output time	t_{DAOD}	t_{ACC}			200	ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}
$\overline{CE}$ ↓ to data output time	t_{DCOD}	t_{CE}			200	ns	$\overline{OE}$ = V_{IL}
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			75	ns	$\overline{CE}$ = V_{IL}
Data hold time from $\overline{OE}$ ↑	t_{HCOD}	t_{DF}	0		60	ns	$\overline{CE}$ = V_{IL}
Data hold time from address	t_{HAOD}	t_{OH}	0			ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}

* Corresponding symbols of the μPD27C256A.

PROM Timing Diagrams

PROM Write/Verify Mode



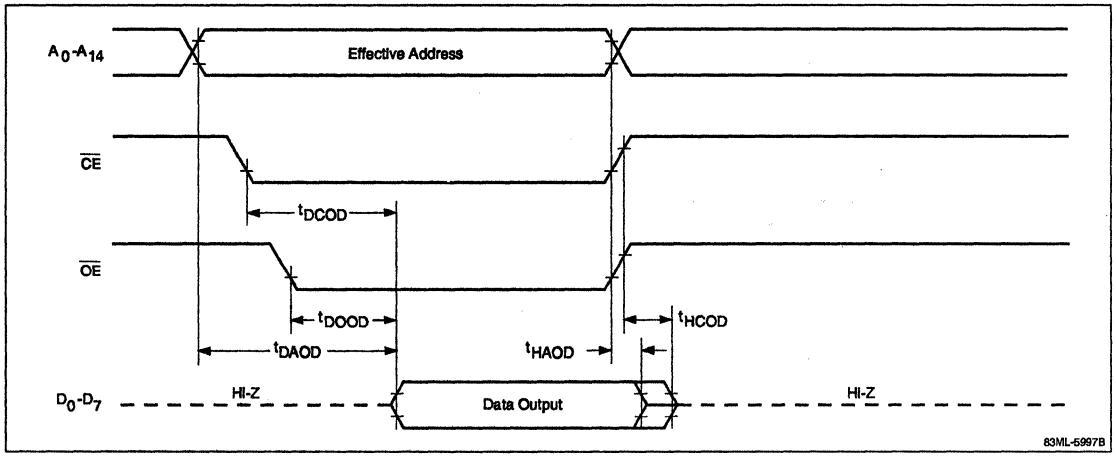
Notes:
(1) V_{DD} must be applied before applying V_{PP}. It should be removed after removing V_{PP}.
(2) V_{PP} must not exceed +13V, including overshoot.

83YL-9210B

4d

PROM Timing Diagrams (cont)

PROM Read Mode



NEC
NEC Electronics Inc.

μPD78244 Family (μPD78243/244) 8-Bit, K-Series Microcontrollers With A/D Converter, EEPROM, Real-Time Output Ports

September 1993

Description

The μPD78243 and μPD78244 are members of the K-Series® of microcontrollers and are designed for real-time embedded control applications. The μPD78244 family is pin compatible with the μPD78214 and μPD78218A families and offers 512 bytes of EEPROM (electrically erasable programmable read-only memory) with enhanced timer and macro service facilities. These 8-bit, single-chip microcontrollers have a minimum instruction time of 333 ns at 12 MHz (500 ns for the μPD78243). They feature 8-bit hardware multiply and divide instructions, four banks of main registers, an advanced interrupt handling facility, a powerful set of memory mapped on-chip peripherals, and the ability to address up to 1M bytes of external data memory. On-board memory includes 512 bytes of RAM, 512 bytes of EEPROM, and 16K bytes of mask ROM.

The advanced interrupt handling facility provides two levels of programmable hardware priority control and two separate methods of servicing interrupt requests: vectored and macro service. The macro service facility reduces the overhead involved in servicing peripheral interrupts by transferring data between the memory-mapped special function registers (SFRs) and memory without the use of time consuming interrupt service routines. In addition, the macro service facility can be initialized to automatically alter timer compare register values or to repeatedly output a prespecified pattern at a fixed or variable rate. By using macro service to control the real-time output ports, the μPD78244 can easily and accurately drive two independent stepper motors.

The combination of the macro service facility, four banks of main registers, EEPROM, extended data memory address space, and powerful on-chip peripherals makes these devices ideal for applications in office automation, communication, HVAC, and industrial control.

NEC is manufacturing and selling this product (with on-chip EEPROM) under microcomputer patent license with BULL CP8. This product should not be used in IC cards (SMART CARD).

Features

- Complete single-chip microcontroller
  - 8-bit ALU
  - Program memory (ROM)
    - μPD78243: ROMless
    - μPD78244: 16K bytes
  - Data memory (RAM)
    - RAM: 512 bytes
    - EEPROM: 512 bytes
- Pin compatible with μPD78214 and μPD78218A families
- Powerful instruction set
  - 8-bit unsigned multiply and divide
  - 16-bit arithmetic instructions
  - 1-bit and 8-bit logic instructions
- Minimum instruction time
  - 333 ns at 12 MHz (μPD78244)
  - 500 ns at 12 MHz (μPD78243)
- Memory expansion
  - 8085 bus-compatible
  - 64K program address space
  - 1M data address space
- Large I/O capacity
  - Up to 54 I/O port lines on μPD78244
  - Up to 36 I/O port lines on μPD78243
  - Software programmable pullup resistors
- Memory-mapped on-chip peripherals (special function registers)
- Timer/counter unit
  - 16-bit timer 0:
    - Two 16-bit compare registers
    - One 16-bit capture register
    - One external interrupt/capture line
  - 8-bit timer 1:
    - One 8-bit compare register
    - One 8-bit capture/compare register
    - One external interrupt/capture line
  - 8-bit timer/counter 2:
    - Two 8-bit compare registers
    - One 8-bit capture register
    - One external interrupt/capture line
    - One external event counter line
  - 8-bit timer 3:
    - One 8-bit compare register
- Four 8-bit precision timer-controlled pulse-width modulated (PWM) output lines

Features (cont)

- Two 4-bit (or one 8-bit) real-time output ports
- Eight-channel 8-bit A/D converter
- Programmable priority interrupt controller (two levels)
- Two methods of interrupt service
  - — Vectored interrupts
  - — Macro service mode with choice of three different types
- Two-channel serial communication interface
  - — Asynchronous serial interface (UART)
    Dedicated baud rate generator
  - — Clock-synchronized interface
    Full-duplex, three-wire mode
    NEC serial bus interface (SBI) mode
- Refresh output for pseudostatic RAM
- STOP and HALT standby functions
- 5-volt CMOS technology

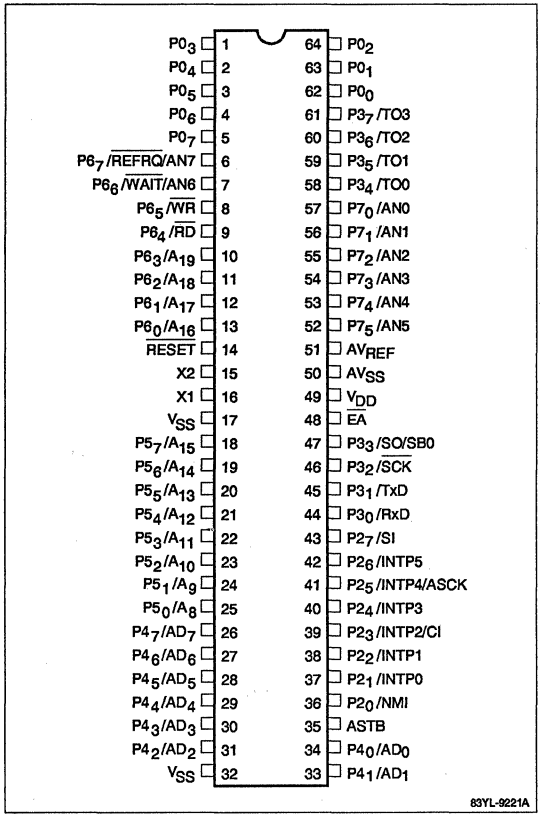
Ordering Information

Part Number	Package (Package Dwg.)	ROM
μPD78243CW	64-pin plastic shrink DIP (P64C-70-750A,C)	ROMless
μPD78243GC-AB8	64-pin plastic QFP (P64GC-80-AB8-2)	
μPD78244CW-xxx	64-pin plastic shrink DIP (P64C-70-750A,C)	16K mask ROM
μPD78244GC-xxx	64-pin plastic QFP (P64GC-80-AB8-2)	

xxx indicates ROM code suffix

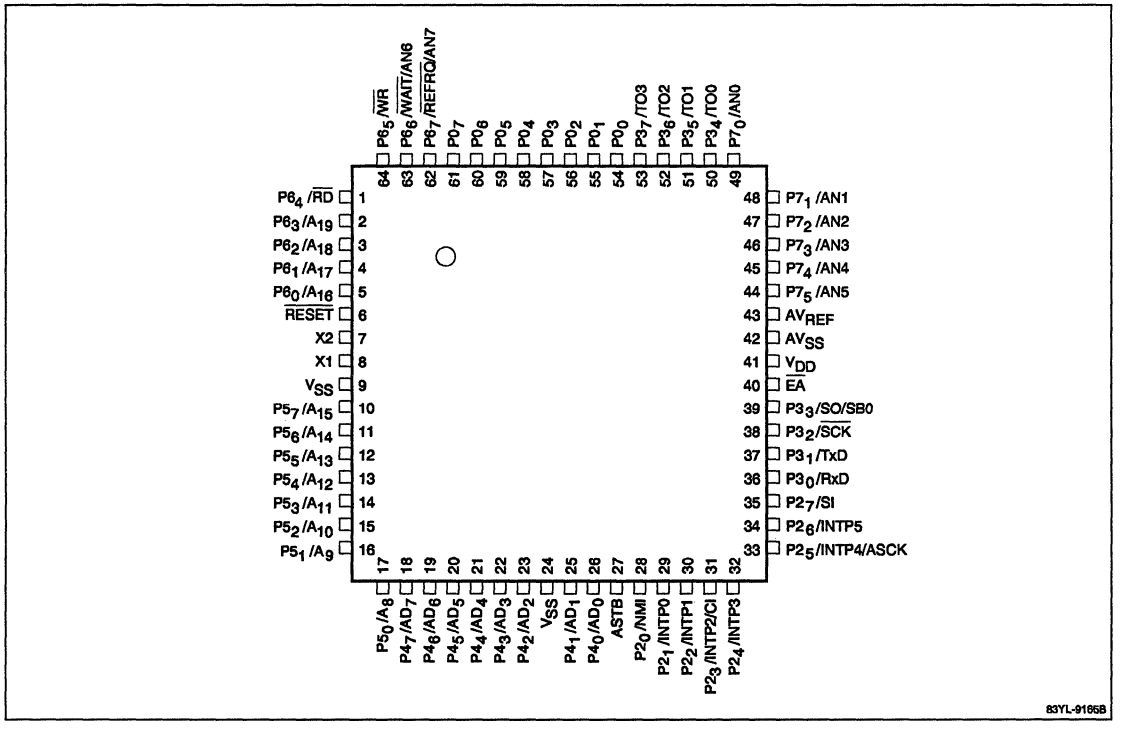
Pin Configurations

64-Pin Shrink DIP (Plastic)



Pin Configurations (cont)

64-Pin Plastic QFP

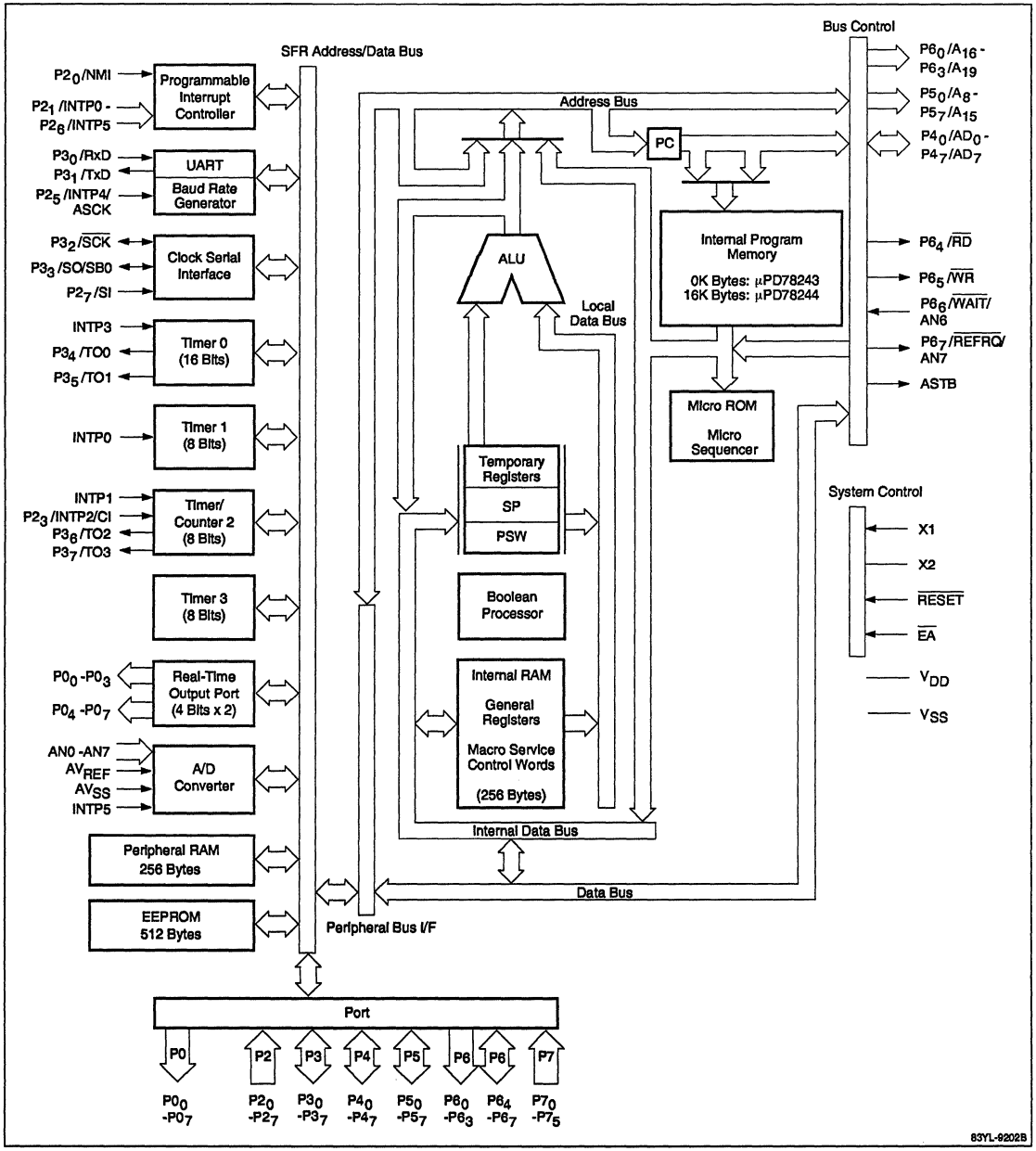


4e

Pin Functions; Normal Operating Mode

Symbol	First Function	Symbol	Second Function
$P0_0$ - $P0_7$	Port 0; 8-bit tristate output port/real time output port		
$P2_0$	Port 2; 8-bit input port	NMI	External nonmaskable interrupt
$P2_1$		INTP0	Maskable external interrupt
$P2_2$		INTP1	
$P2_3$		INTP2	Maskable external interrupt
		CI	External clock input to timer/counter 2
$P2_4$		INTP3	Maskable external interrupt
$P2_5$		INTP4	Maskable external interrupt
		ASCK	Asynchronous serial clock input
$P2_6$		INTP5	Maskable external interrupt
$P2_7$		SI	Serial data input for three-wire serial I/O mode
$P3_0$	Port 3; 8-bit, bit-selectable tristate input/output port	RxD	Asynchronous serial receive data input
$P3_1$		TxD	Asynchronous serial transmit data output
$P3_2$		$\overline{SCK}$	Serial shift clock input/output
$P3_3$		SO	Serial data output for three-wire serial I/O mode
		SB0	I/O bus for NEC serial bus interface (SBI)
$P3_4$ - $P3_7$		TO0 - TO3	Timers T0 to T3 outputs
$P4_0$ - $P4_7$	Port 4; 8-bit tristate input/output port	AD_0 - AD_7	Low-order 8-bit multiplexed address/data bus
$P5_0$ - $P5_7$	Port 5; 8-bit, bit-selectable tristate input/output port	A_8 - A_{15}	High-order 8-bit address bus
$P6_0$ - $P6_3$	Port 6; 4-bit output port	A_{16} - A_{19}	Extended memory address bus
$P6_4$	Port 6; 4-bit, bit-selectable tristate input/output port	$\overline{RD}$	External memory read strobe output
$P6_5$		$\overline{WR}$	External memory write strobe output
$P6_6$		$\overline{WAIT}$	External memory wait signal input
		AN6	Analog voltage input to A/D converter
$P6_7$		$\overline{REFRQ}$	Refresh pulse output used by external pseudostatic memory
		AN7	Analog voltage input to A/D converter
$P7_0$ - $P7_5$	Port 7; 6-bit input port	AN0 - AN5	Analog voltage inputs to A/D converter
ASTB	Address strobe output used to latch the low-order 8 address for external memory		
$\overline{RESET}$	External system reset input		
$\overline{EA}$	Internal ROM or external memory control signal input. Low-level input selects external memory. High-level input selects internal ROM. A low-level input on a μPD78244 places the device in ROMless mode and external memory is accessed.		
X1	Crystal/ceramic resonator connection or external clock input		
X2	Crystal/ceramic resonator connection or inverse of external clock		
AV_{REF}	A/D converter reference voltage		
AV_{SS}	A/D converter ground		
V_{DD}	+5 volt power supply input		
V_{SS}	Power supply ground		
NC	No connection		

Block Diagram



4e

Differences Among the μPD78244, μPD78218A, and μPD78214

The μPD78244 family is a pin compatible enhanced version of the μPD78218A and μPD78214 families. Table 1 highlights the differences.

Table 1. Differences Among the μPD78244, μPD78218A and μPD78214 Families

Item	μPD78244 Family	μPD78218A Family	μPD78214 Family
Maximum on-chip ROM	16K bytes	32K bytes	16K bytes
Maximum on-chip RAM	512 bytes	1024 bytes	512 bytes
On-chip EEPROM	512 bytes	None	None
16-bit timer	Software-triggered one-shot pulse output	Software-triggered one-shot pulse output	Not available
Internal interrupt sources	14	12	12
Macro service counter	8/16 bit selectable (except Type A transfers)	8/16 bit selectable (except Type A transfers)	8-bit only
Type C macro service pointers, MPT and MPD	Increments full 16 bits	Increments full 16 bits	Increments only lower 8 bits
Macro service execution times	Same	Same	Different; refer to hardware user's manual
PUSH PSW instruction execution times	Same	Same	Different; refer to software user's manual
Oscillation stabilization time when exiting STOP mode	Time equivalent to NMI active pulse width plus 16 bits of dedicated counter or 15 bits of dedicated counter	Time equivalent to NMI active pulse width plus 16 bits of dedicated counter or 15 bits of dedicated counter	Time equivalent to NMI active pulse width plus 16 bits of dedicated counter
A/D converter reference voltage	3.6 V to V_{DD}	3.6 V to V_{DD}	3.4 V to V_{DD}
Operating temperature	−10 to +70°C	−40 to +85°C	−40 to +85°C
Programmable device operating voltage	No programmable device	5 V ±0.3 V	5 V ±10%
Package	64-pin plastic shrink DIP	64-pin plastic shrink DIP	64-pin plastic shrink DIP
	64-pin plastic QFP	64-pin plastic QFP	64-pin plastic QFP
		64-pin shrink cerdip w/window	64-pin shrink cerdip w/window
			64-pin plastic QUIP
			68-pin PLCC
			74-pin plastic QFP

FUNCTIONAL DESCRIPTION

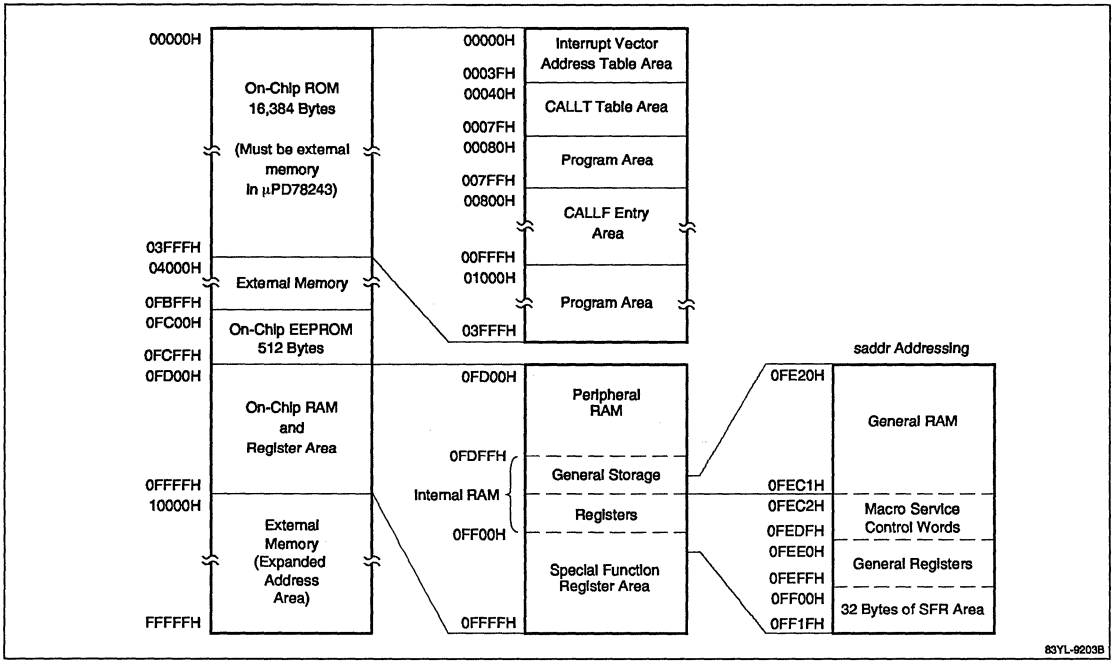
Central Processing Unit (CPU)

The μPD78244 CPU features 8- and 16-bit arithmetic including an 8 x 8-bit unsigned multiply and 16 x 8-bit unsigned divide (producing a 16-bit quotient and an 8-bit remainder). The multiply executes in 3.67 μs and the divide in 12.36 μs at 12 MHz (4.00 and 12.69 μs respectively for μPD78243).

A CALLT vector table and a CALLF program area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction (CALLT) can access up to 32 subroutines through the addresses contained in the CALLT vector table. A 2-byte call instruction (CALLF) can access any routine beginning at a specific address in the CALLF area.

The internal system clock (f_{CLK}) is generated by dividing the oscillator frequency by two. Therefore, at the maximum oscillator frequency of 12 MHz, the internal system clock is 6 MHz. The minimum instruction exe-

Figure 1. Memory Map



cution time for an instruction fetched from internal ROM is 333 ns (500 ns when fetched from external memory).

Memory Space

The μPD78244 family has a 1M byte address space (see figure 1). The first 64K bytes of this address space (00000H-0FFFFH) can be used as both program and data memory. The remaining 960K bytes of this address space (10000H-FFFFFH) can only be used as data memory and is known as expanded memory.

External Memory

The μPD78244 family has an 8-bit wide external data bus and a 16-bit wide external address bus (20-bit wide if expanded memory is enabled). The low-order 8 bits of the address bus are multiplexed to provide the 8-bit data bus and are supplied by I/O port 4. The high-order address bits of the 16-bit address bus are taken from port 5. If expanded memory is enabled, the expanded address nibble is provided by $P6_0$ to $P6_3$. Address latch, read, and write strobes are also provided.

The memory expansion mode register (MM) is used to enable external memory, to specify up to two additional wait states or the use of the $\overline{WAIT}$ input pin for the first 64K bytes of memory, and to enable the high-speed internal ROM fetch. Ports 4, 5, and 6 are available as general purpose I/O ports when only internal ROM is used and no external program or data space is required.

Expanded Data Memory

The MM register is also used to enable the external expanded data memory space, addresses 10000H to FFFFFH. When the expanded data memory is enabled, the entire 1M byte address space is divided into 16 banks of 64K bytes each. The low-order 4-bits of the P6 or the PM6 registers are used as bank selection registers to supply the address information to A_{16} to A_{19}. Data can easily be transferred from one memory bank to another by using the appropriate instructions. Address lines A_{16} to A_{19} are only active when an instruction that uses expanded addressing is being executed. A programmable wait control register (PW) allows the programmer to specify up to two additional wait states or the use of the $\overline{WAIT}$ input pin for expanded data memory space.

4e

On-Chip RAM

The μPD78244 family has a total of 512 bytes of on-chip RAM. The upper 256-byte area (FE00H-FEFFH) features high-speed access and is known as "Internal RAM." The remaining 256 bytes (FD00H-FDFFH) are accessed at the same speed as external memory and is known as "Peripheral RAM." The general register banks and the macro service control words are stored in Internal RAM. The remainder of Internal RAM and any unused register bank locations are available for general storage.

On-Chip EEPROM

The μPD78244 family has 512 bytes of on-chip EEPROM (FB00H-FCFFH) usable as internal data memory. A user program can read from it or write to it using most of the same instructions available for use with internal RAM. Any address of the EEPROM can be read at any time even during a write cycle at the same address. Data is read from EEPROM at the same speed as from peripheral RAM and the EEPROM is guaranteed for 100,000 repetitive rewrites.

Each write cycle, consisting of an auto erase cycle followed by an auto write cycle, takes approximately 10 ms to complete. On completion of a write cycle, an EEPROM write termination interrupt (INTEPW) is generated. If another write cycle is started before the previous cycle is finished, an EEPROM write error interrupt (INTEER) will be generated. Two status flags are also available to indicate whether a write operation is currently in progress and whether a write error has occurred.

The EEPROM can be write protected in 128-byte blocks. The area to be protected can be assigned and enabled only once following reset. It cannot be changed once it has been set by the program.

On-Chip Program Memory

The μPD78244 contains 16K bytes of internal ROM. Instructions from on-chip program memory can be fetched at high speed or at the same rate as from external memory. The μPD78243 does not have on-chip program memory.

CPU Control Registers

Program Counter. The program counter is a 16-bit binary counter register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 0001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

Program Status Word. The program status word (PSW) is an 8-bit register that contains flags that are set or reset depending on the results of an instruction. This register can be written to or read from 8 bits at a time. The individual flags can also be manipulated on a bit-by-bit basis. The assignment of PSW bits follows.

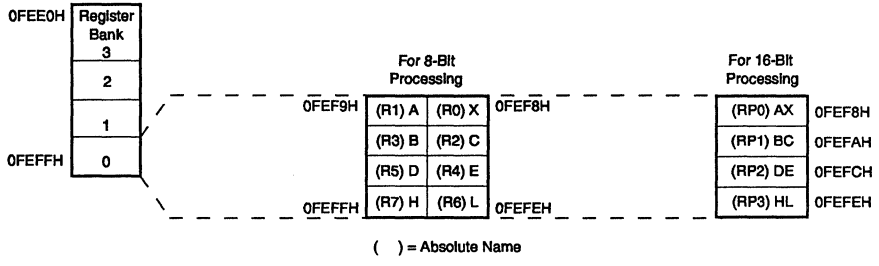
7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

CY	Carry flag
ISP	Interrupt priority status flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

General Registers

The general-purpose registers (figure 2) consist of four banks of registers located at addresses FEE0H to FEFFH in Internal RAM. Each bank consists of eight 8-bit general registers that can also be used in pairs to function as four 16-bit registers. Two bits in the PSW (RBS0 and RBS1) specify which of the register banks is active. The bits are set under program control. Registers have both functional names (like A, X, B, C for 8-bit registers and AX, BC for 16-bit registers) and absolute names (like R1, R0, R3, R2 for 8-bit registers and RP0, RP1 for 16-bit registers). Each instruction determines whether a register is referred to by its functional or absolute name and whether it is 8 or 16 bits.

Figure 2. General Registers



Addressing

The μPD78244 family features 1-byte addressing of both the special function registers and the portion of on-chip RAM from FE20H to FEFFH. The 1-byte sfr addressing accesses the entire SFR area, while the 1-byte saddr addressing accesses the first 32 bytes of the SFR area and 224 bytes of Internal RAM. The 16-bit SFRs and words of memory in these areas can be addressed by 1-byte saddrp addressing, which is valid for even addresses only. Since many instructions use 1-byte addressing, access to these locations is almost as fast and versatile as access to the general registers. There are seven addressing modes for data in main memory: direct, register, register indirect with autoincrement and decrement, saddr, SFR, based, and indexed. There are also both 8-bit and 16-bit immediate operands.

Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and the CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by 1-byte sfr addressing. They are either 8 or 16 bits as required, and many of the 8-bit registers are capable of single-bit access as well. Locations FFD0H through FFDFH are known as the external SFR area. Registers in external circuitry interfaced and mapped to these addresses can be addressed with SFR addressing. Table 2 is a list of the special function registers.

Table 2. Special Function Registers

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
0FF00H	Port 0	P0	R/W	x	x	—	Undefined
0FF02H	Port 2	P2	R	x	x	—	Undefined
0FF03H	Port 3	P3	R/W	x	x	—	Undefined
0FF04H	Port 4	P4	R/W	x	x	—	Undefined
0FF05H	Port 5	P5	R/W	x	x	—	Undefined
0FF06H	Port 6	P6	R/W	x	x	—	x0H
0FF07H	Port 7	P7	R	x	x	—	Undefined
0FF0AH	Port 0 buffer register (low)	P0L	R/W	x	x	—	Undefined
0FF0BH	Port 0 buffer register (high)	P0H	R/W	x	x	—	Undefined
0FF0CH	Real-time output port control register	RTPC	R/W	x	x	—	00H
0FF10H-0FF11H	16-bit compare register 0 (16-bit timer 0)	CR00	R/W	—	—	x	Undefined
0FF12H-0FF13H	16-bit compare register (16-bit timer 0)	CR01	R/W	—	—	x	Undefined
0FF14H	8-bit compare register (8-bit timer 1)	CR10	R/W	—	x	—	Undefined
0FF15H	8-bit compare register (8-bit timer/counter 2)	CR20	R/W	—	x	—	Undefined
0FF16H	8-bit compare register (8-bit timer/counter 2)	CR21	R/W	—	x	—	Undefined
0FF17H	8-bit compare register (8-bit timer 3)	CR30	R/W	—	x	—	Undefined
0FF18H-0FF19H	16-bit capture register (16-bit timer 0)	CR02	R	—	—	x	Undefined
0FF1AH	8-bit capture register (8-bit timer/counter 2)	CR22	R	—	x	—	Undefined
0FF1CH	8-bit capture/compare register (8-bit timer 1)	CR11	R/W	—	x	—	Undefined
0FF20H	Port 0 mode register	PM0	W	—	x	—	FFH
0FF23H	Port 3 mode register	PM3	W	—	x	—	FFH
0FF25H	Port 5 mode register	PM5	W	—	x	—	FFH
0FF26H	Port 6 mode register	PM6	R/W	x	x	—	FxH
0FF30H	Capture/compare control register 0	CRC0	W	—	x	—	10H
0FF31H	Timer output control register	TOC	W	—	x	—	00H
0FF32H	Capture/compare control register 1	CRC1	W	—	x	—	00H
0FF34H	Capture/compare control register 2	CRC2	W	—	x	—	00H
0FF40H	Pullup resistor option register	PUO	R/W	x	x	—	00H
0FF43H	Port 3 mode control register	PMC3	R/W	x	x	—	00H
0FF50H-0FF51H	16-bit timer register 0	TM0	R	—	—	x	0000H
0FF52H	8-bit timer register 1	TM1	R	—	x	—	00H
0FF54H	8-bit timer register 2	TM2	R	—	x	—	00H
0FF56H	8-bit timer register 3	TM3	R	—	x	—	00H
0FF5CH	Prescaler mode register 0	PRM0	W	—	x	—	00H
0FF5DH	Timer control register 0	TMC0	R/W	—	x	—	00H
0FF5EH	Prescaler mode register 1	PRM1	W	—	x	—	00H
0FF5FH	Timer control register 1	TMC1	R/W	—	x	—	00H
0FF68H	A/D converter mode register	ADM	R/W	x	x	—	00H
0FF6AH	A/D conversion result register	ADCR	R	—	x	—	Undefined
0FF78H	EEPROM write control register	EWC	R/W	x	x	—	00110100B

Table 2. Special Function Registers (cont)

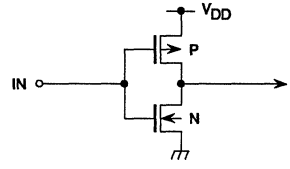
Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
0FF7DH	One-shot pulse output control register	OSPC	R/W	x	x	—	00H
0FF80H	Clocked serial interface mode register	CSIM	R/W	x	x	—	00H
0FF82H	Serial bus interface control register	SBIC	R/W	x	x	—	00H
0FF86H	Serial shift register	SIO	R/W	—	x	—	Undefined
0FF88H	Asynchronous serial interface mode register	ASIM	R/W	x	x	—	80H
0FF8AH	Asynchronous serial interface status register	ASIS	R	x	x	—	00H
0FF8CH	Serial receive buffer: UART	RxB	R	—	x	—	Undefined
0FF8EH	Serial transmit shift register: UART	TxS	W	—	x	—	Undefined
0FF90H	Baud rate generator control register	BRGC	W	—	x	—	00H
0FFC0H	Standby control register	STBC	R/W	—	x	—	00H
0FFC4H	Memory expansion mode register	MM	R/W	x	x	—	20H
0FFC5H	Programmable wait control register	PW	R/W	x	x	—	80H
0FFC6H	Refresh mode register	RFM	R/W	x	x	—	00H
0FFD0H-0FFDFH	External SFR area	—	R/W	x	x	—	Undefined
0FFE0H	Interrupt request flag register 0L	IF0L	R/W	x	x	—	00H
0FFE1H	Interrupt request flag register 0H	IF0H	R/W	x	x	—	00H
0FFE0H-0FFE1H	Interrupt request flag register 0	IF0	R/W	—	—	x	0000H
0FFE2H	Interrupt request flag register 1L	IF1L	R/W	x	x	—	xxxxxx00B
0FFE4H	Interrupt mask flag register 0L	MK0L	R/W	x	x	—	FFH
0FFE5H	Interrupt mask flag register 0H	MK0H	R/W	x	x	—	FFH
0FFE4H-0FFE5H	Interrupt mask flag register 0	MK0	R/W	—	—	x	FFFFH
0FFE6H	Interrupt mask flag register 1L	MK1L	R/W	x	x	—	xxxxxx11B
0FFE8H	Priority specification flag register 0L	PR0L	R/W	x	x	—	FFH
0FFE9H	Priority specification flag register 0H	PR0H	R/W	x	x	—	FFH
0FFE8H-0FFE9H	Priority specification flag register 0	PR0	R/W	—	—	x	FFFFH
0FFEAH	Priority specification flag register 1L	PR1L	R/W	x	x	—	xxxxxx11B
0FFECH	Interrupt service mode specification flag register 0L	ISM0L	R/W	x	x	—	00H
0FFEDH	Interrupt service mode specification flag register 0H	ISM0H	R/W	x	x	—	00H
0FFECH-0FFEDH	Interrupt service mode specification flag register 0	ISM0	R/W	—	—	x	0000H
0FFEEH	Interrupt service mode specification flag register 1L	ISM1L	R/W	x	x	—	xxxxxx00B
0FFF4H	External interrupt mode register 0	INTM0	R/W	x	x	—	00H
0FFF5H	External interrupt mode register 1	INTM1	R/W	x	x	—	00H
0FFF8H	Interrupt status register	IST	R/W	x	x	—	00H

Input/Output Ports

There are up to 54 port lines on the μPD78244 and up to 36 port lines on the μPD78243. (Ports 4, 5, and two bits of port 6 are not available on the μPD78243 since the μPD78243 must always use external memory.) Table 3 lists the features of each port and figure 3 shows the structure of each port pin. The pin levels of all port 2, 3, and 7 pins can always be read or tested regardless of the dual pin function.

Figure 3. Pin I/O Circuits

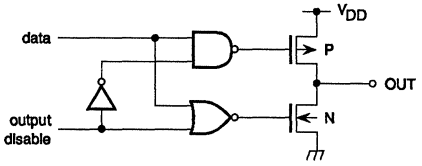
Type 1 ($\overline{EA}$)



Type 2 ($P2_0$, $P1_1$, $\overline{RESET}$)

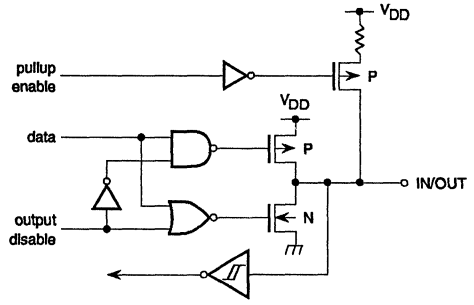
Schmitt trigger input with hysteresis characteristic.

Type 4 (P0, $P6_0$ - $P6_3$, ASTB)

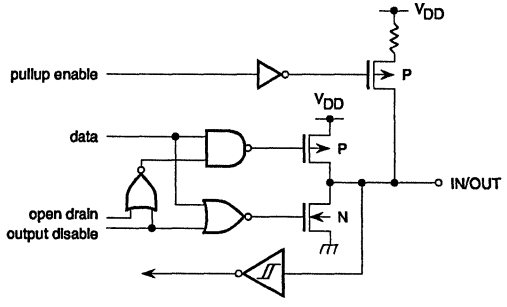


Push-pull output where the output can be placed in high-impedance (both P and N channels are turned off).

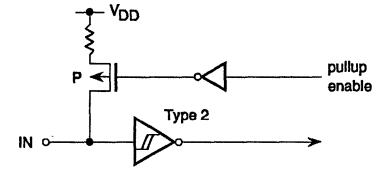
Type 8-A ($P3_2$)



Type 10-A ($P3_3$)

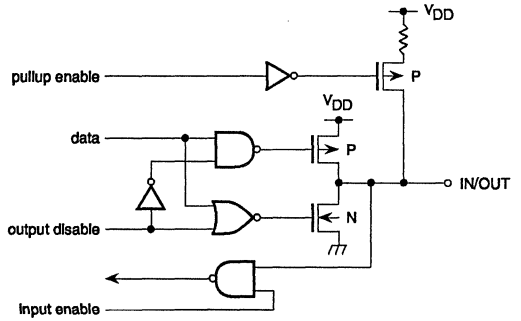


Type 2-A ($P2_2$ - $P2_7$)

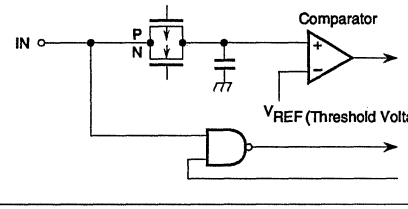


Schmitt trigger input with hysteresis characteristic.

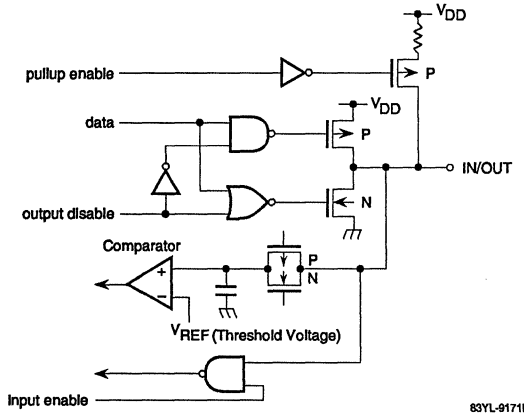
Type 5-A ($P3_0$, $P3_1$, $P3_4$ - $P3_7$, P4, P5, $P6_4$, $P6_5$)



Type 9 (P7)



Type 11 ($P6_6$ - $P6_7$)



83YL-9171B

Table 3. Digital Port Functions

Port	Operational Features	Configuration	Direct Drive Capability	Software Pullup Resistor Connection
Port 0	8-bit high impedance output		Transistor	
Port 2	8-bit Schmitt trigger input			In 6-bit units ($P2_2$-$P2_7$)
Port 3	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 4	8-bit input or output	Byte selectable	LED	Byte selectable
Port 5	8-bit input or output	Byte selectable	LED	Byte selectable, input bits only
Port 6	4-bit output (bits 0 to 3) 4-bit input or output (bits 4 to 7)	Bit selectable		In 4-bit unit, input bits only
Port 7	6-bit input			

Note: Software pullup resistors can be internally connected only on a port-by-port basis to port bits set to input mode. Pullup resistors are not connected to port bits set to output mode.

Real-time Output Port

The real-time output port (RTPC) shares pins with port 0. It can be used as two independent 4-bit real-time output ports or one 8-bit real-time output port. In the real-time output mode, data stored beforehand in the buffer registers, POH and POL, is transferred immediately to the output latch of P0 on the occurrence of a timer 1 interrupt (INTC10 or INTC11) or external interrupt (INTP0) (see figure 4). By using the real-time output port with the macro service function, port 0 can be used to output preprogrammed patterns at preprogrammed variable time intervals. In this mode, two independent stepper motors can accurately be driven at a fixed or variable rate.

4e

Figure 4. Real-time Output Port

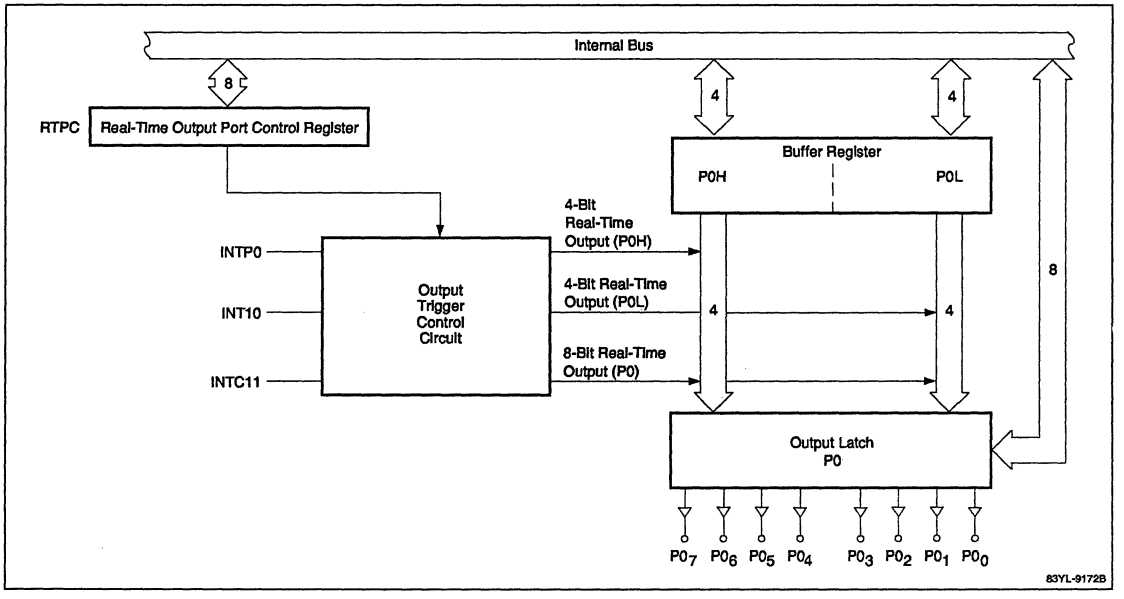
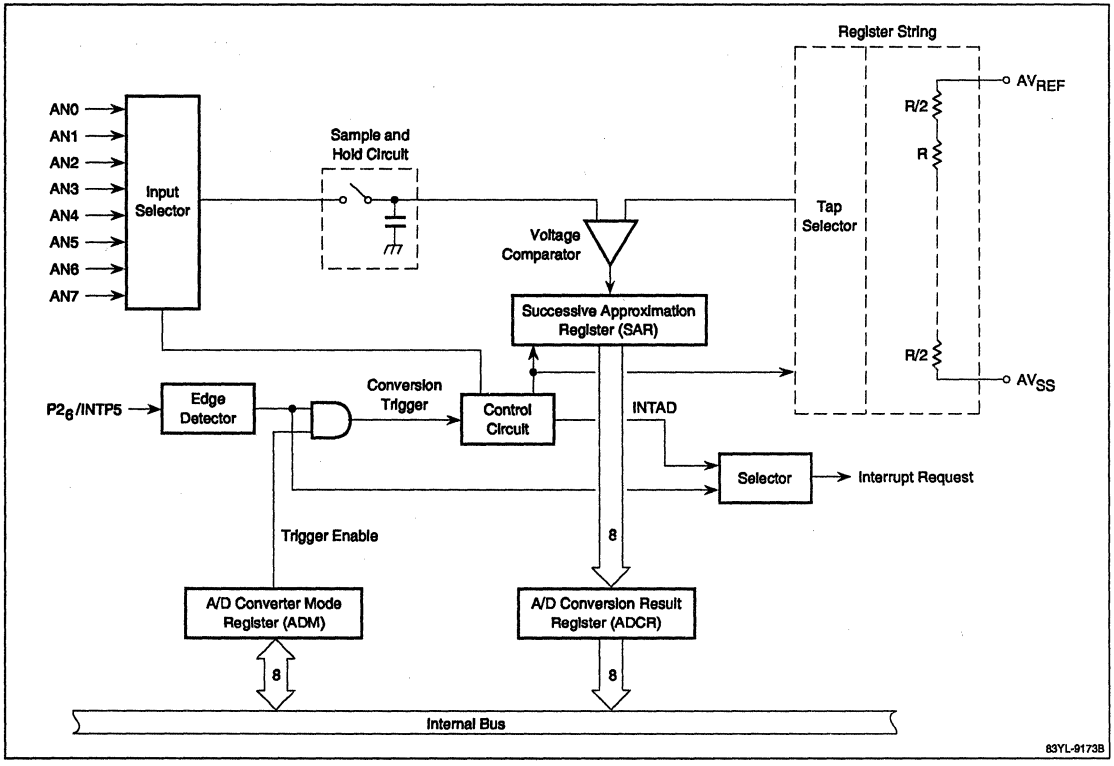


Figure 5. A/D Converter



Analog-to-Digital (A/D) Converter

The μPD78244 family A/D converter (see figure 5) uses the successive-approximation method for converting up to eight multiplexed analog inputs into 8-bit digital data. The conversion time per input is 30 μs at 12 MHz operation. A/D conversion can be started by an external interrupt, INTP5, or under software control.

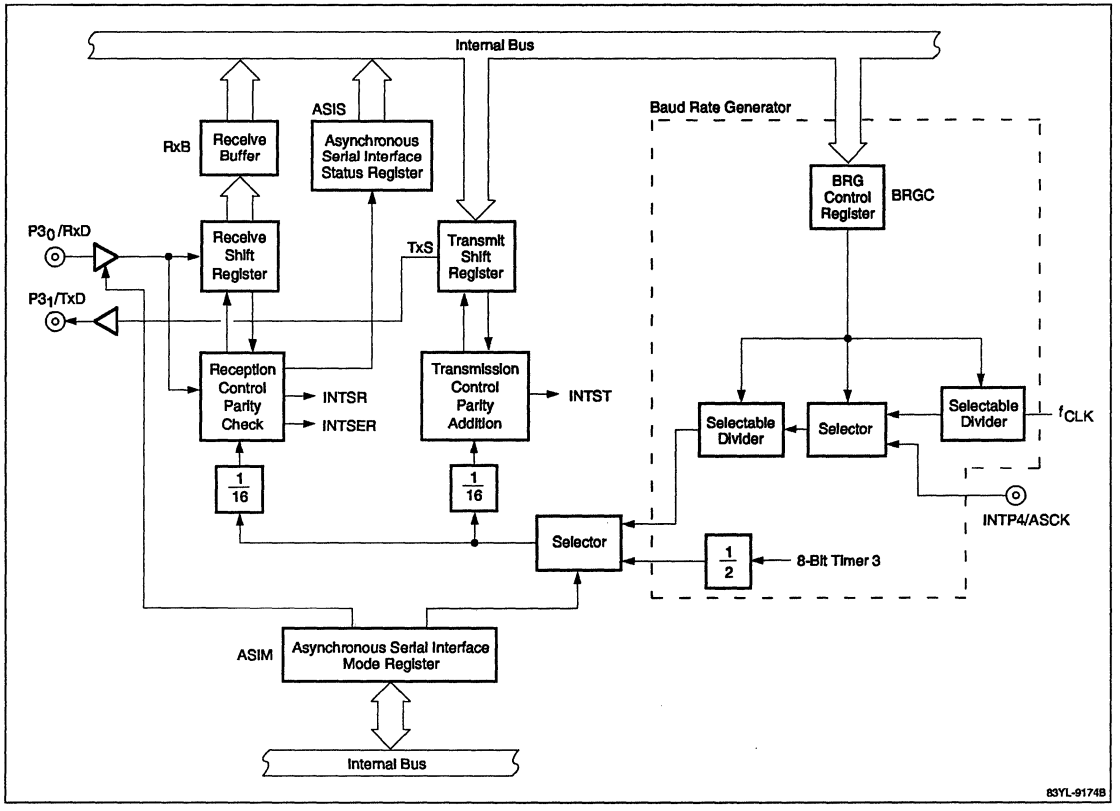
The A/D converter can operate in either scan mode or select mode. In scan mode, from one to eight sequential inputs can be programmed for conversion. The A/D converter selects each input in order, converts the data, stores it in the A/D conversion result (ADCR) register, and generates an interrupt (INTAD). This converted data can be easily transferred to memory by using the macro service function.

In select mode, only one of the eight A/D inputs can be selected for conversion. The ADCR register is continually updated and can be read at any time. If the A/D converter is started by an external interrupt, an INTAD interrupt occurs at the completion of each conversion. If the A/D converter is started by software, no interrupts are generated.

Serial Interface

The μPD78244 family has two independent serial interfaces. The first is a standard UART. The UART (figure 6) permits full-duplex operation and can be programmed for 7- or 8-bits of data after the start bit, followed by one or two stop bits. Odd, even, zero or no parity can also be selected. The serial clock for the UART can be provided by an on-chip baud rate generator or timer 3. By using either the internal system clock or an external clock input into the ASCK pin, the baud rate generator is capable of generating all of the commonly used baud rates. The UART generates three interrupts: INTST (transmission complete), INTSR (reception complete), and INTSER (reception error).

Figure 6. Asynchronous Serial Interface



The second interface is an 8-bit clock-synchronized serial interface (figure 7). It can be operated in either a three-wire serial I/O mode or NEC serial bus interface (SBI) mode.

In the three-wire serial I/O mode, the 8-bit shift register (SIO) is loaded with a byte of data and eight clock pulses are generated. These eight pulses shift the byte of data out of the SO line (MSB first) and in from the SI line providing full-duplex operation. This interface can also be set to receive or to transmit data only. The INTCSI interrupt is generated after each 8-bit transfer. One of three internal clocks or an external clock clocks the data.

The NEC SBI mode is a two-wire high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are connected in a master/slave configuration (see figure 8). There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over the serial bus line (SB0) using a fixed hardware protocol synchronized with the $\overline{SCK}$ line. Each slave μPD78244 family device can be programmed in software to respond to any one of 256 addresses. There are also 256 commands and 256 data types. Since all commands are user definable, any software protocol, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

Timers

The μPD78244 family has one 16-bit timer and three 8-bit timers. The 16-bit timer counts the internal system clock ($f_{CLK}/8$) while the three 8-bit timers can be programmed to count a number of prescaled values of the internal system clock. One of the 8-bit timers can also count external events.

Figure 7. Clock-Synchronized Serial Interface

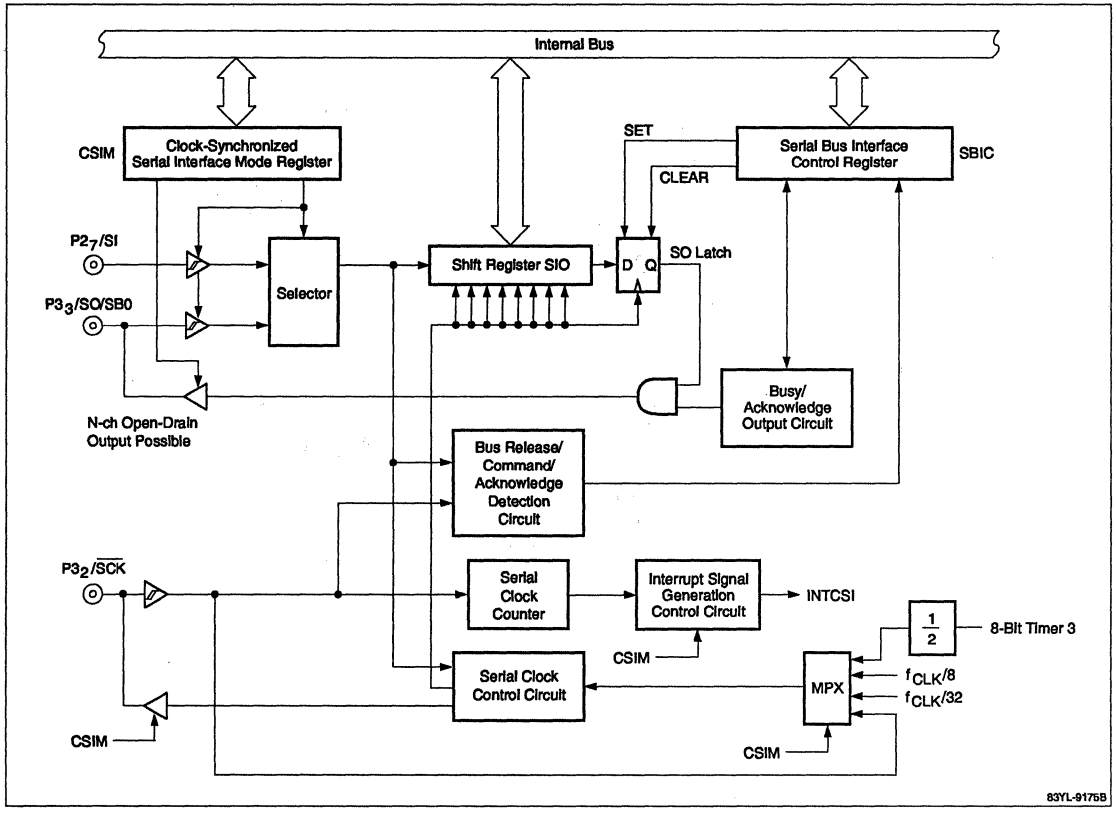
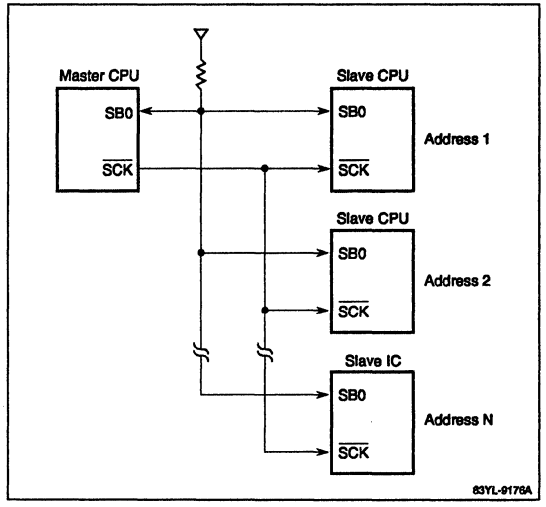


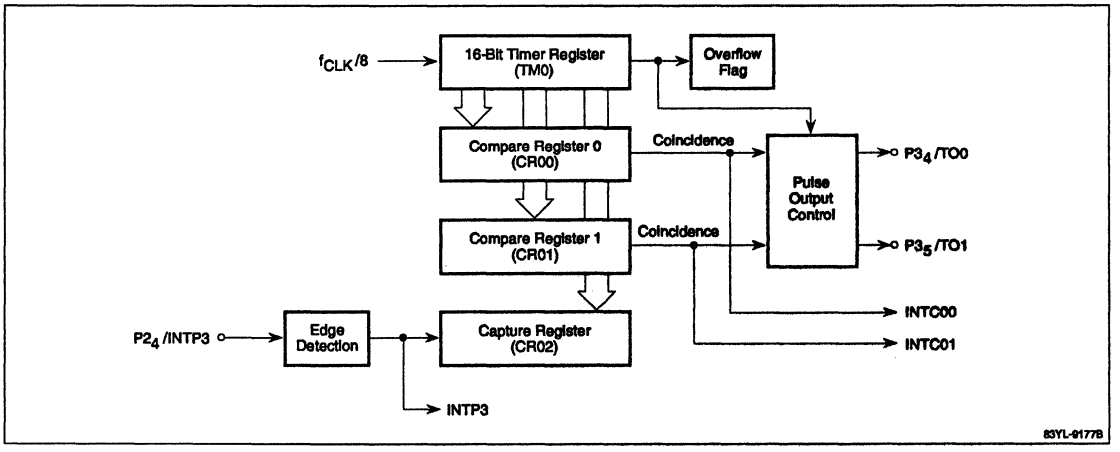
Figure 8. SBI Mode Master/Slave Configuration



Timer 0 consists of a 16-bit timer (TM0), two 16-bit compare registers (CR00 and CR01), and a 16-bit capture register (CR02). Timer 0 can be used as two interval timers, to output a programmable square wave or two pulse-width modulated signals, to measure pulse widths, or to generate a software-triggered one-shot output pulse (see figure 9).

4e

Figure 9. 16-Bit Timer 0



Timer 1 consists of an 8-bit timer (TM1), 8-bit compare register (CR10), and 8-bit capture/compare register (CR11). Timer 1 can be used as two interval timers or to measure pulse widths. In addition, it can be used to generate the output trigger for the real-time output port (see figure 10).

Timer/counter 2 consists of an 8-bit timer (TM2), two 8-bit compare registers (CR20 and CR21), and an 8-bit capture register (CR22). Timer/counter 2 can also be used as two interval timers, to output a programmable square wave or two pulse-width modulated signals, or to measure pulse widths. In addition, it can be used to count external events sensed on the CI line or as a one-shot timer (see figure 11).

Timer 3 consists of an 8-bit timer (TM3) and an 8-bit compare register (CR30). Timer 3 can be used as an interval timer or as a clock for the clocked-synchronized serial interface (see figure 12).

Figure 10. 8-Bit Timer 1

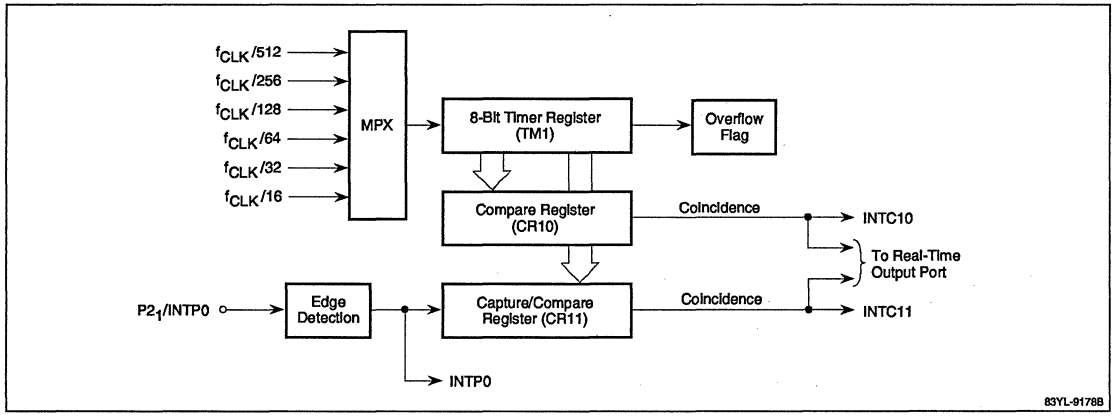


Figure 11. 8-Bit Timer/Counter 2

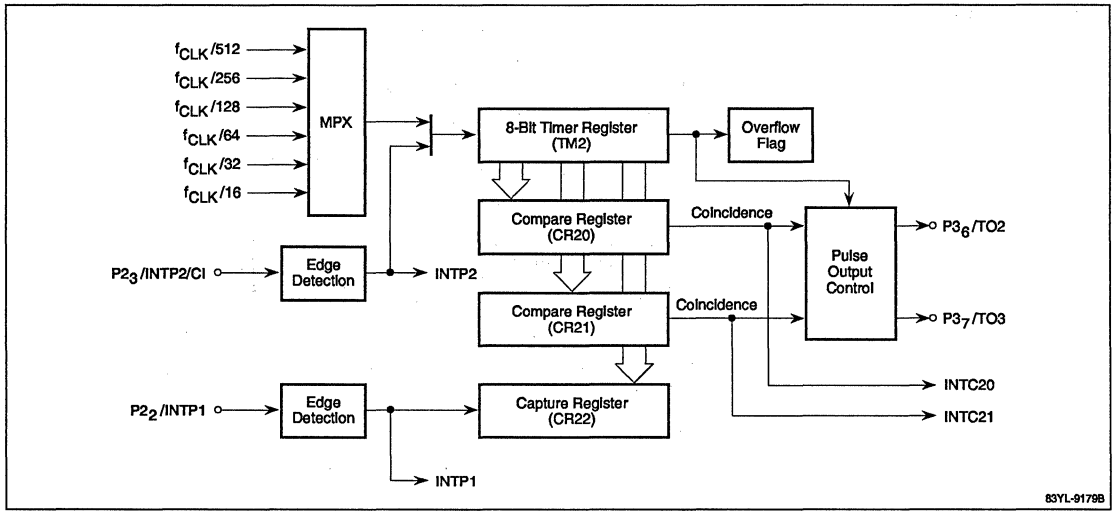
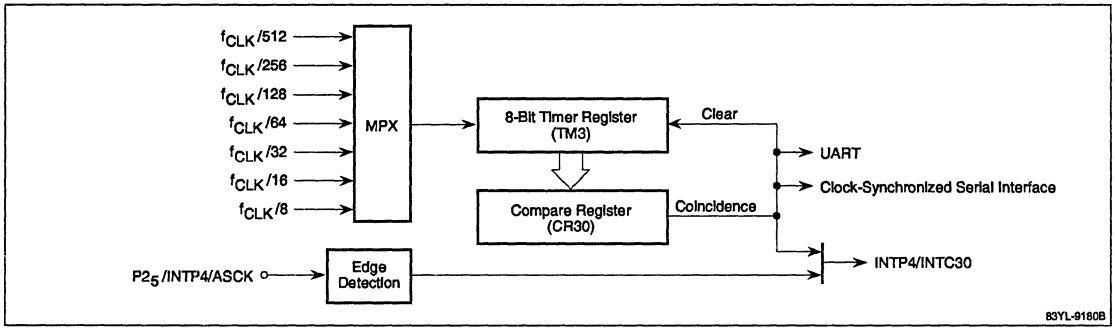


Figure 12. 8-Bit Timer 3



Interrupts

The μPD78244 family has 20 maskable hardware interrupt sources; 6 are external and 14 are internal. Since there are only 18 interrupt vectors and sets of control flags, 2 of the 6 external maskable interrupts, INTP4 and INTP5, share interrupt vectors and control flags with INTC30 and INTAD respectively. The active interrupt source for each shared vector must be chosen by the program. In addition, there is one nonmaskable interrupt and one software interrupt. The software interrupt, generated by the BRK instruction, is not maskable (see table 4).

Interrupt Servicing. The μPD78244 family provides two levels of programmable hardware priority control and two different methods of handling maskable interrupt requests: standard vectoring and macro service. The programmer can choose the priority and mode of servicing each maskable interrupt by using the interrupt control registers.

Interrupt Control Registers. The μPD78244 family has four 16-bit and four 8-bit interrupt control registers. In order to maintain software compatibility with the μPD78214 and μPD78218A families, the four 16-bit registers are identical to the μPD78214 and μPD78218A families and the two additional EEPROM interrupts are handled by the four additional 8-bit registers.

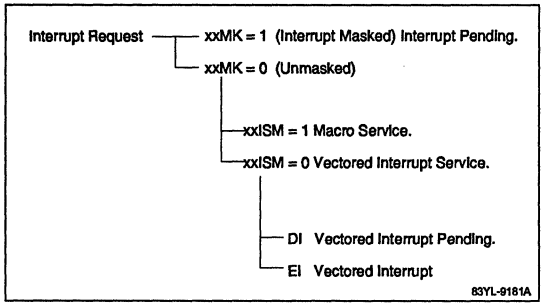
Each bit in each 16-bit register is dedicated to one of the 16 active maskable interrupt sources (excluding the two EEPROM interrupts). The interrupt request flag register (IF0) contains an interrupt request flag for each interrupt. The interrupt mask register (MK0) is used to enable or disable any interrupt. The interrupt service mode register (ISM0) specifies whether an interrupt is processed by vectoring or macro service. The priority flag register (PR0) can be used to specify a high or a low priority level for each interrupt. The 8-bit registers IF1L, MK1L, ISM1L, and PR1L provide similar functions for the two EEPROM interrupt sources.

Two other 8-bit registers are associated with interrupt processing. The interrupt status register (IST) indicates if a nonmaskable interrupt request on the NMI pin is being processed and can be used to allow nesting of nonmaskable interrupt requests. The IE and the ISP bits of the program status word are also used to control interrupts. If the IE bit is zero, all maskable interrupts, but not macro service, are disabled. The IE bit can be set or cleared using the EI and DI instructions, respectively, or by directly writing to the PSW. The IE bit is cleared each time an interrupt is accepted. The ISP bit is used by hardware to hold the priority level flag of the interrupt being serviced.

Interrupt Priority. The nonmaskable interrupt (NMI) has priority over all other interrupts. Two hardware controlled priority levels are available for the maskable interrupts. Either a high or a low priority level can be assigned by software to each of the maskable interrupts. Interrupt requests of a priority higher than the processor's current priority level are accepted; requests of the same or lower priority are held pending until the processor's priority state is lowered by program control within the current service routine or by a return instruction from the current service routine.

Interrupt requests programmed to be handled by macro service have priority over all vectored interrupt service regardless of the assigned priority level, and macro service requests are accepted even when the interrupt enable bit in the PSW is set to the disable state (see figure 13).

Figure 13. Interrupt Service Sequence



The default priorities listed in table 4 are fixed by hardware and are effective only when it is necessary to choose between two interrupt requests of the same software-assigned priority. For example, the default priorities would be used after the completion of a high priority routine if two interrupts of the same priority routine were pending.

The software interrupt, initiated by the BRK instruction, is executed regardless of the processor's priority level and the state of the IE bit. It does not alter the processor's priority level.

Table 4. Interrupt Sources and Vector Addresses

Interrupt Request Type	Default Priority	Interrupt Request Generation Source	Macro Service Type	Vector Table Address
Software	None	BRK instruction execution	—	003EH
Nonmaskable	None	NMI (pin input edge detection)	—	0002H
Maskable	0	INTP0 (pin input edge detection)	A, B	0006H
	1	INTP1 (pin input edge detection)	A, B	0008H
	2	INTP2 (pin input edge detection)	A, B	000AH
	3	INTP3 (pin input edge detection)	B	000CH
	4	INTC00 (TM0-CR00 coincidence signal generation)	B	0014H
	5	INTC01 (TM0-CR01 coincidence signal generation)	B	0016H
	6	INTC10 (TM1-CR10 coincidence signal generation)	A, B, C	0018H
	7	INTC11 (TM1-CR11 coincidence signal generation)	A, B, C	001AH
	8	INTC21 (TM2-CR21 coincidence signal generation)	A, B	001CH
	9	INTP4 (pin input edge detection)	B	000EH
		INTC30 (TM3-CR30 coincidence signal generation)	A, B	
	10	INTP5 (pin input edge detection)	B	0010H
		INTAD (end of A/D conversion)	A, B	
	11	INTC20 (TM2-CR20 coincidence signal generation)	A, B	0012H
Maskable	12	INTSER (generation of asynchronous serial interface receive error)	—	0020H
	13	INTSR (end of asynchronous serial interface reception)	A, B	0022H
	14	INTST (end of asynchronous serial interface transmission)	A, B	0024H
	15	INTCSI (end of clocked serial interface transmission)	A, B	0026H
	16	INTEER (EEPROM write error)	—	0028H
	17	INTEPW (EEPROM write completion)	—	002AH

Vectored Interrupt. When vectored interrupt is specified for a given interrupt request, (1) the program status word and the program counter are saved on the stack, (2) the processor's priority is set to that specified for the interrupt, (3) the IE bit in the PSW is set to zero, and (4) the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (RETB instruction for the software interrupt) reverses the process and the μPD78244 family device resumes the interrupted routine.

Macro Service

When macro service is specified for a given interrupt, the macro service hardware temporarily stops the executing program and begins to transfer data between the special function register area and the memory space. One byte is transferred each interrupt. When the data transfer is complete, control is returned to the executing program, providing a completely transparent method of interrupt service. Macro service significantly improves response time and makes it unnecessary to save any registers.

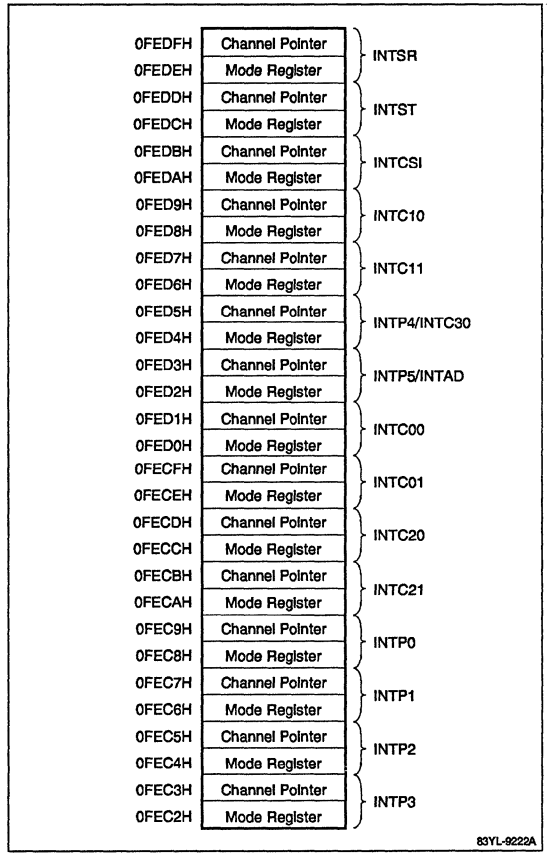
For each request on the interrupt line, one operation is performed, and an 8- or 16-bit counter is decremented. When the counter reaches zero, a vectored interrupt service routine is entered according to the specified priority.

Macro service is provided for all of the maskable interrupt requests except the following: INTSER, the asynchronous serial interface receive error interrupt request; INTEER, the EEPROM write error interrupt request; and INTEPW, the EEPROM write termination interrupt request. Each interrupt request has a dedicated macro service control word stored in Internal RAM (see figure 14). The function to be performed is specified in the control word.

The μPD78244 family provides three different types of macro service transfers:

Macro Service Type A. A byte of data is transferred in either direction between a special function register, preassigned for each interrupt request, and a buffer in internal RAM (FExx). Only the 8-bit macro service counter is available for Type A transfers. The preassigned SFRs for the 12 interrupt requests that support macro service Type A transfers are listed in table 5.

Figure 14. Macro Service Control Word Map



4e

Table 5. Macro Service Type A Interrupts and Assigned SFRs

Interrupt Request	Source/Destination SFR
INTC10: TM1-CR10 coincidence	CR10: Timer 1 8-bit compare register
INTC11: TM1-CR11 coincidence	CR11: Timer 1 8-bit capture/ compare register
INTC20: TM2-CR20 coincidence	CR20: Timer 2 8-bit compare register
INTC21: TM2-CR21 coincidence	CR21: Timer 2 8-bit compare register
INTC30: TM3-CR30 coincidence	CR30: Timer 3 8-bit compare register
INTSR: End of asynchronous serial interface reception	RxB: Serial receive buffer
INTST: End of asynchronous serial interface transmission	TxS: Serial transmit shift register
INTCSI: End of clocked serial interface transmission	SIO: Serial shift register
INTAD: End of A/D conversion	ADCR: A/D conversion result register
INTP0: External interrupt pin $P0_1$	CR11: Timer 1 8-bit capture/ compare register
INTP1: External interrupt pin $P0_2$	CR22: Timer 2 8-bit capture register
INTP2: External interrupt pin $P0_3$	TM2: Timer 2 8-bit timer register

Macro Service Type B. A byte of data is transferred in either direction between any specified special function register and a buffer anywhere in the 64K byte address space. The macro service counter can be programmed either to be an 8- or 16-bit counter. Macro service Type B transfers can be initiated by any maskable interrupt except INTSER, INTEER, and INTEPW.

Macro Service Type C. A byte of data is transferred from a buffer anywhere in the 64K byte address space to one of the 8-bit compare registers of timer 1. At the same time, a second byte of data is transferred from a buffer anywhere in the 64K byte address space to the real-time output port buffer. The macro service counter can be programmed either to be an 8- or 16-bit counter. Macro service Type C transfers can be initiated by INTC10 with data transferred to CR10 and P0L or P0H, or by INTC11 with data transferred to CR11 and P0L or P0H.

In addition, the macro service Type C transfer can be initialized to automatically alter timer compare register values or to repeatedly output a prespecified pattern at a fixed or variable rate. By using macro service Type C transfers to control the real-time output ports, the μPD78244 family can easily and accurately drive two independent stepper motors.

Refresh

The refresh signal is used with any pseudostatic RAM equivalent of the NEC μPD428128. The refresh cycle can be set to one of four intervals: 16, 32, 64, or $128/f_{CLK}$ (2.6, 5.3, 10.7, and 21.3 μs at 12 MHz). The refresh cycle is timed to follow a read or write operation to avoid interference with external memory access cycles.

Standby Modes

HALT and STOP modes are provided to reduce power consumption when CPU action is not required. In HALT mode, the CPU is stopped but the system clock continues to run. The HALT mode is released by any unmasked interrupt, an external NMI, or an external reset pulse. In STOP mode, both the CPU and the system clock are stopped, further minimizing the power consumption. The STOP mode is released by either an external reset pulse or an external NMI. The HALT and STOP modes are entered by programming the standby control register (STBC). This register is a protected location and can be written to only by a special instruction. If the third and fourth bytes of the instruction are not complements of each other, the data is not written and the next instruction is executed.

External Reset

The μPD78244 family is reset by taking the $\overline{\text{RESET}}$ pin low. The $\overline{\text{RESET}}$ input pin contains a noise filter to protect against spurious system resets caused by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low until the power supply reaches its operating voltage and the oscillator has stabilized. During reset, the program counter is loaded with the address contained in the reset vector table (address 0000H, 0001H) and program execution starts at that address upon the $\overline{\text{RESET}}$ pin going high. While $\overline{\text{RESET}}$ is low, all external lines except V_{SS}, V_{DD}, AV_{SS}, AV_{REF}, X1, and X2 are in the high impedance state.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = +25°C

Operating voltage, V_{DD}	−0.5 to +7.0 V
AV_{REF}	−0.5 to V_{DD} + 0.5 V
AV_{SS}	−0.5 to +0.5 V
Input voltage, V_{I1}	−0.5 to V_{DD} + 0.5 V
V_{I2} (Note 1)	−0.5 to AV_{REF} + 0.5 V
Output voltage, V_O	−0.5 to V_{DD} + 0.5 V
Low-level output current, I_{OL}	
per pin	15 mA
total, all output pins	100 mA
High-level output current, I_{OH}	
per pin	−10 mA
total, all output pins	−50 mA
Operating temperature, T_{OPT}	−10 to +70°C
Storage temperature, T_{STG}	−65 to +150°C

Notes:

(1) Pins $P7_0$/AN0 - $P7_5$/AN5, $P6_6/\overline{WAIT}$/AN6, and $P6_7/\overline{REFRQ}$/AN7 when used as the A/D converter input. However, V_{I1} absolute maximum ratings should also be satisfied.

Exposure to absolute maximum ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC characteristics.

Operating Conditions

Oscillation Frequency	T_A	V_{DD}
f_{XX} = 4 to 12 MHz	−10 to +70°C	+5 V ±10%

Capacitance

T_A = +25°C; V_{DD} = V_{SS} = 0 V

Item	Symbol	Max	Unit	Conditions
Input capacitance	C_1	20	pF	f = 1 MHz; pins not used for measurement are at 0 V
Output capacitance	C_O	20	pF	
Input/output capacitance	C_{IO}	20	pF	

External Clock Operation

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V

Item	Symbol	Min	Max	Unit	Conditions
X1 input low-level width	t_{WXL}	30	130	ns	
X1 input high-level width	t_{WXH}	30	130	ns	
X1 input rise time	t_{XR}	0	30	ns	
X1 input fall time	t_{XF}	0	30	ns	
X1 input clock cycle time	t_{CYX}	82	250	ns	

4e

DC Characteristics

T_A = −10 to +70°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V.

Item	Symbol	Min	Typ	Max	Unit	Conditions
Low-level input voltage	V_{IL}	0		0.8	V	
High-level input voltage	V_{IH1}	2.2		V_{DD}	V	Except the specified pins (Notes 1, 2)
	V_{IH2}	2.2		AV_{REF}	V	Specified pins (Note 1)
	V_{IH3}	0.8 V_{DD}		V_{DD}	V	Specified pins (Note 2)
Low-level output voltage	V_{OL1}			0.45	V	I_{OL} = 2.0 mA
	V_{OL2}			1.0	V	I_{OL} = 8.0 mA (Note 3)
High-level output voltage	V_{OH1}	V_{DD} − 1.0			V	I_{OH} = −1.0 mA
	V_{OH2}	V_{DD} − 0.5			V	I_{OH} = −100 μA
	V_{OH3}	2.0			V	I_{OH} = −5.0 mA (Note 4)
X1 low-level input current	I_{IL}			−100	μA	0 V ≤ V_I ≤ V_{IL}
X1 high-level input current	I_{IH}			100	μA	V_{IH3} ≤ V_I ≤ V_{DD}
Input leakage current	I_{LI}			±10	μA	0 V ≤ V_I ≤ V_{DD}
Output leakage current	I_{LO}			±10	μA	0 V ≤ V_O ≤ V_{DD}
AV_{REF} current	AI_{REF}		1.5	5.0	mA	Operating mode, f_{XX} = 12 MHz
V_{DD} power supply current	I_{DD1}		20	40	mA	Operating mode, f_{XX} = 12 MHz
	I_{DD2}		7	20	mA	HALT mode, f_{XX} = 12 MHz
Data retention voltage	V_{DDDR}	2.5		5.5	V	STOP mode

DC Characteristics (cont)

Item	Symbol	Min	Typ	Max	Unit	Conditions
Data retention current	I_{DDDR}		2	20	μA	STOP mode; V_{DDDR} = 2.5 V
			5	50	μA	STOP mode; V_{DDDR} = 5 V ±10%
Pullup resistor	R_L	15	40	80	kΩ	V_I = 0 V
EEPROM write voltage		4.5		5.5	V	f_{XX} = 4 to 12 MHz

Notes:

(1) Pins $P7_0$/AN0 - $P7_5$/AN5, $P6_6$/$\overline{WAIT}$/AN6, and $P6_7$/$\overline{REFRQ}$/AN7 when the pin is used as the A/D converter input.

(2) X1, X2, $\overline{RESET}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1,$P2_3$/$\overline{INTP2}$/CI, $P2_4$/INTP3, $P2_5$/$\overline{INTP4}$/ASCK, $P2_6$/INTP5, $P2_7$/SI, $P3_2$/SCK, $P3_3$/SO/SB0, and $\overline{EA}$ pins.

(3) Pins $P4_0$/AD_0 - $P4_7$/AD_7, and $P5_0$/A_8 - $P5_7$/A_{15}.

(4) Pins $P0_0$ - $P0_7$.

AC Characteristics—Read/Write Operation

T_A = −10 to +70°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF

Item	Symbol	Calculation Formula (Note 2, 3)	Min	Max	Unit	Conditions
X1 input clock cycle time	t_{CYX}	—	82	250	ns	
Address setup time to ASTB ↓	t_{SAST}	t_{CYX} − 30	52		ns	
Address hold time from ASTB ↓ (Note 1)	t_{HSTA}	—	25		ns	
Address hold time from $\overline{RD}$↑	t_{HRA}	—	30		ns	
Address hold time from $\overline{WR}$ ↑	t_{HWA}	—	30		ns	
Address to $\overline{RD}$ ↓ delay time	t_{DAR}	$2t_{CYX}$ − 35	129		ns	
Address float time to $\overline{RD}$ ↓	t_{FAR}	$t_{CYX}/2$ − 30	11		ns	
Address to data input time	t_{DAID}	(4+2n) t_{CYX} − 100		228	ns	No wait states
ASTB ↓ to data input time	t_{DSTID}	(3+2n)t_{CYX} − 65		181	ns	No wait states
$\overline{RD}$ ↓ to data input time	t_{DRID}	(2+2n)t_{CYX} − 64		100	ns	No wait states
ASTB ↓ to $\overline{RD}$ ↓ delay time	t_{DSTR}	t_{CYX} − 30	52		ns	
Data hold time from $\overline{RD}$ ↑	t_{HRID}	—	0		ns	
$\overline{RD}$ ↑ to address active time	t_{DRA}	$2t_{CYX}$ − 40	124		ns	
$\overline{RD}$ ↑ to ASTB ↑ delay time	t_{DRST}	$2t_{CYX}$ − 40	124		ns	
$\overline{RD}$ low-level width	t_{WRL}	(2+2n)t_{CYX} − 40	124		ns	No wait states
ASTB high-level width	t_{WSTH}	t_{CYX} − 30	52		ns	
Address to $\overline{WR}$ ↓ delay time	t_{DAW}	$2t_{CYX}$ − 35	129		ns	
$\overline{ASTB}$ ↓ to data output time	t_{DSTOD}	t_{CYX} + 60		142	ns	
$\overline{WR}$ ↓ to data output time	t_{DWOD}	—		60	ns	
ASTB ↓ to $\overline{WR}$ ↓ delay time	t_{DSTW1}	t_{CYX} − 30	52		ns	
	t_{DSTW2}	$2t_{CYX}$ − 35	129		ns	Refresh mode
Data setup time to $\overline{WR}$ ↑	t_{SODWR}	(3+2n)t_{CYX} − 100	146		ns	No wait states
Data setup time to $\overline{WR}$ ↓	t_{SODWF}	t_{CYX} − 60	22		ns	Refresh mode
Data hold time from $\overline{WR}$ ↑ (Note 1)	t_{HWOD}	—	20		ns	
$\overline{WR}$ ↑ to ASTB ↑ delay time	t_{DWST}	t_{CYX} − 40	42		ns	
$\overline{WR}$ low-level width	t_{WWL1}	(3+2n)t_{CYX} − 50	196		ns	No wait states
	t_{WWL2}	(2+2n)t_{CYX} − 50	114		ns	Refresh mode; No wait states
Address to $\overline{WAIT}$ ↓ input time	t_{DAWT}	$3t_{CYX}$ − 100		146	ns	

AC Characteristics—Read/Write Operation (cont)

Item	Symbol	Calculation Formula (Note 2, 3)	Min	Max	Unit	Conditions
ASTB ↓ to $\overline{WAIT}$ ↓ input time	t_{DSTWT}	$2t_{CYX} - 80$		84	ns	
$\overline{WAIT}$ hold time from ASTB ↓	t_{HSTWT}	$2Xt_{CYX} + 10$	174		ns	One external wait state
ASTB ↓ to $\overline{WAIT}$ ↑ delay time	t_{DSTWTH}	$2(1+X)t_{CYX} - 55$		273	ns	One external wait state
$\overline{RD}$ ↓ to $\overline{WAIT}$ input time	t_{DRWTL}	$t_{CYX} - 60$		22	ns	
$\overline{WAIT}$ hold time from $\overline{RD}$ ↓	t_{HRWT}	$(2X-1)t_{CYX} + 5$	87		ns	One external wait state
$\overline{RD}$ ↓ to $\overline{WAIT}$ ↑ delay time	t_{DRWTH}	$(2X+1)t_{CYX} - 60$		186	ns	One external wait state
$\overline{WAIT}$ ↑ to data input time	t_{DWTID}	$t_{CYX} - 20$		62	ns	
$\overline{WAIT}$ ↑ to $\overline{WR}$ ↑ delay time	t_{DWTW}	$2t_{CYX} - 10$	154		ns	
$\overline{WAIT}$ ↑ to $\overline{RD}$ ↑ delay time	t_{DWTR}	$t_{CYX} - 10$	72		ns	
$\overline{WR}$ ↓ to $\overline{WAIT}$ input time	t_{DWWTL}	$t_{CYX} - 60$		22	ns	Refresh disabled
$\overline{WAIT}$ hold time from $\overline{WR}$ ↓	t_{HWWT1}	$(2X-1)t_{CYX} + 5$	87		ns	One external wait state; refresh disabled
	t_{HWWT2}	$2(X-1)t_{CYX} + 5$	5		ns	One external wait state; refresh enabled
$\overline{WR}$ ↓ to $\overline{WAIT}$ ↑ delay time	t_{DWWTH1}	$(2X+1)t_{CYX} - 60$		186	ns	One external wait state; refresh disabled
	t_{DWWTH2}	$2Xt_{CYX} - 60$		104	ns	One external wait state; refresh enabled
$\overline{RD}$ ↑ to $\overline{REFRQ}$ ↓ delay time	t_{DRRFQ}	$2t_{CYX} - 10$	154		ns	
$\overline{WR}$ ↑ to $\overline{REFRQ}$ ↓ delay time	t_{DWRFQ}	$t_{CYX} - 10$	72		ns	
$\overline{REFRQ}$ low-level width	t_{WRFQL}	$2t_{CYX} - 44$	120		ns	
$\overline{REFRQ}$ ↑ to ASTB ↑ delay time	t_{DRFQST}	$4t_{CYX} - 48$	280		ns	

Notes:

(1) The hold time includes the time during which V_{OH} and V_{OL} are retained under the following load conditions: C_L = 100 pF and R_L = 2 kΩ.

(2) n indicates the number of internal wait states.

(3) X indicates the number of external wait states (1, 2, 3, ...)

Serial Port Operation

T_A = −10 to +70°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF.

Item	Symbol	Min	Max	Unit	Conditions
Serial clock cycle time	t_{CYSK}	1.0		μs	External clock input
		1.3		μs	Internal clock/16 output
		5.3		μs	Internal clock/64 output
Serial clock low-level width	t_{WSKL}	420		ns	External clock input
		556		ns	Internal clock/16 output
		2.5		μs	Internal clock/64 output
Serial clock high-level width	t_{WSKH}	420		ns	External clock input
		556		ns	Internal clock/16 output
		2.5		μs	Internal clock/64 output
SI, SB0 setup time to $\overline{SCK}$ ↑	t_{SSSK}	150		ns	
SI, SB0 hold time from $\overline{SCK}$ ↑	t_{HSSK}	400		ns	

4e

Serial Port Operation (cont)

Item	Symbol	Min	Max	Unit	Conditions
SO/SB0 output delay time from $\overline{SCK}$ ↓	t_{DSBSK1}	0	300	ns	CMOS push-pull output (3-line serial I/O mode)
	t_{DSBSK2}	0	800	ns	Open-drain output (SBI mode), R_L = 1 kΩ
SB0 high, hold time from $\overline{SCK}$ ↑	t_{HSBSK}	4		t_{CYX}	SBI mode
SB0 low, setup time to $\overline{SCK}$ ↓	t_{SSBSK}	4		t_{CYX}	SBI mode
SB0 low-level width	t_{WSBL}	4		t_{CYX}	
SB0 high-level width	t_{WSBH}	4		t_{CYX}	

A/D Converter Operation

T_A = −10 to +70°C; V_{DD} = +5 V ±10%; V_{SS} = AV_{SS} = 0 V.

Item	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		8			Bit	
Full-scale error (Note 1)				0.4	%	AV_{REF} = 4.0 V to V_{DD}
				0.8	%	AV_{REF} = 3.6 V to V_{DD}
Quantization error				±1/2	LSB	
Conversion time	t_{CONV}	360			t_{CYX}	83 ns ≤ t_{CYX} ≤ 125 ns
		240			t_{CYX}	125 ns ≤ t_{CYX} ≤ 250 ns
Sampling time	t_{SAMP}	72			t_{CYX}	83 ns ≤ t_{CYX} ≤ 125 ns
		48			t_{CYX}	125 ns ≤ t_{CYX} ≤ 250 ns
Analog input voltage	V_{IAN}	−0.3		AV_{REF} + 0.3	V	
Analog input impedance	R_{AN}		1000		MΩ	
Analog reference voltage	AV_{REF}	3.6		V_{DD}	V	
AV_{REF} current	AI_{REF}		1.5	5.0	mA	Operating mode, f_{XX} = 12 MHz
			0.2	1.5	mA	(Note 2)

Notes:

(1) Quantization errror is not included. Unit is defined as percent of full-scale value.

(2) When CS bit of the ADM register is set to 0

Interrupt Timing Operation

Item	Symbol	Min	Max	Unit	Conditions
NMI low-level width	t_{WNIL}	10		μs	
NMI high-level width	t_{WNIH}	10		μs	
INTP0-INTP5 low-level width	t_{WITL}	24		t_{CYX}	
INTP0-INTP5 high-level width	t_{WITH}	24		t_{CYX}	
$\overline{RESET}$ low-level width	t_{WRSL}	10		μs	
$\overline{RESET}$ high-level width	t_{WRSH}	10		μs	

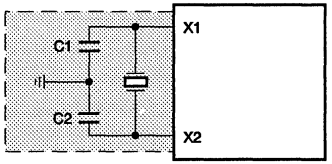
Data Retention Characteristics

T_A = −10 to +70°C.

Item	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.5		5.5	V	STOP mode
Data retention current	I_{DDDR}		2	20	μA	V_{DDDR} = 2.5 V
			5	50	μA	V_{DDDR} = 5 V ±10%
V_{DD} rise time	t_{RVD}	200			μs	
V_{DD} fall time	t_{FVD}	200			μs	
V_{DD} retention time (from STOP mode setting)	t_{HVD}	0			ms	
STOP release signal input time	t_{DREL}	0			ms	
Oscillation stabilization wait time	t_{WAIT}	30			ms	Crystal resonator
		5			ms	Ceramic resonator
Low-level input voltage	V_{IL}	0		0.1 V_{DDDR}	V	Specified pins (Note 1)
High-level input voltage	V_{IH}	0.9 V_{DDDR}		V_{DDDR}	V	Specified pins (Note 1)

Note: $\overline{RESET}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2/CI, $P2_4$/INTP3, $P2_5$/INTP4/ASCK, $P2_6$/INTP5, $P2_7$/SI, $P3_2/\overline{SCK}$, and $P3_3$/SO/SB0

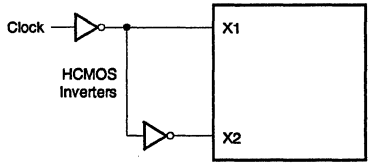
Recommended Resonator Circuit



Ceramic or crystal resonator frequency f_{XX} = 4 to 12 MHz
External oscillation circuit should be as close to the X1 and X2 pins as possible
Do not place other signal lines in the shaded area

83YL-9189A

Recommended External Clock Circuit



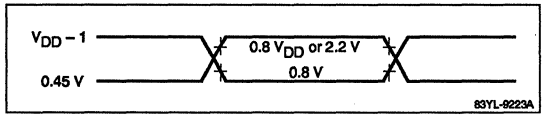
Clock frequency f_{XX} = 4 to 12 MHz
In STOP mode, X1 is internally shorted to V_{SS} to prevent leakage current. Therefore, STOP mode is not available when using this external clock circuit.

83YL-9190A

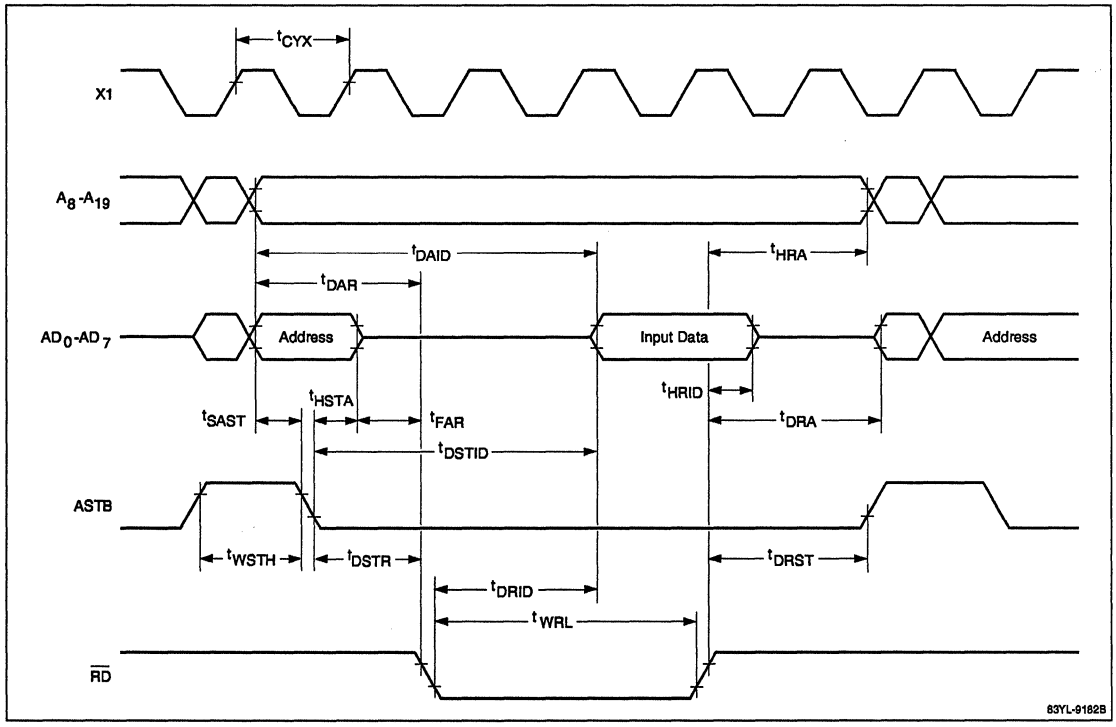
4e

Timing Waveforms

Voltage Thresholds for AC Timing Measurements

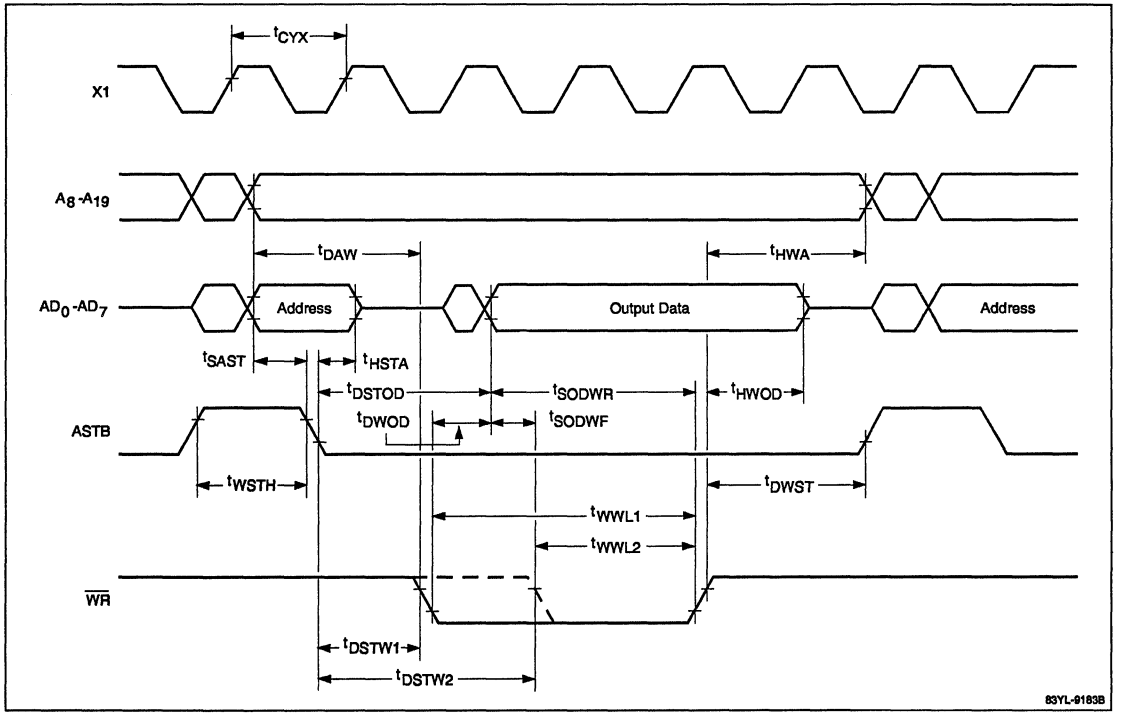


Read Operation



Timing Waveforms (cont)

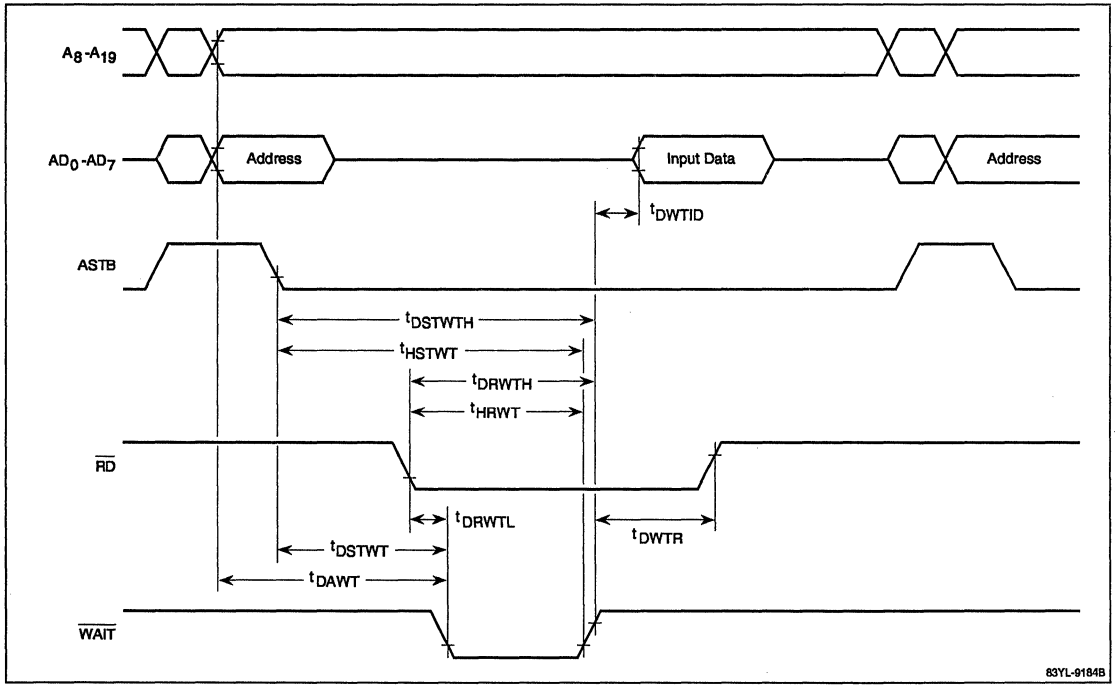
Write Operation



4e

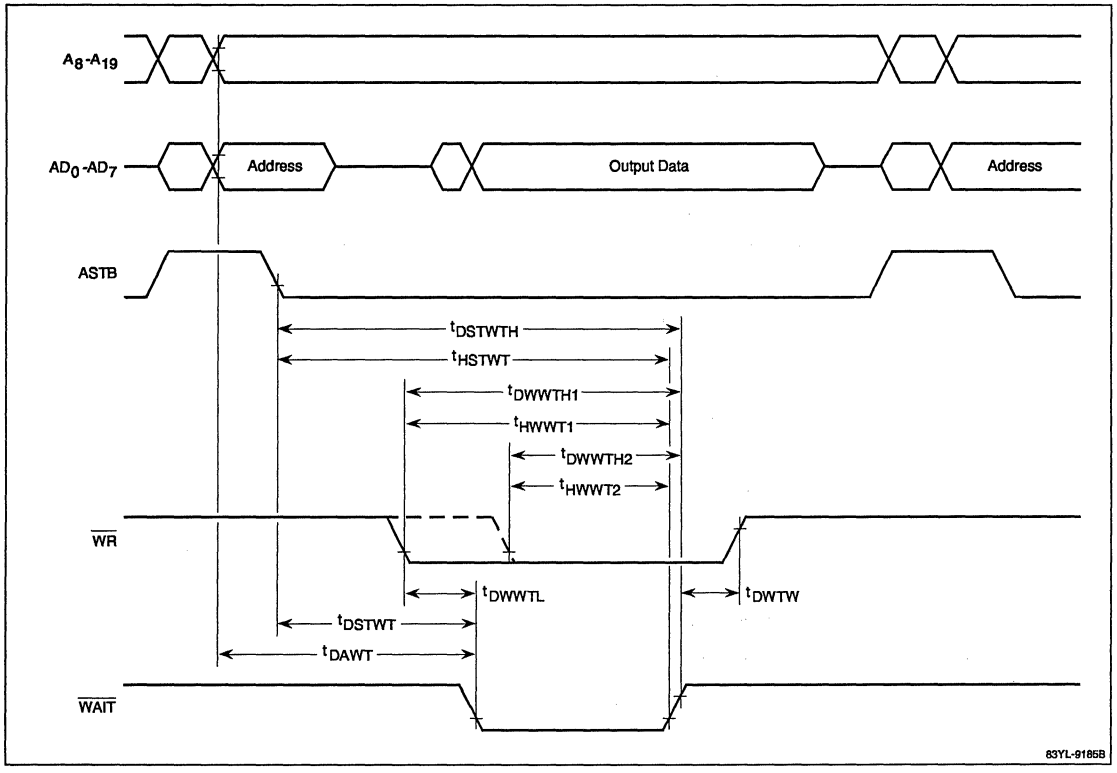
Timing Waveforms (cont)

External $\overline{WAIT}$ Signal Input (Read Operation)



Timing Waveforms (cont)

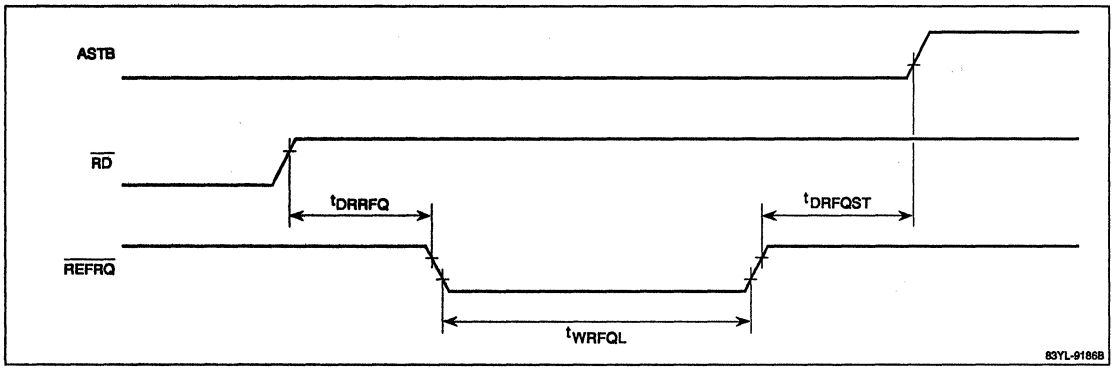
External $\overline{WAIT}$ Signal Input (Write Operation)



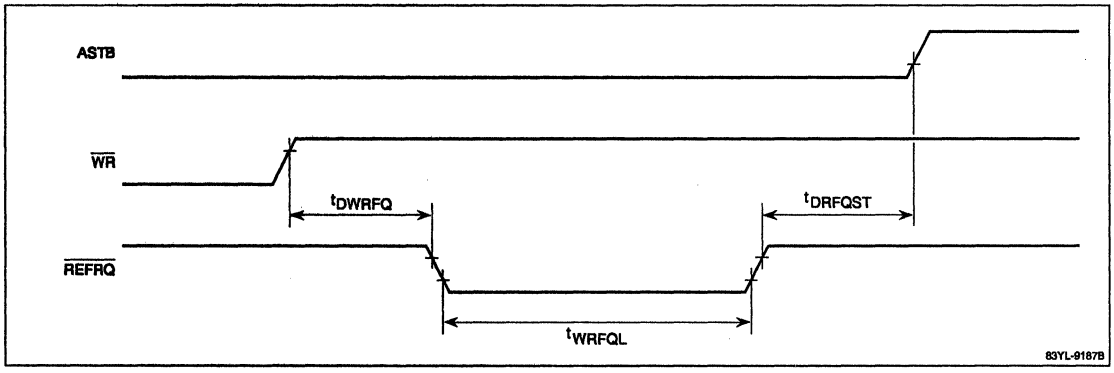
4e

Timing Waveforms (cont)

Refresh After Read



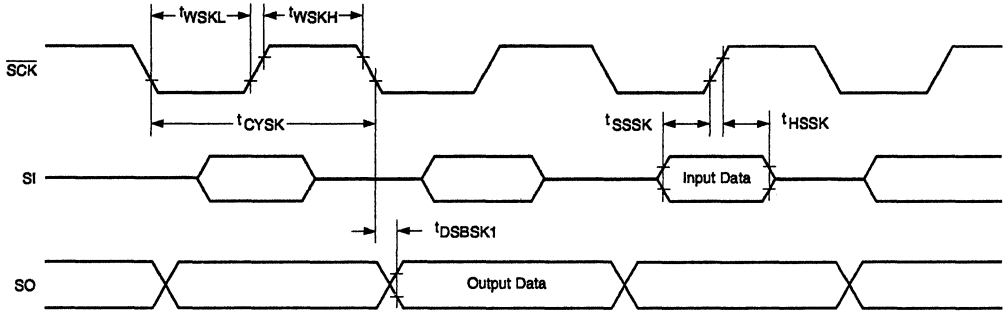
Refresh After Write



Timing Waveforms (cont)

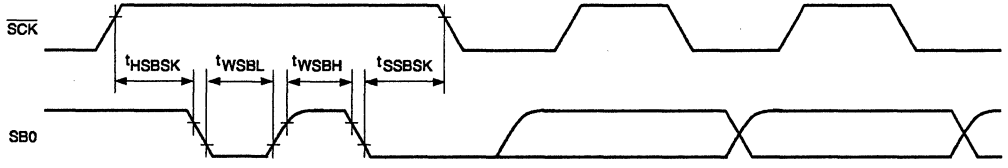
Serial Operation

Three-Line Serial I/O Timing

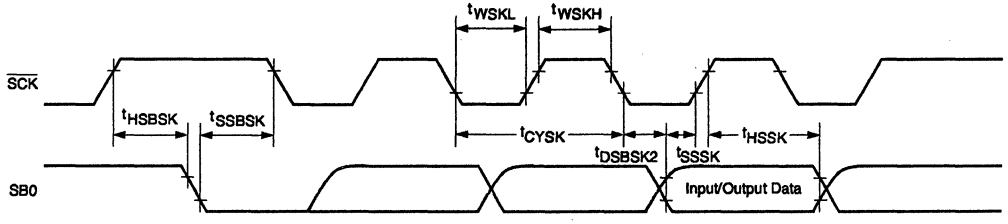


SBI Mode

Bus Release Signal Transfer Timing



Command Signal Transfer Timing

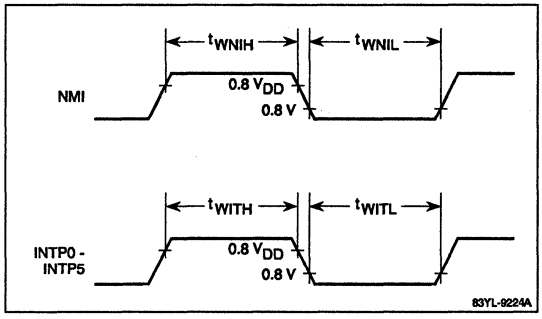


83YL-9188B

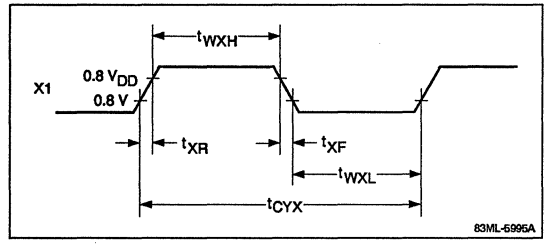
4e

Timing Waveforms (cont)

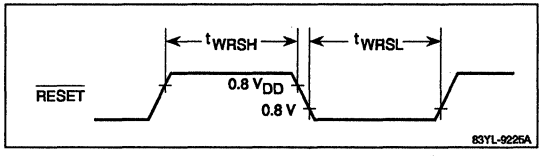
Interrupt Input



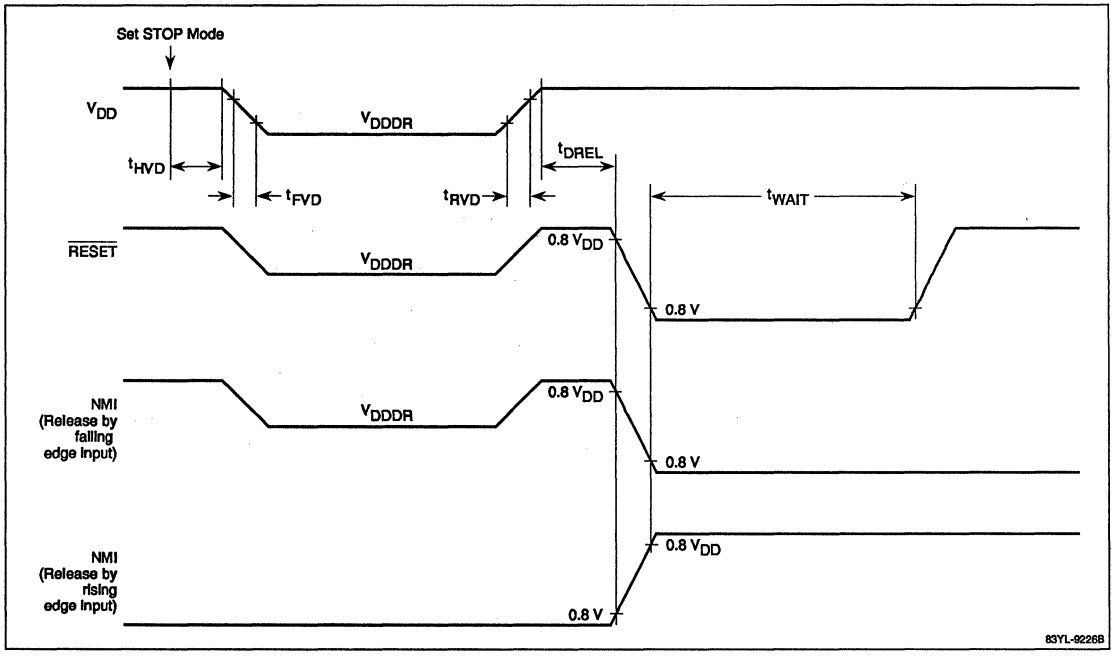
External Clock



Reset Input



Data Retention Characteristics



μPD78K2 Product Line

This programming reference contains the instruction set and the interrupt vector tables for the μPD78K2 product line.

The instruction set features both 8- and 16-bit data transfer and arithmetic instructions, 8-bit logic instructions, and single-bit manipulation instructions. Branch instructions exist to test individual bits in the program status word, the 16-bit accumulator, the special function registers and in the saddr portion of Internal RAM. Instructions range in length from 1 to 5 bytes, depending on the instruction and addressing mode.

Operands and Operations

Refer to the following tables for the definitions of symbols in the operand and operation columns of the Instruction Set table.

Uppercase letters, such as "A" or "PSW," are key symbols and must be written as shown in the Registers and Flags table. See the Registers and Flags table for the list of key symbols. Lowercase letters, such as "sfr" or "mem" are not key symbols and an absolute value or label must be substituted by the user when writing the instruction. For example, "MOV A, sfr" may be written as "MOV A, P0." When the symbols +, –, #, !, $, /, [ ], and & are used as a prefix of a word, the symbol remains while lower case letters are replaced by a value. For example, "ADD A, #byte" may be written as "ADD A, #0AFH," or "BR $addr16" may be written as "BR $LOOP1."

Symbols r and rp can be described using the functional name or absolute name.

Operands

Symbol	Definitions
+	Autoincrement
–	Autodecrement
#	Immediate data
!	Absolute address
$	Relative address
/	Bit inversion
[ ]	Indirect addressing
&	Subbank
r, r'	Register Functional name: X, A, C, B, E, D, L, H Absolute name: R0 to R7
r1	Register group 1: C, B
rp, rp'	Register pair Functional name: AX, BC, DE, HL Absolute name: RP0 to RP3
sfr	Special function register (8-bit): See individual data sheets for specific sfr names.
sfrp	Special function register pair (16-bit): See individual data sheets for specific sfrp names.
mem	Memory address indirectly addressed Register indirect mode: [DE], [HL], [DE+], [HL+], [DE–], [HL–] Base mode: [DE+ byte], [HL+ byte], [SP+ byte] Indexed mode: word[A], word[B], word [DE], word[HL]
mem1	Group 1 memory address indirectly addressed: [DE], [HL]
saddr, saddr'	Memory address addressed by means of short direct addressing: FE20H-FF1FH immediate data or label
saddrp	Memory address addressed by means of short direct addressing pair: FE20H-FF1EH immediate data(LSB = 0; even address) or label
addr16	16-bit address: 0000H-FEFFH immediate data or label
addr11	11-bit address: 800H-FFFH immediate data or label
addr5	5-bit address: 40H-7EH immediate data or label (even address only)
word	16-bit data: 16-bit immediate data or label
byte	8-bit data: 8-bit immediate data or label
bit	3-bit data: 3-bit immediate data or label
n	Number of shift bits: 3-bit immediate data (0-7)
RBn	Register bank: RB0-RB3

Registers and Flags

Symbol	Definitions
A	A register; 8-bit accumulator
X	X register
B	B register
C	C register
D	D register
E	E register
H	H register
L	L register
R0-R7	Registers 0 to 7 (absolute names)
AX	Register pair (AX); 16-bit accumulator
BC	Register pair (BC)
DE	Register pair (DE)
HL	Register pair (HL)
RP0-RP3	Register pairs 0 to 3 (absolute names)
PC	Program counter
SP	Stack pointer
Registers and Flags	
Symbol	Definitions
PSW	Program status word
CY	Carry flag
AC	Auxiliary carry flag
Z	Zero flag
RBS1-RBS0	Register bank select flags
IE	Interrupt enable flag
STBC	Standby control register
jdisp8	Signed 2's complement data (8 bits) indicating relative address distance between first address of next instruction and branch destination address
( )	Memory contents indicated by address or register contents in ( )
xxH	Hexadecimal number
x_H, x_L	Higher 8 bits and lower 8 bits of 16-bit register pair
∧	Logical product (AND)
∨	Logical sump (OR)
∀	Exclusive logical sum (exclusive OR)
—	Inverted data

Flag Column Indicators

Blank	No change
0	Cleared to 0
1	Set to 1
x	Set or cleared depending on the result
R	Value previously saved is restored

Bytes

The number of bytes for instructions with a mem or &mem operand depends on the particular instruction and the memory addressing mode (register indirect, base, and or indexed).

A MOV instruction with register indirect mode specified for mem is a special 1-byte instruction. When base mode or indexed mode is specified for mem, the 8-bit or 16-bit offset data corresponding to byte and word, respectively, is added from the third byte onward.

Bytes for Instructions With "mem" and "&mem" Operands

Instruction		Register Indirect Mode		Base Mode		Indexed Mode
		[DE+] [HL+] [DE–] [HL–]	[DE] [HL]	[DE+byte] [HL+byte]	[SP+byte]	word[A] word[B] word[DE] word[HL]
MOV	A, mem	1	1	3	3	4
	mem, A	1	1	3	3	4
	A, &mem	2	2	4	4	5
	&mem, A	2	2	4	4	5
XCH	A, mem	2	2	3	3	4
	A, &mem	3	3	4	4	5
ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP	A, mem	2	2	3	3	4
	A, &mem	3	3	4	4	5

Instruction Set

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
8-Bit Data Transfer						
MOV	r, #byte	r ← byte	2			
	saddr, #byte	(saddr) ← byte	3			
	sfr, #byte	sfr ← byte	3			
	r, r'	r ← r'	2			
	A, r	A ← r	1			
	A, saddr	A ← (saddr)	2			
	saddr, A	(saddr) ← A	2			
	saddr, saddr'	(saddr) ← (saddr')	3			
	A, sfr	A ← sfr	2			
	sfr, A	sfr ← A	2			
	A, mem*	A ← (mem)	1-4			
	A, &mem*	A ← (&mem)	2-5			
	mem, A*	(mem) ← A	1-4			
	&mem, A*	(&mem) ← A	2-5			
	A, !addr16	A ← (!addr16)	4			
	A, &!addr16	A ← (&!addr16)	5			
	!addr16, A	(!addr16) ← A	4			
	&!addr16, A	(&!addr16) ← A	5			
	PSW, #byte	PSW ← byte	3	x	x	x
	PSW, A	PSW ← A	2	x	x	x
	A, PSW	A ← PSW	2			
XCH	A, r	A ↔r	1			
	r, r'	r ↔ r'	2			
	A, mem	A ↔ (mem)	2-4			
	A, &mem	A ↔ (&mem)	3-5			
	A, saddr	A ↔ (saddr)	2			
	A, sfr	A ↔ sfr	3			
	saddr, saddr'	(saddr) ↔ (saddr')	3			

* If [DE], [HL], [DE+], [DE–], [HL+] or [HL–] is described as mem, these instructions are used as dedicated 1-byte codes. If the register name is described as &mem, the instructions are used as dedicated 2-bytes codes.

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
16-Bit Data Transfer						
MOVW	rp, #word	rp ← word	3			
	saddrp, #word	(saddrp) ← word	4			
	sfrp, #word	sfrp ← word	4			
	rp, rp'	rp ← rp'	2			
	AX, saddrp	AX ← (saddrp)	2			
	saddrp, AX	(saddrp) ← AX	2			
	AX, sfrp	AX ← sfrp	2			
	sfrp, AX	sfrp ← AX	2			
	AX, mem1	AX ← (mem1)	2			
	AX, &mem1	AX ← (&mem1)	3			
	mem1, AX	(mem1) ← AX	2			
	&mem1, AX	(&mem1) ← AX	3			
8-Bit Operations						
ADD	A, #byte	A,CY ← A + byte	2	x	x	x
	saddr, #byte	(saddr),CY ← (saddr) + byte	3	x	x	x
	sfr, #byte	sfr,CY ← sfr + byte	4	x	x	x
	r, r'	r,CY ← r + r'	2	x	x	x
	A, saddr	A,CY ← A + (saddr)	2	x	x	x
	A, sfr	A,CY ← A + sfr	3	x	x	x
	saddr, saddr'	(saddr),CY ← (saddr) + (saddr')	3	x	x	x
	A, mem	A,CY ← A + (mem)	2-4	x	x	x
	A, &mem	A,CY ← A + (&mem)	3-5	x	x	x
ADDC	A, #byte	A,CY ← A + byte + CY	2	x	x	x
	saddr, #byte	(saddr),CY ← (saddr) + byte + CY	3	x	x	x
	sfr, #byte	sfr,CY ← sfr + byte + CY	4	x	x	x
	r, r'	r,CY ← r + r' + CY	2	x	x	x
	A, saddr	A,CY ← A + (saddr) + CY	2	x	x	x
	A, sfr	A,CY ← A + sfr + CY	3	x	x	x
	saddr, saddr'	(saddr),CY ← (saddr) + (saddr') + CY	3	x	x	x
	A, mem	A,CY ← A + (mem) + CY	2-4	x	x	x
	A, &mem	A,CY ← A + (&mem) + CY	3-5	x	x	x

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
8-Bit Operations (cont)						
SUB	A, #byte	A,CY ← A – byte	2	x	x	x
	saddr, #byte	(saddr),CY ← (saddr) – byte	3	x	x	x
	sfr, #byte	sfr,CY ← sfr – byte	4	x	x	x
	r, r'	r,CY ← r – r'	2	x	x	x
	A, saddr	A,CY ← A – (saddr)	2	x	x	x
	A, sfr	A,CY ← A – sfr	3	x	x	x
	saddr, saddr'	(saddr),CY ← (saddr) – (saddr')	3	x	x	x
	A, mem	A,CY ← A – (mem)	2-4	x	x	x
	A, &mem	A,CY ← A – (&mem)	3-5	x	x	x
SUBC	A, #byte	A,CY ← A – byte – CY	2	x	x	x
	saddr, #byte	(saddr),CY ← (saddr) – byte – CY	3	x	x	x
	sfr, #byte	sfr,CY ← sfr – byte – CY	4	x	x	x
	r, r'	r,CY ← r – r' – CY	2	x	x	x
	A, saddr	A,CY ← A – (saddr) – CY	2	x	x	x
	A, sfr	A,CY ← A – sfr – CY	3	x	x	x
	saddr, saddr'	(saddr),CY ← (saddr) – (saddr') – CY	3	x	x	x
	A, mem	A,CY ← A – (mem) – CY	2-4	x	x	x
	A, &mem	A,CY ← A – (&mem) – CY	3-5	x	x	x
AND	A, #byte	A ← A ∧ byte	2	x		
	saddr, #byte	(saddr) ← (saddr) ∧ byte	3	x		
	sfr, #byte	sfr ← sfr ∧ byte	4	x		
	r, r'	r ← r ∧ r'	2	x		
	A, saddr	A ← A ∧ (saddr)	2	x		
	A, sfr	A ← A ∧ sfr	3	x		
	saddr, saddr'	(saddr) ← (saddr) ∧ (saddr')	3	x		
	A, mem	A ← A ∧ (mem)	2-4	x		
	A, &mem	A ← A ∧ (&mem)	3-5	x		
OR	A, #byte	A ← A ∨ byte	2	x		
	saddr, #byte	(saddr) ← (saddr) ∨ byte	3	x		
	sfr, #byte	sfr ← sfr ∨ byte	4	x		
	r, r'	r ← r ∨ r'	2	x		
	A, saddr	A ← A ∨ (saddr)	2	x		
	A, sfr	A ← A ∨ sfr	3	x		
	saddr, saddr'	(saddr) ← (saddr) ∨ (saddr')	3	x		
	A, mem	A ← A ∨ (mem)	2-4	x		
	A, &mem	A ← A ∨ (&mem)	3-5	x		

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
8-Bit Operations (cont)						
XOR	A, #byte	A ← A ∀ byte	2	x		
	saddr, #byte	(saddr) ← (saddr) ∀ byte	3	x		
	sfr, #byte	sfr ← sfr ∀ byte	4	x		
	r, r'	r ← r ∀ r'	2	x		
	A, saddr	A ← A ∀ (saddr)	2	x		
	A, sfr	A ← A ∀ sfr	3	x		
	saddr, saddr'	(saddr) ← (saddr) ∀ (saddr')	3	x		
	A, mem	A ← A ∀ (mem)	2-4	x		
	A, &mem	A ← A ∀ (&mem)	3-5	x		
CMP	A, #byte	A – byte	2	x	x	x
	saddr, #byte	(saddr) – byte	3	x	x	x
	sfr, #byte	sfr – byte	4	x	x	x
	r, r'	r – r'	2	x	x	x
	A, saddr	A – (saddr)	2	x	x	x
	A, sfr	A – sfr	3	x	x	x
	saddr, saddr'	(saddr) – (saddr')	3	x	x	x
	A, mem	A – (mem)	2-4	x	x	x
	A, &mem	A – (&mem)	3-5	x	x	x
16-Bit Operations						
ADDW	AX, #word	AX,CY ← AX + word	3	x	x	x
	AX, rp	AX,CY ← AX + rp	2	x	x	x
	AX, saddrp	AX,CY ← AX + (saddrp)	2	x	x	x
	AX, sfrp	AX,CY ← AX + sfrp	3	x	x	x
SUBW	AX, #word	AX,CY ← AX – word	3	x	x	x
	AX, rp	AX,CY ← AX – rp	2	x	x	x
	AX, saddrp	AX,CY ← AX – (saddrp)	2	x	x	x
	AX, sfrp	AX,CY ← AX – sfrp	3	x	x	x
CMPW	AX, #word	AX – word	3	x	x	x
	AX, rp	AX – rp	2	x	x	x
	AX, saddrp	AX – (saddrp)	2	x	x	x
	AX, sfrp	AX – sfrp	3	x	x	x
Multiplication/Division						
MULU	r	AX ← A x r	2			
DIVUW	r	AX(quotient),r(remainder) ← AX ÷ r	2			

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
Increment/Decrement						
INC	r	r ← r + 1	1	x	x	
	saddr	(saddr) ← (saddr) + 1	2	x	x	
DEC	r	r ← r − 1	1	x	x	
	saddr	(saddr) ← (saddr) − 1	2	x	x	
INCW	rp	rp ← rp + 1	1			
DECW	rp	rp ← rp − 1	1			
Shift/Rotate						
ROR	r, n	(CY,r_7 ← r_0, r_{m-1} ← r_m) x n times, n = 0-7	2			x
ROL	r, n	(CY,r_0 ← r_7, r_{m+1} ← r_m) x n times, n = 0-7	2			x
RORC	r, n	(CY ← r_0, r_7 ← CY, r_{m-1} ← r_m) x n times, n = 0-7	2			x
ROLC	r, n	(CY ←r_7, r_0 ← CY, r_{m+1} ← r_m) x n times, n = 0-7	2			x
SHR	r, n	(CY ←r_0, r_7 ← 0, r_{m-1} ← r_m) x n times, n = 0-7	2	x	0	x
SHL	r, n	(CY ← r_7, r_0 ← 0, r_{m+1} ← r_m) x n times, n = 0-7	2	x	0	x
SHRW	rp, n	(CY ← rp_0, rp_{15} ← 0, rp_{m-1} ← rp_m) x n times, n = 0-7	2	x	0	x
SHLW	rp, n	(CY ← rp_{15}, rp_0 ← 0, rp_{m+1} ← rp_m) x n times, n = 0-7	2	x	0	x
ROR4	mem1	A_{3-0} ← $(mem1)_{3-0}$, $(mem1)_{7-4}$ ← A_{3-0}, $(mem1)_{3-0}$ ← $(mem1)_{7-4}$	2			
	&mem1	A_{3-0} ← $(\&mem1)_{3-0}$, $(\&mem1)_{7-4}$ ← A_{3-0}, $(\&mem1)_{3-0}$ ← $(\&mem1)_{7-4}$	3			
ROL4	mem1	A_{3-0} ← $(mem1)_{7-4}$, $(mem1)_{3-0}$ ← A_{3-0}, $(mem1)_{7-4}$ ← $(mem1)_{3-0}$	2			
	&mem1	A_{3-0} ← $(\&mem1)_{7-4}$, $(\&mem1)_{3-0}$ ← A_{3-0}, $(\&mem1)_{7-4}$ ← $(\&mem1)_{3-0}$	3			
BCD Adjustment						
ADJBA		Decimal adjust accumulator after addition	1	x	x	x
ADJBS		Decimal adjust acccumulator after subtraction	1	x	x	x
Bit Manipulation						
MOV1	CY, saddr.bit	CY ← (saddr.bit)	3			x
	CY, sfr.bit	CY ← sfr.bit	3			x
	CY, A.bit	CY ← A.bit	2			x
	CY, X.bit	CY ← X.bit	2			x
	CY, PSW.bit	CY ← PSW.bit	2			x
	saddr.bit, CY	(saddr.bit) ← CY	3			
	sfr.bit, CY	sfr.bit ← CY	3			
	A.bit, CY	A.bit ← CY	2			
	X.bit, CY	X.bit ← CY	2			
	PSW.bit, CY	PSW.bit ← CY	2	x	x	

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
Bit Manipulation (cont)						
AND1	CY, saddr.bit	CY ← CY ∧ (saddr.bit)	3			x
	CY, /saddr.bit	CY ← CY ∧ $\overline{(saddr.bit)}$	3			x
	CY, sfr.bit	CY ← CY ∧ sfr.bit	3			x
	CY, /sfr.bit	CY ← CY ∧ $\overline{sfr.bit}$	3			x
	CY, A.bit	CY ← CY ∧ A.bit	2			x
	CY, /A.bit	CY ← CY ∧ $\overline{A.bit}$	2			x
	CY, X.bit	CY ← CY ∧ X.bit	2			x
	CY, /X.bit	CY ← CY ∧ $\overline{X.bit}$	2			x
	CY, PSW.bit	CY ← CY ∧ PSW.bit	2			x
	CY, /PSW.bit	CY ← CY ∧ $\overline{PSW.bit}$	2			x
OR1	CY, saddr.bit	CY ← CY ∨ (saddr.bit)	3			x
	CY, /saddr.bit	CY ← CY ∨ $\overline{(saddr.bit)}$	3			x
	CY, sfr.bit	CY ← CY ∨ sfr.bit	3			x
	CY, /sfr.bit	CY ← CY ∨ $\overline{sfr.bit}$	3			x
	CY, A.bit	CY ← CY ∨ A.bit	2			x
	CY, /A.bit	CY ← CY ∨ $\overline{A.bit}$	2			x
	CY, X.bit	CY ← CY ∨ X.bit	2			x
	CY, /X.bit	CY ← CY ∨ $\overline{X.bit}$	2			x
	CY, PSW.bit	CY ← CY ∨ PSW.bit	2			x
	CY, /PSW.bit	CY ← CY ∨ $\overline{PSW.bit}$	2			x
XOR1	CY, saddr.bit	CY ← CY ⊻ (saddr.bit)	3			x
	CY, sfr.bit	CY ← CY ⊻ sfr.bit	3			x
	CY, A.bit	CY ← CY ⊻ A.bit	2			x
	CY, X.bit	CY ← CY ⊻ X.bit	2			x
	CY, PSW.bit	CY ← CY ⊻ PSW.bit	2			x
SET1	saddr.bit	(saddr.bit) ← 1	2			
	sfr.bit	sfr.bit ← 1	3			
	A.bit	A.bit ← 1	2			
	X.bit	X.bit ← 1	2			
	PSW.bit	PSW.bit ← 1	2	x	x	x
	CY	CY ← 1	1			1
CLR1	saddr.bit	(saddr.bit) ← 0	2			
	sfr.bit	sfr.bit ← 0	3			
	A.bit	A.bit ← 0	2			
	X.bit	X.bit ← 0	2			
	PSW.bit	PSW.bit ← 0	2	x	x	x
	CY	CY ← 0	1			0

4f

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
Bit Manipulation (cont)						
NOT1	saddr.bit	(saddr.bit) ← $\overline{\text{(saddr.bit)}}$	3			
	sfr.bit	sfr.bit ← $\overline{\text{sfr.bit}}$	3			
	A.bit	A.bit ← $\overline{\text{A.bit}}$	2			
	X.bit	X.bit ← $\overline{\text{X.bit}}$	2			
	PSW.bit	PSW.bit ← $\overline{\text{PSW.bit}}$	2	x	x	x
	CY	CY ← $\overline{\text{CY}}$	1			x
Call/Return						
CALL	!addr16	(SP–1) ← $(PC + 3)_H$, (SP – 2) ← $(PC + 3)_L$, PC ← addr16, SP ← SP – 2	3			
	rp	(SP – 1) ← $(PC + 2)_H$, (SP – 2) ← $(PC + 2)_L$, PC_H ← rp_H, PC_L ← rp_L, SP ← SP – 2	2			
CALLF	!addr11	(SP – 1) ← $(PC + 2)_H$, (SP – 2) ← $(PC + 2)_L$, PC_{15-11} ← 00001, PC_{10-0} ← addr11, SP ← SP – 2	2			
CALLT	[addr5]	(SP – 1) ← $(PC + 1)_H$, (SP – 2) ← $(PC + 1)_L$, PC_H ← (2 x addr5 + 41H), PC_L ← (2 x addr5 + 40H), SP← SP – 2	1			
BRK		(SP – 1) ← PSW, (SP – 2) ← $(PC + 1)_H$, (SP – 3) ← $(PC + 1)_L$, PC_L ← (003EH), PC_H ← (003FH), SP ← SP – 3, IE ← 0	1			
RET		PC_L ← (SP), PC_H ← (SP + 1), SP ← SP + 2	1			
RET1		PC_L ← (SP), PC_H ← (SP + 1), PSW ← (SP +2), SP ← SP + 3, NMIS ← 0	1	R	R	R
RETB		PC_L ← (SP), PC_H ← (SP + 1), PSW ←(SP+2), SP ← SP + 3	1	R	R	R
Stack Manipulation						
PUSH	PSW	(SP – 1) ← PSW, SP ← SP – 1	1			
	sfr	(SP – 1) ← sfr, SP ← SP – 1	2			
	rp	(SP – 1) ← rp_H, (SP – 2) ← rp_L, SP ← SP – 2	1			
POP	PSW	PSW ← (SP), SP ← SP + 1	1	R	R	R
	sfr	sfr ← (SP), SP ← SP + 1	2			
	rp	rp_L ← (SP), rp_H ← (SP +1), SP ← SP + 2	1			
MOVW	SP, #word	SP ← word	4			
	SP, AX	SP ← AX	2			
	AX, SP	AX ← SP	2			
INCW	SP	SP ← SP + 1	2			
DECW	SP	SP ← SP – 1	2			

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags Z	AC	CY
Unconditional Branch						
BR	!addr16	PC ← addr16	3			
	rp	$PC_H \leftarrow rp_H$, $PC_L \leftarrow rp_L$	2			
	$addr16	PC ← PC + 2 + jdisp8	2			
Conditional Branch						
BC	$addr16	PC ← PC +2 + jdisp8 if CY = 1	2			
BL						
BNC	$addr16	PC ← PC +2 + jdisp8 if CY = 0	2			
BNL						
BZ	$addr16	PC ← PC +2 + jdisp8 if CZ = 1	2			
BE						
BNZ	$addr16	PC ← PC +2 + jdisp8 if CZ = 0	2			
BNE						
BT	saddr.bit, $addr16	PC ← PC +3 + jdisp8 if (saddr.bit) = 1	3			
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 1	4			
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 1	3			
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 1	3			
	PSW.bit, $addr16	PC ← PC + 3 + jdisp8 if PSW.bit = 1	3			
BF	saddr.bit, $addr16	PC ← PC +4 + jdisp8 if (saddr.bit) = 0	4			
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 0	4			
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 0	3			
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 0	3			
	PSW.bit, $addr16	PC ← PC + 3 + jdisp8 if PSW.bit = 0	3			
BTCLR	saddr.bit, $addr16	PC ← PC +4 + jdisp8 if (saddr.bit) = 1 then reset (saddr.bit)	4			
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit	4			
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 1 then reset A.bit	3			
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 1 then reset X.bit	3			
	PSW.bit, $addr16	PC ← PC + 3 + jdisp8 if PSW.bit = 1 then reset PSW.bit	3	x	x	x
DBNZ	r1, $addr16	r1 ← r1 − 1, then PC ← PC + 2 + jdisp8 if r1 ≠ 0	2			
	saddr, $addr16	(saddr) ← (saddr) − 1, then PC ← PC + 3 + jdisp8 if (saddr) ≠ 0	3			
CPU Control						
MOV	STBC, #byte	STBC ← byte	4			
SEL	RBn	RBS1-0 ← n, n = 0-3	2			
NOP		No Operation	1			
EI		IE ← 1 (Enable Interrupt)	1			
DI		IE ← 0 (Disable Interrupt)	1			

4f

Interrupt Vectors: (μPD78214, μPD78218, μPD78238, and μPD78244 Families)

Default Priority Level (Note 1)	Signal Name	Interrupt Source	Register Flag	Mask Flag	Service Mode	Priority Flag	Vector Table Address	Macro Service Type	Macro Service Mode Reg. Address
Software									
None	BRK instruction	Software	—	—	—	—	003EH	None	None
Nonmaskable									
None	NMI (pin input edge)	External signal	—	—	—	—	002H	None	None
Maskable									
0	INTP0 (pin input edge)	External signal	PIF0	PMK0	PISM0	PPR0	0006H	A, B	FEC8H
1	INTP1 (pin input edge)		PIF1	PMK1	PISM1	PPR1	0008H	A, B	FEC6H
2	INTP2 (pin input edge)		PIF2	PMK2	PISM2	PPR2	000AH	A, B	FEC4H
3	INTP3 (pin input edge)		PIF3	PMK3	PISM3	PPR3	000CH	B	FEC2H
4	INTC00 (TM0-CR00 match signal)	16-bit timer 0	CIF00	CMK00	CISM00	CPR00	0014H	B	FED0H
5	INTC01 (TM0-CR01 match signal)		CIF01	CMK01	CISM01	CPR01	0016H	B	FECEH
6	INTC10 (TM1-CR10 match signal)	8-bit timer 1	CIF10	CMK10	CISM10	CPR10	0018H	A, B, C	FED8H
7	INTC11 (TM1-CR11 match signal)		CIF11	CMK11	CISM11	CPR11	001AH	A, B, C	FED6H
8	INTC21 (TM2-CR21 match signal)	8-bit timer/counter 2	CIF21	CMK21	CISM21	CPR21	001CH	A, B	FECAH
9	INTP4 (pin input edge)	External signal	PIF4	PMK4	PISM4	PPR4	000EH	B	FED4H
	INTC30 (TM3-CR30 match signal)	8-bit timer 3						A, B	
10	INTP5 (pin input edge)	External signal	PIF5	PMK5	PISM5	PPR5	0010H	B	FED2H
	INTAD (A/D conversion complete)	A/D converter						A, B	
11	INTC20 (TM2-CR20 match signal)	8-bit timer/counter 2	CIF20	CMK20	CISM20	CPR20	0012H	A, B	FECDH
12	INTSER (asynchronous serial receive error)	Asynchronous serial interface	SERIF	SERMK	—	SERPR	0020H	None	None
13	INTSR (asynchronous serial receive complete)		SRIF	SRMK	SRISM	SRPR	0022H	A, B	FEDEH
14	INTST (asynchronous serial transmit complete)		STIF	STMK	STISM	STPR	0024H	A, B	FEDCH
15	INTCSI (clocked serial interface transfer complete)	Clocked serial interface	CSIIF	CSIMK	CSIISM	CSIPR	0026H	A, B	FEDAH

Interrupt Vectors: (μPD78214, μPD78218, μPD78238, and μPD78244 Families) (cont)

Default Priority Level (Note 1)	Signal Name	Interrupt Source	Register Flag	Mask Flag	Service Mode	Priority Flag	Vector Table Address	Macro Service Type	Macro Service Mode Reg. Address
16	INTEER (EEPROM error) (μPD78244 family only)	EEPROM	EERIF	EERMK	—	EERPR	0028H	None	None
17	INTEPW(EEPROM write) (μPD78244 family only)	EEPROM	EPWIF	EPWMK	—	EPWPR	002AH	None	None

Notes:

(1) The default priority is a fixed numeric value indicating which interrupt takes precedence when more than one interrupt with the same priority has simultaneously occurred.

Interrupt Vectors: (μPD78224 Family)

Default Priority Level (Note 1)	Signal Name	Interrupt Source	Register Flag	Mask Flag	Service Mode	Priority Flag	Vector Table Address	Macro Service Type	Macro Service Mode Reg. Address
Software									
None	BRK instruction	Software	—	—	—	—	003EH	None	None
Nonmaskable									
None	NMI (pin input edge)	External signal	—	—	—	—	002EH	None	None
Maskable									
0	INTP0 (pin input edge)	External signal	PIF0	PMK0	—	PPR0	0006H		None
1	INTP1 (pin input edge)		PIF1	PMK1	—	PPR1	0008H	None	None
2	INTP2 (pin input edge)		PIF2	PMK2	—	PPR2	000AH	None	None
3	INTP3 (pin input edge)		PIF3	PMK3	—	PPR3	000CH	None	None
4	INTC00 (TM0-CR00 match signal)	16-bit timer 0	CIF00	CMK00	—	CPR00	0014H	None	None
5	INTC01 (TM0-CR01 match signal)		CIF01	CMK01	—	CPR01	0016H	None	None
6	INTC10 (TM1-CR10 match signal)	8-bit timer 1	CIF10	CMK10	CISM10	CPR10	0018H	A, B, C	FED8H
7	INTC11 (TM1-CR11 match signal)		CIF11	CMK11	CISM11	CPR11	001AH	A, B, C	FED6H
8	INTC21 (TM2-CR21 match signal)	8-bit timer/counter 2	CIF21	CMK21	—	CPR21	001CH	None	None
9	INTP4 (pin input edge)	External signal	PIF4	PMK4	PISM4	PPR4	000EH	B	FED4H
10	INTP5 (pin input edge)		PIF5	PMK5	—	PPR5	0010H	None	None
11	INTP6 (pin input edge)		CIF20	CMK20	—	CPR20	0012H	None	None

4f

Interrupt Vectors: (μPD78224 Family) (cont)

Default Priority Level (Note 1)	Signal Name	Interrupt Source	Register Flag	Mask Flag	Service Mode	Priority Flag	Vector Table Address	Macro Service Type	Macro Service Mode Reg. Address
12	INTSER (asynchronous serial receive error)	Asynchronous serial interface	SERIF	SERMK	—	SERPR	0020H	None	None
13	INTSR (asynchronous serial receive complete)		SRIF	SRMK	SRISM	SRPR	0022H	A, B	FEDEH
14	INTST (asynchronous serial transmit complete)		STIF	STMK	STISM	STPR	0024H	A, B	FEDCH
15	INTCSI (clocked serial interface transfer complete)	Clocked serial interface	CSIIF	CSIMK	CSIISM	CSIPR	0026H	A, B	FEDAH

Notes:

(1) The default priority is a fixed numeric value indicating which interrupt takes precedence when more than one interrupt with the same priority has simultaneously occurred.

NEC

Section 5
μPD78K3 Product Line
16-/8-Bit, K-Series Microcontrollers

NEC
NEC Electronics Inc.

μPD78312A Family (μPD78310A/312A/P312A) 16-/8-Bit, K-Series Microcontrollers With Real-Time Output Ports

September 1993

Description

The μPD78310A, μPD78312A, and μPD78P312A are members of the K-Series® of microcontrollers and are designed for use in process control. They perform all the usual process control functions and are particularly well-suited for driving stepping motors and dc motors in servo loops. The processors include on-chip memory, timers, input/output registers, and a powerful interrupt handling facility. The μPD78310A/312A is constructed of high-speed CMOS circuitry and operates from a single +5-volt power supply.

The input frequency (maximum 12 MHz) is derived from an external crystal or an external oscillator. The internal processor clock is two-phase, and thus machine states are executed at a rate of 6 MHz. The shortest instructions require three states, making the minimum time 500 ns. The CPU contains a three-byte instruction prefetch queue, which allows a subsequent instruction to be fetched during execution of an instruction that does not reference memory.

Program memory is 8K bytes of mask-programmable ROM (μPD78312A only), and data memory is 256 bytes of static RAM. The μPD78310A is the ROMless version. μPD78P312A is a prototyping chip for μPD78312A. It has an on-chip 8K EPROM instead of a mask ROM.

Features

- Complete single-chip microcontroller
  - 16-bit ALU
  - 8K ROM (μPD78312A only)
  - 256 bytes RAM
  - 1-bit and 8-bit logic
- Instruction prefetch queue
- 16-bit unsigned multiply and divide
- String instructions
- Memory expansion
  - 8085A bus-compatible
  - Total 64K address space
- Large I/O capacity: up to 32 I/O port lines
- Extensive timer/counter system
  - Two 16-bit up/down counters
  - Quadrature counting
  - Two 16-bit timers
  - Free-running counter with two 16-bit capture registers
  - Pulse-width modulated outputs
  - Timebase counter
- Four-channel 8-bit A/D converter
- Two 4-bit real-time output ports
- Two nonmaskable interrupts
- Eight hardware priority interrupt levels
- Macro service facility for interrupts gives the effect of eight DMA channels
- Bidirectional serial port
  - Either UART or interface mode
  - Dedicated baud rate generator
- Watchdog timer
- Refresh output for pseudostatic RAM
- Programmable HALT and STOP modes
- One-byte call instruction
- On-chip clock generator
- CMOS silicon gate technology
- +5-volt power supply

5a

K-Series is a registered trademark of NEC Electronics, Inc.

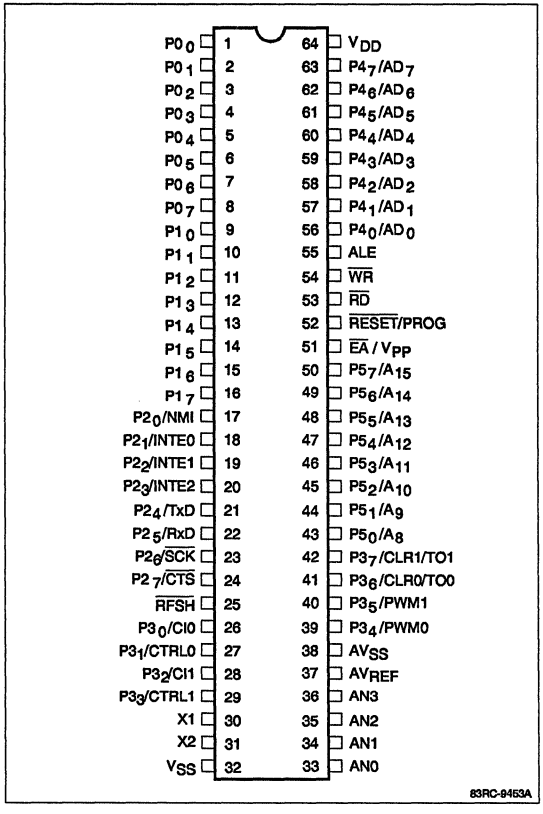
Ordering Information

Part Number	ROM	Package	Package Drawing
μPD78310ACW μPD78312ACW-xxx μPD78P312ACW	ROMless 8K mask ROM 8K OTP ROM	64-pin plastic shrink DIP	P64C-70-750 A,C
μPD78310AGF-3BE μPD78312AGF-xxx-3BE μPD78P312AGF-3BE	ROMless 8K mask ROM 8K mask ROM	64-pin plastic QFP	P64GF-100-3B8, 3BE-1
μPD78310AGQ-36 μPD78312AGQ-xxx-36 μPD78P312AGQ-36	ROMless 8K Mask ROM 8K OTP ROM	64-pin plastic QUIP	P64GQ-100-36
μPD78310AL μPD78312AL-xxx μPD78P312AL	ROMless 8K Mask ROM 8K OTP ROM	68-pin plastic PLCC	P68L-50A1-1
μPD78P312ADW	EPROM	64-pin ceramic shrink DIP with window (350 mil)	P64DW-70-750A
μPD78P312AR	EPROM	64-pin ceramic QUIP with window	P64RQ-100-A

xxx is the ROM code number.

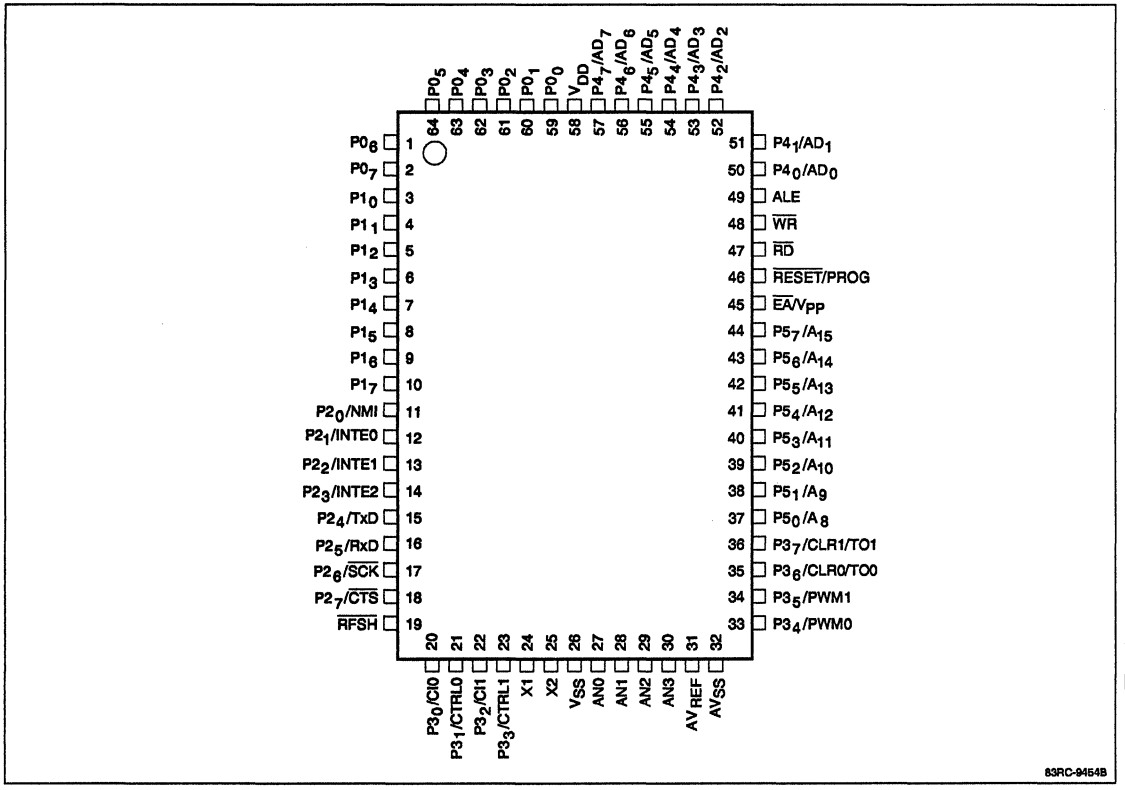
Pin Configurations

64-Pin Shrink DIP and QUIP, Plastic and Ceramic



Pin Configurations (cont)

64-Pin Plastic QFP

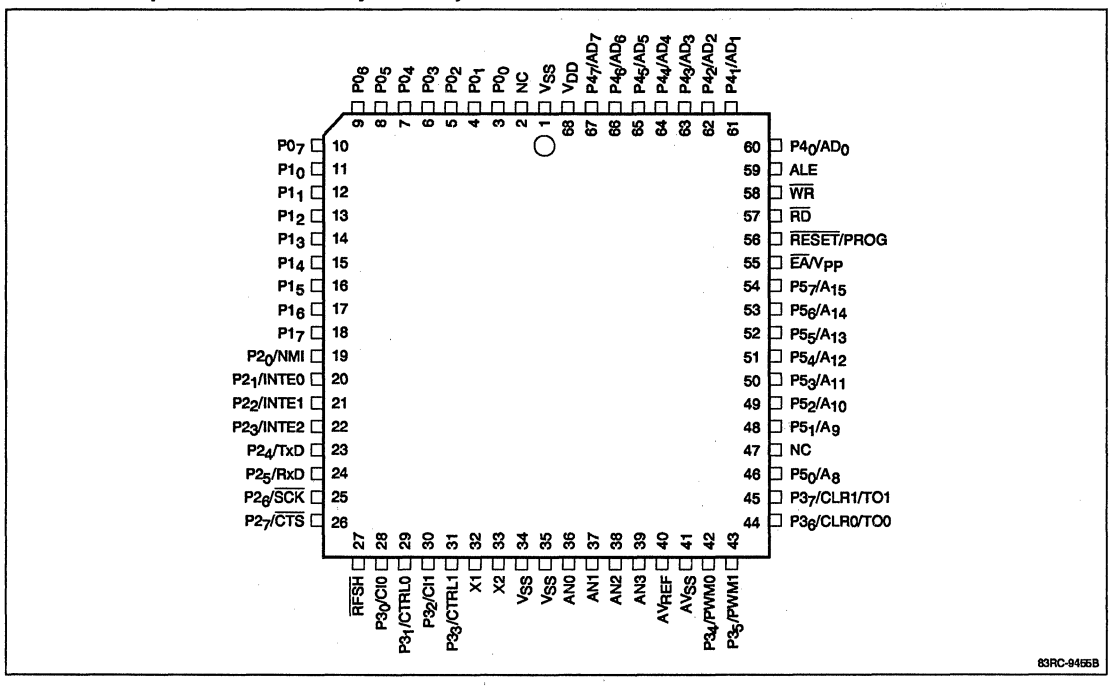


83RC-94548

5a

Pin Configurations (cont)

68-Pin PLCC (Plastic Leaded Chip Carrier)



Pin Identification

Symbol	Function
AN0 - AN3	A/D converter inputs
ALE	Address latch enable output
$\overline{EA}/V_{PP}$	External access control input; programming voltage
$P0_7$ - $P0_0$	I/O port 0
$P1_7$ - $P1_0$	I/O port 1
$P2_0$/NMI	Nonmaskable interrupt input
$P2_1$ - $P2_3$/ INTE0 - INTE2	Maskable interrupt inputs
$P2_4$/TxD	I/O port 2; serial transmit output
$P2_5$/RxD	I/O port 2; serial receive input
$P2_6/\overline{SCK}$	I/O port 2; serial clock output
$P2_7/\overline{CTS}$	I/O port 2; clear to send input
$P3_0$/CI0	Up/down counter 0 input
$P3_1$/CTRL0	Up/down counter 0 control input
$P3_2$/CI1	Up/down counter 1 input
$P3_3$/CTRL1	Up/down counter 1 control input
$P3_4$/PWM0	I/O port 3; pulse width modulated output 0
$P3_5$/PWM1	I/O port 3; pulse width modulated output 1
$P3_6$/CLR0/TO0	I/O port 3; counter 0 clear input timer 0 output
$P3_7$/CLR1/TO1	I/O port 3; counter 1 clear input; timer 1 output
$P4_7$ - $P4_0/AD_7$ - AD_0	I/O port 4; external address/data bus
$P5_7$ - $P5_0/A_{15}$ - A_8	I/O port 5; high address byte output
$\overline{RD}$	Read strobe output
$\overline{RESET}$/PROG	External reset input; PROM programming mode
$\overline{RFSH}$	Refresh output
$\overline{WR}$	Write strobe output
X1	External crystal or external clock input
X2	External crystal
AV_{REF}	A/D reference voltage
AV_{SS}	Analog ground
V_{DD}	Power supply
V_{SS}	Power return

Pin Functions

AN0 - AN3 (A/D Converter Inputs). AN0 - AN3 are the four program selectable input channels for the A/D converter.

ALE (Address Latch Enable). ALE is the address latch enable. It is to be used by external circuitry to latch the low-order 8 address bits during the first part of a read or write cycle.

$\overline{EA}/V_{PP}$. On μPD78312A, a low on $\overline{EA}$ enables use of external memory in place of on-chip ROM. The $\overline{EA}$ pin must be low on μPD78310A. On the μPD78P312A, this pin is used for programming voltage. In normal operation, it must be connected to V_{DD}.

$P0_7$ - $P0_0$ (Port 0). Port 0 consists of 8 bits, individually programmable for input/output or two 4-bit real-time (timer controlled) output ports.

$P1_7$ - $P1_0$ (Port 1). Port 1 consists of 8 bits, individually programmable for input/output.

$P2_0$/NMI (Port 2; Nonmaskable Interrupt). Port $P2_0$ is dedicated to NMI, the nonmaskable external interrupt request.

$P2_1$ - $P2_3$/INTE0-INTE2 (Port 2; Maskable Interrupts). Ports $P2_1$ - $P2_3$ are dedicated to INTE0, INTE1, and INTE2, the maskable external interrupt requests.

$P2_4$/TxD (Port 2; Serial Transmit). $P2_4$ is an I/O port bit or the transmitted serial data output.

$P2_5$/RxD (Port 2; Serial Receive). $P2_5$ is an I/O port bit or the received serial data input.

$P2_6/\overline{SCK}$ (Port 2; Serial Clock). $P2_6$ is an I/O port bit or the serial shift clock output.

$P2_7/\overline{CTS}$ (Port 2; Clear to Send). $P2_7$ is an I/O port bit or clear-to-send input (external serial transmission control) in the asynchronous communication mode. In the serial I/O interface mode, it becomes the serial receive clock I/O pin.

$P3_0$/CI0 (Port 3; Counter 0). Port $P3_0$ is dedicated to CI0, the external count input for up/down counter 0.

$P3_1$/CTRL0 (Port 3; Counter 0 Control). Port $P3_0$ is dedicated to CTRL0, the external control input for up/down counter 0.

$P3_2$/CI1 (Port 3; Counter 1). Port $P3_2$ is dedicated to CI1, the external count input for up/down counter 1.

$P3_3$/CTRL1 (Port 3; Counter 1 Control). Port $P3_3$ is dedicated to CTRL1, the external control input for up/down counter 1.

$P3_4$/PWM0 (Port 3; Pulse Width 0). $P3_4$ is an I/O port bit or the pulse-width modulated output 0.

$P3_5$/PWM1 (Port 3; Pulse Width 1). $P3_5$ is an I/O port bit or the pulse-width modulated output 1.

$P3_6$/CLR0/TO0 (Port 3; Counter 0 Clear; Timer 0). $P3_6$ is an I/O port bit, or the clear input for up/down counter 0, or the timer 0 flip-flop output.

$P3_7$/CLR1/TO1 (Port 3; Counter 1 Clear; Timer 1). $P3_7$ is an I/O port bit, or the clear input for up/down counter 1, or the timer 1 flip-flop output.

$P4_0$ - $P4_7$/AD_0 - AD_7 (Port 4; External Address/Data Bus). Port 4 consists of 8 bits, programmable as a unit for input or output, or as the multiplexed address/data bus if external memory or external interface circuitry is used. The port is controlled by the memory mapping register. If the $\overline{EA}$ pin is low, port 4 is always an address/data bus.

$P5_0$ - $P5_7$/A_8 - A_{15} (Port 5; High-Address Byte). Port 5 consists of 8 bits, individually programmable for input or output, or the high-order address bits for external memory. Under control of the memory mask register, bits $P5_3$ - $P5_0$ are used for 4K memory expansion, bits $P5_5$ - $P5_0$ for 16K memory expansion, or bits $P5_7$ - $P5_0$ for 56K memory expansion. If the $\overline{EA}$ pin is low, port 5 is always the high-order address bus.

$\overline{RD}$ (Read Strobe). $\overline{RD}$ is the read strobe output. It is to be used by external memory (or data registers) to place data on the I/O bus during a read operation.

$\overline{RESET}$/PROG. This pin is used for the external reset input. A low level sets all registers to their specified reset values. During programming of the μPD78P312A, this pin is used to place the device into PROM programming mode.

$\overline{RFSH}$ (Refresh). $\overline{RFSH}$ is the refresh pulse output to be used for external pseudostatic RAM.

$\overline{WR}$ (Write Strobe). $\overline{WR}$ is the write strobe output. It is to be used by external memory (or data registers) to latch data from the I/O bus during a write operation.

X1, X2 (External Crystal or Clock Input). X1 and X2 are the external oscillator inputs or the connections for an external crystal. If an external clock is used, it is connected to X1 and its inverse is connected to X2. The system clock frequency is half the input frequency.

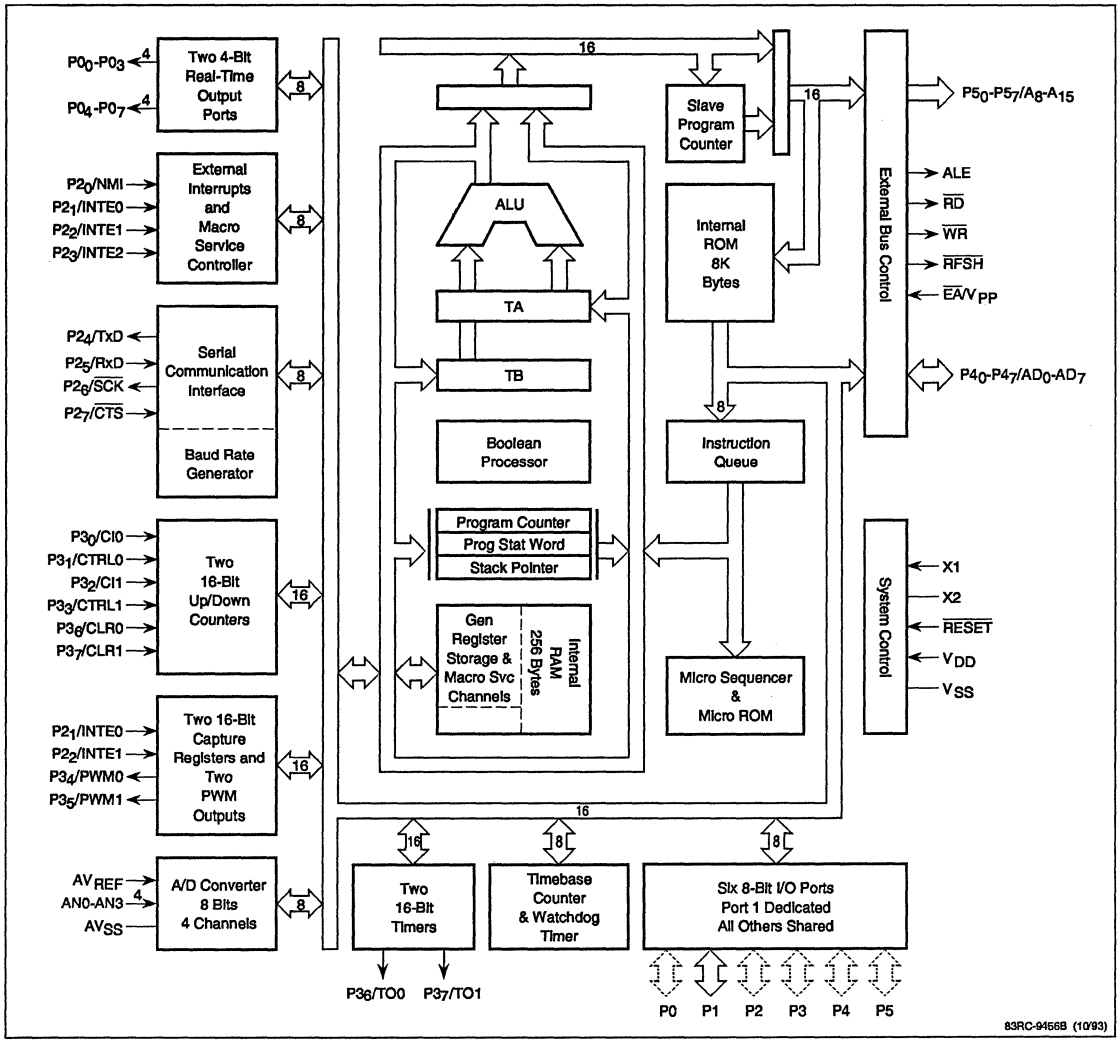
AV_{REF} (A/D Reference Voltage). AV_{REF} is the reference voltage input for the A/D converter.

AV_{SS} (Analog Ground). AV_{SS} is the analog ground pin.

V_{DD} (Power Supply). V_{DD} is the positive power supply input.

V_{SS} (Power Return). V_{SS} is the power supply return, normally ground.

Block Diagram



5a

FUNCTIONAL DESCRIPTION

On-chip features designed to facilitate process control include two 16-bit timers, quadrature counting, two 16-bit up/down counters, two pulse-width modulated outputs, a free-running counter with two capture registers, two 4-bit real-time (timer-controlled) output ports, an 8-bit A/D converter with four input channels, a timebase counter to generate widely spaced interrupts, and a watchdog timer to guard against infinite program loops.

In addition, a serial I/O port can be used in either an interface mode or an asynchronous communication mode. HALT and STOP modes are provided to conserve power at times when CPU action is not required.

All I/O, timer, and control registers are defined as special function registers and assigned addresses in the top 256 bytes of memory. The special function registers may be operated on directly by many of the arithmetic, logic, and move instructions of the CPU. Table 1 describes the registers.

Addressing

The μPD78312A family features 1-byte addressing of the special function registers and 1-byte addressing of the internal RAM. There are nine modes of addressing main memory, including autoincrement, autodecrement, indexing, and double indexing. There are 8- and 16-bit immediate operands.

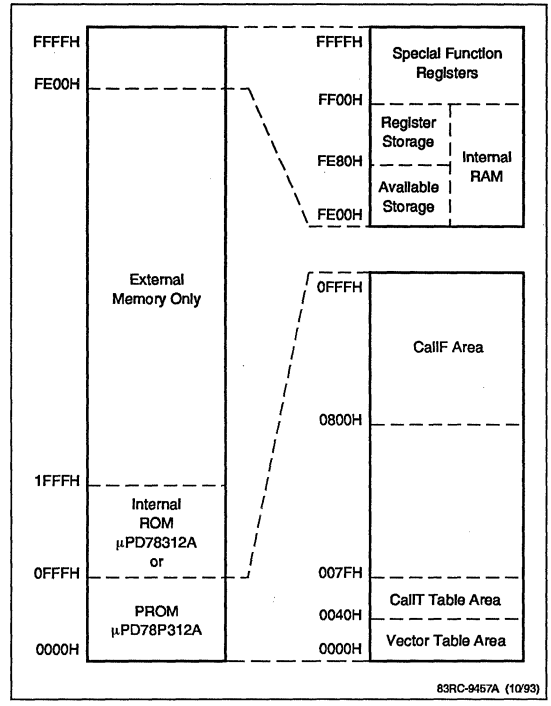
External Memory

External memory (figure 1) is supported by I/O port 4, an 8-bit multiplexed address/data bus. The memory mapping register controls the size of external memory as well as the number of additional wait states. High-order address bits are taken from I/O port 5 as required. No bits are required for 256 bytes of external memory; bits $P5_3$ - $P5_0$ are used for 4K bytes, $P5_5$ - $P5_0$ for 16K bytes, and $P5_7$ - $P5_0$ for 56K bytes. Any remaining port 5 bits are available for I/O.

Refresh

The μPD78312A has a refresh signal for use with the pseudostatic RAM. The refresh cycle can be set to one of four intervals ranging from 2.67 to 21.3 μs. The refresh is timed to follow a read or write operation so that there is no interference.

Figure 1. Memory Map

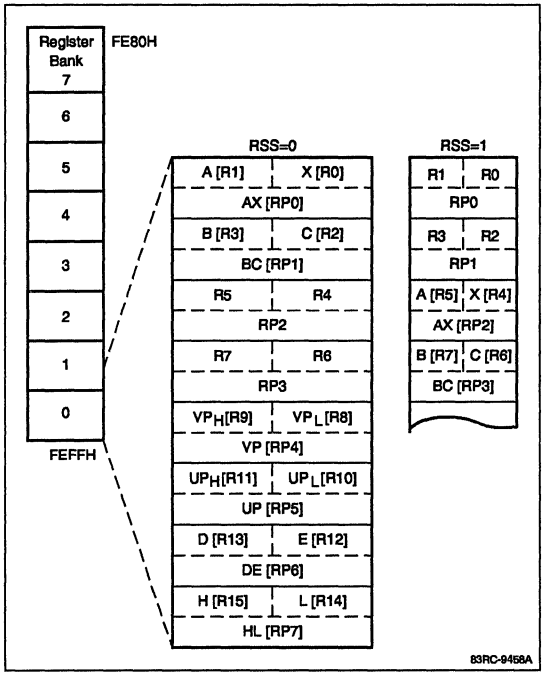


General Registers

The CPU has sixteen 8-bit registers (figure 2) that can also be used in pairs to function as 16-bit registers. A complete set of 16 general registers is mapped into each of 8 program-selectable register banks stored in RAM. Three bits in the PSW specify which of the register banks is active at any given time. Each register bank has two program-selectable accumulators.

The general registers of the μPD78312A have both absolute and functional names. AX is the functional name for the accumulator. Setting the RSS bit in the PSW to 1 transfers the AX and BC registers from their normal RP0 and RP1 positions to RP2 and RP3 as shown in figure 2. This adds considerable programming flexibility.

Figure 2. Register Designation and Storage



Program Status Word

Following is the program status word format.

0	RB_2	RB_1	RB_0	0	0	IE	0

15 8

S	Z	RSS	AC	UF	P/V	SUB	CY

7 0

RB_2 - RB_0	Active register bank number
IE	Interrupt enable
S	Sign (1 if last result was negative)
Z	Zero (1 if last result was zero)
RSS	Register set select
AC	Auxiliary carry (carry out of 3rd bit)
UF	User flag
P/V	Parity or arithmetic overflow
SUB	Subtract (1 if last operation was subtract)
CY	Carry

Input/Output

All ports may be used for either latched output or high-impedance input. All ports except port 4 are bit-programmable for input or output. Port 0 is used for real-time or normal I/O. Port 1 is used for normal I/O. The low nibble of ports 2 and 3 is always used for control and the high nibble for control or normal I/O. Port 4 is used for the external address/data bus or byte-programmable I/O. Port 5 is used for the high bits of the external address or for normal I/O.

Real-Time Output Port

The real-time output port shares pins with I/O port 0. The high and low nibbles are treated separately or together. Data is transferred from a buffer to the port latches on either a timer or software command.

Serial Port

The serial port can operate in UART or interface mode with the baud rate and byte format under program control. The serial port also includes a dedicated baud rate generator.

Pulse-Width Modulated Outputs

The two independent pulse-width modulated outputs are controlled by two 16-bit modulus registers and counters. There are four programmable repetition rates ranging from 91.6 Hz to 23.4 MHz. Figure 3 shows one of these outputs.

5a

Timers

The μPD78312A has two 16-bit timers. The inputs to these timers may be the internal clock divided by 6; 12(TM0) or 128. Each timer has an associated modulus register to store the timer count. The timer counts down to zero, sets a flag, reloads from the modulus register, and then counts down again. The timer flags can be used under program control to generate interrupt requests and/or a square-wave output. TM0 also functions optionally as two one-shot timers.

Figure 4 is a diagram of the interval timers.

There is a free-running counter that counts the internal clock divided by 4 or by 16. The counter has two 16-bit capture registers. Capture is triggered by an external interrupt request or by the up/down counter clock.

The timebase counter generates a signal at one of four intervals ranging from 170 μs to 175 ms. The signal can be used to generate an interrupt request and/or an up/down counter capture.

Figure 3. Pulse-Width Modulated Output

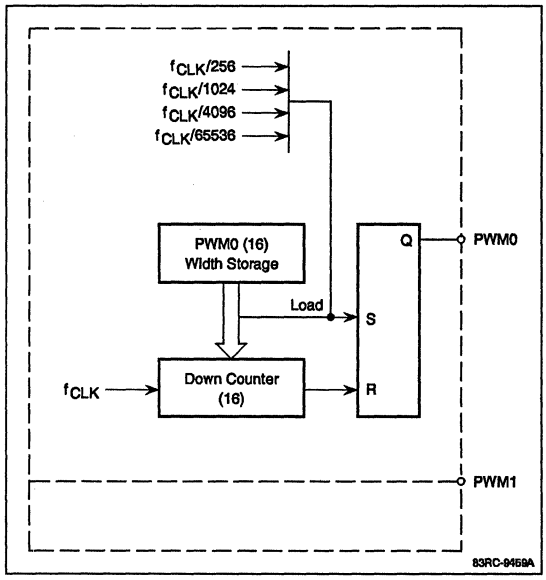
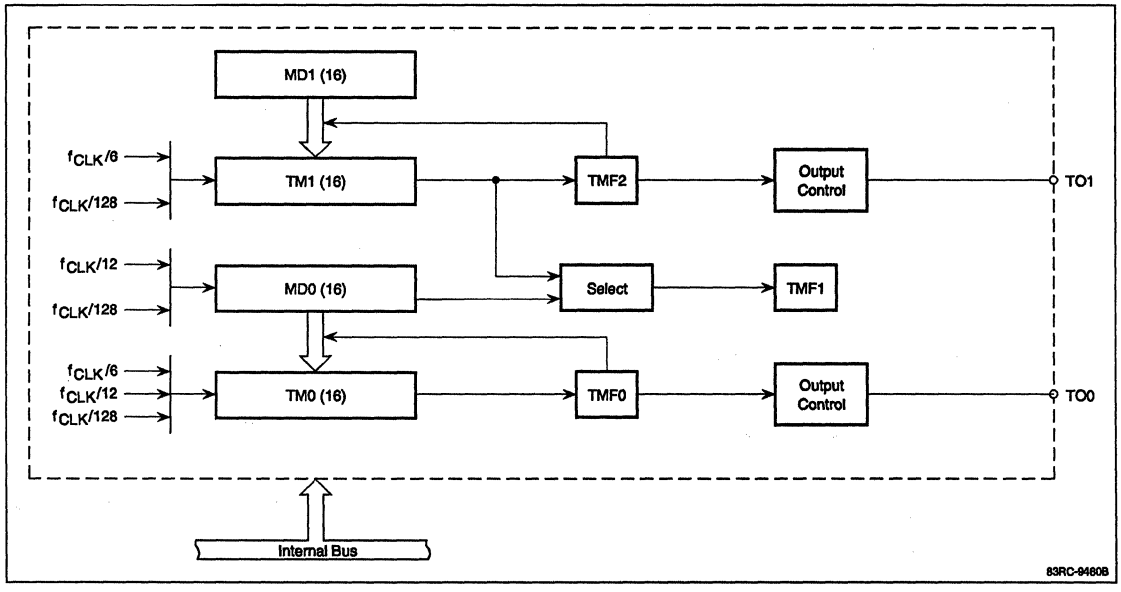


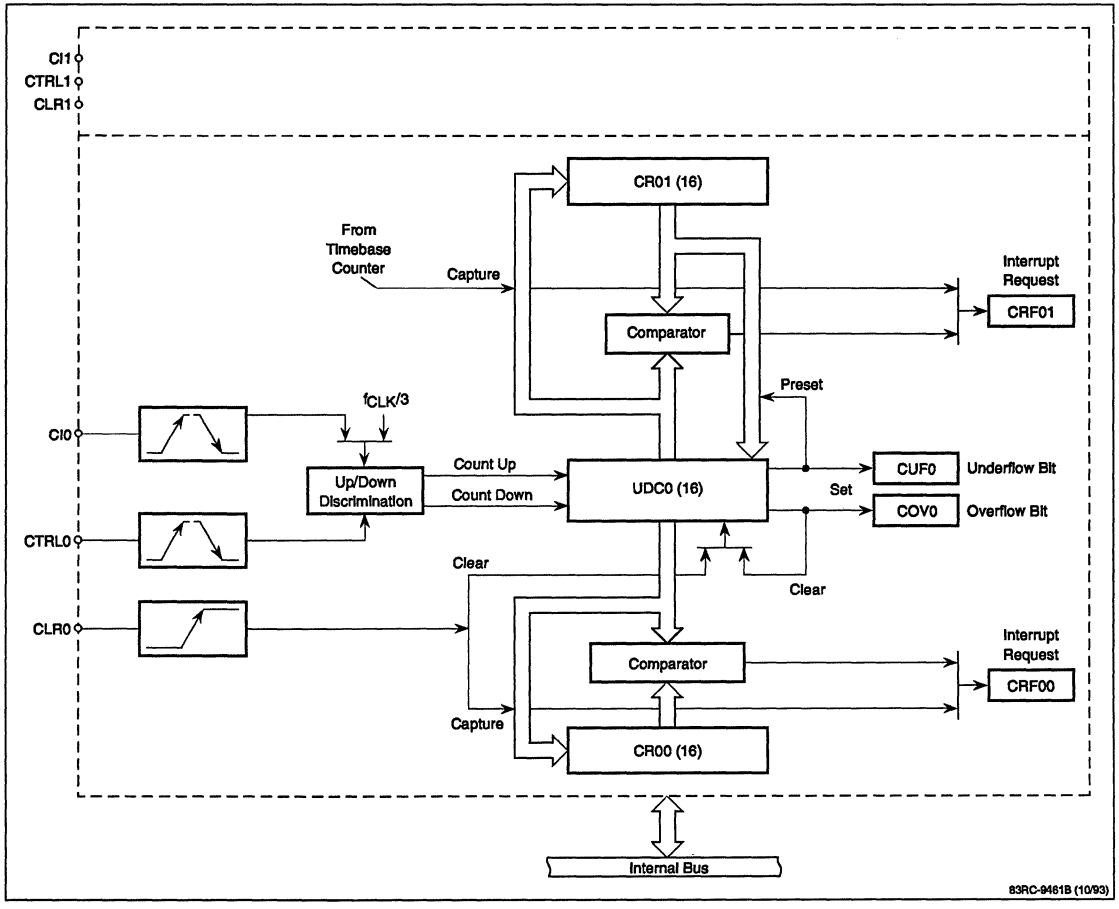
Figure 4. Timer Block Diagram



Up/Down Counters

The μPD78312A has two 16-bit up/down counters, each of which has two capture/compare registers. There are three modes of operation: compare and interrupt, capture on external command, and capture on timebase counter command. There are five sources of counts: the internal clock divided by 3, the external clock, external independent up and down inputs, external clock with direction control, and external clock with automatic up/down discrimination. Figure 5 shows an up/down counter.

Figure 5. Up/Down Counter Block Diagram

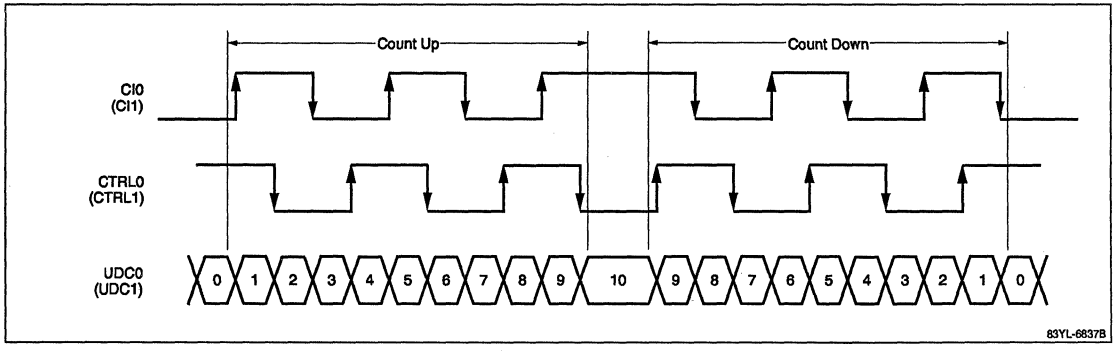


5a

Quadrature Counting

The two up/down counters, UDC0 and UDC1, have an optional quadrature counting mode, which is activated by specifying mode 4 in the counter unit input mode register, CUIM. It is designed to count the output of a two-phase pulsed optical shaft angle encoder. The input for phase A is the CI0 (or CI1) pin, and the input for phase B is the CTRL0 (or CTRL1) pin. The counter UDC0 (or UDC1) is incremented or decremented at both positive and negative transitions of both input signals. Whether it is incremented or decremented is dependent upon the relative phase of the two signals as illustrated in figure 6.

Figure 6. Counter Operation (Mode 4)



Standby Modes

HALT and STOP modes conserve power when CPU action is not required. In HALT mode, the CPU stops and the clock continues to run. Maskable interrupts can restart the CPU.

In STOP mode, the CPU and clock are both stopped. A $\overline{\text{RESET}}$ pulse or the nonmaskable external interrupt is required to restart them. There is also the option of slowing the system clock by a factor of four. The standby control register controls the standby modes and is a protected location written to only by a special instruction.

Watchdog Timer

The watchdog timer protects against inadvertent program loops. A nonmaskable interrupt occurs if the timer is not reset before a timeout occurs. There are four program-selectable intervals ranging from 5.5 to 349.3 ms. The watchdog timer can be disabled by software. The watchdog timer mode register controls the watchdog timer and is a protected location written to only by a special instruction.

A/D Converter

The A/D converter has four input channels and can operate in either scan or select mode. The A/D converter performs 8-bit successive approximation conversions, has a 30-μs conversion time, and is triggered either internally or externally. The A/D converter includes an on-chip sample and hold amplifier.

Interrupts

There are two nonmaskable interrupt sources: the external nonmaskable interrupt and the watchdog timer. Their relative priorities are software selectable.

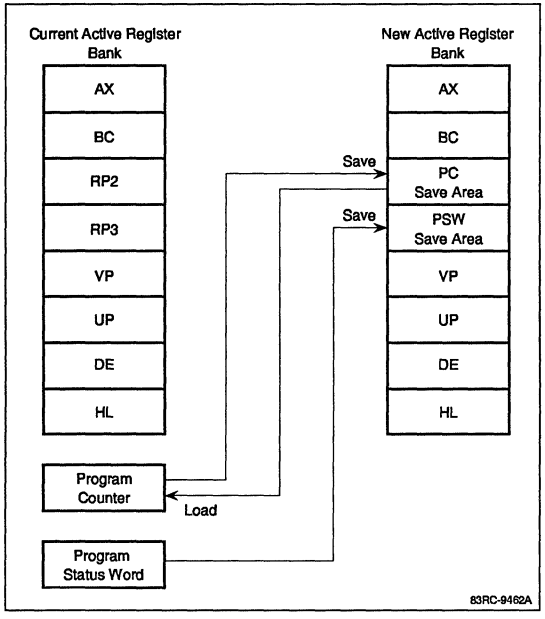
There are eight hardware priority interrupt levels, level 0 having the highest priority and level 7 the lowest. The 15 maskable interrupt sources (table 2) are divided into five groups, and each group can, under program control, be assigned to any one of the priority levels.

Interrupts may be serviced by routines entered either by vectoring or by context switching. Context switching automatically saves all the general registers, the

program status word, and the program counter. Figure 7 illustrates the mechanism of context switching.

Finally, an optional macro service function transfers data between any one special function register and memory without program intervention.

Figure 7. Hardware Context Switching

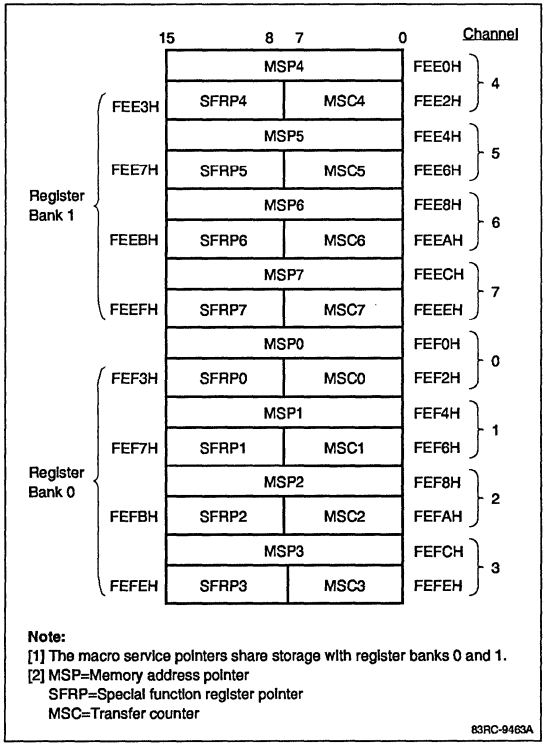


Macro Service

The macro service controller can be programmed to perform word or byte transfers. It can transfer data from a special function register to memory or from memory to a special function register. Transfer events are triggered by interrupt requests and take place without software intervention.

There are eight macro service channels; channel control information is stored in RAM. This information (figure 8) consists of a 16-bit memory address (optionally incremented at each transfer), and 8-bit special function register designator, and an 8-bit transfer counter (decremented at each transfer). When the count equals 0, a context switch or vectored interrupt occurs.

Figure 8. Macroservice Pointer Addresses



Note:
[1] The macro service pointers share storage with register banks 0 and 1.
[2] MSP=Memory address pointer
SFRP=Special function register pointer
MSC=Transfer counter

83RC-9463A

5a

Table 1. Special Function Registers

Address	Register (SFR)	Symbol		R/W	16-Bit Transfer	State After Reset
FF00H	I/O port 0	P0		R/W	No	Undefined
FF01H	I/O port 1	P1		R/W	No	Undefined
FF02H	I/O port 2	P2		R/W (Note 1)	No	Undefined
FF03H	I/O port 3	P3		R/W (Note 1)	No	Undefined
FF04H	I/O port 4	P4		R/W	No	Undefined
FF05H	I/O port 5	P5		R/W	No	Undefined
FF08H FF09H	Capture/compare register 00	CR00L CR00H	CR00	R/W	Yes	Undefined
FF0AH FF08H	Capture/compare register 01	CR01L CR01H	CR01	R/W	Yes	Undefined
FF0CH FF0DH	Capture/compare register 10	CR10L CR10H	CR10	R/W	Yes	Undefined
FF0EH FF0FH	Capture/compare register 11	CR11L CR11H	CR11	R/W	Yes	Undefined
FF10H F11H	Capture register 0 (from FRC)	CPT0L CPT0H	CPT0	R/W	Yes	Undefined
FF12H FF13H	Capture register 1 (from FRC)	CPT1L CPT1H	CPT1	R/W	Yes	Undefined
FF14H FF15H	PWM register 0 (duration)	PWM0L PWM0H	PWM0	R/W	Yes	Undefined
FF16H FF17H	PWM register 1 (duration)	PWM1L PWM1H	PWM1	R/W	Yes	Undefined
FF1CH FF1DH	Presettable up/down counter 0	UDC0L UDC0H	UDC0	R/W	Yes	Undefined
FF1EH FF1FH	Presettable up/down counter 1	UDC1L UDC1H	UDC1	R/W	Yes	Undefined
FF20H	Port 0 mode register	PM0		R/W	No	FFH
FF21H	Port 1 mode register	PM1		R/W	No	FFH
FF22H	Port 2 mode register	PM2		R/W (Note 1)	No	FFH
FF23H	Port 3 mode register	PM3		R/W (Note 1)	No	FFH
FF25H	Port 5 mode register	PM5		R/W	No	FFH
FF32H	Port 2 mode control register	PMC2		R/W	No	0FH
FF33H	Port 3 mode control register	PMC3		R/W	No	0FH
FF38H	Real-time output port control register	RTPC		R/W	No	08H
FF3AH FF3BH	Port 0 buffer register (Note 2)	P0L P0H		R/W	No	Undefined
FF40H	Memory expansion mode register	MM		R/W	No	30H
FF41H	Refresh mode register	RFM		R/W	No	10H
FF42H	Watchdog timer mode register	WDM		R/W	No	00H
FF44H	Standby control register	STBC		R/W	No	2nH (Note 3)
FF46H	Timebase mode register	TBM		R/W	No	00H
FF48H	External interrupt mode register	INTM		R/W	No	00H
FF4AH	In-service priority register	ISPR		R	No	00H
FF4EH	CPU control word	CCW		R/W	No	00H

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol		R/W	16-Bit Transfer	State After Reset
FF50H	Serial communication mode register	SCM		R/W	No	00H
FF52H	Serial communication control register	SCC		R/W	No	00H
FF53H	Baud rate generator	BRG		R/W	No	00H
FF56H	Serial communication receive buffer	RXB		R	No	Undefined
FF57H	Serial communication transmit buffer	TXB		W	No	Undefined
FF60H	Free-running counter control register	FRCC		R/W	No	00H
FF64H	Capture mode register	CPTM		R/W	No	00H
FF66H	PWM mode register	PWMM		R/W	No	00H
FF68H	A/D converter mode register	ADM		R/W	No	00H
FF6AH	A/D converter result register	ADCR		R	No	Undefined
FF70H	Count unit input mode register	CUIM		R/W	No	00H
FF72H	Up/down counter control register 0	UDCC0		R/W	No	00H
FF74H	Capture/compare control register	CRC		R/W	No	00H
FF7AH	Up/down counter control register 1	UDCC1		R/W	No	00H
FF80H	Timer 0 control register	TMC0		R/W	No	00H
FF82H	Timer 1 control register	TMC1		R/W	No	00H
FF88H FF89H	Timer 0	TM0L TM0H	TM0	R/W	Yes	Undefined
FF8AH FF8BH	Modulus/timer register 0	MD0L MD0H	MD0	R/W	Yes	Undefined
FF8CH FF8DH	Timer 1	TM1L TM1H	TM1	R/W	Yes	Undefined
FF8EH FF8FH	Modulus register 1	MD1L TM1H	MD1	R/W	Yes	Undefined
FFB0H to FFBFH	External area (Note 4)					
FFC0H	CRF00 interrupt control Up/down counter 0	CRIC00		R/W	No	47H
FFC1H	CRF00 macro service control Up/down counter 0	CRMS00		R/W	No	Undefined
FFC2H	CRF01 interrupt control Up/down counter 0	CRIC01		R/W	No	47H
FFC4H	CRF10 Interrupt control Up/down counter 1	CRIC10		R/W	No	47H
FFC5H	CRF10 macro service control Up/down counter 1	CRMS10		R/W	No	Undefined
FFC6H	CRF11 interrupt control Up/down counter 1	CRIC11		R/W	No	47H
FFC8H	EXIF0 interrupt control External interrupt INTE0	EXIC0		R/W	No	47H
FFC9H	EXIF0 macro service control External interrupt INTE0	EXMS0		R/W	No	Undefined
FFCAH	EXIF1 interrupt control External interrupt INTE1	EXIC1		R/W	No	47H

5a

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	16-Bit Transfer	State After Reset
FFCBH	EXIF1 macro service control External interrupt INTE1	EXMS1	R/W	No	Undefined
FFCCH	EXIF2 interrupt control External interrupt INTE2	EXIC2	R/W	No	47H
FFCDH	EXIF2 macro service control External interrupt INTE2	EXMS2	R/W	No	Undefined
FFCEH	TMF0 interrupt control Timer flag	TMIC0	R/W	No	47H
FFCFH	TMF0 macro service control Timer flag	TMMS0	R/W	No	Undefined
FFD0H	TMF1 interrupt control Timer flag	TMIC1	R/W	No	47H
FFD1H	TMF1 macro service control Timer flag	TMMS1	R/W	No	Undefined
FFD2H	TMF2 interrupt control Timer flag	TMIC2	R/W	No	47H
FFD3H	TMF2 macro service control Timer flag	TMMS2	R/W	No	Undefined
FFDAH	Receive error interrupt control Serial port	SEIC	R/W	No	47H
FFDCH	Receive interrupt control Serial port	SRIC	R/W	No	47H
FFDDH	Receive macro service control Serial port	SRMS	R/W	No	Undefined
FFDEH	Transmit interrupt control Serial port	STIC	R/W	No	47H
FFDFH	Transmit macro service control Serial port	STMS	R/W	No	Undefined
FFE0H	A/D converter interrupt control	ADIC	R/W	No	47H
FFE1H	A/D converter macro service control	ADMS	R/W	No	Undefined
FFE2H	Timebase counter interrupt control	TBIC	R/W	No	47H

Notes:

(1) Bits 0-3 of port 2 and of port 3 are read-only.

(2) P0H and P0L are 4-bit buffer registers used to store data to be loaded into the high and low nibbles of the real-time output (P0). The high order 4 bits of P0H and the low order 4 bits of P0L are used.

(3) Bit 3 of the STBC is not affected by $\overline{\text{RESET}}$ (n = 0 or 8).

(4) External registers interfaced with these addresses can be accessed by special function register addressing.

Table 2. Interrupt Sources and Vector Addresses

	Default Priority	Mnemonic	Interrupt Source	Macroservice	Vector
Software	—	BRK	Break instruction	No	003EH
Nonmaskable Interrupts		NMI	External nonmaskable interrupt	No	0002H
	—	WDT	Watchdog timer	No	000AH
Maskable interrupts	0	CRF00	Up/down counter 0	Yes	001AH
	1	CRF01	Up/down counter 0	No	001CH
	2	CRF10	Up/down counter 1	Yes	001EH
	3	CRF11	Up/down counter 1	No	0020H
	4	EXIF0	External interrupt 0	Yes	0004H
	5	EXIF1	External interrupt 1	Yes	0006H
	6	EXIF2	External interrupt 2	Yes	0008H
	7	TMF0	Timer flag 0	Yes	000EH
	8	TMF1	Timer flag 1	Yes	0010H
	9	TMF2	Timer flag 2	Yes	0012H
	10	SEF	Serial port error	No	0022H
	11	SRF	Serial port receive buffer	Yes	0024H
	12	STF	Serial port transmit buffer	Yes	0026H
	13	ADF	A/D converter done flag	Yes	0028H
	14	TBF	Timebase counter flag	No	000CH
Reset	—	$\overline{\text{RESET}}$	External reset line	—	0000H

5a

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = +25°C

Power supply voltage V_{DD}	−0.5 to +7.0 V
Reference voltage, AV_{REF}	−0.5 V to V_{DD} +0.3 V
Power supply return, AV_{SS}	−0.5 to +0.5 V
Input voltage, V_{I1} (except $\overline{RESET}$ of μPD78P312A)	−0.5 to + V_{DD} + 0.5
Input voltage, V_{I2} ($\overline{RESET}$ of μPD78P312A only)	−0.5 to +13.5 V
Output voltage, V_O	−0.5 to V_{DD} +0.5 V
Output current, low; I_{OL} (single pin)	4 mA
Output current, low; I_{OL}; total, all output pins (μPD78312/310A)	100 mA
Output current, low; I_{OL}; total, all output pins (μPD78P312A)	60 mA
Output current, high; I_{OH} (single pin)	−1 mA
Output current, high; I_{OH}; total, all output pins (μPD78312/310A)	−25 mA

Absolute Maximum Ratings (cont)

Output current, high; I_{OH}; total, all output pins (μPD78P312A)	−15 mA
Operating temperature, T_{OPT}	−10 to +70°C
Storage temperature, T_{STG}	−65 to +150°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC characteristics.

Operating Frequency

Oscillator Frequency f_{XX}	T_A	V_{DD}
4 MHz ≤ f_{XX} ≤ 12 MHz	−10 to +70°C	+5.0 V 10%

Capacitance

T_A = +25°C; V_{DD} = V_{SS} = 0 V

Parameter	Symbol	Max	Unit	Conditions
Input capacitance	C_I	10	pF	f = 1 MHz; unmeasured pins returned to 0 V.
Output capacitance	C_O	20	pF	
I/O capacitance	C_{IO}	20	pF	

DC Characteristics

T_A = −10 to +70°C; V_{DD} = +5.0 V ±5%; V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input low voltage	V_{IL1}	0		0.8	V	Except $\overline{EA}$ on μPD78310A/312A
	V_{IL2}	0		0.5	V	$\overline{EA}$ on (μPD78310A/312A only)
Input high voltage	V_{IH1}	2.2		V_{DD}	V	Except $P2_0$/NMI, X1, X2, $\overline{RESET}$
	V_{IH2}	3.8		V_{DD}	V	$P2_0$/NMI X1, X2, $\overline{RESET}$
Output low voltage	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output high voltage	V_{OH}	V_{DD} −1			V	I_{OH} = −1 mA
Input current	I_I			±10	μA	$P2_0$/NMI, $\overline{RESET}$ V_I = 0.45 V to V_{DD}
Input leakage current	I_{LI}			±10	μA	
Input/output leakage current	I_{LO}			±10	μA	
AV_{REF} current	AI_{REF}		1.5	5	mA	f_{CLK} = 6 MHz
V_{DD} supply current	I_{DD1}		30	60	mA	Operating mode; f_{CLK} = 6 MHz
	I_{DD2}		5	15	mA	Halt mode; f_{CLK} = 6 MHz
Data retention voltage	V_{DDDR}	2.5			V	Stop mode
Stop mode supply current	I_{DDDR}		3	15	μA	Stop mode; V_{DDDR} = 2.5 V
			10	50	μA	Stop mode; V_{DDDR} = 5.0 V ±10%

AC Characteristics

T_A = −10 to +70°C; V_{DD} = +5.0 V ±10%; V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Read/Write Operation						
System clock cycle time	t_{CYK}	166		1000	ns	(Note 1)
Address setup time to ALE ↓	t_{SAL}	150			ns	
Address hold time after ALE ↓	t_{HLA}	30			ns	(Note 4)
Address to $\overline{RD}$ ↓ delay time	t_{DAR}	230			ns	
$\overline{RD}$ ↓ to address floating	t_{FRA}			0	ns	
Address to data input	t_{DAID}			410	ns	
ALE ↓ to data input	t_{DLID}			230	ns	
$\overline{RD}$ ↓ to data input	t_{DRID}			180	ns	
ALE ↓ to $\overline{RD}$ ↓ delay time	t_{DLR}	60			ns	
Data hold time after $\overline{RD}$ ↑	t_{HRID}	0			ns	
$\overline{RD}$ ↑ to address active	t_{DRA}	50			ns	
$\overline{RD}$ ↑ to ALE ↑ delay time	t_{DRL}	100			ns	
$\overline{RD}$ width low	t_{WRL}	200			ns	
ALE width high	t_{WLH}	120			ns	
Address to $\overline{WR}$ ↓ delay time	t_{DAW}	300			ns	
ALE ↓ to data output	t_{DLOD}			190	ns	
$\overline{WR}$ ↓ to data output	t_{DWOD}			100	ns	
ALE ↓ to $\overline{WR}$ ↓ delay time (Note 2)	t_{DLW}	30			ns	
		110			ns	During refresh mode
Data setup time to $\overline{WR}$ ↑	t_{SODWR}	150			ns	
Data setup time to $\overline{WR}$ ↓ (Note 3)	t_{SODWF}	30			ns	During refresh mode
Data hold time to $\overline{WR}$ ↑	t_{HWOD}	20			ns	(Note 4)
$\overline{WR}$ ↑ to ALE ↑ delay time	t_{DWL}	110			ns	
$\overline{WR}$ width low	t_{WWL}	200			ns	
Serial Port						
Serial clock cycle time	t_{CYSK}	1.33			μs	$\overline{SCK}$ output (Note 5)
		1.33			μs	$\overline{CTS}$ output (Note 6)
		1			μs	$\overline{CTS}$ input (Note 7)
Serial clock low-level width	t_{WSKL}	580			ns	$\overline{SCK}$ output (Note 5)
		580			ns	$\overline{CTS}$ output (Note 6)
		420			ns	$\overline{CTS}$ input (Note 7)
Serial clock high-level width	t_{WSKH}	580			ns	$\overline{SCK}$ output (Note 5)
		580			ns	$\overline{CTS}$ output (Note 6)
		420			ns	$\overline{CTS}$ input (Note 7)
$\overline{CTS}$ high, low level	t_{WCSH}, t_{WCSL}	3			t_{CYK}	Asynchronous mode
RxD setup time to $\overline{CTS}$ ↑	t_{SRXSK}	80			ns	
RxD hold time after $\overline{CTS}$ ↑	t_{HSKRX}	80			ns	
$\overline{SCK}$ ↓ to TxD delay time	t_{DSKTX}			210	ns	

5a

AC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
A/D Converter						
T_A = −10°C to +70°C; V_{DD} = +5 V ±10%; AV_{REF} = 4.0 V to V_{DD}; AV_{SS} = V_{SS} = 0 V						
Resolution		8			Bit	
Full scale error				0.4	%	t_{CYK} = 166 to 500 ns
Quantization error				±1/2	LSB	
Conversion time	t_{CONV}	180			t_{CYK}	t_{CYK} = 166 to 250 ns
		120			t_{CYK}	t_{CYK} = 250 to 500 ns
Sampling time	t_{SAMP}	36			t_{CYK}	t_{CYK} = 166 to 250 ns
		24			t_{CYK}	t_{CYK} = 250 to 500 ns
Analog input voltage	V_{IAN}	0		AV_{REF}	V	
Input impedance	R_{AN}		1000		MΩ	
Analog reference voltage	AV_{REF}	4.0		V_{DD}	V	
AV_{REF} current	AI_{REF}		1.5	5.0	mA	f_{CLK} = 6 MHz
Counter Operation						
CI0, CI1 high, low levels	t_{WCIH}, t_{WCIL}	3			t_{CYK}	
CTRL0, CRTL1 high, low levels	t_{WCTH}, t_{WCTL}	3			t_{CYK}	
CTRL0, CTRL1 setup time to CI ↑	t_{SCTCI}	2			t_{CYK}	Operating mode of count unit is set to mode 3. CI input is set to rising edge active.
CTRL0, CTRL1 hold time after CI ↑	t_{HCICT}	5			t_{CYK}	
CLRO. CLR1 high, low-level width	t_{WCRH}, t_{WCRL}	3			t_{CYK}	
CI0, CI1 setup time to CTRL	t_{S4CTCI}	6			t_{CYK}	Counter mode 4
CTRL0, CTRL1 setup time to CI	t_{H4CTCI}	6			t_{CYK}	Counter mode 4
CI0/CI1, CTRL0/CTRL1 cycle time	t_{CYC4}			250	kHz	Counter mode 4
External Interrupts and Reset						
NMI high, low-level width	t_{WNIH}, t_{WNIL}	10			μs	
INTE0 high, low-level width	t_{WI0H}, t_{WI0L}	3			t_{CYK}	
INTE1 high, low-level width	t_{WI1H}, t_{WI1L}	3			t_{CYK}	
INTE2 high, low-level width	t_{WI2H}, t_{WI2L}	3			t_{CYK}	
$\overline{\text{RESET}}$ high, low-level width	t_{WRSH}, t_{WRSL}	10			μs	
V_{DD} rise, fall time	t_{RVD}, t_{FVD}	200			μs	

Notes:

(1) The internal clock (f_{CLK}) equals the oscillation clock (f_{XX}) divided by 2 or 8 as determined by bit 5 of the STBC. In this table, f_{XX} = 12 MHz and $f_{CLK} = f_{XX}/2$.

(2) During refresh operation, the $\overline{WR}$ signal falls to low level 1/2 clock cycle later than if there is no refresh.

(3) When accessing data from pseudostatic RAMs (e.g. μPD4168) with the falling edge of the $\overline{WR}$ signal, the data setup time is t_{SODWF} instead of t_{SODWR}.

(4) Hold time is measured with C_L = 100 pF and R_L = 2 kΩ load, and includes the period necessary to guarantee V_{OH} and V_{OL}.

(5) I/O interface mode transmit data at a data rate of 750 kb/s.

(6) I/O interface mode receive data, internal clock, at a data rate of 750 kb/s.

(7) In the I/O interface mode this is the optional external clock for received data at a maximum rate of 1 MB/s.

Oscillator Characteristics

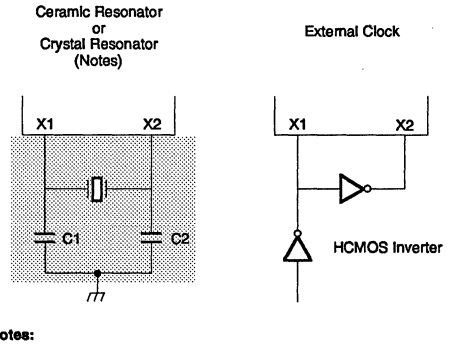
T_A = −10 to 70°C; V_{DD} = +5.0 V ±10%; V_{SS} = AV_{SS} = 0 V; 4 V ≤ AV_{REF} ≤ V_{DD}

Oscillator	Parameter	Symbol	Min	Max	Unit
Ceramic resonator or crystal resonator	Oscillation frequency	f_{XX}	4	12	MHz
External clock	X1 input frequency	f_X	4	12	MHz
	X1 input rise, fall time	t_{XR}, t_{XF}	0	30	ns
	X1 input high-low-level width	t_{WXH}, t_{WXL}	30	130	ns

Recommended Ceramic Resonators (μPD78310/312A)

Manufacturer	Part No.	Frequency (MHz)	External Capacitance (pF) C1	C2
Murata Mfg. Co., Ltd.	CSA12.OMT	12.0	30	30
	CST12.OMT	12.0	Included	Included

Recommended Circuits



Notes:

1. When using a crystal resonator, the following external capacitor is recommended:
   C1 = C2 = 15pF
2. Oscillator circuit must be located as close as possible to the X1 and X2 pins.
3. To prevent noise from affecting operation, avoid locating other signal lines within the shaded area.

83YL-6838A

Timing Dependent on t_{CYK}

Symbol	Formula	Min/Max	Unit
t_{SAL}	1.5T − 100	Min	ns
t_{DAR}	2T − 100		
t_{DAID}	(3.5 + n) T − 170	Max	ns
t_{DLID}	(2 + n) T − 100		
t_{DRID}	(1.5 + n) T − 70		
t_{DLR}	0.5T − 20	Min	ns
t_{DRL}	T − 50		
t_{DRA}	0.5T − 30		
t_{WRL}	(1.5 + n) T − 50		
t_{WLH}	T − 40		
t_{DAW}	2T − 100		
t_{DLOD}	0.5T + 110	Max	ns
t_{DLW}	0.5T − 20 (normal operation)	Min	ns
	T − 50 (during refresh mode)		
t_{SODWR}	(1.5 + n) T − 100		
t_{SODWF}	0.5T − 50		
t_{DWL}	T − 50		
t_{WWL}	(1.5 + n) − 50		

Notes:

(1) n is the number of additional wait cycles specified by the MM register.

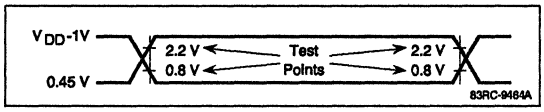
(2) T = t_{CYK} = $1/f_{CLK}$ = $2/f_{XX}$. f_{CLK} is the internal sytem clock frequency.

(3) Any parameter not included in this table is not dependent on f_{CLK}.

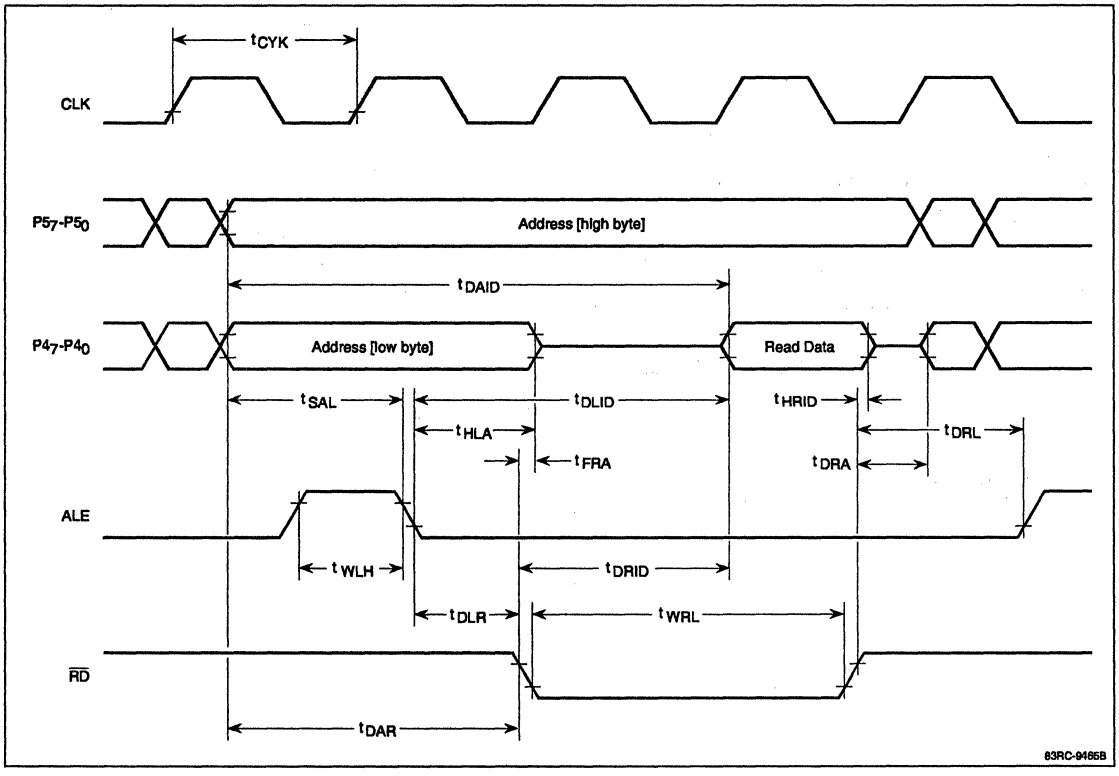
5a

Timing Waveforms

AC Timing Test Points

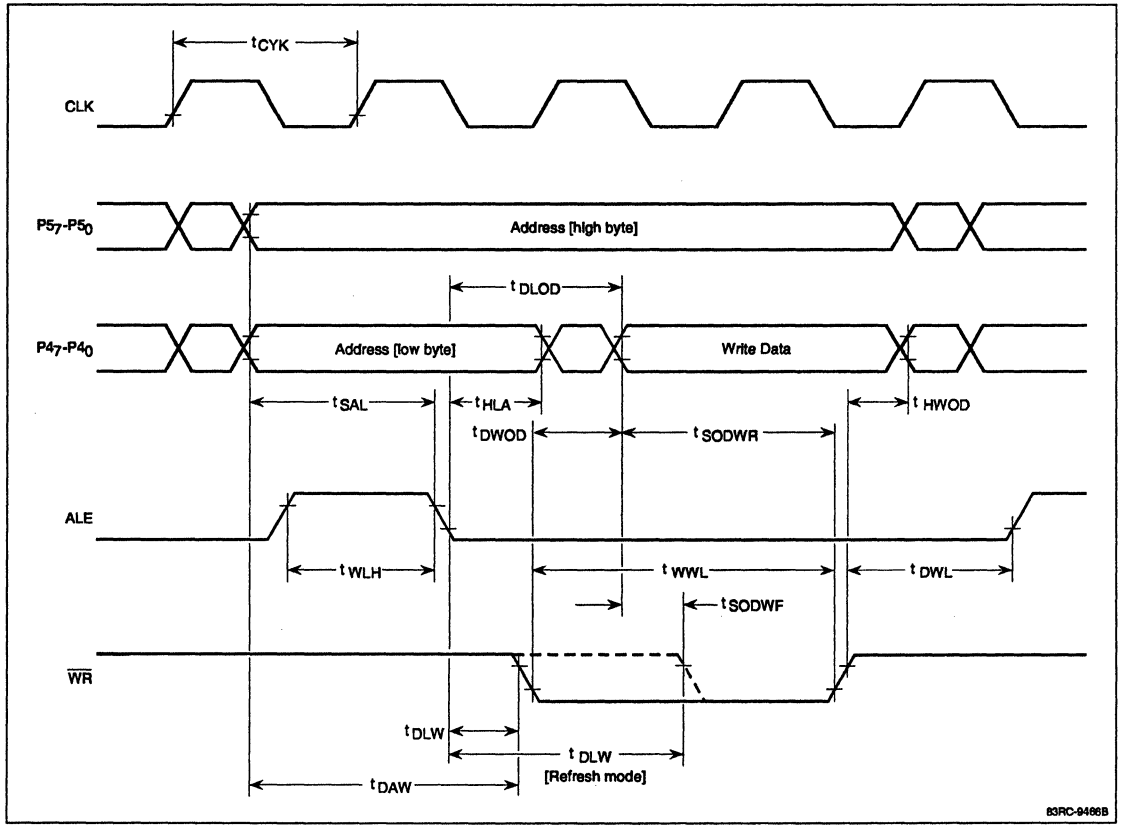


Read Operation



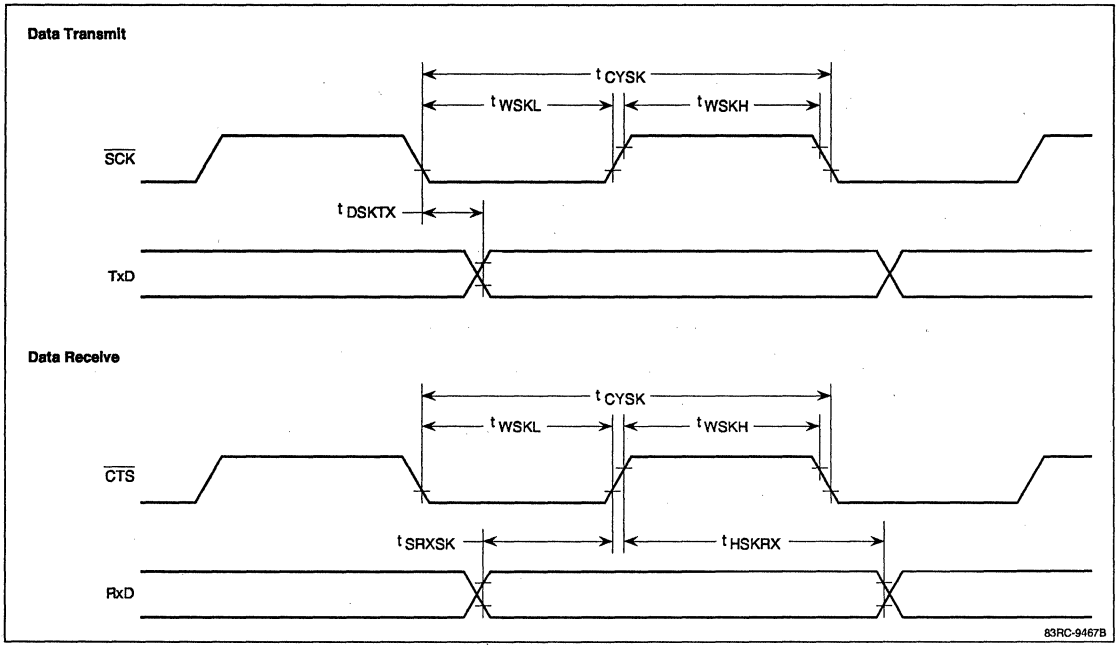
Timing Waveforms (cont)

Write Operation



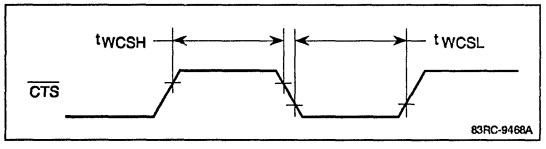
Timing Waveforms (cont)

Serial Port, I/O Interface Mode

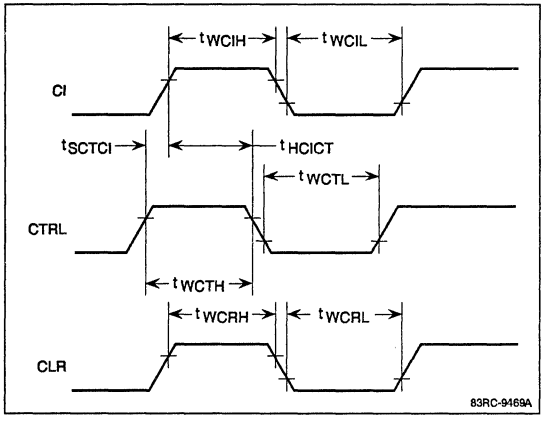


Timing Waveforms (cont)

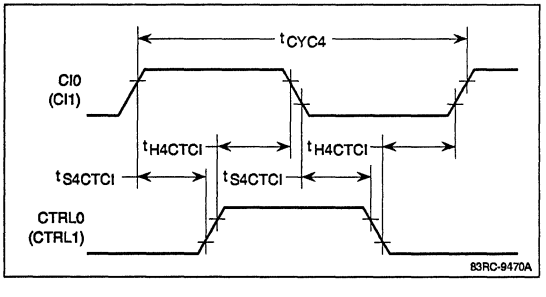
Serial Port, Asynchronous Mode Send Enable Input Timing



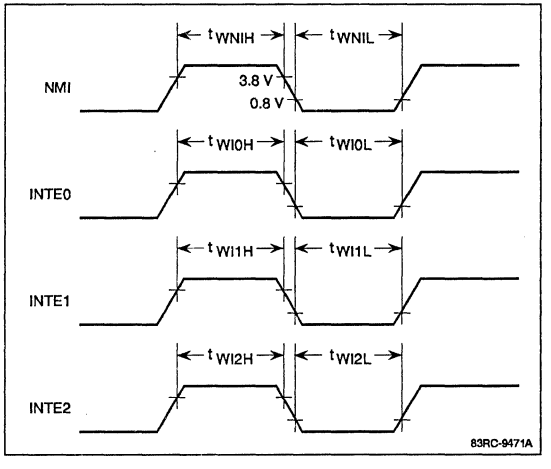
Counter Operation (Mode 3)



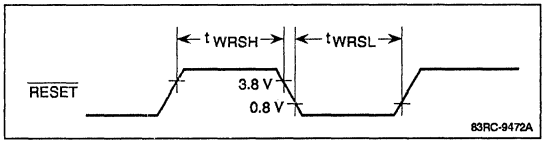
Count Timing Specification (Mode 4)



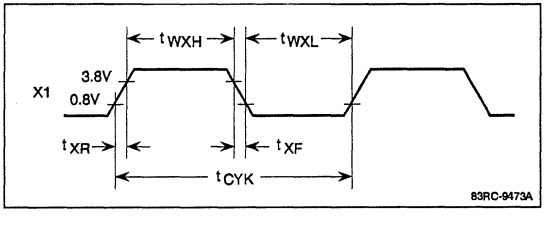
External Interrupts



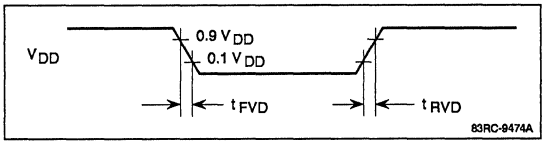
External Reset



External Clock



Data Retention Timing



5a

PROM PROGRAMMING

The PROM in the μPD78P312A is an OTP or UVE EPROM with an 8,192 x 8-bit configuration. The pins listed in the table below are used to program the PROM.

When used in the normal operation mode, 5 V ±10% is applied to the V_{DD} and V_{PP} pins. A voltage higher than V_{DD} should not be applied to other pins.

The programming characteristics of the μPD78P312A are identical to those of the μPD27C256A.

Pin	Function
V_{PP}	High voltage input (write/verify mode), high-level input (read mode)
PROG	High voltage input (write/verify mode, read mode)
A_0 - A_7	Address input (lower 8 bits)
A_8 - A_{12}	Address input (upper 8 bits)
D_0 - D_7	Data input (write mode), data output (verify mode)
$\overline{CE}$	Program pulse input
$\overline{OE}$	Output enable input
V_{DD}	Power supply pin

Notes:

(1) Mask the window of the UVE EPROM version to protect the PROM from being erased accidentally.

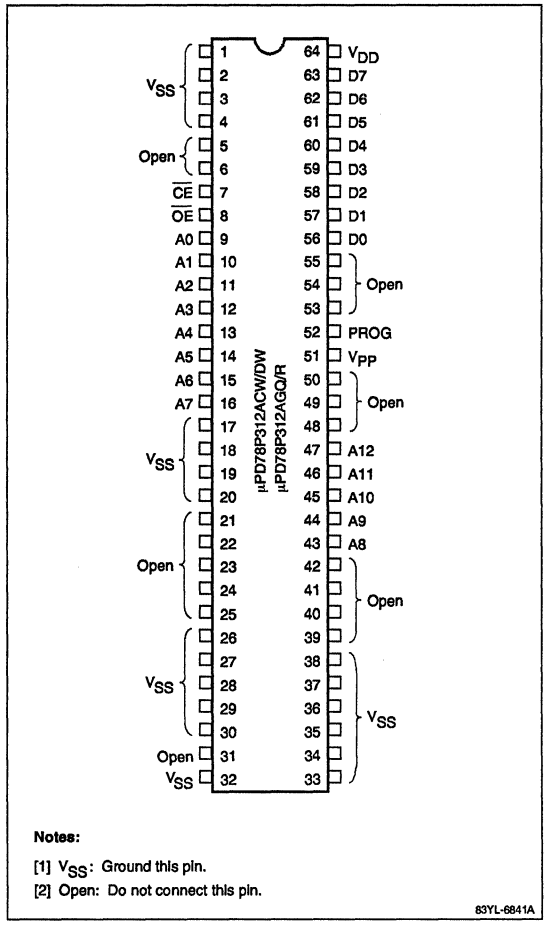
(2) The OTP EPROM version cannot be erased by ultraviolet rays because it does not have a window.

Programming Setup

Programming socket adaptors PA-78P312CW/GF/GQ/L are used to configure the μPD78P312A to fit a standard PROM socket. Set the PROM programmer to program the 27C256A. If the PROM programmer is an older model, check that the programming voltage does not exceed 12.5 volts.

Pin Functions, PROM Programming Mode

64-Pin Shrink DIP and QUIP, Plastic and Ceramic



Notes:

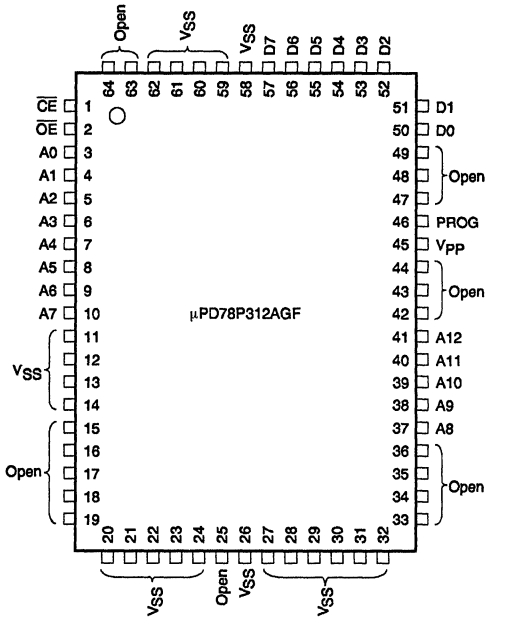
[1] V_{SS}: Ground this pin.

[2] Open: Do not connect this pin.

83YL-6841A

Pin Functions, PROM Programming Mode (cont)

64-Pin Plastic QFP (bent leads)



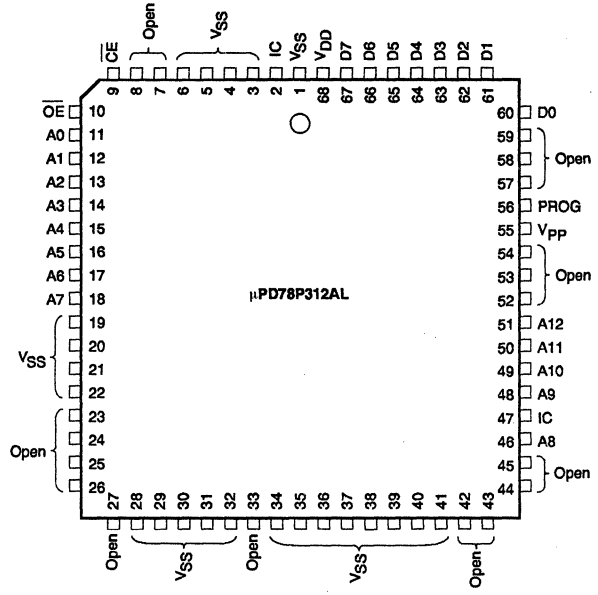
Notes:
V_{SS}: Ground this pin
Open: Do not connect this pin

83YL-6842A

5a

Pin Functions, PROM Programming Mode (cont)

68-Pin PLCC



Notes:

[1] V_{SS}: Ground this pin.

[2] Open: Do not connect this pin.

83YL-6843B

PROM Programming Mode

When +6 V is applied to the V_{DD} pin and +12.5 V is applied to the PROG pin and V_{PP} pin, the μPD78P312A enters the program write/verify mode. Operation in this mode is determined by the setting of $\overline{CE}$ and $\overline{OE}$ pins as indicated in the table below.

Mode	$\overline{CE}$	$\overline{OE}$	V_{PP}	V_{DD}	PROG
Write	L	H	+12.5 V	+6V	+12.5 V
Verify	H	L			
Program inhibit	H	H			
Read (Note 2)	L/H	L	+5 V	+5 V	+12.5 V
Read (Note 3)	L/H	H			

Notes:

(1) When +12.5 V is applied to V_{PP} and +6 V is applied to V_{DD}, both $\overline{CE}$ and $\overline{OE}$ must not be set to the low level (L) simultaneously.

(2) Data is output from the D_0 - D_7 pins.

(3) D_0 - D_7 are high impedance.

Recommended Conditions for Unused Pins

Table 3 describes how to set unused pins when programming the PROM.

Table 3. Recommended Conditions for Unused Pins

Pin	Recommended Connection
$P0_0$ - $P0_3$	Connect to V_{SS}
$P0_4$, $P0_5$	Open
$P2_0$ - $P2_3$	Connect to V_{SS}
$P2_5$ - $P2_7$, $\overline{RFSH}$	Open
$P3_0$ - $P3_3$, X1	Connect to V_{SS}
X2	Open
AN0-AN3, AV_{REF}, AV_{SS}	Connect to V_{SS}
$P3_4$ - $P3_7$, $P5_5$ - $P5_7$, $\overline{RD}$, $\overline{WR}$, ALE	Open

PROM Write Procedure

Data can be written to the PROM by using the following procedure.

(1) Set the pins not used for programming as indicated in table 3, and supply +6 V to the V_{DD} pin, and +12.5 V to the V_{PP} and PROG pins.

(2) Provide the initial address.

(3) Provide write data.

(4) Provide a 1 ms program pulse (active low) to the $\overline{CE}$ pin.

(5) Use the verify mode to test the data. If the data has been written, proceed to (7), if not, repeat steps (3) to (5). If the data cannot be correctly written in 25 attempts, go to step (6).

(6) Classify the PROM as defective and cease write operation.

(7) Provide write data and supply program pulse (for additional writing) for 3 ms times the number of repeats performed between steps (3) to (5).

(8) Increment the address.

(9) Repeat steps (3) to (8) until the last address is reached.

PROM Read Procedure

The contents of the PROM can be read out to the external data bus D_0 - D_7 by using the following procedure.

(1) Set the unused pins as indicated in table 3.

(2) Supply + 5 V to the V_{DD} pin and V_{PP} pin, and +12.5 V to the PROG pin.

(3) Input the address of the data to be read to the A_0 to A_{12} pins.

(4) Put an active low pulse of at least 1 μs on the $\overline{OE}$ pin.

(5) Data is output to the D_0 to D_7 pins.

Erasure

The UVE EPROM can be erased by exposing the window to light having a wavelength shorter than 400 nm, including ultraviolet rays, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15 W s/cm^2 (ultraviolet ray intensity x exposure time) is required to completely erase written data. Erasure by an ultraviolet lamp rated at 12,000 μW/cm^2 takes approximately 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

DC Programming Characteristics

T_A = 25 ±5°C; V_{IP} = 12.0 ±0.5 V; V_{SS} = 0 V

Parameter	Symbol	Symbol (Note)	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH}	V_{IH}	2.2		V_{DDP} + 0.3	V	
Low-level input voltage	V_{IL}	V_{IL}	−0.3		0.8	V	
Input leakage current	V_{LIP}	V_{LI}			10	μA	$0 \le V_I \le V_{DDP}$
High-level output voltage	V_{OH}	V_{OH}	V_{DD}−1			V	I_{OH} = −1.0 mA
Low-level output voltage	V_{OL}	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output leakage current	I_{LO}	—			10	μA	$0 \le V_o \le V_{DDP}$, $\overline{OE}$ = V_{IH}
PROG pin high voltage input current	I_{IP}	—			±10	μA	
V_{DDP} power supply voltage	V_{DDP}	V_{DD}	5.75	6.0	6.25	V	Program memory write mode
			4.5	5.0	5.5	V	Program memory read mode
V_{PP} power supply voltage	V_{PP}	V_{PP}	12.2	12.5	12.8	V	Program memory write mode
				V_{PP} = V_{DDP}		V	Program memory read mode
V_{DDP} power supply current	I_{DD}	I_{DD}		10	30	mA	Program memory write mode
				10	30	mA	Program memory read mode $\overline{CE}$ = V_{IL}, V_I = V_{IH}
V_{PP} power supply current	I_{PP}	I_{PP}		10	30	mA	Program memory write mode $\overline{CE}$ = V_{IL}, $\overline{OE}$ = V_{IH}
				1	100	μA	Program memory read mode

Notes:

(1) Corresponding symbols for the μPD27C256A

AC Programming Characteristics

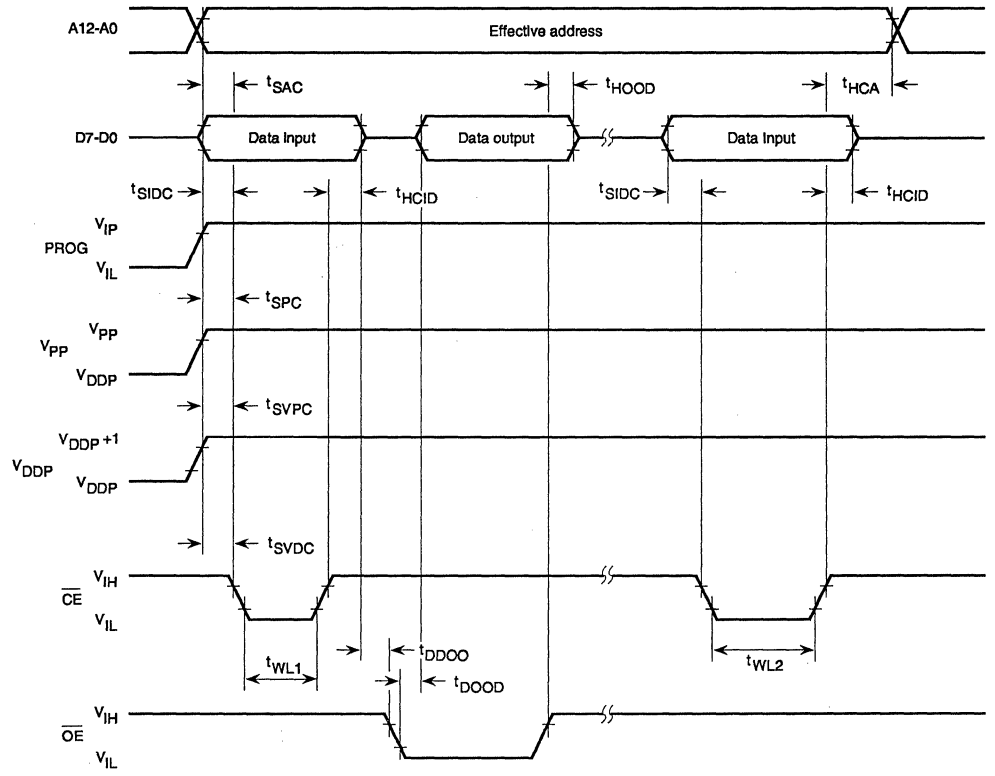
T_A = 25 ±5°C; V_{IP} = 12.0 ±0.5 V; V_{SS} = 0 V

Parameter	Symbol	Symbol (Note)	Min	Typ	Max	Unit	Condition
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}	2			μs	
Data to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}	2			μs	
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}	2			μs	
Address hold time after $\overline{CE}$ ↑	t_{HCA}	t_{AH}	2			μs	
Input data hold time after $\overline{CE}$ ↑	t_{HCID}	t_{DH}	2			μs	
Output data hold time after $\overline{OE}$ ↑	t_{HOOD}	t_{DF}	0		130	ns	
V_{PP} setup time before $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}	2			μs	
V_{DDP} setup time before $\overline{CE}$ ↓	t_{SVDC}	t_{VDS}	2			μs	
Initial program pulse width	t_{WL1}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{WL2}	t_{OPW}	2.85		78.75	ms	
PROG high-voltage input setup time before $\overline{CE}$ ↓	t_{SPC}		2			μs	
Address to data output time	t_{DAOD}	t_{ACC}			2	μs	$\overline{OE}$ = V_{IL}
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			1	μs	
Data hold time after $\overline{OE}$ ↑	t_{HCOD}	t_{DF}	0		130	ns	
Data hold time after address not valid	t_{HAOD}	t_{OH}	0			ns	$\overline{OE}$ = V_{IL}

Notes:

(1) Corresponding symbols for the μPD27C256A

PROM Write Mode Timing



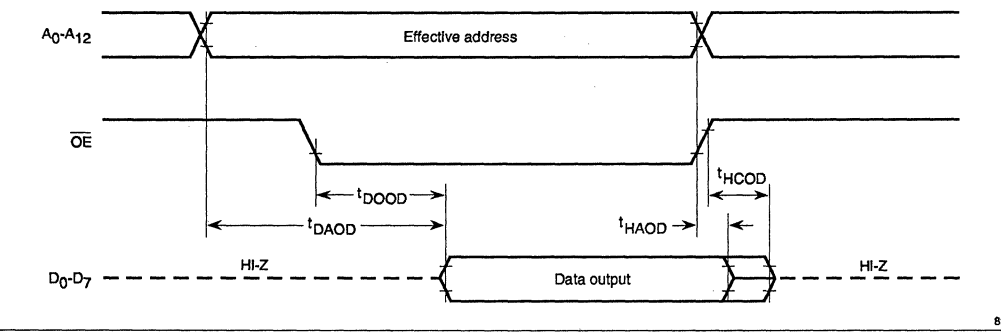
Notes:

[1] V_{DDP} must be applied before V_{PP} is applied and must be removed after V_{PP} is removed.

[2] V_{PP} must not exceed +13 V including overshoot voltage.

83YL-6844B

PROM Read Mode Timing



83YL-6845B

5a

INSTRUCTION SET

The μPD78312A family instruction set features 8- and 16-bit data transfer, arithmetic, and logic instructions and single-bit manipulation instructions. String manipulation instructions are also included. Branch instructions exist to test individual bits in the program status word, the 16-bit accumulator, the special function registers, and the saddr portion of on-chip RAM. Instructions range in length from 1 to 6 bytes depending on the instruction and addressing mode.

Flag Column Indicators

Symbol	Action
(blank)	No change
0	Set to 0
1	Set to 1
X	Set or cleared according to result
P	P/V indicates parity of result
V	P/V indicates arithmetic overflow
R	Restored from saved PSW

Instruction Set Symbols

Symbol	Definition
r	R0, R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14, R15
r1	R0, R1, R2, R3, R4, R5, R6, R7
r2	C, B
rp	RP0, RP1, RP2, RP3, RP4, RP5, RP6, RP7*
rp1	RP0, RP1, RP2, RP3, RP4, RP5, RP6, RP7*
rp2	DE, HL, VP, UP
sfr	Special function register, 8 bits
sfrp	Special function register, 16 bits
post	RP0, RP1, RP2, RP3, RP4, RP5/PSW, RP6, RP7. Bits set to 1 indicate register pairs to be pushed/popped to/from stack; RP5 pushed/popped by PUSH/POP, SP is stack pointer; PSW pushed/popped by PUSHU/POPU, RP5 is stack pointer.
mem	Register indirect: [DE], [HL], [DE+], [HL+], [DE–], [HL–], [VP], [UP] Base Index Mode: [DE+A], [HL+A], [DE+B], [HL+B], [VP+DE], [VP+HL] Base Mode: [DE+byte], [HL+byte], [VP+byte], [UP+byte], [SP+byte] Index Mode: word [A], word [B], word [DE], word [HL]
saddr	FE20-FF1FH: Immediate byte addresses one byte in RAM, or label
saddrp	FE20-FF1FH: Immediate byte (bit 0 = 0) addresses one word in RAM, or label

Instruction Set Symbols (cont)

Symbol	Definition
word	16 bits of immediate data or label
byte	8 bits of immediate data or label
jdisp8	8-bit two's complement displacement (immediate data displacement value –128 to +127)
bit	3 bits of immediate data (bit position in byte), or label
n	3 bits of immediate data
!addr16	16-bit absolute address specified by an immediate address or label
$addr16	Relative branch address or label
addr16	16-bit address
!addr11	11-bit immediate address or label
addr11	0800H-0FFFH: 0800H + (11-bit immediate address), or label
addr5	0040H-007EH: 0040H + 2 X (5-bit immediate address), or label
A	A register (8-bit accumulator)
X	X register
B	B register
C	C register
D	D register
E	E register
H	H register
L	L register
R0-R15	Register 0 to register 15
AX	Register pair AX (16-bit accumulator)
BC	Register pair BC
DE	Register pair DE
HL	Register pair HL
RP0-RP7	Register pair 0 to register pair 7
PC	Program counter
SP	Stack pointer
UP	User stack pointer (RP5)
PSW	Program status word
PSWH	High-order 8 bits of PSW
PSWL	Low-order 8 bits of PSW
CY	Carry flag
AC	Auxiliary carry flag
Z	Zero flag
P/V	Parity/overflow flag
S	Sign flag
TPF	Table position flag

Instruction Set Symbols (cont)

Symbol	Definition
RBS	Register bank select flag
RSS	Register set select flag
IE	Interrupt enable flag
STBC	Standby control register
WDM	Watchdog timer mode register
( )	Contents of the location whose address is within parentheses; (+) and (–) indicate that the address is incremented after or decremented after it is used
(( ))	Contents of the memory location defined by the quantity within the sets of parentheses
xxH	Hexadecimal quantity
X_H, X_L	High-order 8 bits and low-order 8 bits of X

* rp and rp1 describe the same registers but generate different machine code.

Instruction Set

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY	SUB
8-Bit Data Transfer									
MOV	r1, #byte	r1 ← byte	2						
	saddr, #byte	(saddr) ← byte	3						
	sfr, #byte	sfr ← byte	3						
	r, r1	r ← r1	2						
	A, r1	A ← r1	1						
	A, saddr	A ← (saddr)	2						
	saddr, A	(saddr) ← A	2						
	saddr, saddr	(saddr) ← (saddr)	3						
	A, sfr	A ← sfr	2						
	sfr, A	sfr ← A	2						
	A, mem (Note 1)	A ← (mem)	1						
	A, mem	A ← (mem)	2-4						
	mem, A (Note 1)	(mem) ← A	1						
	mem, A	(mem) ← A	2-4						
	A, [saddrp]	A ← ((saddrp))	2						
	[saddrp], A	((saddrp)) ← A	2						
	A, !addr16	A ← (addr16)	4						
	!addr16, A	(addr16) ← A	4						
	PSWL, #byte	PSWL ← byte	3	X	X	X	X	X	X
	PSWH, #byte	PSWH ← byte	3						
	PSWL, A	PSWL ← A	2	X	X	X	X	X	X
	PSWH, A	PSWH ← A	2						
	A, PSWL	A ← PSWL	2						
	A, PSWH	A ← PSWH	2						
XCH	A, r1	A ↔ r1	1						
	r, r1	r ↔ r1	2						
	A, mem	A ↔ (mem)	2-4						
	A, saddr	A ↔ (saddr)	2						
	A, sfr	A ↔ sfr	3						
	A, [saddrp]	A ↔ ((saddrp))	2						
	saddr, saddr	(saddr) ↔ (saddr)	3						
16-Bit Data Transfer									
MOVW	rp1, #word	rp1 ← word	3						
	saddrp, #word	(saddrp) ← word	4						
	sfrp, #word	sfrp ← word	4						
	rp, rp1	rp ← rp1	2						
	AX, saddrp	AX ← (saddrp)	2						
	saddrp, AX	(saddrp) ← AX	2						
	saddrp, saddrp	(saddrp) ← (saddrp)	3						

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY	SUB
16-Bit Data Transfer (cont)									
MOVW (cont)	AX, sfrp	AX ← sfrp	2						
	sfrp, AX	sfrp ← AX	2						
	rp1, !addr16	rp1 ← (addr16)	4						
	!addr16, rp1	(addr16) ← rp1	4						
XCHW	AX, saddrp	AX ↔ (saddrp)	2						
	AX, sfrp	AX ↔ sfrp	3						
	saddrp, saddrp	(saddrp) ↔ (saddrp)	3						
	rp, rp1	rp ↔ rp1	2						
8-Bit Arithmetic									
ADD	A, #byte	A, CY ← A + byte	2	X	X	X	V	X	0
	saddr, #byte	(saddr), CY ← (saddr) + byte	3	X	X	X	V	X	0
	sfr, #byte	sfr, CY ← sfr + byte	4	X	X	X	V	X	0
	r, r1	r, CY ← r + r1	2	X	X	X	V	X	0
	A, saddr	A, CY ← A + (saddr)	2	X	X	X	V	X	0
	A, sfr	A, CY ← A + sfr	3	X	X	X	V	X	0
	saddr, saddr	(saddr), CY ← (saddr) + (saddr)	3	X	X	X	V	X	0
	A, mem	A, CY ← A + (mem)	2-4	X	X	X	V	X	0
	mem, A	(mem), CY ← (mem) + A	2-4	X	X	X	V	X	0
ADDC	A, #byte	A, CY ← A + byte + CY	2	X	X	X	V	X	0
	saddr, #byte	(saddr), CY ← (saddr) + byte + CY	3	X	X	X	V	X	0
	sfr, #byte	sfr, CY ← sfr + byte + CY	4	X	X	X	V	X	0
	r, r1	r, CY ← r + r1 + CY	2	X	X	X	V	X	0
	A, saddr	A, CY ← A + (saddr) + CY	2	X	X	X	V	X	0
	A, sfr	A, CY ← A + sfr + CY	3	X	X	X	V	X	0
	saddr, saddr	(saddr), CY ← (saddr) + (saddr) + CY	3	X	X	X	V	X	0
	A, mem	A, CY ← A + (mem) + CY	2-4	X	X	X	V	X	0
	mem, A	(mem), CY ← (mem) + A + CY	2-4	X	X	X	V	X	0
SUB	A, #byte	A, CY ← A – byte	2	X	X	X	V	X	1
	saddr, #byte	(saddr), CY ← (saddr) – byte	3	X	X	X	V	X	1
	sfr, #byte	sfr, CY ← sfr – byte	4	X	X	X	V	X	1
	r, r1	r, CY ← r – r1	2	X	X	X	V	X	1
	A, saddr	A, CY ← A – (saddr)	2	X	X	X	V	X	1
	A, sfr	A, CY ← A – sfr	3	X	X	X	V	X	1
	saddr, saddr	(saddr), CY ← (saddr) – (saddr)	3	X	X	X	V	X	1
	A, mem	A, CY ← A – (mem)	2-4	X	X	X	V	X	1
	mem, A	(mem), CY ← (mem) – A	2-4	X	X	X	V	X	1

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY	SUB
8-Bit Arithmetic (cont)									
SUBC	A, #byte	A, CY ← A – byte – CY	2	X	X	X	V	X	1
	saddr, #byte	(saddr), CY ← (saddr) – byte – CY	3	X	X	X	V	X	1
	sfr, #byte	sfr, CY ← sfr – byte – CY	4	X	X	X	V	X	1
	r, r1	r, CY ← r – r1 – CY	2	X	X	X	V	X	1
	A, saddr	A, CY ← A – (saddr) – CY	2	X	X	X	V	X	1
	A, sfr	A, CY ← A – sfr – CY	3	X	X	X	V	X	1
	saddr, saddr	(saddr), CY ← (saddr) – (saddr) – CY	3	X	X	X	V	X	1
	A, mem	A, CY ← A – (mem) – CY	2-4	X	X	X	V	X	1
	mem, A	(mem), CY ← (mem) – A – CY	2-4	X	X	X	V	X	1
8-Bit Logic									
AND	A, #byte	A ← A ∧ byte	2	X	X		P		0
	saddr, #byte	(saddr) ← (saddr) ∧ byte	3	X	X		P		0
	sfr, #byte	sfr ← sfr ∧ byte	4	X	X		P		0
	r, r1	r ← r ∧ r1	2	X	X		P		0
	A, saddr	A← A ∧ (saddr)	2	X	X		P		0
	A, sfr	A← A ∧ sfr	3	X	X		P		0
	saddr, saddr	(saddr)← (saddr) ∧ (saddr)	3	X	X		P		0
	A, mem	A ← A ∧ (mem)	2-4	X	X		P		0
	mem, A	(mem) ← (mem) ∧ A	2-4	X	X		P		0
OR	A, #byte	A ← A ∨ byte	2	X	X		P		0
	saddr, #byte	(saddr) ← (saddr) ∨ byte	3	X	X		P		0
	sfr, #byte	sfr ← sfr ∨ byte	4	X	X		P		0
	r, r1	r ← r ∨ r1	2	X	X		P		0
	A, saddr	A ← A ∨ (saddr)	2	X	X		P		0
	A, sfr	A ← A ∨ sfr	3	X	X		P		0
	saddr, saddr	(saddr)← (saddr) ∨ (saddr)	3	X	X		P		0
	A, mem	A ← A ∨ (mem)	2-4	X	X		P		0
	mem, A	(mem) ← (mem) ∨ A	2-4	X	X		P		0
XOR	A, #byte	A ← A ∀ byte	2	X	X		P		0
	saddr, #byte	(saddr) ← (saddr) ∀ byte	3	X	X		P		0
	sfr, #byte	sfr ← sfr ∀ byte	4	X	X		P		0
	r, r1	r ← r ∀ r1	2	X	X		P		0
	A, saddr	A ← A ∀ (saddr)	2	X	X		P		0
	A, sfr	A ← A ∀ sfr	3	X	X		P		0
	saddr, saddr	(saddr)← (saddr) ∀ (saddr)	3	X	X		P		0
	A, mem	A ← A ∀ (mem)	2-4	X	X		P		0
	mem, A	(mem) ← (mem) ∀ A	2-4	X	X		P		0

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY	SUB
8-Bit Logic (cont)									
CMP	A, #byte	A – byte	2	X	X	X	V	X	1
	saddr, #byte	(saddr) – byte	3	X	X	X	V	X	1
	sfr, #byte	sfr – byte	4	X	X	X	V	X	1
	r, r1	r – r1	2	X	X	X	V	X	1
	A, saddr	A – (saddr)	2	X	X	X	V	X	1
	A, sfr	A – sfr	3	X	X	X	V	X	1
	saddr, saddr	(saddr) – (saddr)	3	X	X	X	V	X	1
	A, mem	A – (mem)	2-4	X	X	X	V	X	1
	mem, A	(mem) – A	2-4	X	X	X	V	X	1
16-Bit Arithmetic									
ADDW	AX, #word	AX, CY ← AX + word	3	X	X	X	V	X	0
	saddrp, #word	(saddrp), CY ← (saddrp) + word	4	X	X	X	V	X	0
	sfrp, #word	sfrp, CY ← sfrp + word	5	X	X	X	V	X	0
	rp, rp1	rp, CY ← rp + rp1	2	X	X	X	V	X	0
	AX, saddrp	AX, CY ← AX + (saddrp)	2	X	X	X	V	X	0
	AX, sfrp	AX, CY ← AX + sfrp	3	X	X	X	V	X	0
	saddrp, saddrp	(saddrp), CY ← (saddrp) + (saddrp)	3	X	X	X	V	X	0
SUBW	AX, #word	AX, CY ← AX – word	3	X	X	X	V	X	1
	saddrp, #word	(saddrp), CY ← (saddrp) – word	4	X	X	X	V	X	1
	sfrp, #word	sfrp, CY ← sfrp – word	5	X	X	X	V	X	1
	rp, rp1	rp, CY ← rp – rp1	2	X	X	X	V	X	1
	AX, saddrp	AX, CY ← AX – (saddrp)	2	X	X	X	V	X	1
	AX, sfrp	AX, CY ← AX – sfrp	3	X	X	X	V	X	1
	saddrp, saddrp	(saddrp), CY ← (saddrp) – (saddrp)	3	X	X	X	V	X	1
CMPW	AX, #word	AX – word	3	X	X	X	V	X	1
	saddrp, #word	(saddrp) – word	4	X	X	X	V	X	1
	sfrp, #word	sfrp – word	5	X	X	X	V	X	1
	rp, rp1	rp – rp1	2	X	X	X	V	X	1
	AX, saddrp	AX – (saddrp)	2	X	X	X	V	X	1
	AX, sfrp	AX – sfrp	3	X	X	X	V	X	1
	saddrp, saddrp	(saddrp) – (saddrp)	3	X	X	X	V	X	1
Multiplication/Division									
MULU	r1	AX ← A x r1	2						
DIVUW	r1	AX (quotient), r1 (remainder) ← AX ÷ r1	2						
MULUW	rp1	AX (high-order 16 bits), rp1 (low-order 16 bits) ← AX x rp1	2						
DIVUX	rp1	AXDE (quotient), rp1 (remainder) ← AXDE ÷ rp1	2						

5a

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY	SUB
Increment/Decrement									
INC	r1	r1 ← r1 + 1	1	X	X	X	V		0
	saddr	(saddr) ← (saddr) + 1	2	X	X	X	V		0
DEC	r1	r1 ← r1 − 1	1	X	X	X	V		1
	saddr	(saddr) ← (saddr) − 1	2	X	X	X	V		1
INCW	rp2	rp2 ← rp2 + 1	1						
	saddrp	(saddrp) ← (saddrp) + 1	3						
DECW	rp2	rp2 ← rp2 − 1	1						
	saddrp	(saddrp) ← (saddrp) − 1	3						
Shift/Rotate									
ROR	r1, n	(CY, $r1_7$ ← $r1_0$, $r1_{m-1}$ ← $r1_m$) x n times	2				P	X	0
ROL	r1, n	(CY, $r1_0$ ← $r1_7$, $r1_{m+1}$ ← $r1_m$) x n times	2				P	X	0
RORC	r1, n	(CY ← $r1_0$, $r1_7$ ← CY, $r1_{m-1}$ ← $r1_m$) x n times	2				P	X	0
ROLC	r1, n	(CY ← $r1_7$, $r1_0$ ← CY, $r1_{m+1}$ ← $r1_m$) x n times	2				P	X	0
SHR	r1, n	(CY ← $r1_0$, $r1_7$ ← 0, $r1_{m-1}$ ← $r1_m$) x n times	2	X	X	0	P	X	0
SHL	r1, n	(CY ← $r1_7$, $r1_0$ ← 0, $r1_{m+1}$ ← $r1_m$) x n times	2	X	X	0	P	X	0
SHRW	rp1, n	(CY ← $rp1_0$, $rp1_{15}$ ← 0, $rp1_{m-1}$ ← $rp1_m$) x n times	2	X	X	0	P	X	0
SHLW	rp1, n	(CY ← $rp1_{15}$, $rp1_0$ ← 0, $rp1_{m+1}$ ← $rp1_m$) x n times	2	X	X	0	P	X	0
ROR4	[rp1]	A_{3-0} ← $(rp1)_{3-0}$, $(rp1)_{7-4}$ ← A_{3-0}, $(rp1)_{3-0}$ ← $(rp1)_{7-4}$	2						
ROL4	[rp1]	A_{3-0} ← $(rp1)_{7-4}$, $(rp1)_{3-0}$ ← A_{3-0}, $(rp1)_{7-4}$ ← $(rp1)_{3-0}$	2						
BCD Adjustment									
ADJ4		Decimal adjust accumulator	1	X	X	X	P	X	
Bit Manipulation									
MOV1	CY, saddr.bit	CY ← (saddr.bit)	3					X	
	CY, sfr.bit	CY ← sfr.bit	3					X	
	CY, A.bit	CY ← A.bit	2					X	
	CY, X.bit	CY ← X.bit	2					X	
	CY, PSWH.bit	CY ← PSWH.bit	2					X	
	CY, PSWL.bit	CY ← PSWL.bit	2					X	
	saddr.bit, CY	(saddr.bit) ← CY	3						
	sfr.bit, CY	sfr.bit ← CY	3						
	A.bit, CY	A.bit ← CY	2						
	X.bit, CY	X.bit ← CY	2						
	PSWH.bit, CY	PSWH.bit ← CY	2						
	PSWL.bit, CY	PSWL.bit ← CY	2	X	X	X	X		X

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY	SUB
Bit Manipulation (cont)									
AND1	CY, saddr.bit	CY ← CY ∧ (saddr.bit)	3					X	
	CY, /saddr.bit	CY ← CY ∧ $\overline{(saddr.bit)}$	3					X	
	CY, sfr.bit	CY ← CY ∧ sfr.bit	3					X	
	CY, /sfr.bit	CY ← CY ∧ $\overline{sfr.bit}$	3					X	
	CY, A.bit	CY ← CY ∧ A.bit	2					X	
	CY, /A.bit	CY ← CY ∧ $\overline{A.bit}$	2					X	
	CY, X.bit	CY ← CY ∧ X.bit	2					X	
	CY, /X.bit	CY ← CY ∧ $\overline{X.bit}$	2					X	
	CY, PSWH.bit	CY ← CY ∧ PSWH.bit	2					X	
	CY, /PSWH.bit	CY ← CY ∧ $\overline{PSWH.bit}$	2					X	
	CY, PSWL.bit	CY ← CY ∧ PSWL.bit	2					X	
	CY, /PSWL.bit	CY ← CY ∧ $\overline{PSWL.bit}$	2					X	
OR1	CY, saddr.bit	CY ← CY ∨ (saddr.bit)	3					X	
	CY, /saddr.bit	CY ← CY ∨ $\overline{(saddr.bit)}$	3					X	
	CY, sfr.bit	CY ← CY ∨ sfr.bit	3					X	
	CY, /sfr.bit	CY ← CY ∨ $\overline{sfr.bit}$	3					X	
	CY, A.bit	CY ← CY ∨ A.bit	2					X	
	CY, /A.bit	CY ← CY ∨ $\overline{A.bit}$	2					X	
	CY, X.bit	CY ← CY ∨ X.bit	2					X	
	CY, /X.bit	CY ← CY ∨ $\overline{X.bit}$	2					X	
	CY, PSWH.bit	CY ← CY ∨ PSWH.bit	2					X	
	CY, /PSWH.bit	CY ← CY ∨ $\overline{PSWH.bit}$	2					X	
	CY, PSWL.bit	CY ← CY ∨ PSWL.bit	2					X	
	CY, /PSWL.bit	CY ← CY ∨ $\overline{PSWL.bit}$	2					X	
XOR1	CY, saddr.bit	CY ← CY ⊻ (saddr.bit)	3					X	
	CY, sfr.bit	CY ← CY ⊻ sfr.bit	3					X	
	CY, A.bit	CY ← CY ⊻ A.bit	2					X	
	CY, X.bit	CY ← CY ⊻ X.bit	2					X	
	CY, PSWH.bit	CY ← CY ⊻ PSWH.bit	2					X	
	CY, PSWL.bit	CY ← CY ⊻ PSWL.bit	2					X	
SET1	saddr.bit	(saddr.bit) ← 1	2						
	sfr.bit	sfr.bit ← 1	3						
	A.bit	A.bit ← 1	2						
	X.bit	X.bit ← 1	2						
	PSWH.bit	PSWH.bit ← 1	2						
	PSWL.bit	PSWL.bit ← 1	2	X	X	X	X	X	

5a

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY	SUB
Bit Manipulation (cont)									
CLR1	saddr.bit	(saddr.bit) ← 0	2						
	sfr.bit	sfr.bit ← 0	3						
	A.bit	A.bit ← 0	2						
	X.bit	X.bit ← 0	2						
	PSWH.bit	PSWH.bit ← 0	2						
	PSWL.bit	PSWL.bit ← 0	2	X	X	X	X	X	X
NOT1	saddr.bit	(saddr.bit) ← $\overline{\text{(saddr.bit)}}$	3						
	sfr.bit	sfr.bit ← $\overline{\text{sfr.bit}}$	3						
	A.bit	A.bit ← $\overline{\text{A.bit}}$	2						
	X.bit	X.bit ← $\overline{\text{X.bit}}$	2						
	PSWH.bit	PSWH.bit ← $\overline{\text{PSWH.bit}}$	2						
	PSWL.bit	PSWL.bit ← $\overline{\text{PSWL.bit}}$	2	X	X	X	X	X	X
SET1	CY	CY ← 1	1					1	
CLR1	CY	CY ← 0	1					0	
NOT1	CY	CY ← $\overline{\text{CY}}$	1					X	
Call/Return									
CALL	!addr16	$(SP-1) \leftarrow (PC + 3)_H$, $(SP - 2) \leftarrow (PC + 3)_L$, PC ← addr16, SP ← SP − 2	3						
	rp1	$(SP-1) \leftarrow (PC + 2)_H$, $(SP - 2) \leftarrow (PC + 2)_L$, $PC_H \leftarrow rp1_H$, $PC_L \leftarrow rp1_L$, SP ← SP − 2	2						
	[rp1]	$(SP-1) \leftarrow (PC + 2)_H$, $(SP - 2) \leftarrow (PC + 2)_L$, $PC_H \leftarrow (rp1 + 1)$, $PC_L \leftarrow (rp1)$, SP ← SP − 2	2						
CALLF	!addr11	$(SP-1) \leftarrow (PC + 2)_H$, $(SP - 2) \leftarrow (PC + 2)_L$, $PC_{15-11} \leftarrow 00001$, PC_{10-0} !addr11, SP ← SP − 2	2						
CALLT	[addr5]	$(SP-1) \leftarrow (PC + 1)_H$, $(SP - 2) \leftarrow (PC + 1)_L$, $PC_H \leftarrow$ (TPFx8000H + addr5 + 1), $PC_L \leftarrow$ (TPFx8000H + addr5), SP ← SP − 2	1						
BRK		$(SP-1) \leftarrow$ PSWH, $(SP - 2) \leftarrow$ PSWL, $(SP - 3) \leftarrow (PC + 1)_H$, $(SP - 4) \leftarrow (PC + 1)_L$, $PC_L \leftarrow$ (003EH), $PC_H \leftarrow$ (003FH), SP ← SP − 4, IE ← 0	1						
RET		$PC_L \leftarrow$ (SP), $PC_H \leftarrow$ (SP + 1), SP ← SP + 2	1						
RETI		$PC_L \leftarrow$ (SP), $PC_H \leftarrow$ (SP + 1), PSWL ← (SP + 2), PSWH ← (SP + 3), SP ← SP + 4, E0S ← 0	1	R	R	R	R	R	R
Stack Manipulation									
PUSH	post	{(SP − 1) ← rpp_H, (SP − 2) ← rpp_L, SP ← SP − 2} x n (Note 2)	2						
	PSW	(SP − 1) ← PSWH, (SP − 2) ← PSWL, SP ← SP − 2	1						

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY	SUB
Stack Manipulation (cont)									
PUSHU	post	{(UP – 1) ← rpp_H, (UP – 2) ← rpp_L, UP ← UP – 2} x n (Note 2)	2						
POP	post	{rpp_L ← (SP), rpp_H ← (SP + 1), SP ← SP + 2} x n (Note 2)	2						
	PSW	PSWL ← (SP), PSWH ← (SP + 1), SP ← SP + 2	1	R	R	R	R	R	R
POPU	post	{rpp_L ← (UP), rpp_H ← (UP + 1), UP ← UP + 2} x n (Note 2)	2						
MOVW	SP, #word	SP ← word	4						
	SP, AX	SP ← AX	2						
	AX, SP	AX ← SP	2						
INCW	SP	SP ← SP + 1	2						
DECW	SP	SP ← SP – 1	2						
Unconditional Branch									
BR	!addr16	PC ← !addr16	3						
	rp1	PC_H ← $rp1_H$, PC_L ← $rp1_L$	2						
	[rp1]	PC_H ← $(rp1)_H$, PC_L ← $(rp1)_L$	2						
	$addr16	PC ← addr 16	2						
Conditional Branch									
BC or BL (Note 3)	$addr16	PC ← $addr 16 if CY = 1	2						
BNC or BNL (Note 3)	$addr16	PC ← $addr 16 if CY = 0	2						
BZ or BE (Note 3)	$addr16	PC ← $addr 16 if Z = 1	2						
BNZ or BNE (Note 3)	$addr16	PC ← $addr 16 if Z = 1	2						
BV or BPE (Note 3)	$addr16	PC ← $addr 16 if P/V = 1	2						
BNV or BPO (Note 3)	$addr16	PC ← $addr 16 if P/V = 0	2						
BN	$addr16	PC ← $addr 16 if S = 1	2						
BP	$addr16	PC ← $addr 16 if S = 0	2						
BGT	$addr16	PC ← $addr 16 if (P/V $\forall$ S) $\vee$ Z = 0	3						
BGE	$addr16	PC ← $addr 16 if P/V $\forall$ S = 0	3						
BLT	$addr16	PC ← $addr 16 if P/V $\forall$ S = 1	3						
BLE	$addr16	PC ← $addr 16 if (P/V $\forall$ S) $\vee$ Z = 1	3						
BH	$addr16	PC ← $addr 16 if Z $\vee$ CY = 0	3						
BNH	$addr16	PC ← $addr 16 if Z $\vee$ CY = 1	3						

5a

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY	SUB
Conditional Branch (cont)									
BT	saddr.bit, $addr16	PC ← $addr 16 if (saddr.bit) = 1	3						
	sfr.bit, $addr16	PC ← $addr 16 if sfr.bit = 1	4						
	A.bit, $addr16	PC ← $addr 16 if A.bit = 1	3						
	X.bit, $addr16	PC ← $addr 16 if X.bit = 1	3						
	PSWH.bit, $addr16	PC ← $addr 16 if PSWH.bit = 1	3						
	PSWL.bit, $addr16	PC ← $addr 16 if PSWL.bit = 1	3						
BF	saddr.bit, $addr16	PC ← $addr 16 if (saddr.bit) = 0	4						
	sfr.bit, $addr16	PC ← $addr 16 if sfr.bit = 0	4						
	A.bit, $addr16	PC ← $addr 16 if A.bit = 0	3						
	X.bit, $addr16	PC ← $addr 16 if X.bit = 0	3						
	PSWH.bit, $addr16	PC ← $addr 16 if PSWH.bit = 0	3						
	PSWL.bit, $addr16	PC ← $addr 16 if PSWL.bit = 0	3						
BTCLR	saddr.bit, $addr16	PC ← $addr 16 if (saddr.bit) = 1 then reset (saddr.bit)	4						
	sfr.bit, $addr16	PC ← $addr 16 if sfr.bit = 1 then reset sfr.bit	4						
	A.bit, $addr16	PC ← $addr 16 if A.bit = 1 then reset A.bit	3						
	X.bit, $addr16	PC ← $addr 16 if X.bit = 1 then reset X.bit	3						
	PSWH.bit, $addr16	PC ← $addr 16 if PSWH.bit = 1 then reset PSWH.bit	3						
	PSWL.bit, $addr16	PC ← $addr 16 if PSWL.bit = 1 then reset PSWL.bit	3	X	X	X	X	X	X
BFSET	saddr.bit, $addr16	PC ← $addr 16 if (saddr.bit) = 0 then set (saddr.bit)	4						
	sfr.bit, $addr16	PC ← $addr 16 if sfr.bit = 0 then set sfr.bit	4						
	A.bit, $addr16	PC ← $addr 16 if A.bit = 0 then set A.bit	3						
	X.bit, $addr16	PC ← $addr 16 if X.bit = 0 then set X.bit	3						
	PSWH.bit, $addr16	PC ← $addr 16 if PSWH.bit = 0 then set PSWH.bit	3						
	PSWL.bit, $addr16	PC ← $addr 16 if PSWL.bit = 0 then set PSWL.bit	3	X	X	X	X	X	X
DBNZ	r2, $addr16	r2 ← r2 − 1, then PC ← $addr 16 if (r2) ≠ 0	2						
	saddr, $addr16	(saddr) ← (saddr) − 1, then PC ← $addr 16 if (saddr) ≠ 0	3						
Context Switching									
BRKCS	RBn	RBS_{2-0} ← n, PC_H ↔ R5, PC_L ↔ R4, R7 ← PSWH, R6 ← PSWL, RSS ← 0, IE ← 0	2						
RETCS	!addr16	PC_H ← R5, PC_L ← R4, R5 ← $!addr16_H$, R4 ← $!addr16_L$, PSWH ← R7, PSWL ← R6, EOS ← 0	3	R	R	R	R	R	R

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY	SUB
String Manipulation									
MOVM	[DE+], A	(DE+) ← A, C ← C–1 End if C = 0	2						
	[DE–], A	(DE–) ← A, C ← C–1 End if C = 0	2						
MOVBK	[DE+], [HL+]	(DE+) ← (HL+), C ← C–1 End if C = 0	2						
	[DE–], [HL–]	(DE–) ← (HL–), C ← C–1 End if C = 0	2						
XCHM	[DE+], A	(DE+) ↔ A, C ← C–1 End if C = 0	2						
	[DE–], A	(DE–) ↔ A, C ← C–1 End if C = 0	2						
XCHBK	[DE+], [HL+]	(DE+) ↔ (HL+), C ← C–1 End if C = 0	2						
	[DE–], [HL–]	(DE–) ↔ (HL–), C ← C–1 End if C = 0	2						
CMPME	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X	1
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X	1
CMPBKE	[DE+], [HL+1]	(DE+) – (HL+), C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X	1
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X	1
CMPMNE	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X	1
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X	1
CMPBKNE	[DE+], [HL+]	(DE+) – (HL+), C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X	1
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X	1
CMPMC	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X	1
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X	1
CMPBKC	[DE+], [HL+1]	(DE+) – (HL+), C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X	1
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X	1
CMPMNC	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X	1
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X	1
CMPBKNC	[DE+], [HL+]	(DE+) – (HL+), C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X	1
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X	1
CPU Control									
MOV	STBC, #byte	STBC ← byte	4						
	WDM, #byte	WDM ← byte	4						
SWRS		RSS ← $\overline{RSS}$	1						
SEL	RBn	RBS_{2-0} ← n, RSS ← 0	2						
	RBn, ALT	RBS_{2-0} ← n, RSS ← 1	2						
NOP		No operation	1						
EI		IE ← 1 (Enable interrupt)	1						
DI		IE ← 0 (Disable interrupt)	1						

Notes:

(1) One byte move instruction when [DE], [HL], [DE+], [DE–], [HL+], or [HL–] is specified for mem.

(2) rpp refers to register pairs specified in post byte. "n" is the number of register pairs specified in post byte.

(3) Either of the two mnemonics may be used.

5a

NEC
NEC Electronics Inc.

μPD78322 Family (μPD78320/322/P322) 16/8-Bit, K-Series Microcontrollers With A/D Converter, Real-Time Output Ports

July 1993

Description

The μPD78320, μPD78322, and μPD78P322 are members of the K-Series® of microcontrollers. These 16/8-bit microcontrollers—with a minimum instruction time of 250 ns at 16 MHz—are designed for high-speed, real-time process control. They feature a 16-bit CPU, an 8-bit external data bus, eight banks of main registers, an advanced interrupt handling facility, and a powerful set of memory-mapped on-chip peripherals.

On-board memory includes 640 bytes of RAM and 16K bytes of mask ROM, UV EPROM, or one-time programmable (OTP) ROM. A ROMless version is also available.

The advanced interrupt handling facility has three levels of programmable hardware priority control and three methods of servicing interrupt requests, including vectoring, hardware context switching, and macro service.

The macro service facility reduces the CPU overhead involved in servicing peripheral interrupts by transferring data between the memory-mapped special function registers (SFRs) and memory without the use of time consuming interrupt service routines. In addition, the macro service facility can be used to perform certain CPU functions, such as event counting, math-oriented data alterations, data comparisons, or A/D converter buffering.

The combination of context switching, eight register banks, and the macro service facility makes these devices ideal for applications in the hard-disk drive and tape drive markets.

K-Series is a registered trademark of NEC Electronics Inc.

Features

- Complete single-chip microcontroller
  - 16-bit ALU
  - 640 bytes of RAM
  - 16K bytes of ROM (μPD78322)
- Powerful instruction set
  - 16-bit multiply and divide
  - 1-bit and 8-bit logic instructions
  - String instructions
- Minimum instruction time: 250 ns at 16 MHz
- 3-byte instruction prefetch queue
- Memory expansion
  - 8085 bus-compatible
  - 64K-byte address space
- Large I/O capacity
  - Up to 55 I/O port lines (μPD78322/P322)
  - Up to 37 I/O port lines (μPD78320)
- Memory-mapped on-chip peripherals (special function registers)
- Real-time pulse unit
  - 16/18-bit free-running timer
  - 16-bit timer/event counter
  - Six 16-bit compare registers
  - Four 18-bit capture registers
  - Six external interrupt/capture lines
  - One external event counter/interrupt line
  - Six timer-controlled output lines
- 10-bit, eight-channel A/D converter; on-chip sample and hold amplifier
- Two-channel serial communications interface
  - Asynchronous serial interface (UART)
  - Clock-synchronized interface
  - Full-duplex, three-wire mode
  - NEC serial bus interface (SBI) mode
  - Dedicated baud-rate generator
- Programmable priority interrupt controller (three levels)
- Three methods of interrupt service
  - Vectored interrupts
  - Context switching with hardware save of all general registers
  - Macro service mode with choice of nine different functions
- Watchdog timer with dedicated output
- STOP and HALT standby functions
- 5-volt CMOS technology

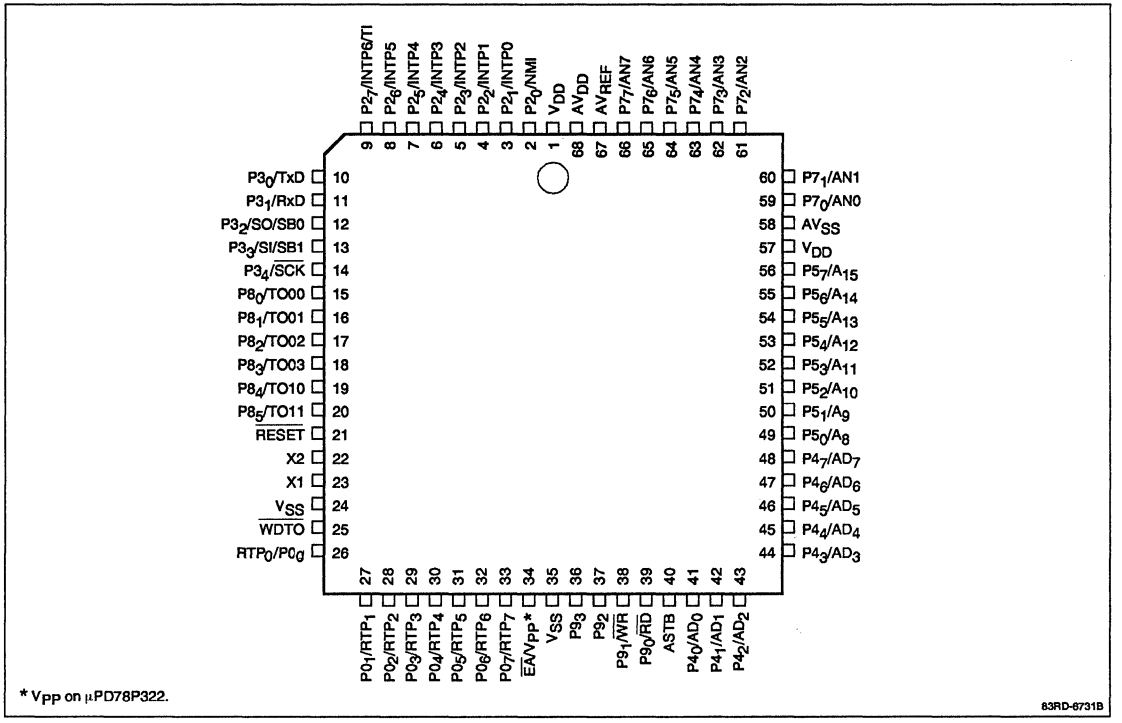
Ordering Information

Part Number	Operating Temperature Range	External Clock	Package	Package Drawing	ROM
μPD78320GF	−10 to +70°C	8 to 16 MHz	80-pin plastic QFP	P80GF-80-3B9-1	ROMless
GF(A)	−40 to +85°C				
GF(A1)	−40 to +110°C	8 to 12 MHz			
GF(A2)	−40 to +125°C				
μPD78320L	−10 to +70°C	8 to 16 MHz	68-pin PLCC	P68L-50A1-1	
L(A)	−40 to +85°C				
L(A1)	−40 to +110°C	8 to 12 MHz			
L(A2)	−40 to +125°C				
μPD78322GF-xxx	−10 to +70°C	8 to 16 MHz	80-pin plastic QFP	P80GF-80-3B9-1	16K mask ROM
GF(A)-xxx	−40 to +85°C				
GF(A1)-xxx	−40 to +110°C	8 to 12 MHz			
GF(A2)-xxx	−40 to +125°C				
μPD78322L-xxx	−10 to +70°C	8 to 16 MHz	68-pin PLCC	P68L-50A1-1	
L(A)-xxx	−40 to +85°C				
L(A1)-xxx	−40 to +110°C	8 to 12 MHz			
L(A2)-xxx	−40 to +125°C				
μPD78P322GF	−10 to +70°C	8 to 16 MHz	80-pin plastic QFP	P80GF-80-3B9-1	16K OTP ROM
L	−10 to +70°C	8 to 16 MHz	68-pin PLCC	P68L-50A1-1	
μPD78P322KE	−10 to +70°C	8 to 16 MHz	80-pin ceramic LCC w/ window	X80KW-80A	16K UV EPROM
KC	−10 to +70°C	8 to 16 MHz	68-pin ceramic LCC w/ window	X68KW-50A	

xxx indicates ROM code suffix

Pin Configurations

68-Pin PLCC or Ceramic LCC



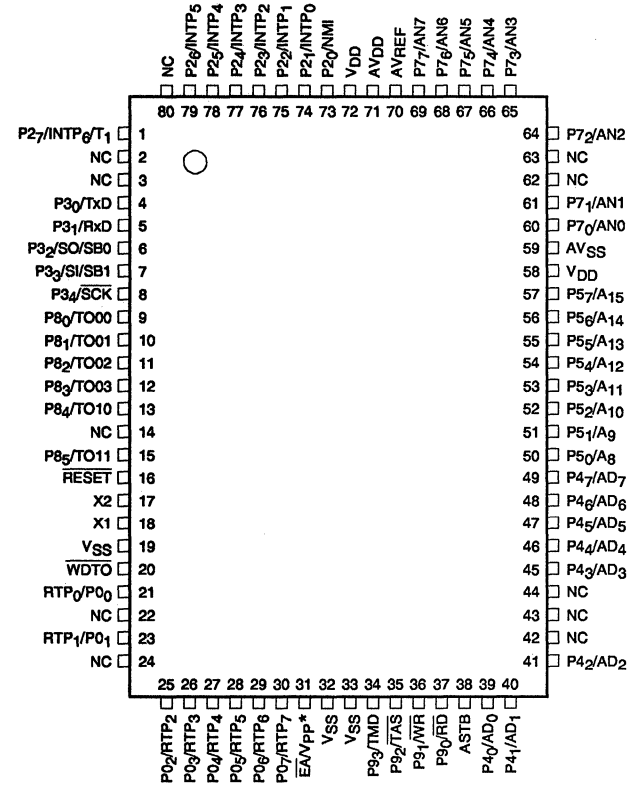
* V_{PP} on μPD78P322.

83RD-6731B

5b

Pin Configurations (cont)

80-Pin Plastic QFP or Ceramic LCC



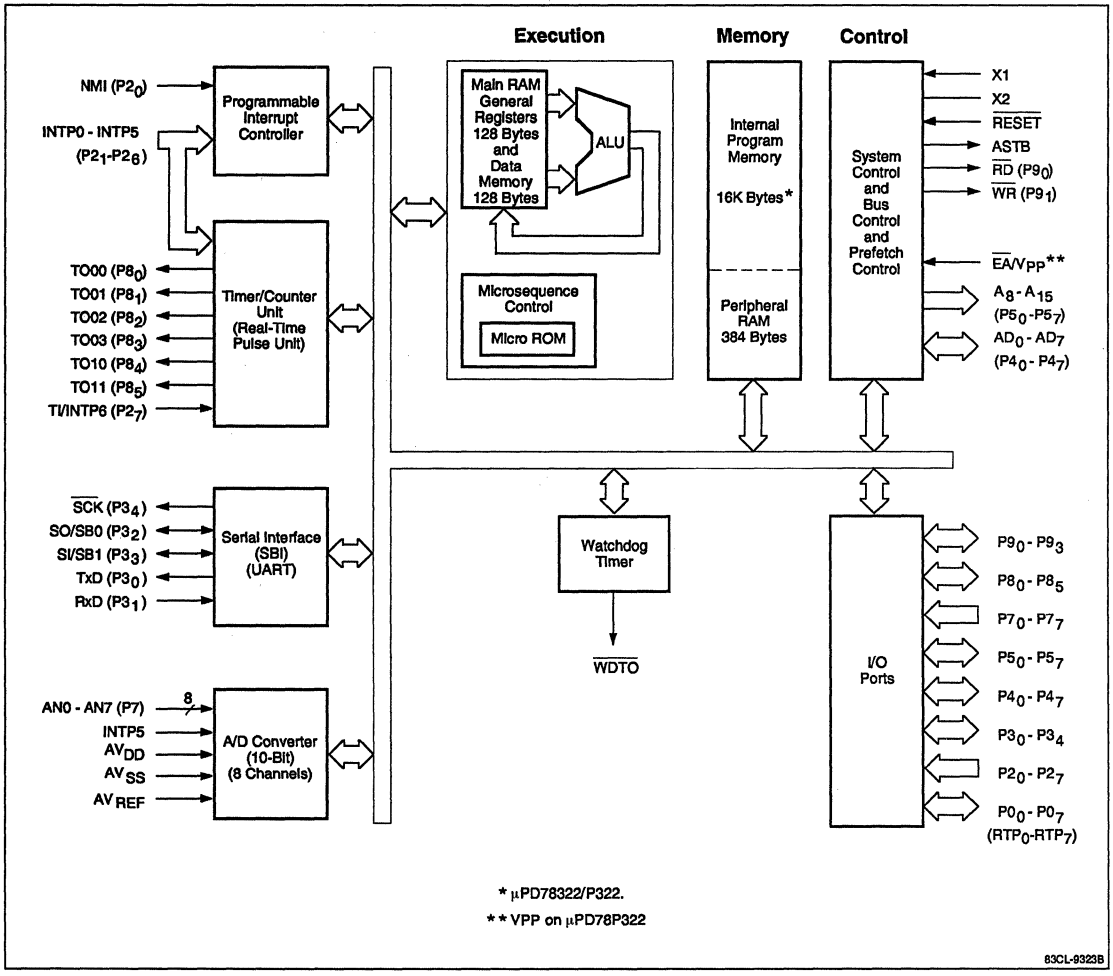
NC No connection; may be connected to V_{SS} to prevent noise.
* V_{PP} on μPD78P322.

83CL-9322B

Pin Functions

Symbol	First Function	Symbol	Second Function
$P0_0$ - $P0_7$	Port 0; 8-bit, bit-selectable I/O port	RTP_0 - RTP_7	Bit-selectable, timer-controlled, real-time output port
$P2_0$	Port 2; 8-bit input port	NMI	External nonmaskable interrupt
$P2_1$ - $P2_6$		INTP0 - INTP5	Maskable external interrupts; edge-selectable
$P2_7$		INTP6/TI	Maskable external interrupt or timer input
$P3_0$	Port 3; 5-bit, bit selectable I/O port	TxD	Asynchronous serial transmit
$P3_1$		RxD	Asysnchronous serial receive
$P3_2$		SO/SB0	SO: serial data output for 3-wire serial I/O mode. SB0: I/O bus for NEC serial bus interface (SBI).
$P3_3$		SI/SB1	SI: serial data input for 3-wire serial I/O mode. SB1: I/O bus for NEC serial bus interface (SBI).
$P3_4$		SCK	Serial clock input or output
$P4_0$ - $P4_7$	Port 4; 8-bit, byte-selectable I/O port	AD_0 - AD_7	Low-order byte of external address/data bus
$P5_0$ - $P5_7$	Port 5; 8-bit, bit-selectable I/O port	A_8 - A_{15}	High-order byte of external address bus
$P7_0$ - $P7_7$	Port 7; 8-bit input port	AN0 - AN7	Inputs for A/D converter
$P8_0$	Port 8; 6-bit, bit-selectable I/O port	TO00	Timer (RPU) output lines
$P8_1$		TO01	
$P8_2$		TO02	
$P8_3$		TO03	
$P8_4$		TO10	
$P8_5$		TO11	
$P9_0$	Port 9; 4-bit, bit-selectable I/O port	$\overline{RD}$	External read strobe
$P9_1$		$\overline{WR}$	External write strobe
$P9_2$ - $P9_3$			
ASTB	External address latch strobe		
$\overline{EA}$	External access control on μPD78320/322; a high level enables access to on-chip ROM; a low level is applied if all program memory is external. Must be tied low for the μPD78320.		
$\overline{RESET}$	External system reset input		
$\overline{WDTO}$	Watchdog timer output		
X1	Crystal connection or external clock input.		
X2	Crystal connection; not necessary to connect with external clock input.		
AV_{REF}	A/D converter reference voltage input		
AV_{DD}	A/D converter +5-volt power input		
AV_{SS}	A/D converter ground		
V_{DD}	+5-volt power input		
V_{SS}	Ground		
V_{PP}	PROM write-verify power input on μPD78P322 only. Must be tied to V_{DD} for normal operation		
NC	Not connected internally. May be connected to V_{SS}		

μPD78322 Family Block Diagram



FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processing unit (CPU) of the μPD78322 family features 16-bit arithmetic including 16-by-16-bit multiply, both unsigned and signed, and 32-by-16-bit unsigned divide (producing a 32-bit quotient and a 16-bit remainder). The signed multiply executes in 3.5 μs and the divide in 5.38 μs at 16 MHz.

A CALLT vector table and a CALLF area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction can access up to 32 subroutines through their addresses in the CALLT vector table. A 2-byte call instruction can access any routine beginning in a specific CALLF area.

The internal system clock (f_{CLK}) is generated by dividing the oscillator frequency by 2. Therefore, at the maximum oscillator frequency of 16 MHz, the clock is 8 MHz. Since some instructions execute in two cycles, the minimum instruction time is 250 ns.

On-Chip RAM

The μPD78322 family has a total of 640 bytes of on-chip RAM. The upper 256-byte area (FE00H-FEFFH) features high-speed access of one word of data per internal system clock and is known as "main RAM." The remainder (FC80H-FDFFH) is accessed at the same speed as external memory (1 byte per three internal system clocks) and is known as "peripheral RAM." The general register banks and the macro service control words are stored in main RAM. The remainder of main RAM and any unused register bank locations are available for general storage.

On-Chip PROM

The μPD78322 contains 16K bytes of internal ROM; the μPD78P322 contains 16K bytes of UV EPROM or one-time programmable ROM. Instructions are fetched from this on-chip memory at a maximum rate of 1 byte every internal system clock through the high-speed fetch mode. The μPD78320 does not have on-chip PROM.

External Memory

The μPD78322 family has a 64K-byte address space. The μPD78322/P322 can access 0, 256, 4K, 16K, or 48K bytes of external memory in the area from 4000H to FFFFH. External memory can be either ROM or RAM (or both) as required. The μPD78322/P322 have an 8-bit wide external data bus and a 16-bit wide external address bus. The low-order 8 bits of the address bus are multiplexed to provide the 8-bit data bus and are supplied by I/O port 4.

High-order address bits are taken from port 5 as required. Address latch, read, and write strobes are also provided. The memory mode register (MM) controls the size of the external memory. It can be programmed for 0, 4, 6, or 8 bits from port 5 for the high-order address. Any remaining port 5 bits can be used for I/O.

The μPD78320 does not have ports 4 and 5. It has eight dedicated high-order address lines and eight dedicated address/data lines. All memory below address FC80H must be external.

The programmable wait control register (PWC) allows the programmer to specify one or two additional wait states if they are required for low-speed memory or external peripheral devices. These wait states for internal and external memory are specified independently.

Program Fetch

The μPD78322 family allows opcode fetch in the area between 0000H and FFFFH under the following constraints: from FC80H to FDFFH, opcodes will be fetched from the peripheral RAM; from FFE0H to FFFFH, opcodes will be fetched from external memory only. The μPD78322 family contains a 3-byte instruction prefetch queue. The bus control unit can fetch an instruction byte from memory during cycles in which the execution unit is not using the memory bus.

Instruction bytes can be fetched from on-chip memory in either high-speed or ordinary fetch cycle mode. The fetch cycle control register (FCC) is used to select the mode. In high-speed fetch cycle mode, one internal system clock is required to fetch each instruction byte from on-chip memory. In ordinary fetch cycle mode, each byte to be fetched requires three, four, or five internal system clocks depending on the setting of the PWC register.

Each instruction byte fetched from external memory requires three, four, or five internal system clocks depending on the setting of the PWC register.

CPU Control Registers

Program Counter. The program counter is a 16-bit register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 0001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

CPU Control Word. The CPU control word (CCW) selects the origin of the interrupt vector and CALLT tables. If the TPF bit (bit 1) is zero, the origin is 0000H; if the TPF bit is one, the origin is 8000H. The CCW is a special function register located at address FFC1H. The addresses of the vectors for the RESET input, operation-code trap, and BRK instruction are fixed at 0000H, 0003CH, and 003EH, respectively, and are not altered by the TPF bit.

Program Status Word. The program status word (PSW) is a 16-bit register containing flags that are set or reset depending on the results of an instruction. This register can be written to or read from 8 bits at a time. The high-order 8 bits are called the PSWH and the low-order 8 bits are called the PSWL. The individual flags can also be manipulated on a bit-by-bit basis. The assignment of PSW bits follows.

	7	6	5	4	3	2	1	0
PSWH	UF	RBS2	RBS1	RBS0	0	0	0	0
	7	6	5	4	3	2	1	0
PSWL	S	Z	RSS	AC	IE	P/V	LT	CY

UF	User flag
RBS2-RBS0	Active register bank number
S	Sign flag (1 if last result was negative)
Z	Zero flag (1 if last result was zero)
RSS	Register set selection flag
AC	Auxiliary carry flag (carry out of 3 bit)
IE	Interrupt enable flag
P/V	Parity or arithmetic overflow flag
LT	Interrupt priority level transition flag
CY	Carry bit (or 1-bit accumulator for logic)

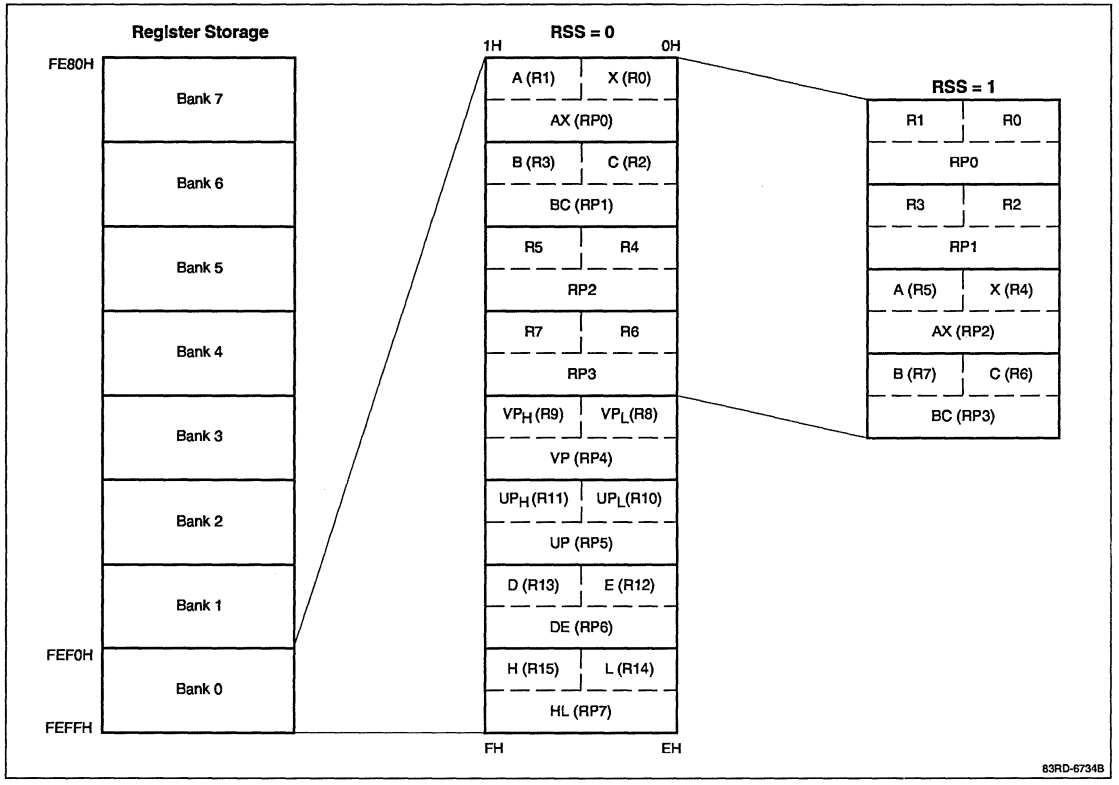
General Registers

There are sixteen 8-bit general registers, which can also be paired to function as 16-bit registers. A complete set of 16 registers is mapped into each of eight program-selectable register banks stored in main RAM. Three bits in the PSW identify active register banks.

Registers have functional names (like A, X, B, C for 8-bit registers and AX, BC for 16-bit registers) and absolute names (like R1, R0, R3, R2 for 8-bit registers and RP0, RP1 for 16-bit registers). Each instruction determines whether a register is referred to by functional or absolute name and whether it is 8 or 16 bits.

Two possible relationships may exist between the absolute and functional names of the first four register pairs. The RSS bit in the PSW determines which of these is active at any time. The effect is that the accumulator and counter registers can be saved, and a new set can be specified by toggling the RSS bit. Figure 1 illustrates the general register configuration.

Figure 1. General Registers



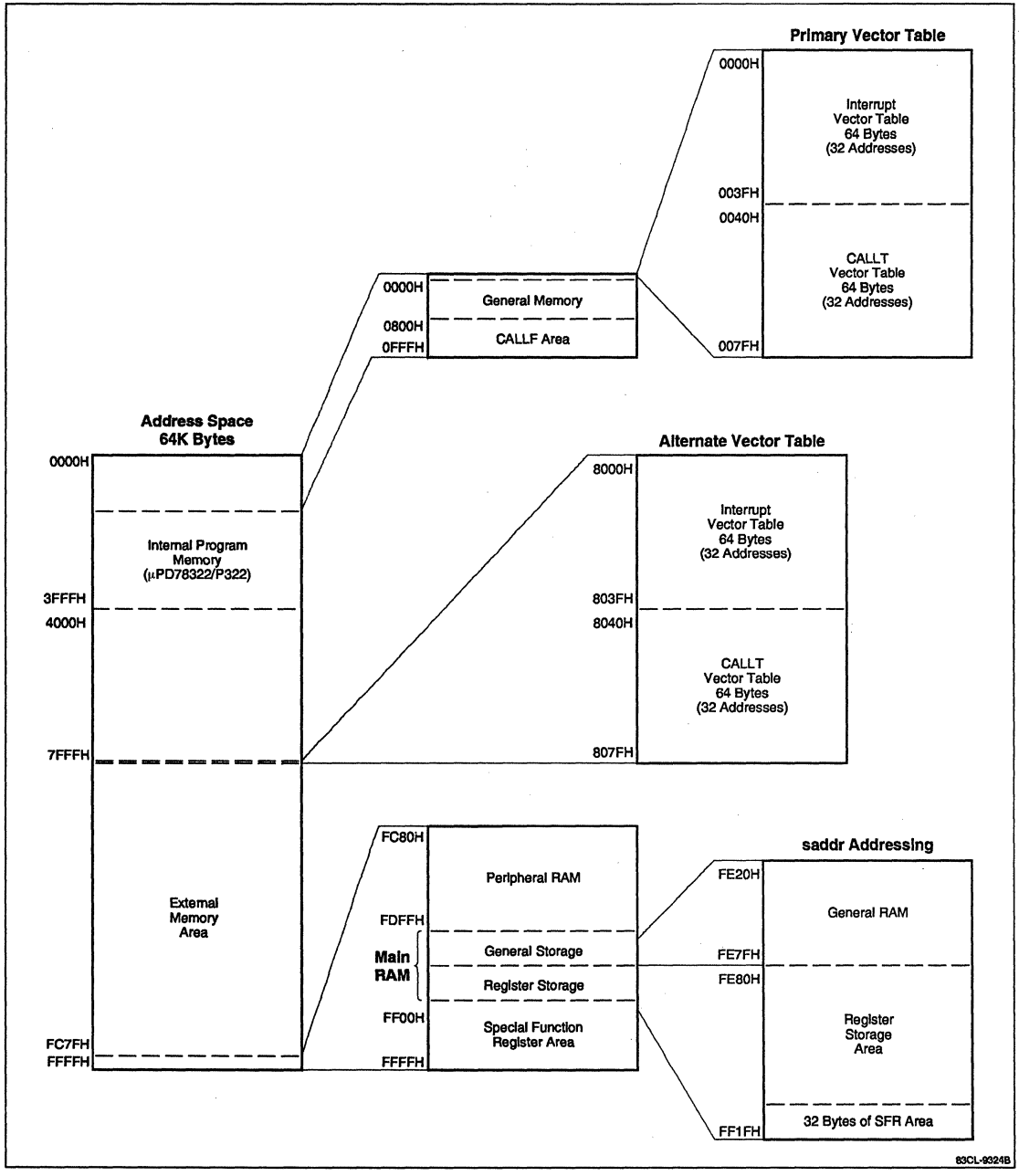
Addressing

The μPD78322 family features 1-byte addressing of both the special function registers and the portion of on-chip RAM from FE20H to FEFFH. The 1-byte sfr addressing accesses the entire SFR area, whereas the 1-byte saddr addressing accesses the first 32 bytes of the SFR area and 224 bytes of the main RAM.

The 16-bit SFRs and words of memory in these areas can be addressed by 1-byte saddrp addressing, which is valid for even addresses only. Since many instructions use 1-byte addressing, access to these locations is almost as fast and as versatile as access to the general registers.

There are nine addressing modes for data in main memory: direct, register, register indirect with autoincrement or decrement, saddr, saddr indirect, SFR, based, indexed, and based indexed. There are also 8-bit and 16-bit immediate operands. Figure 3 is the memory map of the μPD78322 family.

Figure 2. Memory Map



Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and the CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by 1-byte SFR addressing. All except the port mode registers and the asynchronous serial transmission shift register can be read under program control, and most can also be written. They are either 8 or 16 bits, as required, and many of the 8-bit registers are capable of single-bit access as well.

Locations FFD0H through FFDFH are known as the external access area. Registers in external circuitry, interfaced and mapped to these addresses, can be addressed with SFR addressing. Table 1 lists the special function registers.

Table 1. Special Function Registers

				Access Unit (Bits)			
Address	**Register**	**Symbol**	**R/W**	**1**	**8**	**16**	**State After Reset**
FF00H	Port 0	P0	R/W	X	X	—	Undefined
FF02H	Port 2	P2	R	—	X	—	Undefined
FF03H	Port 3	P3	R/W	X	X	—	Undefined
FF04H	Port 4	P4	R/W	X	X	—	Undefined
FF05H	Port 5	P5	R/W	X	X	—	Undefined
FF07H	Port 7	P7	R	—	X	—	Undefined
FF08H	Port 8	P8	R/W	X	X	—	Undefined
FF09H	Port 9	P9	R/W	X	X	—	Undefined
FF0AH-FF0BH	Free-running counter (lower 16 bits)*	TM0LW	R	—	—	X	0000H
FF10H-FF11H	Capture register X0 (lower 16 bits)*	CTX0LW	R	—	—	X	Undefined
FF12H-FF13H	Capture register 01 (lower 16 bits)*	CT01LW	R	—	—	X	Undefined
FF14H-FF15H	Capture register 02 (lower 16 bits)*	CT02LW	R	—	—	X	Undefined
FF16H-FF17H	Capture register 03 (lower 16 bits)*	CT03LW	R	—	—	X	Undefined
FF18H-FF19H	Capture/compare register X0 (lower 16 bits)*	CCX0LW	R/W	—	—	X	Undefined
FF1AH-FF1BH	Capture/compare register 01 (lower 16 bits)*	CC01LW	R/W	—	—	X	Undefined
FF20H	Port 0 mode register	PM0	W	—	X	—	FFH
FF23H	Port 3 mode register	PM3	W	—	X	—	xxx1 1111B
FF25H	Port 5 mode register	PM5	W	—	X	—	FFH
FF28H	Port 8 mode register	PM8	W	—	X	—	xx11 1111B
FF29H	Port 9 mode register	PM9	W	—	X	—	xxxx 1111B
FF2AH-FF2BH	Free running counter (high 16 bits)*	TM0UW	R	—	—	X	0000H
FF2CH-FF2DH	Timer register 1	TM1	R	—	—	X	0000H
FF30H-FF31H	Capture register X0 (high 16 bits)*	CTX0UW	R	—	—	X	Undefined
FF32H-FF33H	Capture register 01 (high 16 bits)*	CT01UW	R	—	—	X	Undefined
FF34H-FF35H	Capture register 02 (high 16 bits)*	CT02UW	R	—	—	X	Undefined
FF36H-FF37H	Capture register 03 (high 16 bits)*	CT03UW	R	—	—	X	Undefined
FF38H-FF39H	Capture/compare register X0 (high 16 bits)*	CCX0UW	R/W	—	—	X	Undefined
FF3AH-FF3BH	Capture/compare register 01 (high 16 bits)*	CC01UW	R/W	—	—	X	Undefined
FF40H	Port 0 mode control register	PMC0	W	—	X	—	00H
FF41H	Real-time output port set register	RTPS	R/W	X	X	—	00H

5b

Table 1. Special Function Registers (cont)

Address	Register	Symbol	R/W	Access Unit (Bits) 1	8	16	State After Reset
FF43H	Port 3 mode control register	PMC3	W	—	X	—	xxx0 0000B
FF48H	Port 8 mode control register	PMC8	W	—	X	—	xx00 0000B
FF4CH-FF4DH	Baud rate generator	BRG	R/W	—	—	X	Undefined
FF60H	Real-time output port register	RTP	R/W	X	X	—	Undefined
FF61H	Real-time output port reset register	RTPR	R/W	X	X	—	00H
FF62H	Port read control register	PRDC	R/W	X	X	—	00H
FF68H	A/D converter mode register	ADM	R/W	X	X	—	00H
FF6AH	A/D converter result register (16-bit access)	ADCR	R	—	—	X	Undefined
FF6BH	A/D converter result register (high 8 bits)	ADCRH	R	—	X	—	Undefined
FF70H-FF71H	Compare register 00	CM00	R/W	—	—	X	Undefined
FF72H-FF73H	Compare register 01	CM01	R/W	—	—	X	Undefined
FF74H-FF75H	Compare register 02	CM02	R/W	—	—	X	Undefined
FF76H-FF77H	Compare register 03	CM03	R/W	—	—	X	Undefined
FF7CH-FF7DH	Compare register 10	CM10	R/W	—	—	X	Undefined
FF7EH-FF7FH	Compare register 11	CM11	R/W	—	—	X	Undefined
FF80H	Clock synchronized serial interface mode register	CSIM	R/W	X	X	—	00H
FF82H	Serial bus interface control register	SBIC	R/W	X	X	—	00H
FF86H	Serial I/O shift register	SIO	R/W	X	X	—	Undefined
FF88H	Asynchronous serial interface mode register	ASIM	R/W	X	X	—	80H
FF8AH	Asynchronous serial interface status register	ASIS	R	—	X	—	00H
FF8CH	Serial receive buffer: UART	RXB	R	—	X	—	Undefined
FF8EH	Serial transmit shift register: UART	TXS	W	—	X	—	Undefined
FFB0H	Timer control register	TMC	R/W	X	X	—	00H
FFB1H	Baud rate generator mode register	BRGM	R/W	X	X	—	00H
FFB2H	Prescalar mode register	PRM	R/W	X	X	—	00H
FFB8H	Timer output control register 0	TOC0	R/W	X	X	—	00H
FFB9H	Timer output control register 1	TOC1	R/W	X	X	—	00H
FFBFH	Real-time pulse unit mode register	RPUM	R/W	X	X	—	00H
FFC0H	Standby control register	STBC	R/W**	X	X	—	0000 X000B
FFC1H	CPU control word	CCW	R/W	X	X	—	00H
FFC2H	Watchdog timer mode register	WDM	R/W**	X	X	—	00H
FFC4H	Memory extension mode register	MM	R/W	X	X	—	00H
FFC6H	Programmable wait control register	PWC	R/W	X	X	—	22H
FFC9H	Fetch cycle control register	FCC	R/W	X	X	—	00H
FFD0H-FFDFH	External access area		R/W	X	X	—	Undefined
FFE0H	Interrupt request flag register 0L	IF0L/IF0	R/W	X	X	X	00H
FFE1H	Interrupt request flag register 0H	IF0H	R/W	X	X	—	00H
FFE2H	Interrupt request flag register 1L	IF1L/IF1	R/W	X	X	X	00H

Table 1. Special Function Registers (cont)

Address	Register	Symbol	R/W	Access Unit (Bits) 1	8	16	State After Reset
FFE4H	Interrupt mask flag register 0L	MK0L/ MK0	R/W	X	X	X	FFH
FFE5H	Interrupt mask flag register 0H	MK0H	R/W	X	X	—	FFH
FFE6H	Interrupt mask flag register 1L	MK1L/ MK1	R/W	X	X	X	xxxx x111B
FFE8H	Priority selection buffer register 0L	PB0L/ PB0	R/W	X	X	X	00H
FFE9H	Priority selection buffer register 0H	PB0H	R/W	X	X	—	00H
FFEAH	Priority selection buffer register 1L	PB1L/ PB1	R/W	X	X	X	00H
FFECH	Interrupt service mode selection register 0L	ISM0L/ ISM0	R/W	X	X	X	00H
FFEDH	Interrupt service mode selection register 0H	ISM0H	R/W	X	X	—	00H
FFEEH	Interrupt service mode selection register 1L	ISM1L/ ISM1	R/W	X	X	X	00H
FFF0H	Context switch enable register 0L	CSE0L/ CSE0	R/W	X	X	X	00H
FFF1H	Context switch enable register 0H	CSE0H	R/W	X	X	—	00H
FFF2H	Context switch enable register 1L	CSE1L/ CSE1	R/W	X	X	X	00H
FFF4H	External interrupt mode register 0	INTM0	R/W	X	X	—	00H
FFF5H	External interrupt mode register 1	INTM1	R/W	X	X	—	00H
FFF8H	In-service priority register	ISPR	R	—	X	—	00H
FFF9H	Priority selection register	PRSL	R/W	X	X	—	00H

* Lower or upper 16 bits of an 18-bit register.
** Protected location: special instruction required for write.

5b

Input/Output Ports

The μPD78322 family has six ports providing a total of 37 I/O lines. P0, P3, P8, and P9 are tri-state input/output ports of 8, 5, 6, and 2 bits, respectively; each bit can be individually selected for input or output. P2 and P7 are 8-bit input ports.

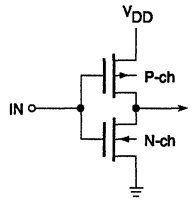
P2 functions only in the control mode as input pins for the NMI signal, the INTP0 to INTP5 interrupt signals, and the INTP6/TI interrupt signal or external count clock for timer 1 (TI). However, any masked interrupt automatically becomes an input line and the state of all the pins can be read by the program using a read instruction to port 2. Each pin of P2 can be programmed for rising, falling, or both rising and falling edge detection.

The output level of the P0, P3, P8, and P9 I/O pins can be tested to determine whether they agree with the contents of the output latch. When the low-order bit of port read control register PRDC is set to 1, the output level of the I/O pins can be read with the port still in the output mode. These data values can be compared with the data known to be in the output latch to determine if the port is functioning correctly. Figure 3 shows the structure of each port pin.

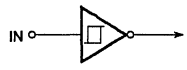
The μPD78322/P322 have two additional input/output ports, P4 and P5, and two additional I/O pins in P9. All these I/O lines are available if external memory or memory-mapped external circuitry is not being used. Port 4 is shared with the low-order address/data bus (AD_0 to AD_7) and is byte-selectable for input or output. Port 5 is shared with the high-order address bus (A_8 to A_{15}). Depending on the amount of external memory used, either 8, 6, 4, or 0 bits are available for bit-selectable I/O. Port 9 is a 4-bit, bit-selectable I/O port; two of its pins are shared with the read and write strobes.

Figure 3. I/O Circuits

Type 1. $\overline{EA}$

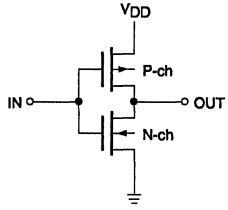


Type 2. P2, $\overline{RESET}$

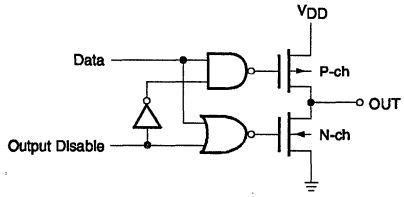


Schmitt trigger input with hysteresis characteristics.

Type 3. $\overline{WDTO}$

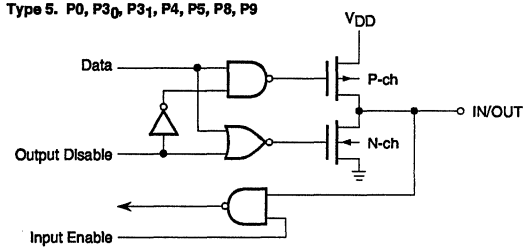


Type 4. ASTB

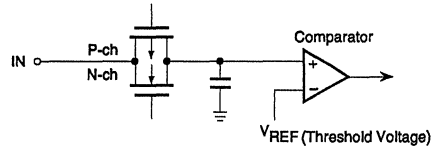


Push-pull output that can be placed in high impedance (both P-ch and N-ch off).

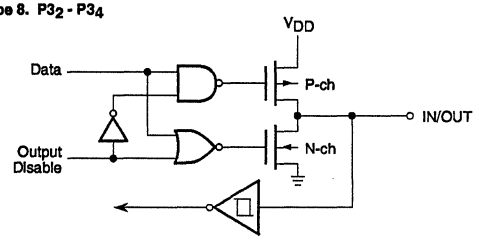
Type 5. P0, $P3_0$, $P3_1$, P4, P5, P8, P9



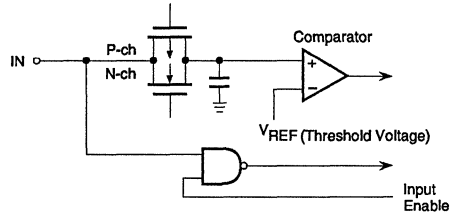
Type 7.



Type 8. $P3_2$ - $P3_4$



Type 9. P7



83CL-9325B

Real-Time Output Port

Port 0 can function on a bit-selectable basis as a real-time output port. Selected bits can be directly written under program control, or they can be set or cleared under control of timing signals generated by the real-time pulse unit. Timing by the latter method is independent of interrupt latency.

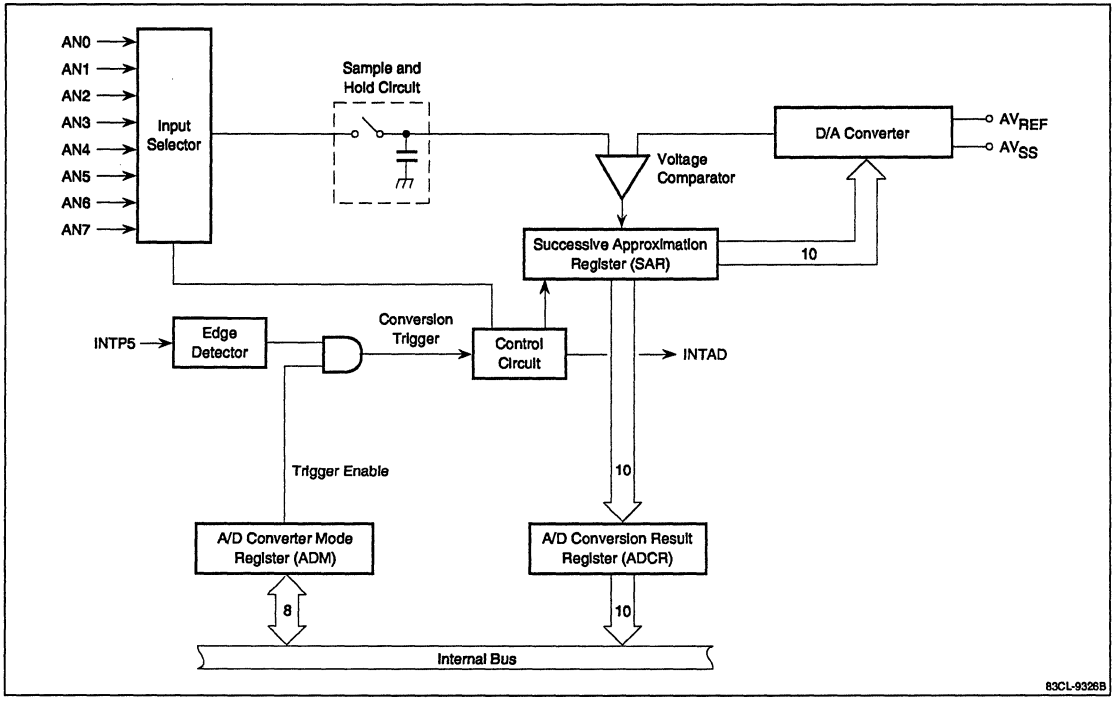
A/D Converter

The analog-to-digital (A/D) converter (figure 4) uses the successive-approximation method for converting up to eight multiplexed analog inputs into 10-bit digital data. The conversion time per input is 18 μs at 16-MHz operation. A/D conversion can be started by an external interrupt, INTP5, or under software control.

The A/D converter can operate in scan mode or select mode. In scan mode, inputs AN0 - AN3 or AN4 - AN7 can be programmed for conversion. The A/D converter selects each of the four inputs in order, converts the data, stores it in the A/D conversion result (ADCR) register, and generates an interrupt (INTAD). This converted data can be easily transferred to memory by the macro service function. In select mode, any one of the eight A/D inputs can be selected for conversion.

Once the A/D converter is started by INTP5 or software, conversion continues until it is disabled by software. The ADCR register is continually updated and either the full 10 bits or only the upper 8 bits of the conversion can be read at any time.

Figure 4. A/D Converter

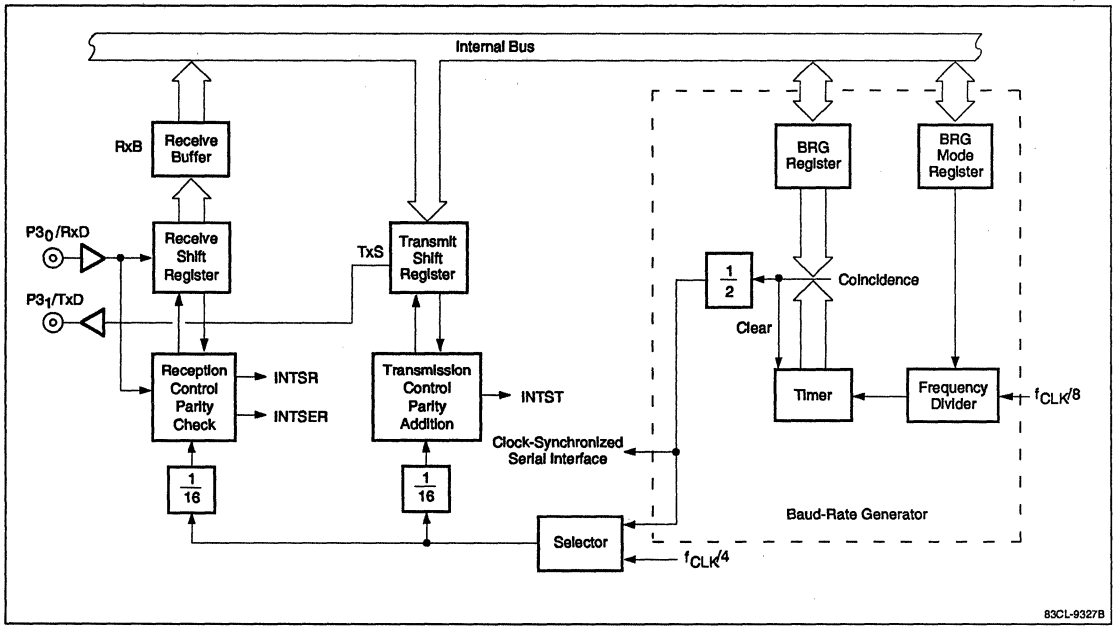


5b

Serial Interface

The μPD78322 family has two independent serial interfaces with a dedicated baud-rate generator. The first is a standard universal asynchronous receiver transmitter (UART). The UART (figure 5) permits full-duplex operation and can be programmed for 7 or 8 bits of data after the start bit, followed by one or two stop bits. Odd, even, zero, or no parity can also be selected.

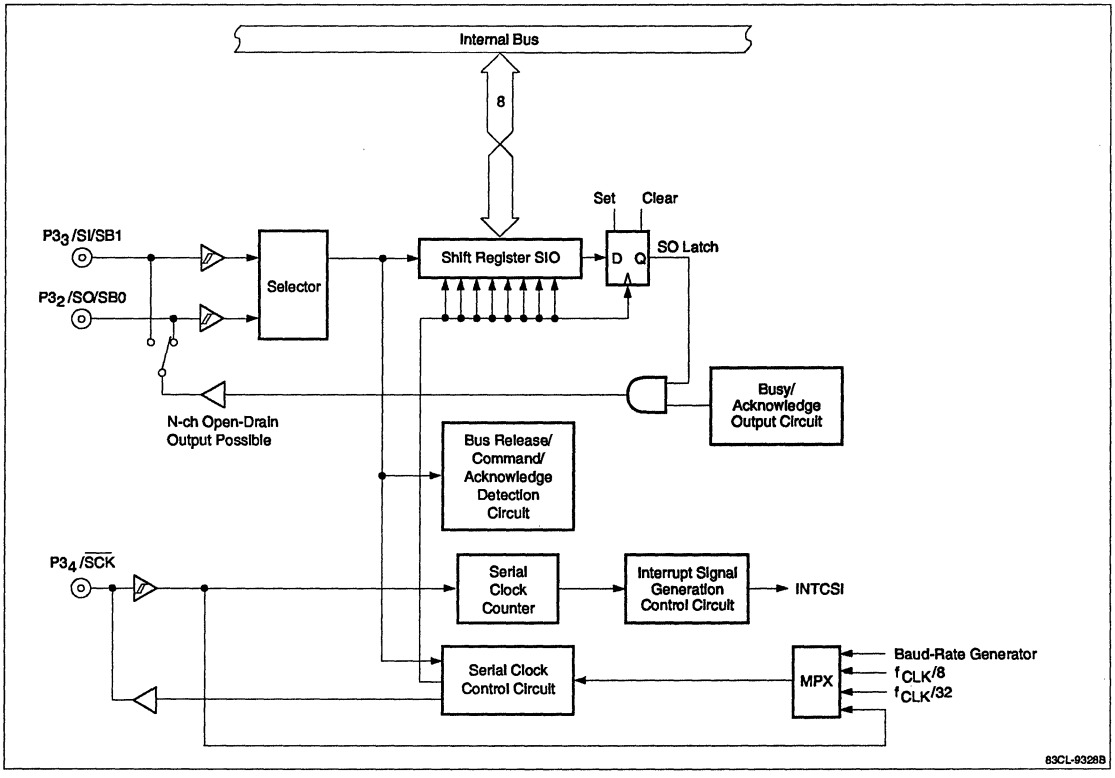
Figure 5. Asynchronous Serial Interface



The source of the serial clock for the UART is the internal system clock (divided by 4) or the on-chip baud-rate generator. The UART generates three interrupts: INTST (transmission complete), INTSR (reception complete), and INTSER (reception error).

The second interface is an 8-bit clock-synchronized serial interface (figure 6). It can be operated in either three-wire serial I/O mode or NEC serial bus interface (SBI) mode.

Figure 6. Clock-Synchronized Serial Interface

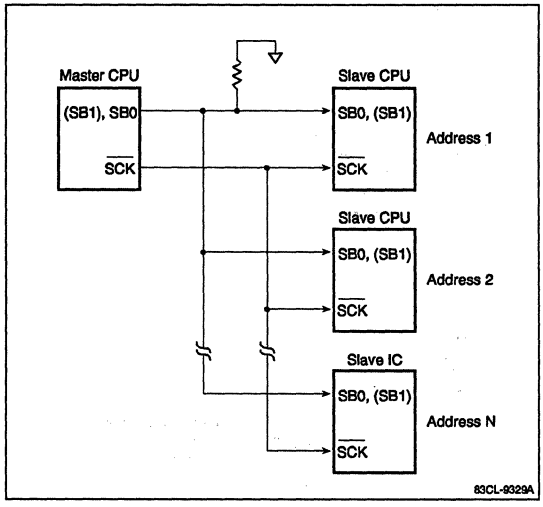


5b

In the three-wire serial I/O mode, the 8-bit shift register (SIO) is loaded with a byte of data and eight clock pulses are generated. These eight pulses shift the byte of data out on the SO line (either MSB or LSB first) and in on the SI line, providing full-duplex operation. This interface can also be set to receive only or to transmit only. The INTCSI interrupt is generated after each 8-bit transfer. One of two internal clocks, an external clock, or the internal baud-rate generator clocks the data.

The NEC SBI mode is a two-wire, high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are connected in a master/slave configuration (figure 7). There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over one of the serial bus lines (SB0 or SB1)using a fixed hardware protocol sysnchronized with the $\overline{SCK}$ line.

Figure 7. SBI Mode Master/Slave Configuration



Each slave μPD78322 family can be programmed in software to respond to any one of 256 addresses. There are also 256 commands and 256 data types. Since all commands are user-definable, any software protocol, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

A dedicated baud-rate generator can be programmed to provide the serial clock to both asynchronous and clock-synchronized serial interfaces. By choosing the correct oscillator frequency, the baud-rate generator is capable of generating all of the commonly used baud rates from 75 to 19,200 b/s.

Real-Time Pulse Unit

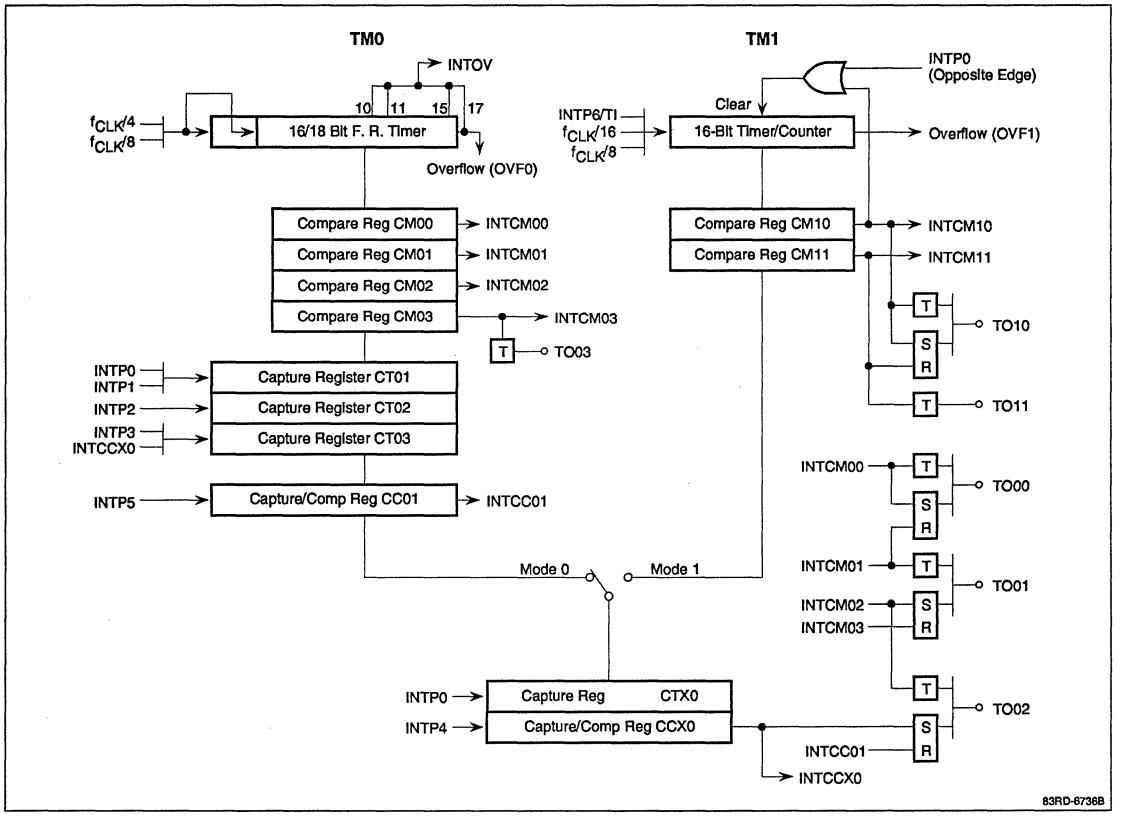
The real-time pulse unit (RPU, figure 8) can function as an interval timer to measure pulse widths and frequencies, generate pulse-width modulated outputs, count external events, and control the real-time output port. It consists of 18-bit free-running timer TM0, 16-bit timer/counter TM1, six 16-bit compare registers, four 18-bit capture registers, two 18-bit registers (capture or compare), and six timed output latches.

TM0 always counts the internal system clock (divided by 4 or 8) and can be reset by an external reset pulse only. TM1 can count either the internal system clock (divided by 8 or 16) or external events. TM1 can be reset by a compare event (a match between a timer and an associated compare register) or by an external signal in INTP0.

Capture events can be triggered by external maskable interrupts INTP0 - INTP5, and compare events can generate interrupts, control timed output pins, or both. In addition, interrupts INTCM03 and INTCCX0 can control the real-time output port

The timed output latches share the six pins of port P8. Four latches can be toggled or set and reset by compare events, and the remaining two can be toggled. These latches, with the macro service facility, can generate up to four pulse-width modulated outputs.

Figure 8. Real-Time Pulse Unit



5b

Interrupts

The μPD78322 family has 19 maskable hardware interrupt sources: 7 external and 12 internal. The external maskable interrupts share pins with port P2. Six of them, INTP0 to INTP5, can also be used to trigger capture events in the real-time pulse unit. In addition, there are two nonmaskable interrupts, two software interrupts, and RESET. The software interrupts, generated by the BRK or BRKCS instruction and the operation code trap, are not maskable. See table 2.

Table 2. Interrupt Sources

Type of Request	Default Priority	Signal Name	Source	Location	Macro Service Control Word	Vector Address TPF = 0	Vector Address TPF = 1
Software	—	—	Operation code trap	CPU	—	003CH	
	—	—	Break instruction	CPU	—	003EH	
Nonmaskable	—	NMI	NMI input pin	External	—	0002H	8002H
	—	INTWDT	Watchdog timer overflow	Internal	—	0004H	8004H
Maskable	0	INTOV	Timer 0 overflow	Internal	FE06H	0006H	8006H
	1	INTP0	INTP0 pin	External	FE08H	0008H	8008H
	2	INTP1	INTP1 pin	External	FE0AH	000AH	800AH
	3	INTP2	INTP2 pin	External	FE0CH	000CH	800CH
	4	INTP3	INTP3 pin	External	FE0EH	000EH	800EH
	5	INTP4 INTCCX0	INTP4 pin CCX0 coincidence	External Internal	FE10H	0010H	8010H
	6	INTP5 INTCC01	INTP5 pin CC01 coincidence	External Internal	FE12H	0012H	8012H
	7	INTP6	INTP6 pin	External	FE14H	0014H	8014H
	8	INTCM00	CM00 coincidence	Internal	FE16H	0016H	8016H
	9	INTCM01	CM01 coincidence	Internal	FE18H	0018H	8018H
	10	INTCM02	CM02 coincidence	Internal	FE1AH	001AH	801AH
	11	INTCM03	CM03 coincidence	Internal	FE1CH	001CH	801CH
	12	INTCM10	CM10 coincidence	Internal	FE1EH	001EH	801EH
	13	INTCM11	CM11 coincidence	Internal	FE20H	0020H	8020H
	14	INTSER	Asynchronous serial Interface reception error	Internal	—	0022H	8022H
	15	INTSR	End of asynchronous serial Interface reception	Internal	FE24H	0024H	8024H
	16	INTST	End of asynchronous serial Interface transmission	Internal	FE26H	0026H	8026H
	17	INTCSI	End of clocked serial Interface transmission/reception	Internal	FE28H	0028H	8028H
	18	INTAD	End of A/D conversion	Internal	FE2AH	002AH	802AH
Reset	—	$\overline{\text{RESET}}$	$\overline{\text{RESET}}$ pin	External	—	0000H	

Interrupt Servicing

The μPD78322 family provides three levels of programmable hardware priority control and three different methods of handling maskable interrupt requests: standard vectoring, context switching, and macro service. The programmer can choose the priority and mode of servicing each maskable interrupt by using the interrupt control registers.

Interrupt Control Registers

The μPD78322 family has ten 16-bit interrupt control registers. Each bit in each register is dedicated to one of the 19 maskable interrupt sources. The interrupt request flag registers (IF0, IF1) contain an interrupt request flag for each interrupt. The interrupt mask registers (MK0, MK1) are used to enable or disable any interrupt. The interrupt service mode registers (ISM0,

ISM1) specify whether an interrupt is processed by vectoring or macro service.

The priority specification buffer registers (PB0, PB1), in conjunction with the 8-bit priority specification register (PRSL), can be used to specify one of three priority levels for each interrupt. The context switching enable flag registers (CSE0, CSE1) specify whether an interrupt is processed by vectoring or context switching.

Two other 8-bit registers are associated with interrupt processing. The in-service priority register (ISPR) is used by the hardware to hold the priority level of the interrupt request currently being serviced. It is manipulated by hardware only, but it can be read by software.

The IE and LT bits of the program status word (PSW) are also used to control interrupts. If the IE bit is zero, all maskable interrupts, but not macro service, are disabled. The IE bit can be set or cleared using the EI and DI instructions, respectively, or by directly writing to the PSW. The IE bit is cleared each time an interrupt is accepted. The LT bit is set by hardware when a newly accepted maskable interrupt request is assigned a priority higher than the interrupt currently being serviced. The LT flag is used to control resetting the ISPR register when a return instruction from an interrupt service routine is executed.

Interrupt Priority

The two nonmaskable interrupts, NMI and WDT, have priority over all others. Their priority relative to each other is under program control.

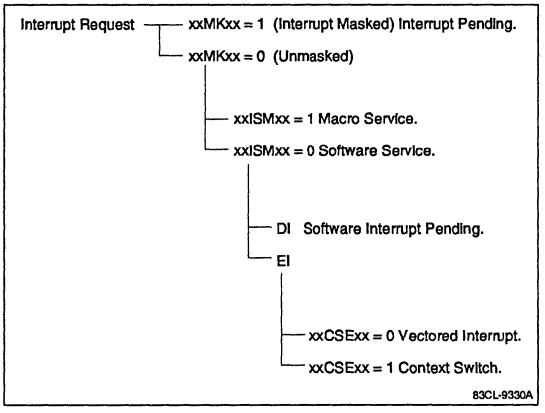
Three hardware-controlled priority levels are available for the maskable interrupts. Any one of the three levels can be assigned by software to each of the maskable interrupt lines. Interrupt requests of a priority equal to or higher than the processor's current priority level are accepted; lower priority requests are held pending until the processor's priority state is lowered by a return instruction from the current service routine.

Interrupt requests programmed to be handled by macro service have priority over all software interrupt service regardless of the assigned priority level, and macro service requests are accepted even when the interrupt enable bit in the PSW is set to the disable state. See figure 9.

The "Default Priorities" listed in table 2 are fixed by hardware; they are effective only when it is necessary to choose between two interrupt requests of the same software-assigned priority. For example, the default priorities would be used after the completion of a high-priority routine if two interrupts of the same lower priority were pending.

Software interrupts, the BRK and BRKCS instructions, and the operation code trap are executed regardless of the processor's priority level and the state of the IE bit. They do not alter the processor's priority level.

Figure 9. Interrupt Service Sequence



Vectored Interrupt

When vectored interrupt is specified for a given interrupt request, the program status word and the program counter are saved on the stack. The processor's priority is raised to that specified for the interrupt, the IE bit in the PSW is set to zero, and the routine whose address is in the interrupt vector table is entered.

At completion of the service routine, the RETI instruction (or RETB instruction for software interrupts) reverses the process, and the μPD78322 family resumes the interrupted routine.

5b

Context Switch

When context switching (figure 10) is specified for a given interrupt, the active register bank is changed to the register bank specified by the three low-order bits of the word in the interrupt vector table. The program counter is loaded from RP2 of the new register bank, the old program counter and program status word are saved in RP2 and RP3 of the new register bank, and the IE bit in the PSW is set to zero.

At completion of the service routine, the RETCS instruction for routines entered from hardware requests, or the RETCSB instruction for routines entered from the BRKCS instruction, reverse the process. The old program counter and program status word are restored from RP2 and RP3 of the new register bank. The entry address of the service routine, which must be specified in the 16-bit immediate operand of these return instructions, is stored again in RP2.

Macro Service

When macro service is specified for a given interrupt, the macro service hardware temporarily stops the executing program and transfers data between the special function register area and the memory space. Control is then returned to the executing program, providing a completely transparent method of interrupt service. Macro Service significantly improves response time and makes it unnecessary to save any registers.

For each request on the interrupt line, one operation is performed, and an 8-bit counter is decremented. When the counter reaches 0 (or when some other completion condition is met), a software service routine is entered. Either vectored interrupt or context switch can be specified for entry to this routine, which is known as the macro service completion routine, and the routine is entered according to the specified priority.

Macro service is provided for all of the maskable interrupt requests except INTSER, the asynchronous serial interface receive error interrupt request. Each request has a dedicated macro service control word stored in on-chip main RAM. The function to be performed is specified in the control word. See table 3.

Figure 10. Context Switching and Return

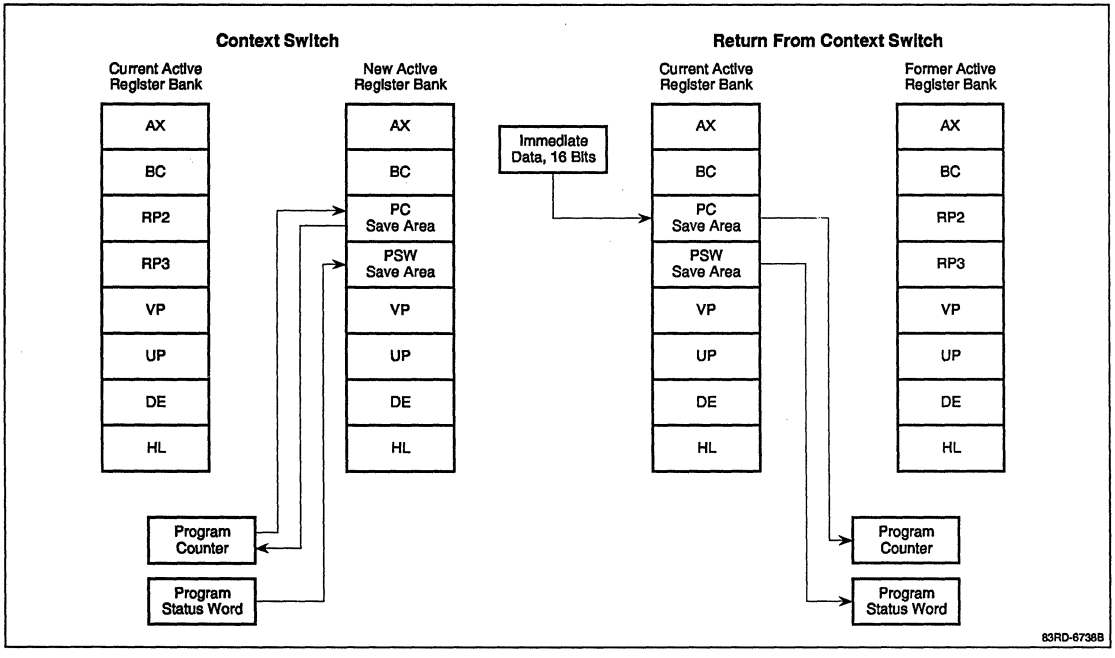


Table 3. Macro Service Functions

Control Word	Function
EVTCNT	Event counter. Counts up to 256 events by incrementing or decrementing the macro service counter. When the counter reaches 00H, the software service routine is entered.
DTACMP	Data compare. If the value of the special-function register requesting macro service matches a byte of data, the software service routine is entered. This mode can be used to detect an address match in SBI mode.
BITSHT	Data shift. Shifts the contents of a specified special function register one bit to the right (toward the LSB). The software service routine is entered when a data bit of 1 has been shifted out from the LSB. This mode can be used to control the real-time output port by shifting the values of the real-time output port set or reset registers.
BITLOG	Bit logic. Performs the logical AND or OR of a data byte and the contents of the specified special function register, stores the result back in the SFR, and enters the software service routine.
ADCBUF	A/D converter buffering. Stores the contents of the A/D conversion result register in a byte or word buffer in main RAM (FExxH). When the macro service counter reaches 0, A/D conversion is stopped but *no software service routine is entered*
BLKTRS	Block transfer. Transfers a byte or word of data in either direction between a specified special-function register and a buffer anywhere in the 64K-byte address space.
DTADIF	Data difference. Stores the difference between the current value of a specified 16-bit special-function register and its previous value in a byte or word buffer in main RAM (FExxH).
DTADIF-P	Data difference with memory pointer. Stores the difference between the current value of a specified special function register and its previous value in a byte or word buffer anywhere in the 64K-byte address space.
DTAADD	Data addition. Stores the sum of a byte or word of data and a specified special function register in the same or another specified special-function register.

Standby Modes

The standby modes, HALT and STOP, reduce power consumption when CPU action is not required. In HALT mode, the CPU is stopped but the system clock continues to run. The HALT mode is released by any nonmaskable interrupt request, unmasked maskable interrupt request, or an external reset pulse. In STOP mode, both the CPU and the system clock are stopped, further minimizing power consumption. The STOP mode is released by either an external reset pulse or an external NMI.

The HALT and STOP modes are entered by programming standby control register STBC. This register is a protected location and can be written to only by a special instruction. If the third and fourth bytes of the instruction are not complements of each other, the data is not written and an operation code trap interrupt occurs.

Watchdog Timer

The watchdog timer protects against inadvertent program loops. A nonmaskable interrupt occurs if the timer is not reset by the program before it overflows. At the same time, watchdog timer output pin $\overline{\text{WDTO}}$ goes active low for a period of 32 system clocks. The $\overline{\text{WDTO}}$ can be connected to the $\overline{\text{RESET}}$ pin or used to control external circuitry. Three program-selectable intervals are available: 8.19, 32.7, and 131.0 ms at 16 MHz.

Once started, the timer can be stopped only by an external reset. Watchdog timer mode register WDM is used to select the time interval, to set the relative priority of the watchdog timer interrupt and NMI, and to clear the timer. This register is a protected location and can be written to only by a special instruction. If the third and fourth bytes of the instruction are not complements of each other, the data is not written and an operation code trap interrupt occurs.

External Reset

The μPD78322 family is reset by taking the $\overline{\text{RESET}}$ pin low. The reset circuit contains a noise filter to protect against spurious system resets caused by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low until the power supply reaches its operating voltage and the oscillator has stabilized. During reset, the program counter is loaded with the address contained in the reset vector table (addresses 0000H, 0001H); program execution starts at that address upon the $\overline{\text{RESET}}$ pin going high. While $\overline{\text{RESET}}$ is low, all external lines except $\overline{\text{WDTO}}$, AV_{REF}, AV_{DD}, AV_{SS}, V_{SS}, V_{DD}, X1, and X2 are in the high-impedance state.

5b

ELECTRICAL SPECIFICATIONS

Note: Unless otherwise noted, these specifications apply to the entire μPD78322 family. Exceptions for extended temperature range devices may be singled out in the tables by the symbols below. The applicable symbol is also included in the device part number

Symbol	Operating Temperature Range
( )*	−10 to + 70°C (standard)
(A)	−40 to + 85°C (extended)
(A1)	−40 to + 110°C (extended)
(A2)	−40 to + 125°C (extended)

* ( ) = no suffix

Absolute Maximum Ratings

T_A = 25°C

Supply voltage, V_{DD}	−0.5 to +7.0 V
Supply voltage, AV_{DD}	−0.5 to V_{DD} + 0.5 V
Supply voltage, AV_{SS}	−0.5 to + 0.5 V
Supply voltage, V_{PP}	−0.5 to +13.5 V
Input voltage, V_I	
Except $P2_0$/NMI of μPD78P322	−0.5 to V_{DD} + 0.5 V
$P2_0$/NMI of μPD78P322	−0.5 to +13.5 V
Output voltage, V_O	−0.5 to V_{DD} + 0.5 V
Reference intput voltage, AV_{REF} (f_{XX} ≤ 16 MHz)	−0.5 to AV_{DD} + 0.3 V
Output current, low; I_{OL}	
Each output pin	4.0 mA
Total	90 mA
Output current, high; I_{OH}	
Each output pin	−1.0 mA
Total	−20 mA (A) devices
Operating temperature, T_{OPT}	
( ) devices	−10 to +70°C
(A) devices	−40 to +85°C
(A1) devices	−40 to +110°C
(A2) devices	−40 to +125°C
Storage temperature, T_{STG}	−65 to +150°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC characteristics.

Recommended Operating Conditions

Device	T_A	V_{DD}	Oscillator Freq, f_{XX}
μPD78320/322	−10 to +70°C	+5.0 V ±10%	8 to 16 MHz
μPD78P322	−10 to +70°C	+5.0 V ±5%	
(A) devices	−40 to +85°C	+5.0 V ±10%	
(A1) devices	−40 to +110°C	+5.0 V±10%	8 to 12 MHz
(A2) devices	−40 to +125°C		

Capacitance

T_A = 25°C; V_{DD} = V_{SS} = 0 V

Parameter	Symbol	Max	Unit	Conditions
Input pin capacitance	C_I	20†	pF	f = 1 MHz; unmeasured pins returned to 0 V
Output pin capacitance	C_O	20	pF	
I/O pin capacitance	C_{IO}	20	pF	

† C_I = 10 pF on (A), (A1), and (A2) devices.

Oscillator Characteristics

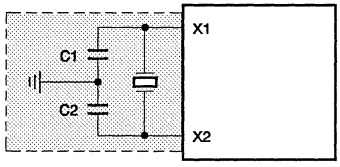
V_{DD} = +5 V ±10% (±5% for μPD78P322); V_{SS} = 0 V
Devices and their T_A ratings are defined earlier by symbols ( ), (A), (A1), and (A2)

Oscillator	Parameter	Symbol	Min	Max	Unit	Conditions
Ceramic or cystal resonator	Oscillation frequency	f_{XX}	8	16 12†	MHz	
External clock input at X1	Frequency	f_X	8	16 12†	MHz	
	Rise time, fall time	t_{XR}, t_{XF}	0	20	ns	
	Low-level, high-level width	t_{WXL}, t_{WXH}	25 46†	80 100†	ns	

† Applicable to (A1) and (A2) devices.

Recommended Oscillator Circuits

A. Resonator Circuit

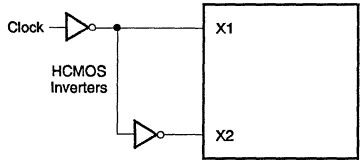


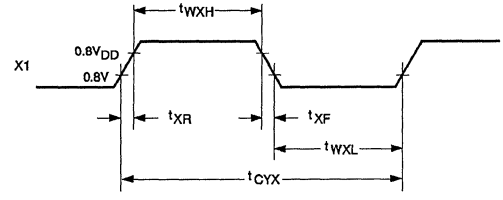
Mount resonator as close as possible to pins X1 and X2.
Do not place other signal lines in shaded area.

Crystal Resonator

Manufacturer	Part No.	C1, C2
Kinseki Co. Ltd.	HC49/U-S or HC49/U	10 pF

B. External Clock Circuit





83CL-9331B

5b

DC Characteristics

V_{DD} = +5.0 V ±10% (±5% for μPD78P322); V_{SS} = 0 V
Devices and their T_A ratings are defined earlier by symbols ( ), (A), (A1), and (A2)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input voltage, low	V_{IL}	0		0.8	V	
Input voltage, high	V_{IH1}	2.2			V	(Note 1)
	V_{IH2}	0.8 V_{DD}			V	(Note 2)
Output voltage, low	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output voltage, high	V_{OH}	V_{DD} – 1.0			V	I_{OH} = –400 μA
Input leakage current	I_{LI}			±10	μA	V_I = 0 to V_{DD}
Output leakage current	I_{LO}			±10	μA	V_O = 0 to V_{DD}
V_{DD} supply current	I_{DD1}		40	65	mA	Operating mode
	I_{DD2}		20	35 45 (Note 3)	mA	HALT mode
Data retention voltage	V_{DDDR}	2.5			V	STOP mode
Data retention current	I_{DDDR}		2	10 100 (Note 3)	μA	STOP mode; V_{DDDR} = 2.5 V
			10	50 1000 (Note 3)	μA	STOP mode; V_{DDDR} = 5.0 V ±10% (±5% for μPD78P322)

Notes:

(1) All except pins in Note 2.

(2) Pins $\overline{RESET}$, X1, X2, $P2_n$, INTPn, NMI, TI, $P3_2$/SB0/SO, $P3_3$/SB1/SI, $P3_4$/SCK.

(3) Applicable to (A1) and (A2) devices.

AC Characteristics

V_{DD} = +5.0 V ±10% (±5% for μPD78P322); V_{SS} = 0 V
Devices and their T_A ratings are defined earlier by symbols ( ), (A), (A1), and (A2)

		( ) and (A)		(A1) and (A2)			
Parameter	Symbol	Min	Max	Min	Max	Unit	Conditions
External Memory Read/Write Operation							
System clock cycle time	t_{CYK}	125	250	166	250	ns	t_{CYK} equals twice the period of the crystal or external clock input.
Address setup time to ASTB ↓	t_{SAST}	32		43		ns	t_{CYK} = 125 ns
Address hold time after ASTB ↓	t_{HSTA}	32		53		ns	
Address to $\overline{RD}$ ↓ delay time	t_{DAR}	85		126		ns	
$\overline{RD}$ ↓ to address floating	t_{FRA}		0		0	ns	
Address to data input	t_{DAID}		222		326	ns	
$\overline{RD}$ ↓ to data input	t_{DRID1}		112		174	ns	
ASTB ↓ to $\overline{RD}$ ↓ delay time	t_{DSTR}	42		63		ns	
Data hold time from $\overline{RD}$ ↑	t_{HRID}	0		0		ns	
$\overline{RD}$ ↑ to address active	t_{DRA}	50 ( ) 37 (A)		58		ns	
$\overline{RD}$ width low	t_{WRL}	157		219 (A1) 209 (A2)		ns	
ASTB width, high	t_{WSTH}	37		58		ns	
Address to $\overline{WR}$ ↓ delay time	t_{DAW}	85		126		ns	
ASTB ↓ to data output	t_{DSTOD}		102		123	ns	

AC Characteristics (cont)

Parameter	Symbol	( ) and (A) Min	( ) and (A) Max	(A1) and (A2) Min	(A1) and (A2) Max	Unit	Conditions
External Memory Read/Write Operation (cont)							
$\overline{WR}$ ↓ to data output	t_{DWOD}		40		40	ns	t_{CYK} = 125 ns
ASTB ↓ to $\overline{WR}$ ↓ delay	t_{DSTW}	42		63		ns	
Data setup time to $\overline{WR}$ ↑	t_{SODW}	147		204		ns	
Data hold time after $\overline{WR}$ ↑	t_{HWOD}	32		53		ns	
$\overline{WR}$ width, low	t_{WWL}	157		209		ns	
$\overline{WR}$ ↑ to $\overline{ASTB}$ delay time	t_{DWST}	42		63		ns	
Serial Port Operation							
$\overline{SCK}$ cycle time	t_{CYSK}	1000		1328		ns	$\overline{SCK}$ output
		1000		1328		ns	$\overline{SCK}$ input
$\overline{SCK}$ width low	t_{WSKL}	420		584		ns	$\overline{SCK}$ output
		420		584		ns	$\overline{SCK}$ input
$\overline{SCK}$ width high	t_{WSKH}	420		584		ns	$\overline{SCK}$ output
		420		584		ns	$\overline{SCK}$ input
SI setup time to $\overline{SCK}$	t_{SRXSK}	80		80		ns	
SI hold time after $\overline{SCK}$ ↑	t_{HSKRX}	80		80		ns	
$\overline{SCK}$ ↓ to SO delay time	t_{DSKTX}		210		210	ns	R = 1 kΩ; C = 100 pF
Other Operations							
NMI high/low-level width	t_{WNIH}, t_{WNIL}	5				μs	
INTP0 high/low-level width	t_{WI0H}, t_{WI0L}	8				t_{CYK}	
INTP1 high/low-level width	t_{WI1H}, t_{WI1L}	8				t_{CYK}	
INTP2 high/low-level width	t_{WI2H}, t_{WI2L}	8				t_{CYK}	
INTP3 high/low-level width	t_{WI3H}, t_{WI3L}	8				t_{CYK}	
INTP4 high/low-level width	t_{WI4H}, t_{WI4L}	8				t_{CYK}	
INTP5 high/low-level width	t_{WI5H}, t_{WI5L}	8				t_{CYK}	
INTP6 high/low-level width	t_{WI6H}, t_{WI6L}	8				t_{CYK}	
$\overline{RESET}$ high/low-level width	t_{WRSH}, t_{WRSL}	5				μs	
TI high/low-level width	t_{WTIH}, t_{WTIL}	8				t_{CYK}	During TM1 even-counter mode

5b

Timing Dependent on t_{CYK}

Symbol	Calculation Formula	Min/Max	Unit
t_{DAID}	(2.5 + n)T – 90	Max	ns
t_{DAR}	T – 40	Min	ns
t_{DAW}	T – 40	Min	ns
t_{DRA}	0.5T – 12 ( ) 0.5T – 25 (A) (A1) (A2)	Min	ns
t_{DRID1}	(1.5 + n)T – 75	Max	ns
t_{DSTOD}	0.5T + 40	Max	ns
t_{DSTR}	0.5T – 20	Min	ns
t_{DSTW}	0.5T – 20	Min	ns
t_{DWST}	0.5T – 20	Min	ns
t_{DWOD}	0.5T – 11	Min	ns
t_{HSTA}	0.5T – 30	Min	ns
t_{HWOD}	0.5T – 30	Min	ns
t_{SAST}	0.5T – 30 ( ) (A) 0.5T – 40 (A1) (A2)	Min	ns
t_{SODW}	1.5T – 40 ( ) (A) 0.5T – 45 (A1) (A2)	Max	ns
t_{WRL}	(1.5 + n)T – 30	Min	ns

Timing Dependent on t_{CYK} (cont)

Symbol	Calculation Formula	Min/Max	Unit
t_{WSTH}	0.5T – 25	Min	ns
t_{WWL}	(1.5 + n)T – 30 ( ) (A) (1.5 + n)T – 40 (A1) (A2)	Min	ns
t_{WI0H}, t_{WI0L}	8T	Min	ns
t_{WI1H}, t_{WI1L}	8T	Min	ns
t_{WI2H}, t_{WI2L}	8T	Min	ns
t_{WI3H}, t_{WI3L}	8T	Min	ns
t_{WI4H}, t_{WI4L}	8T	Min	ns
t_{WI5H}, t_{WI5L}	8T	Min	ns
t_{WI6H}, t_{WI6L}	8T	Min	ns
t_{WTIH}, t_{WTIL}	8T	Min	ns

Notes:

(1) n is the number of wait cycles specified by the PWC register.

(2) T = t_{CYK} (ns)

(3) Devices and their T_A ratings are defined earlier by symbols ( ), (A), (A1), and (A2)

A/D Converter Characteristics

V_{DD} = +5 V ±10% (±5% for μPD78P322); AV_{SS} = V_{SS} = 0 V; AV_{DD} = (V_{DD} – 0.5 V) to V_{DD}
Devices and their T_A ratings are defined earlier by symbols ( ), (A), (A1), and (A2)

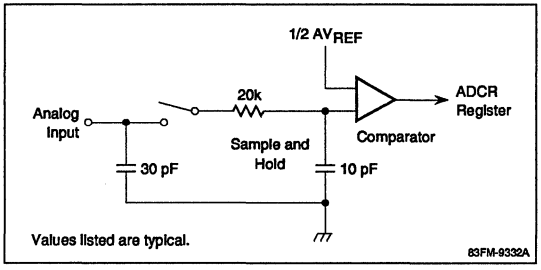
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		10			Bit	
Relative accuracy*				±0.4 ( ) ±0.3 (A) (A1) (A2)	% †	AV_{REF} = 4.5 V to AV_{DD}
				±0.7 ( ) ±0.3 (A) (A1) (A2)	% †	AV_{REF} = 3.4 V to AV_{DD}
Quantization error				±1/2	LSB	
Conversion time	t_{CONV}	144			t_{CYK}	
Sampling time	t_{SAMP}	24			t_{CYK}	
Zero-scale error*			±1.5	±2.5	LSB	AV_{REF} = 4.5 V to AV_{DD}
			±1.5	±4.5	LSB	AV_{REF} = 3.4 V to AV_{DD}
Full-scale error*			±1.5	±2.5	LSB	AV_{REF} = 4.5 V to AV_{DD}
			±1.5	±4.5	LSB	AV_{REF} = 3.4 V to AV_{DD}
Nonlinearity error*			±1.5	±2.5	LSB	AV_{REF} = 4.5 V to AV_{DD}
			±1.5	±4.5	LSB	AV_{REF} = 3.4 V to AV_{DD}
Analog input voltage	V_{IAN}	–0.3		AV_{REF} + 0.3	V	
Reference voltage	AV_{REF}	3.4 or 4.5		AV_{DD}	V	
AV_{REF} current	AI_{REF}		1.0	3.0	mA	
AV_{DD} power current	AI_{DD}		2.0	6.0	mA	

A/D Converter Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
A/D converter data retention current	AI_{DDR}		2	10 ( ) (A) 100 (A1) (A2)	μA	STOP mode; AV_{DDR} = 2.5 V
			10	50 ( ) (A) 1000 (A1) (A2)	μA	STOP mode; AV_{DDR} = 5 V ±10% (±5% for μPD78P322)
Analog input impedance	R_{AN}		10		MΩ	Nonsampling
						Sampling

* Does not include quantization error
† Unit is percent of full-scale range (FSR)

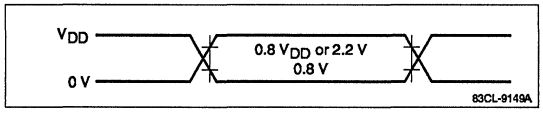
A/D Converter Input Circuit



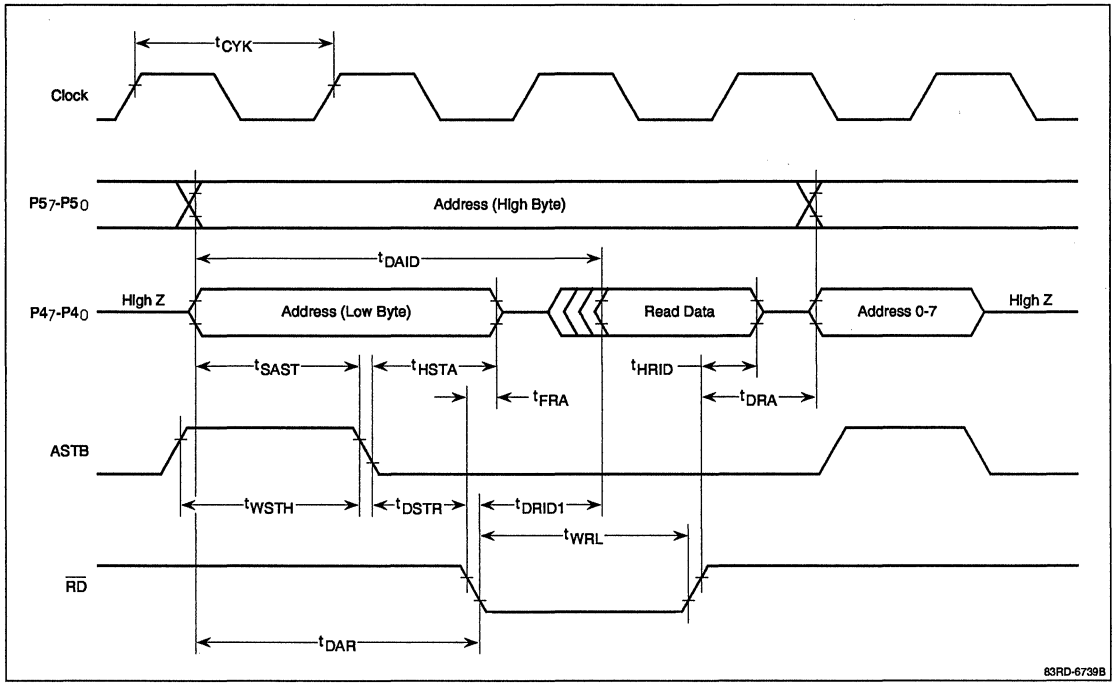
5b

Timing Waveforms

AC Timing Test Points

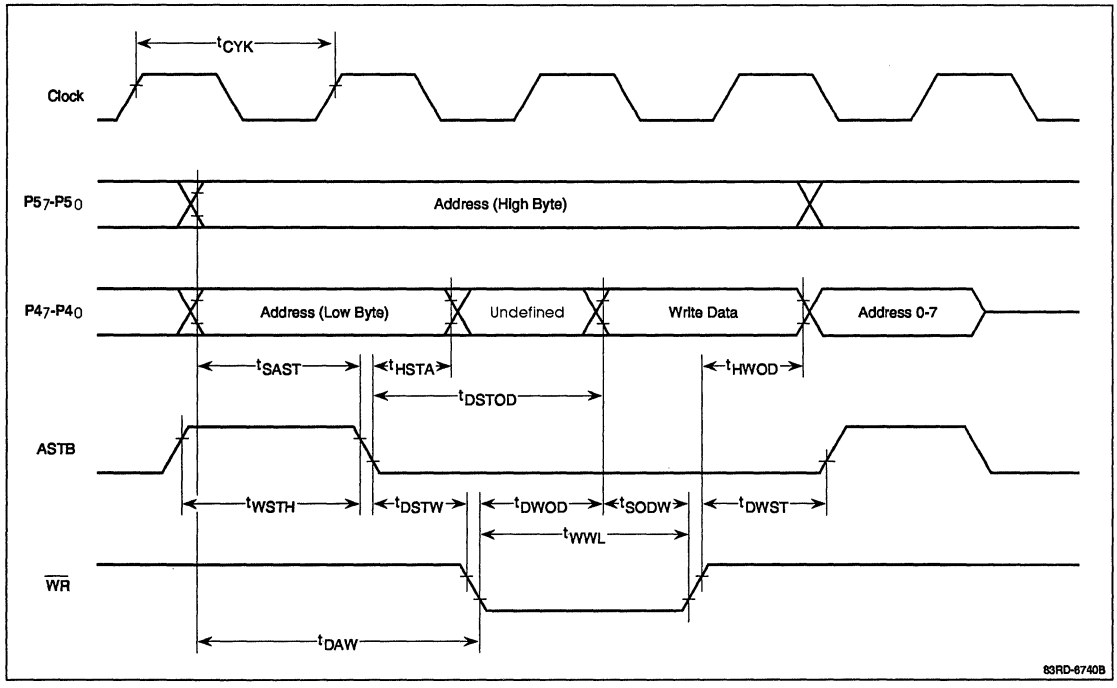


Read Operation

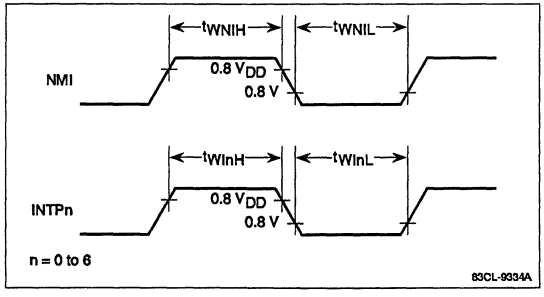


Timing Waveforms (cont)

Write Operation



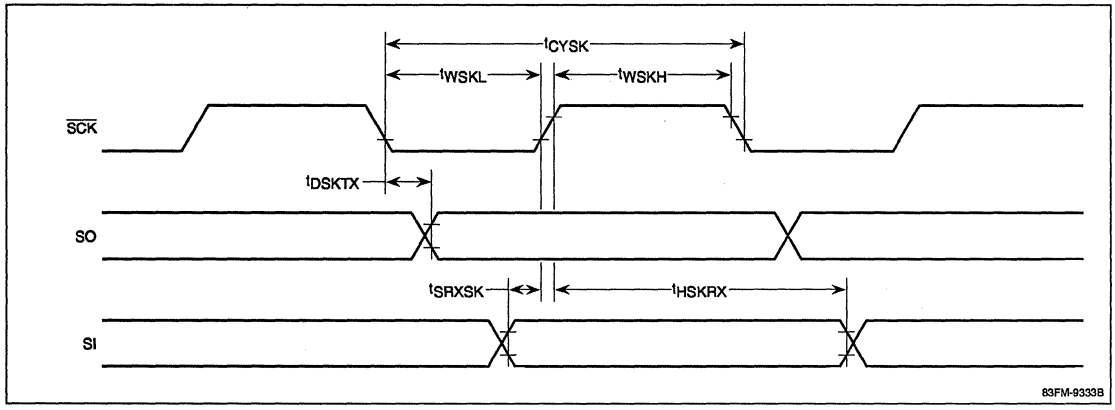
Interrupt Input



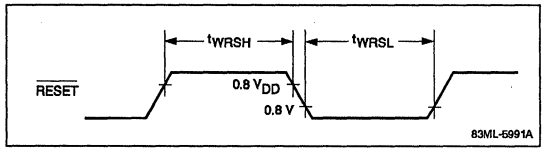
5b

Timing Waveforms (cont)

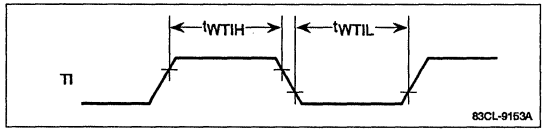
Serial Port Operation



Reset Input



TI Input



PROM PROGRAMMING

The PROM in the μPD78P322 is one-time programmable (OTP) or ultraviolet erasable (UV EPROM). The 16,384 x 8-bit PROM has the programming characteristics of an NEC μPD27C256A. Table 3 shows the functions of the μPD78P322 pins in normal operating mode and PROM programming mode.

PROM Programming Mode

When the $\overline{RESET}$ pin is set high and AV_{DD} is set low, the μPD78P322 enters the PROM programming mode. Operation in this mode is determined by the setting of the $\overline{CE}$, $\overline{OE}$, V_{PP}, and V_{DD} pins as indicated in table 4.

Table 3. Pin Functions During PROM Programming

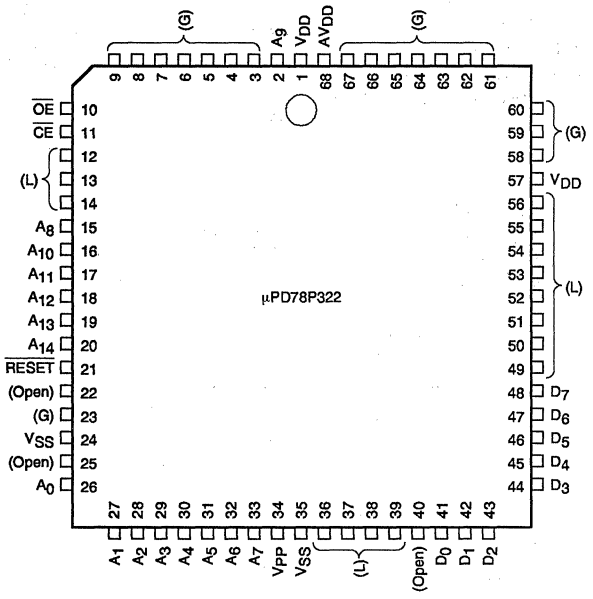
Function	Normal Operating Mode	Programming Mode
Address input	$P0_0$ - $P0_7$, $P8_0$, $P2_0$, $P8_1$ - $P8_7$	A_0 - A_{14}
Data input	$P4_0$ - $P4_7$	D_0 - D_7
Chip enable/ program pulse	$P3_1$	$\overline{CE}$
Output enable	$P3_0$	$\overline{OE}$
Program voltage	V_{PP}	V_{PP}
Mode voltage	$\overline{RESET}$, AV_{DD}	$\overline{RESET}$, AV_{DD}

Table 4. Operation Modes For Programming

Mode	$\overline{RESET}$	AV_{DD}	$\overline{CE}$	$\overline{OE}$	V_{PP}	V_{DD}	D_0 - D_7
Program write	H	L	L	H	+12.5 V	+6.0 V	Data input
Program verify	H	L	H	L	+12.5 V	+6.0 V	Data output
Program inhibit	H	L	H	H	+12.5 V	+6.0 V	High impedance
Read	H	L	L	L	+5.0 V	+5.0 V	Data output
Ouput disable	H	L	L	H	+5.0 V	+5.0 V	High impedance
Standby	H	L	H	L/H	+5.0 V	+5.0 V	High impedance

5b

Figure 11. Pin Functions in μPD78P322 PROM Programming Mode; 68-Pin PLCC or LCC



Note:

Recommended connections for pins not used during PROM programming:

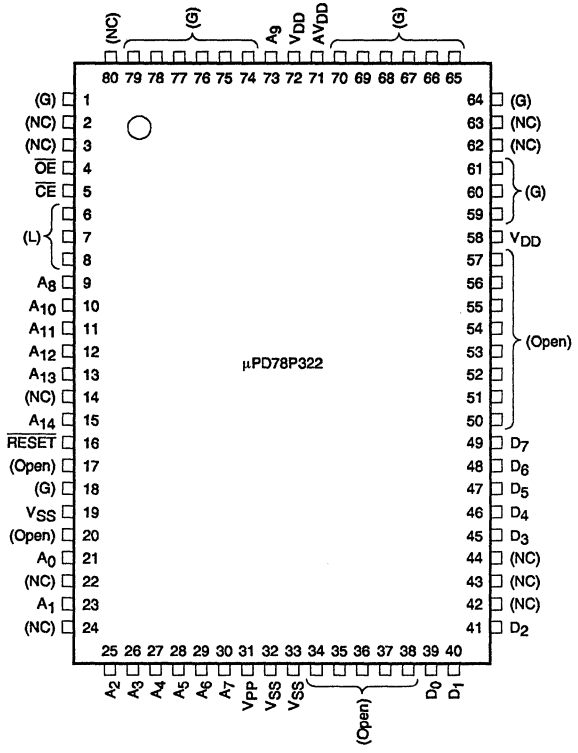
(L) Connect to V_{SS} through a resistor.

(G) Connect to V_{SS}.

(Open) No connection.

83CL-9335B

Figure 12. Pin Functions in μPD78P322 PROM Programming Mode; 80-Pin QFP or LCC



Note:

Recommended connections for pins not used during PROM programming:

(L) Connect to V_{SS} through a resistor.

(G) Connect to V_{SS}.

(Open) No connection

(NC) Connect to V_{SS} to prevent noise.

83CL-9336B

5b

PROM Write Procedure

Data can be written to the PROM by the following procedure.

(1) Set the pins not used for programming as indicated in figure 11 or 12. Set RESET high and AV_{DD} low. $\overline{CE}$ and $\overline{OE}$ should be high.

(2) Supply +6.0 V to V_{DD} pin and +12.5 V to V_{PP} pin.

(3) Provide initial address to pins A_0 - A_{14}.

(4) Provide write data.

(5) Input a 1-ms program pulse (active low) to $\overline{CE}$ pin.

(6) Use verify mode (pulse $\overline{OE}$ low) to test data. If data has been written, proceed to step 8; if not, repeat steps 4–6. If data cannot be written in 25 attempts, go to step 7.

(7) Classify PROM as defective and cease write operation.

(8) Perform one additional write with a program pulse width (in ms) equal to 3 times the number of writes performed in step 5.

(9) Increment address.

(10) Repeat steps 4–9 until last address is programmed.

PROM Read Procedure

The contents of the PROM can be read out to the external data bus (D_0 - D_7) by the following procedure.

(1) Set the pins not used for programming as indicated in figure 11 or 12. Set RESET high and AV_{DD} low. $\overline{CE}$ and $\overline{OE}$ should be active high.

(2) Supply +5 V to V_{DD} pin and V_{PP} pin.

(3) Input address of data to be read to pins A_0 - A_{14}.

(4) Put an active-low pulse on $\overline{CE}$ and $\overline{OE}$ pins.

(5) Data is output to pins D_0 - D_7.

Program Erasure

The UV EPROM can be erased by exposing the window to light having a wavelength shorter than 400 nm, including ultraviolet, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15 Ws/cm^2 (ultraviolet ray intensity x exposure time) is required to completely erase the written data. Erasure by an ultraviolet lamp rated at 12,000 $\mu W/cm^2$ takes 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

DC Programming Characteristics

T_A = 25 ±5°C; V_{SS} = 0 V

Parameter	Symbol	Symbol (Note 1)	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH}	V_{IH}	2.2		V_{DDP} + 0.3	V	
Low-level input voltage	V_{IL}	V_{IL}	−0.3		0.8	V	
Input leakage current	I_{LIP}	I_{LI}			±10	μA	0 ≤ V_I ≤ V_{DDP} (Note 2)
High-level output voltage	V_{OH}	V_{OH}	2.4			V	I_{OH} = −400 μA
Low-level output voltage	V_{OL}	V_{OL}			0.45	V	I_{OL} = 2.0 mA
A9 pin input current	I_{A9}	—			±10	μA	A9 ($P2_0$/NMI)
Output leakage current	I_{LO}	—			10	μA	0 ≤ V_O ≤ V_{DDP}, $\overline{OE}$ = V_{IH}
PROG pin high-voltage input current	I_{IP}	—			±10	μA	
V_{DDP} power supply voltage	V_{DDP}	V_{DD}	5.75	6.0	6.25	V	Program memory write mode
			4.5	5.0	5.5	V	
V_{PP} power supply voltage	V_{PP}	V_{PP}	12.2	12.5	12.8	V	
				V_{PP} = V_{DDP}		V	
V_{DDP} power supply current	I_{DD}	I_{DD}		10	30	mA	
				10	30	mA	Program memory read mode $\overline{CE}$ = V_{IL}, V_I = V_{IH}
V_{PP} power supply current	I_{PP}	I_{PP}		10	30	mA	Program memory write mode $\overline{CE}$ = V_{IL}, $\overline{OE}$ = V_{IH}
				1	100	μA	Program memory read mode

Notes:

(1) Corresponding symbols for the μPD27C256A

(2) V_{DDP} is the V_{DD} pin during programming

AC Programming Characteristics

T_A = 25 ±5°C; V_{SS} = 0 V

Parameter	Symbol	Symbol (Note 1)	Min	Typ	Max	Unit	Condition
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}	2			μs	
Data to to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}	2			μs	
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}	2			μs	
Address hold time after $\overline{CE}$ ↑	t_{HCA}	t_{AH}	2			μs	
Input data hold time after $\overline{CE}$ ↑	t_{HCID}	t_{DH}	2			μs	
Output data hold time after $\overline{OE}$ ↑	t_{HOOD}	t_{DF}	0		130	ns	
V_{PP} setup time before $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}	2			μs	
V_{DDP} setup time before $\overline{CE}$ ↓	t_{SVDC}	t_{VDS}	2			μs	
Initial program pulse width	t_{WL1}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{WL2}	t_{OPW}	2.85		78.75	ms	
Address to data output time	t_{DAOD}	t_{ACC}			2	μs	$\overline{OE}$ = V_{IL}
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			1	μs	
Data hold time after $\overline{OE}$ ↓	t_{HCOD}	t_{DF}	0		130	ns	
Data hold time after address not valid	t_{HAOD}	t_{OH}	0			ns	$\overline{OE}$ = V_{IL}

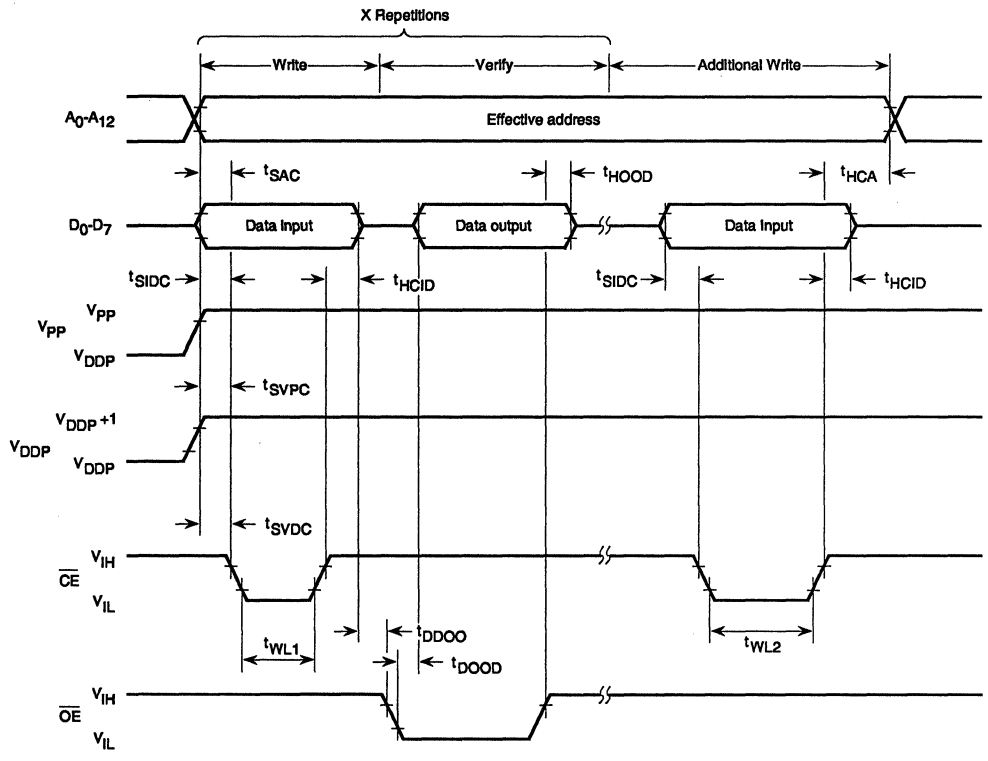
Notes:

(1) Corresponding symbols for the μPD27C256A.

5b

PROM Timing Diagrams

Write Mode



Notes:

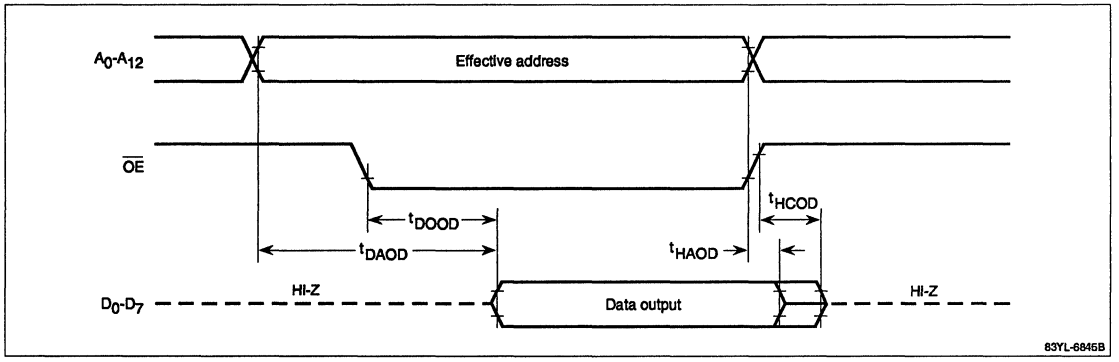
[1] V_{DDP} must be applied before V_{PP} is applied and must be removed after V_{PP} is removed.

[2] V_{PP} must not exceed +13 V including overshoot voltage.

83CL-9337B

PROM Timing Diagrams (cont)

Read Mode



5b

INSTRUCTION SET

The μPD78322 family instruction set features 8- and 16-bit data transfer, arithmetic, and logic instructions and single-bit manipulation instructions. String manipulation instructions are also included. Branch instructions exist to test individual bits in the program status word, the 16-bit accumulator, the special function registers, and the saddr portion of on-chip RAM. Instructions range in length from 1 to 6 bytes depending on the instruction and addressing mode.

Flag Column Indicators

Symbol	Action
(blank)	No change
0	Set to 0
1	Set to 1
X	Set or cleared according to result
P	P/V indicates parity of result
V	P/V indicates arithmetic overflow
R	Restored from saved PSW

Instruction Set Symbols

Symbol	Definition
r	R0, R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14, R15
r1	R0, R1, R2, R3, R4, R5, R6, R7
r2	C, B
rp	RP0, RP1, RP2, RP3, RP4, RP5, RP6, RP7*
rp1	RP0, RP1, RP2, RP3, RP4, RP5, RP6, RP7*
rp2	DE, HL, VP, UP
sfr	Special function register, 8 bits
sfrp	Special function register, 16 bits
post	RP0, RP1, RP2, RP3, RP4, RP5/PSW, RP6, RP7. Bits set to 1 indicate register pairs to be pushed/popped to/from stack; RP5 pushed/popped by PUSH/POP, SP is stack pointer; PSW pushed/popped by PUSHU/POPU, RP5 is stack pointer.
mem	Register indirect: [DE], [HL], [DE+], [HL+], [DE–], [HL–], [VP], [UP] Base Index Mode: [DE+A], [HL+A], [DE+B], [HL+B], [VP+DE], [VP+HL] Base Mode: [DE+byte], [HL+byte], [VP+byte], [UP+byte], [SP+byte] Index Mode: word [A], word [B], word [DE], word [HL]
saddr	FE20-FF1FH: Immediate byte addresses one byte in RAM, or label
saddrp	FE20-FF1FH: Immediate byte (bit 0=0) addresses one word in RAM, or label

Instruction Set Symbols (cont)

Symbol	Definition
word	16 bits of immediate data or label
byte	8 bits of immediate data or label
jdisp8	8-bit two's complement displacement (immediate data displacement value –128 to +127)
bit	3 bits of immediate data (bit position in byte), or label
n	3 bits of immediate data
!addr16	16-bit absolute address specified by an immediate address or label
$addr16	Relative branch address or label
addr16	16-bit address
!addr11	11-bit immediate address or label
addr11	0800H-0FFFH: 0800H + (11-bit immediate address), or label
addr5	0040H-007EH: 0040H + 2 X (5-bit immediate address), or label
A	A register (8-bit accumulator)
X	X register
B	B register
C	C register
D	D register
E	E register
H	H register
L	L register
R0-R15	Register 0 to register 15
AX	Register pair AX (16-bit accumulator)
BC	Register pair BC
DE	Register pair DE
HL	Register pair HL
RP0-RP7	Register pair 0 to register pair 7
PC	Program counter
SP	Stack pointer
UP	User stack pointer (RP5)
PSW	Program status word
PSWH	High-order 8 bits of PSW
PSWL	Low-order 8 bits of PSW
CY	Carry flag
AC	Auxiliary carry flag
Z	Zero flag
P/V	Parity/overflow flag
S	Sign flag
TPF	Table position flag

Instruction Set Symbols (cont)

Symbol	Definition
RBS	Register bank select flag
RSS	Register set select flag
IE	Interrupt enable flag
STBC	Standby control register
WDM	Watchdog timer mode register
( )	Contents of the location whose address is within parentheses; (+) and (–) indicate that the address is incremented after or decremented after it is used
(( ))	Contents of the memory location defined by the quantity within the sets of parentheses
xxH	Hexadecimal quantity
X_H, X_L	High-order 8 bits and low-order 8 bits of X

* rp and rp1 describe the same registers but generate different machine code.

Instruction Set

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
8-Bit Data Transfer								
MOV	r1, #byte	r1 ← byte	2					
	saddr, #byte	(saddr) ← byte	3					
	sfr, #byte (Note 1)	sfr ← byte	3					
	r, r1	r ← r1	2					
	A, r1	A ← r1	1					
	A, saddr	A ← (saddr)	2					
	saddr, A	(saddr) ← A	2					
	saddr, saddr	(saddr) ← (saddr)	3					
	A, sfr	A ← sfr	2					
	sfr, A	sfr ← A	2					
	A, mem (Note 2)	A ← (mem)	1					
	A, mem	A ← (mem)	2-4					
	mem, A (Note 2)	(mem) ← A	1					
	mem, A	(mem) ← A	2-4					
	A, [saddrp]	A ← ((saddrp))	2					
	[saddrp], A	((saddrp)) ← A	2					
	A, !addr16	A ← (addr16)	4					
	!addr16, A	(addr16) ← A	4					
	PSWL, #byte	PSWL ← byte	3	X	X	X	X	X
	PSWH, #byte	PSWH ← byte	3					
	PSWL, A	PSWL ← A	2	X	X	X	X	X
	PSWH, A	PSWH ← A	2					
	A, PSWL	A ← PSWL	2					
	A, PSWH	A ← PSWH	2					
XCH	A, r1	A ↔ r1	1					
	r, r1	r ↔ r1	2					
	A, mem	A ↔ (mem)	2-4					
	A, saddr	A ↔ (saddr)	2					
	A, sfr	A ↔ sfr	3					
	A, [saddrp]	A ↔ ((saddrp))	2					
	saddr, saddr	(saddr) ↔ (saddr)	3					
16-Bit Data Transfer								
MOVW	rp1, #word	rp1 ← word	3					
	saddrp, #word	(saddrp) ← word	4					
	sfrp, #word	sfrp ← word	4					
	rp, rp1	rp ← rp1	2					
	AX, saddrp	AX ← (saddrp)	2					
	saddrp, AX	(saddrp) ← AX	2					
	saddrp, saddrp	(saddrp) ← (saddrp)	3					

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
16-Bit Data Transfer (cont)								
MOVW (cont)	AX, sfrp	AX ← sfrp	2					
	sfrp, AX	sfrp ← AX	2					
	rp1, !addr16	rp1 ← (addr16)	4					
	!addr16, rp1	(addr16) ← rp1	4					
	AX, mem	AX ← (mem)	2-4					
	mem, AX	(mem) ← AX	2-4					
XCHW	AX, saddrp	AX ↔ (saddrp)	2					
	AX, sfrp	AX ↔ sfrp	3					
	saddrp, saddrp	(saddrp) ↔ (saddrp)	3					
	rp, rp1	rp ↔ rp1	2					
	AX, mem	AX ↔ (mem)	2-4					
8-Bit Arithmetic								
ADD	A, #byte	A, CY ← A + byte	2	X	X	X	V	X
	saddr, #byte	(saddr), CY ← (saddr) + byte	3	X	X	X	V	X
	sfr, #byte	sfr, CY ← sfr + byte	4	X	X	X	V	X
	r, r1	r, CY ← r + r1	2	X	X	X	V	X
	A, saddr	A, CY ← A + (saddr)	2	X	X	X	V	X
	A, sfr	A, CY ← A + sfr	3	X	X	X	V	X
	saddr, saddr	(saddr), CY ← (saddr) + (saddr)	3	X	X	X	V	X
	A, mem	A, CY ← A + (mem)	2-4	X	X	X	V	X
	mem, A	(mem), CY ← (mem) + A	2-4	X	X	X	V	X
ADDC	A, #byte	A, CY ← A + byte + CY	2	X	X	X	V	X
	saddr, #byte	(saddr), CY ← (saddr) + byte + CY	3	X	X	X	V	X
	sfr, #byte	sfr, CY ← sfr + byte + CY	4	X	X	X	V	X
	r, r1	r, CY ← r + r1 + CY	2	X	X	X	V	X
	A, saddr	A, CY ← A + (saddr) + CY	2	X	X	X	V	X
	A, sfr	A, CY ← A + sfr + CY	3	X	X	X	V	X
	saddr, saddr	(saddr), CY ← (saddr) + (saddr) + CY	3	X	X	X	V	X
	A, mem	A, CY ← A + (mem) + CY	2-4	X	X	X	V	X
	mem, A	(mem), CY ← (mem) + A + CY	2-4	X	X	X	V	X
SUB	A, #byte	A, CY ← A – byte	2	X	X	X	V	X
	saddr, #byte	(saddr), CY ← (saddr) – byte	3	X	X	X	V	X
	sfr, #byte	sfr, CY ← sfr – byte	4	X	X	X	V	X
	r, r1	r, CY ← r – r1	2	X	X	X	V	X
	A, saddr	A, CY ← A – (saddr)	2	X	X	X	V	X
	A, sfr	A, CY ← A – sfr	3	X	X	X	V	X
	saddr, saddr	(saddr), CY ← (saddr) – (saddr)	3	X	X	X	V	X
	A, mem	A, CY ← A – (mem)	2-4	X	X	X	V	X
	mem, A	(mem), CY ← (mem) – A	2-4	X	X	X	V	X

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
8-Bit Arithmetic (cont)								
SUBC	A, #byte	A, CY ← A – byte – CY	2	X	X	X	V	X
	saddr, #byte	(saddr), CY ← (saddr) – byte – CY	3	X	X	X	V	X
	sfr, #byte	sfr, CY ← sfr – byte – CY	4	X	X	X	V	X
	r, r1	r, CY ← r – r1 – CY	2	X	X	X	V	X
	A, saddr	A, CY ← A – (saddr) – CY	2	X	X	X	V	X
	A, sfr	A, CY ← A – sfr – CY	3	X	X	X	V	X
	saddr, saddr	(saddr), CY ← (saddr) – (saddr) – CY	3	X	X	X	V	X
	A, mem	A, CY ← A – (mem) – CY	2-4	X	X	X	V	X
	mem, A	(mem), CY ← (mem) – A – CY	2-4	X	X	X	V	X
8-Bit Logic								
AND	A, #byte	A ← A ∧ byte	2	X	X		P	
	saddr, #byte	(saddr) ← (saddr) ∧ byte	3	X	X		P	
	sfr, #byte	sfr ← sfr ∧ byte	4	X	X		P	
	r, r1	r ← r ∧ r1	2	X	X		P	
	A, saddr	A← A ∧ (saddr)	2	X	X		P	
	A, sfr	A← A ∧ sfr	3	X	X		P	
	saddr, saddr	(saddr)← (saddr) ∧ (saddr)	3	X	X		P	
	A, mem	A ← A ∧ (mem)	2-4	X	X		P	
	mem, A	(mem) ← (mem) ∧ A	2-4	X	X		P	
OR	A, #byte	A ← A ∨ byte	2	X	X		P	
	saddr, #byte	(saddr) ← (saddr) ∨ byte	3	X	X		P	
	sfr, #byte	sfr ← sfr ∨ byte	4	X	X		P	
	r, r1	r ← r ∨ r1	2	X	X		P	
	A, saddr	A ← A ∨ (saddr)	2	X	X		P	
	A, sfr	A ← A ∨ sfr	3	X	X		P	
	saddr, saddr	(saddr)← (saddr) ∨ (saddr)	3	X	X		P	
	A, mem	A ← A ∨ (mem)	2-4	X	X		P	
	mem, A	(mem) ← (mem) ∨ A	2-4	X	X		P	
XOR	A, #byte	A ← A ⊻ byte	2	X	X		P	
	saddr, #byte	(saddr) ← (saddr) ⊻ byte	3	X	X		P	
	sfr, #byte	sfr ← sfr ⊻ byte	4	X	X		P	
	r, r1	r ← r ⊻ r1	2	X	X		P	
	A, saddr	A ← A ⊻ (saddr)	2	X	X		P	
	A, sfr	A ← A ⊻ sfr	3	X	X		P	
	saddr, saddr	(saddr)← (saddr) ⊻ (saddr)	3	X	X		P	
	A, mem	A ← A ⊻ (mem)	2-4	X	X		P	
	mem, A	(mem) ← (mem) ⊻ A	2-4	X	X		P	

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
8-Bit Logic (cont)								
CMP	A, #byte	A – byte	2	X	X	X	V	X
	saddr, #byte	(saddr) – byte	3	X	X	X	V	X
	sfr, #byte	sfr – byte	4	X	X	X	V	X
	r, r1	r – r1	2	X	X	X	V	X
	A, saddr	A – (saddr)	2	X	X	X	V	X
	A, sfr	A – sfr	3	X	X	X	V	X
	saddr, saddr	(saddr) – (saddr)	3	X	X	X	V	X
	A, mem	A – (mem)	2-4	X	X	X	V	X
	mem, A	(mem) – A	2-4	X	X	X	V	X
16-Bit Arithmetic								
ADDW	AX, #word	AX, CY ← AX + word	3	X	X	X	V	X
	saddrp, #word	(saddrp), CY ← (saddrp) + word	4	X	X	X	V	X
	sfrp, #word	sfrp, CY ← sfrp + word	5	X	X	X	V	X
	rp, rp1	rp, CY ← rp + rp1	2	X	X	X	V	X
	AX, saddrp	AX, CY ← AX + (saddrp)	2	X	X	X	V	X
	AX, sfrp	AX, CY ← AX + sfrp	3	X	X	X	V	X
	saddrp, saddrp	(saddrp), CY ← (saddrp) + (saddrp)	3	X	X	X	V	X
SUBW	AX, #word	AX, CY ← AX – word	3	X	X	X	V	X
	saddrp, #word	(saddrp), CY ← (saddrp) – word	4	X	X	X	V	X
	sfrp, #word	sfrp, CY ← sfrp – word	5	X	X	X	V	X
	rp, rp1	rp, CY ← rp – rp1	2	X	X	X	V	X
	AX, saddrp	AX, CY ← AX – (saddrp)	2	X	X	X	V	X
	AX, sfrp	AX, CY ← AX – sfrp	3	X	X	X	V	X
	saddrp, saddrp	(saddrp), CY ← (saddrp) – (saddrp)	3	X	X	X	V	X
CMPW	AX, #word	AX – word	3	X	X	X	V	X
	saddrp, #word	(saddrp) – word	4	X	X	X	V	X
	sfrp, #word	sfrp – word	5	X	X	X	V	X
	rp, rp1	rp – rp1	2	X	X	X	V	X
	AX, saddrp	AX – (saddrp)	2	X	X	X	V	X
	AX, sfrp	AX – sfrp	3	X	X	X	V	X
	saddrp, saddrp	(saddrp) – (saddrp)	3	X	X	X	V	X
Multiplication/Division								
MULU	r1	AX ← A x r1	2					
DIVUW	r1	AX (quotient), r1 (remainder) ← AX ÷ r1	2					
MULUW	rp1	AX (high-order 16 bits), rp1 (low-order 16 bits) ← AX x rp1	2					
DIVUX	rp1	AXDE (quotient), rp1 (remainder) ← AXDE ÷ rp1	2					
MULW (Note 3)	rp1	AX (high-order 16 bits), rp1 (low-order 16 bits) ← AX x rp1	2					

5b

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Increment/Decrement								
INC	r1	r1 ← r1 + 1	1	X	X	X	V	
	saddr	(saddr) ← (saddr) + 1	2	X	X	X	V	
DEC	r1	r1 ← r1 − 1	1	X	X	X	V	
	saddr	(saddr) ← (saddr) − 1	2	X	X	X	V	
INCW	rp2	rp2 ← rp2 + 1	1					
	saddrp	(saddrp) ← (saddrp) + 1	3					
DECW	rp2	rp2 ← rp2 − 1	1					
	saddrp	(saddrp) ← (saddrp) − 1	3					
Shift/Rotate								
ROR	r1, n	(CY, $r1_7 \leftarrow r1_0$, $r1_{m-1} \leftarrow r1_m$) x n times	2				P	X
ROL	r1, n	(CY, $r1_0 \leftarrow r1_7$, $r1_{m+1} \leftarrow r1_m$) x n times	2				P	X
RORC	r1, n	(CY ← $r1_0$, $r1_7$ ← CY, $r1_{m-1} \leftarrow r1_m$) x n times	2				P	X
ROLC	r1, n	(CY ← $r1_7$, $r1_0$ ← CY, $r1_{m+1} \leftarrow r1_m$) x n times	2				P	X
SHR	r1, n	(CY ← $r1_0$, $r1_7$ ← 0, $r1_{m-1} \leftarrow r1_m$) x n times	2	X	X	0	P	X
SHL	r1, n	(CY ← $r1_7$, $r1_0$ ← 0, $r1_{m+1} \leftarrow r1_m$) x n times	2	X	X	0	P	X
SHRW	rp1, n	(CY ← $rp1_0$, $rp1_{15}$ ← 0, $rp1_{m-1} \leftarrow rp1_m$) x n times	2	X	X	0	P	X
SHLW	rp1, n	(CY ← $rp1_{15}$, $rp1_0$ ← 0, $rp1_{m+1} \leftarrow rp1_m$) x n times	2	X	X	0	P	X
ROR4	[rp1]	$A_{3-0} \leftarrow (rp1)_{3-0}$, $(rp1)_{7-4} \leftarrow A_{3-0}$, $(rp1)_{3-0} \leftarrow (rp1)_{7-4}$	2					
ROL4	[rp1]	$A_{3-0} \leftarrow (rp1)_{7-4}$, $(rp1)_{3-0} \leftarrow A_{3-0}$, $(rp1)_{7-4} \leftarrow (rp1)_{3-0}$	2					
BCD Adjustment								
ADJBA		Decimal adjust accumulator after add	2	X	X	X	P	X
ADJBS		Decimal adjust accumulator after subtract	2	X	X	X	P	X
Data Expansion								
CVTBW		X ← A, $A_{6-0} \leftarrow A_7$	1					
Bit Manipulation								
MOV1	CY, saddr.bit	CY ← (saddr.bit)	3					X
	CY, sfr.bit	CY ← sfr.bit	3					X
	CY, A.bit	CY ← A.bit	2					X
	CY, X.bit	CY ← X.bit	2					X
	CY, PSWH.bit	CY ← PSWH.bit	2					X
	CY, PSWL.bit	CY ← PSWL.bit	2					X
	saddr.bit, CY	(saddr.bit) ← CY	3					
	sfr.bit, CY	sfr.bit ← CY	3					
	A.bit, CY	A.bit ← CY	2					
	X.bit, CY	X.bit ← CY	2					
	PSWH.bit, CY	PSWH.bit ← CY	2					
	PSWL.bit, CY	PSWL.bit ← CY	2	X	X	X	X	

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Bit Manipulation (cont)								
AND1	CY, saddr.bit	CY ← CY ∧ (saddr.bit)	3					X
	CY, /saddr.bit	CY ← CY ∧ $\overline{(saddr.bit)}$	3					X
	CY, sfr.bit	CY ← CY ∧ sfr.bit	3					X
	CY, /sfr.bit	CY ← CY ∧ $\overline{sfr.bit}$	3					X
	CY, A.bit	CY ← CY ∧ A.bit	2					X
	CY, /A.bit	CY ← CY ∧ $\overline{A.bit}$	2					X
	CY, X.bit	CY ← CY ∧ X.bit	2					X
	CY, /X.bit	CY ← CY ∧ $\overline{X.bit}$	2					X
	CY, PSWH.bit	CY ← CY ∧ PSWH.bit	2					X
	CY, /PSWH.bit	CY ← CY ∧ $\overline{PSWH.bit}$	2					X
	CY, PSWL.bit	CY ← CY ∧ PSWL.bit	2					X
	CY, /PSWL.bit	CY ← CY ∧ $\overline{PSWL.bit}$	2					X
OR1	CY, saddr.bit	CY ← CY ∨ (saddr.bit)	3					X
	CY, /saddr.bit	CY ← CY ∨ $\overline{(saddr.bit)}$	3					X
	CY, sfr.bit	CY ← CY ∨ sfr.bit	3					X
	CY, /sfr.bit	CY ← CY ∨ $\overline{sfr.bit}$	3					X
	CY, A.bit	CY ← CY ∨ A.bit	2					X
	CY, /A.bit	CY ← CY ∨ $\overline{A.bit}$	2					X
	CY, X.bit	CY ← CY ∨ X.bit	2					X
	CY, /X.bit	CY ← CY ∨ $\overline{X.bit}$	2					X
	CY, PSWH.bit	CY ← CY ∨ PSWH.bit	2					X
	CY, /PSWH.bit	CY ← CY ∨ $\overline{PSWH.bit}$	2					X
	CY, PSWL.bit	CY ← CY ∨ PSWL.bit	2					X
	CY, /PSWL.bit	CY ← CY ∨ $\overline{PSWL.bit}$	2					X
XOR1	CY, saddr.bit	CY ← CY ⊻ (saddr.bit)	3					X
	CY, sfr.bit	CY ← CY ⊻ sfr.bit	3					X
	CY, A.bit	CY ← CY ⊻ A.bit	2					X
	CY, X.bit	CY ← CY ⊻ X.bit	2					X
	CY, PSWH.bit	CY ← CY ⊻ PSWH.bit	2					X
	CY, PSWL.bit	CY ← CY ⊻ PSWL.bit	2					X
SET1	saddr.bit	(saddr.bit) ← 1	2					
	sfr.bit	sfr.bit ← 1	3					
	A.bit	A.bit ← 1	2					
	X.bit	X.bit ← 1	2					
	PSWH.bit	PSWH.bit ← 1	2					
	PSWL.bit	PSWL.bit ← 1	2	X	X	X	X	X

5b

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Bit Manipulation (cont)								
CLR1	saddr.bit	(saddr.bit) ← 0	2					
	sfr.bit	sfr.bit ← 0	3					
	A.bit	A.bit ← 0	2					
	X.bit	X.bit ← 0	2					
	PSWH.bit	PSWH.bit ← 0	2					
	PSWL.bit	PSWL.bit ← 0	2	X	X	X	X	X
NOT1	saddr.bit	(saddr.bit) ← $\overline{(saddr.bit)}$	3					
	sfr.bit	sfr.bit ← $\overline{sfr.bit}$	3					
	A.bit	A.bit ← $\overline{A.bit}$	2					
	X.bit	X.bit ← $\overline{X.bit}$	2					
	PSWH.bit	PSWH.bit ← $\overline{PSWH.bit}$	2					
	PSWL.bit	PSWL.bit ← $\overline{PSWL.bit}$	2	X	X	X	X	X
SET1	CY	CY ← 1	1					1
CLR1	CY	CY ← 0	1					0
NOT1	CY	CY ← $\overline{CY}$	1					X
Subroutine Linkage								
CALL	!addr16	$(SP-1) \leftarrow (PC + 3)_H$, $(SP - 2) \leftarrow (PC + 3)_L$, PC ← addr16, SP ← SP – 2	3					
	rp1	$(SP-1) \leftarrow (PC + 2)_H$, $(SP - 2) \leftarrow (PC + 2)_L$, $PC_H \leftarrow rp1_H$, $PC_L \leftarrow rp1_L$, SP ← SP – 2	2					
	[rp1]	$(SP-1) \leftarrow (PC + 2)_H$, $(SP - 2) \leftarrow (PC + 2)_L$, $PC_H \leftarrow (rp1 + 1)$, $PC_L \leftarrow (rp1)$, SP ← SP – 2	2					
CALLF	!addr11	$(SP-1) \leftarrow (PC + 2)_H$, $(SP - 2) \leftarrow (PC + 2)_L$, $PC_{15-11} \leftarrow 00001$, $PC_{10-0} \leftarrow$ addr11, SP ← SP – 2	2					
CALLT	[addr5]	$(SP-1) \leftarrow (PC + 1)_H$, $(SP - 2) \leftarrow (PC + 1)_L$, $PC_H \leftarrow$ (TPFx8000H + 2 x addr5 + 41H), $PC_L \leftarrow$ (TPFx8000H + 2 x addr5 + 40H), SP ← SP – 2	1					
BRK		$(SP-1) \leftarrow$ PSWH, $(SP - 2) \leftarrow$ PSWL, $(SP - 3) \leftarrow (PC + 1)_H$, $(SP - 4) \leftarrow (PC + 1)_L$, $PC_L \leftarrow$ (003EH), $PC_H \leftarrow$ (003FH), SP ← SP – 4, IE ← 0	1					
RET		$PC_L \leftarrow$ (SP), $PC_H \leftarrow$ (SP + 1), SP ← SP + 2	1					
RETB		$PC_L \leftarrow$ (SP), $PC_H \leftarrow$ (SP + 1), PSWL ← (SP + 2), PSWH ← (SP + 3), SP ← SP + 4	1	R	R	R	R	R
RETI		$PC_L \leftarrow$ (SP), $PC_H \leftarrow$ (SP + 1), PSWL ← (SP + 2), PSWH ← (SP + 3), SP ← SP + 4	1	R	R	R	R	R
Stack Manipulation								
PUSH	sfrp	$(SP - 1) \leftarrow sfr_H$, $(SP - 2) \leftarrow sfr_L$, SP ← SP – 2	3					
	post	{$(SP - 1) \leftarrow rpp_H$, $(SP - 2) \leftarrow rpp_L$, SP ← SP – 2} x n (Note 5)	2					
	PSW	(SP – 1) ← PSWH, (SP – 2) ← PSWL, SP ← SP – 2	1					

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Stack Manipulation (cont)								
PUSHU	post	{(UP – 1) ← rpp_H, (UP – 2) ← rpp_L, UP ← UP – 2} x n (Note 5)	2					
POP	sfrp	sfr_L ← (SP), sfr_H ← (SP + 1), SP ← SP + 2	3					
	post	{rpp_L ← (SP), rpp_H ← (SP + 1), SP ← SP + 2} x n (Note 5)	2					
	PSW	PSWL ← (SP), PSWH ← (SP + 1), SP ← SP + 2	1	R	R	R	R	R
POPU	post	{rpp_L ← (UP), rpp_H ← (UP + 1), UP ← UP + 2} x n (Note 5)	2					
MOVW	SP, #word	SP ← word	4					
	SP, AX	SP ← AX	2					
	AX, SP	AX ← SP	2					
INCW	SP	SP ← SP + 1	2					
DECW	SP	SP ← SP – 1	2					
Pin Level Test								
CHKL	sfr	(Pin level) $\forall$ (internal signal level)	3	X	X		P	
CHKLA	sfr	A ← (Pin level) $\forall$ (internal signal level)	3	X	X		P	
Unconditional Branch								
BR	!addr16	PC ← addr16	3					
	rp1	PC_H ← $rp1_H$, PC_L ← $rp1_L$	2					
	[rp1]	PC_H ← (rp1 + 1), PC_L ← (rp1)	2					
	$addr16	PC ← PC + 2 + jdisp8	2					
Conditional Branch								
BC, BL	$addr16	PC ← PC + 2 + jdisp8 if CY = 1	2					
BNC, BNL	$addr16	PC ← PC + 2 + jdisp8 if CY = 0	2					
BZ, BE	$addr16	PC ← PC + 2 + jdisp8 if Z = 1	2					
BNZ, BNE	$addr16	PC ← PC + 2 + jdisp8 if Z = 0	2					
BV, BPE	$addr16	PC ← PC + 2 + jdisp8 if P/V = 1	2					
BNV, BPO	$addr16	PC ← PC + 2 + jdisp8 if P/V = 0	2					
BN	$addr16	PC ← PC + 2 + jdisp8 if S = 1	2					
BP	$addr16	PC ← PC + 2 + jdisp8 if S = 0	2					
BGT	$addr16	PC ← PC + 3 + jdisp8 if (P/V $\forall$ S) $\vee$ Z = 0	3					
BGE	$addr16	PC ← PC + 3 + jdisp8 if P/V $\forall$ S = 0	3					
BLT	$addr16	PC ← PC + 3 + jdisp8 if P/V $\forall$ S = 1	3					
BLE	$addr16	PC ← PC + 3 + jdisp8 if (P/V $\forall$ S) $\vee$ Z = 1	3					
BH	$addr16	PC ← PC + 3 + jdisp8 if Z $\vee$ CY = 0	3					
BNH	$addr16	PC ← PC + 3 + jdisp8 if Z $\vee$ CY = 1	3					

5b

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Conditional Branch (cont)								
BT	saddr.bit, $addr16	PC ← PC + 3 + jdisp8 if (saddr.bit) = 1	3					
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 1	4					
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 1	3					
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 1	3					
	PSWH.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWH.bit = 1	3					
	PSWL.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWL.bit = 1	3					
BF	saddr.bit, $addr16	PC ← PC + 4 + jdisp8 if (saddr.bit) = 0	4					
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 0	4					
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 0	3					
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 0	3					
	PSWH.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWH.bit = 0	3					
	PSWL.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWL.bit = 0	3					
BTCLR	saddr.bit, $addr16	PC ← PC + 4 + jdisp8 if (saddr.bit) = 1 then reset (saddr.bit)	4					
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit	4					
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 1 then reset A.bit	3					
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 1 then reset X.bit	3					
	PSWH.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWH.bit = 1 then reset PSWH.bit	3					
	PSWL.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWL.bit = 1 then reset PSWL.bit	3	X	X	X	X	X
BFSET	saddr.bit, $addr16	PC ← PC + 4 + jdisp8 if (saddr.bit) = 0 then set (saddr.bit)	4					
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 0 then set sfr.bit	4					
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 0 then set A.bit	3					
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 0 then set X.bit	3					
	PSWH.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWH.bit = 0 then set PSWH.bit	3					
	PSWL.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWL.bit = 0 then set PSWL.bit	3	X	X	X	X	X
DBNZ	r2, $addr16	r2 ← r2 – 1, then PC ← PC + 2 + jdisp8 if (r2) ≠ 0	2					
	saddr, $addr16	(saddr) ← (saddr) – 1, then PC ← PC + 3 + jdisp8 if (saddr) ≠ 0	3					
Context Switching								
BRKCS	RBn	RBS_{2-0} ← n, PC_H ↔ R5, PC_L ↔ R4, R7 ← PSWH, R6 ← PSWL, RSS ← 0, IE ← 0	2					
RETCS	!addr16	PC_H ← R5, PC_L ← R4, R5 ← $addr16_H$, R4 ← $addr16_L$, PSWH ← R7, PSWL ← R6 (priority change)	3	R	R	R	R	R
RETCSB	!addr16	PC_H ← R5, PC_L ← R4, R5 ← $addr16_H$, R4 ← $addr16_L$, PSWH ← R7, PSWL ← R6 (no priority change)	4	R	R	R	R	R

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
String Manipulation								
MOVM	[DE+], A	(DE+) ← A, C ← C–1 End if C = 0	2					
	[DE–], A	(DE–) ← A, C ← C–1 End if C = 0	2					
MOVBK	[DE+], [HL+]	(DE+) ← (HL+), C ← C–1 End if C = 0	2					
	[DE–], [HL–]	(DE–) ← (HL–), C ← C–1 End if C = 0	2					
XCHM	[DE+], A	(DE+) ↔ A, C ← C–1 End if C = 0	2					
	[DE–], A	(DE–) ↔ A, C ← C–1 End if C = 0	2					
XCHBK	[DE+], [HL+]	(DE+) ↔ (HL+), C ← C–1 End if C = 0	2					
	[DE–], [HL–]	(DE–) ↔ (HL–), C ← C–1 End if C = 0	2					
CMPME	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X
CMPBKE	[DE+], [HL+1]	(DE+) – (HL+), C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X
CMPMNE	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X
CMPBKNE	[DE+], [HL+]	(DE+) – (HL+), C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X
CMPMC	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X
CMPBKC	[DE+], [HL+1]	(DE+) – (HL+), C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X
CMPMNC	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X
CMPBKNC	[DE+], [HL+]	(DE+) – (HL+), C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X
CPU Control								
MOV	STBC, #byte	STBC ← byte (Note 6)	4					
	WDM, #byte	WDM ← byte (Note 6)	4					
SWRS		RSS ← $\overline{RSS}$	1					
SEL	RBn	RBS_{2-0} ← n, RSS ← 0	2					
	RBn, ALT	RBS_{2-0} ← n, RSS ← 1	2					
NOP		No operation	1					
EI		IE ← 1 (Enable interrupt)	1					
DI		IE ← 0 (Disable interrupt)	1					

5b

Instruction Set (cont)

Notes:

(1) A special instruction is used to write to STBC and WDM.

(2) One byte move instruction when [DE], [HL], [DE+], [DE–], [HL+], or [HL–] is specified for mem.

(3) 16-bit signed multiply instruction

(4) Addressing range is 0FE00H to 0FEFFH.

(5) rpp refers to register pairs specified in post byte. "n" is the number of register pairs specified in post byte.

(6) Trap if data bytes in operation code are not one's complement. If trap, then:
(SP–1) ← PSWH, (SP–2) ← PSWL, (SP–3) ← $(PC–4)_H$,
(SP–4) ← $(PC–4)_L$, PC_L ← (003CH), PC_H ← (003DH).
SP ← SP–4, IE ← 0.

NEC
NEC Electronics Inc.

μPD78352 Family (μPD78350/352A/P352) 16-/8-Bit, K-Series Microcontrollers With Real-Time Output Ports

September 1993

Description

The μPD78350, μPD78352A, and μPD78P352 are members of the K-Series® of microcontrollers. These 16-/8-bit microcontrollers—with a minimum instruction time of 125 ns at 32 MHz (160 ns at 25 MHz for the μPD78350)—are designed for high-speed, real-time process control. They feature a 16-bit CPU, an 8-bit external data bus, eight banks of main registers, an advanced interrupt handling facility, and a powerful set of memory-mapped on-chip peripherals. A 16-bit multiply and accumulate instruction provides hardware convolution capability. On-board memory includes 640 bytes of RAM, 32K bytes of mask ROM in the μPD78352A, and 32K bytes of UV EPROM or one-time programmable (OTP) ROM in the μPD78P352.

The advanced interrupt handling facility has four levels of programmable hardware-priority control and three methods of servicing interrupt requests, including vectoring, hardware context switching, and macro service. The macro service facility reduces the CPU overhead involved in servicing peripheral interrupts by transferring data between the memory-mapped special function registers (SFRs) and memory without the use of time consuming interrupt service routines. In addition, the macro service facility can be used to perform certain CPU functions, such as event counting and math-oriented data alterations.

The combination of high-speed hardware convolution capability and context switching, eight register banks, and the macro service facility makes these devices ideal for applications in the hard-disk drive and tape drive markets as well as the automotive and industrial control/robotics markets.

K-Series is a registered trademark of NEC Electronics Inc.

Features

- Complete single-chip microcontroller
  - 16-bit ALU
  - 640 bytes of RAM
  - 32K bytes of mask ROM (μPD78352A)
  - 32K bytes of UV EPROM or OTP ROM (μPD78P352)
- Powerful instruction set
  - 16-bit unsigned and signed multiply
  - 16-bit unsigned divide
  - 16-bit multiply and accumulate instruction
  - 1-bit and 8-bit logic instructions
  - String instructions
- Minimum instruction time
  - 160 ns at 25 MHz (μPD78350)
  - 125 ns at 32 MHz (μPD78352A/P352)
- 5-byte instruction prefetch queue
- Memory expansion
  - 8085 bus-compatible
  - 64K-byte address space
- Large I/O capacity
  - Up to 30 I/O port lines (μPD78350)
  - Up to 50 I/O port lines (μPD78352A/P352)
- Memory-mapped, on-chip peripherals (special function registers)
- Timer/counter unit
  - 16-bit free-running timer:
    Two 16-bit capture registers;
    Two external interrupt/capture lines
  - 16-bit timer/event counter:
    One 16-bit compare register;
    One external event counter line
  - 16-bit interval timer:
    One 16-bit compare register
- Two 8-bit precision pulse-width modulated (PWM) output lines
- Programmable priority interrupt controller (four levels)
- Three methods of interrupt service
  - Vectored interrupts
  - Context switching with hardware register bank switch
  - Macro service mode with choice of five different functions
- Watchdog timer with dedicated output
- STOP and HALT standby functions
- Single 5-volt power supply

5c

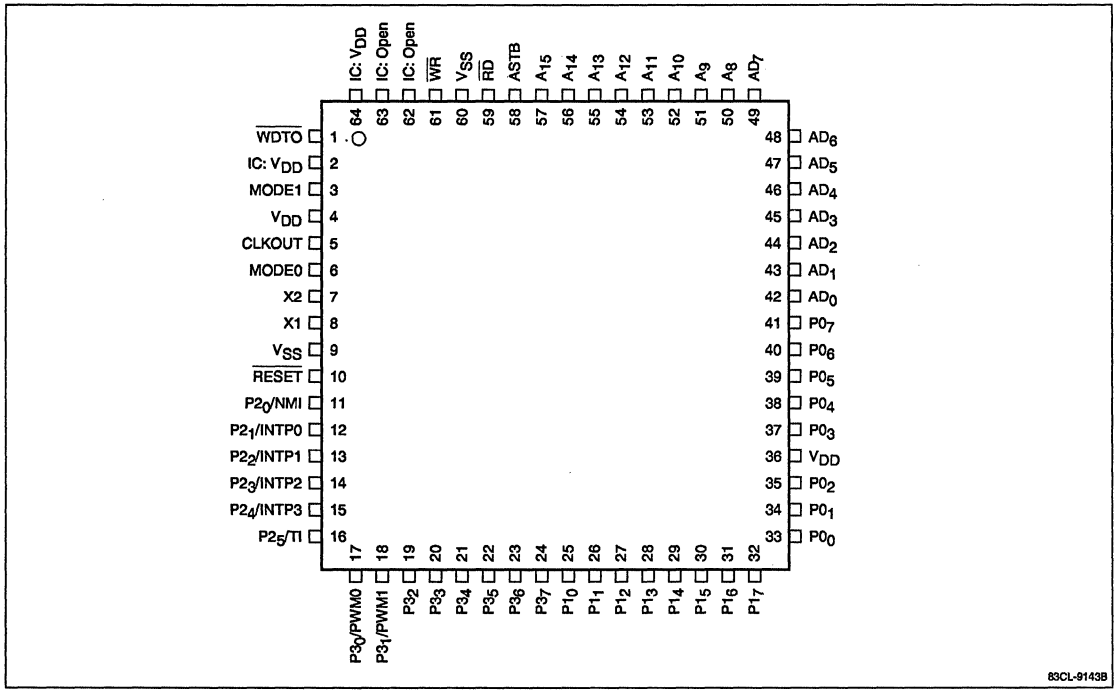
Ordering Information

Part Number	ROM	Package	Package Drawing
μPD78350GC-3BE	ROMless	64-pin plastic QFP (3.0-mm height)	P64GC-80-3BE
μPD78352AG-xxx-22	32K mask ROM	64-pin plastic QFP (1.7-mm height)	P64G-80-22-1
μPD78P352G-22	32K OTP ROM		
μPD78P352KK	32K UV EPROM	64-pin ceramic LCC with window	X80KW-80B

xxx indicates ROM code suffix.

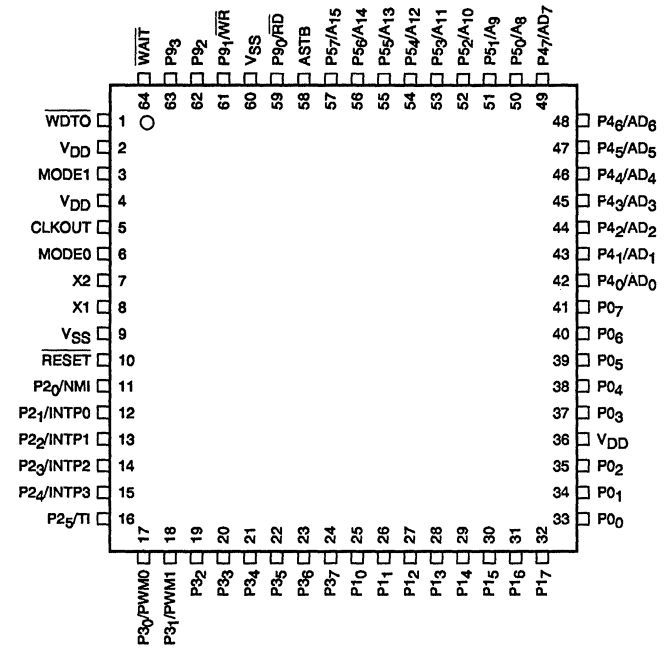
Pin Configurations

64-Pin Plastic QFP (μPD78350)



Pin Configurations (cont)

64-Pin Plastic QFP and Ceramic LCC (μPD78352A/P352)



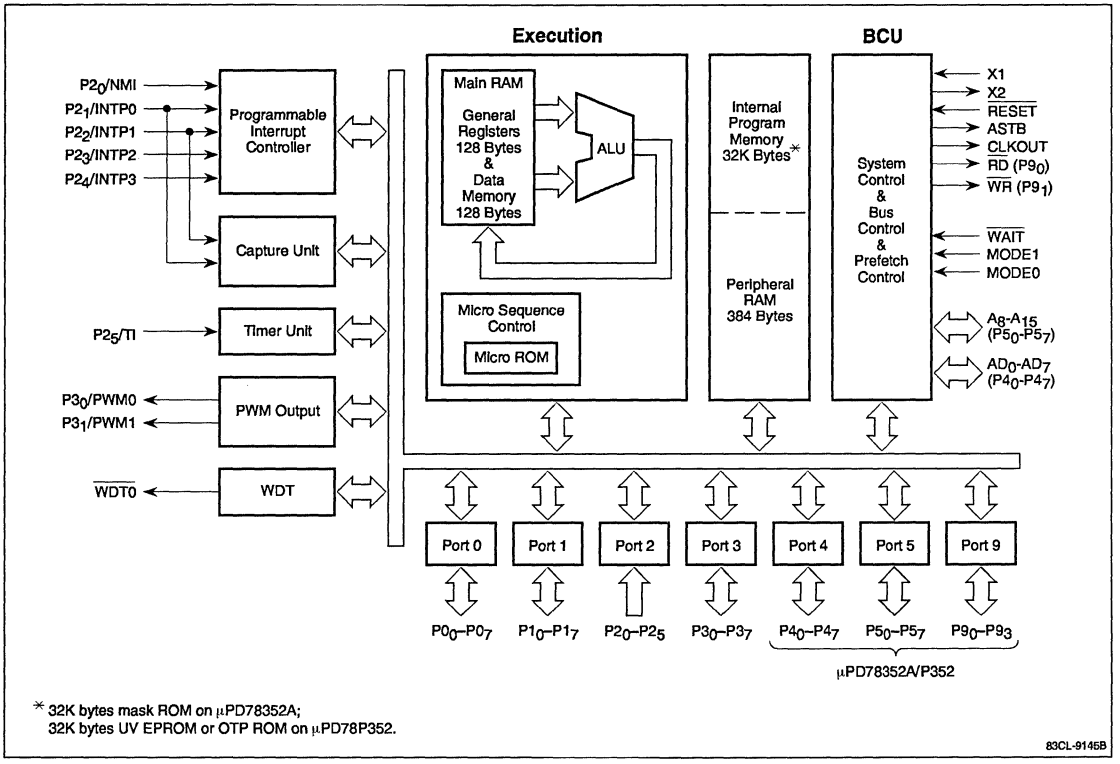
83CL-9144B

5c

Pin Functions; Normal Operating Mode

Symbol	Function	Alternate Symbol	Alternate Function
$P0_0$ - $P0_7$	Port 0; 8-bit, bit-selectable I/O port		
$P1_0$ - $P1_7$	Port 1; 8-bit, bit-selectable I/O port		
$P2_0$	Port 2; 6-bit input port	NMI	External nonmaskable interrupt
$P2_1$		INTP0	Maskable external interrupts
$P2_2$		INTP1	
$P2_3$		INTP2	
$P2_4$		INTP3	
$P2_5$		TI	External input for timer 1
$P3_0$	Port 3; 8-bit, bit-selectable I/O port	PWM0	Pulse-width modulated outputs
$P3_1$		PWM1	
$P3_2$ - $P3_7$			
$P4_0$ - $P4_7$	Port 4; byte-selectable I/O port (μPD78352A/P352)	AD_0 - AD_7	Low-order 8 bits of the multiplexed external address/data bus
$P5_0$ - $P5_7$	Port 5; bit-selectable I/O port (μPD78352A/P352)	A_8 - A_{15}	High-order 8 bits of the external address bus
$P9_0$	Port 9; 4-bit, bit-selectable I/O port (μPD78352A/P352). For 78350, $P9_0$ functions as $\overline{RD}$ and $P9_1$ functions as $\overline{WR}$ signals only. $P9_2$ and $P9_3$ are not provided for 78350.	$\overline{RD}$	External read strobe output
$P9_1$		$\overline{WR}$	External write strobe output
$P9_2$		IC	Internally connected; must be left open (μPD78350).
$P9_3$		IC	
ASTB	Address strobe output; used to latch address for external memory.		
CLKOUT	Output of the system clock		
IC	Internally connected; must be left open.		
MODE0	Connect to V_{DD} for μPD78350 and μPD78P352 in programing mode. Connect to V_{SS} for normal operation of μPD78352A/P352. The level of this pin cannot be changed during normal operation.		
MODE1	Always connect to V_{SS}. The level of this pin cannot be changed during normal operation.		
$\overline{RESET}$	External system reset input		
$\overline{WAIT}$	A low-level input adds wait states to the external bus cycle; used by very-slow memory and/or peripherals (only for 78352A/P352).		
$\overline{WDTO}$	Open-drain output from the watchdog timer		
X1	Crystal connection or external clock input		
X2	Crystal connection or open for external clock		
V_{DD}	+5-volt power input		
V_{SS}	Ground		

Block Diagram



* 32K bytes mask ROM on μPD78352A;
32K bytes UV EPROM or OTP ROM on μPD78P352.

83CL-9145B

5c

FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processing unit (CPU) of the μPD78352 family features 16-bit arithmetic including 16 x 16-bit multiply, both unsigned and signed, and 32 x 16-bit unsigned divide (producing a 32-bit quotient and a 16-bit remainder). The signed multiply executes in 1.12 μs and the divide in 3.44 μs at 25 MHz (0.875 and 2.69 μs, respectively, for μPD78352A/P352 at 32 MHz).

Also, a multiply-and-accumulate instruction, "MACW n," performs a signed multiply on factors from a pair of tables and sums the results in the 32-bit register AXDE. The total execution time for 10 terms is 17.2 μs at 25 MHz for the μPD78350 and 13.44 μs at 32 MHz for the μPD78352A/P352.

A CALLT vector table and a CALLF area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction can access up to 32 subroutines through their addresses in the CALLT vector table. A 2-byte call instruction can access any routine beginning in a specific CALLF area.

The internal system clock (f_{CLK}) is generated by dividing the oscillator frequency by 2. Therefore, at the maximum oscillator frequency of 25 MHz for the μPD78350, the clock is 12.5 MHz. Since instructions execute in two or more cycles, the minimum instruction time is 160 ns. For the μPD78352A/P352 running at 32 MHz, the clock is 16 MHz and the minimum instruction time is 125 ns.

Internal RAM

The μPD78352 family has total of 640 bytes of internal RAM. The upper 256-byte area (FE00H-FEFFH) features high-speed access of one or two internal system clocks per word of data depending on the addressing mode and is known as "Main RAM." The remainder (FC80H-FDFFH) is accessed at the same speed as external memory (1 byte per three internal system clocks) and is known as "Peripheral RAM." The general register banks and the macro service control words are stored in Main RAM. The remainder of Main RAM and any unused register bank locations are available for general storage.

Main RAM Access Speed

Access Mode	Internal System Clocks (f_{CLK})
Memory access	2
Saddr access	1
Register access	1

Internal Program Memory

The μPD78352A contains 32K bytes of mask ROM; μPD78P352 contains 32K bytes of UV EPROM or one-time programmable ROM. Instructions are fetched from this program memory at a maximum rate of 1 byte every two internal system clocks. The μPD78350 does not have internal program memory.

External Memory

The μPD78352 family has a 64K-byte address space. The μPD78352A/P352 can access 0, 256, 4K, 16K, or 32K bytes of external memory in the area from 8000H to FDFFH. External memory can be either ROM, RAM, or peripheral as required. The μPD78352A/P352 has an 8-bit wide external data bus and a 16-bit wide external address bus. The low-order 8 bits of the address bus are multiplexed to provide the 8-bit data bus at I/O port 4.

High-order address bits are taken from port 5 as required. Address latch, read, and write strobes are also provided. In the μPD78352A/P352, the memory mode register (MM) controls the size of the external memory It can be programmed to use 0, 4, 6, or 8 bits from port 5 for the high-order address. Any remaining port 5 bits can be used for I/O.

The μPD78350 does not have ports 4 and 5. It has eight dedicated high-order address lines and eight dedicated address/data lines. All memory below address FC80H must be external, and the MM register is not used.

The programmable wait control register (PWC) allows the programmer to specify one or two additional wait states if they are required for slow-speed memory or external peripheral devices. These wait states for internal and external memory are specified independently in 16K-blocks. If additional wait states are required, an external $\overline{WAIT}$ pin is provided.

In addition, by using the AW0 and AW1 bits of the PWC register, the width of the ASTB signal can be increased by one cycle to allow more precharge time for dynamic RAMs or more address decoding time. This address wait signal can be enabled in 32K-byte blocks. See figure 1.

Figure 1. Programmed Wait Control Register

15							8	7							0
AW1	AW0	0	0	0	0	0	0	PWC7	PWC6	PWC5	PWC4	PWC3	PWC2	PWC1	PWC0

16K Memory Block	Wait Control Register Bits		Wait States	Data Access Clocks	Fetch Cycle Mode	Fetch Clocks
0000H-3FFFH	PWC1, PWC0	0 0	0	3	Normal	3
		0 1	1	4		4
		1 0	2	5		5
		1 1	0	4	High-speed	2
4000H-7FFFH	PWC3, PWC2	0 0	0	3	Normal	3
		0 1	1	4		4
		1 0	2	5		5
		1 1	0	4	High-speed	2
8000H-BFFFH	PWC5, PWC4	0 0	0	3	Normal	3
		0 1	1	4		4
		1 0	2	5		5
		1 1	—	N/A	N/A	N/A
C000H-FC7FH, *FFD0H-FFDFH	PWC7, PWC6	0 0	0	3	Normal	3
		0 1	1	4		4
		1 0	2	5		5
		1 1	—	N/A	N/A	N/A
FC80H-FDFFH	PWC7, PWC6	0 0	0	3	Normal	3
		0 1	0	3		3
		1 0	0	3		3
		1 1	—	N/A	N/A	N/A

* Data in the SFR external access area, FFD0H-FFDFH, cannot be fetched.

32K Memory Block	Wait Control Register Bits		Address Wait
0000H-7FFFH	AW0	0	Disabled
		1	Enabled
8000H-FC7FH	AW1	0	Disabled
		1	Enabled

5c

Program Fetch

The μPD78352 family devices allow opcode fetch in the area between 0000H and FDFFH; they contain a 5-byte instruction prefetch queue.The bus control unit can fetch an instruction byte from memory during cycles in which the execution unit is not using the memory bus. If the instruction byte is fetched from on-chip memory, two internal system clocks are required for each byte, and the queue can hold 5 bytes. If the instruction is fetched from external memory, three internal system clocks are required for each byte, and the queue can hold 3 bytes. For programs located in internal memory, the PWC register also can be programmed to allow 1 byte to be fetched every two, three, four, or five internal system clocks.

CPU Control Registers

Program Counter. The program counter is a 16-bit register that holds the address of the next instruction to be executed. After reset line goes high, the program counter is loaded with the address stored in locations 0000H and 0001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

CPU Control Word. The CPU control word (CCW) selects the origin of the interrupt vector and CALLT tables. If the TPF bit (bit 1) is zero, the origin is 0000H; if the TPF bit is one, the origin is 8000H. The CCW is a special function register located at address FFC1H. The addresses of the vectors for the $\overline{\text{RESET}}$ input, operation-code trap, and BRK instruction are fixed at 0000H, 003CH, and 003EH, respectively, and are not altered by the TPF bit.

Program Status Word. The program status word (PSW) is a 16-bit register containing flags that are set or reset depending on the results of an instruction. This register can be written to or read from 8 bits at a time. The high-order 8 bits are called the PSWH and the low-order 8 bits are called the PSWL. The individual flags can also be manipulated on a bit-by-bit basis. The assignment of PSW bits follows.

	7	6	5	4	3	2	1	0
PSWH	UF	RBS2	RBS1	RBS0	0	0	0	0

	7	6	5	4	3	2	1	0
PSWL	S	Z	RSS	AC	IE	P/V	0	CY

UF	User flag
RBS2-RBS0	Active register bank number
S	Sign flag (1 if last result was negative)
Z	Zero flag (1 if last result was zero)
RSS	Register set selection flag
AC	Auxiliary carry flag (carry out of 3 bit)
IE	Interrupt enable flag
P/V	Parity or arithmetic overflow flag
CY	Carry bit (or 1-bit accumulator for logic)

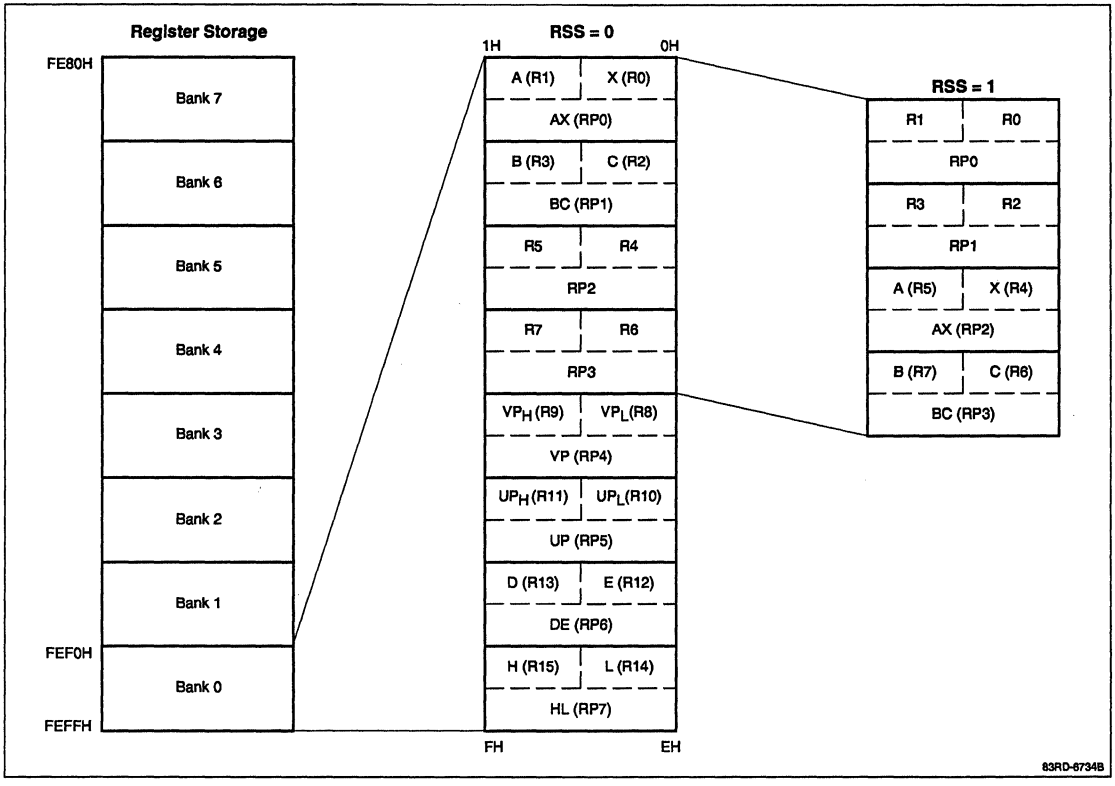
General Registers

There are sixteen 8-bit general registers, which can also be paired to function as 16-bit registers. A complete set of 16 registers is mapped into each of eight program-selectable register banks stored in Main RAM. Three bits in the PSW specify the active register bank.

Registers have functional names (like A, X, B, C for 8-bit registers and AX, BC for 16-bit registers) and absolute names (like R1, R0, R3, R2 for 8-bit registers and RP0, RP1 for 16-bit registers). Each instruction determines whether a register is referred to by functional or absolute name and whether it is 8 or 16 bits.

Two possible relationships may exist between the absolute and functional names of the first four register pairs. The RSS bit in the PSW determines which of these is active at any time. The effect is that the accumulator and counter registers can be saved, and a new set can be specified by toggling the RSS bit. Figure 2 illustrates the general register configuration.

Figure 2. General Registers



Addressing

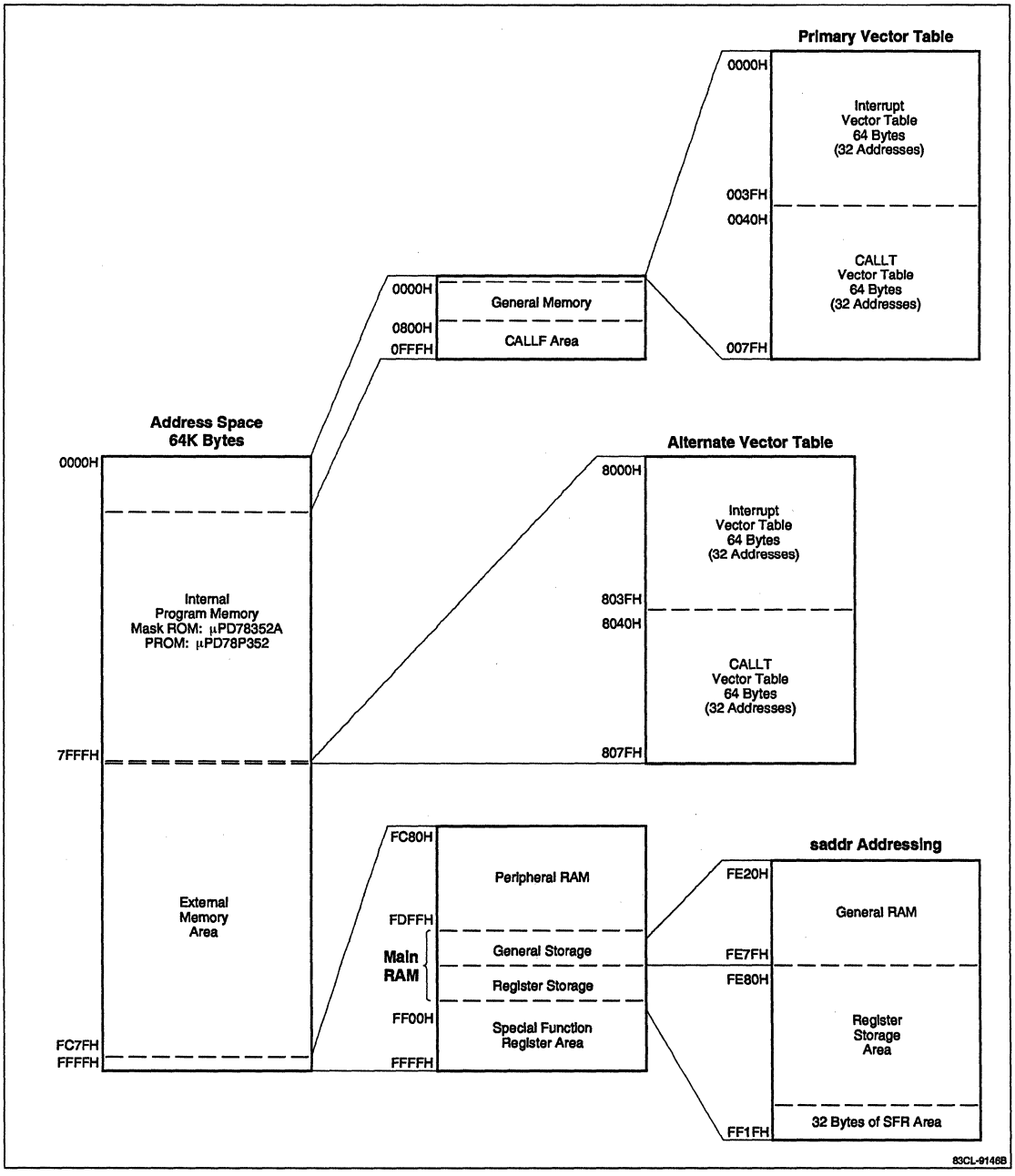
The μPD78352 family features 1-byte addressing of both the special function registers and the portion of on-chip RAM from FE20H to FEFFH. The 1-byte sfr addressing accesses the entire SFR area, while the 1-byte saddr addressing accesses the first 32 bytes of the SFR area and 224 bytes of the Main RAM.

The 16-bit SFRs and words of memory in these areas can be addressed by 1-byte saddrp addressing, which is valid for even addresses only. Since many instructions use 1-byte addressing, access to these locations is almost as fast and as versatile as access to the general registers.

There are nine addressing modes for data in main memory: direct, register, register indirect with autoincrement or autodecrement, saddr, saddr indirect, SFR, based, indexed, and based indexed. There are also 8-bit and 16-bit immediate operands. Figure 3 is the memory map of the μPD78352 family.

5c

Figure 3. Memory Map



Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and the CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by 1-byte SFR addressing. All can be read under program control, and most can also be written. They are either 8 or 16 bits, as required, and many of the 8-bit registers are capable of single-bit access as well.

Locations FFD0H through FFDFH are known as the external access area. Registers in external circuitry, interfaced and mapped to these addresses, can be addressed with SFR addressing. Table 1 lists the special function registers.

Table 1. Special Function Registers

				Access Units (Bits)			
Address	**Register (SFR)**	**Symbol**	**R/W**	**1**	**8**	**16**	**State After Reset**
FF00H	Port 0	P0	R/W	x	x	—	Undefined
FF01H	Port 1	P1	R/W	x	x	—	Undefined
FF02H	Port 2	P2	R	x	x	—	Undefined
FF03H	Port 3	P3	R/W	x	x	—	Undefined
FF04H	Port 4 (Note 1)	P4	R/W	x	x	—	Undefined
FF05H	Port 5 (Note 1)	P5	R/W	x	x	—	Undefined
FF09H	Port 9 (Note 1)	P9	R/W	x	x	—	Undefined
FF10H-FF11H	Compare register 00	CT00	R/W	—	—	x	Undefined
FF12H-FF13H	Compare register 01	CT01	R/W	—	—	x	Undefined
FF14H-FF15H	Compare register 10	CM10	R/W	—	—	x	Undefined
FF1EH-FF1FH	Compare register 20	CM20	R/W	—	—	x	Undefined
FF20H	Port 0 mode register	PM0	R/W	x	x	—	FFH
FF21H	Port 1 mode register	PM1	R/W	x	x	—	FFH
FF23H	Port 3 mode register	PM3	R/W	x	x	—	FFH
FF25H	Port 5 mode register (Note 1)	PM5	R/W	x	x	—	FFH
FF29H	Port 9 mode register (Note 1)	PM9	R/W	x	x	—	xFH
FF30H-FF31H	Timer register 0	TM0	R	—	—	x	00H
FF32H-FF33H	Timer register 1	TM1	R	—	—	x	00H
FF34H-FF35H	Timer register 2	TM2	R	—	—	x	00H
FF38H	Timer control register 0	TMC0	R/W	x	x	—	00H
FF39H	Timer control register 1	TMC1	R/W	x	x	—	00H
FF3CH	External interrupt mode register 0	INTM0	R/W	x	x	—	00H
FF3DH	External interrupt mode register 1	INTM1	R/W	x	x	—	00H
FF43H	Port 3 mode control register 0	PMC3	R/W	x	x	—	00H
FF62H	Port read control register	PRDC	R/W	x	x	—	00H
FF64H	PWM control register	PWMC	R/W	x	x	—	00H
FF66H	PWM buffer register 0	PWM0	R/W	x	x	—	Undefined
FF6EH	PWM buffer register 1	PWM1	R/W	x	x	—	Undefined
FFA8H	In-service priority register	ISPR	R	x	x	—	00H
FFAAH	Interrupt mode control register	IMC	R/W	x	x	—	80H
FFACH	Interrupt mask flag register	MKL	R/W	x	x	—	7FH

5c

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FFACH-FFADH	Interrupt mask flag register (Note 2)	MK	R/W	—	—	x	xx7FH
FFCOH	Standby control register (Note 3)	STBC	R/W	—	x	—	0000 x000B
FFC1H	CPU control word	CCW	R/W	x	x	—	00H
FFC2H	Watchdog timer mode register (Note 3)	WDM	R/W	—	x	—	00H
FFC4H	Memory expansion mode register	MM	R/W	x	x	—	00H
FFC6H-FFC7H	Programmable wait control register	PWC	R/W	—	—	x	C0AAH
FFD0H-FFDFH	External access area	—	R/W	x	x	—	Undefined
FFE0H	Interrupt control register (INTOV)	OVIC	R/W	x	x	—	43H
FFE1H	Interrupt control register (INTP0)	PIC0	R/W	x	x	—	43H
FFE2H	Interrupt control register (INTP1)	PIC1	R/W	x	x	—	43H
FFE3H	Interrupt control register (INTCM10)	CMIC10	R/W	x	x	—	43H
FFE4H	Interrupt control register (INTCM20)	CMIC20	R/W	x	x	—	43H
FFE5H	Interrupt control register (INTP2)	PIC2	R/W	x	x	—	43H
FFE6H	Interrupt control register (INTP3)	PIC3	R/W	x	x	—	43H

Notes:

(1) μPD78352A/P352 only.

(2) Used only when a word is accessed by an instruction with the sfrp operand.

(3) These are protected registers, which can be written by a special instruction only.

Input/Output Ports

The μPD78350 has four I/O ports providing a total of 30 I/O lines. The μPD78352A/P352 have an additional three I/O ports for a total of 50 I/O lines.

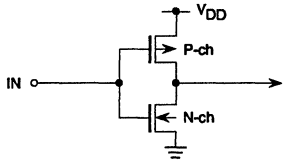
Ports P0, P1, and P3 are 8-bit input/output ports and P2 is a 6-bit input port. All the bits in P0, P1, and P3 can be individually selected for either input or output using port mode registers PM0, PM1, and PM3. Bits $P3_0$ and $P3_1$ can also be programmed for use as PWM outputs PWM0 and PWM1 by using port 3 mode control register PMC3.

Port P2 functions only in the control mode as input pins for the NMI signal, the INTP0 to INTP3 interrupt signals, and the external count clock for timer 1 (TI). However, any masked interrupt automatically becomes an input line and the state of all the pins can be read by the program using a read instruction to port 2. Each pin of P2 can be programmed for rising, falling, or both rising and falling edge detection.

The output level of the P0, P1, and P3 I/O pins can be tested to determine whether they agree with the contents of the output latch. When the low-order bit of port read control register PRDC is set to 1, the output level of the I/O pins can be read with the port still in the output mode. These data values can be compared with the data known to be in the output latch to determine if the port is functioning correctly. Figure 4 shows the structure of each port pin.

Figure 4. I/O Circuits

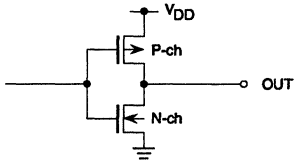
Type 1. $\overline{WAIT}$, MODE0, MODE1



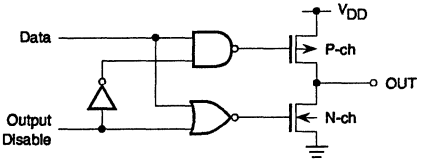
Type 2. Port P2, $\overline{RESET}$

Schmitt trigger input with hysteresis characteristics.

Type 3. CLKOUT

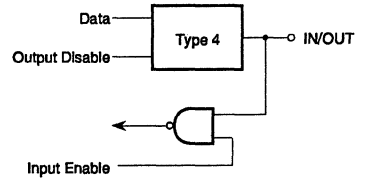


Type 4. Port P9, $\overline{RD}$, $\overline{WR}$, ASTB

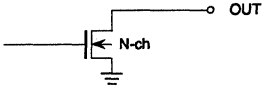


With both P-ch and N-ch off, the output is high impedance.

Type 5. Ports P0, P1, P3, P4, P5; AD_0-AD_7; A_8-A_{15}



Type 14. $\overline{WDTO}$



83CL-9147B

5c

The three additional input/output ports in the μPD78352A/P352 are ports P4, P5, and P9. These ports are available if external memory or memory-mapped external circuitry is not being used. Port 4 is shared with the low-order address/data bus (AD_0 to AD_7) and is byte-selectable for input or output. Port 5 is shared with the high-order address bus (A_8 to A_{15}). Depending on the amount of external memory used, either 8, 6, 4, or 0 bits are available for bit-selectable I/O. Port 9 is a 4-bit, bit-selectable I/O port. Two of its pins are shared with the read and write strobes.

Timers

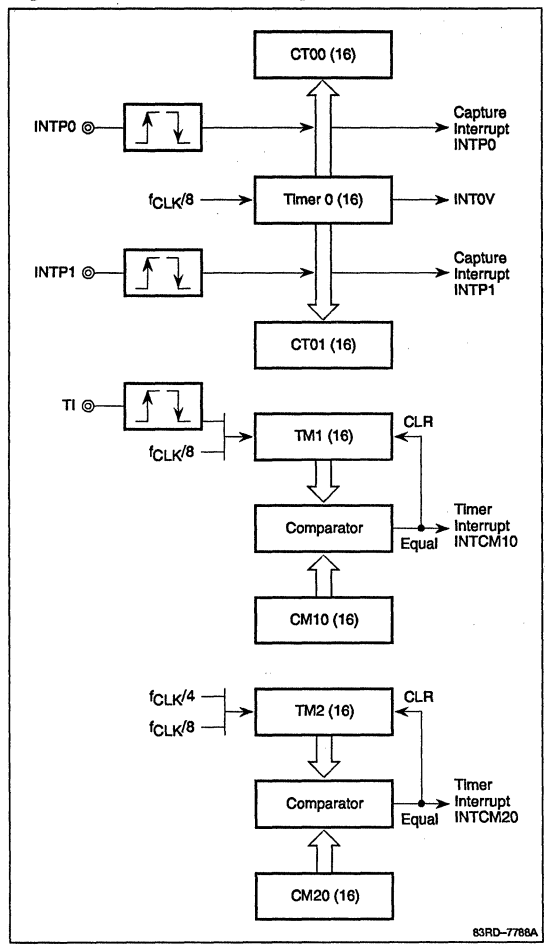
The μPD78352 family has three 16-bit timers. Two of them count only the internal system clock; the third counts either the internal system clock or external events. Refer to the block diagram, figure 5.

Timer 0 is a 16-bit, free-running counter that counts the internal system clock ($f_{CLK}/8$) and generates an interrupt request (INTOV) when it overflows. It also has two associated capture registers, CT00 and CT01. The timer value can be captured in synchronization with external interrupt lines INTP0 and INTP1, respectively. These lines can be programmed to trigger interrupts as well.

Timer 1 is a 16-bit counter serving as an interval timer or an event counter. It can count either the internal system clock ($f_{CLK}/8$) or external events sensed on the TI line. It has an associated comparator register, CM10. When the counter contents match the CM10 contents, the counter is cleared to 0, and an interrupt request (INTCM10) is generated. The counter continues to count until disabled by software.

Timer 2 is a 16-bit counter that serves as an interval timer. It can be programmed to count the internal system clock ($f_{CLK}/4$ or $f_{CLK}/8$). It also has an associated comparator register, CM20. When the counter contents match the CM20 contents, the counter is cleared to 0 and an interrupt request (INTCM20) is generated. The counter continues to count until disabled by software.

Figure 5. Timers Block Diagram

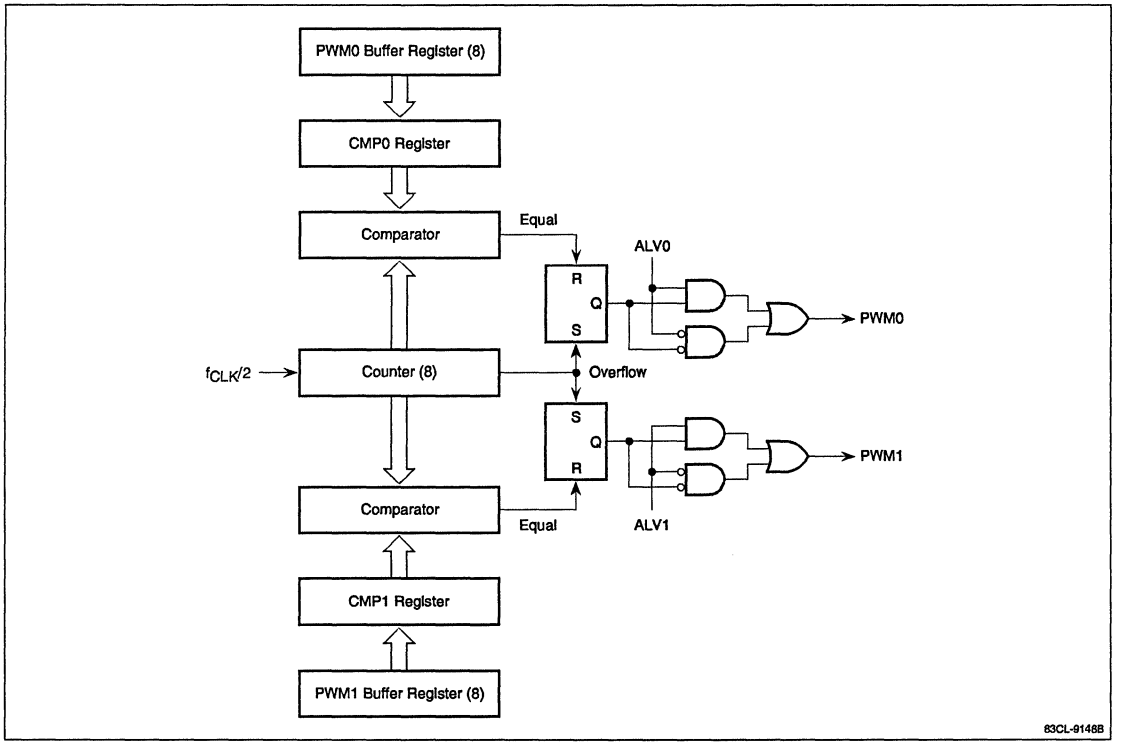


Pulse-Width Modulated Outputs

The μPD78352 family has two high-speed, pulse-width modulated (PWM) outputs. A single 8-bit, free-running counter counts the internal system clock $f_{CLK}/2$ and serves both outputs. For the μPD78350 running at 25 MHz (f_{CLK} = 12.5 MHz), the resolution is 160 ns and the repetition rate is 24.4 kHz. For the μPD78352A/P352 running at 32 MHz (f_{CLK} = 16 MHz), the resolution is 125 ns and the repetition rate is 31.25 kHz.

The polarity of each output can be selected under program control. Whenever the counter overflows, the CMP0 and CMP1 registers are loaded from their respective PWM buffer registers and each output becomes active. When the counter value matches the value in the associated compare register, that output goes inactive. The two PWM outputs, PWM0 and PWM1, share pins with port 3 bits 0 and 1, respectively.

Figure 6. Pulse-Width Modulated Outputs



5c

Interrupts

The μPD78352 family has seven maskable hardware interrupt sources: four external and three internal. The four external maskable interrupts share pins with port 2. Two of them, INTP0 and INTP1, can also be used to trigger capture events in registers CT00 and CT01 associated with timer 0. In addition, there are two nonmaskable interrupts, three software interrupts, and reset. The software interrupts, generated by the BRK or BRKCS instruction and the operation code trap, are not maskable. See table 2.

Table 2. Interrupt Sources

Type of Request	Default Priority	Signal Name	Source	Location	Macro Service Control Word	Vector Address TPF = 0	Vector Address TPF = 1
Software	—	—	Operation code trap	CPU	—	003CH	
	—	—	BRK instruction	CPU	—	003EH	
	—	—	BRKCS instruction (Note 1)	CPU	—	—	
Nonmaskable	—	NMI	NMI input pin	External	—	0002H	8002H
	—	INTWDT	Watchdog timer overflow	Internal	—	0004H	8004H
Maskable	0	INTOV	Timer 0 overflow	Internal	FE06H	0006H	8006H
	1	INTP0	INTP0 pin	External	FE08H	0008H	8008H
	2	INTP1	INTP1 pin	External	FE0AH	000AH	800AH
	3	INTCM10	CM10 coincidence	Internal	FE0CH	000CH	800CH
	4	INTCM20	CM20 coincidence	Internal	FE0EH	000EH	800EH
	5	INTP2	INTP2 pin	External	FE10H	0010H	8010H
	6	INTP3	INTP3 pin	External	FE12H	0012H	8012H
Reset	—	$\overline{\text{RESET}}$	$\overline{\text{RESET}}$ pin	External	—	0000H	

Note:

(1) Initiates context switch

Interrupt Servicing

The μPD78352 family provides four levels of programmable hardware priority control and three different methods of handling maskable interrupt requests: standard vectoring, context switching, and macro service. The programmer can choose the priority and mode of servicing each maskable interrupt by using the interrupt control registers.

Interrupt Control Registers

The μPD78352 family has 10 interrupt control registers. Each maskable interrupt request has its own control register, which includes bits to specify interrupt request, interrupt mask, macro service enable, context switch enable, and priority. Priorities range from 0 (highest) to 3. See figure 7.

There is also a mask flag register, MKL, with a bit for each maskable interrupt. Since each interrupt has two mask bits, the masking of the interrupt is the "or" function of those two bits.

Interrupt mode control register IMC can be used to enable or disable nesting of interrupts set to the lowest priority level (level 3). Inservice priority register ISPR is used by the hardware to hold the priority level of the interrupt request currently being serviced. It is manipulated by hardware only, but it can be read by software.

Finally, the IE bit of the program status word also is used to control the interrupts. If the IE bit is 0, all maskable interrupts, but not macro service, are disabled. The IE bit can be set or cleared by the EI or DI instruction, respectively, or by direct writing to the PSW. The IE bit is cleared each time an interrupt is accepted.

Figure 7. Interrupt Control Register (xxICx)

7	6	5	4
xxIFxx	xxMKxx	xxISMxx	xxCSExx
3	2	1	0
0	0	xxPRx1	xxPRx0

xxIFxx		Interrupt Request Flag
0		No interrupt request
1		Interrupt request received
xxMKxx		**Interrupt Mask Flag**
0		Interrupt request enabled
1		Interrupt will be pending
xxISMxx		**Macro Service Enable**
0		Software service
1		Macro service
xxCSExx		**Context Switch Enable**
0		Vector service
1		Context switch
xxPRx1	**xxPRx0**	**Priority Specification**
0	0	Priority 0 (highest)
0	1	Priority 1
1	0	Priority 2
1	1	Priority 3

Interrupt Priority

The two nonmaskable interrupts, NMI and INTWDT, have priority over all others. Their priority relative to each other is under program control.

Four hardware-controlled priority levels are available for the maskable interrupts. Any one of the four levels can be assigned by software to each of the maskable interrupt lines. Interrupt requests of a priority higher that the processor's current priority level are accepted; requests of the same or lower priority are held pending until the processor's priority state is lowered by a return instruction from the current service routine.

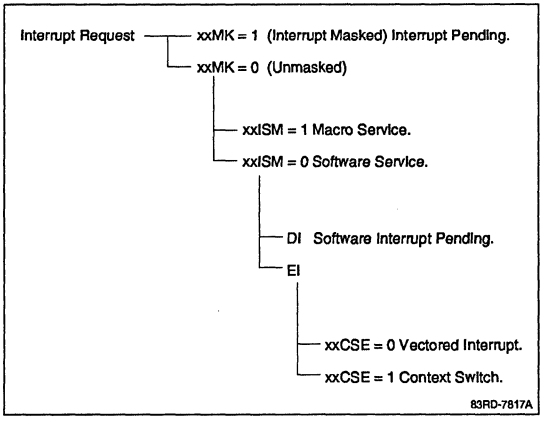
By setting the PRSL bit of the IMC register to zero, it is possible to specify in software that level 3 interrupts (the lowest level) can be accepted when the processor is operating at level 3. This nesting within a level applies to level 3 only.

Interrupt requests programmed to be handled by macro service have priority over all software interrupt service regardless of the assigned priority level, and macro service requests are accepted even when the interrupt enable bit in the PSW is set to the disable state. See figure 8.

The "Default Priorities" listed in table 2 are fixed by hardware; they are effective only when it is necessary to choose between two interrupt requests of the same software-assigned priority. For example, the default priorities would be used after the completion of a high-priority routine if two interrupts of the same lower priority were pending.

Software interrupts, the BRK and BRKCS instructions, and the operation code trap are executed regardless of the processor's priority level and the state of the IE bit. They do not alter the processor's priority level.

Figure 8. Interrupt Service Sequence



Vectored Interrupt

When vectored interrupt is specified for a given interrupt request, (1) the program status word and the program counter are saved on the stack, (2) the processor's priority is raised to that specified for the interrupt, (3) the IE bit in the PSW is set to zero, and (4) the routine whose address is in the interrupt vector table is entered. At completion of the service routine, the RETI instruction (or RETB instruction for software interrupts) reverses the process, and the μPD78352 family device resumes the interrupted routine.

Context Switch

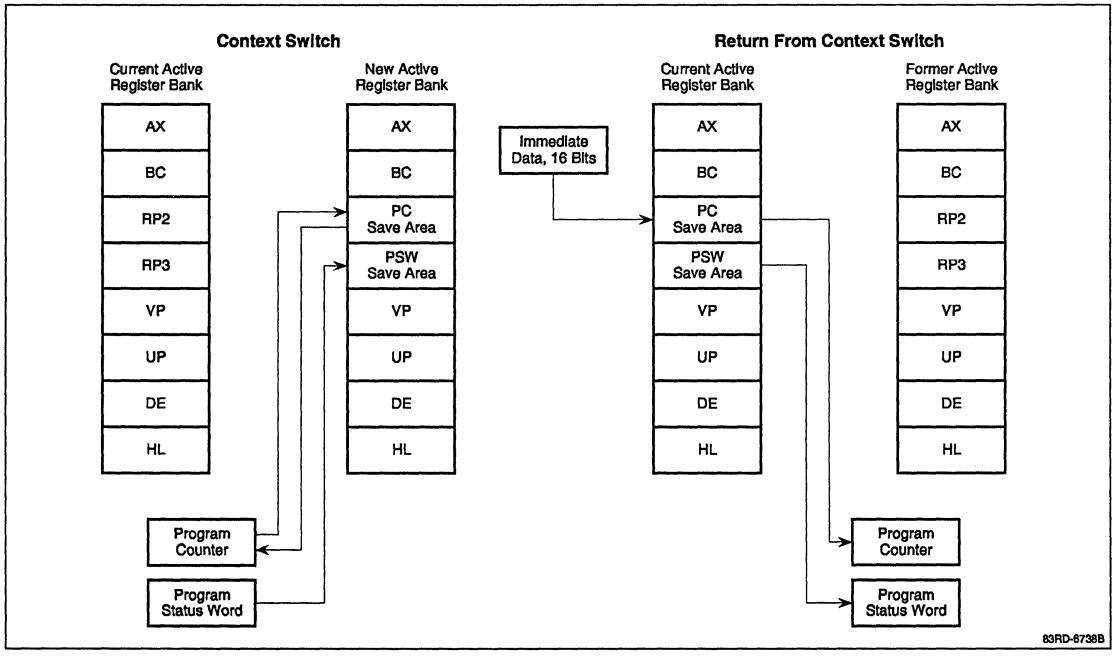
When context switching (figure 9) is specified for a given interrupt, the active register bank is changed to the register bank specified by the three low-order bits of the word in the interrupt vector table. The program counter is loaded from RP2 of the new register bank,

5c

the old program counter and program status word are saved in RP2 and RP3 of the new register bank, and the IE bit in the PSW is set to zero.

At completion of the service routine, the RETCS instruction for routines entered from hardware requests, or the RETCSB instruction for routines entered from the BRKCS instruction, reverse the process. The old program counter and program status word are restored from RP2 and RP3 of the new register bank. The entry address of the service routine, which must be specified in the 16-bit immediate operand of these return instructions, is stored again in RP2.

Figure 9. Context Switching and Return



Macro Service

When macro service is specified for a given interrupt, the macro service hardware temporarily stops the executing program and transfers data between the special function register area and the memory space. Control is then returned to the executing program, providing a completely transparent method of interrupt service. Macro service significantly improves response time and makes it unnecessary to save any registers.

For each request on the interrupt line, one operation is performed, and an 8-bit counter is decremented. When the counter reaches 0, a software service routine is entered according to its specified priority. Either vectored interrupt or context switch can be specified for entry to this routine, which is known as the macro service completion routine.

Macro service is provided for all of the maskable interrupt requests, and each has a specific macro service control word stored in Main RAM. The function to be performed is specified in the control word.

The μPD78352 family provides five different macro service functions.

Function	Description
EVTCNT	Event counter. Counts up to 256 events by incrementing or decrementing the macro service counter. When the counter reaches 00H, the software service routine is entered.
BLKTRS	Block transfer. Transfers a byte or word of data in either direction between a specified special function register and a buffer in Main RAM (FExx).
BLKTRS-P	Block transfer with memory pointer. Transfers a byte or word of data in either direction between a specified special function register and a buffer anywhere in the 64K-byte address space.
DTADIF	Data difference. Stores the difference between the current value of a specified 16-bit special function register and its previous value in a word buffer in Main RAM (FExx).
DTADIF-P	Data difference with memory pointer. Stores the difference between the current value of a specified 16-bit special function register and its previous value in a word buffer anywhere in the 64K-byte address space.

Standby Modes

The standby modes, HALT and STOP, reduce power consumption when CPU action is not required. In HALT mode, the CPU is stopped but the system clock continues to run. The HALT mode is released by any unmasked interrupt, an external NMI, or an external reset pulse. In STOP mode, both the CPU and the system clock are stopped, further minimizing power consumption. The STOP mode is released by either an external reset pulse or an external NMI.

The HALT and STOP modes are entered by programming the standby control register STBC. This register is a protected location and can be written to only by a special instruction. If the third and fourth bytes of the instruction are not complements of each other, the data is not written and an operation code trap interrupt occurs.

Watchdog Timer

The watchdog timer protects against inadvertent program loops. A nonmaskable interrupt occurs if the timer is not reset by the program before it overflows. At the same time, the watchdog timer output pin, $\overline{\text{WDTO}}$, goes active low for a period of 32 system clocks. The $\overline{\text{WDTO}}$ can be connected to the $\overline{\text{RESET}}$ pin or used to control external circuitry. Three program-selectable intervals are available: 10.5, 41.9, and 167.8 ms at 25 MHz; 8.2, 32.8, and 131.1 ms at 32 MHz.

Once started, the timer can be stopped only by an external reset. Watchdog timer mode register WDM is used to select the time interval, to set the relative priority of the watchdog timer interrupt and NMI, and to clear the timer. This register is a protected location and can be written to only by a special instruction. If the third and fourth bytes of the instruction are not complements of each other, the data is not written and an operation code trap interrupt occurs.

External Reset

The μPD78352 family is reset by taking the $\overline{\text{RESET}}$ pin low. The reset circuit contains a noise filter to protect against spurious system resets caused by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low until the power supply reaches its operating voltage and the oscillator has stabilized. During reset, the program counter is loaded with the address contained in the reset vector table (addresses 0000H, 0001H); program execution starts at that address upon the $\overline{\text{RESET}}$ pin going high. While $\overline{\text{RESET}}$ is low, all external lines except $\overline{\text{WDTO}}$, CLKOUT, V_{SS}, V_{DD}, X1, and X2 are in the high-impedance state.

5c

ELECTRICAL SPECIFICATIONS

Note: Specifications are preliminary for μPD78352A and final for μPD78350/P352.

Absolute Maximum Ratings

T_A = 25°C

Supply voltage, V_{DD}	−0.5 to +7.0 V
Supply voltage, V_{PP}	−0.5 to +13.5 V
Input voltage, V_I	
Except $P2_0$/NMI (A9) of μPD78P352	−0.5 to V_{DD} + 0.5 V
$P2_0$/NMI (A9) of μPD78P352	−0.5 to +13.5 V
Output voltage, V_O	−0.5 to V_{DD} + 0.5 V
Output current, low; I_{OL}	
Each output pin	4.0 mA
Total	100 mA
Output current, high; I_{OH}	
Each output pin	−1.0 mA
Total	−20 mA
Operating temperature, T_{OPT}	−10 to +70°C
Storage temperature, T_{STG}	−65 to +150°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC characteristics.

Capacitance

T_A = 25°C; V_{DD} = V_{SS} = 0 V

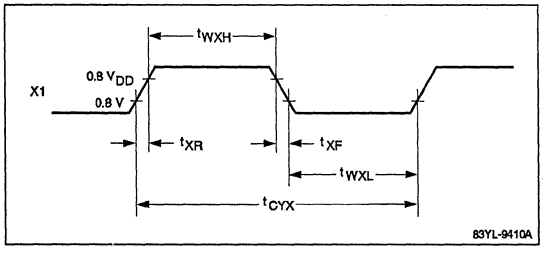
Parameter	Symbol	Max	Unit	Conditions
Input pin capacitance	C_I	20	pF	f = 1 MHz; unmeasured pins returned to 0 V
Output pin capacitance	C_O	20	pF	
I/O pin capacitance	C_{IO}	20	pF	

Oscillator Conditions

T_A = −10 to 70°C; V_{DD} = +5 V ±10%

			μPD78350		μPD78352A/P352		
Oscillator	**Parameter**	**Symbol**	**Min**	**Max**	**Min**	**Max**	**Unit**
Ceramic resonator or crystal	Oscillation frequency	f_{XX}	8	25	8	32	MHz
External clock	X1 input frequency	f_X	8	25	8	32	MHz
	X1 clock cycle time	t_{CYX}	40	125	31.25	125	ns
	X1 input rise/fall time	t_{XR}, t_{XF}	0	10	0	10	ns
	X1 input high/low level width	t_{WXH}, t_{WXL}	15	60	10	60	ns

External Clock



DC Characteristics

T_A = −10 to +70°C; V_{DD} = +5.0 V ±10%; V_{SS} = 0 V

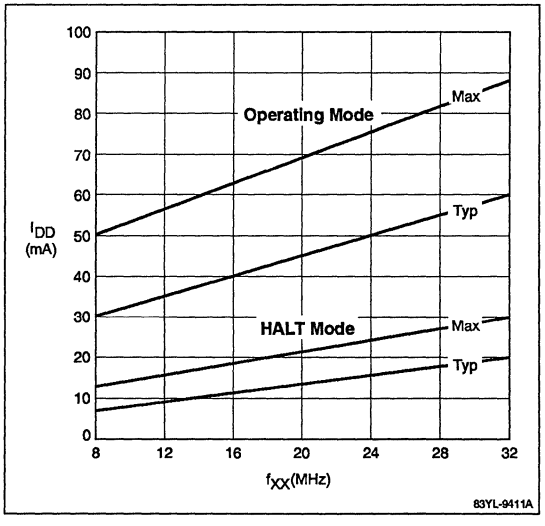
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input voltage, low	V_{IL}	0		0.8	V	
Input voltage, high	V_{IH1}	2.2			V	(Note 1)
	V_{IH2}	0.8 V_{DD}			V	(Note 2)
Output voltage, low	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output voltage, high	V_{OH}	V_{DD} − 1.0			V	I_{OH} = −400 μA
Input leakage current	I_{LI}			±10	μA	V_I = 0 to V_{DD}
Output leakage current	I_{LO}			±10	μA	V_O = 0 to V_{DD}
V_{DD} supply current	I_{DD1}		50	90	mA	Operating mode, μPD78350
			60	87	mA	Operating mode, μPD78352A; f_{XX} = 32 MHz
			80	120	mA	Operating mode, μPD78P352; f_{XX} = 32 MHz
	I_{DD2}		25	40	mA	HALT mode, μPD78350; f_{XX} = 25 MHz
			20	30	mA	HALT mode, μPD78352A; f_{XX} = 32 MHz
			35	50	mA	HALT mode, μPD78P352; f_{XX} = 32 MHz
Data retention voltage	V_{DDDR}	2.5			V	STOP mode
Data retention current	I_{DDDR}		2	10	μA	STOP mode; V_{DDDR} = 2.5 V
			10	50	μA	STOP mode; V_{DDDR} = 5.0 V ±10%

Notes:

(1) All except pins in Note 2.

(2) Pins $\overline{\text{RESET}}$, X1, X2, INTPn, NMI, and TI.

Power Consumption, 78352A



5c

AC Characteristics

T_A = −10 to +70°C; V_{DD} = +5.0 V ±10%; V_{SS} = 0 V

Parameter	Symbol	μPD78350 f_{XX} = 25 MHz Min	μPD78350 f_{XX} = 25 MHz Max	μPD78352A/P352 f_{XX} = 32 MHz Min	μPD78352A/P352 f_{XX} = 32 MHz Max	Unit	Conditions
External Memory Read/Write Operation							
System clock cycle time (Note 1)	t_{CYK}	80	250	62.5	250	ns	C_L = 50 pF
Address setup time to ASTB ↓	t_{SAST}	16		7		ns	C_L = 100 pF (Note 2)
Address hold after ASTB ↓	t_{HSTA}	26		11		ns	C_L = 100 pF
$\overline{RD}$ ↓ to address floating	t_{FRA}		0		0	ns	C_L = 100 pF
Address to data input valid	t_{DAID}		144		100	ns	C_L = 100 pF (Notes 2, 3)
$\overline{RD}$ ↓ to data input valid	t_{DRID}		76		49	ns	C_L = 100 pF (Note 3)
ASTB ↓ to $\overline{RD}$ ↓ delay time	t_{DSTR}	24		15		ns	C_L = 100 pF
Data hold time from $\overline{RD}$ ↑	t_{HRID}	0		0		ns	C_L = 100 pF
$\overline{RD}$ ↑ to address active	t_{DRA}	26		25		ns	C_L = 100 pF
$\overline{RD}$ width low	t_{WRL}	90		63		ns	C_L = 100 pF (Note 3)
ASTB width, high	t_{WSTH}	23		14		ns	C_L = 100 pF (Note 2)
$\overline{WR}$ to data output	t_{DWOD}		29		21	ns	C_L = 100 pF
ASTB ↓ to $\overline{WR}$ ↓ delay	t_{DSTW}	24		15		ns	C_L = 100 pF
Data setup time to $\overline{WR}$ ↑	t_{SODW}	75		57		ns	C_L = 100 pF (Note 3)
Data hold time after $\overline{WR}$ ↑	t_{HWOD}	8		8		ns	C_L = 100 pF
$\overline{WR}$ width, low	t_{WWL}	90		57		ns	C_L = 100 pF (Note 3)
$\overline{WAIT}$ setup time from address	t_{SAWT}		—		107	ns	C_L = 100 pF (Note 2, 4)
$\overline{WAIT}$ setup time from $\overline{RD}$ ↓ or $\overline{WR}$ ↓	t_{SRWRY}		—		37	ns	C_L = 100 pF (Note 4)
$\overline{WAIT}$ hold time from address	t_{HAWT}	—		149		ns	C_L = 100 pF (Note 2, 4)
$\overline{WAIT}$ hold time from $\overline{RD}$ ↓ or $\overline{WR}$ ↓	t_{HRWRY}	—		80		ns	C_L = 100 pF (Note 4)
ASTB ↑ delay time from $\overline{WR}$ ↑	t_{DWST}	110		78		ns	C_L = 100 pF
Address to $\overline{RD}$ ↓ or $\overline{WR}$ ↓ delay	t_{DARW}	89			69	ns	C_L = 100 pF
Other Operations							
NMI high/low level width	t_{WNIH}, t_{WNIL}	2.5		2.0		μs	
INTP0 high/low level width	t_{WI0H}, t_{WI0L}	640		500		ns	
INTP1 high/low level width	t_{WI1H}, t_{WI1L}	640		500		ns	
INTP2 high/low level width	t_{W12H}, t_{W12L}	640		500		ns	
INTP3 high/low level width	t_{WI3H}, t_{W13L}	640		500		ns	
$\overline{RESET}$ high/low level width	t_{WRSH}, t_{WRSL}	2.5		2.0		μs	
TI high/low level width	t_{WTIH}, t_{WTIL}	640		500		ns	

Notes:

(1) t_{CYK} equals twice the period of the crystal or external clock input.

(2) No address wait

(3) No wait states

(4) One external wait state and one internal wait state

Timing Dependent on t_{CYK}

Symbol	Calculation Formula	Min/Max	Unit
t_{SAST}	(0.5 + a)T – 24	Min	ns
t_{HSTA}	0.5T – 14 0.5T – 20 (Note 1)	Min	ns
t_{WSTH}	(0.5 + a)T – 17	Min	ns
t_{DSTR}	0.5T – 16	Min	ns
t_{WRL}	(1.5 + n)T – 30	Min	ns
t_{DAID}	(2.5 + a + n)T – 56	Max	ns
t_{DRID}	(1.5 + n)T – 44	Max	ns
t_{DRA}	0.5T – 14 0.5T – 6 (Note 1)	Min	ns
t_{DSTW}	0.5T – 16	Min	ns
t_{WWL}	(1.5 + n)T – 30 (1.5 + n)T – 36 (Note 1)	Min	ns
t_{DWOD}	0.5T – 10	Max	ns
t_{SODW}	(1 + n)T – 5	Min	ns
t_{SAWT}	(a + n)T – 18 (Note 1)	Max	ns
t_{HAWT}	(0.5 + a + n)T –7 (Note 1)	Min	ns
t_{SRWRY}	(n – 1)T – 25 (Note 1)	Max	ns
t_{HRWRY}	(n – 0.5)T – 14 (Note 1)	Min	ns
t_{DARW}	(a + 1)T+9 (a +1)T+7 (Note 1)	Max	ns
t_{DWST}	1.5T – 10 1.5T – 15 (Note 1)	Min	ns
t_{WIOH}	8T	Min	ns
t_{WIOL}	8T	Min	ns
t_{WI1H}	8T	Min	ns
t_{WI1L}	8T	Min	ns
t_{WI2H}	8T	Min	ns
t_{WI2L}	8T	Min	ns
t_{WI3H}	8T	Min	ns
t_{WI3L}	8T	Min	ns
t_{WTIH}	8T	Min	ns
t_{WTIL}	8T	Min	ns

Notes:

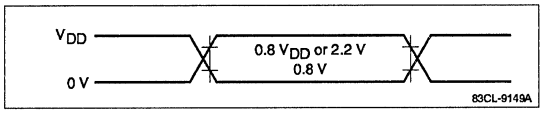
(1) 78352A/P352 only

(2) T = t_{CYK} (ns)

(3) When an address wait is inserted, the value of letter "a" is 1. Otherwise, it is 0.

(4) Letter "n" is the number of wait cycles specified by the external wait pin $\overline{WAIT}$ and the PWC register.

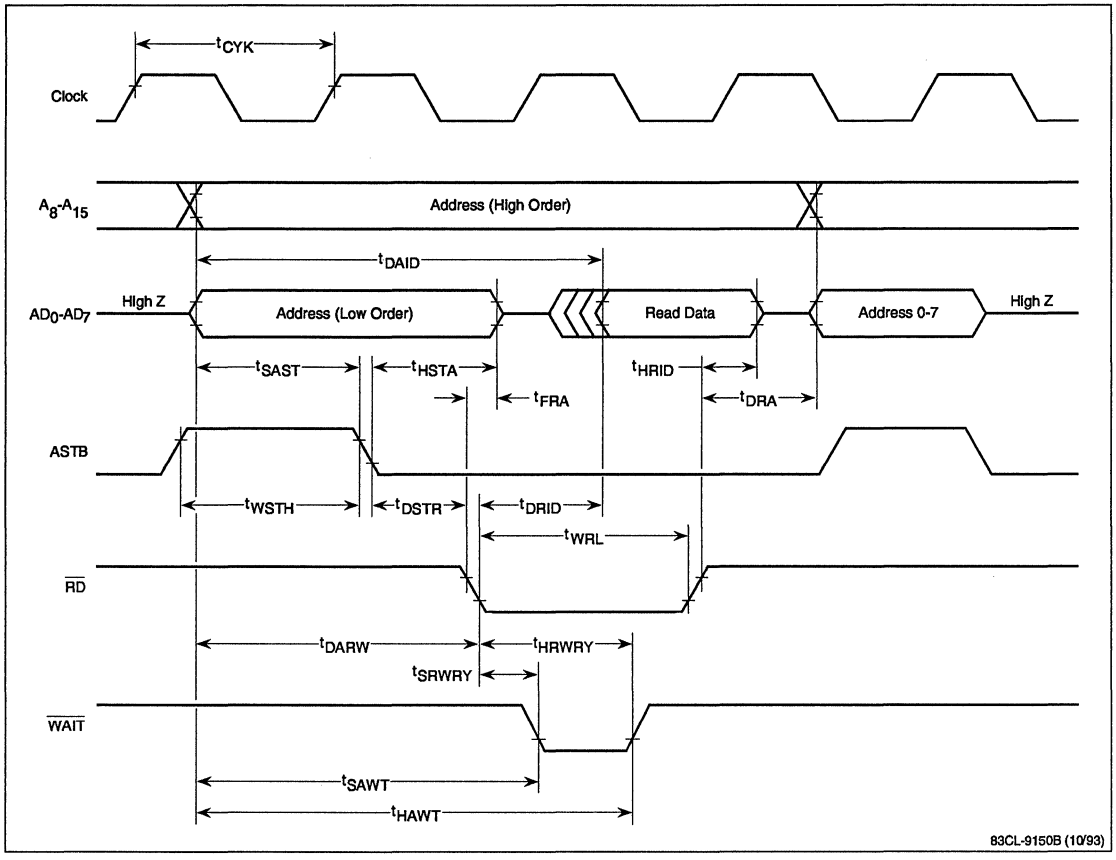
AC Timing Test Points



5c

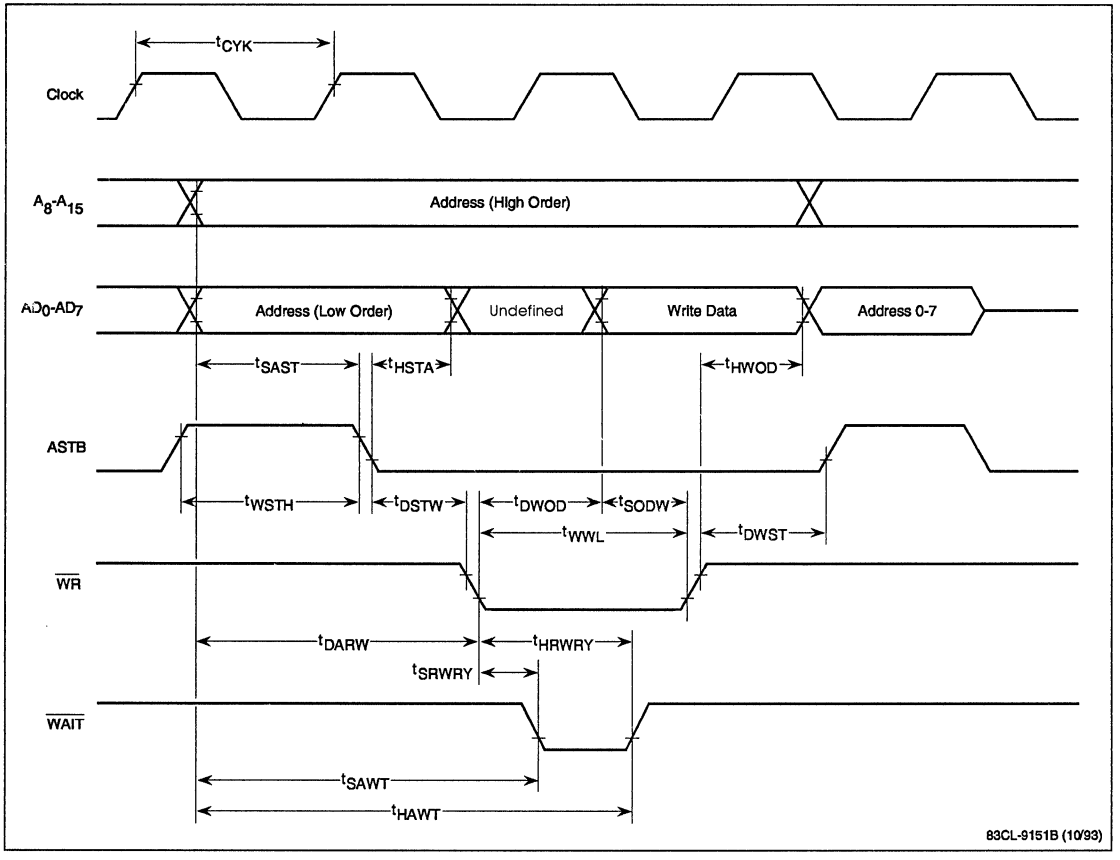
Timing Waveforms

Read Operation



Timing Waveforms (cont)

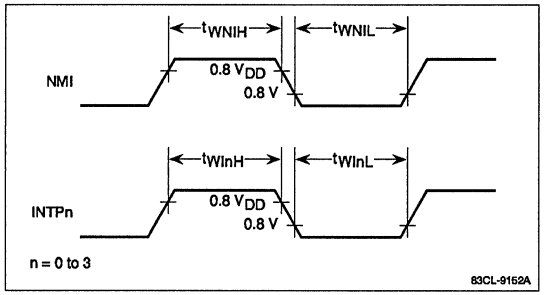
Write Operation



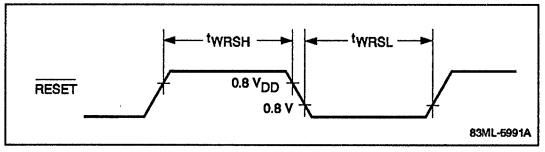
5c

Timing Waveforms (cont)

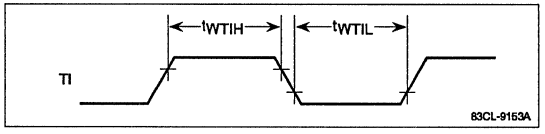
Interrupt Input



Reset Input



TI Input



PROM PROGRAMMING

The PROM in the μPD78P352 is one-time programmable (OTP) or ultraviolet erasable (UV EPROM). The 32,758 x 8-bit PROM has the programming characteristics of an NEC μPD27C1001A, including both page and byte programming modes. The MODE0/V_{PP}, MODE1, $P2_1$, and $\overline{RESET}$ pins are used to place the μPD78P352 into the PROM programmming mode. Table 3 shows the functions of the μPD78P352 pins in normal operating mode and PROM programming mode.

Table 3. Pin Functions During PROM Programming

Function	Normal Operating Mode	Programming Mode
Address input	$P0_0$ - $P0_7$, $P5_0$, $P2_0$, $P5_1$ - $P5_7$	A_0 - A_{16}
Data input	$P4_0$ - $P4_7$	D_0 - D_7
Program pulse	$P1_2$	$\overline{PGM}$
Chip enable	$P1_1$	$\overline{CE}$
Output enable	$P1_0$	$\overline{OE}$
Program voltage	MODE0/V_{PP}	MODE0/V_{PP}
Mode voltage	MODE1, $P2_1$, $\overline{RESET}$	MODE1, $P2_1$, $\overline{RESET}$

PROM Programming Mode

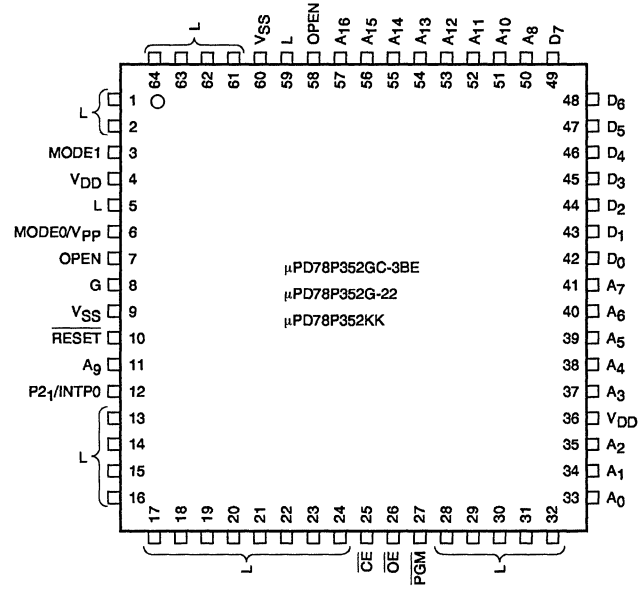
When +6.5 V is applied to the V_{DD} pin and +12.5 V to the MODE0/V_{PP} pin, the μPD78P352 enters the PROM programming mode. Operation in this mode is determined by the setting of $\overline{CE}$, $\overline{OE}$, and $\overline{PGM}$ pins as indicated in table 4.

Table 4. Operation Modes For Programming

Mode	MODE1	$P2_1$	$\overline{RESET}$	$\overline{CE}$	$\overline{OE}$	$\overline{PGM}$	MODE0/V_{PP}	V_{DD}	D_0 - D_7
Page data latch	L	L	L	H	L	H	+12.5 V	+6.5 V	Data input
Page program	L	L	L	H	H	L	+12.5 V	+6.5 V	High impedance
Byte program	L	L	L	L	H	L	+12.5 V	+6.5 V	Data input
Program verify	L	L	L	L	L	H	+12.5 V	+6.5 V	Data output
Program inhibit	L	L	L	X X	L H	L H	+12.5 V	+6.5 V	High impedance
Read	L	L	L	L	L	H	+5.0 V	+5.0 V	Data output
Ouput disable	L	L	L	L	H	X	+5.0 V	+5.0 V	High impedance
Standby	L	L	L	H	X	X	+5.0 V	+5.0 V	High impedance

X can be either H or L.

Figure 10. Pin Functions in μPD78P352 PROM Programming Mode



L Connect these pins separately to V_{SS} through resistors.
G Connect this pin to V_{SS}.
OPEN Do not connect these pins.

83CL-9154B

5c

PROM Byte Programming Procedure

Data can be written to the PROM one byte at a time by the following procedure.

(1) Set the pins not used for programming as indicated in figure 10. Set MODE0/V_{PP} and V_{DD} pins to +5 V and MODE1, $P2_1$, and $\overline{RESET}$ pins to 0 V. The $\overline{CE}$, $\overline{OE}$, and $\overline{PGM}$ pins should be high.

(2) Supply +6.5 V to V_{DD} pin and +12.5 V to MODE0/V_{PP} pin. Set $\overline{CE}$ pin low and $\overline{OE}$ pin high.

(3) Provide initial address to pins A_0 - A_{16}.

(4) Provide write data.

(5) Input a 0.1-ms program pulse (active low) to $\overline{PGM}$ pin.

(6) Use verify mode (pulse $\overline{OE}$ low) to test data. If data has been written, proceed to step 8; if not, repeat steps 4–6. If data cannot be written in 10 attempts, go to step 7.

(7) Classify PROM as defective and cease write operation.

(8) Increment address.

(9) Repeat steps 4–8 until last address is programmed.

PROM Page Programming Procedure

Data can be written to the PROM four bytes at a time (page programming) by the following procedure.

(1) Set the pins not used for programming as indicated in figure 10. Set MODE0/V_{PP} and V_{DD} pins to +5 V and MODE1, $P2_1$, and $\overline{RESET}$ pins to 0 V. The $\overline{CE}$, $\overline{OE}$, and $\overline{PGM}$ pins should be high.

(2) Supply +6.5 V to V_{DD} pin and +12.5 V to MODE0/V_{PP} pin. Set $\overline{CE}$ pin low.

(3) Provide initial page address to pins A_0 - A_{16}.

(4) Provide first byte of data and latch it into PROM by pulsing $\overline{OE}$ low. Continue incrementing address and latching in data until four bytes have been loaded.

(5) Input a 0.1-ms program pulse (active low) to $\overline{PGM}$ pin. Data bus D_0 - D_7 is in a high-impedance state.

(6) Use verify mode (pulse $\overline{OE}$ low four times) to test four bytes of data. If all four bytes of data have been written, proceed to step 8; if not, repeat steps 4–6. If data cannot be written in 10 attempts, go to step 7.

(7) Classify PROM as defective and cease write operation.

(8) Increment address.

(9) Repeat steps 4–8 until last address is programmed.

PROM Read Procedure

The contents of the PROM can be read out to the external data bus (D_0 - D_7) by the following procedure.

(1) Set the pins not used for programming as indicated in figure 10. Set MODE0/V_{PP} and V_{DD} pins to +5 V and MODE1, $P2_1$, and $\overline{RESET}$ pins to 0 V. The $\overline{CE}$, $\overline{OE}$, and $\overline{PGM}$ pins should be high.

(2) Supply +5 V to V_{DD} pin and MODE0/V_{PP} pin.

(3) Input address of data to be read to pins A_0 - A_{16}.

(4) Put an active-low pulse on $\overline{CE}$ and $\overline{OE}$ pins.

(5) Data is output to pins D_0 - D_7.

Program Erasure

The UV EPROM can be erased by exposing the window to light having a wavelength shorter than 400 nm, including ultraviolet, direct sunlight, and fluoresecent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15 Ws/cm^2 (ultraviolet ray intensity x exposure time) is required to completely erase the written data. Erasure by an ultraviolet lamp rated at 12,000 μW/cm^2 takes 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

DC Programming Characteristics

T_A = 25°C ±5°C; V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH1}	2.2		V_{DD}	V	(Note 1)
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	(Note 2)
V_{DDP} power supply voltage	V_{DDP}	6.25	6.5	6.75	V	Memory program mode
		4.5	5.0	5.5	V	Memory read mode
V_{PP} power supply voltage	V_{PP}	12.2	12.5	12.8	V	Memory program mode
			V_{PP} = V_{DDP}		V	Memory read mode
V_{DDP} power supply current	I_{DDP}			30	mA	Memory program mode
				100	mA	Memory read mode
V_{PP} power supply current	I_{PP}			50	mA	Memory program mode
			1	100	μA	Memory read mode

Notes:

(1) All except pins in Note 2.

(2) Pins $\overline{RESET}$, X1, X2, $P2_n$, INTPn, NMI, and TI.

AC Programming Characteristics

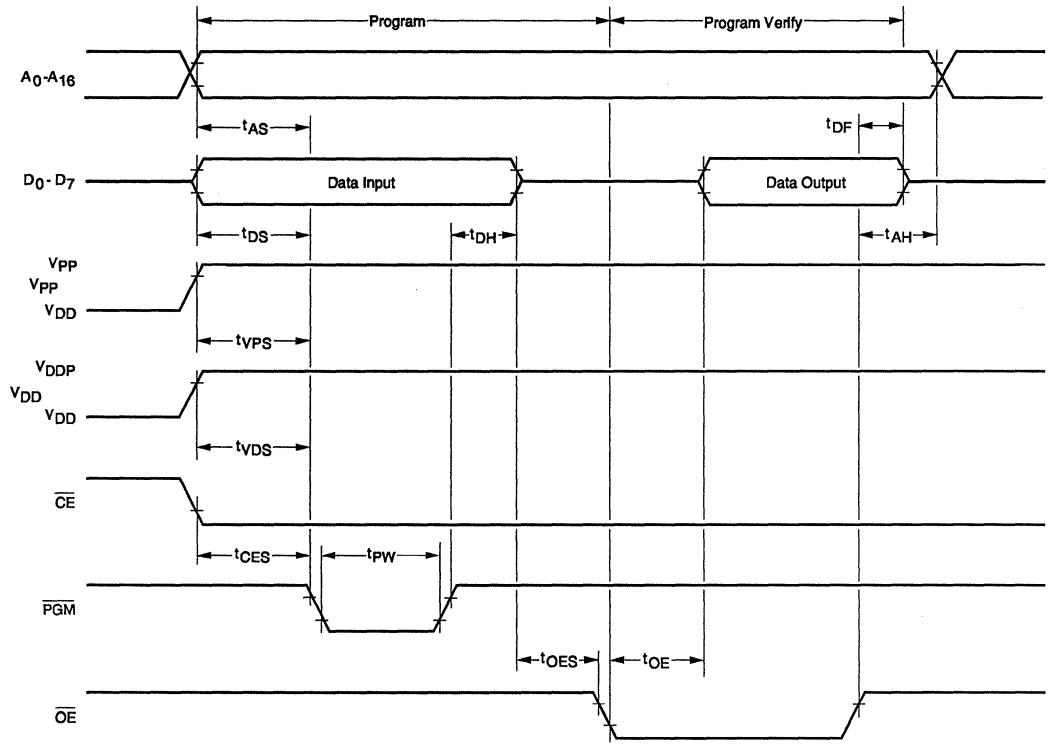
T_A = 25°C ±5°C; V_{DD} = 6.5 ±0.25 V; V_{PP} = 12.5 ±0.3 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Byte Programming Mode						
Address setup time to $\overline{PGM}$ ↓	t_{AS}	2			μs	
$\overline{CE}$ setup time to $\overline{PGM}$ ↓	t_{CES}	2			μs	
Input data setup time to $\overline{PGM}$ ↓	t_{DS}	2			μs	
Address hold time after $\overline{OE}$ ↑	t_{AH}	2			μs	
Input data hold time after $\overline{PGM}$ ↑	t_{DH}	2			μs	
Output data hold time after $\overline{OE}$ ↑	t_{DF}	0		130	ns	
V_{PP} setup time before $\overline{PGM}$ ↓	t_{VPS}	2			μs	
V_{DD} setup time before $\overline{PGM}$ ↓	t_{VDS}	2			μs	
Program pulse width	t_{PW}	0.095	0.1	0.105	ms	
Data to $\overline{OE}$ ↓ delay time	t_{OES}	2			μs	
$\overline{OE}$ ↓ to data output time	t_{OE}			150	ns	
Page Programming Mode						
Address setup time to $\overline{OE}$ ↓	t_{AS}	2			μs	
$\overline{CE}$ setup time to $\overline{OE}$ ↓	t_{CES}	2			μs	
Input data setup time to $\overline{OE}$ ↓	t_{DS}	2			μs	
Address hold time from $\overline{OE}$ ↑	t_{AH}	2			μs	
	t_{AHL}	2			μs	
	t_{AHV}	0			μs	
Input data hold time after $\overline{OE}$ ↑	t_{DH}	2			μs	
Output data hold time after $\overline{OE}$ ↑	t_{DF}	0		130	ns	
V_{PP} setup time to $\overline{OE}$ ↓	t_{VPS}	2			μs	
V_{DD} setup time to $\overline{OE}$ ↓	t_{VDS}	2			μs	
Program pulse width	t_{PW}	0.095	0.1	0.105	ms	
Address to $\overline{OE}$ ↓ delay time	t_{OES}	2			μs	
$\overline{OE}$ ↓ to data output time	t_{OE}			150	ns	
$\overline{OE}$ pulse width during data latch	t_{LW}	1			μs	
Data to $\overline{PGM}$ ↓ delay time	t_{PGMS}	2			μs	
$\overline{CE}$ hold time from $\overline{PGM}$ ↑	t_{CEH}	2			μs	
$\overline{CE}$ hold time from $\overline{OE}$ ↑	t_{OEH}	2			μs	
Read Mode						
Address to data output time	t_{ACC}			200	ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}
$\overline{CE}$ ↓ to data output time	t_{CE}			200	ns	$\overline{OE}$ = V_{IL}
$\overline{OE}$ ↓ to data output time	t_{OE}			75	ns	$\overline{CE}$ = V_{IL}
Data hold time from $\overline{OE}$ ↑	t_{DF}	0		60	ns	$\overline{CE}$ = V_{IL}
Data hold time from address	t_{OH}	0			ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}

5c

PROM Timing Diagrams

Byte Programming Mode



Notes:

[1] V_{DD} must be applied before V_{PP} and removed after V_{PP}.

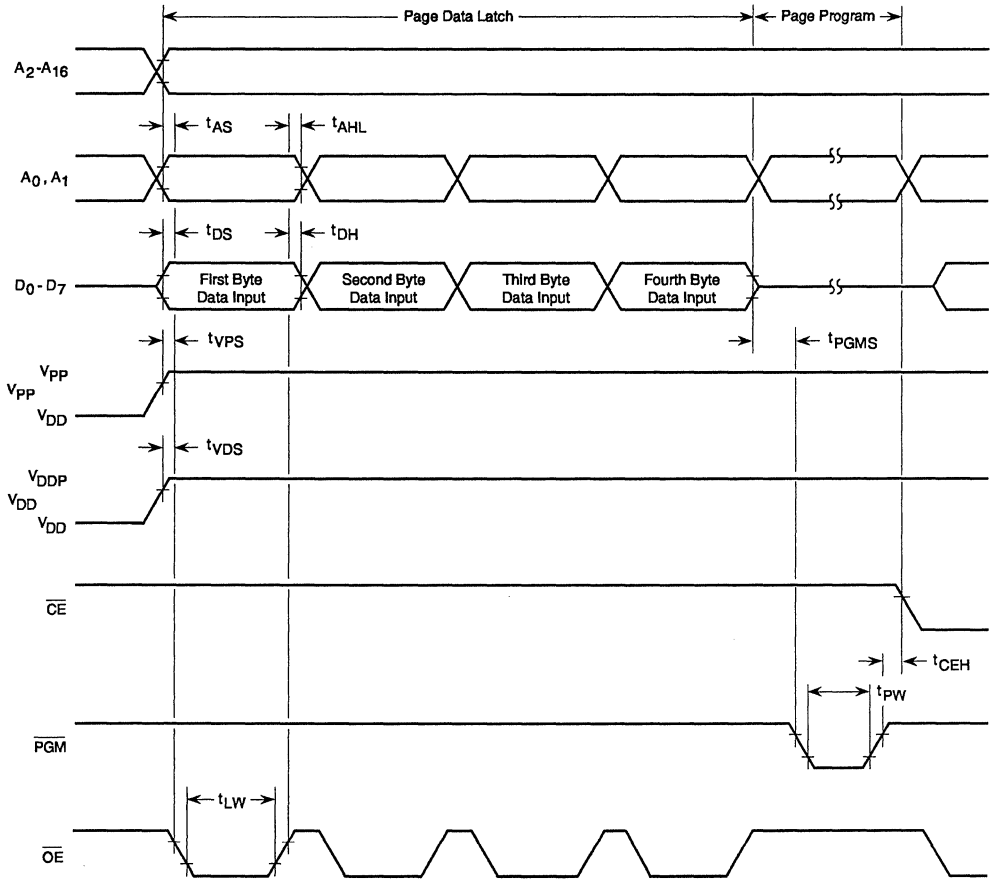
[2] V_{PP} must not be greater than +13.5 V, including overshoot.

[3] Removing and reinserting the device while a voltage of +12.5 V is applied to pin V_{PP} may affect device reliability.

83CL-9156B

PROM Timing Diagrams (cont)

Page Programming Mode; Page Data Latch → Page Program



Notes:

[1] V_{DD} must be applied before V_{PP} and removed after V_{PP}.

[2] V_{PP} must not be greater than +13.5 V, including overshoot.

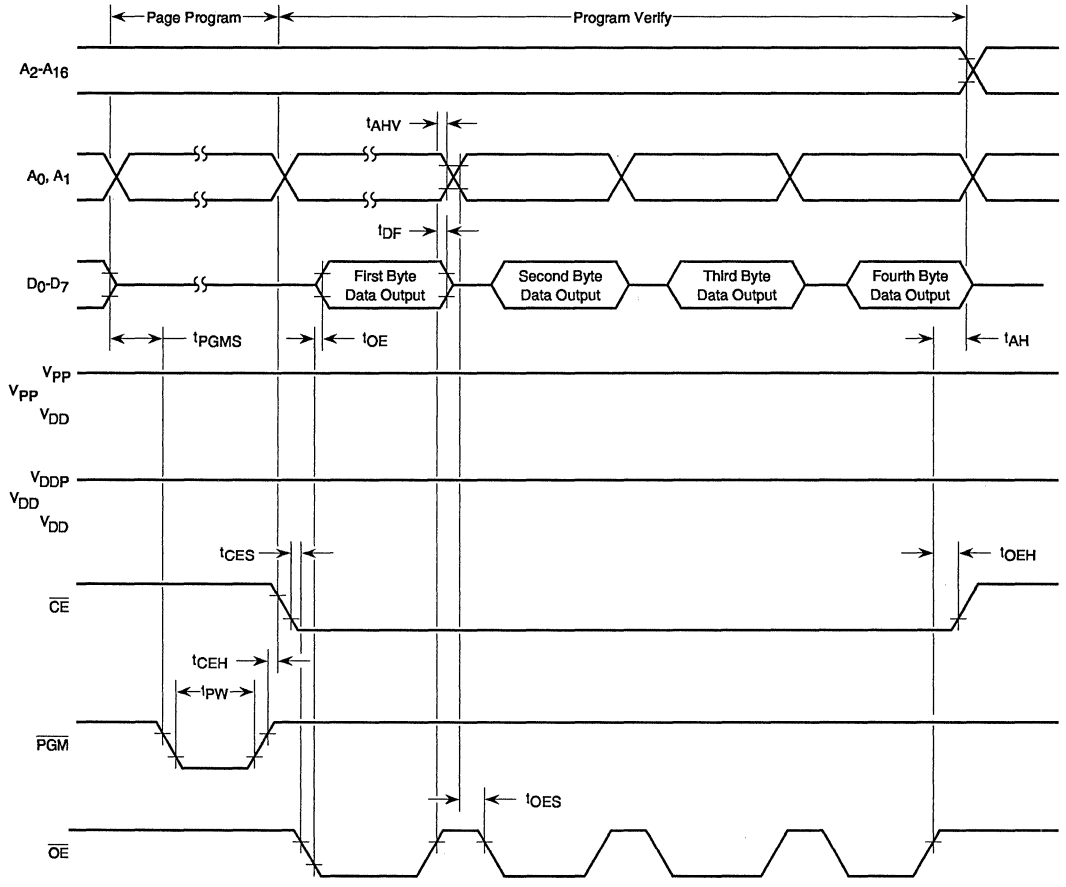
[3] Removing and reinserting the device while a voltage of +12.5 V is applied to pin V_{PP} may affect device reliability.

83CL-9156B

5c

PROM Timing Diagrams (cont)

Page Programming Mode; Page Program → Program Verify



Notes:

[1] V_{DD} must be applied before V_{PP} and removed after V_{PP}.

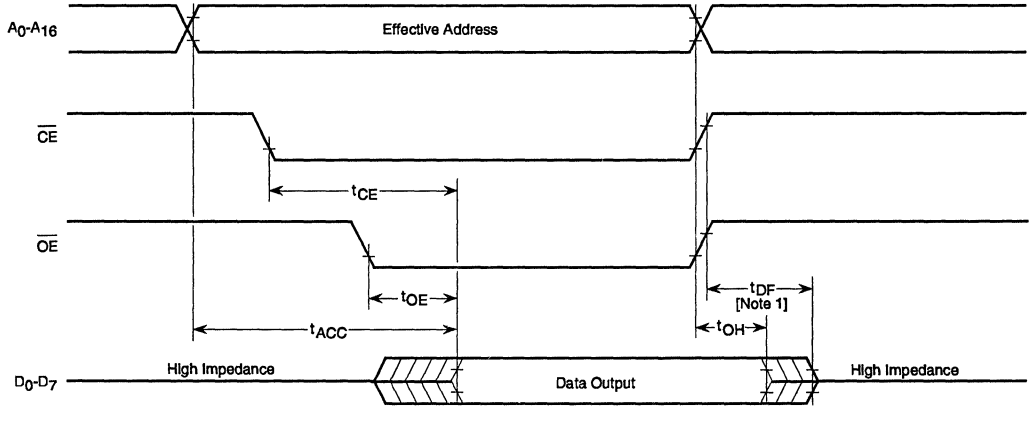
[2] V_{PP} must not be greater than +13.5 V, including overshoot.

[3] Removing and reinserting the device while a voltage of +12.5 V is applied to pin V_{PP} may affect device reliability.

83CL-9157B

PROM Timing Diagrams (cont)

Read Mode



Note:

[1] t_{DF} is specified from $\overline{OE}$ or $\overline{CE}$, whichever goes high first.

83CL-9158B

5c

INSTRUCTION SET

The instruction set of the μPD78350/P352 is upward compatible with the μPD78322 and μPD78312A families. Two instructions have been added to facilitate digital signal processing. The convolution instruction, MACW, calculates the sum of the products of "n" pairs of terms stored in Main RAM. The value of "n" is limited only by the amount of Main RAM available. The MOVTBL instruction displaces a data table by one 16-bit word to make room for a new data word.

The instruction set features both 8- and 16-bit data transfer, arithmetic, and logic instructions and single-bit manipulation instructions. String manipulation instructions are also included. Branch instructions exist to test individual bits in the program status word, the 16-bit accumulator, the special function registers, and the saddr portion of on-chip RAM. Instructions range in length from 1 to 6 bytes depending on the instruction and addressing mode.

Flag Column Indicators

Symbol	Action
(blank)	No change
0	Set to 0
1	Set to 1
X	Set or cleared according to result
P	P/V indicates parity of result
V	P/V indicates arithmetic overflow
R	Restored from saved PSW

Instruction Set Symbols

Symbol	Definition
r	R0, R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14, R15
r1	R0, R1, R2, R3, R4, R5, R6, R7
r2	C, B
rp	RP0, RP1, RP2, RP3, RP4, RP5, RP6, RP7*
rp1	RP0, RP1, RP2, RP3, RP4, RP5, RP6, RP7*
rp2	DE, HL, VP, UP
sfr	Special function register, 8 bits
sfrp	Special function register, 16 bits
post	RP0, RP1, RP2, RP3, RP4, RP5/PSW, RP6, RP7. Bits set to 1 indicate register pairs to be pushed/popped to/from stack; RP5 pushed/popped by PUSH/POP, SP is stack pointer; PSW pushed/popped by PUSHU/POPU, RP5 is stack pointer.

Instruction Set Symbols (cont)

Symbol	Definition
mem	Register indirect: [DE], [HL], [DE+], [HL+], [DE–], [HL–], [VP], [UP] Base Index Mode: [DE+A], [HL+A], [DE+B], [HL+B], [VP+DE], [VP+HL] Base Mode: [DE+byte], [HL+byte], [VP+byte], [UP+byte], [SP+byte] Index Mode: word [A], word [B], word [DE], word [HL]
saddr	FE20-FF1FH: Immediate byte addresses one byte in RAM, or label
saddrp	FE20-FF1FH: Immediate byte (bit 0=0) addresses one word in RAM, or label
word	16 bits of immediate data, or label
byte	8 bits of immediate data, or label
jdisp8	8-bit two's complement displacement (immediate data, displacement value –128 to +127)
bit	3 bits of immediate data (bit position in byte), or label
n	3 bits of immediate data
!addr16	16-bit absolute address specified by an immediate address or label
$addr16	Relative branch address or label
addr16	16-bit address
!addr11	11-bit immediate address or label
addr11	0800H-0FFFH: 0800H + (11-bit immediate address), or label
addr5	0040H-007EH: 0040H + 2 X (5-bit immediate address), or label
A	A register (8-bit accumulator)
X	X register
B	B register
C	C register
D	D register
E	E register
H	H register
L	L register
R0-R15	Register 0 to register 15
AX	Register pair AX (16-bit accumulator)
BC	Register pair BC
DE	Register pair DE
HL	Register pair HL

Instruction Set Symbols (cont)

Symbol	Definition
RP0-RP7	Register pair 0 to register pair 7
PC	Program counter
SP	Stack pointer
UP	User stack pointer (RP5)
PSW	Program status word
PSWH	High-order 8 bits of PSW
PSWL	Low-order 8 bits of PSW
CY	Carry flag
AC	Auxiliary carry flag
Z	Zero flag
P/V	Parity/overflow flag
S	Sign flag
TPF	Table position flag
RBS	Register bank select flag
RSS	Register set select flag
IE	Interrupt enable flag
STBC	Standby control register
WDM	Watchdog timer mode register
( )	Contents of the location whose address is within parentheses; (+) and (–) indicate that the address is incremented after or decremented after it is used
(( ))	Contents of the memory location defined by the quantity within the sets of parentheses
xxH	Hexadecimal quantity
X_H, X_L	High-order 8 bits and low-order 8 bits of X
∧	Logical product (AND)
∨	Logical sum (OR)
∀	Exclusive logical sum (exclusive OR)
—	Inverted data

* rp and rp1 describe the same registers but generate different machine code.

5c

Instruction Set

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
8-Bit Data Transfer								
MOV	r1, #byte	r1 ← byte	2					
	saddr, #byte	(saddr) ← byte	3					
	sfr, #byte (Note 1)	sfr ← byte	3					
	r, r1	r ← r1	2					
	A, r1	A ← r1	1					
	A, saddr	A ← (saddr)	2					
	saddr, A	(saddr) ← A	2					
	saddr, saddr	(saddr) ← (saddr)	3					
	A, sfr	A ← sfr	2					
	sfr, A	sfr ← A	2					
	A, mem (Note 2)	A ← (mem)	1					
	A, mem	A ← (mem)	2-4					
	mem, A (Note 2)	(mem) ← A	1					
	mem, A	(mem) ← A	2-4					
	A, [saddrp]	A ← ((saddrp))	2					
	[saddrp], A	((saddrp)) ← A	2					
	A, !addr16	A ← (addr16)	4					
	!addr16, A	(addr16) ← A	4					
	PSWL, #byte	PSWL ← byte	3	X	X	X	X	X
	PSWH, #byte	PSWH ← byte	3					
	PSWL, A	PSWL ← A	2	X	X	X	X	X
	PSWH, A	PSWH ← A	2					
	A, PSWL	A ← PSWL	2					
	A, PSWH	A ← PSWH	2					
XCH	A, r1	A ↔ r1	1					
	r, r1	r ↔ r1	2					
	A, mem	A ↔ (mem)	2-4					
	A, saddr	A ↔ (saddr)	2					
	A, sfr	A ↔ sfr	3					
	A, [saddrp]	A ↔ ((saddrp))	2					
	saddr, saddr	(saddr) ↔ (saddr)	3					
16-Bit Data Transfer								
MOVW	rp1, #word	rp1 ← word	3					
	saddrp, #word	(saddrp) ← word	4					
	sfrp, #word	sfrp ← word	4					
	rp, rp1	rp ← rp1	2					
	AX, saddrp	AX ← (saddrp)	2					
	saddrp, AX	(saddrp) ← AX	2					
	saddrp, saddrp	(saddrp) ← (saddrp)	3					

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
16-Bit Data Transfer (cont)								
MOVW (cont)	AX, sfrp	AX ← sfrp	2					
	sfrp, AX	sfrp ← AX	2					
	rp1, !addr16	rp1 ← (addr16)	4					
	!addr16, rp1	(addr16) ← rp1	4					
	AX, mem	AX ← (mem)	2-4					
	mem, AX	(mem) ← AX	2-4					
XCHW	AX, saddrp	AX ↔ (saddrp)	2					
	AX, sfrp	AX ↔ sfrp	3					
	saddrp, saddrp	(saddrp) ↔ (saddrp)	3					
	rp, rp1	rp ↔ rp1	2					
	AX, mem	AX ↔ (mem)	2-4					
8-Bit Arithmetic								
ADD	A, #byte	A, CY ← A + byte	2	X	X	X	V	X
	saddr, #byte	(saddr), CY ← (saddr) + byte	3	X	X	X	V	X
	sfr, #byte	sfr, CY ← sfr + byte	4	X	X	X	V	X
	r, r1	r, CY ← r + r1	2	X	X	X	V	X
	A, saddr	A, CY ← A + (saddr)	2	X	X	X	V	X
	A, sfr	A, CY ← A + sfr	3	X	X	X	V	X
	saddr, saddr	(saddr), CY ← (saddr) + (saddr)	3	X	X	X	V	X
	A, mem	A, CY ← A + (mem)	2-4	X	X	X	V	X
	mem, A	(mem), CY ← (mem) + A	2-4	X	X	X	V	X
ADDC	A, #byte	A, CY ← A + byte + CY	2	X	X	X	V	X
	saddr, #byte	(saddr), CY ← (saddr) + byte + CY	3	X	X	X	V	X
	sfr, #byte	sfr, CY ← sfr + byte + CY	4	X	X	X	V	X
	r, r1	r, CY ← r + r1 + CY	2	X	X	X	V	X
	A, saddr	A, CY ← A + (saddr) + CY	2	X	X	X	V	X
	A, sfr	A, CY ← A + sfr + CY	3	X	X	X	V	X
	saddr, saddr	(saddr), CY ← (saddr) + (saddr) + CY	3	X	X	X	V	X
	A, mem	A, CY ← A + (mem) + CY	2-4	X	X	X	V	X
	mem, A	(mem), CY ← (mem) + A + CY	2-4	X	X	X	V	X
SUB	A, #byte	A, CY ← A – byte	2	X	X	X	V	X
	saddr, #byte	(saddr), CY ← (saddr) – byte	3	X	X	X	V	X
	sfr, #byte	sfr, CY ← sfr – byte	4	X	X	X	V	X
	r, r1	r, CY ← r – r1	2	X	X	X	V	X
	A, saddr	A, CY ← A – (saddr)	2	X	X	X	V	X
	A, sfr	A, CY ← A – sfr	3	X	X	X	V	X
	saddr, saddr	(saddr), CY ← (saddr) – (saddr)	3	X	X	X	V	X
	A, mem	A, CY ← A – (mem)	2-4	X	X	X	V	X
	mem, A	(mem), CY ← (mem) – A	2-4	X	X	X	V	X

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
8-Bit Arithmetic (cont)								
SUBC	A, #byte	A, CY ← A – byte – CY	2	X	X	X	V	X
	saddr, #byte	(saddr), CY ← (saddr) – byte – CY	3	X	X	X	V	X
	sfr, #byte	sfr, CY ← sfr – byte – CY	4	X	X	X	V	X
	r, r1	r, CY ← r – r1 – CY	2	X	X	X	V	X
	A, saddr	A, CY ← A – (saddr) – CY	2	X	X	X	V	X
	A, sfr	A, CY ← A – sfr – CY	3	X	X	X	V	X
	saddr, saddr	(saddr), CY ← (saddr) – (saddr) – CY	3	X	X	X	V	X
	A, mem	A, CY ← A – (mem) – CY	2-4	X	X	X	V	X
	mem, A	(mem), CY ← (mem) – A – CY	2-4	X	X	X	V	X
8-Bit Logic								
AND	A, #byte	A ← A ∧ byte	2	X	X		P	
	saddr, #byte	(saddr) ← (saddr) ∧ byte	3	X	X		P	
	sfr, #byte	sfr ← sfr ∧ byte	4	X	X		P	
	r, r1	r ← r ∧ r1	2	X	X		P	
	A, saddr	A ← A ∧ (saddr)	2	X	X		P	
	A, sfr	A ← A ∧ sfr	3	X	X		P	
	saddr, saddr	(saddr)← (saddr) ∧ (saddr)	3	X	X		P	
	A, mem	A ← A ∧ (mem)	2-4	X	X		P	
	mem, A	(mem) ← (mem) ∧ A	2-4	X	X		P	
OR	A, #byte	A ← A ∨ byte	2	X	X		P	
	saddr, #byte	(saddr) ← (saddr) ∨ byte	3	X	X		P	
	sfr, #byte	sfr ← sfr ∨ byte	4	X	X		P	
	r, r1	r ← r ∨ r1	2	X	X		P	
	A, saddr	A ← A ∨ (saddr)	2	X	X		P	
	A, sfr	A ← A ∨ sfr	3	X	X		P	
	saddr, saddr	(saddr)← (saddr) ∨ (saddr)	3	X	X		P	
	A, mem	A ← A ∨ (mem)	2-4	X	X		P	
	mem, A	(mem) ← (mem) ∨ A	2-4	X	X		P	
XOR	A, #byte	A ← A ⊻ byte	2	X	X		P	
	saddr, #byte	(saddr) ← (saddr) ⊻ byte	3	X	X		P	
	sfr, #byte	sfr ← sfr ⊻ byte	4	X	X		P	
	r, r1	r ← r ⊻ r1	2	X	X		P	
	A, saddr	A ← A ⊻ (saddr)	2	X	X		P	
	A, sfr	A ← A ⊻ sfr	3	X	X		P	
	saddr, saddr	(saddr)← (saddr) ⊻ (saddr)	3	X	X		P	
	A, mem	A ← A ⊻ (mem)	2-4	X	X		P	
	mem, A	(mem) ← (mem) ⊻ A	2-4	X	X		P	

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
8-Bit Logic (cont)								
CMP	A, #byte	A – byte	2	X	X	X	V	X
	saddr, #byte	(saddr) – byte	3	X	X	X	V	X
	sfr, #byte	sfr – byte	4	X	X	X	V	X
	r, r1	r – r1	2	X	X	X	V	X
	A, saddr	A – (saddr)	2	X	X	X	V	X
	A, sfr	A – sfr	3	X	X	X	V	X
	saddr, saddr	(saddr) – (saddr)	3	X	X	X	V	X
	A, mem	A – (mem)	2-4	X	X	X	V	X
	mem, A	(mem) – A	2-4	X	X	X	V	X
16-Bit Arithmetic								
ADDW	AX, #word	AX, CY ← AX + word	3	X	X	X	V	X
	saddrp, #word	(saddrp), CY ← (saddrp) + word	4	X	X	X	V	X
	sfrp, #word	sfrp, CY ← sfrp + word	5	X	X	X	V	X
	rp, rp1	rp, CY ← rp + rp1	2	X	X	X	V	X
	AX, saddrp	AX, CY ← AX + (saddrp)	2	X	X	X	V	X
	AX, sfrp	AX, CY ← AX + sfrp	3	X	X	X	V	X
	saddrp, saddrp	(saddrp), CY ← (saddrp) + (saddrp)	3	X	X	X	V	X
SUBW	AX, #word	AX, CY ← AX – word	3	X	X	X	V	X
	saddrp, #word	(saddrp), CY ← (saddrp) – word	4	X	X	X	V	X
	sfrp, #word	sfrp, CY ← sfrp – word	5	X	X	X	V	X
	rp, rp1	rp, CY ← rp – rp1	2	X	X	X	V	X
	AX, saddrp	AX, CY ← AX – (saddrp)	2	X	X	X	V	X
	AX, sfrp	AX, CY ← AX – sfrp	3	X	X	X	V	X
	saddrp, saddrp	(saddrp), CY ← (saddrp) – (saddrp)	3	X	X	X	V	X
CMPW	AX, #word	AX – word	3	X	X	X	V	X
	saddrp, #word	(saddrp) – word	4	X	X	X	V	X
	sfrp, #word	sfrp – word	5	X	X	X	V	X
	rp, rp1	rp – rp1	2	X	X	X	V	X
	AX, saddrp	AX – (saddrp)	2	X	X	X	V	X
	AX, sfrp	AX – sfrp	3	X	X	X	V	X
	saddrp, saddrp	(saddrp) – (saddrp)	3	X	X	X	V	X
Multiplication/Division								
MULU	r1	AX ← AX x r1	2					
DIVUW	r1	AX (quotient), r1 (remainder) ← AX ÷ r1	2					
MULUW	rp1	AX (high-order 16 bits), rp1 (low-order 16 bits) ← AX x rp1	2					
DIVUX	rp1	AXDE (quotient), rp1 (remainder) ← AXDE ÷ rp1	2					
MULW (Note 3)	rp1	AX (high-order 16 bits), rp1 (low-order 16 bits) ← AX x rp1	2					

5c

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Sum-of-Products								
MACW	n	AXDE ← (B) x (C) + AXDE, B ← B + 2, C ← C + 2, n ← n–1. End if n = 0 or P/V = 1	3	X	X	X	V	X
Table Shift								
MOVTBLW	!addr 16, n (Note 4)	(addr16 + 2) ← (addr16), n ← n–1, addr16 ← addr16 – 2. End if n = 0	4					
Incrememt/Decrement								
INC	r1	r1 ← r1 + 1	1	X	X	X	V	
	saddr	(saddr) ← (saddr) + 1	2	X	X	X	V	
DEC	r1	r1 ← r1 – 1	1	X	X	X	V	
	saddr	(saddr) ← (saddr) – 1	2	X	X	X	V	
INCW	rp2	rp2 ← rp2 + 1	1					
	saddrp	(saddrp) ← (saddrp) + 1	3					
DECW	rp2	rp2 ← rp2 – 1	1					
	saddrp	(saddrp) ← (saddrp) – 1	3					
Shift/Rotate								
ROR	r1, n	(CY, $r1_7$ ← $r1_0$, $r1_{m-1}$ ← $r1_m$) x n times	2				P	X
ROL	r1, n	(CY, $r1_0$ ← $r1_7$, $r1_{m+1}$ ← $r1_m$) x n times	2				P	X
RORC	r1, n	(CY ← $r1_0$, $r1_7$ ← CY, $r1_{m-1}$ ← $r1_m$) x n times	2				P	X
ROLC	r1, n	(CY ← $r1_7$, $r1_0$ ← CY, $r1_{m+1}$ ← $r1_m$) x n times	2				P	X
SHR	r1, n	(CY ← $r1_0$, $r1_7$ ← 0, $r1_{m-1}$ ← $r1_m$) x n times	2	X	X	0	P	X
SHL	r1, n	(CY ← $r1_7$, $r1_0$ ← 0, $r1_{m+1}$ ← $r1_m$) x n times	2	X	X	0	P	X
SHRW	rp1, n	(CY ← $rp1_0$, $rp1_{15}$ ← 0, $rp1_{m-1}$ ← $rp1_m$) x n times	2	X	X	0	P	X
SHLW	rp1, n	(CY ← $rp1_{15}$, $rp1_0$ ← 0, $rp1_{m+1}$ ← $rp1_m$) x n times	2	X	X	0	P	X
ROR4	[rp1]	A_{3-0} ← $(rp1)_{3-0}$, $(rp1)_{7-4}$ ← A_{3-0}, $(rp1)_{3-0}$ ← $(rp1)_{7-4}$	2					
ROL4	[rp1]	A_{3-0} ← $(rp1)_{7-4}$, $(rp1)_{3-0}$ ← A_{3-0}, $(rp1)_{7-4}$ ← $(rp1)_{3-0}$	2					
BCD Adjustment								
ADJBA		Decimal adjust accumulator after add	2	X	X	X	P	X
ADJBS		Decimal adjust accumulator after subtract	2	X	X	X	P	X
Data Expansion								
CVTBW		X ← A, A_{6-0} ← A_7	1					
Bit Manipulation								
MOV1	CY, saddr.bit	CY ← (saddr.bit)	3					X
	CY, sfr.bit	CY ← sfr.bit	3					X
	CY, A.bit	CY ← A.bit	2					X
	CY, X.bit	CY ← X.bit	2					X
	CY, PSWH.bit	CY ← PSWH.bit	2					X

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Bit Manipulation (cont)								
MOV1 (cont)	CY, PSWL.bit	CY ← PSWL.bit	2					X
	saddr.bit, CY	(saddr.bit) ← CY	3					
	sfr.bit, CY	sfr.bit ← CY	3					
	A.bit, CY	A.bit ← CY	2					
	X.bit, CY	X.bit ← CY	2					
	PSWH.bit, CY	PSWH.bit ← CY	2					
	PSWL.bit, CY	PSWL.bit ← CY	2	X	X	X	X	
AND1	CY, saddr.bit	CY ← CY ∧ (saddr.bit)	3					X
	CY, /saddr.bit	CY ← CY ∧ $\overline{(saddr.bit)}$	3					X
	CY, sfr.bit	CY ← CY ∧ sfr.bit	3					X
	CY, /sfr.bit	CY ← CY ∧ $\overline{sfr.bit}$	3					X
	CY, A.bit	CY ← CY ∧ A.bit	2					X
	CY, /A.bit	CY ← CY ∧ $\overline{A.bit}$	2					X
	CY, X.bit	CY ← CY ∧ X.bit	2					X
	CY, /X.bit	CY ← CY ∧ $\overline{X.bit}$	2					X
	CY, PSWH.bit	CY ← CY ∧ PSWH.bit	2					X
	CY, /PSWH.bit	CY ← CY ∧ $\overline{PSWH.bit}$	2					X
	CY, PSWL.bit	CY ← CY ∧ PSWL.bit	2					X
	CY, /PSWL.bit	CY ← CY ∧ $\overline{PSWL.bit}$	2					X
OR1	CY, saddr.bit	CY ← CY ∨ (saddr.bit)	3					X
	CY, /saddr.bit	CY ← CY ∨ $\overline{(saddr.bit)}$	3					X
	CY, sfr.bit	CY ← CY ∨ sfr.bit	3					X
	CY, /sfr.bit	CY ← CY ∨ $\overline{sfr.bit}$	3					X
	CY, A.bit	CY ← CY ∨ A.bit	2					X
	CY, /A.bit	CY ← CY ∨ $\overline{A.bit}$	2					X
	CY, X.bit	CY ← CY ∨ X.bit	2					X
	CY, /X.bit	CY ← CY ∨ $\overline{X.bit}$	2					X
	CY, PSWH.bit	CY ← CY ∨ PSWH.bit	2					X
	CY, /PSWH.bit	CY ← CY ∨ $\overline{PSWH.bit}$	2					X
	CY, PSWL.bit	CY ← CY ∨ PSWL.bit	2					X
	CY, /PSWL.bit	CY ← CY ∨ $\overline{PSWL.bit}$	2					X
XOR1	CY, saddr.bit	CY ← CY ⊻ (saddr.bit)	3					X
	CY, sfr.bit	CY ← CY ⊻ sfr.bit	3					X
	CY, A.bit	CY ← CY ⊻ A.bit	2					X
	CY, X.bit	CY ← CY ⊻ X.bit	2					X
	CY, PSWH.bit	CY ← CY ⊻ PSWH.bit	2					X
	CY, PSWL.bit	CY ← CY ⊻ PSWL.bit	2					X

5c

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Bit Manipulation (cont)								
SET1	saddr.bit	(saddr.bit) ← 1	2					
	sfr.bit	sfr.bit ← 1	3					
	A.bit	A.bit ← 1	2					
	X.bit	X.bit ← 1	2					
	PSWH.bit	PSWH.bit ← 1	2					
	PSWL.bit	PSWL.bit ← 1	2	X	X	X	X	X
CLR1	saddr.bit	(saddr.bit) ← 0	2					
	sfr.bit	sfr.bit ← 0	3					
	A.bit	A.bit ← 0	2					
	X.bit	X.bit ← 0	2					
	PSWH.bit	PSWH.bit ← 0	2					
	PSWL.bit	PSWL.bit ← 0	2	X	X	X	X	X
NOT1	saddr.bit	(saddr.bit) ← $\overline{\text{(saddr.bit)}}$	3					
	sfr.bit	sfr.bit ← $\overline{\text{sfr.bit}}$	3					
	A.bit	A.bit ← $\overline{\text{A.bit}}$	2					
	X.bit	X.bit ← $\overline{\text{X.bit}}$	2					
	PSWH.bit	PSWH.bit ← $\overline{\text{PSWH.bit}}$	2					
	PSWL.bit	PSWL.bit ← $\overline{\text{PSWL.bit}}$	2	X	X	X	X	X
SET1	CY	CY ← 1	1					1
CLR1	CY	CY ← 0	1					0
NOT1	CY	CY ← $\overline{\text{CY}}$	1					X
Subroutine Linkage								
CALL	!addr16	$(SP-1) \leftarrow (PC + 3)_H$, $(SP - 2) \leftarrow (PC + 3)_L$, PC ← addr16, SP ← SP − 2	3					
	rp1	$(SP-1) \leftarrow (PC + 2)_H$, $(SP - 2) \leftarrow (PC + 2)_L$, $PC_H \leftarrow rp1_H$, $PC_L \leftarrow rp1_L$, SP ← SP − 2	2					
	[rp1]	$(SP-1) \leftarrow (PC + 2)_H$, $(SP - 2) \leftarrow (PC + 2)_L$, $PC_H \leftarrow (rp1 + 1)$, $PC_L \leftarrow (rp1)$, SP ← SP − 2	2					
CALLF	!addr11	$(SP-1) \leftarrow (PC + 2)_H$, $(SP - 2) \leftarrow (PC + 2)_L$, $PC_{15-11} \leftarrow 00001$, $PC_{10-0} \leftarrow$ addr11, SP ← SP − 2	2					
CALLT	[addr5]	$(SP-1) \leftarrow (PC + 1)_H$, $(SP - 2) \leftarrow (PC + 1)_L$, $PC_H \leftarrow$ (TPFx8000H + 2 x addr5 + 41H), $PC_L \leftarrow$ (TPFx8000H + 2 x addr5 + 40H), SP ← SP − 2	1					
BRK		$(SP-1) \leftarrow$ PSWH, $(SP - 2) \leftarrow$ PSWL, $(SP - 3) \leftarrow (PC + 1)_H$, $(SP - 4) \leftarrow (PC + 1)_L$, $PC_L \leftarrow$ (003EH), $PC_H \leftarrow$ (003FH), SP ← SP − 4, IE ← 0	1					
RET		$PC_L \leftarrow$ (SP), $PC_H \leftarrow$ (SP + 1), SP ← SP + 2	1					
RETB		$PC_L \leftarrow$ (SP), $PC_H \leftarrow$ (SP + 1), PSWL ← (SP + 2), PSWH ← (SP + 3), SP ← SP + 4	1	R	R	R	R	R
RETI		$PC_L \leftarrow$ (SP), $PC_H \leftarrow$ (SP + 1), PSWL ← (SP + 2), PSWH ← (SP + 3), SP ← SP + 4	1	R	R	R	R	R

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Stack Manipulation								
PUSH	sfrp	$(SP - 1) \leftarrow sfr_H$, $(SP - 2) \leftarrow sfr_L$, $SP \leftarrow SP - 2$	3					
	post	$\{(SP - 1) \leftarrow rpp_H$, $(SP - 2) \leftarrow rpp_L$, $SP \leftarrow SP - 2\}$ x n (Note 5)	2					
	PSW	$(SP - 1) \leftarrow PSWH$, $(SP - 2) \leftarrow PSWL$, $SP \leftarrow SP - 2$	1					
PUSHU	post	$\{(UP - 1) \leftarrow rpp_H$, $(UP - 2) \leftarrow rpp_L$, $UP \leftarrow UP - 2\}$ x n (Note 5)	2					
POP	sfrp	$sfr_L \leftarrow (SP)$, $sfr_H \leftarrow (SP + 1)$, $SP \leftarrow SP + 2$	3					
	post	$\{rpp_L \leftarrow (SP)$, $rpp_H \leftarrow (SP + 1)$, $SP \leftarrow SP + 2\}$ x n (Note 5)	2					
	PSW	$PSWL \leftarrow (SP)$, $PSWH \leftarrow (SP + 1)$, $SP \leftarrow SP + 2$	1	R	R	R	R	R
POPU	post	$\{rpp_L \leftarrow (UP)$, $rpp_H \leftarrow (UP + 1)$, $UP \leftarrow UP + 2\}$ x n (Note 5)	2					
MOVW	SP, #word	$SP \leftarrow$ word	4					
	SP, AX	$SP \leftarrow AX$	2					
	AX, SP	$AX \leftarrow SP$	2					
INCW	SP	$SP \leftarrow SP + 1$	2					
DECW	SP	$SP \leftarrow SP - 1$	2					
Pin Level Test								
CHKL	sfr	(Pin level) $\forall$ (internal signal level)	3	X	X		P	
CHKLA	sfr	$A \leftarrow$ (Pin level) $\forall$ (internal signal level)	3	X	X		P	
Unconditional Branch								
BR	!addr16	$PC \leftarrow$ addr16	3					
	rp1	$PC_H \leftarrow rp1_H$, $PC_L \leftarrow rp1_L$	2					
	[rp1]	$PC_H \leftarrow (rp1 + 1)$, $PC_L \leftarrow (rp1)$	2					
	$addr16	$PC \leftarrow PC + 2 +$ jdisp8	2					
Conditional Branch								
BC, BL	$addr16	$PC \leftarrow PC + 2 +$ jdisp8 if $CY = 1$	2					
BNC, BNL	$addr16	$PC \leftarrow PC + 2 +$ jdisp8 if $CY = 0$	2					
BZ, BE	$addr16	$PC \leftarrow PC + 2 +$ jdisp8 if $Z = 1$	2					
BNZ, BNE	$addr16	$PC \leftarrow PC + 2 +$ jdisp8 if $Z = 0$	2					
BV, BPE	$addr16	$PC \leftarrow PC + 2 +$ jdisp8 if $P/V = 1$	2					
BNV, BPO	$addr16	$PC \leftarrow PC + 2 +$ jdisp8 if $P/V = 0$	2					
BN	$addr16	$PC \leftarrow PC + 2 +$ jdisp8 if $S = 1$	2					
BP	$addr16	$PC \leftarrow PC + 2 +$ jdisp8 if $S = 0$	2					
BGT	$addr16	$PC \leftarrow PC + 3 +$ jdisp8 if $(P/V \forall S) \vee Z = 0$	3					
BGE	$addr16	$PC \leftarrow PC + 3 +$ jdisp8 if $P/V \forall S = 0$	3					
BLT	$addr16	$PC \leftarrow PC + 3 +$ jdisp8 if $P/V \forall S = 1$	3					
BLE	$addr16	$PC \leftarrow PC + 3 +$ jdisp8 if $(P/V \forall S) \vee Z = 1$	3					
BH	$addr16	$PC \leftarrow PC + 3 +$ jdisp8 if $Z \vee CY = 0$	3					
BNH	$addr16	$PC \leftarrow PC + 3 +$ jdisp8 if $Z \vee CY = 1$	3					

5c

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Conditional Branch								
BT	saddr.bit, $addr16	PC ← PC + 3 + jdisp8 if (saddr.bit) = 1	3					
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 1	4					
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 1	3					
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 1	3					
	PSWH.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWH.bit = 1	3					
	PSWL.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWL.bit = 1	3					
BF	saddr.bit, $addr16	PC ← PC + 4 + jdisp8 if (saddr.bit) = 0	4					
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 0	4					
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 0	3					
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 0	3					
	PSWH.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWH.bit = 0	3					
	PSWL.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWL.bit = 0	3					
BTCLR	saddr.bit, $addr16	PC ← PC + 4 + jdisp8 if (saddr.bit) = 1 then reset (saddr.bit)	4					
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit	4					
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 1 then reset A.bit	3					
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 1 then reset X.bit	3					
	PSWH.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWH.bit = 1 then reset PSWH.bit	3					
	PSWL.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWL.bit = 1 then reset PSWL.bit	3	X	X	X	X	X
BFSET	saddr.bit, $addr16	PC ← PC + 4 + jdisp8 if (saddr.bit) = 0 then set (saddr.bit)	4					
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 0 then set sfr.bit	4					
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 0 then set A.bit	3					
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 0 then set X.bit	3					
	PSWH.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWH.bit = 0 then set PSWH.bit	3					
	PSWL.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWL.bit = 0 then set PSWL.bit	3	X	X	X	X	X
DBNZ	r2, $addr16	r2 ← r2 – 1, then PC ← PC + 2 + jdisp8 if r2 = 0	2					
	saddr, $addr16	(saddr) ← (saddr) – 1, then PC ← PC + 3 + jdisp8 if (saddr) = 0	3					
Context Switching								
BRKCS	RBn	RBS_{2-0} ← n, PC_H ↔ R5, PC_L ↔ R4, R7 ← PSWH, R6 ← PSWL, RSS ← 0, IE ← 0	2					
RETCS	!addr16	PC_H ← R5, PC_L ← R4, R5 ← $addr16_H$, R4 ← $addr16_L$, PSWH ← R7, PSWL ← R6	3	R	R	R	R	R
RETCSB	!addr16	PC_H ← R5, PC_L ← R4, R5 ← $addr16_H$, R4 ← $addr16_L$, PSWH ← R7, PSWL ← R6	4	R	R	R	R	R

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
String Manipulation								
MOVM	[DE+], A	(DE+) ← A, C ← C–1 End if C = 0	2					
	[DE–], A	(DE–) ← A, C ← C–1 End if C = 0	2					
MOVBK	[DE+], [HL+]	(DE+) ← (HL+), C ← C–1 End if C = 0	2					
	[DE–], [HL–]	(DE–) ← (HL–), C ← C–1 End if C = 0	2					
XCHM	[DE+], A	(DE+) ↔ A, C ← C–1 End if C = 0	2					
	[DE–], A	(DE–) ↔ A, C ← C–1 End if C = 0	2					
XCHBK	[DE+], [HL+]	(DE+) ↔ (HL+), C ← C–1 End if C = 0	2					
	[DE–], [HL–]	(DE–) ↔ (HL–), C ← C–1 End if C = 0	2					
CMPME	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X
CMPBKE	[DE+], [HL+]	(DE+) – (HL+), C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X
CMPMNE	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X
CMPBKNE	[DE+], [HL+]	(DE+) – (HL+), C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X
CMPMC	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X
CMPBKC	[DE+], [HL+]	(DE+) – (HL+), C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X
CMPMNC	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X
CMPBKNC	[DE+], [HL+]	(DE+) – (HL+), C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X
CPU Control								
MOV	STBC, #byte	STBC ← byte (Note 6)	4					
	WDM, #byte	WDM ← byte (Note 6)	4					
SWRS		RSS ← $\overline{RSS}$	1					
SEL	RBn	RBS_{2-0} ← n, RSS ← 0	2					
	RBn, ALT	RBS_{2-0} ← n, RSS ← 1	2					
NOP		No operation	1					
EI		IE ← 1 (Enable interrupt)	1					
DI		IE ← 0 (Disable interrupt)	1					

5c

Instruction Set (cont)

Notes:

(1) A special instruction is used to write to STBC and WDM.

(2) One byte move instruction when [DE], [HL], [DE+], [DE–], [HL+], or [HL–] is specified for mem.

(3) 16-bit signed multiply instruction

(4) Addressing range is 0FE00H to 0FEFFH.

(5) rpp refers to register pairs specified in post byte. "n" is the number of register pairs specified in post byte.

(6) Trap if data bytes in operation code are not one's complement. If trap, then:
(SP–1) ← PSWH, (SP–2) ← PSWL, (SP–3) ← $(PC–4)_H$,
(SP–4) ← $(PC–4)_L$, PC_L ← (003CH), PC_H ← (003DH).
SP ← SP–4, IE ← 0.

NEC
NEC Electronics Inc.

μPD78356 Family (μPD78355/356/P356) 16-/8-Bit, K-Series Microcontrollers With A/D Converter and Convolution Capability

Preliminary

September 1993

Description

The μPD78355, μPD78356, and μPD78P356 are K-Series® microcontrollers. These 16-/8-bit devices —with a minimum instruction time of 125 ns at 32 MHz—are designed for high-speed, real-time process control. They feature a 16-bit CPU, a 16-/8-bit external data bus, eight banks of main registers, an advanced interrupt handling facility, and a powerful set of memory-mapped on-chip peripherals. A 16-bit multiply and accumulate instruction with or without a saturation word provides hardware convolution capability; a 16-bit subtract and accumulate absolute value instruction provides correlation capability.

On-board memory includes 2048 bytes of RAM and 48K bytes of mask ROM, UV EPROM, or one-time programmable (OTP) ROM. A ROMless version is also available. The UV EPROM and OTP ROM versions feature a PROM error correction function capable of correcting one 1-bit error per four bytes of code. This achieves a significant improvement in reliability over devices without error correction and is suited for applications that require high reliability under rigorous conditions.

The advanced interrupt handling facility has four levels of programmable hardware-priority control and three methods of servicing interrupt requests, including vectoring, hardware context switching, and macro service. The macro service facility reduces the CPU overhead involved in servicing peripheral interrupts by transferring data between the memory-mapped special function registers (SFRs) and memory without the use of time consuming interrupt service routines. In addition, the macro service facility can perform certain CPU functions, such as event counting and math-oriented data alterations.

The combination of high-speed hardware convolution capability and context switching, eight register banks, and the macro service facility makes these devices ideal for applications in the hard-disk drive and tape drive markets as well as the automotive, office automation, and industrial control/robotics markets.

K-Series is a registered trademark of NEC Electronics Inc.

Features

- Complete single-chip microcontroller
  - 16-bit ALU
  - 2048 bytes of RAM
  - 48K bytes of ROM (μPD78356) or PROM (μPD78P356)
- Powerful instruction set
  - 16-bit unsigned and signed multiply
  - 16-bit unsigned divide
  - 16-bit multiply and accumulate instruction with or without saturation word
  - 16-bit subtraction and accumulate absolute value instructions
  - 1-bit and 8-bit logic instructions
  - String instructions
- Minimum instruction time: 125 ns at 32 MHz
- 5-byte instruction prefetch queue
- Memory expansion
  - 8- or 16-bit external data bus
  - 64K-byte address space
- Large I/O capacity
  - Up to 57 I/O port lines (μPD78355)
  - Up to 76 I/O port lines (μPD78356/P356)
- Memory-mapped, on-chip peripherals (special function registers)
- Real-time pulse unit (RPU)
  - Two 16-bit interval timers
  - Two 16-bit timer/event counters
  - One 10-bit interval timer
  - One 16-bit up/down counter
  - Ten 16-bit capture/compare registers
  - Five external interrupt/capture lines
  - Three external event counter inputs
  - Three external timer clear inputs
  - Ten timer outputs
- Two pulse-width modulated (PWM) output lines with 8-, 10-, or 12-bit precision
- One 8-bit real-time output port
- Eight-channel, high-speed 10-bit A/D converter; conversion time: 2 μs at 32 MHz
- Two-channel, 8-bit D/A converter

Features (cont)

- Three-channel serial communications interface
  - — Asynchronous serial interface (UART)
  - — Clock synchronous serial interface 0
    - Full-duplex, three-wire mode
    - NEC serial bus interface (SBI) mode
  - — Clock synchronous serial interface 1
    - Full-duplex three-wire mode
    - Pin switching function
- Programmable priority interrupt controller (four levels)
- Three methods of interrupt service
  - — Vectored interrupts
  - — Context switching with hardware register bank switch
  - — Macro service mode with choice of five different functions
- Watchdog timer with dedicated output
- STOP and HALT standby functions
- Single 5-volt power supply

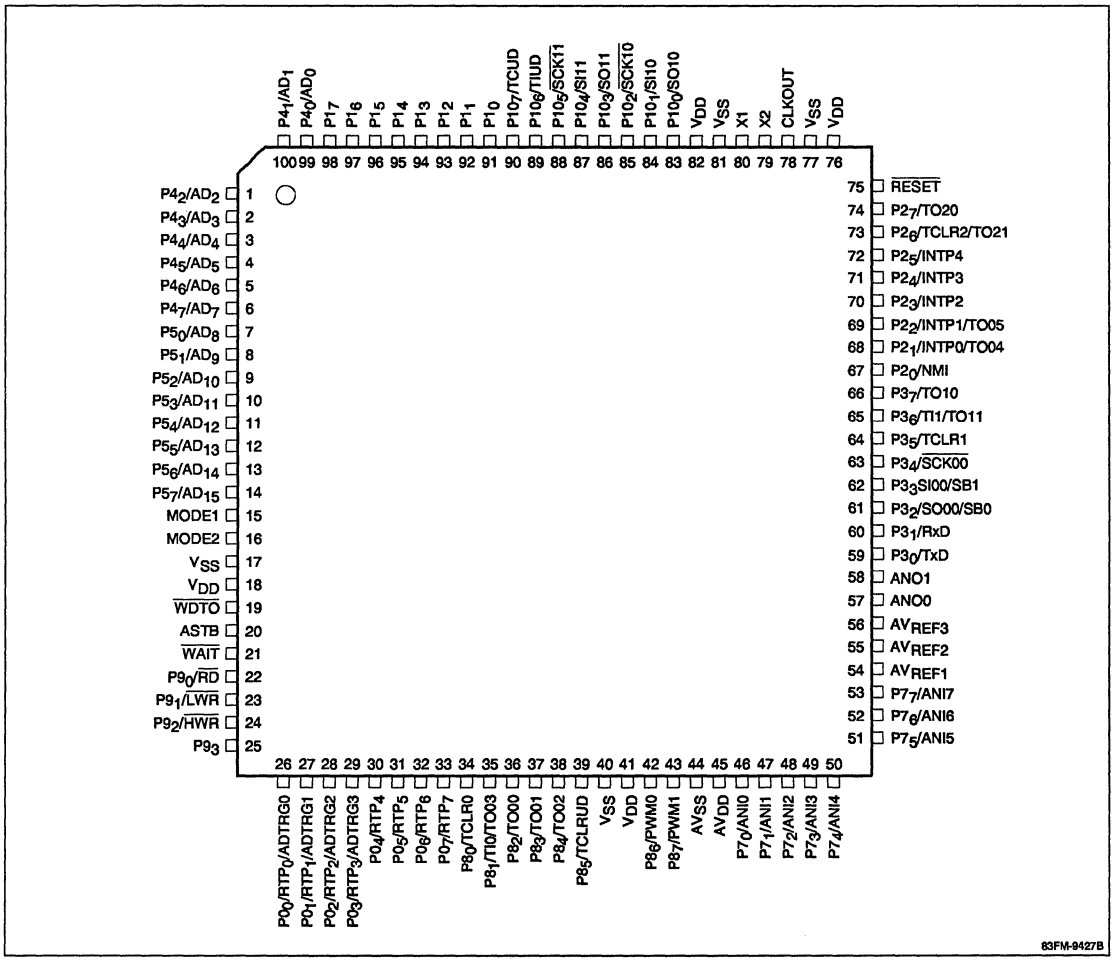
Ordering Information

Part Number	Package	ROM
μPD78355GC-7EA	100-pin plastic QFP (Dwg P100GC-50-7EA)	ROMless
μPD78356GC-xxx-7EA	100-pin plastic QFP (Dwg P100GC-50-7EA)	48K mask ROM
μPD78P356GC-7EA	100-pin plastic QFP (Dwg P100GC-50-7EA)	48K OTP ROM
μPD78P356KP-S	120-pin ceramic LCC with window (Dwg X120KW-80A)	48K UV EPROM

xxx indicates ROM code suffix.

Pin Configurations

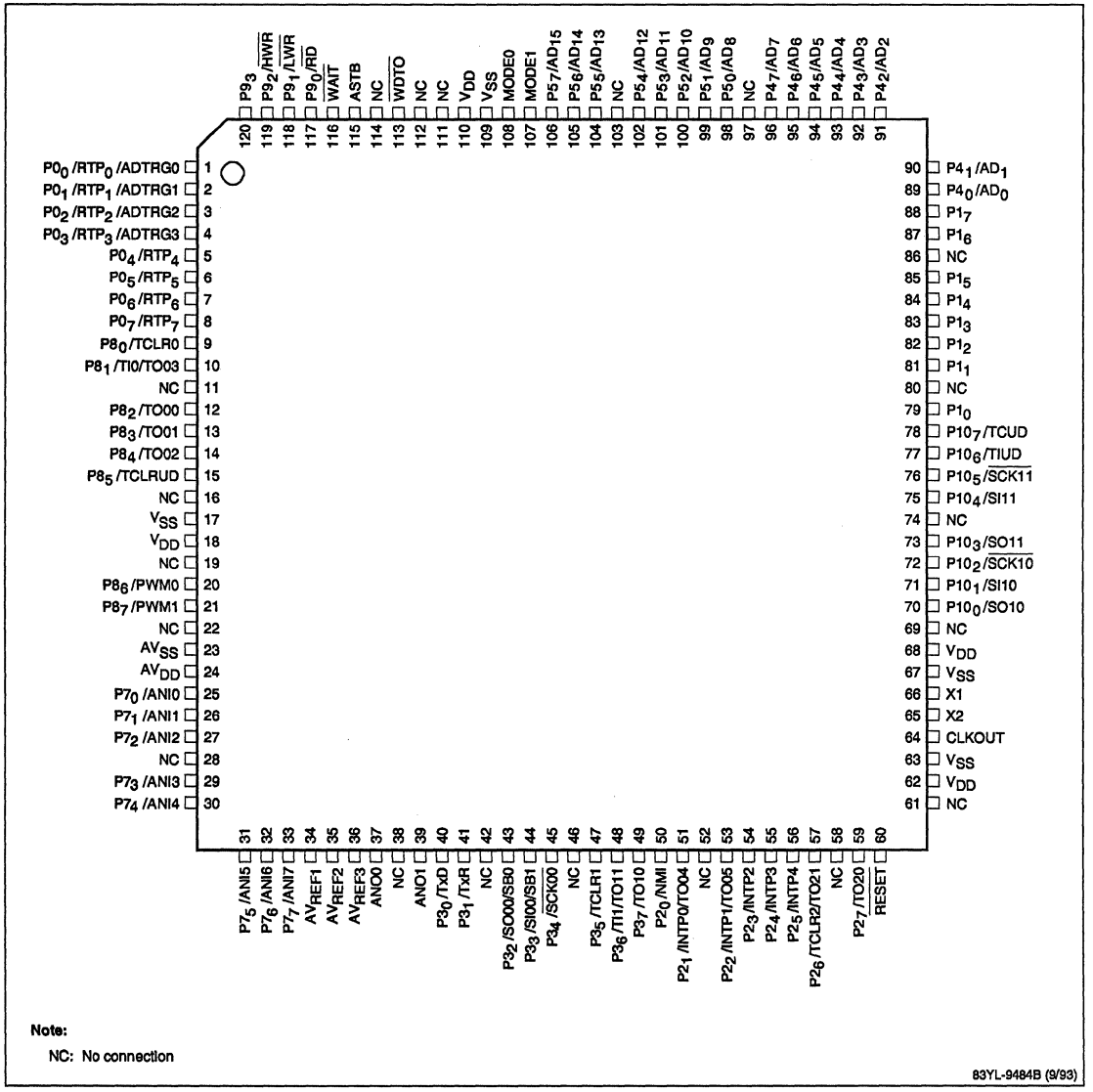
100-Pin Plastic QFP



5d

Pin Configurations (cont)

120-Pin Ceramic LCC



Note:

NC: No connection

83YL-9484B (9/93)

Pin Functions; Normal Operating Mode

Pin Name	Function	Alternate Pin Name	Alternate Function
$P0_0$	Port 0; 8-bit, bit-selectable I/O port	RTP_0 ADTRG0	Real-time output port External trigger input for A/D converter
$P0_1$		RTP_1 ADTRG1	Real-time output port External trigger input for A/D converter
$P0_2$		RTP_2 ADTRG2	Real-time output port External trigger input for A/D converter
$P0_3$		RTP_3 ADTRG3	Real-time output port External trigger input for A/D converter
$P0_4$ - $P0_7$		RTP_4 - RTP_7	Real-time output port
$P1_0$ - $P1_7$	Port 1; 8-bit, bit selectable I/O port		
$P2_0$	Port 2; 8-bit, bit-selectable I/O port ($P2_0$ is input only)	NMI	External nonmaskable interrupt
$P2_1$		INTP0 TO04	External maskable interrupt Timer output from real-time pulse unit
$P2_2$		INTP1 TO05	External maskable interrupt Timer output from real-time pulse unit
$P2_3$		INTP2	External maskable interrupt
$P2_4$		INTP3	External maskable interrupt
$P2_5$		INTP4	External maskable interrupt
$P2_6$		TCLR2 TO21	Clear input to real-time pulse unit Timer output from real-time pulse unit
$P2_7$		TO20	Timer output from real-time pulse unit
$P3_0$	Port 3; 8-bit, bit-selectable I/O port	TxD	Asynchronous serial transmit data output
$P3_1$		RxD	Asynchronous serial receive data input
$P3_2$		SO00 SB0	Serial data output; three-wire serial I/O mode I/O bus for NEC serial bus interface mode
$P3_3$		SI00 SB1	Serial data input; three-wire serial I/O mode I/O bus for NEC serial bus interface mode
$P3_4$		$\overline{SCK00}$	Serial clock I/O for synchronous serial interface
$P3_5$		TCLR1	Clear input to real-time pulse unit
$P3_6$		TI1 TO11	External clock to timer 1 Timer output from real-time pulse unit
$P3_7$		TO10	Timer output from real-time pulse unit
$P4_0$ - $P4_7$	Port 4; 8-bit, byte-selectable I/O port (78356/P356)	AD_0 - AD_7	Low-order 8 bits of external multiplexed address/data bus
$P5_0$ - $P5_7$	Port 5; 8-bit, bit-selectable I/O port (78356/P356)	AD_8 - AD_{15}	High-order 8 bits of external multiplexed address/data bus
$P7_0$ - $P7_7$	Port 7; 8-bit input port	ANI0 - ANI7	Analog inputs to A/D converter

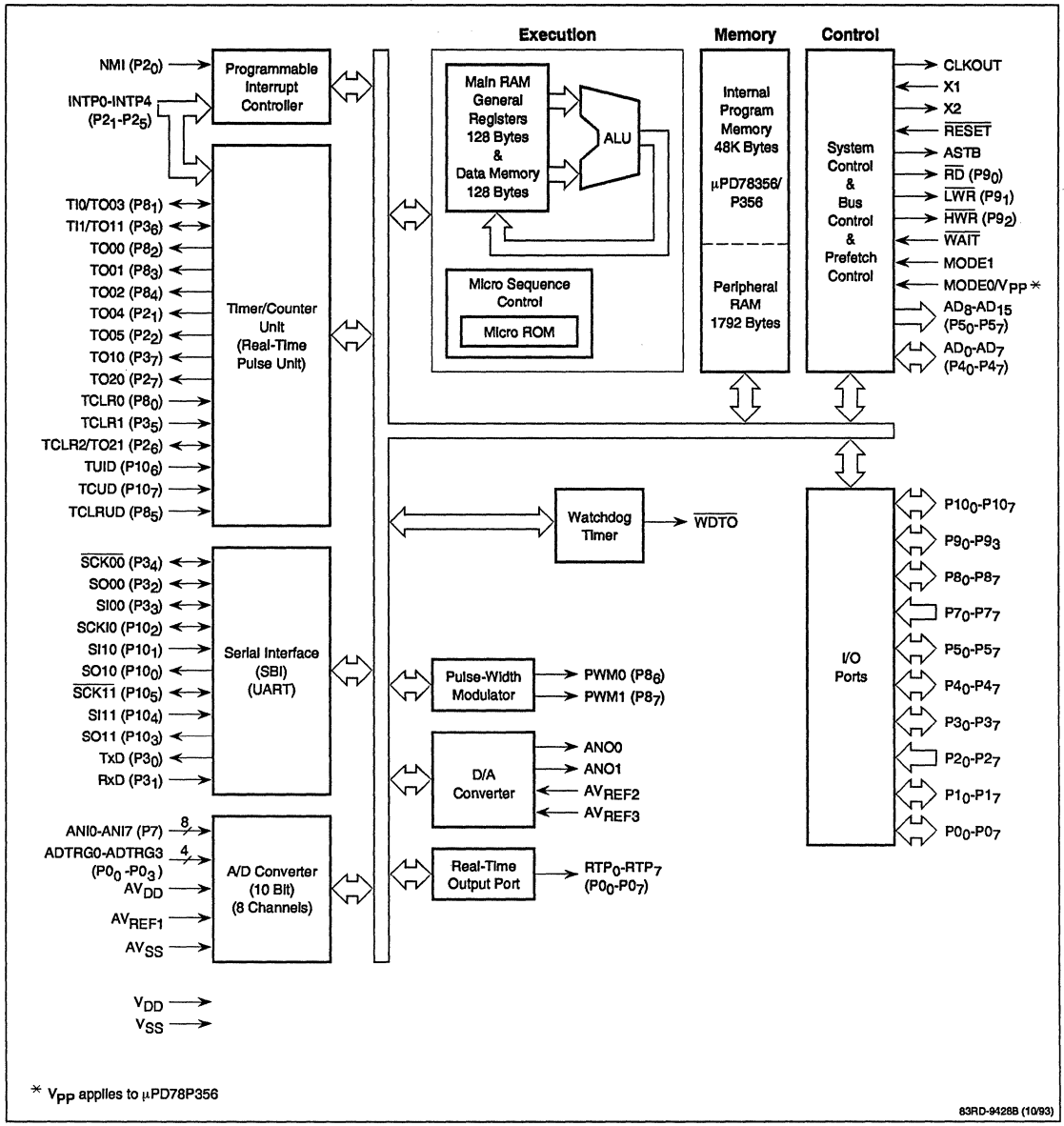
Pin Functions; Normal Operating Mode (cont)

Pin Name	Function	Alternate Pin Name	Alternate Function
$P8_0$	Port 8; 8-bit, bit-selectable I/O port	TCLR0	Clear input to real-time pulse unit
$P8_1$		TI0 TO03	External count clock to timer 0 Timer output from real-time pulse unit
$P8_2$		TO00	Timer output from real-time pulse unit
$P8_3$		TO01	Timer output from real-time pulse unit
$P8_4$		TO02	Timer output from real-time pulse unit
$P8_5$		TCLRUD	Clear input to real-time pulse unit
$P8_6$		PWM0	Pulse-width modulated output
$P8_7$		PWM1	Pulse-width modulated output
$P9_0$	Port 9; 4-bit, bit-selectable I/O port ($P9_0$ to $P9_2$ on 78356/P356)	$\overline{RD}$	External memory read strobe output
$P9_1$		$\overline{LWR}$	External memory write strobe output to low-order 8-bits in memory
$P9_2$		$\overline{HWR}$	External memory write strobe output to high-order 8-bits in memory
$P9_3$			
$P10_0$	Port 10; 8-bit, bit-selectable I/O port	SO10	Serial data output; three-wire serial I/O mode
$P10_1$		SI10	Serial data input; three-wire serial I/O mode
$P10_2$		$\overline{SCK10}$	Serial clock I/O for synchronous serial interface
$P10_3$		SO11	Serial data output; three-wire serial I/O mode
$P10_4$		SI11	Serial data input; three-wire serial I/O mode
$P10_5$		$\overline{SCK11}$	Serial clock I/O for synchronous serial interface
$P10_6$		TIUD	External clock for up/down counter
$P10_7$		TCUD	Up/down counter count direction control signal
ANO0, ANO1	Analog outputs from D/A converter		
ASTB	Address strobe output; used to latch address for external memory		
CLKOUT	Output of the system clock		
MODE0, MODE1	Set MODE0 and MODE1 to V_{SS} to access internal program memory on 78356/P356. If all program memory is external, set MODE0 to V_{DD} and MODE1 to to V_{SS} for an 8-bit data bus. Or set both to V_{DD} for a 16-bit data bus. To place 78P356 in programming mode, set MODE0 to V_{DD} and MODE1 to V_{SS}. The level of this pin cannot be changed during normal operation.		
NC	Pins labeled NC are not internally connected and may be connected to V_{SS}		
$\overline{RESET}$	External system reset input		
$\overline{WAIT}$	A low-level input adds wait states to the external bus cycle		
$\overline{WDTO}$	Open-drain output from the watchdog timer		

Pin Functions; Normal Operating Mode (cont)

Pin Name	Function	Alternate Pin Name	Alternate Function
X1	Crystal connection or external clock input		
X2	Crystal connection or open for external clock		
AV_{DD}	A/D converter power input		
AV_{REF1}	A/D converter reference voltage high		
AV_{REF2}	D/A converter reference voltage high		
AV_{REF3}	D/A converter reference voltage low		
AV_{SS}	A/D converter ground		
V_{DD}	+5 volt power input		
V_{SS}	Ground		

Block Diagram; μPD78356 Family



* V_{PP} applies to μPD78P356

83RD-9428B (10/93)

FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processing unit (CPU) of the μPD78356 family features 16-bit arithmetic including 16 x 16-bit multiply, both unsigned and signed, and 32 x 16-bit unsigned divide (producing a 32-bit quotient and a 16-bit remainder). The signed multiply executes in 0.875 μs and the divide in 2.69 μs at 32 MHz.

Also, a multiply-and-accumulate instruction, MACW n, performs a signed multiply on factors from a pair of tables and sums the results in the 32-bit register AXDE. The total execution time for 10 terms is 13.44 μs at 32 MHz.

A subtract and accumulate absolute values instruction, SACW [DE+], [HL+], subtracts corresponding factors of two tables and calculates the sum of the absolute values of these subtractions. The total execution time for two tables of 10 terms each is 17.125 μs at 32 MHz.

A CALLT vector table and a CALLF area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction can access up to 32 subroutines through their addresses in the CALLT vector table. A 2-byte call instruction can access any routine beginning in a specific CALLF area.

The internal system clock (f_{CLK}) is generated by dividing the oscillator frequency by 2. Therefore, at the maximum oscillator frequency of 32 MHz, the clock is 16 MHz with a 62.5 ns clock cycle. Since instructions execute in two or more cycles, the minimum instruction time is 125 ns.

Internal RAM

The μPD78356 family has a maximum of 2048 bytes of internal RAM. The upper 256-byte area (FE00H-FEFFH) features high-speed data access of one data word per two internal system clocks and is known as "main RAM." The remainder (F700H-FDFFH) is accessed at the same speed as external memory (one word per three internal system clocks) and is known as "peripheral RAM." The μPD78356 family can be programmed to have 1K or 2K bytes of internal RAM. The general register banks and the macro service control words are stored in main RAM. The remainder of main RAM and any unused register bank locations are available for general storage.

Internal Program Memory

The 78356 contains up to 48K bytes of mask ROM and the 78P356 up to 48K bytes of UV EPROM or OTP ROM. The 78356/P356 can be programmed to have 16K, 24K, 32K, or 48K bytes of internal program memory by using the memory expansion mode register (MM). Instructions are fetched from this internal memory at a maximum rate of one word every two internal system clocks. The 78355 does not have internal program memory.

External Memory

The μPD78356 family has a 64K-byte address space. The 78356/P356 can access 0, 256, 4K, or 16K bytes of external memory in the area from C000H to F6FFH. External memory can be either ROM, RAM, or peripheral devices as required. The 78356/P356 can have an 8- or 16-bit wide external data bus with a 16-bit wide external address bus. The data bus size is specified on 16K-byte boundaries by the programmable wait control register (PWC). The upper 16K-block includes only external memory addresses C000H to F6FFH and the external SFR area, FFD0H to FFDFH.

For 8-bit data bus operation of the 78356/P356, data bits are multiplexed with low-order address bits at port 4. The high-order address bits are taken from port 5 as required. The memory mode register (MM) controls the size of the external memory. It can be programmed to use 0, 4, 6, or 8 bits from port 5 for the high-order address. Any remaining port 5 bits can be used for I/O. The address latch (ASTB), read ($\overline{RD}$), and write ($\overline{LWR}$) strobes are provided from port 9.

For 16-bit data bus operation of the 78356/P356, low-order data bits are multiplexed with low-order address bits at port 4. High-order data bits are multiplexed with high-order address bits at port 5. The address latch, read, and two write strobes ($\overline{LWR}$ and $\overline{HWR}$) are provided from port 9.

The 78355 does not have ports 4 and 5. The MODE0 and MODE1 pins specify whether the ROMless 78355 has an 8- or 16-bit data bus. When set for an 8-bit data bus, it has eight dedicated high-order address lines and eight multiplexed low-order address and data lines. When set for a 16-bit bus, it has 16 dedicated address/data lines. All memory below address F700H must be external.

Table 1 summarizes the operation of the μPD78356 family during word and byte accesses (on even or odd addresses) to external memory and the external SFR access area in 16-bit data bus mode.

Table 1. 16-Bit Bus Access Cycles

Addr	Operation	$\overline{RD}$ †	$\overline{LWR}$ †	$\overline{HWR}$ †	Data
Word Access					
Even	Write	1	0	0	AD_0 - AD_{15}
	Read	0	1	1	AD_0 - AD_{15}
Odd*	Write				
	1st byte	1	1	0	AD_8 - AD_{15}
	2nd byte	1	0	1	AD_0 - AD_7
	Read				
	1st byte	0	1	1	AD_8 - AD_{15}
	2nd byte	0	1	1	AD_0 - AD_7
Byte Access					
Even	Write	1	0	1	AD_0 - AD_7
	Read	0	1	1	AD_0 - AD_7
Odd	Write	1	1	0	AD_8 - AD_{15}
	Read	0	1	1	AD_8 - AD_{15}

* Word access to an odd address is accomplished by two byte accesses: 1st byte from odd address; 2nd byte from odd address plus 1.

† 1 = Inactive, 0 = Active

The programmable wait control register (PWC) also allows the programmer to specify one or two additional wait states if they are required for low-speed memory or external peripheral devices. These wait states are specified independently in 16K blocks and are applicable to internal ROM, external memory, and the external SFR access area. If additional wait states are required, an external WAIT pin is provided.

In addition, the width of the ASTB signal can be increased by one internal clock cycle to allow more precharge time for dynamic RAMs or more decoding time for addresses. This address wait can be enabled in 32K-byte blocks by using the PWC register and is also applicable to internal ROM, external memory, and the external SFR access area.

Program Fetch

The μPD78356 family allows opcode fetch in the area between 0000H and FDFFH; fetches from addresses F700H to FDFFH are always from the peripheral RAM. The μPD78356 family contains a 5-byte instruction prefetch queue. The bus control unit can fetch an instruction byte from memory during cycles in which the execution unit is not using the memory bus. If the instruction byte is fetched from internal memory, a minimum of two internal system clocks are required for each byte, and the queue can hold 5 bytes. If the instruction is fetched from external memory, a minimum of three internal system clocks are required from each byte, and the queue can hold 3 bytes.

Instructions can be fetched from internal memory in either high-speed or normal fetch cycle mode using the 16-bit bus. The PWC register is used to select the mode for each 16K block. In high-speed fetch cycle mode, two internal system clocks are required to fetch an instruction word from internal ROM/PROM or the peripheral RAM. In normal fetch cycle mode, each word fetched from internal ROM/PROM requires three, four, or five internal system clocks (address wait can also be included) depending on the setting of the PWC register.

Instructions can be fetched from external memory using either an 8- or 16-bit bus. Only normal fetch cycle mode is available and each byte or word fetched requires three, four, or five internal system clocks. One address wait state can also be included depending on the setting of the PWC register.

CPU Control Registers

Program Counter. The program counter is a 16-bit register that holds the address of the next instruction to be executed. After reset line goes high, the program counter is loaded with the address stored in locations 0000H and 0001H.

Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

CPU Control Word. The CPU control word (CCW) selects the origin of the interrupt vector and CALLT tables. If the TPF bit (bit 1) is zero, the origin is 0000H; if the TPF bit is one, the origin is 8000H. The CCW is a special function register located at address FFC1H. The addresses of the vectors for the $\overline{RESET}$ input, operation-code trap, and BRK instruction are fixed at 0000H, 003CH, and 003EH, respectively, and are not altered by the TPF bit.

Program Status Word. The program status word (PSW) is a 16-bit register containing flags that are set or reset depending on the results of an instruction. This register can be written to or read from 8 bits at a time. The high-order 8 bits are called the PSWH and the low-order 8 bits are called the PSWL. The individual flags can also be manipulated on a bit-by-bit basis. The assignment of PSW bits follows.

	7	6	5	4	3	2	1	0
PSWH	UF	RBS2	RBS1	RBS0	0	0	0	0

	7	6	5	4	3	2	1	0
PSWL	S	Z	RSS	AC	IE	P/V	0	CY

UF	User flag
RBS2-RBS0	Active register bank number
S	Sign flag (1 if last result was negative)
Z	Zero flag (1 if last result was zero)
RSS	Register set selection flag
AC	Auxiliary carry flag (carry out of 3 bit)
IE	Interrupt enable flag
P/V	Parity or arithmetic overflow flag
CY	Carry bit (or 1-bit accumulator for logic)

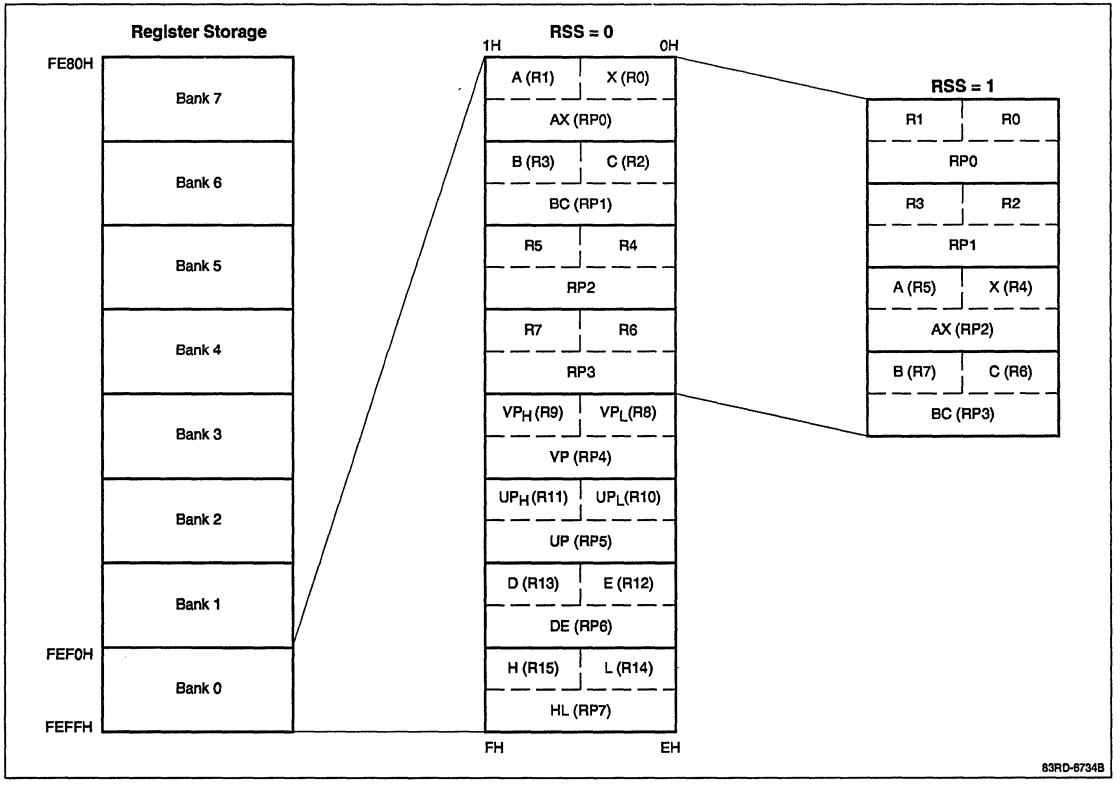
General Registers

There are sixteen 8-bit general registers, which can also be paired to function as 16-bit registers. A complete set of 16 registers is mapped into each of eight program-selectable register banks stored in main RAM. Three bits in the PSW specify the active register bank.

Registers have functional names (like A, X, B, C for 8-bit registers and AX, BC for 16-bit registers) and absolute names (like R1, R0, R3, R2 for 8-bit registers and RP0, RP1 for 16-bit registers). Each instruction determines whether a register is referred to by functional or absolute name and whether it is 8 or 16 bits.

Two possible relationships may exist between the absolute and functional names of the first four register pairs. The RSS bit in the PSW determines which of these is active at any time. The effect is that the accumulator and counter registers can be saved, and a new set can be specified by toggling the RSS bit. Figure 1 illustrates the general register configuration.

Figure 1. General Registers



5d

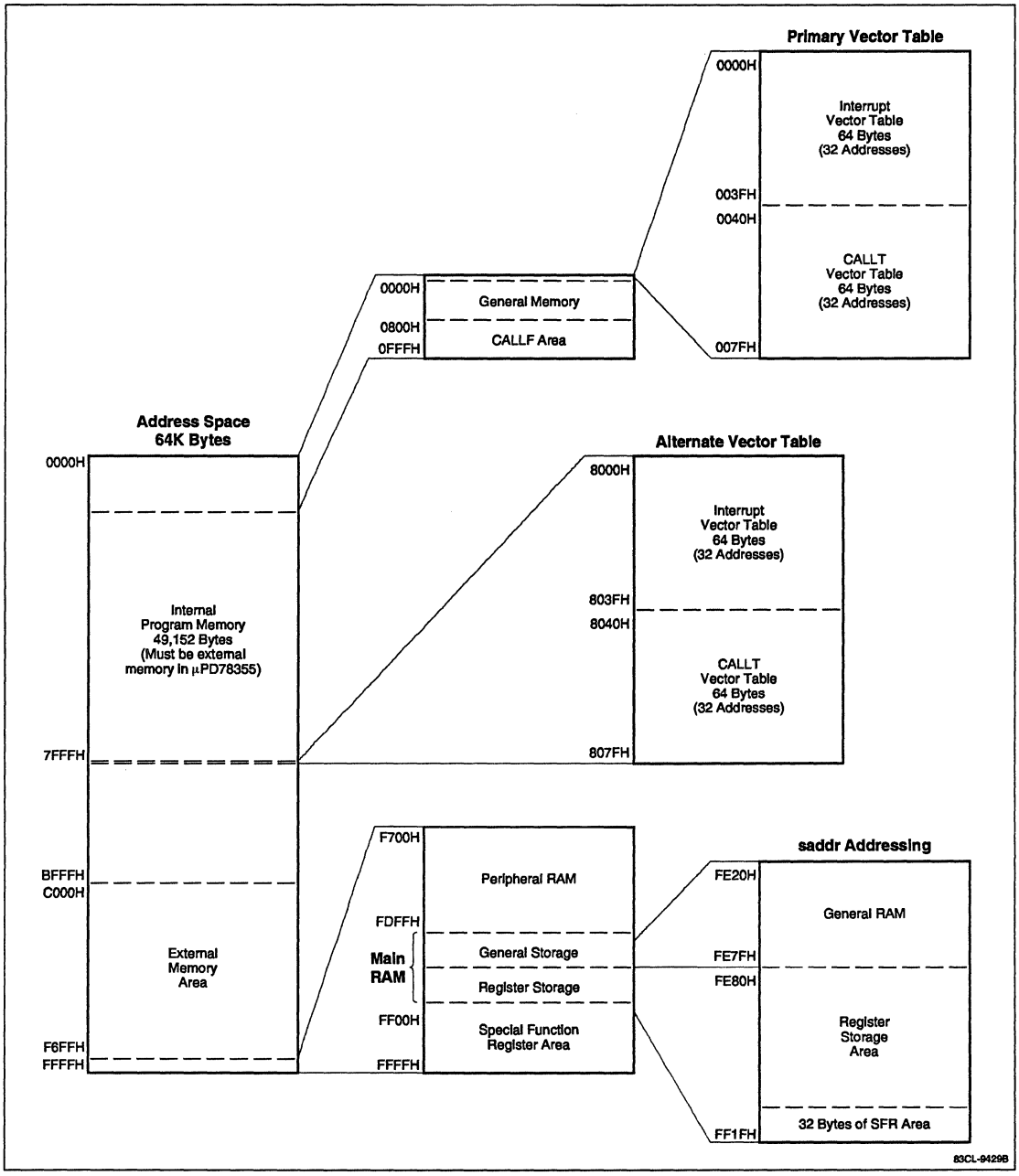
Addressing

The μPD78356 family features 1-byte addressing of both the special function registers and the portion of on-chip RAM from FE20H to FEFFH. The 1-byte sfr addressing accesses the entire SFR area, while the 1-byte saddr addressing accesses the first 32 bytes of the SFR area and 224 bytes of the main RAM.

The 16-bit SFRs and words of memory in these areas can be addressed by 1-byte saddrp addressing, which is valid for even addresses only. Since many instructions use 1-byte addressing, access to these locations is almost as fast and as versatile as access to the general registers.

There are nine addressing modes for data in main memory: direct, register, register indirect with autoincrement or autodecrement, saddr, saddr indirect, SFR, based, indexed, and based indexed. There are also 8-bit and 16-bit immediate operands. Figure 2 is the memory map of the μPD78356 family.

Figure 2. Memory Map



5d

Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and the CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by 1-byte SFR addressing. All can be read under program control, and most can also be written. They are either 8 or 16 bits, as required, and many of the 8-bit registers are capable of single-bit access as well.

Locations FFD0H through FFDFH are known as the external access area. Registers in external circuitry, interfaced and mapped to these addresses, can be addressed with SFR addressing. Table 2 lists the special function registers.

Table 2. Special Function Registers

				Access Units (Bits)			
Address	**Register (SFR)**	**Symbol**	**R/W**	**1**	**8**	**16**	**State After Reset**
FF00H	Port 0	P0	R/W	x	x	—	Undefined
FF01H	Port 1	P1	R/W	x	x	—	Undefined
FF02H	Port 2 (Note 1)	P2	R/W	x	x	—	Undefined
FF03H	Port 3	P3	R/W	x	x	—	Undefined
FF04H	Port 4	P4	R/W	x	x	—	Undefined
FF05H	Port 5	P5	R/W	x	x	—	Undefined
FF07H	Port 7	P7	R	x	x	—	Undefined
FF08H	Port 8	P8	R/W	x	x	—	Undefined
FF09H	Port 9	P9	R/W	x	x	—	Undefined
FF0AH	Port 10	P10	R/W	x	x	—	Undefined
FF10H-FF11H	Capture/compare register 00	CC00	R/W	—	—	x	Undefined
FF12H-FF13H	Capture/compare register 01	CC01	R/W	—	—	x	Undefined
FF14H-FF15H	Capture/compare register 02	CC02	R/W	—	—	x	Undefined
FF16H-FF17H	Capture/compare register 30	CC30	R/W	—	—	x	Undefined
FF18H-FF19H	Capture/compare register 31	CC31	R/W	—	—	x	Undefined
FF1AH-FF1BH	Compare register 00	CM00	R/W	—	—	x	Undefined
FF1CH-FF1DH	Compare register 01	CM01	R/W	—	—	x	Undefined
FF1EH-FF1FH	Compare register 02	CM02	R/W	—	—	x	Undefined
FF20H	Port 0 mode register	PM0	R/W	x	x	—	FFH
FF21H	Port 1 mode register	PM1	R/W	x	x	—	FFH
FF22H	Port 1 mode register (Note 2)	PM2	R/W	x	x	—	FFH
FF23H	Port 3 mode register	PM3	R/W	x	x	—	FFH
FF25H	Port 5 mode register	PM5	R/W	x	x	—	FFH
FF28H	Port 8 mode register	PM8	R/W	x	x	—	FFH
FF29H	Port 9 mode register	PM9	R/W	x	x	—	0FH
FF2AH	Port 10 mode register	PM10	R/W	x	x	—	FFH
FF30H-FF31H	Timer register 0	TM0	R	—	—	x	00H
FF32H-FF33H	Timer register 1	TM1	R	—	—	x	00H
FF34H-FF35H	Timer register 2	TM2	R	—	—	x	00H
FF36H-FF37H	Timer register 3	TM3	R	—	—	x	00H
FF38H-FF39H	Timer register 4	TM4	R	—	—	x	00H

Table 2. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF3AH-FF3BH	Presettable up/down counter register	UDC	R/W	—	—	x	00H
FF3CH	External interrupt mode register 0	INTM0	R/W	x	x	—	00H
FF3DH	External interrupt mode register 1	INTM1	R/W	x	x	—	00H
FF40H	Port 0 mode control register	PMC0	R/W	x	x	—	00H
FF42H	Port 2 mode control register (Note 3)	PMC2	R/W	x	x	—	01H
FF43H	Port 3 mode control register	PMC3	R/W	x	x	—	00H
FF44H	Pullup resistor option register L	PUOL	R/W	x	x	—	00H
FF45H	Pullup resistor option register H	PUOH	R/W	x	x	—	00H
FF48H	Port 8 mode control register	PMC8	R/W	x	x	—	00H
FF4AH	Port 10 mode control register	PMC10	R/W	x	x	—	00H
FF50H-FF51H	Compare register 03	CM03	R/W	—	—	x	Undefined
FF52H-FF53H	Compare register 10	CM10	R/W	—	—	x	Undefined
FF54H-FF55H	Compare register 11	CM11	R/W	—	—	x	Undefined
FF56H-FF57H	Compare register 20	CM20	R/W	—	—	x	Undefined
FF58H-FF59H	Compare register 21	CM21	R/W	—	—	x	Undefined
FF5AH-FF5BH	Compare register 40	CM40	R/W	—	—	x	Undefined
FF5CH-FF5DH	Up/down counter compare register 0	CMUD0	R/W	—	—	x	Undefined
FF5EH-FF5FH	Up/down counter compare register 1	CMUD1	R/W	—	—	x	Undefined
FF60H	Real-time output port register L	RTPL	R/W	x	x	—	Undefined
FF61H	Real-time output port register H	RTPH	R/W	x	x	—	Undefined
FF62H	Port read control register	PRDC	R/W	x	x	—	00H
FF63H	Real-time output port mode register	RTPM	R/W	x	x	—	00H
FF68H	A/D converter mode register 0	ADM0	R/W	x	x	—	00H
FF69H	A/D converter mode register 1	ADM1	R/W	x	x	—	07H
FF6AH	D/A conversion setup register 0	DACS0	R/W	x	x	—	00H
FF6BH	D/A conversion setup register 1	DACS1	R/W	x	x	—	00H
FF6AH-FF6BH	D/A conversion setup register	DACS	R/W	—	—	x	0000H
FF70H	Timer unit mode register 0	TUM0	R/W	x	x	—	00H
FF71H	Timer unit mode register 1	TUM1	R/W	x	x	—	00H
FF72H	Timer unit mode register 2	TUM2	R/W	x	x	—	00H
FF73H	Timer unit mode register 3	TUM3	R/W	x	x	—	00H
FF74H	Timer control register 0	TMC0	R/W	x	x	—	00H
FF75H	Timer control register 1	TMC1	R/W	x	x	—	00H
FF76H	Timer control register 2	TMC2	R/W	x	x	—	04H
FF77H	Up/down counter control register	UDCC	R/W	x	x	—	00H
FF78H	Timer output control register 0	TOC0	R/W	x	x	—	00H
FF79H	Timer output control register 1	TOC1	R/W	x	x	—	00H
FF7AH	Timer output control register 2	TOC2	R/W	x	x	—	00H
FF7BH	Timer overflow status register (Note 4)	TOVS	R/W	x	x	—	00H
FF7CH	Noise protection control register	NPC	R/W	x	x	—	00H

5d

Table 2. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF80H	Clock synchronous serial interface mode register 0	CSIM0	R/W	x	x	—	00H
FF82H	Serial bus interface control register (Note 5)	SBIC	R/W	x	x	—	00H
FF86H	Serial I/O shift register 0	SIO0	R/W	x	x	—	Undefined
FF88H	Asynchronous serial interface mode register	ASIM	R/W	x	x	—	80H
FF8AH	Asynchronous serial interface status register	ASIS	R	x	x	—	00H
FF8CH	Serial reception buffer: UART	RxB	R	—	x	—	Undefined
FF8EH	Serial transmission shift register: UART	TxS	W	—	x	—	Undefined
FF90H	Clock synchronous serial interface mode register 1	CSIM1	R/W	x	x	—	00H
FF96H	Serial I/O shift register 1	SIO1	R/W	x	x	—	Undefined
FFA0H	PWM control register	PWMC	R/W	x	x	—	00H
FFA2H	PWM register 0L	PWM0L	R/W	x	x	—	Undefined
FFA2H-FFA3H	PWM register 0	PWM0	R/W	—	—	x	Undefined
FFA4H	PWM register 1L	PWM1L	R/W	x	x	—	Undefined
FFA4H-FFA5H	PWM register 1	PWM1	R/W	—	—	x	Undefined
FFA8H	Inservice priority register	ISPR	R	x	x	—	00H
FFAAH	Interrupt mode control register	IMC	R/W	x	x	—	80H
FFACH	Interrupt mask register 0L	MK0L	R/W	x	x	—	FFH
FFACH-FFADH	Interrupt mask register 0	MK0	R/W	—	—	x	FFFFH
FFADH	Interrupt mask register 0H	MK0H	R/W	x	x	—	FFH
FFAEH	Interrupt mask register 1L	MK1L	R/W	x	x	—	FFH
FFAEH-FFAFH	Interrupt mask register 1	MK1	R/W	—	—	x	00FFH
FFB0H-FFB1H	A/D conversion result register 0	ADCR0	R	—	—	x	Undefined
FFB1H	A/D conversion result register 0H	ADCR0H	R	—	x	—	Undefined
FFB2H-FFB3H	A/D conversion result register 1	ADCR1	R	—	—	x	Undefined
FFB3H	A/D conversion result register 1H	ADCR1H	R	—	x	—	Undefined
FFB4H-FFB5H	A/D conversion result register 2	ADCR2	R	—	—	x	Undefined
FFB5H	A/D conversion result register 2H	ADCR2H	R	—	x	—	Undefined
FFB6H-FFB7H	A/D conversion result register 3	ADCR3	R	—	—	x	Undefined
FFB7H	A/D conversion result register 3H	ADCR3H	R	—	x	—	Undefined
FFB8H-FFB9H	A/D conversion result register 4	ADCR4	R	—	—	x	Undefined
FFB9H	A/D conversion result register 4H	ADCR4H	R	—	x	—	Undefined
FFBAH-FFBBH	A/D conversion result register 5	ADCR5	R	—	—	x	Undefined
FFBBH	A/D conversion result register 5H	ADCR5H	R	—	x	—	Undefined
FFBCH-FFBDH	A/D conversion result register 6	ADCR6	R	—	—	x	Undefined
FFBDH	A/D conversion result register 6H	ADCR6H	R	—	x	—	Undefined
FFBEH-FFBFH	A/D conversion result register 7	ADCR7	R	—	—	x	Undefined
FFBFH	A/D conversion result register 7H	ADCR7H	R	—	x	—	Undefined
FFC0H	Standby control register (Note 6)	STBC	R/W	—	x	—	0000 x000B
FFC1H	CPU control word	CCW	R/W	x	x	—	00H

Table 2. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FFC2H	Watchdog timer mode register (Note 6)	WDM	R/W	—	x	—	00H
FFC4H	Memory expansion mode register	MM	R/W	x	x	—	00H
FFC6H-FFC7H	Programmable wait control register	PWC	R/W	—	—	x	C0AAH
FFD0H-FFDFH	External SFR area	—	R/W	x	x	—	Undefined
FFE0H	Interrupt control register (INTOV0)	OVIC0	R/W	x	x	—	43H
FFE1H	Interrupt control register (INTOV3)	OVIC3	R/W	x	x	—	43H
FFE2H	Interrupt control register (INTP0/INTCC00)	PIC0	R/W	x	x	—	43H
FFE3H	Interrupt control register (INTP1/INTCC01)	PIC1	R/W	x	x	—	43H
FFE4H	Interrupt control register (INTP2/INTCC02)	PIC2	R/W	x	x	—	43H
FFE5H	Interrupt control register (INTP3/INTCC30)	PIC3	R/W	x	x	—	43H
FFE6H	Interrupt control register (INTP4/INTCC31)	PIC4	R/W	x	x	—	43H
FFE7H	Interrupt control register (INTCM00)	CMIC00	R/W	x	x	—	43H
FFE8H	Interrupt control register (INTCM01)	CMIC01	R/W	x	x	—	43H
FFE9H	Interrupt control register (INTCM02)	CMIC02	R/W	x	x	—	43H
FFEAH	Interrupt control register (INTCM03)	CMIC03	R/W	x	x	—	43H
FFEBH	Interrupt control register (INTCM10)	CMIC10	R/W	x	x	—	43H
FFECH	Interrupt control register (INTCM11)	CMIC11	R/W	x	x	—	43H
FFEDH	Interrupt control register (INTCM20)	CMIC20	R/W	x	x	—	43H
FFEEH	Interrupt control register (INTCM21)	CMIC21	R/W	x	x	—	43H
FFEFH	Interrupt control register (INTCM40)	CMIC40	R/W	x	x	—	43H
FFF0H	Interrupt control register (INTCMUD0)	CMICUD0	R/W	x	x	—	43H
FFF1H	Interrupt control register (INTCMUD1)	CMICUD1	R/W	x	x	—	43H
FFF2H	Interrupt control register (INTSER)	SERIC	R/W	x	x	—	43H
FFF3H	Interrupt control register (INTSR)	SRIC	R/W	x	x	—	43H
FFF4H	Interrupt control register (INTST)	STIC	R/W	x	x	—	43H
FFF5H	Interrupt control register (INTCSI0)	CSIIC0	R/W	x	x	—	43H
FFF6H	Interrupt control register (INTCSI1)	CSIIC1	R/W	x	x	—	43H
FFF7H	Interrupt control register (INTAD)	ADIC	R/W	x	x	—	43H

Notes:

(1) P2 bit 0 is read only

(2) PM2 bit 0 is always 1.

(3) PMC2 bit 0 is always 1.

(4) TOVS bits 6 and 7 are always 0; bits 1, 2, 4, and 5 are read/write; bits 0 and 3 are read only.

(5) SBIC bits 5 and 7 are read/write; bits 2, 3, and 6 are read only; bits 0, 1, and 4 are write only.

(6) Protected register that can be written by a special instruction only.

Input/Output Ports

The 78355 has a total of 57 I/O lines. The 78356/P356 have an additional 19 for a total of 76 I/O lines. Ports P0, P1, P3, P8, and P10 are tri-state, 8-bit input/output ports and P7 is an 8-bit input port. Port P2 is an 8-bit I/O port with pin $P2_0$ always an input. Bit 3 of port 9 is available as an I/O line at all times. All the I/O bits in P0 to P3 and P8 to P10 can be individually selected for either input or output.

Each pin of P2 can be programmed for rising or falling edge detection; pins 1 to 7 can also be programmed for both rising and falling edge detection.

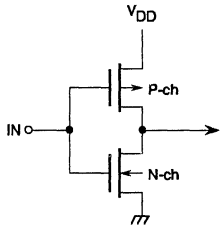
Software programmable internal pullup resistors are available at each pin of ports P0 to P3 and P8 to P10 except $P2_0$. These resistors are enabled on a port basis for all I/O pins set to input mode.

The output level of the P0 to P3 and P8 to P10 I/O pins can be tested to determine whether they agree with the contents of the output latch. When the low-order bit of port read control register PRDC is set to 1, the output level of the I/O pins can be read with the port still in the output mode. These data values can be compared with the data known to be in the output latch to determine if the port is functioning correctly. Figure 3 shows the structure of each port pin.

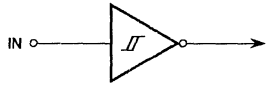
The 19 additional I/O lines in the 78356/P356 are P4, P5, and bits 0 to 2 of P9. These ports are available if external memory or memory-mapped external circuitry is not being used. Port 4 is shared with the low-order address/data bus (AD_0 to AD_7) and is byte-selectable for input or output. Port 5 is shared with the high-order address bus in 8-bit bus mode or the high-order address/data bus in 16-bit bus mode. Depending on the amount of external memory used, either 8, 4, 2, or 0 bits of port 5 are available for bit-selectable I/O. Port 9 is a 4-bit, bit-selectable I/O port. One of its pins is shared with the read strobe and two others with the high and low write strobes. Internal pullup resistors are available at each pin of port 4 and 5 when they are set to input mode.

Figure 3. I/O Circuits

Type 1. $\overline{\text{WAIT}}$, MODE0, MODE1

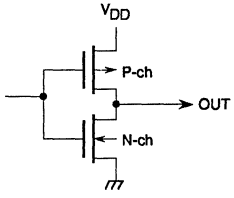


Type 2. $P2_0$, $\overline{\text{RESET}}$

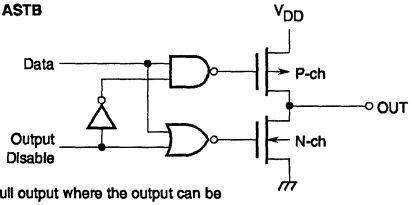


Schmitt trigger input with hysteresis characteristics.

Type 3. CLKOUT

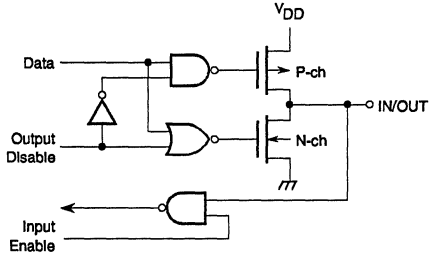


Type 4. ASTB

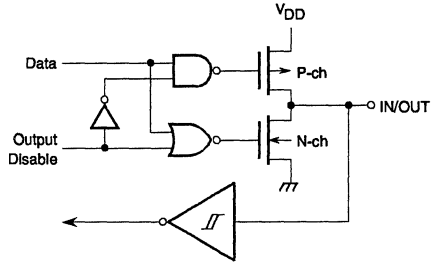


Push-pull output where the output can be placed in high impedance (Both the P and N channels are off.)

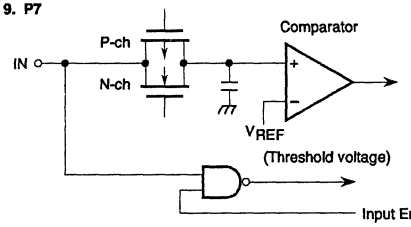
Type 5. $P0_4$-$P0_7$, P1, $P3_0$, $P3_1$, $P3_7$, P4, P5 $P8_2$-$P8_4$, $P8_6$, $P8_7$, P9, $P10_0$, $P10_3$



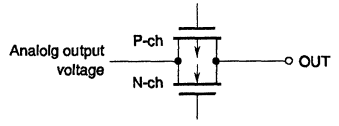
Type 8. $P0_0$-$P0_3$, $P2_1$-$P2_7$, $P3_2$-$P3_6$, $P8_0$, $P8_1$, $P8_5$, $P10_1$, $P10_2$, $P10_4$-$P10_7$



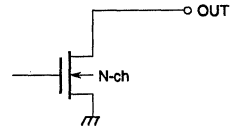
Type 9. P7



Type 12. ANO0, ANO1



Type 14. $\overline{\text{WDTO}}$



83CL-9430B (9/93)

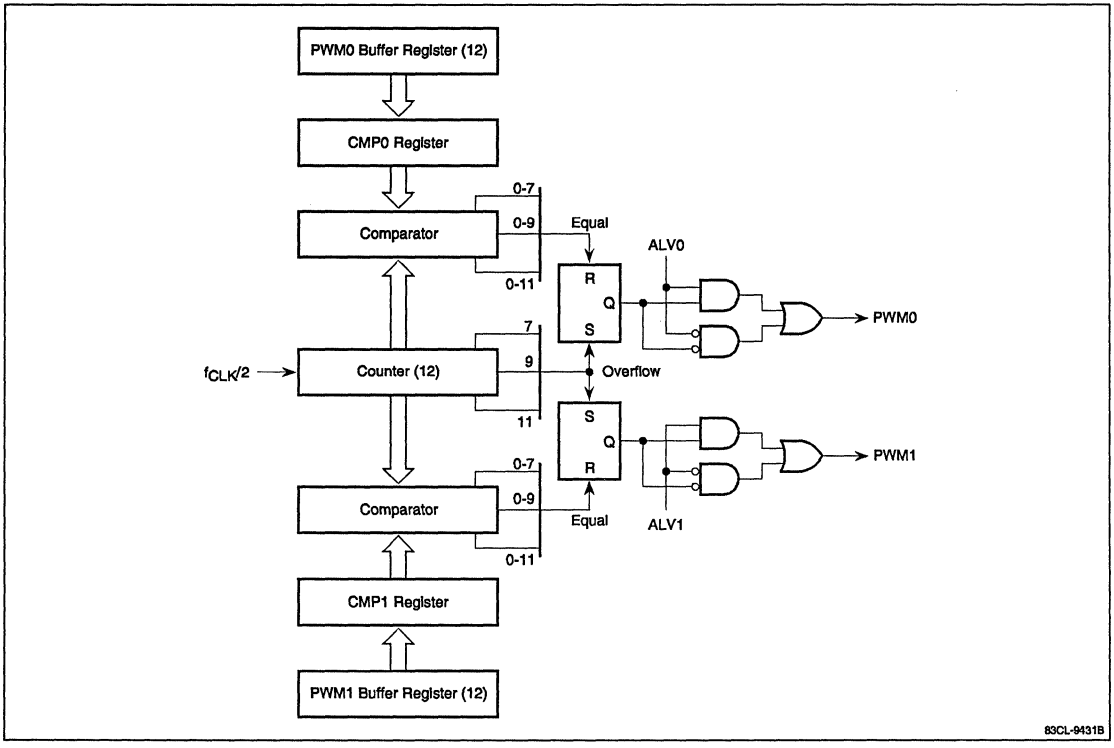
5d

Pulse-Width Modulated Outputs

The μPD78356 family has two high-speed, pulse-width modulated (PWM) outputs. A single 12-bit, free-running counter counts the internal system clock $f_{CLK}/2$ and serves both outputs. The resolution is 125 ns per bit at 32 MHz. By setting the counter and comparator to either 8, 10, or 12 bits, repetition rates of 31.2, 7.8, and 1.9 kHz, respectively, can be achieved.

The polarity of each output can be selected under program control. Whenever the counter overflows, the CMP0 and CMP1 registers are loaded from their respective PWM buffer registers and each output becomes active. When the counter value matches the value in the associated compare register, that output goes inactive. The two PWM outputs, PWM0 and PWM1, share pins with port 8 bits 6 and 7, respectively.

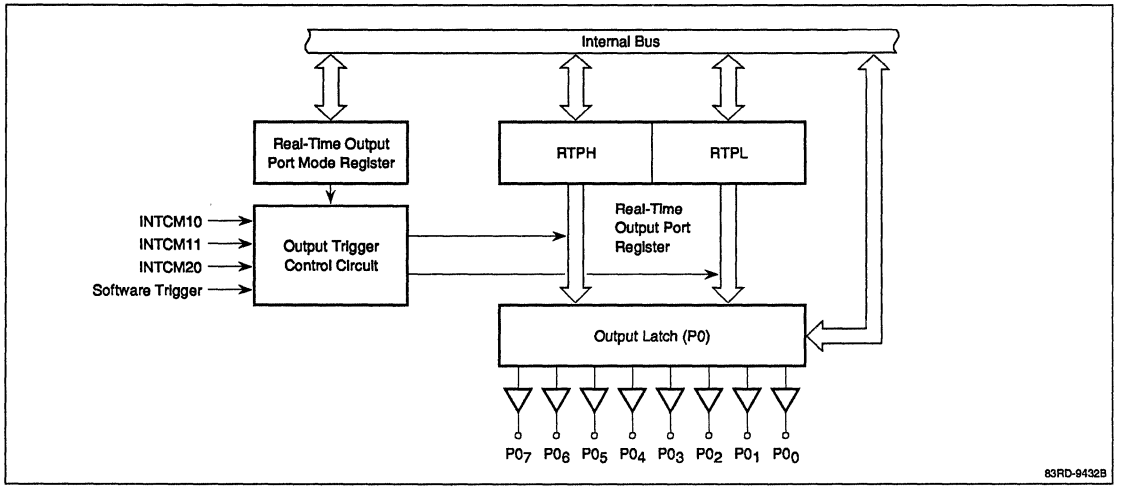
Figure 4. Pulse-Width Modulated Outputs



Real-Time Output Port

Port 0 can function on a bit-selectable basis as a real-time output port. See figure 5. Real-time port bits can be directly written under program control or they can be written under control of timing signals INTCM10, INTCM11, and INTCM20 generated by the real-time pulse unit. The latter method provides output timing that is independent of interrupt latency.

Figure 5. Real-Time Output Port

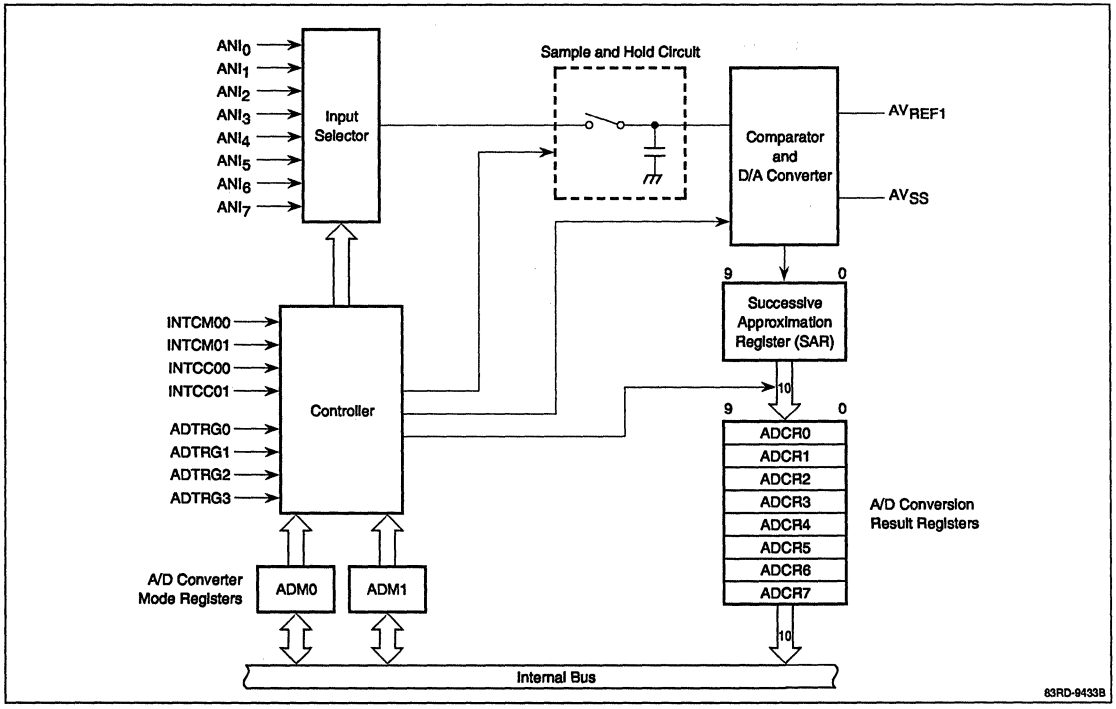


5d

A/D Converter

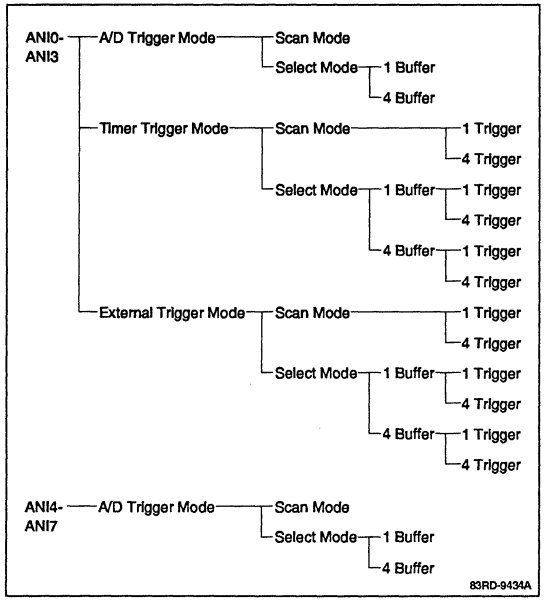
The analog-to-digital (A/D) converter (see figure 6) uses the successive-approximation method for converting up to eight multiplexed analog inputs into 10-bit digital data. The minimum conversion time per input is 2 μs at 32-MHz operation. There are eight 16-bit A/D conversion results registers (ADCR0 to ADCR7). During word access, the low-order 10-bits contain the result and the upper 6-bits are set to zero. During byte access, the high-order 8-bits of the 10-bit A/D conversion result are read. This converted data can be easily transferred to memory by the macro service function.

Figure 6. A/D Converter



The A/D converter of the μPD78356 family supports a wide variety of operating modes. See figure 7. A/D conversion can be started by one of three modes: A/D trigger mode, timer trigger mode, and external trigger mode. Analog inputs ANI0 to ANI3 can be started by all three modes; ANI4 to ANI7 can be started only by the A/D trigger mode. In A/D trigger mode, conversion is started by writing data into the A/D converter mode register 0 (ADM0) and proceeds automatically.

Figure 7. A/D Converter Operating Modes



In timer trigger mode, coincidence signals from compare registers CM00, CM01, CC00, and CC01 associated with timer 0 start the conversions. Either all four signals are used (four-trigger mode) or only the coincidence signal CM00 is used (one-trigger mode). The timing and sequence of the A/D conversions are controlled by these trigger signals. Conversions can be performed once or repeatedly.

In external trigger mode, four external signals ADTRG0 to ADTRG3 start the conversions. Either all four signals (four-trigger mode) or one signal ADTRG0 (one-trigger mode) starts the conversions. The timing and sequence of the A/D conversions are controlled by these trigger signals.

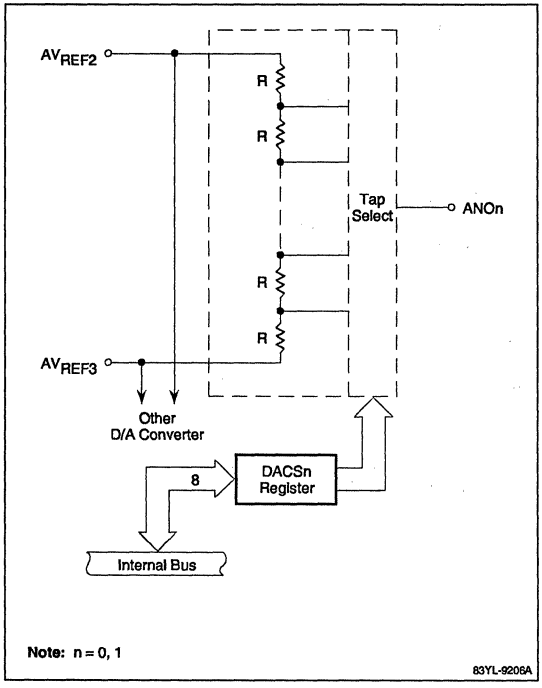
The A/D converter can operate in either scan mode or select mode. In scan mode, from one to eight analog inputs (ANI0 to ANI7) can be programmed for conversion. The A/D converter selects each of the inputs (order and timing sequence are based on the trigger mode), converts the data, and stores it in its associated A/D conversion result register (ADCR0 to ADCR7). An interrupt (INTAD) is generated when all inputs have been converted.

In select mode, only one of the eight A/D inputs can be selected for conversion. Data from this input can be stored in its associated ADCR register (one-buffer mode) or four values can be stored sequentially in registers ADCR0 to ADCR3 (four-buffer mode). In one-buffer mode, an interrupt (INTAD) is generated after each conversion. In four-buffer mode, an interrupt is generated after four conversions. The timing of the conversions depends on the trigger mode selected.

D/A Converter

The μPD78356 family has two digital-to-analog (D/A) converters as shown in figure 8. The 8-bit digital data, written to the D/A conversion setup registers (DASCn; n = 0, 1), selects one of the 256 taps on a resistor ladder between AV_{REF2} and AV_{REF3}. The selected voltage becomes the analog output at the ANOn pin. ANOn is a high-impedance output and requires an external buffer to drive a low-impedance load.

Figure 8. D/A Converter

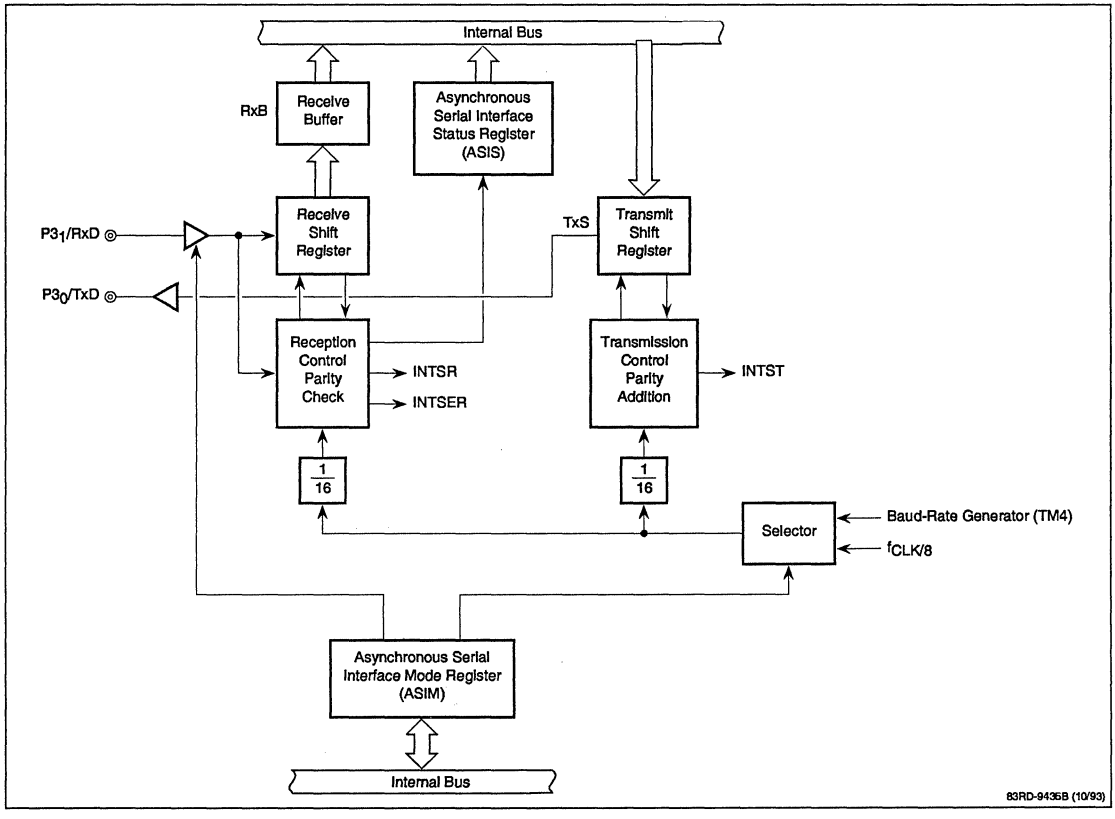


Serial Interfaces

The μPD78356 family has three independent serial interfaces: one asynchronous and two clock synchronous. All three share an internal baud-rate generator.

The asychrounous serial interface is a standard universal asynchronous receiver transmitter (UART). The UART (figure 9) permits full-duplex operation and can be programmed for 7 or 8 bits of data after the start bit, followed by one or two stop bits. Odd, even, zero, or no parity can also be selected. The serial clock for the UART is from the internal system clock divided by eight or from the internal baud-rate generator. The UART generates three interrupts: INTST (transmission complete), INTSR (reception complete), and INTSER (reception error).

Figure 9. Asynchronous Serial Interface



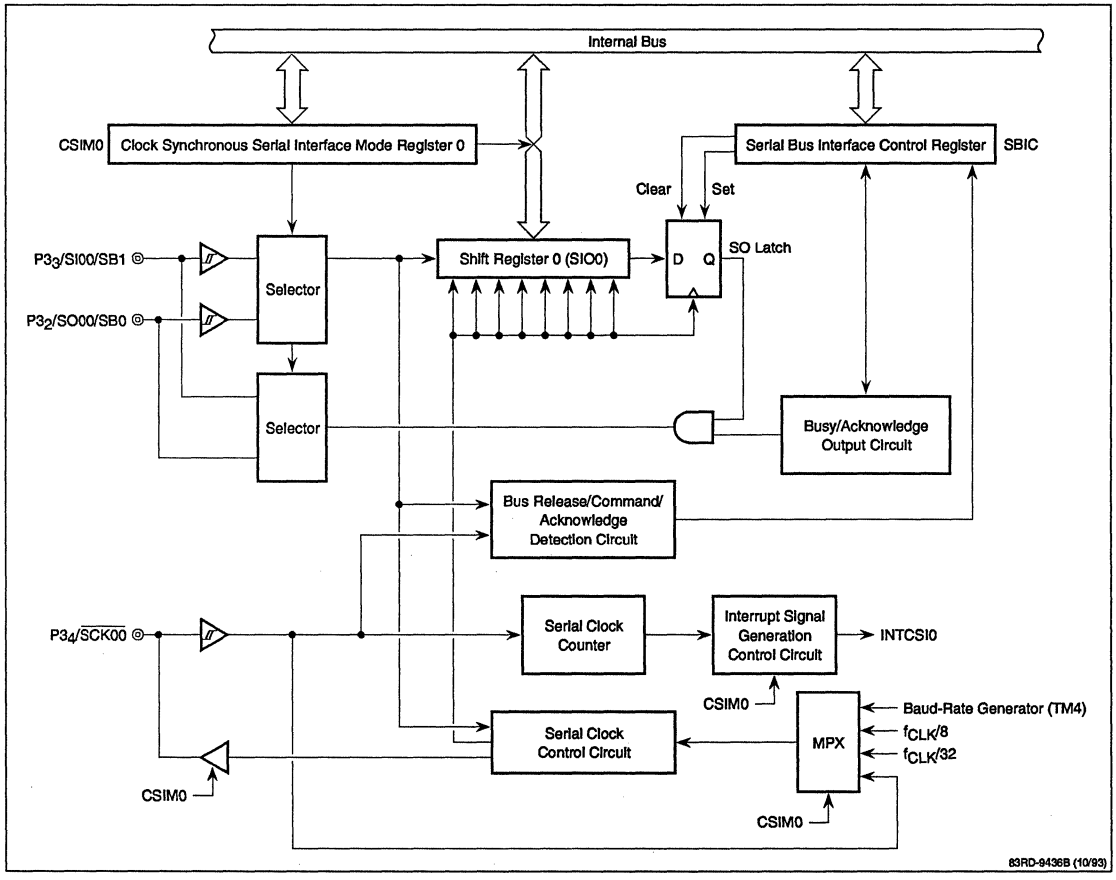
5d

Clock synchronous serial interface 0 (figure 10) is an 8-bit interface that operates in either a three-wire serial I/O mode or NEC serial bus interface (SBI) mode.

In the three-wire serial I/O mode, the 8-bit shift register (SIO) is loaded with a byte of data and eight clock pulses are generated. These eight pulses shift the byte of data out the SO line (either MSB or LSB first) and in from the SI line, providing full-duplex operation. This interface can also be set to receive or transmit data only. The INTCSI0 interrupt is generated after each 8-bit transfer. One of two internal clocks, an external clock, or the internal baud-rate generator clocks the data.

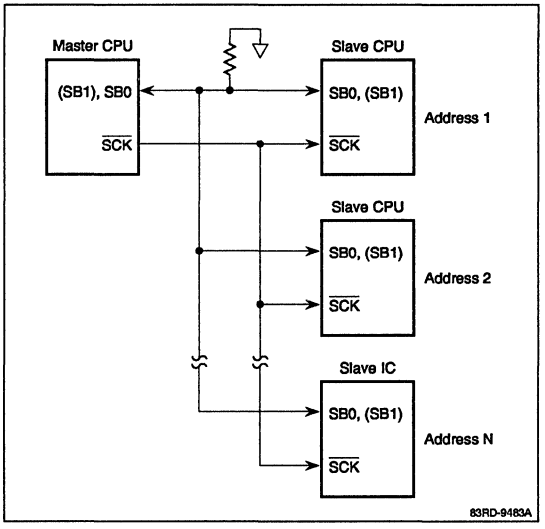
The NEC SBI mode is a two-wire, high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are connected in a master/slave configuration. See figure 11. There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over one of the serial bus lines (SB0 or SB1) using a fixed hardware protocol synchronized with the SCK line.

Figure 10. Clock Synchronous Serial Interface 0



Each slave 78355/356/P356 can be programmed in software to respond to any one of 256 addresses. There are also 256 commands and 256 data types. Since all commands are user-definable, any software protocol, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

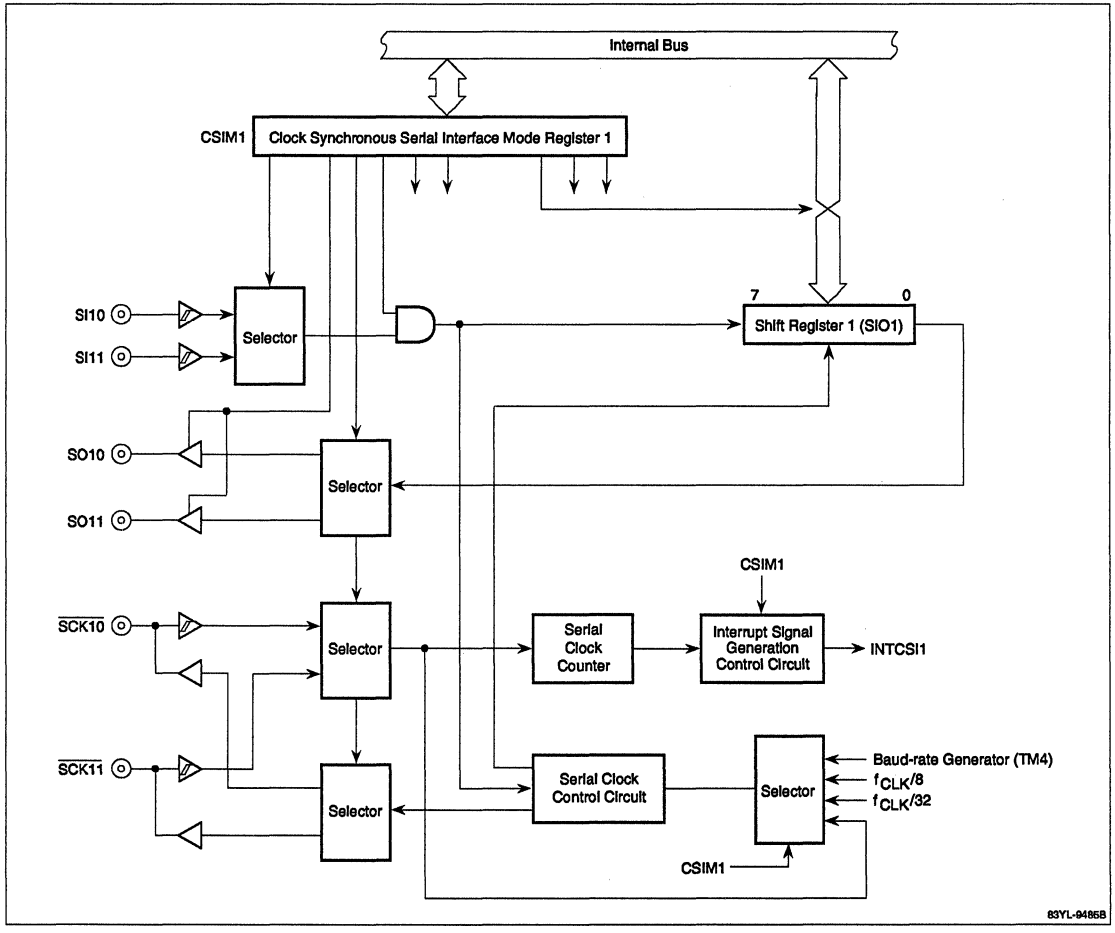
Figure 11. SBI Mode Master/Slave Configuration



Clock synchronous serial interface 1 (figure 12) with a pin switching function is also an 8-bit interface that operates only in the three-wire serial I/O mode. It can be switched under program control between two sets of I/O pins: SCK10, SO10, and SI10 or SCK11, SO11, and SI11. With the exception of this pin switching function, its operation in three-wire serial I/O mode is identical to clock synchronous interface 0.

5d

Figure 12. Clock Synchronous Serial Interface 1 With Pin Switching



A dedicated baud-rate generator can be programmed to provide a common serial clock to the asynchronous and clock synchronous serial interfaces. The baud-rate generator uses timer 4 and compare register CM40 of the real-time pulse unit to generate a serial clock from one of four internal clocks. By choosing the correct oscillator frequency, the baud-rate generator is capable of generating all the commonly used baud rates from 2400 to 154K b/s for the UART and 150 to 2M b/s for the clock synchronous interfaces.

Real-Time Pulse Unit

The real-time pulse unit (RPU, figure 13) can be used as an interval timer, to measure pulse widths and frequencies, to generate puse-width modulated outputs, to count external events, to control the real-time output port and the A/D converter, and to generate the serial clock. It consists of two 16-bit timer/counters (TM0 and TM1), two 16-bit timers (TM2 and TM3), one 10-bit interval timer (TM4), one 16-bit up/down counter (UDC), ten 16-bit compare registers, five 16-bit registers that can be used for either capture or compare, and ten timed output latches.

All the timers count various clocks derived from the internal system clock. Timers TM0 and TM1 also count external events on the TI0 and TI1 pins, respectively. All timers are cleared by an external reset. Timers TM0 to TM2 can also be cleared by either an external clear input or a coincidence interrupt from one of its compare registers. Timers TM3 and TM4 are cleared only by a coincidence interrupt from one of its associated compare registers. When any of timers TM0 to TM4 overflow, an overflow bit is set; in the case of timers TM0 and TM3, an overflow interrupt is also generated. Timer TM4 can also be used as the baud-rate generator for the serial interfaces.

Capture events for TM0 can be triggered by external maskable interrupts INTP0 to INTP2; capture events for TM3 can be triggered by INTP3 and INTP4. Compare events associated with timers TM0 to TM4 can be used to generate interrupts, control timed output pins, or both. In addition, three of them (INTCM10, INTCM11, INTCM20) can control the real-time output port and four of them (INTCM00, INTCM01, INTCC00, INTCC01) can control the A/D converter. The timed output latches share pins in ports P2, P3, and P8. Five of them can be toggled or set and reset by compare events, and the remaining five can be toggled. These latches can generate pulse-width modulated outputs.

Figure 13. Real-Time Pulse Unit (Sheet 1 of 3)

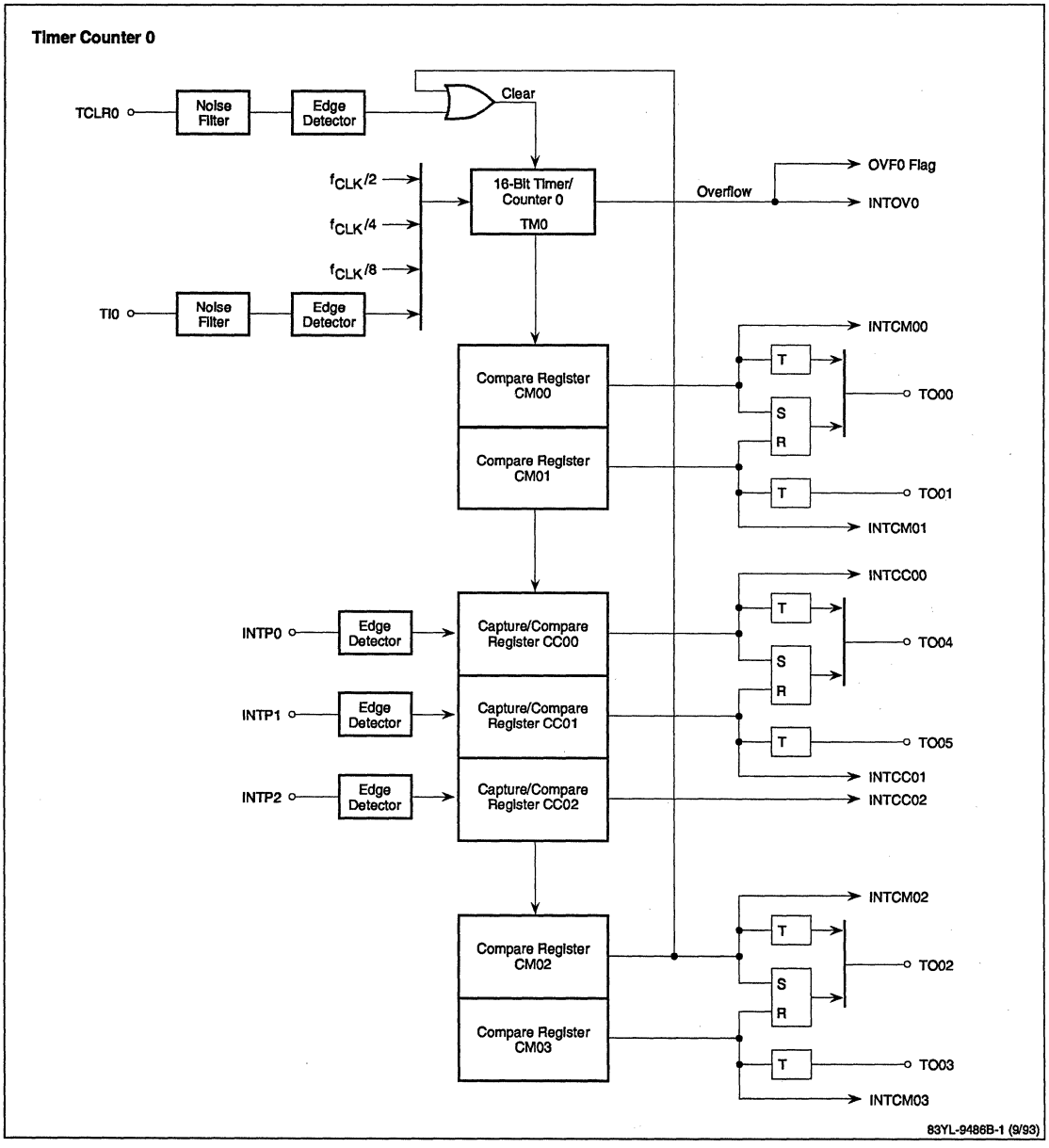
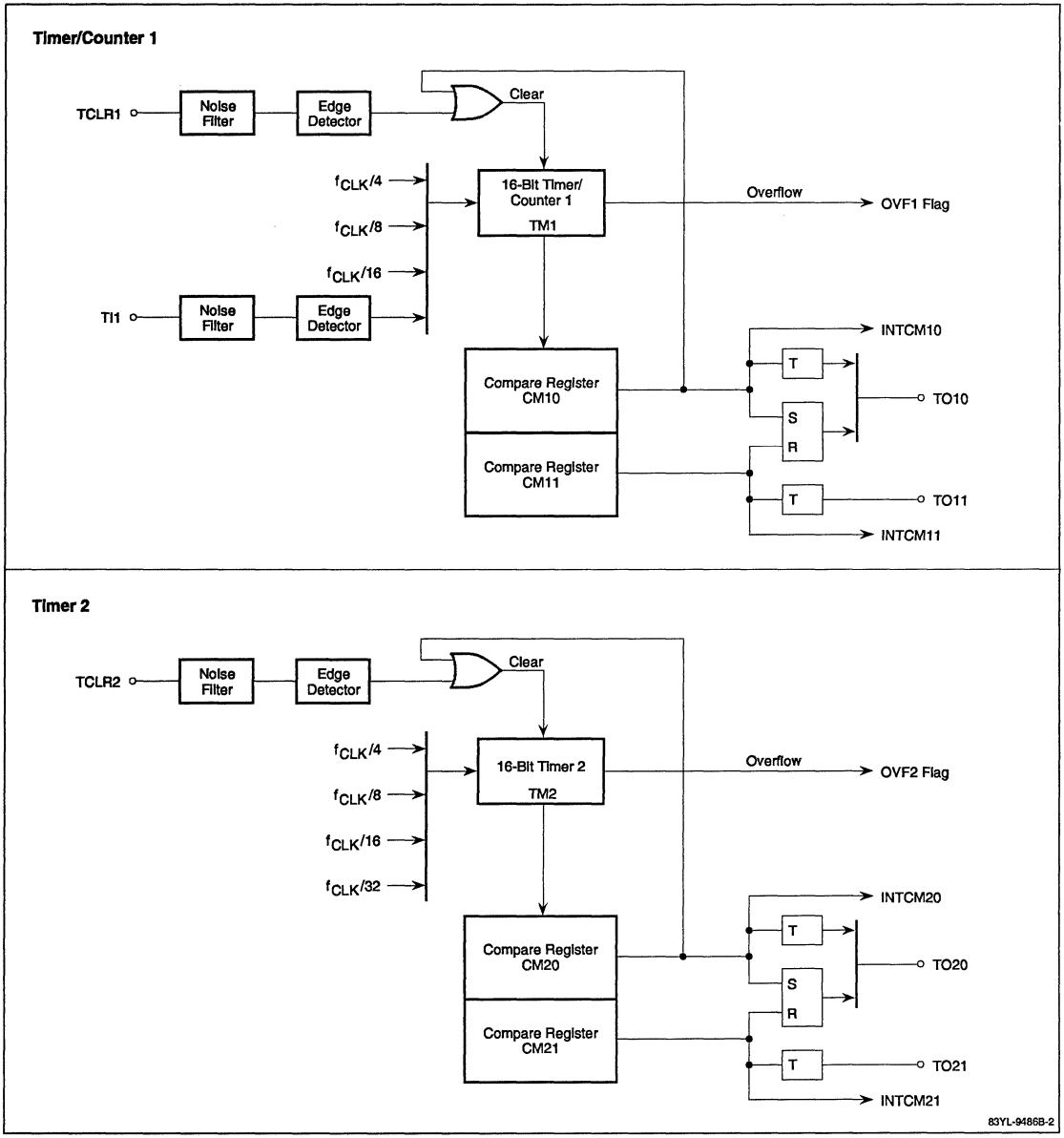
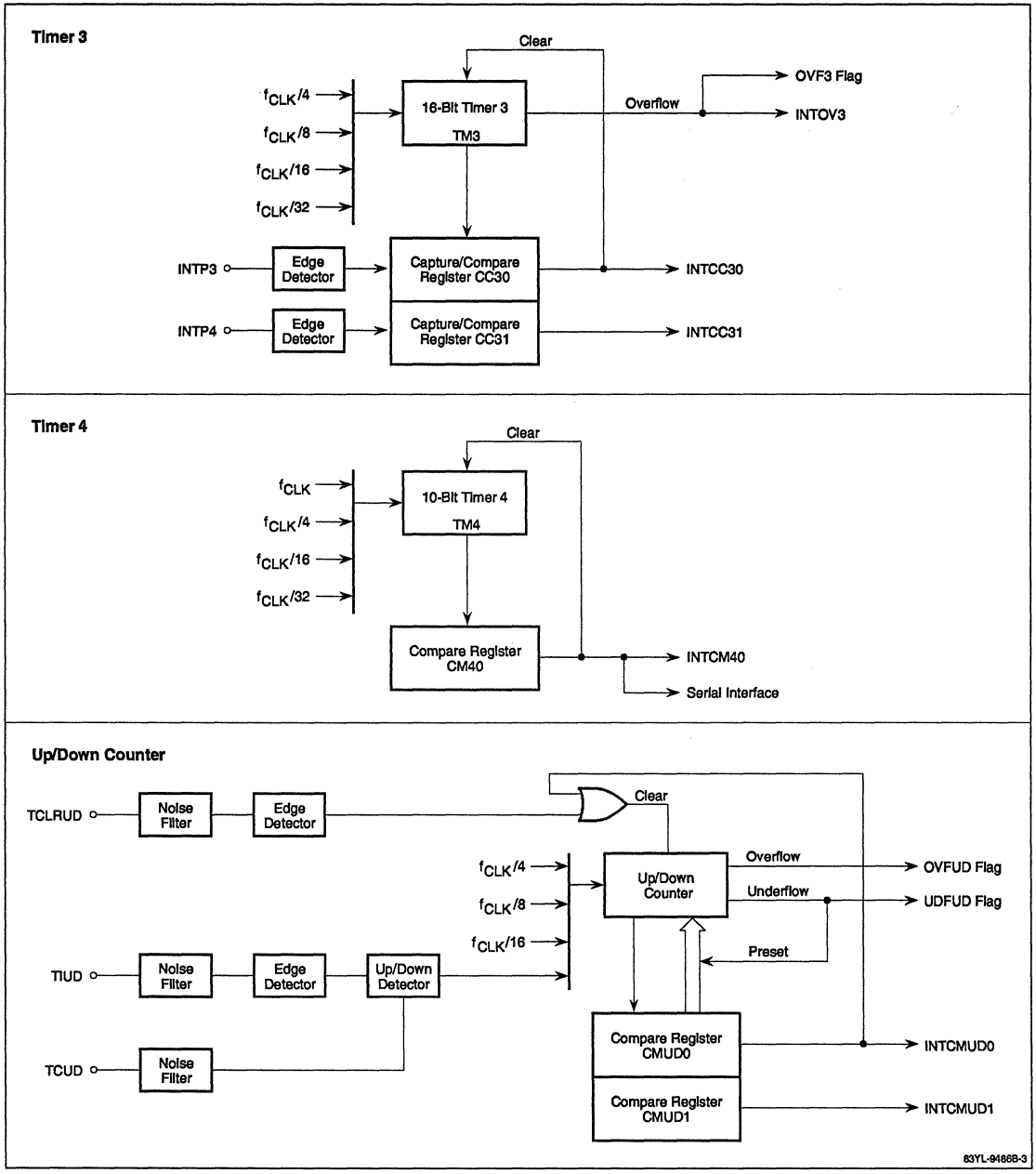


Figure 13. Real-Time Pulse Unit (Sheet 2 of 3)



5d

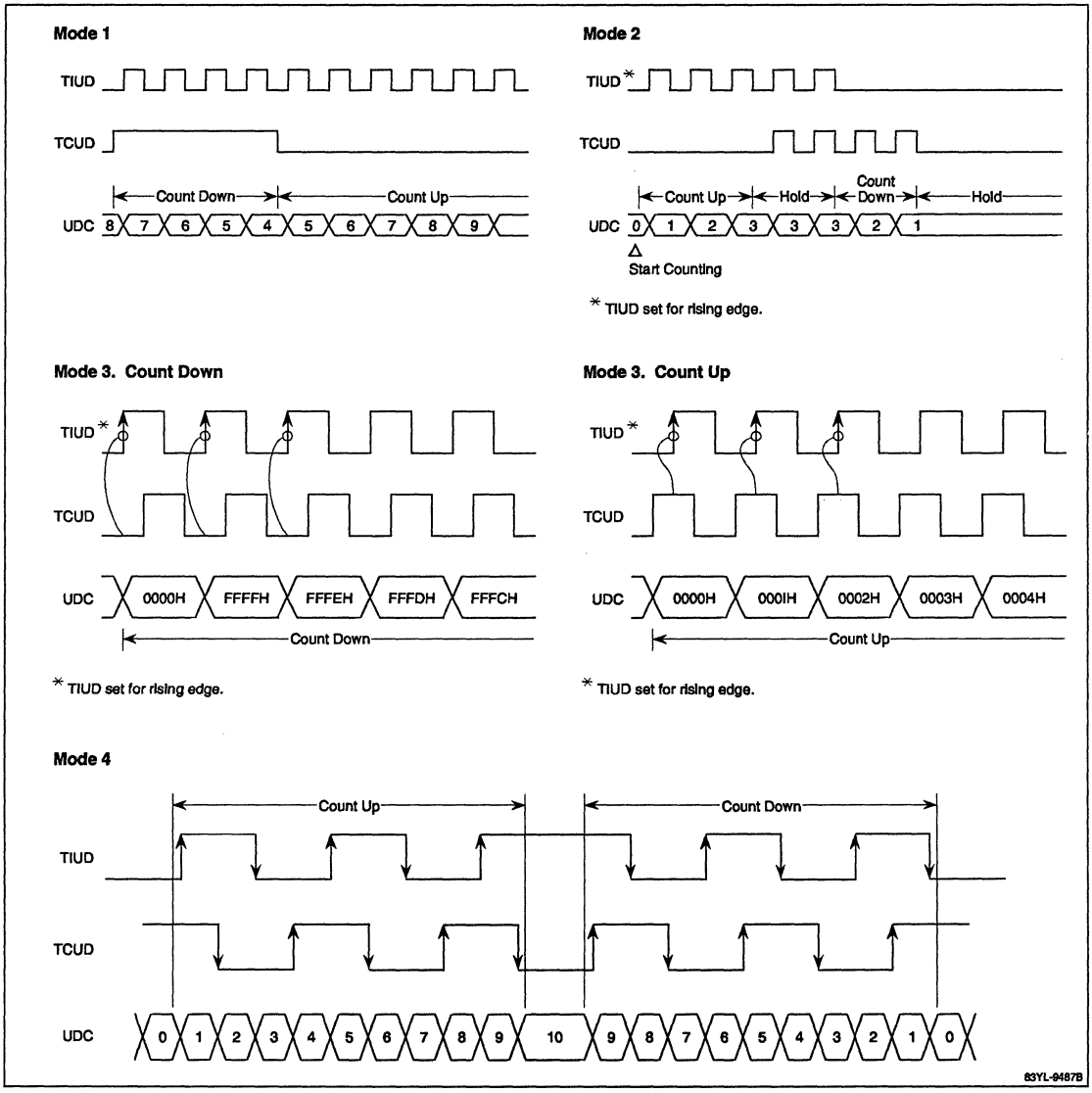
Figure 13. Real-Time Pulse Unit (Sheet 3 of 3)



The 16-bit up/down counter (UDC) can count the internal system clock (divided by 4, 8, or 16) or count external events on the TIUD pin. When the counter overflows, an overflow bit is set. If the counter underflows, an underflow bit is set. The UDC can be cleared by an external clear input (TCLRUD) or by a coincidence signal from its compare register.

When counting external events, the UDC can be programmed to operate in four modes. See figure 14. In mode 1, the UDC counts external events on the TIUD pin. When direction control pin TCUD is high, the UDC counts down. When TCUD is low, the UDC counts up.

Figure 14. Up/Down Counter; Modes 1, 2, 3, 4



5d

In mode 2, when the preset edge (rising, falling, or both) is detected at the TIUD pin, the UDC counts up; when a rising edge is detected at the TCUD pin, the UDC counts down. If TIUD and TCUD are active simultaneously, they are not counted and the value in the UDC is held.

Modes 3 and 4 are designed to count the output of a two-phase shaft encoder on a servomotor. In mode 3, two signals having a 90-degree phase shift are entered into the TIUD and TCUD pins. When the preset edge (rising, falling, or both) is detected on the TIUD pin, the signal level on TCUD is sampled. If the level of TCUD is low, the UDC counts down; if the level is high, the UDC counts up.

When mode 4 is specified, quadrature counting is enabled. The UDC is incremented or decremented at positive and negative transitions of both input signals. Whether it is incremented or decremented is dependent upon the relative phase of the two signals as illustrated in figure 14.

The μPD78356 family has programmable noise detection on the external clock inputs and external clear inputs to the RPU. The noise detection time can be set for each input at 4 or 16 internal system clocks by the noise protection control register (NPC).

Interrupts

The μPD78356 family has 24 maskable hardware interrupt sources: 5 software selectable as external or internal and 19 internal. The four external maskable interrupts share pins with port 2. Any of them, INTP0 and INTP4, can also be used to trigger capture events in the real-time pulse unit. In addition, there are two nonmaskable interrupts, three software interrupts, and reset. The software interrupts, generated by the BRK or BRKCS instruction and the operation code trap, are not maskable. See table 3.

Table 3. Interrupt Sources

Type of Request	Default* Priority	Signal Name	Source	Location	Macro Service Control Word	Vector Address TPF = 0	Vector Address TPF = 1
Software			Operation code trap	CPU		003CH	003CH
			BRK instruction	CPU		003EH	003EH
			BRKCS instruction (Initiates context switch)	CPU			
Nonmaskable		NMI	NMI input pin	External		0002H	8002H
		INTWDT	Watchdog timer overflow	Internal		0004H	8004H
Maskable	0	INTOV0	Timer 0 overflow	Internal	FE06H	0006H	8006H
	1	INTOV3	Timer 3 overflow	Internal	FE06H	0006H	8006H
	2	INTP0 INTCC00	INTP0 pin CC00 coincidence	External Internal	FE0AH	000AH	800AH
	3	INTP1 INTCC01	INTP1 pin CC01 coincidence	External Internal	FE0CH	000CH	800CH
	4	INTP2 INTCC02	INTP2 pin CC02 coincidence	External Internal	FE0EH	000EH	800EH
	5	INTP3 INTCC30	INTP3 pin CC30 coincidence	External Internal	FE10H	0010H	8010H
	6	INTP4 INTCC31	INTP4 pin CC31 coincidence	External Internal	FE12H	0012H	8012H
	7	INTCM00	CM00 coincidence	Internal	FE14H	0014H	8014H
	8	INTCM01	CM01 coincidence	Internal	FE16H	0016H	8016H
	9	INTCM02	CM02 coincidence	Internal	FE18H	0018H	8018H
	10	INTCM03	CM03 coincidence	Internal	FE1AH	001AH	801AH
	11	INTCM10	CM10 coincidence	Internal	FE1CH	001CH	801CH
	12	INTCM11	CM11 coincidence	Internal	FE1EH	001EH	801EH
	13	INTCM20	CM20 coincidence	Internal	FE20H	0020H	8020H
	14	INTCM21	CM21 coincidence	Internal	FE22H	0022H	8022H
	15	INTCM40	CM40 coincidence	Internal	FE24H	0024H	8024H
	16	INTCMUD0	CMUD0 coincidence	Internal	FE26H	0026H	8026H
	17	INTCMUD1	CMUD1 coincidence	Internal	FE28H	0028H	8028H
	18	INTSER	Asynchronous serial interface reception error	Internal	FE2AH	002AH	802AH
	19	INTSR	End of asynchronous serial interface reception	Internal	FE2CH	002CH	802CH
	20	INTST	End of asynchronous serial interface transmission	Internal	FE2EH	002EH	802EH
	21	INTCSI0	End of clocked serial interface CSI0 transmission/reception	Internal	FE30H	0030H	8030H
	22	INTCSI1	End of clocked serial interface CSI1 transmission/reception	Internal	FE32H	0032H	8032H
	23	INTAD	End of A/D conversion	Internal	FE34H	0034H	8034H
Reset		RESET	RESET pin	External		0000H	0000H

* 0 is the highest priority.

5d

Interrupt Servicing

The μPD78356 family provides four levels of programmable hardware priority control and three different methods of handling maskable interrupt requests: standard vectoring, context switching, and macro service. The programmer can choose the priority and mode of servicing each maskable interrupt by using the interrupt control registers.

Interrupt Control Registers

The μPD78356 family has 24 interrupt control registers. Each maskable interrupt request has its own control register, which includes bits to specify interrupt request, interrupt mask, macro service enable, context switch enable, and priority. Priorities range from 0 (highest) to 3. See figure 15.

There are also three mask flag register, MK0L, MK0H, and MK1L, with a bit for each maskable interrupt. Since each interrupt has two mask bits, the masking of the interrupt is the "or" function of those two bits.

Interrupt mode control register IMC can enable or disable nesting of interrupts set to the lowest priority level (level 3). Inservice priority register ISPR is used by the hardware to hold the priority level of the interrupt request currently being serviced. It is manipulated by hardware only, but it can be read by software.

Finally, the IE bit of the program status word also is used to control the interrupts. If the IE bit is 0, all maskable interrupts, but not macro service, are disabled. The IE bit can be set or cleared by the EI or DI instruction, respectively, or by direct writing to the PSW. The IE bit is cleared each time an interrupt is accepted.

Figure 15. Interrupt Control Register (xxICx)

7	6	5	4
xxIFxx	xxMKxx	xxISMxx	xxCSExx
3	2	1	0
0	0	xxPRx1	xxPRx0

xxIFxx		Interrupt Request Flag
0		No interrupt request
1		Interrupt request received
xxMKxx		**Interrrupt Mask Flag**
0		Interrupt request enabled
1		Interrupt will be pending
xxISMxx		**Macro Service Enable**
0		Software interrrupt
1		Macro service
xxCSExx		**Context Switch Enable**
0		Vector interrupt
1		Context switch
xxPRx1	**xxPRx0**	**Priority Specification**
0	0	Priority 0 (highest
0	1	Priority 1
1	0	Priority 2
1	1	Priority 3

Interrupt Priority

The two nonmaskable interrupts, NMI and INTWDT, have priority over all others. Their priority relative to each other is under program control.

Four hardware-controlled priority levels are available for the maskable interrupts. Any one of the four levels can be assigned by software to each of the maskable interrupt lines. Interrupt requests of a priority higher than the processor's current priority level are accepted; requests of the same or lower priority are held pending until the processor's priority state is lowered by a return instruction from the current service routine.

By setting the PRSL bit of the IMC register to zero, it is possible to specify in software that level 3 interrupts (the lowest level) can be accepted when the processor is operating at level 3. This nesting within a level applies to level 3 only.

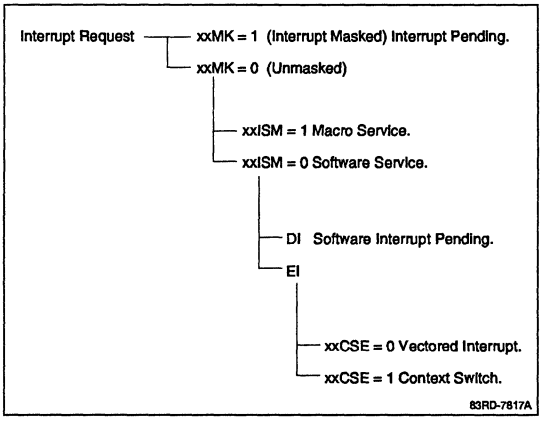
Interrupt requests programmed to be handled by macro service have priority over all software interrupt service regardless of the assigned priority level, and

macro service requests are accepted even when the interrupt enable bit in the PSW is set to the disable state. See figure 16.

The default priorities listed in table 3 are fixed by hardware; they are effective only when it is necessary to choose between two interrupt requests of the same software-assigned priority. For example, the default priorities would be used after the completion of a high-priority routine if two interrupts of the same lower priority were pending.

Software interrupts, the BRK and BRKCS instructions, and the operation code trap are executed regardless of the processor's priority level and the state of the IE bit. They do not alter the processor's priority level.

Figure 16. Interrupt Service Sequence



Vectored Interrupt

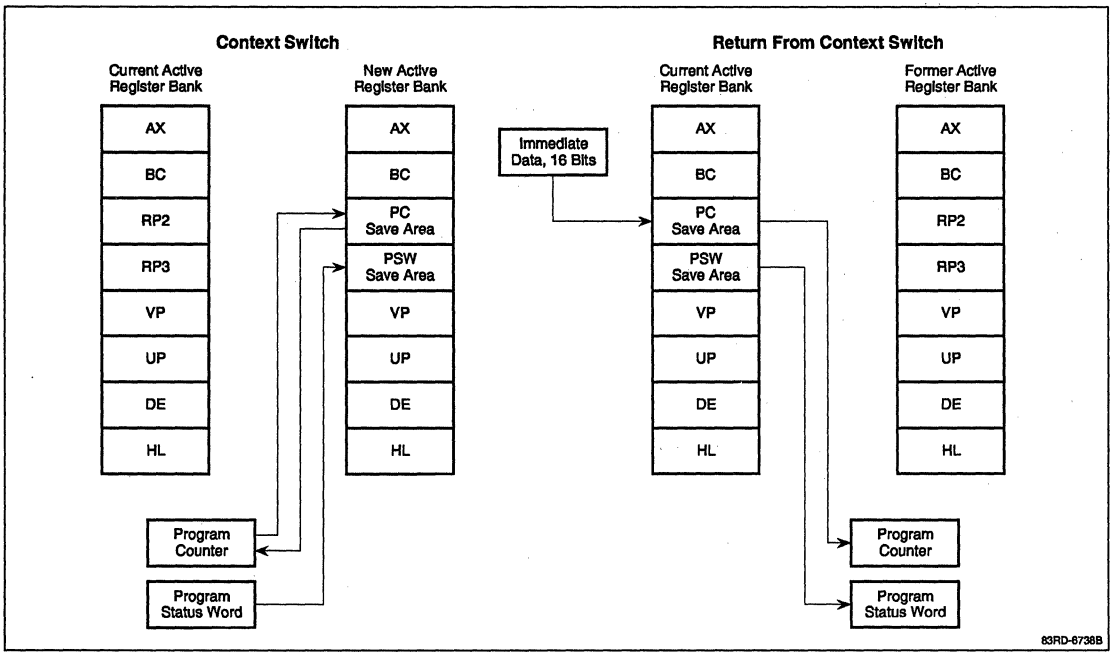
When vectored interrupt is specified for a given interrupt request, (1) the program status word and the program counter are saved on the stack, (2) the processor's priority is raised to that specified for the interrupt, (3) the IE bit in the PSW is set to zero, and (4) the routine whose address is in the interrupt vector table is entered. At completion of the service routine, the RETI instruction (or RETB instruction for software interrupts) reverses the process, and the μPD78356 family device resumes the interrupted routine. The corresponding interrupt request flag is cleared before executing the interrupt service routine.

Context Switch

When context switching (figure 17) is specified for a given interrupt, the active register bank is changed to the register bank specified by the three low-order bits of the word in the interrupt vector table. The program counter is loaded from RP2 of the new register bank, the old program counter and program status word are saved in RP2 and RP3 of the new register bank, and the IE bit in the PSW is set to zero.

At completion of the service routine, the RETCS instruction for routines entered from hardware requests or the RETCSB instruction for routines entered from the BRKCS instruction reverses the process. The old program counter and program status word are restored from RP2 and RP3 of the new register bank. The entry address of the service routine, which must be specified in the 16-bit immediate operand of these return instructions, is stored again in RP2.

Figure 17. Context Switching and Return



Macro Service

When macro service is specified for a given interrupt, the macro service hardware temporarily stops the executing program and transfers data between the special function register area and the memory space. Control is then returned to the executing program, providing a completely transparent method of interrupt service. Macro service significantly improves response time and makes it unnecessary to save any registers.

For each request on the interrupt line, one operation is performed, and an 8-bit counter is decremented. When the counter reaches 0, a software service routine is entered according to its specified priority. Either vectored interrupt or context switch can be specified for entry to this routine, which is known as the macro service completion routine.

Macro service is provided for all of the maskable interrupt requests, and each has a specific macro service control word stored in on-chip main RAM. The function to be performed is specified in the control word.

The μPD78356 family provides five different macro service functions.

Function	Description
EVTCNT	Event counter. Counts up to 256 events by incrementing or decrementing the macro service counter. When the counter reaches 00H, the software service routine is entered.
BLKTRS	Block transfer. Transfers a byte or word of data in either direction between a specified special function register and a buffer in main RAM (FExx).
BLKTRS-P	Block transfer with memory pointer. Transfers a byte or word of data in either direction between a specified special function register and a buffer anywhere in the 64K-byte address space.
DTADIF	Data difference. Stores the difference between the current value of a specified 16-bit special function register and its previous value in a word buffer in main RAM (FExx).

DTADIF-P Data difference with memory pointer. Stores the difference between the current value of a specified 16-bit special function register and its previous value in a word buffer anywhere in the 64K-byte address space.

Standby Modes

The standby modes, HALT and STOP, reduce power consumption when CPU action is not required. In HALT mode, the CPU is stopped but the system clock continues to run. The HALT mode is released by any unmasked interrupt, any nonmaskable interrupt, or an external reset pulse. In STOP mode, both the CPU and the system clock are stopped, further minimizing power consumption. The STOP mode is released by either an external reset pulse or an external NMI.

The HALT and STOP modes are entered by programming standby control register STBC. This register is a protected location and can be written to only by a special instruction. If the third and fourth bytes of the instruction are not complements of each other, the data is not written and an operation code trap interrupt occurs.

Watchdog Timer

The watchdog timer protects against inadvertent program loops. A nonmaskable interrupt occurs if the timer is not reset by the program before it overflows. At the same time, watchdog timer output pin $\overline{\text{WDT0}}$ goes active low for a period of 32 system clocks. The $\overline{\text{WDT0}}$ pin can be connected to the $\overline{\text{RESET}}$ pin or used to control external circuitry. Three program-selectable intervals are available: 8.2, 32.8, and 131.1 ms at 32 MHz.

Once started, the timer can be stopped only by an external reset. Watchdog timer mode register WDM is used to select the time interval, to set the relative priority of the watchdog timer interrupt and NMI, and to clear the timer. This register is a protected location and can be written to only by a special instruction. If the third and fourth bytes of the instruction are not complements of each other, the data is not written and an operation code trap interrupt occurs.

External Reset

The μPD78356 family is reset by taking the $\overline{\text{RESET}}$ pin low. The reset circuit contains a noise filter to protect against spurious system resets caused by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low until the power supply reaches its operating voltage and the oscillator has stabilized. During reset, the program counter is loaded with the address contained in the reset vector table (addresses 0000H, 0001H); program execution starts at that address upon the $\overline{\text{RESET}}$ pin going high. While $\overline{\text{RESET}}$ is low, all external lines except $\overline{\text{WDTO}}$, CLKOUT, V_{SS}, V_{DD}, AV_{SS}, AV_{DD}, AV_{REF1}, AV_{REF2}, AV_{REF3}, X1, and X2 are in the high-impedance state.

5d

ELECTRICAL SPECIFICATIONS (Preliminary)

Absolute Maximum Ratings

T_A = 25°C

Supply voltage, V_{DD}	−0.5 to +7.0 V
Supply voltage, V_{PP}	−0.5 to +13.5 V
Input voltage, V_I	
Except $P2_0$/NMI (A9) of 78P356	−0.5 to V_{DD} + 0.5 V
$P2_0$/NMI (A9) of 78P356	−0.5 to +13.5 V
Output voltage, V_O	−0.5 to V_{DD} + 0.5 V
Output current, low; I_{OL}	
Each output pin	4.0 mA
Total	140 mA
Output current, high; I_{OH}	
Each output pin	−1.0 mA
Total	30 mA
Operating temperature, T_{OPT}	−10 to +70°C
Storage temperature, T_{STG}	−65 to +150°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC characteristics.

Operating Conditions

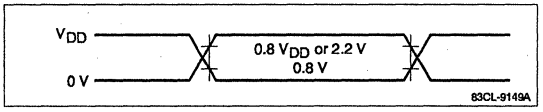
Oscillator Frequency, f_{XX}	T_A	V_{DD}
8 to 32 MHz	−10 to +70°C	+5.0 V ±10%

Capacitance

T_A = 25°C; V_{DD} = V_{SS} = 0 V

Parameter	Symbol	Max	Unit	Conditions
Input pin capacitance	C_I	TBD	pF	f = 1 MHz; unmeasured pins returned to 0 V
Output pin capacitance	C_O	TBD	pF	
I/O pin capacitance	C_{IO}	TBD	pF	

AC Timing Test Points



DC Characteristics

T_A = −10 to +70°C; V_{DD} = +5.0 V ±10%; V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input voltage, low	V_{IL}	0		0.8	V	
Input voltage, high	V_{IH1}	2.2			V	(Note 1)
	V_{IH2}	0.8 V_{DD}			V	(Note 2)
Output voltage, low	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output voltage, high	V_{OH}	V_{DD} − 1.0			V	I_{OH} = −400 μA
Input leakage current	I_{LI}			±10	μA	V_I = 0 to V_{DD}
Output leakage current	I_{LO}			±10	μA	V_O = 0 to V_{DD}
V_{DD} supply current	I_{DD1}		75	107	mA	Operating mode; 78355/356
			86	125	mA	Operating mode; 78P356
	I_{DD2}		40	60	mA	HALT mode; 78355/356
			40	60	mA	HALT mode; 78P356
Data retention voltage	V_{DDDR}	2.5			V	STOP mode
Data retention current	I_{DDDR}		2	10	μA	STOP mode; V_{DDDR} = 2.5 V
			10	50	μA	STOP mode; V_{DDDR} = 5.0 V ±10%

Notes:

(1) All except pins in Note 2.

(2) Pins $\overline{\text{RESET}}$, X1, $P0_0$ - $P0_3$, P2, $P3_2$ - $P3_6$, $P8_0$ - $P8_1$, $P8_5$, $P10_1$, $P10_2$, $P10_4$ - $P10_7$.

AC Characteristics

T_A = −10 to +70°C; V_{DD} = +5.0 V ±10%; V_{SS} = 0 V; f_{XX} = 32 MHz

Parameter	Symbol	Calculation Formula	Min	Max	Unit	Conditions
External Memory Read/Write Operation						
System clock cycle time (Note 1)	t_{CYK}	—	62.5	250	ns	C_L = 50 pF
Address setup time to ASTB ↓	t_{SAST}	(0.5 + a)T − 24	7		ns	C_L = 100 pF (Note 2)
Address hold after ASTB ↓	t_{HSTA}	0.5T − 16	15		ns	C_L = 100 pF
$\overline{\text{RD}}$ ↓ to address floating	t_{FRA}	—		0	ns	C_L = 100 pF
Address to data input valid	t_{DAID}	(2.5 + a + n)T − 56		100	ns	C_L = 100 pF (Note 2, 3)
$\overline{\text{RD}}$ ↓ to data input valid	t_{DRID}	(1.5 + n)T − 44		49	ns	C_L = 100 pF (Note 3)
ASTB ↓ to $\overline{\text{RD}}$ ↓ delay time	t_{DSTR}	0.5T − 16	15		ns	C_L = 100 pF
Data hold time from $\overline{\text{RD}}$ ↑	t_{HRID}	—	0		ns	C_L = 100 pF
$\overline{\text{RD}}$ ↑ to next address active	t_{DRA}	0.5T − 14	17		ns	C_L = 100 pF
$\overline{\text{RD}}$ width low	t_{WRL}	(1.5 + n)T − 30	63		ns	C_L = 100 pF (Note 3)
ASTB width high	t_{WSTH}	(0.5 + a)T − 17	14		ns	C_L = 100 pF (Note 2)
$\overline{\text{LWR}}$ or $\overline{\text{HWR}}$ to data output	t_{DWOD}	0.5T − 10		21	ns	C_L = 100 pF
ASTB ↓ to $\overline{\text{LWR}}$ ↓ or $\overline{\text{HWR}}$ ↓ delay	t_{DSTW}	0.5T − 16	15		ns	C_L = 100 pF
Data setup time to $\overline{\text{LWR}}$ ↑ or $\overline{\text{HWR}}$ ↑	t_{SODW}	(1 + n)T − 5	57		ns	C_L = 100 pF (Note 3)
Data hold time after $\overline{\text{LWR}}$ ↑ or $\overline{\text{HWR}}$ ↑	t_{HWOD}	—	8		ns	C_L = 100 pF
$\overline{\text{LWR}}$ or $\overline{\text{HWR}}$ width, low	t_{WWL}	(1.5 + n)T − 30	63		ns	C_L = 100 pF (Note 2, 3)
$\overline{\text{WAIT}}$ setup time from address	t_{SAWT}	(a + n)T − 15		110	ns	C_L = 100 pF (Notes 2, 4)
$\overline{\text{WAIT}}$ hold time from address	t_{HAWT}	(0.5 + a + n)T	156		ns	C_L = 100 pF (Notes 2, 4)

AC Characteristics (cont)

Parameter	Symbol	Calculation Formula	Min	Max	Unit	Conditions
$\overline{WAIT}$ setup time from $\overline{RD}$ ↓ or $\overline{WR}$ ↓	t_{SRWRY}	(n – 1)T – 25		37	ns	C_L = 100 pF (Note 4)
$\overline{WAIT}$ hold time from $\overline{RD}$ ↓ or $\overline{WR}$ ↓	t_{HRWRY}	(n – 0.5)T – 14	79		ns	C_L = 100 pF (Note 4)
ASTB ↑ delay time from $\overline{WR}$ ↑	t_{DWST}	1.5T – 15	TBD		ns	C_L = 100 pF
Serial Port Operation						
$\overline{SCK}$ cycle time	t_{CYSK}	—	8T		ns	$\overline{SCK}$ output
			500		ns	$\overline{SCK}$ input
$\overline{SCK}$ width, low	t_{WSKL}	—	4T – 40		ns	$\overline{SCK}$ output
			210		ns	$\overline{SCK}$ input
$\overline{SCK}$ width, high	t_{WSKH}	—	4T – 40		ns	$\overline{SCK}$ output
			210		ns	$\overline{SCK}$ input
SI setup time to $\overline{SCK}$ ↑	t_{SRXSK}	—	80		ns	
SI hold time after $\overline{SCK}$	t_{HSKRX}	—	80		ns	
$\overline{SCK}$ ↓ to SO delay time	t_{DSKTX}	—		110	ns	

Definitions:

(1) T = t_{CYK} (ns)

(2) a = address wait state: a = 0 or 1

(3) n = number of wait states specified by the external wait pin $\overline{WAIT}$ and the PWC register

Notes:

(1) t_{CYK} equals twice the period of the crystal or external clock input.

(2) No address wait state

(3) No wait states

(4) One external wait state and one internal wait state.

A/D Converter Characteristics

T_A = –10 to +70°C; AV_{DD} = V_{DD} = +5 V ±10% or +3 V ±10%; V_{SS} = AV_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Resolution				10	Bit	
Total error*				1.2	%	
Quantization error				±1/2	LSB	
Conversion time	t_{CONV}	32			t_{CYK}	AD trigger mode
		37			t_{CYK}	Timer/external trigger mode
Sampling time	t_{SAMP}	7.5			t_{CYK}	
Analog input voltage	A_{IAN}	–0.3		AV_{REF1} + 0.3	V	
Analog input impedance	R_{AN}		TBD		MΩ	
Reference voltage	AV_{REF1}	TBD		V_{DD}	V	
AV_{REF1} current	AI_{REF1}		3.0	9	mA	f_{XX} = 32 MHz
			TBD	TBD	mA	ADM0.7(CS) = 0
AV_{DD} supply current	AI_{DD}		3.3	13	mA	Operational mode

* Does not include quantization error. Percentage of full-scale value is shown.

D/A Converter Characteristics

T_A = −10 to +70°C; AV_{REF2} = V_{DD} = +5 V ±10%; AV_{REF3} = V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Resolution				8	Bit	
Overall error				TBD	%	
Settling time				2	μs	Load condition 2 MΩ, 30 pF
Output resistance	R_O		20		kΩ	When DACS0 and DACS1 are set to 7FH
Analog reference voltage	AV_{REF2}	0.75 V_{DD}		V_{DD}	V	
	AV_{REF3}	V_{SS}		0.2 V_{DD}	V	
Reference supply input current	AI_{REF2}	0		5	mA	
	AI_{REF3}	−5		0	mA	

Up/Down Counter Operation

T_A = −10 to +70°C; V_{DD} = +5 V ±10% V_{SS} = 0 V; T = t_{CYK} (ns)

Parameter	Symbol	Min	Max	Unit	Conditions
TIUD high-, low-level width	t_{WTIUH}, t_{WTIUL}	aT		ns	Except mode 4
		bT		ns	Mode 4
TCUD high-, low-level width	t_{WTCUH}, t_{WTCUL}	aT		ns	Except mode 4
		bT		ns	Mode 4
TCLRUD high-, low-level width	t_{WCLUH}, t_{WCLUL}	aT		ns	
TCUD setup time to TIUD	t_{STCU}	0		ns	Mode 3, rise
TCUD hold time from TIUD ↑	t_{HTCU}	2T		ns	Mode 3, rise
TIUD setup time to TCUD	t_{S4TIU}	4T		ns	Mode 4
TIUD hold time from TCUD	t_{H4TIU}	4T		ns	Mode 4
TIUD, TCUD cycle time	t_{CYC4}		1	MHz	

Note: a, b are defined by NPC register.
NPC.bit = 0, then a = 4, b = 8; NPC.bit = 1, then a = b = 16.

5d

Other Operations

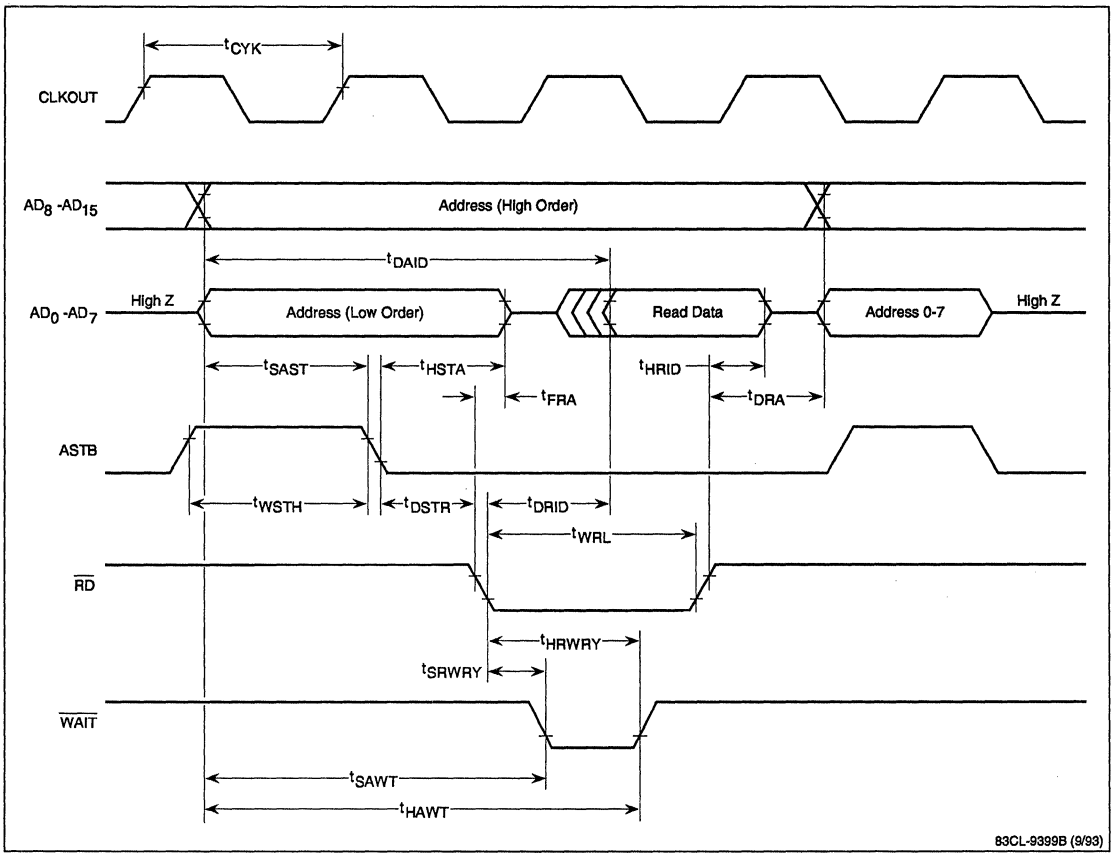
T_A = −10 to +70°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V; T = t_{CYK} (ns)

Parameter	Symbol	Min	Max	Unit	Conditions
NMI high-, low-level width	t_{WNIH}, t_{WNIL}	2		μs	
INTP0 to INTP4 high-, low-level width	t_{WInH}, t_{WInL}	4T		ns	
$\overline{RESET}$ high-, low-level width	t_{WRSH}, t_{WRSL}	2		μs	
TIn high-, low-level width	t_{WTInH}, t_{WTInL}	aT		ns	
TCLRn high-, low-level width	t_{WCLnH}, t_{WCLnL}	aT		ns	
ADTRGn high-, low-level width	t_{WADnH}, t_{WADnL}	4T		ns	
ADC external trigger inputs ADTRGn to ADTRGm (valid edge to valid edge)	$t_{ADCININ}$	37T		ns	

Note: a is defined by NPC register (a = 4 or 16).

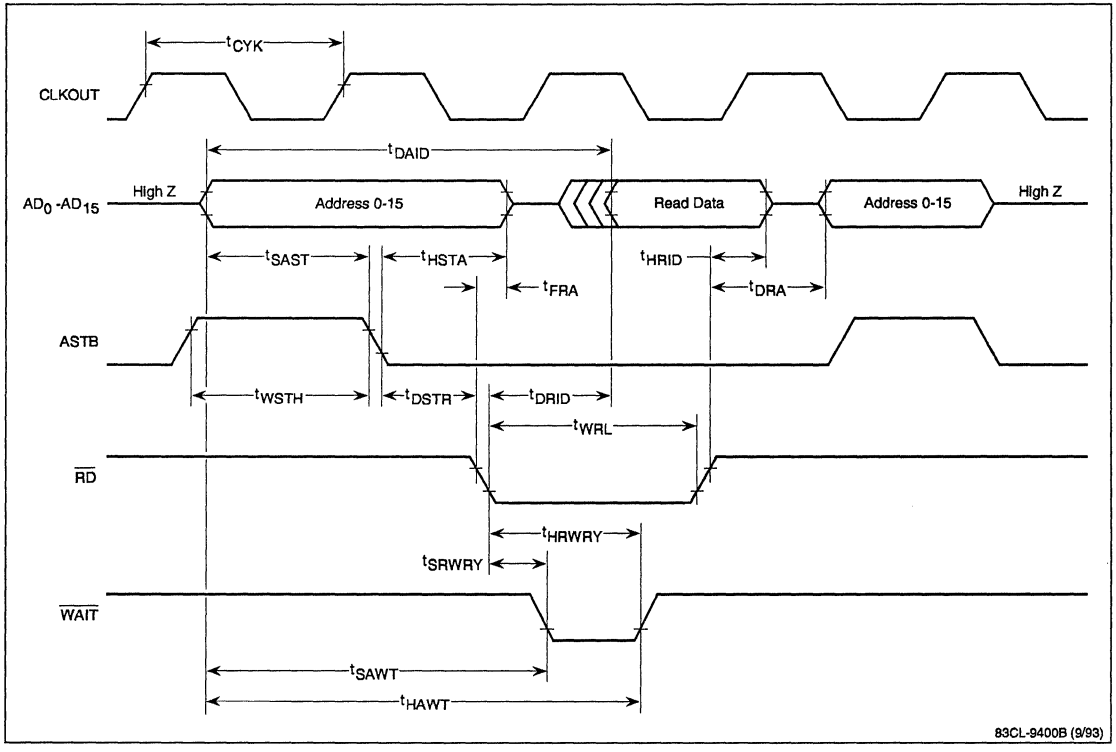
Timing Waveforms

Read Operation (8-Bit Bus)



Timing Waveforms (cont)

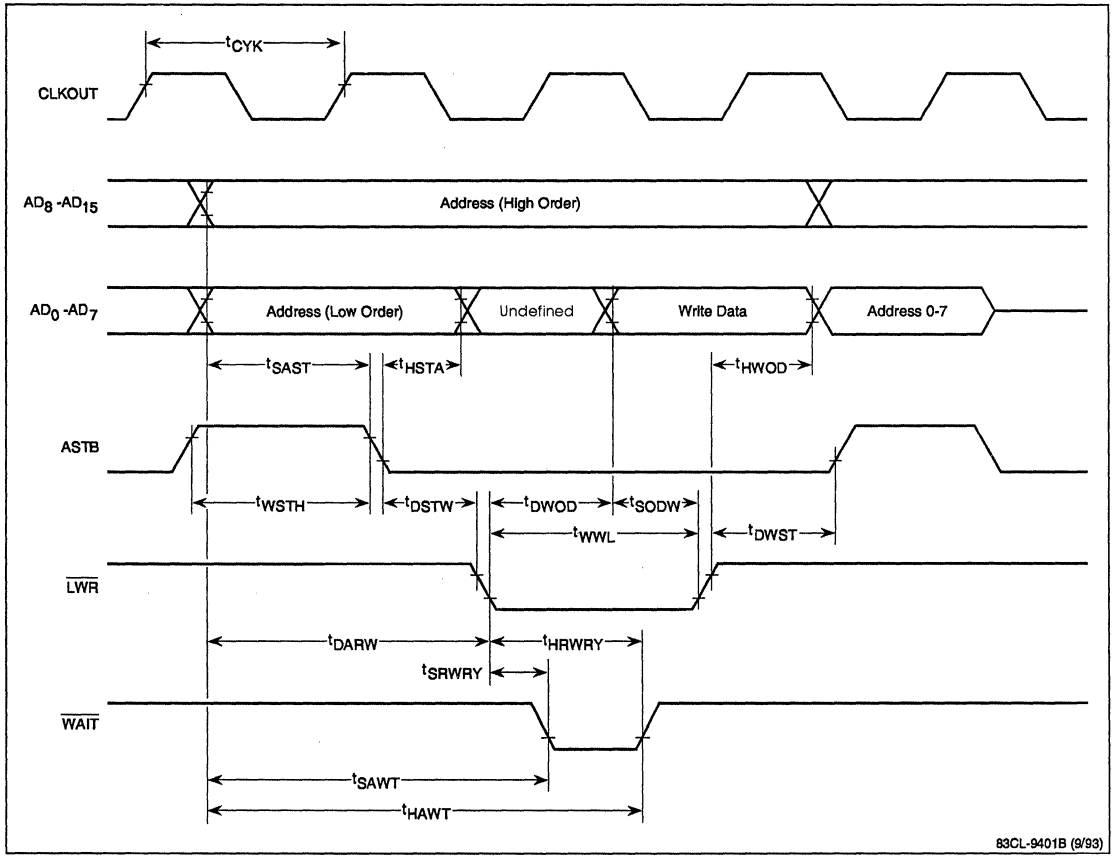
Read Operation (16-Bit Bus)



5d

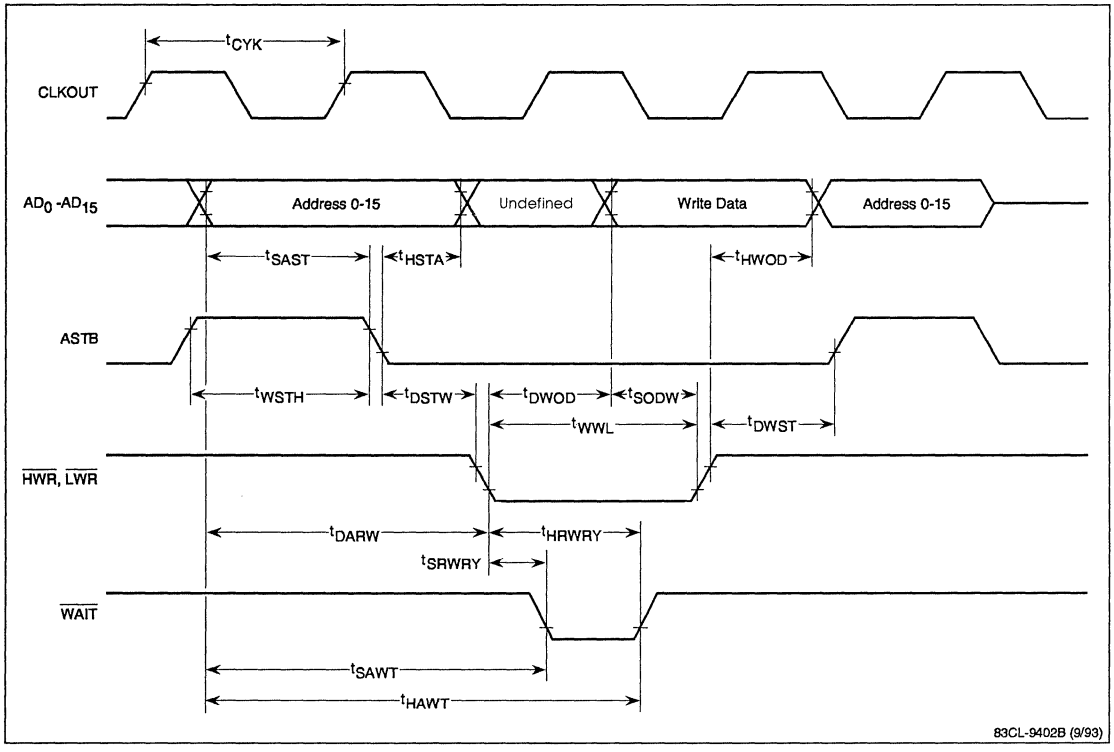
Timing Waveforms (cont)

Write Operation (8-Bit Bus)



Timing Waveforms (cont)

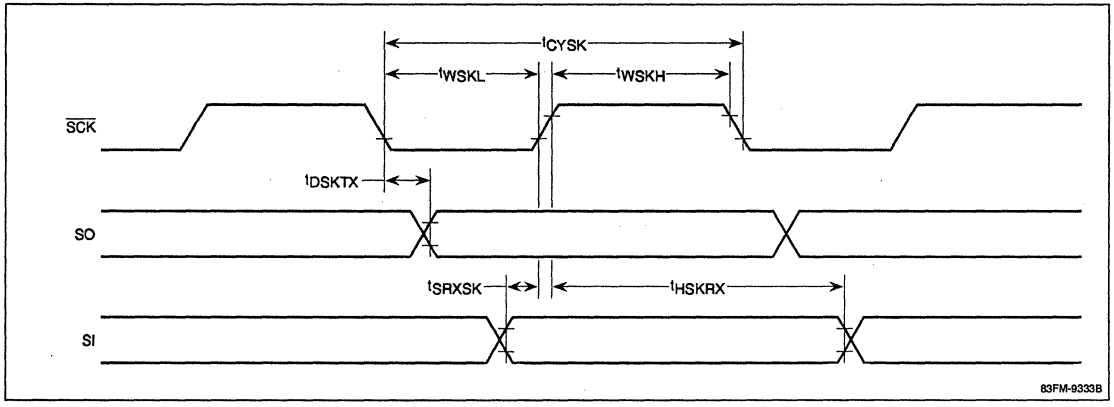
Write Operation (16-Bit Bus)



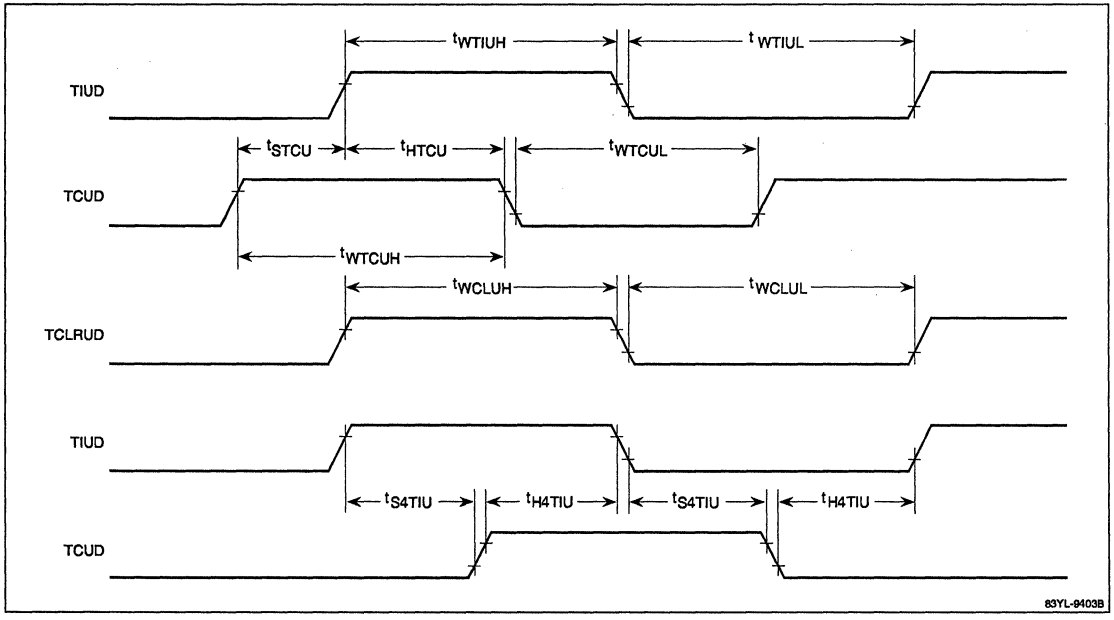
5d

Timing Waveforms (cont)

Serial Port Operation, Clock Synchronous Mode

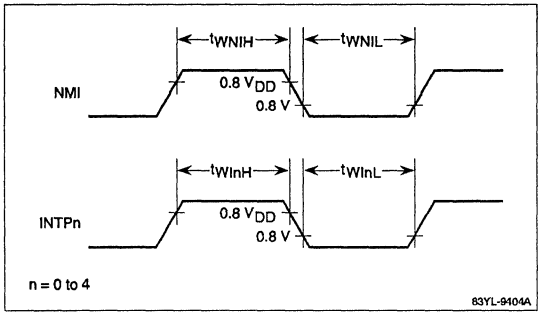


Up/Down Counter Input

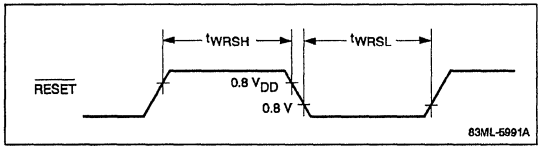


Timing Waveforms (cont)

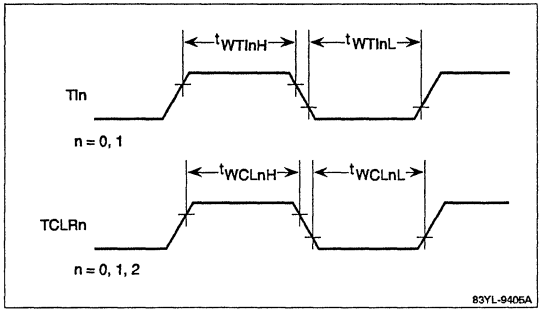
Interrupt Input



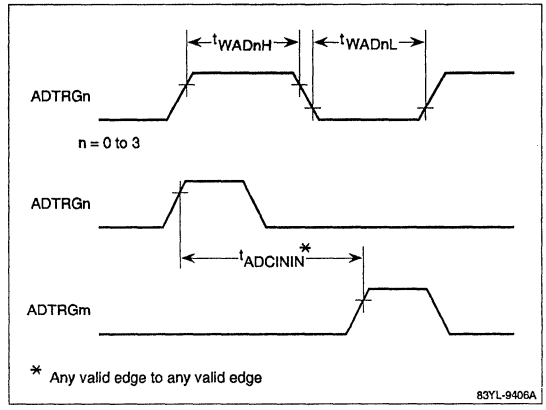
Reset Input



Timer Input



A/D Converter Trigger Input



5d

PROM PROGRAMMING

The PROM in the μPD78P356 is one-time programmable (OTP) or ultraviolet erasable (UV EPROM). The 49,152 x 8-bit PROM has the programming characteristics of an NEC μPD27C1001A, including both page and byte programming modes. Table 4 shows the functions of the μPD78P356 pins in normal operating mode and PROM programming mode.

Table 4. Pin Functions During PROM Programming

Function	Normal Operating Mode	Programming Mode
Address input	$P0_0$ - $P0_7$, $P5_0$, $P2_0$, $P5_1$ - $P5_7$	A_0 - A_{16}
Data input	$P4_0$ - $P4_7$	D_0 - D_7
Program pulse	$P1_2$	$\overline{PGM}$
Chip enable	$P1_0$	$\overline{CE}$
Output enable	$P1_1$	$\overline{OE}$
Program voltage	MODE0/V_{PP}	MODE0/V_{PP}
Mode voltage	MODE1, $P2_1$, $\overline{RESET}$	MODE1, $P2_1$, $\overline{RESET}$

The μPD78P356 also includes a PROM error correction function capable of correcting one 1-bit error per four bytes of code. Each device contains 49,152 bytes of PROM for program storage (0000H to BFFFH) and 12,288 bytes of PROM for error correction code (C000H to EFFFH). A four-byte ECC Control Word (ECW) is located at addresses F000H to F003H and controls the ECC circuitry. For additional protection, four bytes of ECC data for the ECW are located at addresses F004H to F007H. The error correction code and information to be stored in the ECW are automatically generated and added to your HEX file by the ECCGEN program supplied with the RA78K3 Relocatable Assembler Package.

PROM Programming Mode

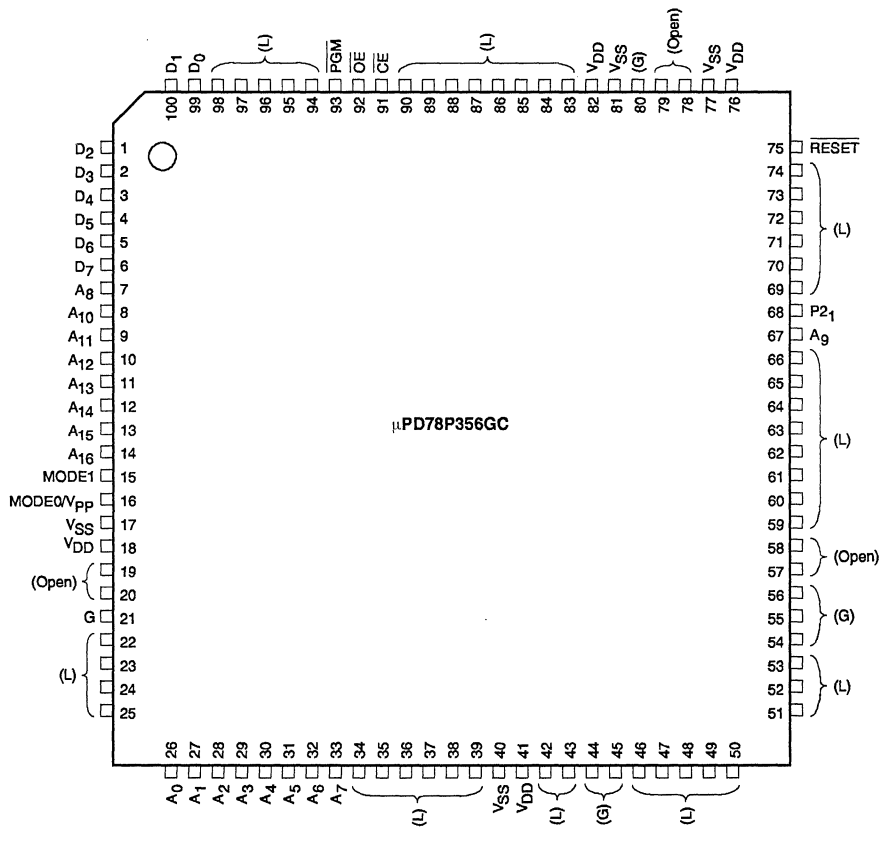
When MODE1, $P2_1$, and $\overline{RESET}$ pins are set low, the μPD78P356 enters the PROM programming mode. Operation in this mode is determined by the setting of $\overline{CE}$, $\overline{OE}$, $\overline{PGM}$, MODE0/V_{PP}, and V_{DD} pins as indicated in table 5.

Table 5. Operation Modes for Programming

Mode	MODE1	$P2_1$	$\overline{RESET}$	$\overline{CE}$	$\overline{OE}$	$\overline{PGM}$	MODE0/V_{PP}	V_{DD}	D_0 - D_7
Page data latch	L	L	L	H	L	H	+12.5 V	+6.5 V	Data input
Page program	L	L	L	H	H	L	+12.5 V	+6.5 V	High impedance
Byte program	L	L	L	L	H	L	+12.5 V	+6.5 V	Data input
Program verify	L	L	L	L	L	H	+12.5 V	+6.5 V	Data output
Program inhibit	L	L	L	X X	L H	L H	+12.5 V	+6.5 V	High impedance
Read	L	L	L	L	L	H	+5.0 V	+5.0 V	Data output
Ouput disable	L	L	L	L	H	X	+5.0 V	+5.0 V	High impedance
Standby	L	L	L	H	X	X	+5.0 V	+5.0 V	High impedance

X can be either H or L.

Figure 18. Pin Functions in μPD78P356 PROM Programming Mode; 100-Pin Plastic QFP



Notes:

Recommended connections for pins not used during PROM programming:

(L) Connect each pin through a resistor to V_{SS}.

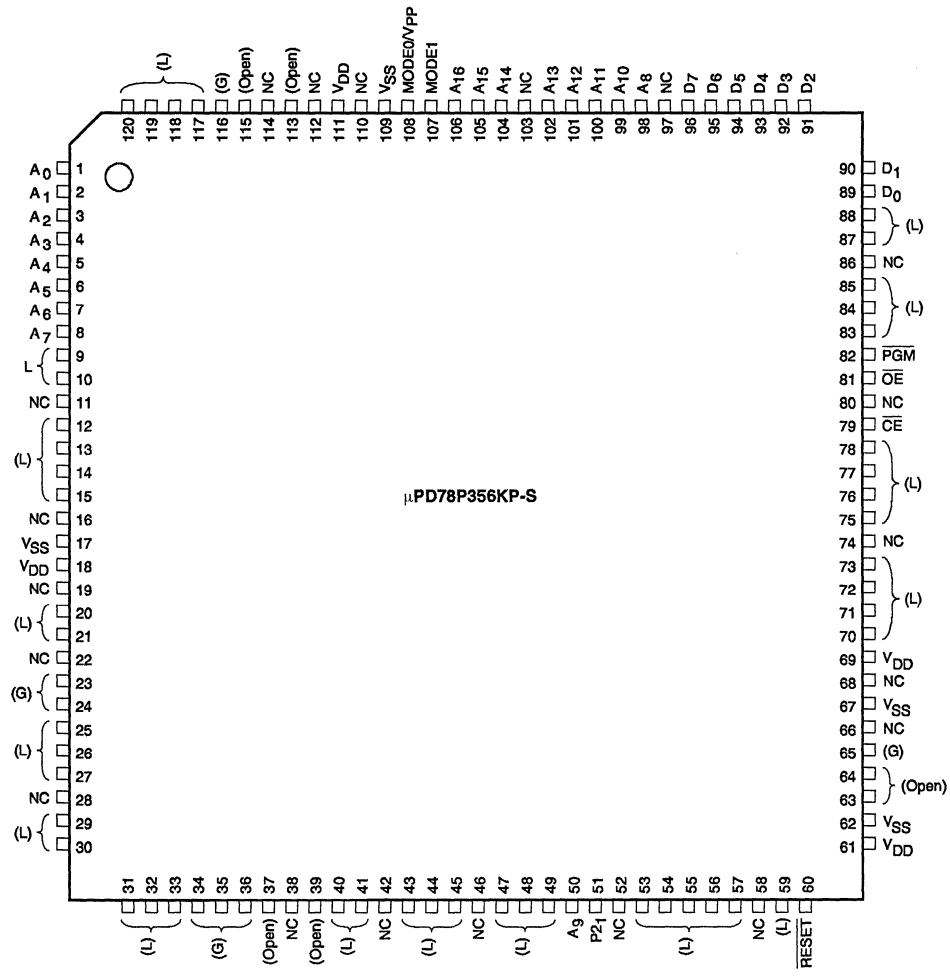
(G) Connect to V_{SS}.

(Open) No connection.

83YL-9407B

5d

Figure 19. Pin Functions in μPD78P356 PROM Programming Mode; 120-Pin Ceramic LCC



Notes:

Recommended connections for pins not used during PROM programming:

(L) Connect each pin through a resistor to V_{SS}.

(G) Connect to V_{SS}.

(Open) No connection.

NC Connect to V_{SS} to prevent noise.

83YL-9408B (10/93)

PROM Byte Programming Procedure

Data can be written to the PROM one byte at a time by the following procedure.

(1) Set the pins not used for programming as indicated in figures 18 and 19. Set MODE0/V_{PP} and V_{DD} pins to +5 V and MODE1, $P2_1$, and $\overline{RESET}$ pins to 0 V. The $\overline{CE}$, $\overline{OE}$, and $\overline{PGM}$ pins should be high.

(2) Supply +6.5 V to V_{DD} pin and +12.5 V to MODE0/V_{PP} pin. Set $\overline{CE}$ pin low and $\overline{OE}$ pin high.

(3) Provide initial address to pins A_0 - A_{16}.

(4) Provide write data.

(5) Input a 0.1-ms program pulse (active low) to $\overline{PGM}$ pin.

(6) Use verify mode (pulse $\overline{OE}$ low) to test data. If data has been written, proceed to step 8; if not, repeat steps 4–6. If data cannot be written in 10 attempts, go to step 7.

(7) Classify PROM as defective and cease write operation.

(8) Increment address.

(9) Repeat steps 4–8 until last address is programmed.

PROM Page Programming Procedure

Data can be written to the PROM four bytes at a time (page programming) by the following procedure.

(1) Set the pins not used for programming as indicated in figures 18 and 19. Set MODE0/V_{PP} and V_{DD} pins to +5 V and MODE1, $P2_1$, and $\overline{RESET}$ pins to 0 V. The $\overline{CE}$, $\overline{OE}$, and $\overline{PGM}$ pins should be high.

(2) Supply +6.5 V to V_{DD} pin and +12.5 V to MODE0/V_{PP} pin. Set $\overline{CE}$ pin low.

(3) Provide initial page address to pins A_0 - A_{16}.

(4) Provide first byte of data and latch it into PROM by pulsing $\overline{OE}$ low. Continue incrementing address and latching in data until four bytes have been loaded.

(5) Input a 0.1-ms program pulse (active low) to $\overline{PGM}$ pin. Data bus D_0 - D_7 is in a high-impedance state.

(6) Use verify mode (pulse $\overline{OE}$ low four times) to test four bytes of data. If all four bytes of data have been written, proceed to step 8; if not, repeat steps 4–6. If data cannot be written in 10 attempts, go to step 7.

(7) Classify PROM as defective and cease write operation.

(8) Increment address.

(9) Repeat steps 4–8 until last address is programmed.

PROM Read Procedure

The contents of the PROM can be read out to the external data bus (D_0 - D_7) by the following procedure.

(1) Set the pins not used for programming as indicated in figures 18 and 19. Set MODE0/V_{PP} and V_{DD} pins to +5 V and MODE1, $P2_1$, and $\overline{RESET}$ pins to 0 V. The $\overline{CE}$, $\overline{OE}$, and $\overline{PGM}$ pins should be high.

(2) Supply +5 V to V_{DD} pin and MODE0/V_{PP} pin.

(3) Input address of data to be read to pins A_0 - A_{16}.

(4) Put an active-low pulse on $\overline{CE}$ and $\overline{OE}$ pins.

(5) Data is output to pins D_0 - D_7.

Program Erasure

The UV EPROM can be erased by exposing the window to light having a wavelength shorter than 400 nm, including ultraviolet, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15 Ws/cm^2 (ultraviolet ray intensity x exposure time) is required to completely erase the written data. Erasure by an ultraviolet lamp rated at 12,000 μW/cm^2 takes 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

DC Programming Characteristics

T_A = 25°C ±5°C; V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH1}	2.2		V_{DD}	V	(Note 1)
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	(Note 2)
V_{DDP} power supply voltage	V_{DDP}	6.25	6.5	6.75	V	Memory program mode
		4.5	5.0	5.5	V	Memory read mode
V_{PP} power supply voltage	V_{PP}	12.2	12.5	12.8	V	Memory program mode
			V_{PP} = V_{DDP}		V	Memory read mode
V_{DDP} power supply current	I_{DDP}			30	mA	Memory program mode
				100	mA	Memory read mode
V_{PP} power supply current	I_{PP}			50	mA	Memory program mode
			1	100	μA	Memory read mode

Notes:

(1) All except pins in Note 2.

(2) Pins $\overline{\text{RESET}}$, X1, $P0_0$ - $P0_3$, P2, $P3_2$ - $P3_6$, $P8_0$ - $P8_1$, $P8_5$, $P10_1$ - $P10_2$, $P10_4$ - $P10_7$.

AC Programming Characteristics

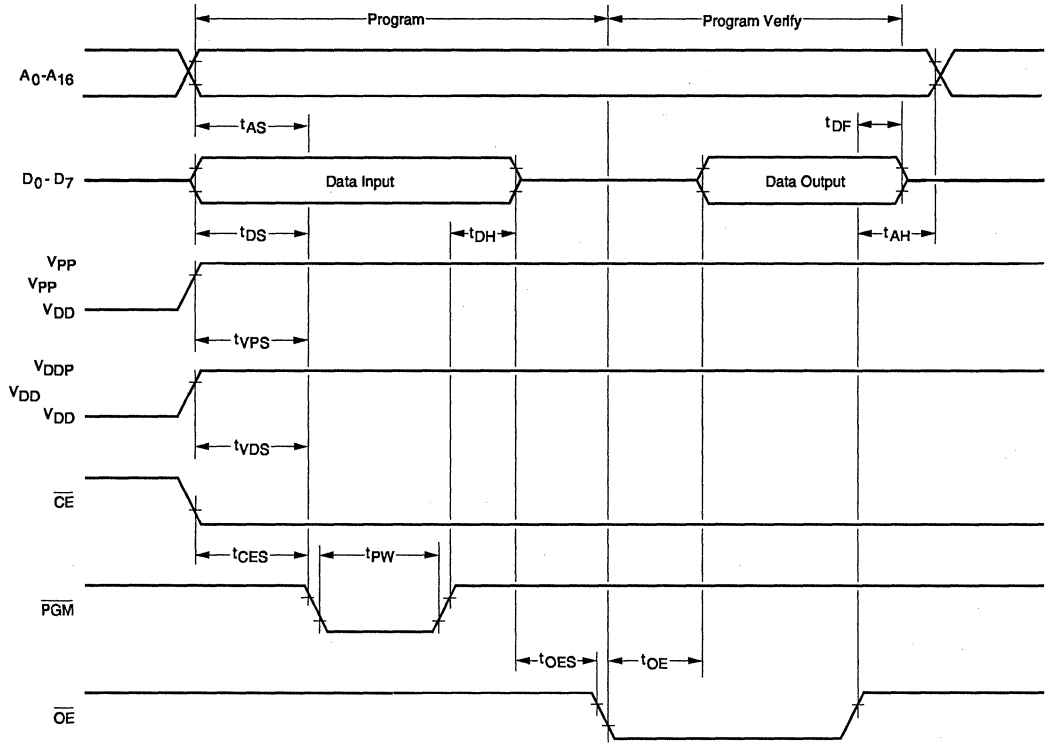
T_A = 25°C ±5°C; V_{DD} = 6.5 ±0.25 V; V_{PP} = 12.5 ±0.3 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Byte Programming Mode						
Address setup time to $\overline{PGM}$ ↓	t_{AS}	2			μs	
$\overline{CE}$ setup time to $\overline{PGM}$ ↓	t_{CES}	2			μs	
Input data setup time to $\overline{PGM}$ ↓	t_{DS}	2			μs	
Address hold time after $\overline{OE}$ ↑	t_{AH}	2			μs	
Input data hold time after $\overline{PGM}$ ↑	t_{DH}	2			μs	
Output data hold time after $\overline{OE}$ ↑	t_{DF}	0		130	ns	
V_{PP} setup time before $\overline{PGM}$ ↓	t_{VPS}	2			μs	
V_{DD} setup time before $\overline{PGM}$ ↓	t_{VDS}	2			μs	
Program pulse width	t_{PW}	0.095	0.1	0.105	ms	
Data to $\overline{OE}$ ↓ delay time	t_{OES}	2			μs	
$\overline{OE}$ ↓ to data output time	t_{OE}			150	ns	
Page Programming Mode						
Address setup time to $\overline{OE}$ ↓	t_{AS}	2			μs	
$\overline{CE}$ setup time to $\overline{OE}$ ↓	t_{CES}	2			μs	
Input data setup time to $\overline{OE}$ ↓	t_{DS}	2			μs	
Address hold time from $\overline{OE}$ ↑	t_{AH}	2			μs	
	t_{AHL}	2			μs	
	t_{AHV}	0			μs	
Input data hold time after $\overline{OE}$ ↑	t_{DH}	2			μs	
Output data hold time after $\overline{OE}$ ↑	t_{DF}	0		130	ns	
V_{PP} setup time to $\overline{OE}$ ↓	t_{VPS}	2			μs	
V_{DD} setup time to $\overline{OE}$ ↓	t_{VDS}	2			μs	
Program pulse width	t_{PW}	0.095	0.1	0.105	ms	
Address to $\overline{OE}$ ↓ delay time	t_{OES}	2			μs	
$\overline{OE}$ ↓ to data output time	t_{OE}			150	ns	
$\overline{OE}$ pulse width during data latch	t_{LW}	1			μs	
Data to $\overline{PGM}$ ↓ delay time	t_{PGMS}	2			μs	
$\overline{CE}$ hold time from $\overline{PGM}$ ↑	t_{CEH}	2			μs	
$\overline{CE}$ hold time from $\overline{OE}$ ↑	t_{OEH}	2			μs	
Read Mode						
Address to data output time	t_{ACC}			200	ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}
$\overline{CE}$ ↓ to data output time	t_{CE}			200	ns	$\overline{OE}$ = V_{IL}
$\overline{OE}$ ↓ to data output time	t_{OE}			75	ns	$\overline{CE}$ = V_{IL}
Data hold time from $\overline{OE}$ ↑	t_{DF}	0		60	ns	$\overline{CE}$ = V_{IL}
Data hold time from address	t_{OH}	0			ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}

5d

PROM Timing Diagrams

Byte Programming Mode



Notes:

[1] V_{DD} must be applied before V_{PP} and removed after V_{PP}.

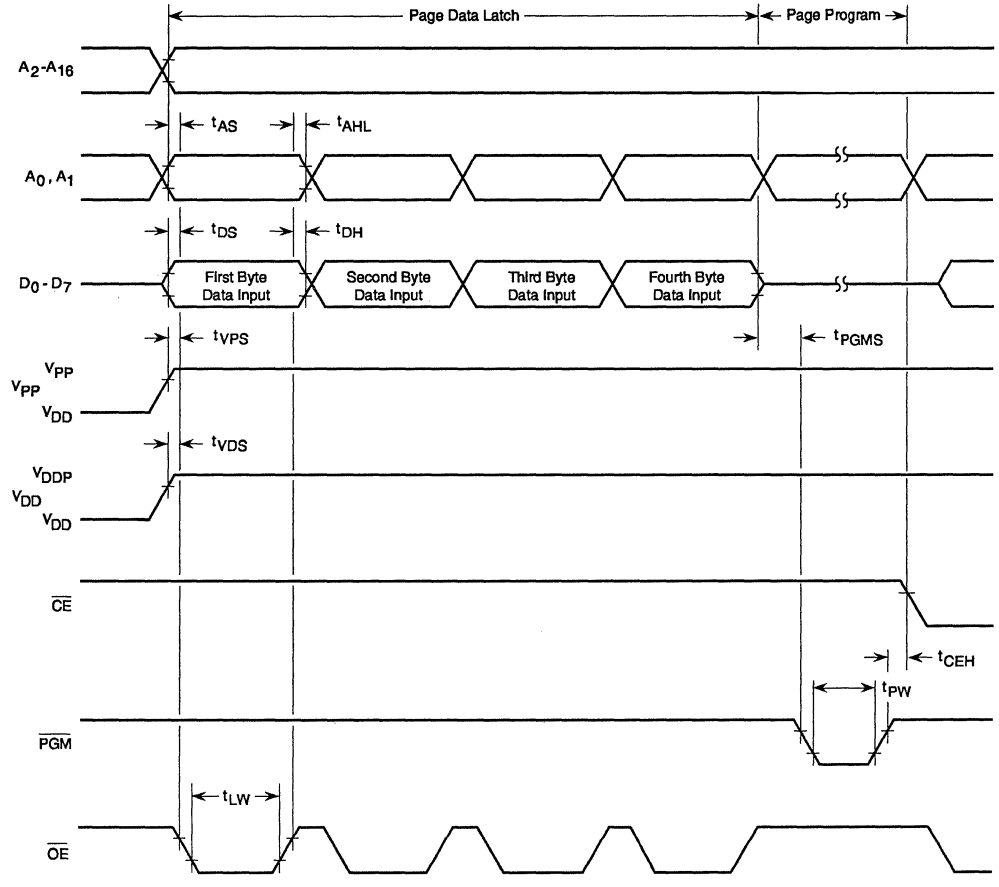
[2] V_{PP} must not be greater than +13.5 V, including overshoot.

[3] Removing and reinserting the device while a voltage of +12.5 V is applied to pin V_{PP} may affect device reliability.

83CL-9156B

PROM Timing Diagrams (cont)

Page Programming Mode; Page Data Latch → Page Program



Notes:

[1] V_{DD} must be applied before V_{PP} and removed after V_{PP}.

[2] V_{PP} must not be greater than +13.5 V, including overshoot.

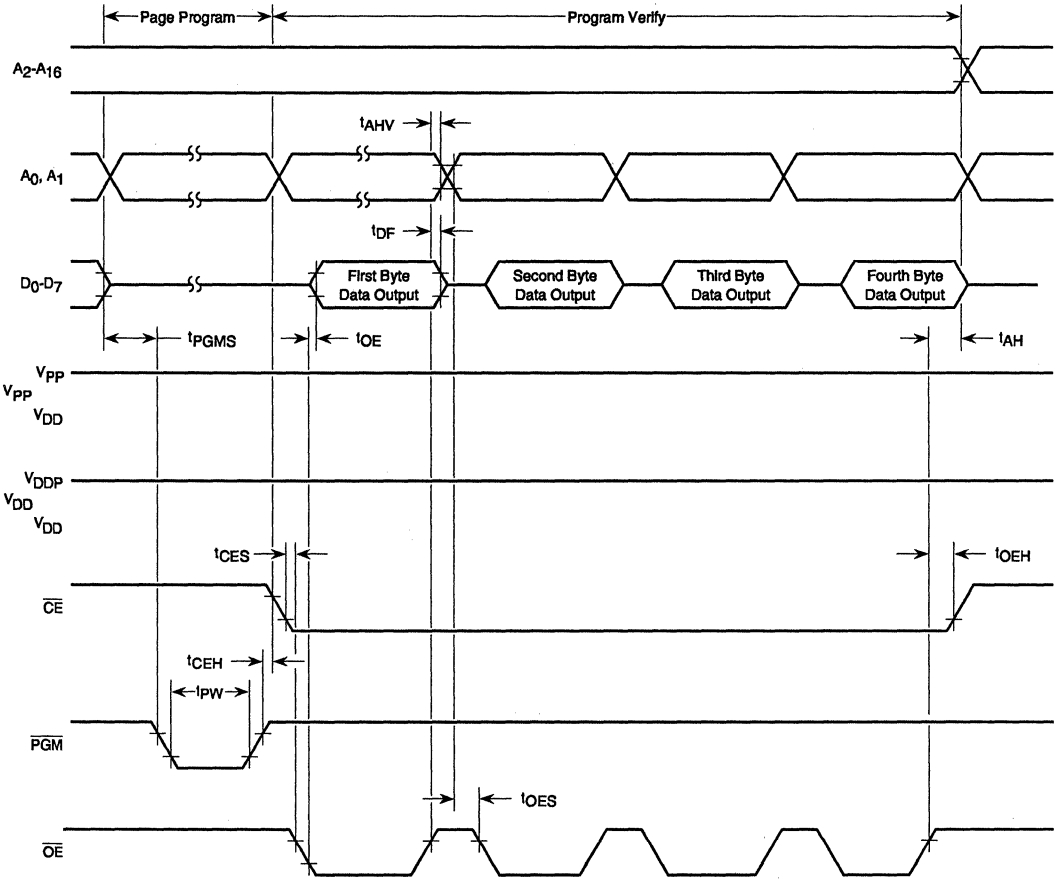
[3] Removing and reinserting the device while a voltage of +12.5 V is applied to pin V_{PP} may affect device reliability.

83CL-9156B

5d

PROM Timing Diagrams (cont)

Page Programming Mode; Page Program → Program Verify



Notes:

[1] V_{DD} must be applied before V_{PP} and removed after V_{PP}.

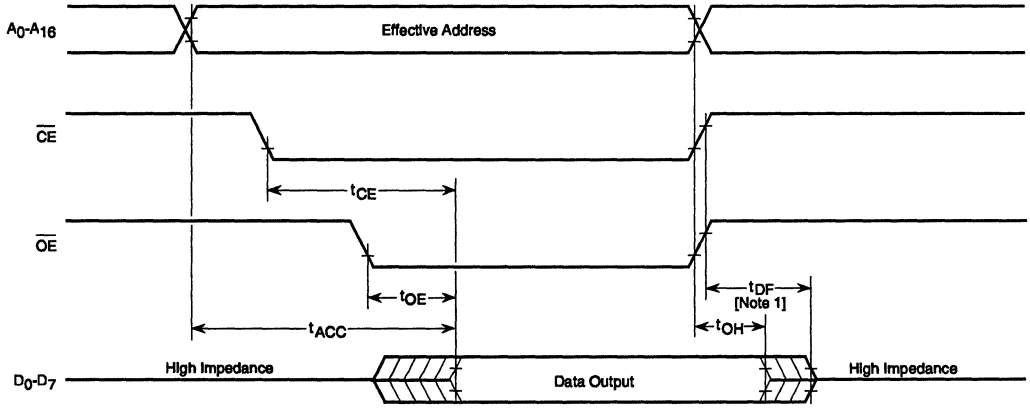
[2] V_{PP} must not be greater than +13.5 V, including overshoot.

[3] Removing and reinserting the device while a voltage of +12.5 V is applied to pin V_{PP} may affect device reliability.

83CL-9157B

PROM Timing Diagrams (cont)

Read Mode



Note:
[1] t_{DF} is specified from $\overline{OE}$ or $\overline{CE}$, whichever goes high first.

83CL-9158B

5d

INSTRUCTION SET

The instruction set of the μPD78356 family is upward compatible with the μPD78322, and μPD78352 families. Four new instructions (MACW, MACSW, SACW, and MOVTBL) have been added to the μPD78322 and two (MACSW and SACW) to the μPD78352 . These additional instructions facilitate digital signal processing.

Convolution instruction MACW calculates the sum of the products of "n" pairs of terms stored in main RAM. The value of "n" is limited only by the amount of main RAM available. The operation of the convolution instruction with saturation word MACSW is identical to instruction MACW except when the instuction terminates with the P/V flag set. In this case, the AXDE register will be set to 7FFFFFFFH by an overflow or 80000000H by an underflow.

Correlation instruction SACW subtracts corresponding factors of two tables and calculates the sum of the absolute values of these subtractions. Instruction MOVTBL displaces a data table by one 16-bit word to make room for a new data word.

The instruction set features both 8- and 16-bit data transfer, arithmetic, and logic instructions and single-bit manipulation instructions. String manipulation instructions are also included. Branch instructions exist to test individual bits in the program status word, the 16-bit accumulator, the special function registers, and the saddr portion of on-chip RAM. Instructions range in length from 1 to 6 bytes depending on the instruction and addressing mode.

Flag Column Indicators

Symbol	Action
(blank)	No change
0	Set to 0
1	Set to 1
X	Set or cleared according to result
P	P/V indicates parity of result
V	P/V indicates arithmetic overflow
R	Restored from saved PSW

Instruction Set Symbols

Symbol	Definition
r	R0, R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14, R15
r1	R0, R1, R2, R3, R4, R5, R6, R7
r2	C, B
rp	RP0, RP1, RP2, RP3, RP4, RP5, RP6, RP7*
rp1	RP0, RP1, RP2, RP3, RP4, RP5, RP6, RP7*
rp2	DE, HL, VP, UP
sfr	Special function register, 8 bits
sfrp	Special function register, 16 bits
post	RP0, RP1, RP2, RP3, RP4, RP5/PSW, RP6, RP7. Bits set to 1 indicate register pairs to be pushed/popped to/from stack; RP5 pushed/popped by PUSH/POP, SP is stack pointer; PSW pushed/popped by PUSHU/POPU, RP5 is stack pointer.
mem	Register indirect: [DE], [HL], [DE+], [HL+], [DE–], [HL–], [VP], [UP] Base Index Mode: [DE+A], [HL+A], [DE+B], [HL+B], [VP+DE], [VP+HL] Base Mode: [DE+byte], [HL+byte], [VP+byte], [UP+byte], [SP+byte] Index Mode: word [A], word [B], word [DE], word [HL]
saddr	FE20-FF1FH: Immediate byte addresses one byte in RAM, or label
saddrp	FE20-FF1FH: Immediate byte (bit 0=0) addresses one word in RAM, or label
word	16 bits of immediate data, or label
byte	8 bits of immediate data, or label
jdisp8	8-bit two's complement displacement (immediate data, displacement value –128 to +127)
bit	3 bits of immediate data (bit position in byte), or label
n	3 bits of immediate data
!addr16	16-bit absolute address specified by an immediate address or label
$addr16	Relative branch address or label
addr16	16-bit address
!addr11	11-bit immediate address or label
addr11	0800H-0FFFH: 0800H + (11-bit immediate address), or label
addr5	0040H-007EH: 0040H + 2 X (5-bit immediate address), or label
A	A register (8-bit accumulator)
X	X register
B	B register
C	C register
D	D register

Instruction Set Symbols (cont)

Symbol	Definition
E	E register
H	H register
L	L register
R0-R15	Register 0 to register 15
AX	Register pair AX (16-bit accumulator)
BC	Register pair BC
DE	Register pair DE
HL	Register pair HL
RP0-RP7	Register pair 0 to register pair 7
PC	Program counter
SP	Stack pointer
UP	User stack pointer (RP5)
PSW	Program status word
PSWH	High-order 8 bits of PSW
PSWL	Low-order 8 bits of PSW
CY	Carry flag
AC	Auxiliary carry flag
Z	Zero flag
P/V	Parity/overflow flag
S	Sign flag
TPF	Table position flag
RBS	Register bank select flag
RSS	Register set select flag
IE	Interrupt enable flag
STBC	Standby control register
WDM	Watchdog timer mode register
( )	Contents of the location whose address is within parentheses; (+) and (–) indicate that the address is incremented after or decremented after it is used
(( ))	Contents of the memory location defined by the quantity within the sets of parentheses
xxH	Hexadecimal quantity
X_H, X_L	High-order 8 bits and low-order 8 bits of X
∧	Logical product (AND)
∨	Logical sum (OR)
∀	Exclusive logical sum (exclusive OR)
—	Inverted data

* rp and rp1 describe the same registers but generate different machine code.

5d

Instruction Set

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
8-Bit Data Transfer								
MOV	r1, #byte	r1 ← byte	2					
	saddr, #byte	(saddr) ← byte	3					
	sfr, #byte (Note 1)	sfr ← byte	3					
	r, r1	r ← r1	2					
	A, r1	A ← r1	1					
	A, saddr	A ← (saddr)	2					
	saddr, A	(saddr) ← A	2					
	saddr, saddr	(saddr) ← (saddr)	3					
	A, sfr	A ← sfr	2					
	sfr, A	sfr ← A	2					
	A, mem (Note 2)	A ← (mem)	1					
	A, mem	A ← (mem)	2-4					
	mem, A (Note 2)	(mem) ← A	1					
	mem, A	(mem) ← A	2-4					
	A, [saddrp]	A ← ((saddrp))	2					
	[saddrp], A	((saddrp)) ← A	2					
	A, !addr16	A ← (addr16)	4					
	!addr16, A	(addr16) ← A	4					
	PSWL, #byte	PSWL ← byte	3	X	X	X	X	X
	PSWH, #byte	PSWH ← byte	3					
	PSWL, A	PSWL ← A	2	X	X	X	X	X
	PSWH, A	PSWH ← A	2					
	A, PSWL	A ← PSWL	2					
	A, PSWH	A ← PSWH	2					
XCH	A, r1	A ↔ r1	1					
	r, r1	r ↔ r1	2					
	A, mem	A ↔ (mem)	2-4					
	A, saddr	A ↔ (saddr)	2					
	A, sfr	A ↔ sfr	3					
	A, [saddrp]	A ↔ ((saddrp))	2					
	saddr, saddr	(saddr) ↔ (saddr)	3					
16-Bit Data Transfer								
MOVW	rp1, #word	rp1 ← word	3					
	saddrp, #word	(saddrp) ← word	4					
	sfrp, #word	sfrp ← word	4					
	rp, rp1	rp ← rp1	2					
	AX, saddrp	AX ← (saddrp)	2					
	saddrp, AX	(saddrp) ← AX	2					
	saddrp, saddrp	(saddrp) ← (saddrp)	3					

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
16-Bit Data Transfer (cont)								
MOVW (cont)	AX, sfrp	AX ← sfrp	2					
	sfrp, AX	sfrp ← AX	2					
	rp1, !addr16	rp1 ← (addr16)	4					
	!addr16, rp1	(addr16) ← rp1	4					
	AX, mem	AX ← (mem)	2-4					
	mem, AX	(mem) ← AX	2-4					
XCHW	AX, saddrp	AX ↔ (saddrp)	2					
	AX, sfrp	AX ↔ sfrp	3					
	saddrp, saddrp	(saddrp) ↔ (saddrp)	3					
	rp, rp1	rp ↔ rp1	2					
	AX, mem	AX ↔ (mem)	2-4					
8-Bit Arithmetic								
ADD	A, #byte	A, CY ← A + byte	2	X	X	X	V	X
	saddr, #byte	(saddr), CY ← (saddr) + byte	3	X	X	X	V	X
	sfr, #byte	sfr, CY ← sfr + byte	4	X	X	X	V	X
	r, r1	r, CY ← r + r1	2	X	X	X	V	X
	A, saddr	A, CY ← A + (saddr)	2	X	X	X	V	X
	A, sfr	A, CY ← A + sfr	3	X	X	X	V	X
	saddr, saddr	(saddr), CY ← (saddr) + (saddr)	3	X	X	X	V	X
	A, mem	A, CY ← A + (mem)	2-4	X	X	X	V	X
	mem, A	(mem), CY ← (mem) + A	2-4	X	X	X	V	X
ADDC	A, #byte	A, CY ← A + byte + CY	2	X	X	X	V	X
	saddr, #byte	(saddr), CY ← (saddr) + byte + CY	3	X	X	X	V	X
	sfr, #byte	sfr, CY ← sfr + byte + CY	4	X	X	X	V	X
	r, r1	r, CY ← r + r1 + CY	2	X	X	X	V	X
	A, saddr	A, CY ← A + (saddr) + CY	2	X	X	X	V	X
	A, sfr	A, CY ← A + sfr + CY	3	X	X	X	V	X
	saddr, saddr	(saddr), CY ← (saddr) + (saddr) + CY	3	X	X	X	V	X
	A, mem	A, CY ← A + (mem) + CY	2-4	X	X	X	V	X
	mem, A	(mem), CY ← (mem) + A + CY	2-4	X	X	X	V	X
SUB	A, #byte	A, CY ← A – byte	2	X	X	X	V	X
	saddr, #byte	(saddr), CY ← (saddr) – byte	3	X	X	X	V	X
	sfr, #byte	sfr, CY ← sfr – byte	4	X	X	X	V	X
	r, r1	r, CY ← r – r1	2	X	X	X	V	X
	A, saddr	A, CY ← A – (saddr)	2	X	X	X	V	X
	A, sfr	A, CY ← A – sfr	3	X	X	X	V	X
	saddr, saddr	(saddr), CY ← (saddr) – (saddr)	3	X	X	X	V	X
	A, mem	A, CY ← A – (mem)	2-4	X	X	X	V	X
	mem, A	(mem), CY ← (mem) – A	2-4	X	X	X	V	X

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
8-Bit Arithmetic (cont)								
SUBC	A, #byte	A, CY ← A – byte – CY	2	X	X	X	V	X
	saddr, #byte	(saddr), CY ← (saddr) – byte – CY	3	X	X	X	V	X
	sfr, #byte	sfr, CY ← sfr – byte – CY	4	X	X	X	V	X
	r, r1	r, CY ← r – r1 – CY	2	X	X	X	V	X
	A, saddr	A, CY ← A – (saddr) – CY	2	X	X	X	V	X
	A, sfr	A, CY ← A – sfr – CY	3	X	X	X	V	X
	saddr, saddr	(saddr), CY ← (saddr) – (saddr) – CY	3	X	X	X	V	X
	A, mem	A, CY ← A – (mem) – CY	2-4	X	X	X	V	X
	mem, A	(mem), CY ← (mem) – A – CY	2-4	X	X	X	V	X
8-Bit Logic								
AND	A, #byte	A ← A ∧ byte	2	X	X		P	
	saddr, #byte	(saddr) ← (saddr) ∧ byte	3	X	X		P	
	sfr, #byte	sfr ← sfr ∧ byte	4	X	X		P	
	r, r1	r ← r ∧ r1	2	X	X		P	
	A, saddr	A← A ∧ (saddr)	2	X	X		P	
	A, sfr	A← A ∧ sfr	3	X	X		P	
	saddr, saddr	(saddr)← (saddr) ∧ (saddr)	3	X	X		P	
	A, mem	A ← A ∧ (mem)	2-4	X	X		P	
	mem, A	(mem) ← (mem) ∧ A	2-4	X	X		P	
OR	A, #byte	A ← A ∨ byte	2	X	X		P	
	saddr, #byte	(saddr) ← (saddr) ∨ byte	3	X	X		P	
	sfr, #byte	sfr ← sfr ∨ byte	4	X	X		P	
	r, r1	r ← r ∨ r1	2	X	X		P	
	A, saddr	A ← A ∨ (saddr)	2	X	X		P	
	A, sfr	A ← A ∨ sfr	3	X	X		P	
	saddr, saddr	(saddr) ← (saddr) ∨ (saddr)	3	X	X		P	
	A, mem	A ← A ∨ (mem)	2-4	X	X		P	
	mem, A	(mem) ← (mem) ∨ A	2-4	X	X		P	
XOR	A, #byte	A ← A ⊻ byte	2	X	X		P	
	saddr, #byte	(saddr) ← (saddr) ⊻ byte	3	X	X		P	
	sfr, #byte	sfr ← sfr ⊻ byte	4	X	X		P	
	r, r1	r ← r ⊻ r1	2	X	X		P	
	A, saddr	A ← A ⊻ (saddr)	2	X	X		P	
	A, sfr	A ← A ⊻ sfr	3	X	X		P	
	saddr, saddr	(saddr) ← (saddr) ⊻ (saddr)	3	X	X		P	
	A, mem	A ← A ⊻ (mem)	2-4	X	X		P	
	mem, A	(mem) ← (mem) ⊻ A	2-4	X	X		P	

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
8-Bit Logic (cont)								
CMP	A, #byte	A – byte	2	X	X	X	V	X
	saddr, #byte	(saddr) – byte	3	X	X	X	V	X
	sfr, #byte	sfr – byte	4	X	X	X	V	X
	r, r1	r – r1	2	X	X	X	V	X
	A, saddr	A – (saddr)	2	X	X	X	V	X
	A, sfr	A – sfr	3	X	X	X	V	X
	saddr, saddr	(saddr) – (saddr)	3	X	X	X	V	X
	A, mem	A – (mem)	2-4	X	X	X	V	X
	mem, A	(mem) – A	2-4	X	X	X	V	X
16-Bit Arithmetic								
ADDW	AX, #word	AX, CY ← AX + word	3	X	X	X	V	X
	saddrp, #word	(saddrp), CY ← (saddrp) + word	4	X	X	X	V	X
	sfrp, #word	sfrp, CY ← sfrp + word	5	X	X	X	V	X
	rp, rp1	rp, CY ← rp + rp1	2	X	X	X	V	X
	AX, saddrp	AX, CY ← AX + (saddrp)	2	X	X	X	V	X
	AX, sfrp	AX, CY ← AX + sfrp	3	X	X	X	V	X
	saddrp, saddrp	(saddrp), CY ← (saddrp) + (saddrp)	3	X	X	X	V	X
SUBW	AX, #word	AX, CY ← AX – word	3	X	X	X	V	X
	saddrp, #word	(saddrp), CY ← (saddrp) – word	4	X	X	X	V	X
	sfrp, #word	sfrp, CY ← sfrp – word	5	X	X	X	V	X
	rp, rp1	rp, CY ← rp – rp1	2	X	X	X	V	X
	AX, saddrp	AX, CY ← AX – (saddrp)	2	X	X	X	V	X
	AX, sfrp	AX, CY ← AX – sfrp	3	X	X	X	V	X
	saddrp, saddrp	(saddrp), CY ← (saddrp) – (saddrp)	3	X	X	X	V	X
CMPW	AX, #word	AX – word	3	X	X	X	V	X
	saddrp, #word	(saddrp) – word	4	X	X	X	V	X
	sfrp, #word	sfrp – word	5	X	X	X	V	X
	rp, rp1	rp – rp1	2	X	X	X	V	X
	AX, saddrp	AX – (saddrp)	2	X	X	X	V	X
	AX, sfrp	AX – sfrp	3	X	X	X	V	X
	saddrp, saddrp	(saddrp) – (saddrp)	3	X	X	X	V	X
Multiplication/Division								
MULU	r1	AX ← A x r1	2					
DIVUW	r1	AX (quotient), r1 (remainder) ← AX ÷ r1	2					
MULUW	rp1	AX (high-order 16 bits), rp1 (low-order 16 bits) ← AX x rp1	2					
DIVUX	rp1	AXDE (quotient), rp1 (remainder) ← AXDE ÷ rp1	2					
MULW (Note 3)	rp1	AX (high-order 16 bits), rp1 (low-order 16 bits) ← AX x rp1	2					

5d

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Sum-of-Products								
MACW	n	AXDE ← (B) x (C) + AXDE, B ← B + 2, C ← C + 2, n ← n–1. End if n = 0 or P/V = 1	3	X	X	X	V	X
Sum-of-Products With Saturation								
MACSW	n	AXDE ← (B) x (C) + AXDE, B ← B+2, C ← C + 2, n ← n–1, if overflow (P/V = 1) then AXDE ← 7FFFFFFFH, if underflow (P/V = 1) then AXDE ← 80000000H. End if n = 0 or P/V = 1	3	x	x	x	x	x
Correlation Operation								
SACW	[DE+], [HL+]	AX ← AX + |(DE)–(HL)|, DE ← DE+2, HL ← HL + 2, C ← C–1. End if C = 0 or CY = 1	4	x	x	x	V	x
Table Shift								
MOVTBLW	!addr 16, n (Note 4)	(addr16 + 2) ← (addr16), n ← n–1, addr16 ← addr16 – 2. End if n = 0	4					
Increment/Decrement								
INC	r1	r1 ← r1 + 1	1	X	X	X	V	
	saddr	(saddr) ← (saddr) + 1	2	X	X	X	V	
DEC	r1	r1 ← r1 – 1	1	X	X	X	V	
	saddr	(saddr) ← (saddr) – 1	2	X	X	X	V	
INCW	rp2	rp2 ← rp2 + 1	1					
	saddrp	(saddrp) ← (saddrp) + 1	3					
DECW	rp2	rp2 ← rp2 – 1	1					
	saddrp	(saddrp) ← (saddrp) – 1	3					
Shift/Rotate								
ROR	r1, n	(CY, $r1_7$ ← $r1_0$, $r1_{m-1}$ ← $r1_m$) x n times	2				P	X
ROL	r1, n	(CY, $r1_0$ ← $r1_7$, $r1_{m+1}$ ← $r1_m$) x n times	2				P	X
RORC	r1, n	(CY ← $r1_0$, $r1_7$ ← CY, $r1_{m-1}$ ← $r1_m$) x n times	2				P	X
ROLC	r1, n	(CY ← $r1_7$, $r1_0$ ← CY, $r1_{m+1}$ ← $r1_m$) x n times	2				P	X
SHR	r1, n	(CY ← $r1_0$, $r1_7$ ← 0, $r1_{m-1}$ ← $r1_m$) x n times	2	X	X	0	P	X
SHL	r1, n	(CY ← $r1_7$, $r1_0$ ← 0, $r1_{m+1}$ ← $r1_m$) x n times	2	X	X	0	P	X
SHRW	rp1, n	(CY ← $rp1_0$, $rp1_{15}$ ← 0, $rp1_{m-1}$ ← $rp1_m$) x n times	2	X	X	0	P	X
SHLW	rp1, n	(CY ← $rp1_{15}$, $rp1_0$ ← 0, $rp1_{m+1}$ ← $rp1_m$) x n times	2	X	X	0	P	X
ROR4	[rp1]	A_{3-0} ← $(rp1)_{3-0}$, $(rp1)_{7-4}$ ← A_{3-0}, $(rp1)_{3-0}$ ← $(rp1)_{7-4}$	2					
ROL4	[rp1]	A_{3-0} ← $(rp1)_{7-4}$, $(rp1)_{3-0}$ ← A_{3-0}, $(rp1)_{7-4}$ ← $(rp1)_{3-0}$	2					
BCD Adjustment								
ADJBA		Decimal adjust accumulator after add	2	X	X	X	P	X
ADJBS		Decimal adjust accumulator after subtract	2	X	X	X	P	X
Data Expansion								
CVTBW		X ← A, A_{6-0} ← A_7	1					

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Bit Manipulation								
MOV1	CY, saddr.bit	CY ← (saddr.bit)	3					X
	CY, sfr.bit	CY ← sfr.bit	3					X
	CY, A.bit	CY ← A.bit	2					X
	CY, X.bit	CY ← X.bit	2					X
	CY, PSWH.bit	CY ← PSWH.bit	2					X
	CY, PSWL.bit	CY ← PSWL.bit	2					X
	saddr.bit, CY	(saddr.bit) ← CY	3					
	sfr.bit, CY	sfr.bit ← CY	3					
	A.bit, CY	A.bit ← CY	2					
	X.bit, CY	X.bit ← CY	2					
	PSWH.bit, CY	PSWH.bit ← CY	2					
	PSWL.bit, CY	PSWL.bit ← CY	2	X	X	X	X	
AND1	CY, saddr.bit	CY ← CY ∧ (saddr.bit)	3					X
	CY, /saddr.bit	CY ← CY ∧ $\overline{(saddr.bit)}$	3					X
	CY, sfr.bit	CY ← CY ∧ sfr.bit	3					X
	CY, /sfr.bit	CY ← CY ∧ $\overline{sfr.bit}$	3					X
	CY, A.bit	CY ← CY ∧ A.bit	2					X
	CY, /A.bit	CY ← CY ∧ $\overline{A.bit}$	2					X
	CY, X.bit	CY ← CY ∧ X.bit	2					X
	CY, /X.bit	CY ← CY ∧ $\overline{X.bit}$	2					X
	CY, PSWH.bit	CY ← CY ∧ PSWH.bit	2					X
	CY, /PSWH.bit	CY ← CY ∧ $\overline{PSWH.bit}$	2					X
	CY, PSWL.bit	CY ← CY ∧ PSWL.bit	2					X
	CY, /PSWL.bit	CY ← CY ∧ $\overline{PSWL.bit}$	2					X
OR1	CY, saddr.bit	CY ← CY ∨ (saddr.bit)	3					X
	CY, /saddr.bit	CY ← CY ∨ $\overline{(saddr.bit)}$	3					X
	CY, sfr.bit	CY ← CY ∨ sfr.bit	3					X
	CY, /sfr.bit	CY ← CY ∨ $\overline{sfr.bit}$	3					X
	CY, A.bit	CY ← CY ∨ A.bit	2					X
	CY, /A.bit	CY ← CY ∨ $\overline{A.bit}$	2					X
	CY, X.bit	CY ← CY ∨ X.bit	2					X
	CY, /X.bit	CY ← CY ∨ $\overline{X.bit}$	2					X
	CY, PSWH.bit	CY ← CY ∨ PSWH.bit	2					X
	CY, /PSWH.bit	CY ← CY ∨ $\overline{PSWH.bit}$	2					X
	CY, PSWL.bit	CY ← CY ∨ PSWL.bit	2					X
	CY, /PSWL.bit	CY ← CY ∨ $\overline{PSWL.bit}$	2					X

5d

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Bit Manipulation (cont)								
XOR1	CY, saddr.bit	CY ← CY ∀ (saddr.bit)	3					X
	CY, sfr.bit	CY ← CY ∀ sfr.bit	3					X
	CY, A.bit	CY ← CY ∀ A.bit	2					X
	CY, X.bit	CY ← CY ∀ X.bit	2					X
	CY, PSWH.bit	CY ← CY ∀ PSWH.bit	2					X
	CY, PSWL.bit	CY ← CY ∀ PSWL.bit	2					X
SET1	saddr.bit	(saddr.bit) ← 1	2					
	sfr.bit	sfr.bit ← 1	3					
	A.bit	A.bit ← 1	2					
	X.bit	X.bit ← 1	2					
	PSWH.bit	PSWH.bit ← 1	2					
	PSWL.bit	PSWL.bit ← 1	2	X	X	X	X	X
CLR1	saddr.bit	(saddr.bit) ← 0	2					
	sfr.bit	sfr.bit ← 0	3					
	A.bit	A.bit ← 0	2					
	X.bit	X.bit ← 0	2					
	PSWH.bit	PSWH.bit ← 0	2					
	PSWL.bit	PSWL.bit ← 0	2	X	X	X	X	X
NOT1	saddr.bit	(saddr.bit) ← $\overline{\text{(saddr.bit)}}$	3					
	sfr.bit	sfr.bit ← $\overline{\text{sfr.bit}}$	3					
	A.bit	A.bit ← $\overline{\text{A.bit}}$	2					
	X.bit	X.bit ← $\overline{\text{X.bit}}$	2					
	PSWH.bit	PSWH.bit ← $\overline{\text{PSWH.bit}}$	2					
	PSWL.bit	PSWL.bit ← $\overline{\text{PSWL.bit}}$	2	X	X	X	X	X
SET1	CY	CY ← 1	1					1
CLR1	CY	CY ← 0	1					0
NOT1	CY	CY ← $\overline{\text{CY}}$	1					X
Subroutine Linkage								
CALL	!addr16	(SP–1) ← $(PC + 3)_H$, (SP – 2) ← $(PC + 3)_L$, PC ← addr16, SP ← SP – 2	3					
	rp1	(SP–1) ← $(PC + 2)_H$, (SP – 2) ← $(PC + 2)_L$, PC_H ← $rp1_H$, PC_L ← $rp1_L$, SP ← SP – 2	2					
	[rp1]	(SP–1) ← $(PC + 2)_H$, (SP – 2) ← $(PC + 2)_L$, PC_H ← (rp1 + 1), PC_L ← (rp1), SP ← SP – 2	2					
CALLF	!addr11	(SP–1) ← $(PC + 2)_H$, (SP – 2) ← $(PC + 2)_L$, PC_{15-11} ← 00001, PC_{10-0} ← addr11, SP ← SP – 2	2					
CALLT	[addr5]	(SP–1) ← $(PC + 1)_H$, (SP – 2) ← $(PC + 1)_L$, PC_H ← (TPFx8000H + 2 x addr5 + 41H), PC_L ← (TPFx8000H + 2 x addr5 + 40H), SP ← SP – 2	1					

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Subroutine Linkage (cont)								
BRK		(SP–1) ← PSWH, (SP – 2) ← PSWL, (SP – 3) ← (PC + 1)$_H$, (SP – 4) ← (PC + 1)$_L$, PC$_L$ ← (003EH), PC$_H$ ← (003FH), SP ← SP – 4, IE ← 0	1					
RET		PC$_L$ ← (SP), PC$_H$ ← (SP + 1), SP ← SP + 2	1					
RETB		PC$_L$ ← (SP), PC$_H$ ← (SP + 1), PSWL ← (SP + 2), PSWH ← (SP + 3), SP ← SP + 4	1	R	R	R	R	R
RETI		PC$_L$ ← (SP), PC$_H$ ← (SP + 1), PSWL ← (SP + 2), PSWH ← (SP + 3), SP ← SP + 4	1	R	R	R	R	R
Stack Manipulation								
PUSH	sfrp	(SP – 1) ← sfr$_H$, (SP – 2) ← sfr$_L$, SP ← SP – 2	3					
	post	{(SP – 1) ← rpp$_H$, (SP – 2) ← rpp$_L$, SP ← SP – 2} x n (Note 5)	2					
	PSW	(SP – 1) ← PSWH, (SP – 2) ← PSWL, SP ← SP – 2	1					
PUSHU	post	{(UP – 1) ← rpp$_H$, (UP – 2) ← rpp$_L$, UP ← UP – 2} x n (Note 5)	2					
POP	sfrp	sfr$_L$ ← (SP), sfr$_H$ ← (SP + 1), SP ← SP + 2	3					
	post	{rpp$_L$ ← (SP), rpp$_H$ ← (SP + 1), SP ← SP + 2} x n (Note 5)	2					
	PSW	PSWL ← (SP), PSWH ← (SP + 1), SP ← SP + 2	1	R	R	R	R	R
POPU	post	{rpp$_L$ ← (UP), rpp$_H$ ← (UP + 1), UP ← UP + 2} x n (Note 5)	2					
MOVW	SP, #word	SP ← word	4					
	SP, AX	SP ← AX	2					
	AX, SP	AX ← SP	2					
INCW	SP	SP ← SP + 1	2					
DECW	SP	SP ← SP – 1	2					
Pin Level Test								
CHKL	sfr	(Pin level) $\forall$ (internal signal level)	3	X	X		P	
CHKLA	sfr	A ← (Pin level) $\forall$ (internal signal level)	3	X	X		P	
Unconditional Branch								
BR	!addr16	PC ← addr16	3					
	rp1	PC$_H$ ← rp1$_H$, PC$_L$ ← rp1$_L$	2					
	[rp1]	PC$_H$ ← (rp1 + 1), PC$_L$ ← (rp1)	2					
	$addr16	PC ← PC + 2 + jdisp8	2					
Conditional Branch								
BC, BL	$addr16	PC ← PC + 2 + jdisp8 if CY = 1	2					
BNC, BNL	$addr16	PC ← PC + 2 + jdisp8 if CY = 0	2					
BZ, BE	$addr16	PC ← PC + 2 + jdisp8 if Z = 1	2					
BNZ, BNE	$addr16	PC ← PC + 2 + jdisp8 if Z = 0	2					
BV, BPE	$addr16	PC ← PC + 2 + jdisp8 if P/V = 1	2					

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Conditional Branch (cont)								
BNV, BPO	$addr16	PC ← PC + 2 + jdisp8 if P/V = 0	2					
BN	$addr16	PC ← PC + 2 + jdisp8 if S = 1	2					
BP	$addr16	PC ← PC + 2 + jdisp8 if S = 0	2					
BGT	$addr16	PC ← PC + 3 + jdisp8 if (P/V ∀ S) ∨ Z = 0	3					
BGE	$addr16	PC ← PC + 3 + jdisp8 if P/V ∀ S = 0	3					
BLT	$addr16	PC ← PC + 3 + jdisp8 if P/V ∀ S = 1	3					
BLE	$addr16	PC ← PC + 3 + jdisp8 if (P/V ∀ S) ∨ Z = 1	3					
BH	$addr16	PC ← PC + 3 + jdisp8 if Z ∨ CY = 0	3					
BNH	$addr16	PC ← PC + 3 + jdisp8 if Z ∨ CY = 1	3					
BT	saddr.bit, $addr16	PC ← PC + 3 + jdisp8 if (saddr.bit) = 1	3					
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 1	4					
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 1	3					
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 1	3					
	PSWH.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWH.bit = 1	3					
	PSWL.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWL.bit = 1	3					
BF	saddr.bit, $addr16	PC ← PC + 4 + jdisp8 if (saddr.bit) = 0	4					
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 0	4					
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 0	3					
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 0	3					
	PSWH.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWH.bit = 0	3					
	PSWL.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWL.bit = 0	3					
BTCLR	saddr.bit, $addr16	PC ← PC + 4 + jdisp8 if (saddr.bit) = 1 then reset (saddr.bit)	4					
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit	4					
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 1 then reset A.bit	3					
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 1 then reset X.bit	3					
	PSWH.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWH.bit = 1 then reset PSWH.bit	3					
	PSWL.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWL.bit = 1 then reset PSWL.bit	3	X	X	X	X	X
BFSET	saddr.bit, $addr16	PC ← PC + 4 + jdisp8 if (saddr.bit) = 0 then set (saddr.bit)	4					
	sfr.bit, $addr16	PC ← PC + 4 + jdisp8 if sfr.bit = 0 then set sfr.bit	4					
	A.bit, $addr16	PC ← PC + 3 + jdisp8 if A.bit = 0 then set A.bit	3					
	X.bit, $addr16	PC ← PC + 3 + jdisp8 if X.bit = 0 then set X.bit	3					
	PSWH.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWH.bit = 0 then set PSWH.bit	3					
	PSWL.bit, $addr16	PC ← PC + 3 + jdisp8 if PSWL.bit = 0 then set PSWL.bit	3	X	X	X	X	X

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
Conditional Branch (cont)								
DBNZ	r2, $addr16	r2 ← r2 – 1, then PC ← PC + 2 + jdisp8 if r2 = 0	2					
	saddr, $addr16	(saddr) ← (saddr) – 1, then PC ← PC + 3 + jdisp8 if (saddr) = 0	3					
Context Switching								
BRKCS	RBn	RBS_{2-0} ← n, PC_H ↔ R5, PC_L ↔ R4, R7 ← PSWH, R6 ← PSWL, RSS ← 0, IE ← 0	2					
RETCS	!addr16	PC_H ← R5, PC_L ← R4, R5 ← $addr16_H$, R4 ← $addr16_L$, PSWH ← R7, PSWL ← R6	3	R	R	R	R	R
RETCSB	!addr16	PC_H ← R5, PC_L ← R4, R5 ← $addr16_H$, R4 ← $addr16_L$, PSWH ← R7, PSWL ← R6	4	R	R	R	R	R
String Manipulation								
MOVM	[DE+], A	(DE+) ← A, C ← C–1 End if C = 0	2					
	[DE–], A	(DE–) ← A, C ← C–1 End if C = 0	2					
MOVBK	[DE+], [HL+]	(DE+) ← (HL+), C ← C–1 End if C = 0	2					
	[DE–], [HL–]	(DE–) ← (HL–), C ← C–1 End if C = 0	2					
XCHM	[DE+], A	(DE+) ↔ A, C ← C–1 End if C = 0	2					
	[DE–], A	(DE–) ↔ A, C ← C–1 End if C = 0	2					
XCHBK	[DE+], [HL+]	(DE+) ↔ (HL+), C ← C–1 End if C = 0	2					
	[DE–], [HL–]	(DE–) ↔ (HL–), C ← C–1 End if C = 0	2					
CMPME	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X
CMPBKE	[DE+], [HL+]	(DE+) – (HL+), C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or Z = 0	2	X	X	X	V	X
CMPMNE	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X
CMPBKNE	[DE+], [HL+]	(DE+) – (HL+), C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or Z = 1	2	X	X	X	V	X
CMPMC	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X
CMPBKC	[DE+], [HL+]	(DE+) – (HL+), C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or CY = 0	2	X	X	X	V	X
CMPMNC	[DE+], A	(DE+) – A, C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X
	[DE–], A	(DE–) – A, C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X
CMPBKNC	[DE+], [HL+]	(DE+) – (HL+), C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X
	[DE–], [HL–]	(DE–) – (HL–), C ← C–1 End if C = 0 or CY = 1	2	X	X	X	V	X

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Flags S	Z	AC	P/V	CY
CPU Control								
MOV	STBC, #byte	STBC ← byte (Note 6)	4					
	WDM, #byte	WDM ← byte (Note 6)	4					
SWRS		RSS ← $\overline{RSS}$	1					
SEL	RBn	RBS_{2-0} ← n, RSS ← 0	2					
	RBn, ALT	RBS_{2-0} ← n, RSS ← 1	2					
NOP		No operation	1					
EI		IE ← 1 (Enable interrupt)	1					
DI		IE ← 0 (Disable interrupt)	1					

Notes:

(1) A special instruction is used to write to STBC and WDM.

(2) One byte move instruction when [DE], [HL], [DE+], [DE–], [HL+], or [HL–] is specified for mem.

(3) 16-bit signed multiply instruction

(4) Addressing range is 0FE00H to 0FEFFH.

(5) rpp refers to register pairs specified in post byte. "n" is the number of register pairs specified in post byte.

(6) Trap if data bytes in operation code are not one's complement. If trap, then:
(SP–1) ← PSWH, (SP–2) ← PSWL, (SP–3) ← $(PC–4)_H$,
(SP–4) ← $(PC–4)_L$, PC_L ← (003CH), PC_H ← (003DH).
SP ← SP–4, IE ← 0.

NEC

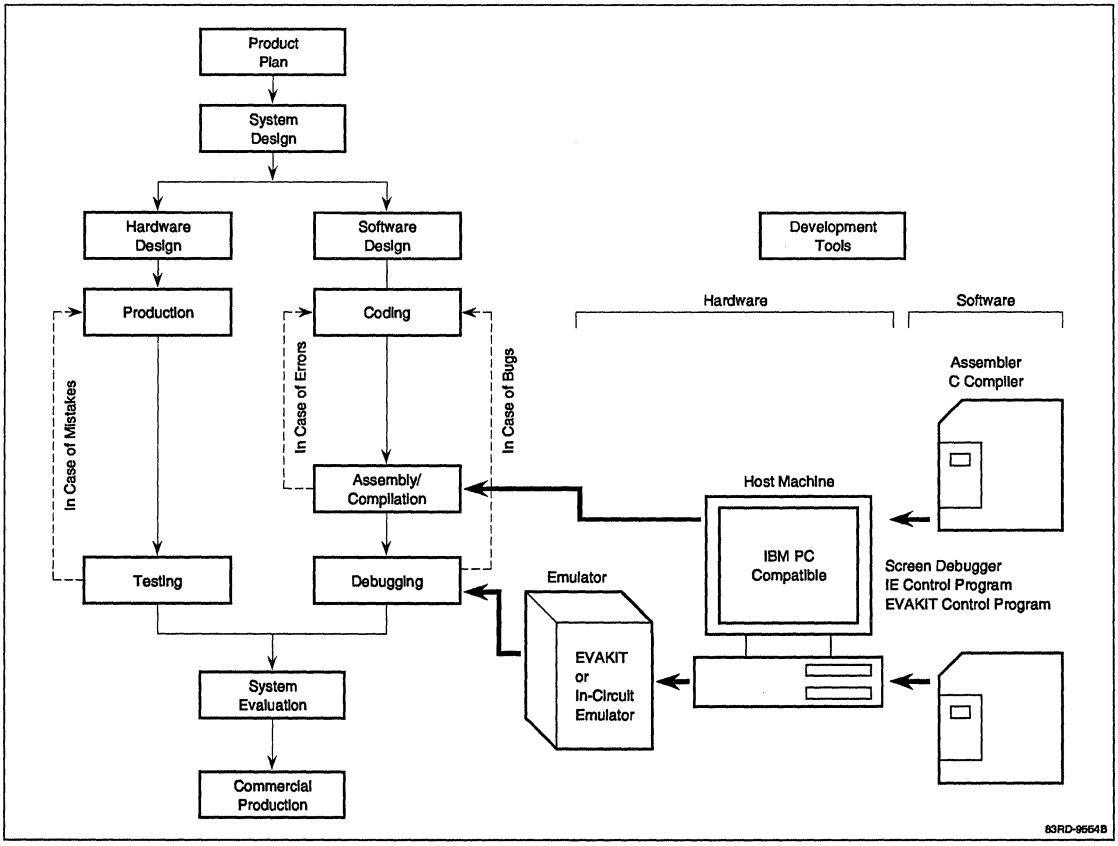
Section 6
Development Tools

NEC provides a variety of development support tools to enable effective program development and hardware development for various microcontrollers. The development tools are used for efficient software assembly/compilation or debugging.

A debugger can be used alone in a debugging system. However, development time and cost can be reduced by using symbolic debugging, which is made available by connecting the debugger to a host machine.

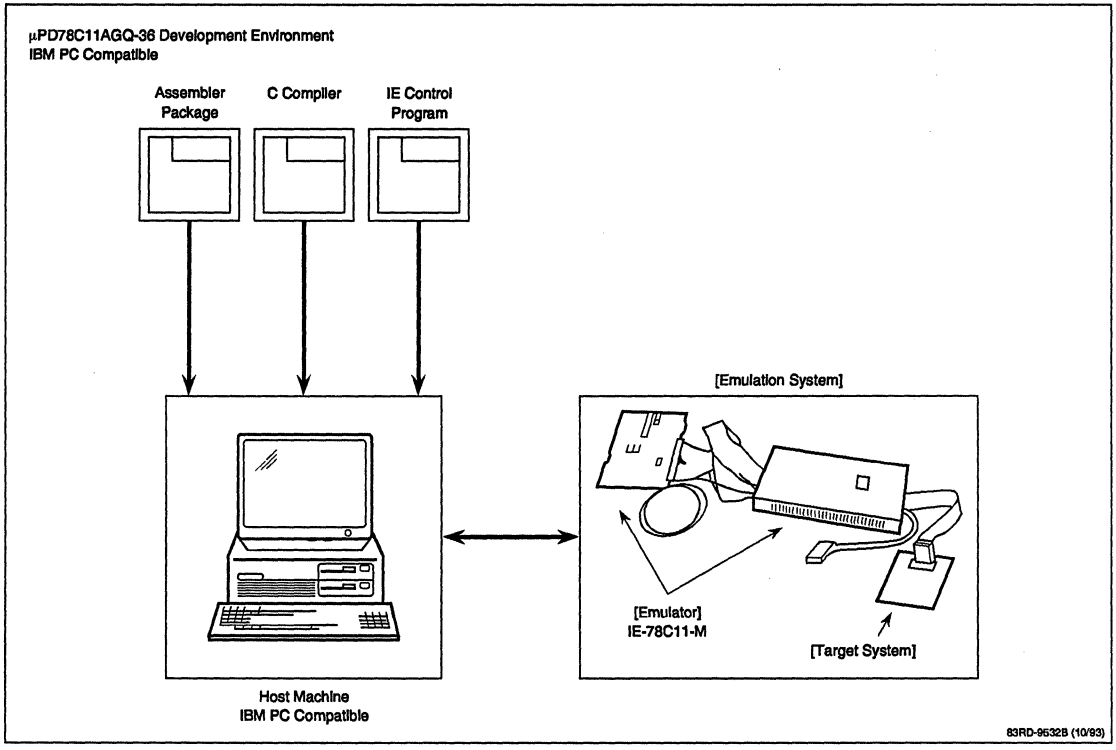
NEC currently supports development environments that use an IBM-PC compatible personal computer.

Development Procedure

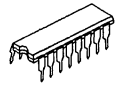
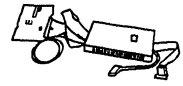
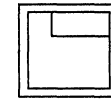


6a

Typical Development Environment (μPD78C00)



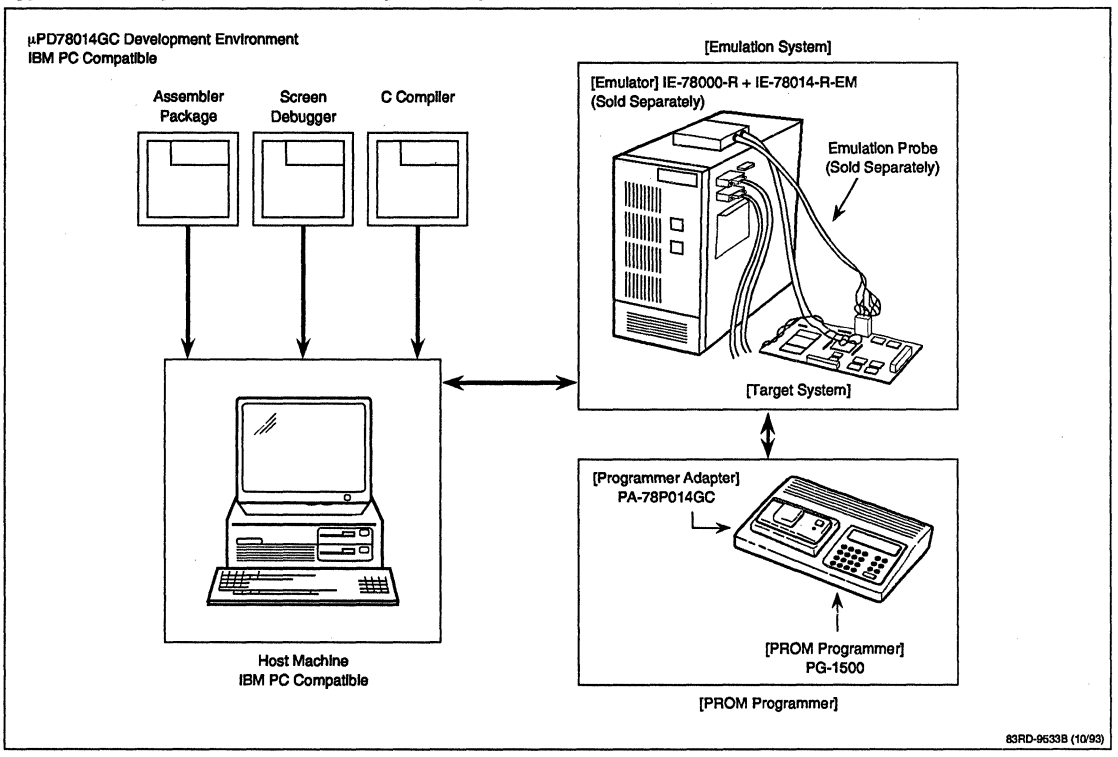
μPD78C00 Product Line Development Tools

Target Device	Package	In-Circuit Emulator	Conversion Socket	Software Packages
				
μPD78C10ACW μPD78C11ACW μPD78C12ACW μPD78C14CW μPD78C17CW μPD78C18CW μPD78CP14CW μPD78CP14DW μPD78CP18CW μPD78CP18DW	64-pin SDIP	IE-78C11-M	EV-9001-64	RA87-D52 (Relocatable assembler) and CCMSD-15DD-87 (C compiler)
μPD78C10AGQ36 μPD78C11AGQ μPD78C12AGQ μPD78C14G36 μPD78C17GQ μPD78C18GQ μPD78CP14GQ μPD78CP14R μPD78CP18GQ	64-pin QUIP		Supplied with IE	
μPD78C10AGF μPD78C11AGF μPD78C12AGF μPD78C14GF μPD78C14G-1B μPD78C17GF μPD78C18GF μPD78CP14GF μPD78CP18GF	64-pin QFP		—	
μPD78CP18KB	64-pin ceramic LCC with window		EV-9200G-64	
μPD78C10AL μPD78C11AL μPD78C12AL μPD78C14L μPD78CP14L	68-pin PLCC		AS-QIP-PCC-D78C1X	

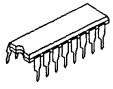
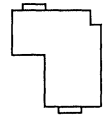
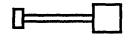
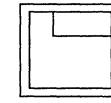
Note: The 64-pin QFP package conversion adapter is manufactured by San Hayato Co., Ltd. The QUIP to PLCC adapter ("AS-QIP-PCC-D78C1X") is manufactured by Emulation Technology.

6a

Typical Development Environment (μPD78K0)



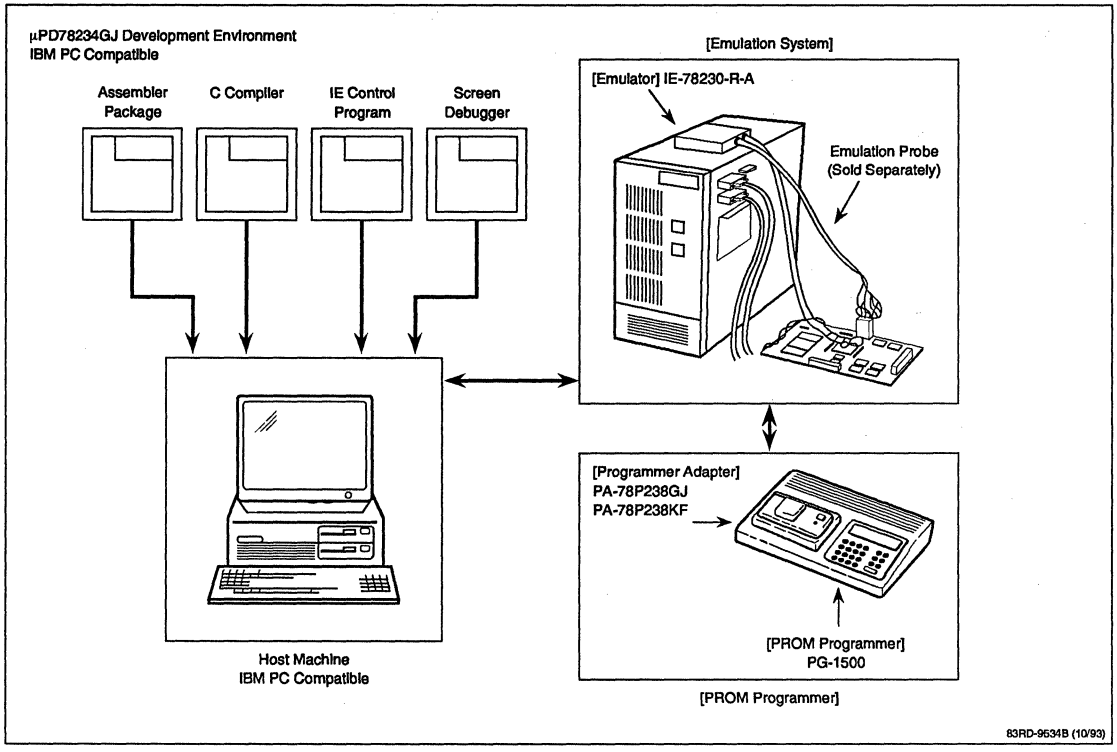
μPD78K0 Product Line Development Tools

Target Device	Package	In-circuit Emulator	Emulation Board	Emulation Probe	Conversion Socket	Software Packages
						
μPD78001CW μPD78002CW	64-pin SDIP	IE-78000-R	IE-78014-R-EM (option)	EP-78240CW-R (option)	—	RA78K0-D52 (Relocatable Assembler) and CC78K0-D52 (C compiler) and CL78K0-D52 (C compiler library) and SD78K0-D52 (Screen debugger) and (GUBED due late 1993)
μPD78001GC μPD78002GC	64-pin QFP			EP-78240GC-R* (option)	EV-9200GC-64	
μPD78011CW μPD78012CW μPD78013CW μPD78014CW μPD78P014CW μPD78P014DW	64-pin SDIP			EP-78240CW-R (option)	—	
μPD78011GC μPD78012GC μPD78013GC μPD78014GC μPD78P014GC	64-pin QFP			EP-78240GC-R* (option)	EV-9200GC-64	
μPD78042GF μPD78043GF μPD78044GF μPD78P044GF μPD78P044KL-S	80-pin QFP		IE-78044-R-EM (option)	EP-78130GF-R* (option)	EV-9200G-80	
μPD78052GC μPD78053GC μPD78054GC μPD78P054GC	80-pin QFP (14 x 14 mm)		IE-78064-R-EM (option)	EP-78230GC-R* (option)	EV-9200GC-80	
μPD78062GC μPD78063GC μPD78064GC μPD78P064GC	100-pin QFP (14 x 14 mm)			EP-78064GC-R* (option)	EP-9500GC-100	
μPD78062GF μPD78063GF μPD78064GF μPD78P064GF μPD78P064KL-T	100-pin QFP (14 x 20 mm)			EP-78064GF-R* (option)	EV-9200GF-100	

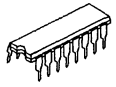
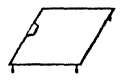
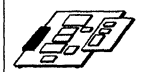
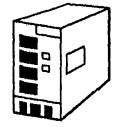
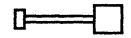
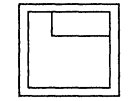
* Includes one required socket adapter shown at right.

6a

Typical Development Environment (μPD78K2)



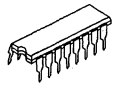
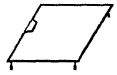
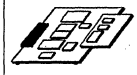
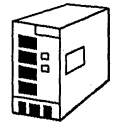
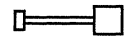
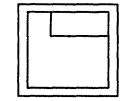
μPD78K2 Product Line Development Tools

Target Device	Package	Design Development Board	Low-End Emulator	In-Circuit Emulator	Emulation Probe	Conversion Socket	Software Packages
							
μPD78212CW μPD78213CW μPD78214CW μPD78P214CW μPD78P214DW μPD78217ACW μPD78218ACW μPD78P218ACW μPD78P218ADW	64-Pin SDIP	DDB-78K2-21X ——	EB-78210-PC or EB-78240-PC ——	IE-78240-R (Old device) or IE-78240-R-A (New device)	EP-78240CW-R (option)		CC78K2-D52 (C compiler) and CL78K2-D52 (C library) and RA78K2-D52 (Relocatable Assembler) and Avocet Tool chain from Avocet Systems, Inc. (207) 236-9055 (800) 448-8500 Consists of: C compiler, assembler, debugger, simulator.
μPD78213GQ μPD78214GQ μPD78P214GQ	64-pin QUIP	DDB-78K2-21X	EB-78210-PC or EB-78240-PC		EP-78240GQ-R (option)	—	
μPD78213L μPD78214L μPD78P214L	68-pin PLCC				EP-78240LP-R (option)		
μPD78212GJ μPD78213GJ μPD78214GJ μPD78P214GJ	74-pin QFP				EP-78240GJ-R* or EP-78210GJ EP-78240LP-R	EV-9200G-74	
μPD78212GC μPD78213GC μPD78214GC μPD78P214GC μPD78217AGC μPD78218AGC	64-pin QFP	——	——		EP-78240GC-R* (option)	EV-9200GC-64	
μPD78220L μPD78224L μPD78P224L	84-pin PLCC	DDB-78K2-22X	EB-78220-PC or EB-78230-PC	IE-78230-R (Old device) or IE-78230-R-A (New device)	EP-78230LQ-R (option)	—	
μPD78220GJ μPD78224GJ μPD78P224GJ	94-pin QFP				EP-78230GJ-R* or EP-78220GJ EP-78230LQ-R	EV-9200G-94	
μPD78233GC μPD78234GC μPD78237GC μPD78238GC μPD78P238GC	80-pin QFP	DDB-78K2-23X	EB-78230-PC		EP-78230GC-R* (option)	EV-9200GC-80	
μPD78233GJ μPD78234GJ μPD78237GJ μPD78238GJ μPD78P238GJ μPD78P238KF	94-pin QFP				EP-78230GJ-R* (option)	EV-9200G-94	

* Includes one required socket adapter shown at right.

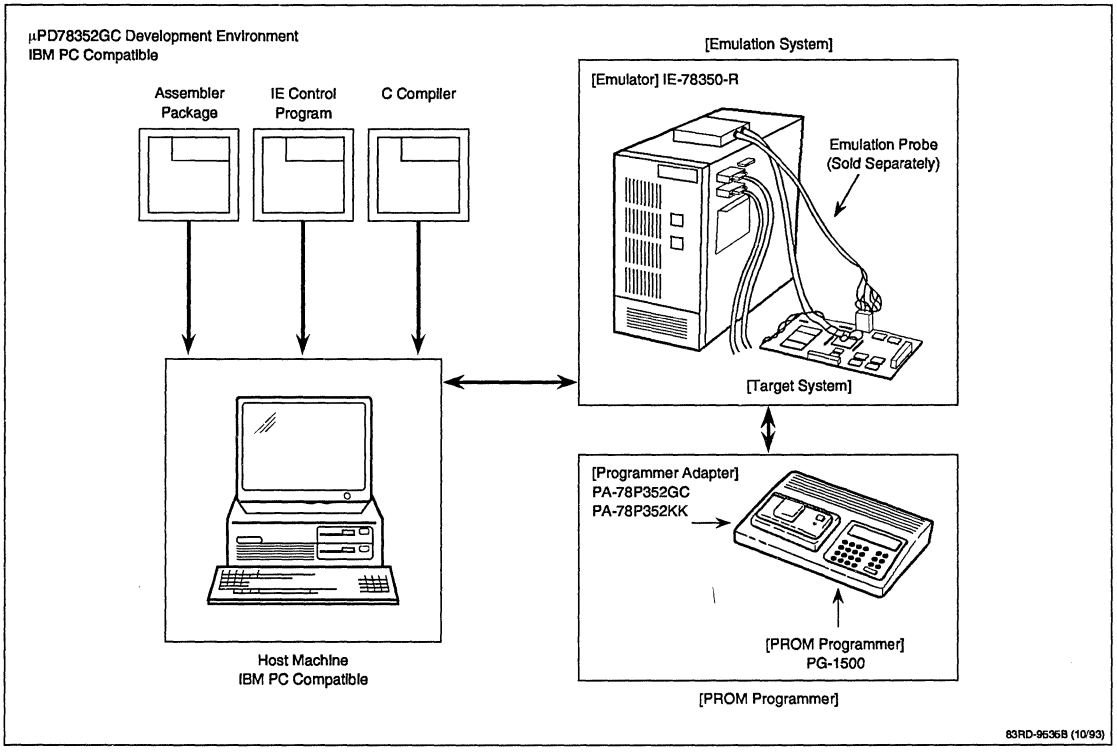
6a

μPD78K2 Product Line Development Tools (cont)

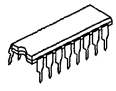
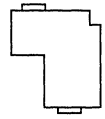
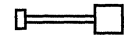
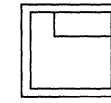
Target Device	Package	Design Development Board	Low-End Emulator	In-Circuit Emulator	Emulation Probe	Conversion Socket	Software Packages
							
μPD78233LQ μPD78234LQ μPD78237LQ μPD78238LQ μPD78P238LQ	84-pin PLCC	DDB-78K2-23X	EB-78230-PC	IE-78230-R (Old device) or IE-78230-R-A (New device)	EP-78230LQ-R (option)	—	CC78K2-D52 (C compiler) and CL78K2-D52 (C library) and RA78K2-D52 (Relocatable Assembler) and Avocet Tool chain from Avocet Systems, Inc. (207) 236-9055 (800) 448-8500 Consists of: C compiler, assembler, debugger, simulator.
μPD78243CW μPD78244CW	64-pin SDIP	—	EB-78240-PC	IE-78240-R (Old device) or IE-78240-R-A (New device)	EP-78240CW-R (option)	—	
μPD78243GC μPD78244GC	64-pin QFP				EP-78240GC-R* (option)	EV-9200GC-64	

* Includes one required socket adapter shown at right.

Typical Development Environment (μPD78K3)

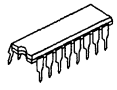
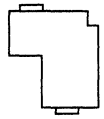
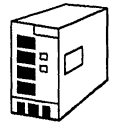
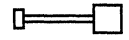
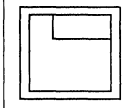


μPD78K3 Product Line Development Tools

Target Device	Package	Evaluation Board	In-Circuit Emulator	Emulation Probe	Conversion Socket	Software Packages
						
μPD78310ACW μPD78312ACW μPD78P312ACW μPD78P312ADW	64-pin SDIP	DDK-78310A	IE-78310A-R	EP-78310GQ Supplied with IE	—	CC78K3-D52 (C compiler) and RA78K3-D52 (Relocatable Assembler) and GUBED (New Graphical Users binary evaluation display) due end 1993
μPD78310AGQ μPD78312AGQ μPD78P312AGQ μPD78P312AR	64-pin QUIP					
μPD78310AGF μPD78312AGF μPD78P312AGF	64-pin QFP			EP-78310GF* (option)	EV-9200G-64	
μPD78310AL μPD78312AL μPD78P312AL	68-pin PLCC			EP-78310L (option)	—	
μPD78320GJ μPD78322GJ μPD78P322GJ μPD78P322KD	74-pin QFP	EB-78320-PC	IE-78327-R	EP-78320GJ-R* (option)	EV-9200G-74	
μPD78320L μPD78322L μPD78P322KC μPD78P322L	68-pin PLCC			EP-78320L-R (option)	—	
μPD78322GF μPD78P322GF μPD78P322K	80-pin QFP			EP-78320GF-R* (option)	EV-9200G-80	
μPD78327GF μPD78328GF μPD78P328GF	64-pin QFP	EB-78327-PC	IE-78327-R	EP-78327GF-R* (option)	EV-9200G-64	
μPD78327CW μPD78328CW μPD78P328DW	64-pin SDIP			EP-78327CW-R (option)	—	
μPD78330LQ μPD78334LQ μPD78P334LQ	84-pin PLCC	EB-78330-PC	IE-78330-R	EP-78330LQ-R (option)	—	
μPD78330GJ μPD78334GJ μPD78P334GJ μPD78P334KE	94-pin QFP			EP-78330GJ-R* (option)	EV-9200G-94	

* Includes one required socket adapter shown at right.

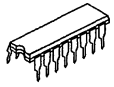
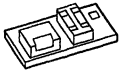
μPD78K3 Product Line Development Tools (cont)

Target Device	Package	Evaluation Board	In-Circuit Emulator	Emulation Probe	Conversion Socket	Software Packages
						
μPD78350GC μPD78352G-22 μPD78P352G-22 μPD78P352KK	64-pin QFP	EB-78350-PC —	IE-78350-R IE-78350-R-EM1	EP-78240GC-R* (option)	EV-9200GC-64	CC78K3-D52 (C compiler) and RA78K3-D52 (Relocatable Assembler) and GUBED (New Graphical Users binary evaluation display) due end 1993
μPD78355GC μPD78356GC μPD78P356GC	100-pin QFP		IE-78350-R IE-78355-R-EM1	EP-78355GC-R* (option)	EV-9500GC-100	
μPD78P356KP-S	120-pin LCC				EV-9500GC-100 EV-9501GC-100	

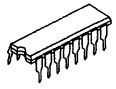
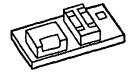
* Includes one required socket adapter shown at right.

6a

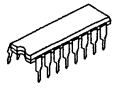
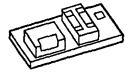
PROM Programmers (K-Series)

Target PROM	Programmer Adapter	PROM Programmer
		
μPD78P014CW μPD78P014DW	PA-78P014CW	PG-1500
μPD78P014GC	PA-78P014GC	
μPD78P044GF	PA-78P044GF	
μPD78P044KL-S	PA-78P044KL-S	
μPD78P054GC	PA-78P054GC	
μPD78P054GK	PA-78P054GK	
μPD78P054KK-S	PA-78P054KK-S	
μPD78P064GC	PA-78P064GC	
μPD78P064GF	PA-78P064GF	
μPD78P064KL-T	PA-78P064KL-T	
μPD78P214CW/DW μPD78P218CW μPD78P218ADW	PA-78P214CW	
μPD78P214GC	PA-78P214GC	
μPD78P214GJ	PA-78P214GJ	
μPD78P214GQ	PA-78P214GQ	
μPD78P214L	PA-78P214L	
μPD78P224GJ	PA-78P224GJ	
μPD78P224L	PA-78P224L	
μPD78P238GC	PA-78P238GC	
μPD78P238GJ	PA-78P238GJ	
μPD78P238KF	PA-78P238KF	
μPD78P238LQ	PA-78P238LQ	
μPD78P312ACW μPD78P312ADW	PA-78P312CW	
μPD78P312AR μPD78P312AGQ	PA-78P312GQ	
μPD78P312AL	PA-78P312L	
μPD78P312AGF	PA-78P312GF	
μPD78P322L	PA-78P322L	
μPD78P322GJ	PA-78P322GJ	
μPD78P322GF	PA-78P322GF	
μPD78P322KC	PA-78P322KC	
μPD78P322KD	PA-78P322KD	
μPD78P324LP	PA-78P324LP	

PROM Programmers (K-Series) (cont)

Target PROM	Programmer Adapter	PROM Programmer
		
μPD78P324KC	PA-78P324KC	PG-1500
μPD78P328GF	PA-78P328GF	
μPD78P334LQ	PA-78P334LQ	
μPD78P334GJ	PA-78P334GJ	
μPD78P334KE	PA-78P334KE	
μPD78P334KF	PA-78P334KF	
μPD78P352GC	PA-78P352GC	
μPD78P352KK	PA-78P352KG	
μPD78P356GC	PA-78P356GC	
μPD78P356KP-S	PA-78P356KP	

PROM Programmers (μPD78C00)

Target PROM	Programmer Adapter	PROM Programmer
		
μPD78CP14CW μPD78CP14DW μPD78CP18CW μPD78CP18DW	PA-78CP14CW	PG-1500
μPD78CP14R μPD78CP14GQ μPD78CP18GQ	PA-78CP14GQ	
μPD78CP14GF μPD78CP18GF	PA-78CP14GF	
μPD78CP14L	PA-78CP14L	
μPD78CP18KB	PA-78CP14KB	

NEC
NEC Electronics Inc.

ROM Code Submission Guide

September 1993

Introduction

This guide provides direction for submitting the data files used by NEC to program semicustomized integrated circuits (a complete list of which appears in table 3).

Where to Send Files

Data files should be sent directly to Micro SBU, Customer Marketing, NEC Electronics, Mountain View, California, 94039.

Acceptable Media

NEC accepts data from the following:

- In programmable ICs such as NEC's μPD27C2000 UV EPROM or μPD75P308 programmable 4-bit microprocessor
- On floppy diskette in MS-DOS® or PC DOS™ formats

— 5-1/4 inch disk (either 360K or 1.2 Mbyte)
— 3-1/2 inch disk (either 360K, 720K or 1.44 Mbyte)

- Via modem over the telephone lines

— At 300 to 14,400 bps
— With 8 data bits, no parity, 1 stop bit
— Using XMODEM, YMODEM or KERMIT protocol

- Via InterNet at shin @ asic.mtv.nec.com
- When opting for modem transmittal, call Customer Marketing for the appropriate engineering contact, dial-up number, and hours of availability.

MS-DOS is a registered trademark of Microsoft.

PC DOS is a trademark of International Business Machines Corp.

Intel is a trademark of Intel.

*This will ensure that NEC has two copies for file comparison at the NEC site, and that you will be able to perform a file comparison.

Minimum Requirements

Any one of the means described may be selected, but NEC requires multiple copies of every file, as well as device-specific information such as chip type, package type, and package lead type.

- A minimum of two copies of the chosen media must be submitted. Three copies are preferred, as this lessens the problems caused when one copy is flawed. Please note that for modem transfers, the file must be transmitted twice* to NEC and transmitted once from NEC back to you.
- Unless submitted by means of programmable ICs, data files must be in Intel™ hexadecimal or extended hexadecimal format.
- Source code or the executable binary of a data file may be submitted but is not required.
- Files taken from or read from previously built ICs, even those produced for you by NEC, will not be accepted.

Taking Precautions

NEC assumes no responsibility for data bits within a target device's programmable area, and it is therefore imperative that you define all bits within the possible range of addresses. For example, if the programmable area is 128K x 8 bits, and your data file defines only the first 64K x 8 bits, there is no way for NEC to know how to program the remaining 64K.

Furthermore, if your programming equipment was left with random data in its memory from a previous operation, valid data would be built into the first half of the programmable area and garbage data into the second half—causing unforeseen and possibly very expensive problems in the final design. Blank space must be defined as either all 0s or all 1s, or as binary NOP, in which case the binary code for an NOP instruction needs to be specified.

What You Can Expect from NEC

After the media or devices have been received by NEC, the copies will be compared to ensure they match (customers whose files don't match will be contacted by Customer Marketing). NEC will then duplicate the media and return the following for verification:

- One copy of the original media
- One copy of NEC's duplicate media
- A hard copy listing of the data files (for target devices with less than 512K, i.e., 64K x 8, of ROM)
- Two floppy diskettes containing those data files transmitted via modem, or InterNet
- A ROM Code Verification Form, which must be signed and returned before any devices can be built

Upon receipt of the signed verification form, NEC will produce ten engineering samples for testing and approval prior to building and shipping the entire order. Data files are kept in archival storage for two years (more than two years is not guaranteed) in one or more of the following formats:

- In one original IC and one NEC duplicate IC, if programmable ICs were initially submitted
- In a hard-copy listing of the hexadecimal file
- In electronic storage using either magnetic or optical media (Write Once Read Many, i.e., WORM, disk)

Please note that IC masks will be stored at NEC's manufacturing facility for only one year after the last order is received.

Peculiar Addressing

Although addresses are usually contiguous, e.g., from 0000_{16} to a maximum, these devices require special consideration.

ROMs with 16-Bit Data Buses. Data submitted for these ROMs in devices with 8-bit data buses should be organized this way.

Table 1. Sequence for 8-Bit Data Bus Devices

Sequence	Addresses	Outputs
Device #1	00000 - 0FFFF	O_0 - O_7
Device #2	10000 - 1FFFF	O_0 - O_7
Device #3	00000 - 0FFFF	O_8 - O_{15}
Device #4	10000 - 1FFFF	O_8 - O_{15}

Data submitted from a 16-bit data bus device would be handled similarly.

Table 2. Sequence for 16-Bit Data Bus Devices

Sequence	Addresses	Outputs
Device #1	00000 - 0FFFF	O_0 - O_7
	10000 - 1FFFF	O_8 - O_{15}

In both cases, segments corresponding to the lower data outputs are submitted first, a pattern that should be followed for larger devices as well. Please be aware that NEC has no way of identifying the sequential order of individual segments if they're submitted in incorrect order.

Table 3. Applicable Device Types

Part Number(s)	Organization	Last Address
μPD78C11A	4,096 x 8 bits	$0FFF_{16}$
C12A	8,192 x 8 bits	$1FFF_{16}$
C14	16,384 x 8 bits	$3FFF_{16}$
C14A	16,384 x 8 bits	$3FFF_{16}$
C18	32,768 x 8 bits	$7FFF_{16}$
μPD78001	8,192 x 8 bits	$1FFF_{16}$
002	16,384 x 8 bits	$3FFF_{16}$
μPD78001Y	8,192 x 8 bits	$1FFF_{16}$
002Y	16,384 x 8 bits	$3FFF_{16}$
μPD78011	8,192 x 8 bits	$1FFF_{16}$
012	16,384 x 8 bits	$3FFF_{16}$
013	24,576 x 8 bits	$5FFF_{16}$
014	32,768 x 8 bits	$7FFF_{16}$
μPD78011Y	8,192 x 8 bits	$1FFF_{16}$
012Y	16,384 x 8 bits	$3FFF_{16}$
013Y	24,576 x 8 bits	$5FFF_{16}$
014Y	32,768 x 8 bits	$7FFF_{16}$
μPD78042	16,384 x 8 bits	$3FFF_{16}$
043	24,576 x 8 bits	$5FFF_{16}$
044	32,768 x 8 bits	$7FFF_{16}$
μPD78052	16,384 x 8 bits	$3FFF_{16}$
053	24,576 x 8 bits	$5FFF_{16}$
054	32,768 x 8 bits	$7FFF_{16}$
μPD78062	16,384 x 8 bits	$3FFF_{16}$
063	24,576 x 8 bits	$5FFF_{16}$
064	32,768 x 8 bits	$7FFF_{16}$
μPD78212	8,192 x 8 bits	$1FFF_{16}$
214	16,384 x 8 bits	$3FFF_{16}$
μPD78218A	32,768 x 8 bits	$7FFF_{16}$
μPD78224	16,384 x 8 bits	$3FFF_{16}$
μPD78234	16,384 x 8 bits	$3FFF_{16}$
238	32,768 x 8 bits	$7FFF_{16}$
μPD78312A	8,192 x 8 bits	$1FFF_{16}$
μPD78322	16,384 x 8 bits	$3FFF_{16}$
μPD78352A	32,768 x 8 bits	$7FFF_{16}$
μPD78356	49,152 x 8 bits	$BFFF_{16}$

Using The Intel Hexadecimal Format

Intel's hexadecimal format allows addressing of up to 512 kbits of data, or 64K x 8 bits (0000_{16} through $FFFF_{16}$). Data records larger than 64K x 8 must be expressed in multiple segments, with each individually addressed segment equal to or smaller than 64K x 8.

Each byte of data must be expressed as a printable ASCII character, and each line must contain these elements:

- A colon to begin each line
- A two-character data word count for the line
- A four-character address of the first word of data
- A two-character record type identifier, e.g., 00, 01, 02 for data, end of file, segment address
- The data words
- A two-character line checksum at the end of each line

For example, a line showing the "End of File" record would be formatted as :00000001FF, while a typical data line would be constructed this way:

:WWAAAATTDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDCC

Table 4. Description of Elements

Code	Description
:	Beginning of line
WW	Data word count
AAAA	Address of the first data word
TT	Record type
D...D	Data words
CC	Checksum

Table 5. Description of Record Type

TT	Description
00	Data follows
01	End of file
02	Begin new 64K x 8 data segment

Table 6 shows an address shift from one 32K x 8 segment to the next segment in hexadecimal format.

Table 6. Sample Hexadecimal Addressing

Coded Segment	Instruction
:100000007F7F7F7F7F7E7D7B797878787A7D2C	Begin first segment
:10FFF000FFFFFFFFFFFFFFFFFFFFFFFFFFFF11	End first segment
:020000021000EC	Second segment record
:100000007F7F7F7F7E7E7D7B7A797877797B7F2C	Begin second segment
:10FFF000FFFFFFFFFFFFFFFFFFFFFFFFFFFF11	End second segment
:00000001FF	End of file record

NEC
NEC Electronics Inc.

PG-1500
PROM Programmer

September 1993

Description

The PG-1500 series is a stand-alone PROM programmer for programming 256-kilobit to 1-megabit PROMs and PROM/OTP devices for NEC's 4/8/16-bit single-chip microcontrollers and digital signal processors. The system consists of the PG-1500 base programmer, two interchangeable programmer adapter modules, and a variety of programmer adapters to support the individual devices and package types. The PG-1500 can be controlled directly from either a remote terminal or host computer via an RS-232-C serial port or directly from the on-board keypad in stand-alone mode.

Features

- Interchangeable modules for programming:
  - 256-kilobit to 1-megabit PROMs
  - NEC μPD75xx and μPD75xxx 4-bit microcontrollers
  - NEC K-series® microcontrollers
  - NEC V-series™ microcontrollers
  - NEC μPD77xxx digital signal processors
- 512K-bytes data RAM
- Silicon signature read function
- PROM insertion error detection circuitry
- Memory edit function to change/confirm PG-1500 buffer
- Address/data/message display LCD
- RS-232-C serial interface
- Centronics compatible parallel interface
- Power-on diagnostics
- Supports three data transfer formats
  - Intel Extended Hex (Note 1)
  - Extended Tektronix Hex (Note 2)
  - Motorola S (Note 3)
- Two modes of operation
  - Remote controlled
  - Stand-alone
- Host Controller Program for IBM PC® Series

IBM PC is a registered trademark of International Business Machines Corporation.
V-Series and V40 are trademarks of NEC Electronics, Inc.
K-series is a registered trademark of NEC Electronics, Inc.

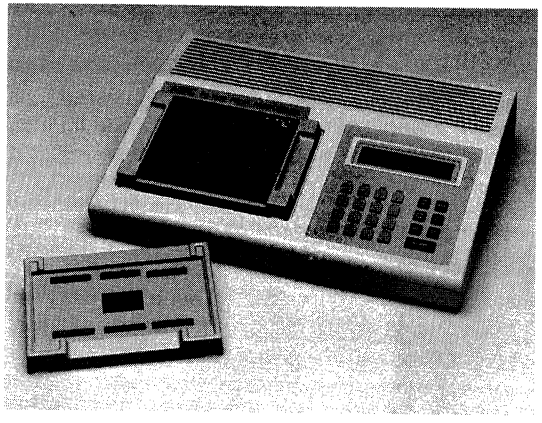
Notes:

(1) Developed by Intel Corporation.

(2) Developed by Tektronix Corporation.

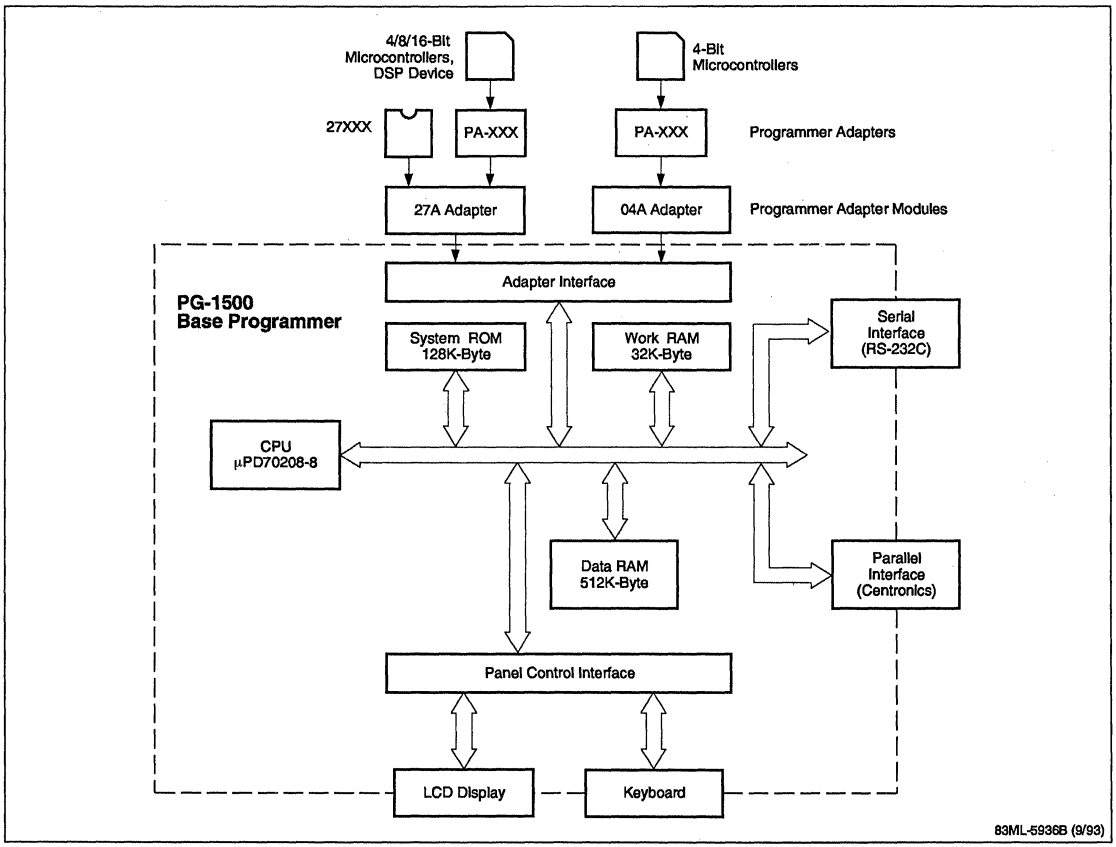
(3) Developed by Motorola Incorporated

PG-1500 Series



6c

Figure 1. PG-1500 System Block Diagram



Architecture

The PG-1500 base unit contains an NEC μPD70208 (V40™) microprocessor with 128K bytes of monitor ROM, 32K bytes of working RAM, 512K bytes of data memory, an RS-232-C serial port, a Centronics compatible parallel interface, an LCD display, and a 23-key keypad. Figure 1 shows a block diagram of the PG-1500.

The PG-1500 has two interchangeable programmer adapter modules. These adapter modules plug directly into the top of the PG-1500 and can accept a wide variety of programmer socket adapters to support NEC's devices. Refer to the Development Tool Selection Guide for a list of all available adapters.

Operation

The PG-1500 operates in stand-alone mode from the on-board keypad, or in remote-control mode from an external terminal or a host computer via an RS-232-C serial port.

Host Controller Program

The PC-1500 can be controlled from an IBM PC series host computer using the PG-1500 controller program. The controller program has three modes of operation: control mode, auto mode, and terminal mode.

Silicon Signature

A silicon signature is stored in all NEC devices and contains information on the device type, start and stop addresses, and programming voltages. The PG-1500 can read the silicon signature of the particular device being programmed either manually or automatically, or the device code can be entered manually.

Basic Specifications

- Power requirements:
  — 90 to 250 VAC, 50 to 60 Hz
- Environment conditions:
  — Operating temperature range: 10 to 35°C
  — Operating humidity range: 20 to 80% relative humidity
- RS-232-C serial port:
  — Baud rates: 1200, 2400, 4800, 9600, 19200
  — Parity: none, even, odd
  — X-ON/X-OFF: on, off
  — Bit configuration: 7, 8
  — Stop bits: 1, 2

Equipment Supplied

The PG-1500 package includes the following:

- PG-1500 PROM programmer base unit
- 027A socket board for 27xxx PROMS and μPD27C256A-like devices
- 04A interface board for NEC μPD75xx/μPD75xxx microcontrollers
- PG-1500 controller program disk for IBM PC
- Power cord
- Power ground plug adapter
- Spare fuses (2)

Documentation

For further information on the operation of the PG-1500, NEC provides the following documentation:

- PG-1500 PROM Programmer User's Manual
- PG-1500 Controller User's Manual (IBM PC-based)

NEC
NEC Electronics Inc.

IE-78C11-M
In-Circuit Emulator
for the μPD78C00 Product Line

September 1993

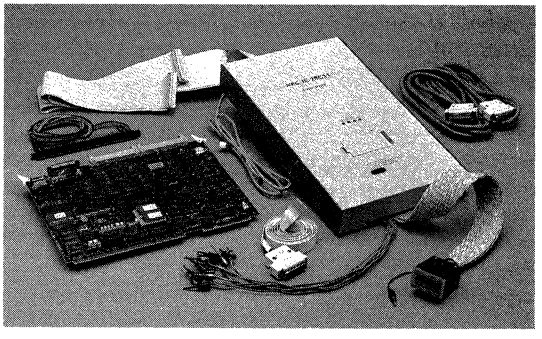
Description

The IE-78C11-M is an in-circuit emulator providing both hardware emulation and software debugging capabilities for the NEC μPD78C00 product line of 8-bit, single-chip microcontrollers. Real-time and single-step emulation, coupled with sophisticated memory mapping features, breakpoints, and trace capabilities, create a powerful debugging environment. A line assembler and disassembler, full register and memory control, and complete upload/download capabilities simplify the task of debugging hardware and software. The IE-78C11-M is designed to operate as a stand-alone, in-circuit emulator controlled from either a user terminal or a host computer system.

Features

- Real-time and single-step emulation
- User-specified breakpoints:
  Logical OR of up to four sets of break conditions
  - Opcode fetch count
  - External sense clips condition
  - Emulation time
  - Logical AND of addresses, data values, CPU controls, and number of loops
- Sophisticated trace capabilities
  - — Instruction or machine cycle display
  - — 1,024 trace frames
  - — Address, control, data, and port trace
- Powerful memory mapping: 64K bytes of RAM mappable in 256-byte blocks
- Line assembler/disassembler
- Operating state LED indicators
- CMOS latch-up warning and protection
- Eight external sense probes
- Self-diagnostic command
- Stand-alone configuration
  - — User terminal controlled
  - — Host computer system controlled
- IE78C11 controller program for IBM PC®, PC/XT®, PC/AT®, or compatibles
  - — Symbolic debugging
  - — Autoexecution of commands
  - — On-line help facility
  - — Debug session logging

IE-78C11-M In-Circuit Emulator



Ordering Information

Part Number	Description
IE-78C11-M	In-circuit emulator for μPD78C1x/C1xA series
EP-7811HGQ	Emulator probe for 64-pin QUIP package (shipped with IE-78C11)
EV-9001-64	Optional emulator probe adapter for 64-pin shrink DIP package (used with EP-7811HGQ)
EV-9200G-64	64-pin LCC socket used with the μPD78CP18KB
AS-QIP-PCC-D781X	Optional QUIP to PLCC adapter (used with EP-7811HGQ)

NEC
NEC Electronics Inc.

CC87
Micro-Series
C Compiler Package for the
μPD78C00 Product Line

September 1993

Description

The CC87 Micro-Series™ C compiler package for the NEC μPD78C00 product line of microcontrollers consists of an ANSI C cross-compiler, relocatable macro assembler, linker, and librarian. Developed by IAR systems in Sweden for NEC, the Micro-Series C compiler package runs on an MS-DOS® system with a free-standing system as target (embedded system).

Ordering Information

Part Number	System	Description
CCMSD-I5DD-87	MS-DOS	5-1/4-inch double-density floppy diskette

Features

- ANSI standard C
  - Const, volatile, signed, void, enum keywords
  - Function prototyping
  - Hex string constants
  - Structure and union assignments
- Optimization for code size or execution speed
- Extended functions for μPD78C00 product line code generation
  - Saddr area usage for variables
  - Non-initialized variable declaration
  - Interrupt vector table generation
  - Function execution with interrupts disabled
  - Input byte/word from special register
  - Output byte/word to special register
  - Modify special register with a byte constant
  - Disable/enable interrupts
  - Halt CPU
  - Check/reset interrupt flags
- IEEE 32-bit floating-point data representation
- UNIX LINT functions (legal C code verification) integrated into the compiler
- Interface checking between modules performed by the linker XLINK
- ROMable object file creation
- Generation of list and full cross-reference files
- Built-in help facility
- Extensive error reporting

C Library Functions

The CC87 Micro-Series C compiler package includes most of the important C library functions that apply to PROM-based embedded systems. All library functions reside in the supplied library files. Header files that declare the set of library functions are also included.

The following library functions are available:

CHARACTER HANDLING < ctype.h>
isalnum isalpha iscntrl isdigit islower
isprint ispunct
isspace isupper isxdigit tolower toupper

VARIABLE ARGUMENTS < stdarg.h>
va_arg va_end va_start

NON-LOGICAL JUMPS < setjmp.h>
longjmp setjmp

FORMATTED INPUT/OUTPUT < stdio.h>
getchar gets printf putchar scanf, sscanf, sprintf

GENERAL UTILITIES < stdlib.h>
atof atoi atol calloc exit free malloc ralloc

STRING HANDLING < string.h>
strcat strcmp strcpy strlen strncat strncmp strncpy

MATHEMATICS < math.h>
atan atan2 cos exp log log10 modf pow sin sqrt tan

LOW-LEVEL ROUTINES < icclbutl.h>
_formatted_write

Memory Allocation

The two memory allocation modes, static and reentrant, differ only in allocation of auto variables. In the reentrant mode, all local auto variables are allocated and deallocated dynamically; the auto variables reside on the stack, which is necessary if recursive or reentrant functions are needed. This option sometimes generates more code and slower code than the static mode. In the static mode, all function level variables are put into static memory, with the exception of function arguments, which are always placed on the stack.

C Cross-Compiler (ICC7800)

The C cross-compiler, which is the ICC7800 program, converts standard C source code into relocatable object modules in the IAR systems' proprietary universal binary relocatable object format (UBROF). This format is used for all relocatable object files in the Micro-

Series development system, whether generated by an assembler or compiler. During compilation, an optional optimizer can be invoked to optimize the object code for size or execution speed.

In addition, CC87 supports extended functions for μPD78C00 product line code generation. These extended functions allow the C compiler to take advantage of many of the powerful features in the μPD78C00 product line microcontrollers to decrease object code size and improve program execution speed.

Relocatable Macro Assembler (A7800)

The relocatable macro assembler (A7800) translates symbolic source code for the NEC μPD78C00 product line of microcontrollers into relocatable object modules in the IAR systems proprietary UBROF format.

Features. The relocatable macro assembler features are as follows:

- Absolute or relocatable address object code output
- Directives
  - — List formatting
  - — Conditional assembly, separate assembly
  - — Memory allocation
  - — Macro definition and value assignments to symbol directives
- Generation of list files
- Generation of cross-reference and symbol tables
- Ability to include files in another source

Directives. Assembler directives give instructions to the program but are not translated into machine code during assembly. Basic directives include those for storage definition and memory allocation (DB, DD, DW, DS); symbol control and usability (PUBLIC, EXTERN, LOCSYM); and value assignments to symbols (SET, EQU, =, DEFINE).

Program control directives include those for module definition (NAME, MODULE, ENDMOD); segment definition and control (ASEG, RSEG, STACK, COMMON, ORG); conditional assembly (IF, ELSE, ENDIF); macro processing (MACRO, ENDMAC); and listings control (LSTOUT, LSTCND, LSTCOD, LSTEXP, LSTMAC, LSTWID, LSTFOR, LSTPAG, PAGSIZ, PAGE, TITL, STITL, PTITL, PSTITL, LSTXRF).

Linker (XLINK)

The universal linker, XLINK, combines relocatable object modules and absolute load modules and produces one absolute load module. The controls for XLINK may be specified either on the command line or in a parameter file. In addition to being able to generate several types of absolute load module formats, it is also possible to generate cross-reference lists with an index list; define segment allocation; force load and conditional load of files; bank segments; and define a symbol on a command line. The absolute load module can contain symbol information as well as absolute object code.

Librarian (XLIB)

The XLIB librarian creates and maintains files containing relocatable object modules. With XLIB, the user can merge object files from different assemblies/compilations in order to create libraries; delete individual modules; change the order of modules and check the CRC in a module; and rename modules, segments, externals or entries. In addition, XLIB can change the properties of a module to be conditionally or unconditionally loaded. Use of XLIB reduces the number of files that need to be linked together by allowing several modules to be kept in a single file, providing an easy way to link frequently used modules into programs.

License Agreement

CC87 Micro-Series C Compiler package is sold under terms of a license agreement included with the compiler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license.

Documentation

For further information on source program format, compiler operation, assembler operation, linker, librarian, and converter programs, and actual program examples, refer to the following manual supplied with the CC87 package. Additional copies may be obtained from NEC Electronics Inc.

- CC87 μPD7800 Series C Compiler, User's Manual

NEC
NEC Electronics Inc.

RA87
Relocatable Assembler Package for the μPD78C00 Product Line

September 1993

Description

The RA87 relocatable assembler package converts symbolic source code for the μPD78C00 product line of microcontrollers into executable absolute address object code. The RA87 package consists of six separate programs: assembler (RA87), linker (LK87), hexadecimal format object converter (OC87), librarian (LB87), list converter (LCNV87), and macroprocessor (MP).

RA87 translates a symbolic source module into a relocatable object module. The assembler verifies that each instruction assembled is valid for the target microcontroller specified at assembly time.

LK87 combines relocatable object modules and absolute load modules and converts them into an absolute load module. OC87 converts an absolute object module or an absolute load module into an ASCII hexadecimal format object file.

LB87 allows commonly used relocatable object modules to be stored in one file and linked into multiple programs, greatly increasing programming efficiency. When a library file is included in the input to the linker, the linker extracts only those modules required to resolve external references from the file and relocates and links them into the absolute load module.

LCNV87 allows relocatable list files to be converted into absolute list files. MP expands macros contained in a source program prior to assembling.

Features

- Absolute address object code output
- Generic jump capability
- User-selectable and directable output files
- Extensive error reporting
- Macro capabilities
- Runs under MS-DOS® operating system

MS-DOS is a registered trademark of Microsoft Corporation.

Ordering Information

Part Number	System	Description
RA87-D52	MS-DOS	5-1/4-inch, double-density floppy diskette

Program Syntax

An RA87 source module consists of a series of code, byte-oriented data, or bit-oriented data segments. Each segment consists of statements composed of up to four fields: symbol, mnemonic, operand, comment.

The symbol field may contain a label whose value is the instruction or data address or a name that represents an instruction address, data address, or constant. The mnemonic field may contain an instruction or an assembler directive. The operand field contains the data or expression for the specified instruction or directive. The comment field allows explanatory comments to be added to a program.

Character constants are translated into 7-bit ASCII codes. Numeric constants may be specified as binary, octal, decimal, or hexadecimal. Arithmetic expressions may include the operators +, -, *, /, NOT, AND, OR, XOR, EQ, NE, GT, GE, LT, LE, SHR, SHL, HIGH byte, LOW byte, MOD, and the - sign.

Assembler Directives

Assembler directives give instructions to the assembler but are not translated into machine code during assembly. Basic assembler directives include storage definition (DB, DW, DS, DBIT); symbol definition (EQU, SET, CODE, DATA, BIT); and program boundary definition (ORG, END). Program linkage directives are provided to NAME the module and to declare symbols as PUBLIC or external (EXTRN).

Segment definition directives define whether a segment is a code segment (CSEG), allocated to ROM; a data segment (DSEG) or a bit segment (BSEG), allocated to RAM; or a working register segment (VREG). The address boundary conditions for each segment directive are specified in its operand. These include UNIT, PAGE, INPAGE, FIXEDAREA, BYTE, CALLTABLE, AT, BITADDRESSABLE. The combination types of PUBLIC, COMMON, and COMPLETE specified in the operand define how to link segments with the same name and segment definition.

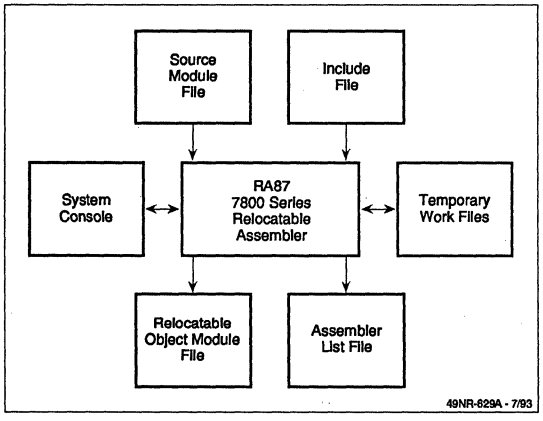
The μPD78C00 product line instruction set contains three jump instructions with varying legal address ranges. To avoid calculating which jump instruction to use, the programmer can substitute the generic jump (GJMP) directive for any relative jump (JR), any extended relative jump (JRE), or any long jump (JMP) instruction in the source program. During assembly, a suitable jump instruction is chosen for each GJMP directive.

Assembler Controls

The RA87 assembler (figure 1) has two types of controls. The primary controls, which are specified in the assembler command line, a parameter file, or at the beginning of the source module, are as follows:

- Target microcomputer specification
- Output file selection and destination
- Listing format controls
- Date specification

Figure 1. Relocatable Assembler Functional Diagram



The general controls, specified in the assembler command line, a parameter file, or at any place in the source program, are as follows:

- Generation/suppression of listing
- Listing titles
- Inclusion of other source files (in source program only)
- Page eject (in source program only)

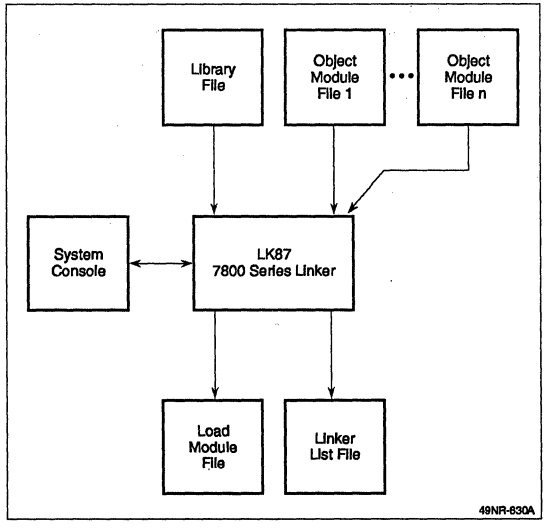
The listing file may contain the complete assembly listing or only lines with errors, and a symbol table or a cross-reference table. The symbol table shows all defined symbols in alphabetical order, their types, attributes, and the values initially assigned to them. The cross-reference table contains all defined symbols and the numbers of all statements that refer to them.

The object file contains the relocatable object module. The format of this module is an NEC proprietary relocatable object module format. This object file may also contain local symbol information for the symbolic debugger.

Linker

The LK87 linker (figure 2) combines several relocatable object modules or absolute load modules, resolving PUBLIC/EXTRN references between modules, to create an absolute load module. This load module contains both absolute object code and symbol information. The linker will also search library files for required modules to resolve external references. The linker controls for LK87 can be specified in either the command line or in a parameter file. The programmer can specify the date, module name, stack size and starting address, ROM/RAM segment allocation, starting address and order for code/data/bit relocatable segments, and the page address for the working register group. The programmer may also specify that a list file containing a link map, a local symbol table, or a public symbol table be created.

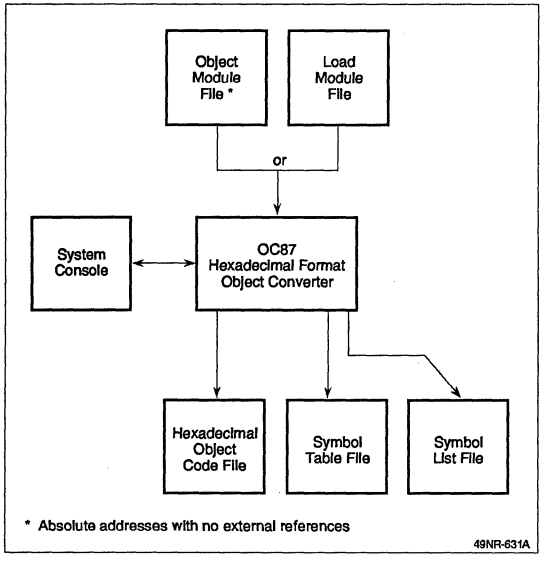
Figure 2. Linker Functional Diagram



Hexadecimal Format Object Converter

The OC87 object converter (figure 3) outputs the object code file in ASCII hexadecimal format, which can be downloaded to a PROM programmer or hardware debugger. The programmer can specify whether or not to generate a symbol file for a hardware debugger.

Figure 3. Hexadecimal Format Object Code Converter Functional Diagram



Librarian

The LB87 librarian (figure 4) creates and maintains library files containing relocatable object modules. This reduces the number of files to be linked together by allowing several modules to be stored in a single file. This provides an easy way to link frequently used modules into programs. Modules can be added to or deleted from, or the contents of the library file can be listed.

List Converter

Normally, listing files produced by a relocatable assembler do not show the final absolute address for instructions because their location is not decided until link time. The address shown in the listing is only the offset from the start of the code or data segment.

The LCNV87 list converter (figure 5) uses the assembly list and object module files from the assembler and the load module file from the linker to create an absolute address assembly listing. This absolute listing shows the addresses of instructions as their final absolute address in memory and is useful in debugging or program documentation. The programmer can specify the load module (-L), assembly list (-A), and output assembly (-O) file names.

Figure 4. Librarian Functional Diagram

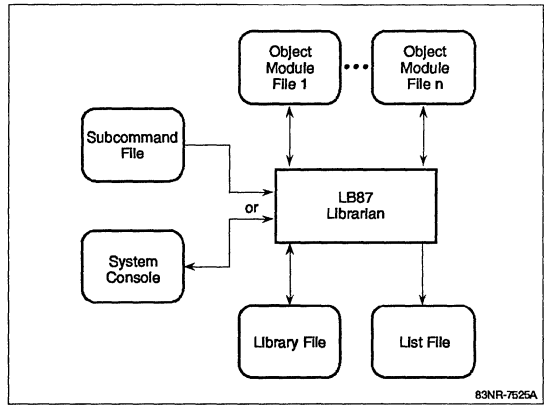
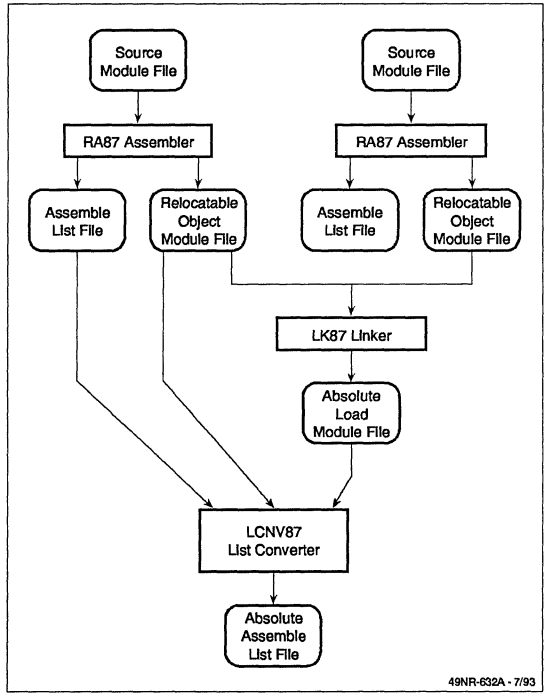


Figure 5. List Converter Functional Diagram



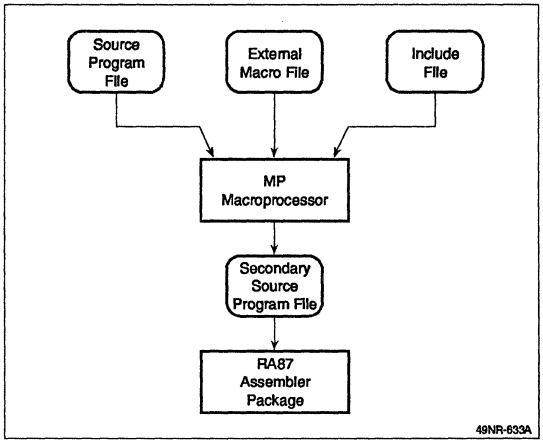
6f

Macroprocessor

The MP macroprocessor (figure 6) interprets the macros described in a source program and expands them to create another source program, which can then be input to the assembler. It has the following three main functions:

- Expands macros by defining and referencing them
- Reads and expands include files
- Selects an assembler source based on a conditional macro instruction

Figure 6. Macroprocessor Functional Diagram



Emulator Controller Program

Absolute object files produced by the RA87 relocatable assembler package can be debugged using the appropriate NEC stand-alone in-circuit emulator. NEC emulator controller programs allow communication with the emulator through an RS-232C serial line. An emulator controller program can run on the IBM PC®, PC/XT®, and PC/AT® under MS-DOS and is provided with the in-circuit emulator at no extra charge.

These emulator controller programs provide the following features:

- Uploading and downloading of object and symbol files
- Symbolic debugging
- Complete emulator control from host console
- On-line help facilities
- Macro command file capabilities
- Host system directory and file display
- Disk storage of debug session

License Agreement

RA87 is sold under terms of a license agreement included with the assembler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users for one year.

Documentation

For further information on source program formats, assembler operation, and actual program examples, refer to the following manuals supplied with the RA87 package. Additional copies may be obtained from NEC Electronics Inc.

- RA87 μPD78C00 Product Line Relocatable Assembler Package, User's Manual
- MP Macroprocessor, User's Manual

IBM PC, PC/XT, and PC/AT are registered trademarks of International Business Machines Corporation.

NEC
NEC Electronics Inc.

IE-78000-R
In-Circuit Emulator
for the μPD78K0 Product Line

September 1993

Description

The IE-78000-R is an in-circuit emulator providing both hardware emulation and software debugging capabilities for the NEC μPD78K0 product line of single-chip microcontrollers. The IE-78000-R, combined with a SD78K0 screen debugger and IE-78000-R-BK break board, create a powerful debug environment.

The SD78K0 is screen debugger software included in an IE-78000-R package that will allow the user to debug a program through a host machine (IBM PC/IBM compatible) in a window-oriented environment. This screen debugger software is used by connecting it to the IE-78000-R.

Different target devices can be emulated by connecting a separately purchased emulation board, IE-780xx-R-EM to the IE-78000-R . The product selection guide shows the list of available emulation boards, emulation probes, socket adapters, and corresponding target devices.

Features

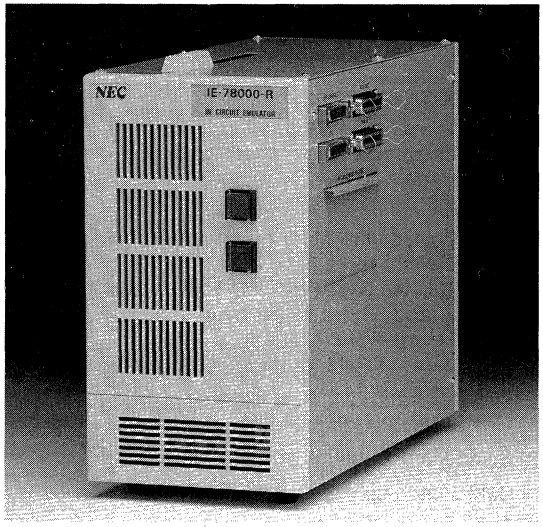
- 10-MHz maximum operating frequency
- Real-time and non-real-time emulation
- Sophisticated break events
  - Logical OR of up to four sets of events
  - Logical AND of addresses, data values, CPU controls, and loop count
  - Executed instruction count
  - Parallel or sequential fetch addresses
  - External sense clip condition
- Sophisticated trace capabilities
  - Trace program fetch or data access
  - 2K x 49-bit trace buffer
  - Address, control, data, and external signal trace
  - Instruction or frame display
  - Trace search capability
  - Specify trigger point at beginning, middle, or end of trace buffer
- Supports debug activities during real-time emulation
  - Displays trace buffer
  - Modifies trace conditions
  - Modifies trace event conditions
  - Restarts trace
- Powerful memory mapping
  - 64K bytes of RAM for off-chip RAM/ROM emulation
  - 200K bytes DRAM memory for symbols (192K byte) and programs (8K byte)
  - 56K bytes ROM
  - 4K bytes of RAM for internal RAM emulation
  - 14K trace RAM
- Line assembler/disassembler
- Symbolic debugging, 700/900 symbols
- CMOS latch-up warning and protection
- Eight external sense clips on emulator probe
- Two serial and two parallel interface channels
- Memory bank selector to select ROM, DRAM or trace RAM
- Host control program for IBM PC, PC/XT, PC/AT, or compatible

Equipment Supplied

The IE-78000-R package includes the following:

- IE-78000-R emulator chassis
- IE-78000-R-BK break board
- Control/trace board
- SD78K0-D52 screen debugger software
- RS-232-C interface cable

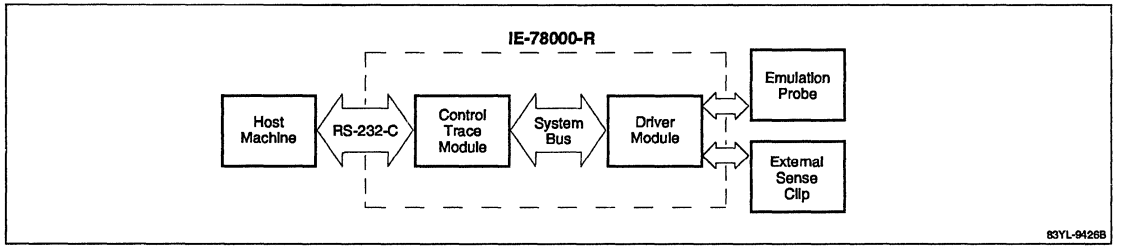
IE-78000-R In-Circuit Emulator



Ordering Information (Also, see selection guide.)

Part Number	Description
IE-78000-R	In-circuit emulator for 78K0 product line
IE-78014-R-EM	Emulation board for 78014 family (optional)
IE-78044-R-EM	Emulation board for 78044 family (optional)
IE-78064-R-EM	Emulation board for 78064 family (optional)
EP-78240GC-R	(64-pin plastic QFP) emulator probe used with IE-78014-R-EM board
EP-78240CW-R	(64-pin shrink DIP) emulator probe used with IE-78014-R-EM board
EP-78130GF-R	(80-pin plastic QFP) emulator probe used with IE-78044-R-EM board (optional)
EP-78064GF-R	(100-pin QFP (14x20)) emulator probe for 78064GF family, used with IE-78064-R-EM board (optional)
EP-78064GC-R	(100-pin QFP (14x14)) emulator probe for 78064GC family used with IE-78064-R-EM board (optional)
EP-78054GK-R	(80-pin plastic TQFP (12x12)) emulator probe for 78054GK family, used with IE-78064-R-EM board (optional)
EP-78230GC-R	(80-pin QFP (14x14)) emulator probe for 78054GC family, used with IE-78064-R-EM board (optional)
EV-9200G-80	Five socket adapters; converts 80-pin LCC probe tip to 80-pin QFP (14x20) device footprint (optional)
EV-9200GC-64	Five socket adapters; converts 64-pin LCC probe tip to 64-pin QFP device footprint (optional)
EV-9200GC-80	Five socket adapters; converts 80-pin LCC probe tip to 80-pin plastic QFP (14x14) device footprint (optional)
EV-9200GF-100	Five socket adapters; converts 100-pin LCC probe tip to 100-pin QFP (14x20) device footprint (optional).
EV-9500GC-100	One socket adapter, 100-pin PGA to 100-pin QFP (optional)
CC78K0-D52	C compiler package for 78K0 product line (optional)
RA78K0-D52	Relocatable assembler for 78K0 product line (optional)
SD78K0-D52	Screen debugger software included in IE-78000-R package

IE-78000-R Block Diagram



6g

NEC
NEC Electronics Inc.

CC78K0
C Compiler for the μPD78K0 Product Line

September 1993

Description

The CC78K0 C compiler is an ANSI standard C cross-compiler for the NEC μPD78K0 product line of microcontrollers. The CC78K0 (figure 1) converts ANSI standard C source code into NEC format object module or assembly language source files. During compilation, an optional optimizer can be invoked to optimize the object code for size and/or execution speed.

In addition, CC78K0 supports extended functions for μPD78K0 code generation. These extended functions allow the C compiler to take advantage of many powerful features in the μPD78K0 microcontrollers to decrease object code size and improve program execution speed.

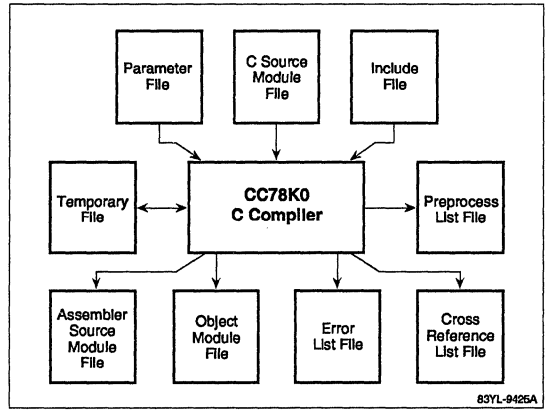
The relocatable object file produced by the CC78K0 can be converted into an absolute object file by the linker program and object converter program contained in the RA78K0 relocatable assembler package. The resulting ASCII hexadecimal format absolute object file then can be debugged using an NEC in-circuit emulator or evaluation board.

Features

- ANSI standard C compiler
- Extended functions for optimized μPD78K0 code generation
- Various optimization options for code size and/or execution speed
- Legal C code verification
- Outputs NEC format object module or assembly source file
- Run-time error checking
- Outputs debug information
- ROMable object file creation
- User selectable and directable output files, list, and full cross-reference files
- Extensive error reporting
- Built-in help facility
- Runs under MS-DOS® operating system

MS-DOS is a registered trademark of Microsoft Corporation

Figure 1. CC78K0 Functional Diagram



Ordering Information

Part Number	System	Description
CC78K0-D52	MS-DOS	5-1/4-inch, double-density floppy diskette

CC78K0 Extended Functions

- Register variables can be stored in the registers and Saddr area
- Saddr area usage for variables
- Direct peripherals access with SFR names
- Saddr area usage for function arguments and automatic variables
- Functions can be called using the CALLT table
- Functions can be stored in the CALLF area
- Bit data type
- In-line assembly language
- Interrupt functions
  - — Generate interrupt vector table
  - — Disable/enable interrupts

6h

Compiler Options

The CC78K0 C compiler supports the following options during compilation:

- Target chip selection
- Parameter file specification
- Macro name definition
- Include files search path specification
- Symbol length extension
- Symbol name conversion to uppercase
- Outputs debug information
- Generates
  - — Object file
  - — Assembler source file (with/without C source)
  - — Cross-reference list file
  - — Error list file (with/without C source)
  - — Preprocess list file
- Listing format control
- ROMable processing
- Optimization option selection
- Run-time error check selection
- Temporary directory specification
- Warning level selection
- Outputs compilation status information

C Library Functions

The CC78K0 C compiler library includes most of the important C library functions that apply to PROM-based embedded systems. All library functions reside in the library files supplied. Header files that declare the set of library functions are also included.

The following library functions are available:

I/O Functions

sprintf	sscanf

Character Functions

isalpha	isupper	islower	isdigit
isalnum	isxdigit	isspace	ispunct
isprint	isgraph	iscntrl	isascii
toupper	tolower	_toupper	_tolower
toascii			

String Functions

strlen	strcpy	strncpy	strcat
strncat	strcmp	strncmp	strchr
strrchr	strpbrk	strspn	strcspn
strstr	strtok	strtol	strtoul
atoi	atol	itoa	ltoa
ultoa			

Memory Functions

malloc	calloc	realloc	free
brk	sbrk	memcpy	memmove
memcmp	memchr	memset	

Program Control Functions

setjmp	longjmp	abort	atexit
exit			

Mathematical Functions

abs	labs	rand	srand
div	ldiv		

Special Functions

qsort	bsearch	strerror	va_start
va_arg	va_end		

License Agreement

CC78K0 is sold under terms of a license agreement included with the compiler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users for one year.

Documentation

For further information on source program formats, C compiler, and actual program examples, refer to the following manuals supplied with the compiler. Additional copies may be obtained from NEC Electronics Inc.

- CC78K Series C Compiler for Language
- CC78K Series C Compiler for Operation

NEC
NEC Electronics Inc.

RA78K0
Relocatable Assembler Package for the μPD78K0 Product Line

September 1993

Description

The RA78K0 relocatable assembler package converts symbolic source code for the μPD78K0 product line of 8-bit, single-chip microcontrollers into executable absolute address object code. The RA78K0 package consists of six separate programs: assembler (RA78K0), linker (LK78K0), hexadecimal format object converter (OC78K0), librarian (LB78K0), list converter (LCNV78K0), and structured assembler (ST78K0).

RA78K0 translates a symbolic source module into a relocatable object module. The assembler verifies that each instruction assembled is valid for the target microcontroller specified at assembly time and produces a listing file and a relocatable object module.

LK78K0 combines multiple relocatable object modules and library modules and converts them into a load module. OC78K0 converts a load module into an ASCII hexadecimal format absolute object code file.

LB78K0 allows commonly used relocatable object modules to be stored in one file and linked into multiple programs, greatly increasing programming efficiency. When a library file is included in the input of the linker, the linker extracts from the library file only those modules required to resolve external references and links them with the other modules.

LCNV78K0 allows relocatable list files to be converted into absolute list files.

The ST78K0 structured assembler preprocessor is a companion program to the RA78K0 relocatable assembler for the NEC μPD78K0 product line of microcontrollers. ST78K0 converts a source code file containing C-like structured assembly statements into a pure assembly language source file, which then can be assembled with RA78K0.

MS-DOS is a registered trademark of Microsoft Corporation.

Features

- Absolute address object code output
- User-selectable and directable output files
- Macro definitions
- Branch optimization
- Conditional assembly
- Extensive error reporting
- Powerful librarian
- C-like structured assembly statements
- Runs under MS-DOS® operating system

Ordering Information

Part Number	System	Description
RA78K0-D52	MS-DOS	5-1/4-inch, double-density floppy diskette

Program Syntax

An RA78K0 source module consists of a series of code, data, or bit segments. Each segment consists of statements composed of up to four fields: symbol, mnemonic, operand, and comment.

The symbol field may contain a label, whose value is the instruction or data address, or a name that represents an instruction address, data address, or constant. The mnemonic field may contain an instruction or assembler directive. The operand field contains the data or expression for the specified instruction or directive. The comment field allows explanatory comments to be added to a program.

Character constants are translated into 7-bit ASCII codes. Numeric constants may be specified as binary, octal, decimal, or hexadecimal. Arithmetic expressions may include the operators +, -, *, /, MOD, OR, AND, NOT, XOR, EQ, NE, LT, LE, GT, GE, SHR, SHL, LOW, HIGH, ., (), and character constants.

Macro Definition

RA78K0 allows the definition of macro code sequences with up to five parameters, LOCAL symbols, and special repeated code sequences. The macro code sequence differs from a subroutine call in that the invocation of a macro in the source code results in the direct replacement of the macro call with the defined code sequence.

Assembler Directives

Assembler directives give instructions to the assembler, but they are not translated into machine code during assembly. Basic assembler directives include storage definition and allocation directives (DB, DW, DS, DBIT); symbol directives (EQU, SET); and location counter control directive (ORG). Program control directives include segment directives (CSEG, DSEG, BSEG, ENDS); linkage directives (NAME, PUBLIC, EXTRN, EXTBIT); macro directives (MACRO, LOCAL, REPT, IRP, EXITM, ENDM); automatic BR instruction directive (BR); and assembly termination directive (END).

Assembler Controls

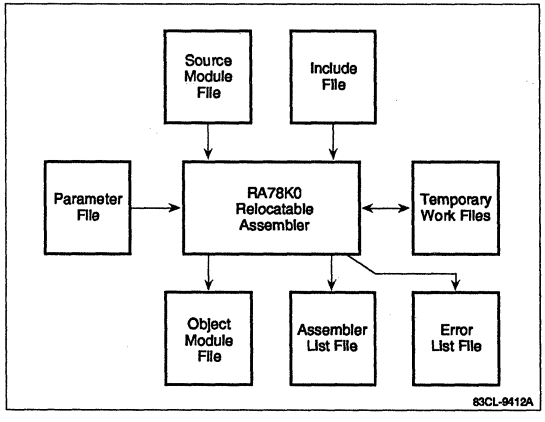
The RA78K0 assembler (figure 1) has two types of controls. Primary controls, specified in the assembler command line or at the beginning of the source module, are as follows:

- Processor selection
- Output object creation selection
- Output list file selection
- Listing format controls
- Optimization selection
- Work file drive specification

General controls, specified in the source program, are as follows:

- Inclusion of other source files
- Page eject
- Generation/suppression of listing
- Listing titles
- Conditional assembly controls

Figure 1. Relocatable Assembler Functional Diagram



The listing file contains either the complete assembly listing or only the lines with errors, and a symbol or cross-reference table. The symbol table shows all defined symbols in alphabetical order with the types, attributes, and the values initially assigned to them.

The cross-reference table contains all defined symbols and the numbers of all statements referring to them. The object file contains the relocatable object module. This is an NEC proprietary relocatable object module format.

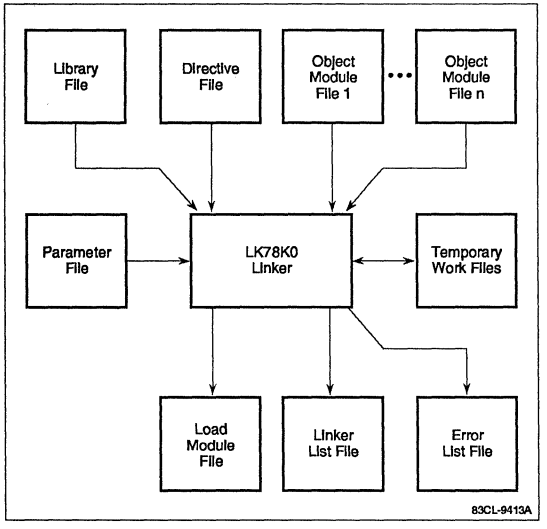
If the optimization option is chosen, the assembler will generate the most efficient code by converting, where possible, three-byte absolute branches into two-byte relative branches.

Linker

The LK78K0 linker (figure 2) combines several relocatable object modules, resolving PUBLIC/EXTRN references between modules, to create a load module. This output module contains both absolute object code and symbol information. The linker will also search library files for required modules to resolve external references. The linker controls for LK78K0 can be specified in either the command line or in a parameter file.

The programmer can specify the starting address and order for code/data/stack segments, and protect areas of memory from being assigned.

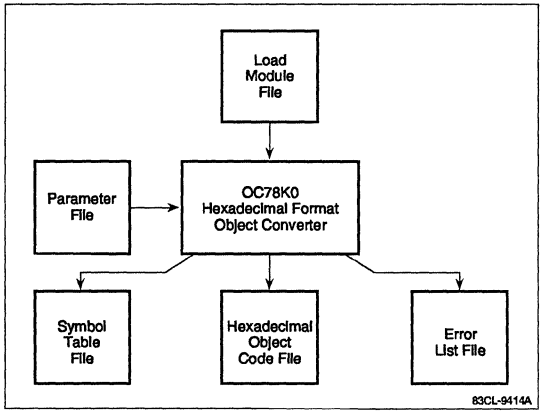
Figure 2. Linker Functional Diagram



Object Converter

The OC78K0 object converter (figure 3) outputs two files: an absolute load file in ASCII hexadecimal format, which can be downloaded to a PROM programmer, and a symbol file for the symbolic debugger. The programmer can also specify an error list file for error logging.

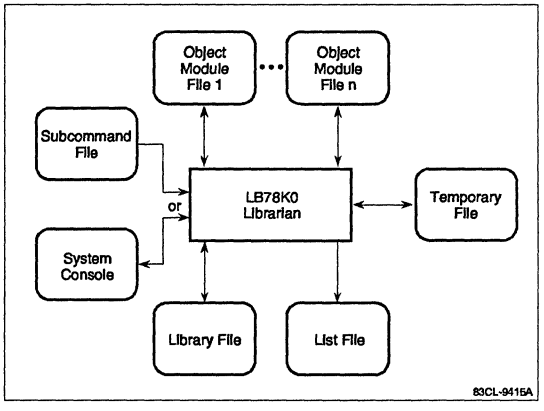
Figure 3. Object Converter Functional Diagram



Librarian

The LB78K0 librarian (figure 4) creates and maintains library files containing relocatable object modules. This reduces the number of files to be linked together by storing several modules in a single file. This provides an easy way to link frequently used modules into programs. Modules can be added to, deleted from, or replaced within a library file, or the contents of the library file can be listed.

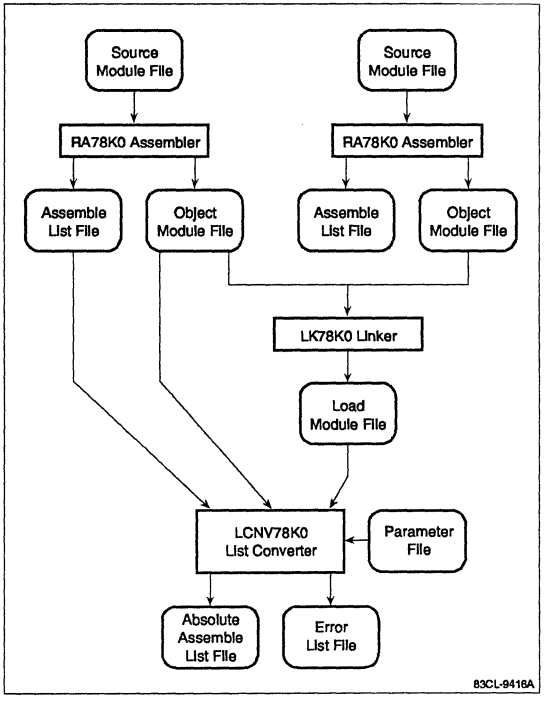
Figure 4. Librarian Functional Diagram



List Converter

The LCNV78K0 list converter (figure 5) converts a relocatable assembly list file into an absolute assembly list file, which contains absolute addresses and symbol values.

Figure 5. List Converter Functional Diagram



6i

Structured Assembler

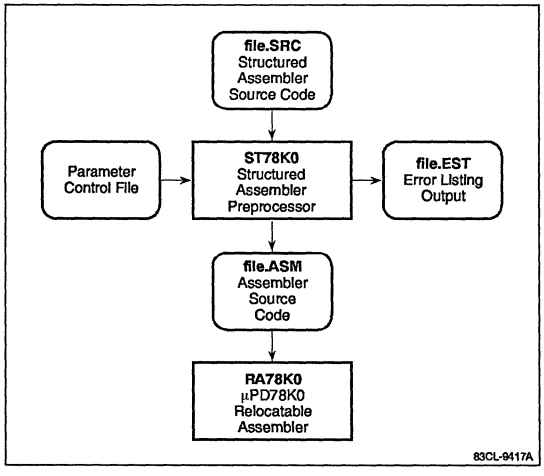
The ST78K0 (figure 6) converts a structured assembly statement into one or more μPD78K0 assembly language instructions that perform the desired operation. Because ST78K0 converts only structured assembly statements and not μPD78K0 assembly language instructions, a structured source program can include a combination of μPD78K0 structured assembly statements and assembly language.

ST78K0 enables the assembly language programmer to use some of the structures and syntax of higher-level languages such as the C language. This improves program readability and reliability, and increases programmer productivity.

Features of the ST78K0

- Control structures for conditions, looping, and switch-case
- Preprocessor directives for conditional code generation
- C-like representation of comparison operations
- C-like representation of assignment/arithmetic operations
- Increment and decrement operators
- Allows use of all μPD78K0 mnemonics, registers, and features

Figure 6. Structured Assembler Preprocessor Functional Diagram



Summary Of Structured Language

A line of source code for ST78K0 contains either a structured assembly statement or a μPD78K0 assembly language statement. μPD78K0 assembly language statements (μPD78K0 instructions, RA78K0 directives, or RA78K0 controls) pass through ST78K0 without change.

Structured assembly statements consist of preprocessor directives, assignment statements, and control statements. These statements are entered one per line, and are terminated by a line feed character. An optional comment may follow a semicolon at the end of the statement; all text following a semicolon is ignored by ST78K0.

Preprocessor directives cause ST78K0 to include or omit portions of code. Assignment statements cause ST78K0 to generate one or more μPD78K0 assembly language instructions to alter the contents of a register or variable. Control statements cause ST78K0 to generate the necessary instructions to test conditions and change control flow based on those conditions.

Preprocessor Directives

ST78K0 preprocessor directives set and test variables, allowing conditional processing of code, include external files and map instructions to μPD78K0 CALLT table reference instructions. Table 1 lists the preprocessor directives and their functions.

Table 1. Preprocessor Directives and Functions

Directive	Function
#define NAME value	Defines the variable NAME, set to the supplied value.
#ifdef ABC <statements> #else <statements> #endif	If ABC has been defined as above, or on the command line with the -D option, the first set of statements is processed and the second set ignored; if ABC has not been defined, or defined as zero, the first set of statements is ignored and the second set is processed.
#include "filename"	The named file is read from disk and processed as if included in the source.
#defcallt @LABEL CALL !label #endcallt	Whenever the instruction "CALL !label" is encountered in the source program, it is replaced by "CALLT [@LABEL]". The label must be defined in the CALLT table.

Assignment, Increment, and Decrement Statements

ST78K0 provides the ability to represent an assignment, or an assignment with an arithmetic operation, in C language syntax:

destination < assign-op> source

The assignment operators (table 2) allow either simple assignment or the combination of an assignment with an arithmetic operation on the source and destination.

Examples:

```
A = B       ;Move contents of B register to A
A + = [HL]  ;Add contents of memory at HL to A,
            ;store in A
```

Where an assignment requires an intermediate register to hold the value being assigned, the register is designated by naming it in parentheses following the assignment operation.

Examples:

```
DATA1 = B (A)   ;Store contents of B into memory at
                ;DATA1, using A as temporary storage
BC & = HL (XA)  ;and BC with HL, store in BC,
                ;use XA as temp
```

The increment and decrement operators (+ + and --) operate on a single operand.

Table 2. Assignment Operators With Examples and Functions

Operator	Example	Function
=	A = B	A ← B
< - >	A < - > B	Contents of A and B are exchanged
+ =	A + = B	A ← A + B
-=	A -= B	A ← A - B
*=	AX *= B	AX ← AX * B
/=	AX /= C	AX ← AX / C
&=	A &= B	A ← A & B (logical AND)
|=	A |= B	A ← A | B (logical OR)
^=	A ^= B	A ← A ^ B (logical XOR)
> > =	A> > = B	(CY←A_0,A_{n-1}←A_n,...,A_{max}←0) x B times
< < =	A< < = B	(CY←A_{max},A_{n+1}←A_n,...A_0←0) x B times
+ +	A+ +	A ← A + 1
--	A--	A ← A - 1

Control Statements

Control statements (table 3) allow conditions to be tested. Based on the results of the test, blocks of code are allowed to be executed or skipped. Reserved words in the control statement define the start and end of blocks of code and expressions to be evaluated.

Example:

```
if ( A = = [HL])    ;The condition is tested.
   P5 = B (A)       ;If A equals the content of memory
   A = [HL]         ;at HL, this code is executed.
else
   A + = [HL]       ;Otherwise, this code is executed.
   A – = B
   P5 = A
endif
```

6i

Table 3. Control Statement Directives

Control Statement	Function
if - elseif - else - endif	Test variable expressions
if_bit - elseif_bit - else - endif	Test bit expressions
switch - case - default - ends	Select based on variable
for - next	Loop, test variable
while - endw	Loop, test variable
repeat - until	Loop, test variable
while_bit - endw	Loop, test bit
repeat - until_bit	Loop, test bit
break	Exit control block
continue	Skip to top of block
goto LABEL	Branch to label

Variable and Bit Expressions

Variable expressions for tests consist of a single value, comparison between two variables, or a logical combination of comparisons. Bit expressions test individual bits. Table 4 shows examples of comparisons. The allowable expressions using variables are shown in table 5.

Table 4. Examples of Variable Expression Comparisons

Comparison	Meaning
if (A)	True if A is non-zero
if (A < B)	True if A is less than B
if ((A < B) && (A > C))	True if A is less than B and greater than C
if_bit (P3.2)	True if bit 2 of P3 is 1
if_bit (!P3.2)	True if bit 2 of P3 is 0

Table 5. Expressions and Examples

Expression	Example
Primary	(A)
Term	(A < = B)
Term && Term	((A< B) && (A> C)) (logical AND)
Term II Term	((A= = C) II (A= = B)) (logical OR)

A primary value for a variable expression is a register name or defined symbol. A term consists of two primary values compared with a binary operator. Table 6 lists the supported binary operators and their meanings.

Table 6. Binary Operators

Binary Operator	Meaning
= =	Equal
!=	Not equal
>	Greater than
> =	Greater than or equal to
<	Less than
< =	Less than or equal to

Bit expressions test individual bits of registers, ports, or memory locations. Table 7 shows the acceptable forms of bit expressions.

Table 7. Bit Expressions and Examples

Bit Expression	Example
Bit_primary	(P2.1)
!Bit_primary	(!CY)
Bit_primary && Bit_primary	(A.0 && CY)
Bit_primary II Bit_primary	(P2.2 II CY)

A Bit_primary can be either a reserved word bit identifier, such as a bit of a register or port (P2.1, CY), or a bit definition symbol (SB0 EQU P2.2).

ST78K0 Operation and Controls

ST78K0 is invoked by specifying the name of the source file, followed by optional controls.

Example:

C> ST78K0 ABC.SRC -DXYZ= 3

ST78K0 reads the specified source file and produces an output assembly language file, which can be input to RA78K0. The output file contains all lines provided in the input source file, plus those generated by ST78K0. Lines containing no statements for the structured assembler are passed through unchanged. Lines with structured assembly statements are placed in the output preceded by a semicolon. RA78K0 treats these lines as comments. These commented lines are then followed by the code generated by ST78K0.

The controls for ST78K0 (table 8) are specified in the preprocessor command line or in a parameter file invoked in the command line.

Table 8. ST78K0 Preprocessor Controls

Control	Function
-Ofilename	Specify name of output assembly source file
-Ffilename	Specify name of parameter file to be read
-Efilename	Specify name of error listing file
-Dsymbol[= value]	Define a symbol (like #define in code)
-I[d:][directory]	Define path for include file
-WTn1,n2,n3	Define TAB settings for generated code
-SCcharacter	Defines word symbol last character

The -O option allows the name of the output file to be specified. If not specified, the output file name defaults to the name of the input source file with the extension .ASM.

The -F option allows a parameter file to be specified, which will be read by ST78K0. This parameter file can contain a list of controls to be given to ST78K0 instead of or in addition to those specified on the command line.

The -E option specifies the name of the error listing file. The error file contains the file name, error number, description of error and the line containing the error. If the -E option is not specified, the error file name defaults to the name of the input source file with the extension .EST.

The -D control allows a symbol to be defined on the command line with an optional value provided. If a symbol is defined but no value is specified, the value defaults to 1. If the source file contains a #define directive that specifies a variable with the same name as the -D control, the value on the command line will override the value in the #define directive.

The -I control specifies a drive or directory other than the current drive and directory to search for include files.

The -WT control specifies the number of TAB characters to insert before labels, instruction mnemonics, and instruction operands generated by ST78K0. This allows clear separation of assembly language instructions coded in the source file from those generated by ST78K0.

The -SCcharacter control specifies the character used as the last character in a word symbol. The character must be a letter of the alphabet or the @, _ or ?. This allows ST78K0 to distinguish between word and byte operations. Symbols ending in this character are treated as word symbols and will generate a word operation (e.g. MOVW). If the -SC option is not specified, ST78K0 assumes that a symbol ending with the character "P" or "p" is a word symbol.

Emulator Controller Program

Absolute object files produced by the RA78K0 relocatable assembler package can be debugged with the appropriate NEC standalone in-circuit emulator. NEC emulator controller programs allow communication with the emulator through an RS-232C serial line. An emulator controller program can run on the IBM PC®, PC/XT®, or PC AT® under MS-DOS and is provided with the in-circuit emulator at no extra charge.

These emulator controller programs provide the following features:

- Uploading/downloading of object and symbol files
- Symbolic debugging capability
- Complete emulator control from host console
- On-line help facilities
- Macro command file capabilities
- Host system directory and file display
- Disk storage of debug session
- Storage of last 20 commands for recall

License Agreement

RA78K0 is sold under terms of a license agreement included with the assembler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users for one year.

Documentation

For further information on source program formats, assembler operation, and actual program examples, refer to the following manuals supplied with the RA78K0 package. Additional copies may be obtained from NEC Electronics Inc.

- RA78K Series Assembler Package, Language Manual
- RA78K Series Assembler Package, Operation Manual
- RA78K Series Structured Assembler Preprocessor, User's Manual

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

NEC
NEC Electronics Inc.

SD78K0
Screen Debugger for the μPD78K0 Product Line

September 1993

Description

The SD78K0 screen debugger is a versatile software tool for debugging C and assembly language programs written for the μPD78K0 product line of microcontrollers. SD78K0 has an easy-to-use, window-like user interface and can perform both source-level and symbolic debugging.

SD78K0 performs software debugging by monitoring and controlling program execution on an NEC in-circuit emulator from a host computer. Program execution information is presented in an organized format on several windows.

The SD78K0 screen debugger is an integral component of the software development tool chain for the μPD78K0 product line of microcontrollers. Its structured and user-friendly debugging environment can significantly improve software reliability as well as programmer productivity.

Features

- C and assembly language source-level debugging
- Extensive program debugging functions
- Window-like user interface
- Mouse support
- User-configurable windowing environment
- User-defined key macro
- Screen dump to a printer or a disk file
- On-line help facility
- Extensive error reporting
- Runs under MS-DOS® operating system

MS-DOS is a registered trademark of Microsoft Corporation

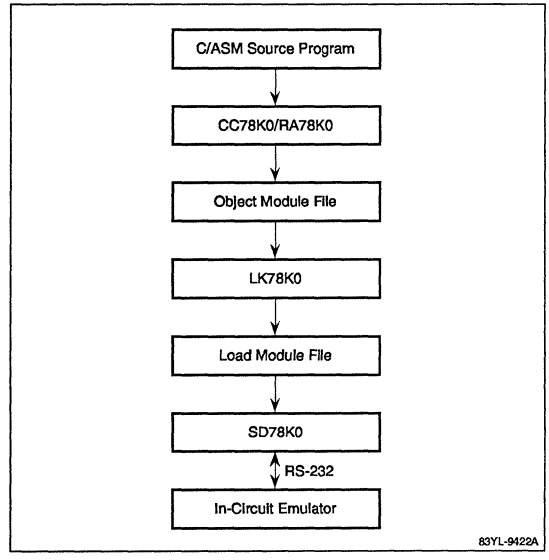
Ordering Information

Part Number	System	Description
SD78K0-D52	MS-DOS	5-1/4-inch, double-density floppy diskette

Development Tool Chain

C source programs can be compiled with the CC78K0 C compiler to generate relocatable object code as shown in figure 1. The RA78K0 relocatable assembler converts assembly language source programs into relocatable object code. The debug option must be specified during program compilation or assembly, which will embed debug information into the object code. The resulting object code can be linked using the LK78K0 linker, which is part of the RA78K0 relocatable assembler package, to generate a load module file. The SD78K0 screen debugger then takes this load module file as input for program debugging on the in-circuit emulator.

Figure 1. Development Tool Chain



6j

Debugging Functions

- Program upload/download
- On-line program assembly/disassembly
- Separate display/working windows for:
- — Register banks
  — Special function registers
  — Memory
  — Source code
  — Symbols
  — Variables
  — Functions
  — Stack
- Breakpoint/watchpoint setting
- Event conditions setting
- Program tracing
- Real-time step emulation
- Performance measurement
- PROM programmer control
- Command file execution
- Save debug session

License Agreement

SD78K0 is sold under terms of a license agreement included with purchased copies of the screen debugger. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users for one year.

Documentation

For further information on features, operations, and session examples, refer to the following manuals supplied with the SD78K0. Additional copies may be obtained from NEC Electronics Inc.

- SD78K/0 Screen Debugger, SD Primer
- SD78K/0 Screen Debugger, SD Reference

NEC Electronics Inc.

IE-78230-R
In-Circuit Emulator
for the μPD78224/238 Families

September 1993

Description

The IE-78230-R is an in-circuit emulator providing both hardware emulation and software debugging capabilities for the NEC μPD78224/238 families of single-chip microcontrollers. Real-time and single-step emulation, combined with sophisticated memory mapping features, breakpoint, and trace capabilities, create a powerful debugging environment. A line assembler/disassembler, full register and memory control, symbolic debugging, and complete upload/download capabilities simplify the task of debugging hardware and software.

Features

- 12-MHz maximum operating frequency
- Real-time and non-real-time emulation
- Sophisticated break events
  - Logical OR of up to four sets of events
    Executed instruction count
    External sense clip condition
    Parallel or sequential fetch addresses
    Logical AND of addresses, data values, CPU controls, and loop count
- Sophisticated trace capabilities
  - Traces program fetch or data access
  - 2K x 44 bit trace buffer
  - Address, control, data, and external signal trace
  - Instruction or frame display
  - Trace search capability
  - Specify trigger point at beginning, middle, or end of trace buffer
- Supports debug activities during real-time emulation
  - Displays trace buffer
  - Modifies trace conditions
  - Modifies trace event conditions
  - Restarts trace
- Powerful memory mapping
  - 32K bytes RAM for μPD78224 or μPD78238 ROM emulation
  - 3840 bytes of RAM for internal RAM emulation
  - 64K bytes of RAM for off-chip RAM/ROM emulation
- Line assembler/disassembler
- Symbolic debugging: 7000 symbols maximum
- CMOS latch-up warning and protection
- Eight external sense clips on emulator probe
- Host control program for IBM PC®, PC/XT®, PC/AT®, or compatible
- Centronics parallel interfaces for optional high-speed program download and printer

IBM PC, PC/XT, and PC/AT are registered trademarks of International Business Machines Corporation.

Equipment Supplied

The IE-78230-R package includes the following:

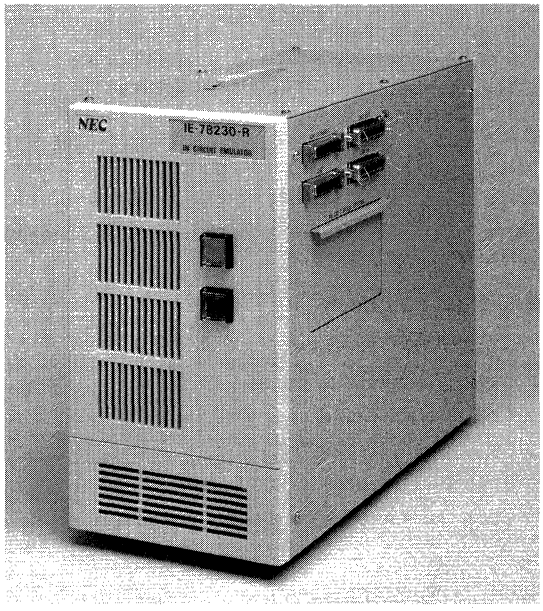
- IE-78230-R emulator frame
- Control/trace board
- Break board (IE-78200-R-EM)
- Emulation board (IE-78230-R-EM)
- Controller software program
- RS-232-C interface cable

Ordering Information (Also, see selection guide.)

Part Number	Description
IE-78230-R	In-circuit emulator for μPD78224/238 families
IE-78230-R-EM	Emulation board included in IE-78230-R package (separately purchased to upgrade 75X or 78X product line emulators to have functions equivalent to IE-78230-R)
EP-78230GC-R	Emulator probe for 80-pin QFP (optional) (includes one EV-9200GC-80 socket adapter)
EP-78230GJ-R	Emulator probe for 94-pin QFP (optional) (includes one EV-9200G-94 socket adapter)
EP-78230LQ-R	Emulator probe for 84-pin PLCC package (optional)
EV-9200GC-80	Five socket adapters; converts 80-pin LCC probe tip to 80-pin plastic QFP (14x14) device footprint (optional)
EV-9200G-94	Five socket adapters; converts 94-pin LCC probe tip to 94-pin QFP device package (optional)
RA78K2-D52	Relocatable assembler for 78K2 product line
CC78K2-D52	C compiler for 78K2 product line
SD78K2-D52	Screen debugger for 78K2 product line*

*Under development

IE-78230-R In-Circuit Emulator



NEC
NEC Electronics Inc.

IE-78240-R
In-Circuit Emulator
for the μPD78214/218A/244 Families

September 1993

Description

The IE-78240-R is an in-circuit emulator providing both hardware emulation and software debugging capabilities for the NEC μPD78214/218A/244 families of single-chip microcontrollers. Real-time and single-step emulation capability, combined with sophisticated memory-mapping features, breakpoints, and trace capabilities, create a powerful debugging environment. A line assembler/disassembler, full register and memory control, symbolic debugging, and complete upload/download capabilities simplify the task of debugging hardware and software.

The IE-78240-R-EM is an optional emulation board available to upgrade the 75X or 78K product lines of in-circuit emulators to have functions equivalent to the IE-78240-R.

Features

- 12-MHz max operating frequency
- Real-time and non-real-time emulation
- Sophisticated break events
  - Logical OR of up to four sets of events
    Executed instruction count
    External sense clip condition
    Parallel or sequential fetch addresses
    Logical AND of addresses, data values, CPU controls, and loop count
- Sophisticated trace capabilities
  - Traces program fetch or data access
  - 2K x 44-bit trace buffer
  - Address, control, data, and external signal trace
  - Instruction or frame display
  - Trace search capability
  - Specify trigger point at beginning, middle, or end of trace buffer
- Supports debug activities during real-time emulation
  - Displays trace buffer
  - Modifies trace conditions
  - Modifies trace event conditions
  - Restarts trace
- Powerful memory mapping
  - 32K bytes RAM for μPD7821x or μPD7824x ROM emulation
  - 3840 bytes of RAM for internal RAM emulation
  - 3840 bytes of EEPROM for μPD7824x EEPROM emulation
  - 64K bytes of RAM for off-chip RAM/ROM emulation
- Line assembler/disassembler
- Symbolic debugging: 7000 symbols maximum
- CMOS latch-up warning and protection
- Eight external sense clips on emulator probe
- Host control program for IBM PC®, PC/XT®, PC/AT®, or compatible
- Centronics parallel interfaces for optional high-speed program download and printer

Equipment Supplied

The IE-78240-R package includes the following:

- IE-78240-R emulator frame
- Control/trace board
- Break board (IE-78200-R-EM)
- Emulation board (IE-78230-R-EM)
- Controller software program
- RS-232-C interface cable

Ordering Information (Also, see selection guide.)

Part Number	Description
IE-78240-R	In-circuit emulator for μPD78214/218A/244 families
IE-78240-R-EM	Separately sold emulation board to upgrade 75X or 78X product lines of in-circuit emulators to have functions equivalent to IE-78240-R
EP-78240CW-R	Emulator probe for 64-pin shrink DIP package (optional)
EP-78240GC-R	Emulator probe for 64-pin QFP (optional) (Includes one EV-9200GC-64 socket adapter)
EP-78240GJ-R	Emulator probe for 74-pin QFP (optional) (Includes one EV-9200G-74 socket adapter)
EP-78240GQ-R	Emulator probe for 64-pin QUIP (optional)
EP-78240LP-R	Emulator probe for 68-pin PLCC package (optional)
EV-9200GC-64	Five socket adapters; converts 64-pin LCC probe tip to 64-pin QFP device footprint. (Optional)
EV-9200G-74	Five socket adapters; converts 74-pin probe tip to 74-pin QFP device footprint. (Optional)
RA78K2-D52	Relocatable assembler for 78K2 product line
CC78K2-D52	C compiler for 78K2 product line
SD78K2-D52*	Screen debugger for 78K2 product line

*Under development

IE-78240-R In-Circuit Emulator

NEC
NEC Electronics Inc.

DDB-78K2
Evaluation Board
for the μPD78K2 Product Line

September 1993

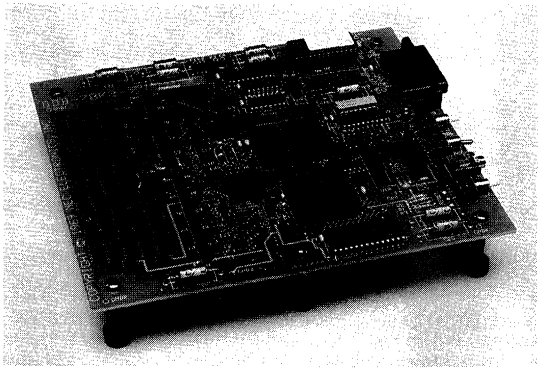
Description

The DDB-78K2 are evaluation boards for the NEC μPD78K2 product line of 8-bit, single-chip microcontrollers. The DDB-78K2 provides maximum flexibility when evaluating and designing with the μPD78K2 product line. Every DDB-78K2 features a μPD78213, μPD78220, μPD78233, or μPD78343 microcontroller, 32K bytes of ROM, 32K bytes of RAM, μPD27C512 footprint for 64K bytes of optional extended data memory, RS-232-C communication port, and a powerful monitor program. A playpen area is included for evaluating the μPD78K2 product line with application specific hardware.

Features

- μPD78213, μPD78220, μPD78233, or μPD78243 evaluation board
  - — Convertible by changing microcontroller
- On-board memory
  - — ROM: 32K bytes
  - — RAM: 32K bytes
- μPD27C512 footprint for 64K bytes of extended data memory
- Powerful on-board debug monitor
  - — Real-time and single-step operation
  - — Display/change memory and internal registers
  - — Multiple software breakpoints
  - — User program download capability
- RS-232-C serial interface for terminal or host computer
- Playpen area for user circuitry
- Includes AC/DC converter

DDB-78K2 Evaluation Board



Equipment and Documentation Supplied

The DDB-78K2-2xx package includes the following:

- DDB-78K2-2XX evaluation board for 78K2 product line
- RA78K2-D52 relocatable assembler for 78K2 product line
- AC/DC converter power supply
- Data book/user's manual

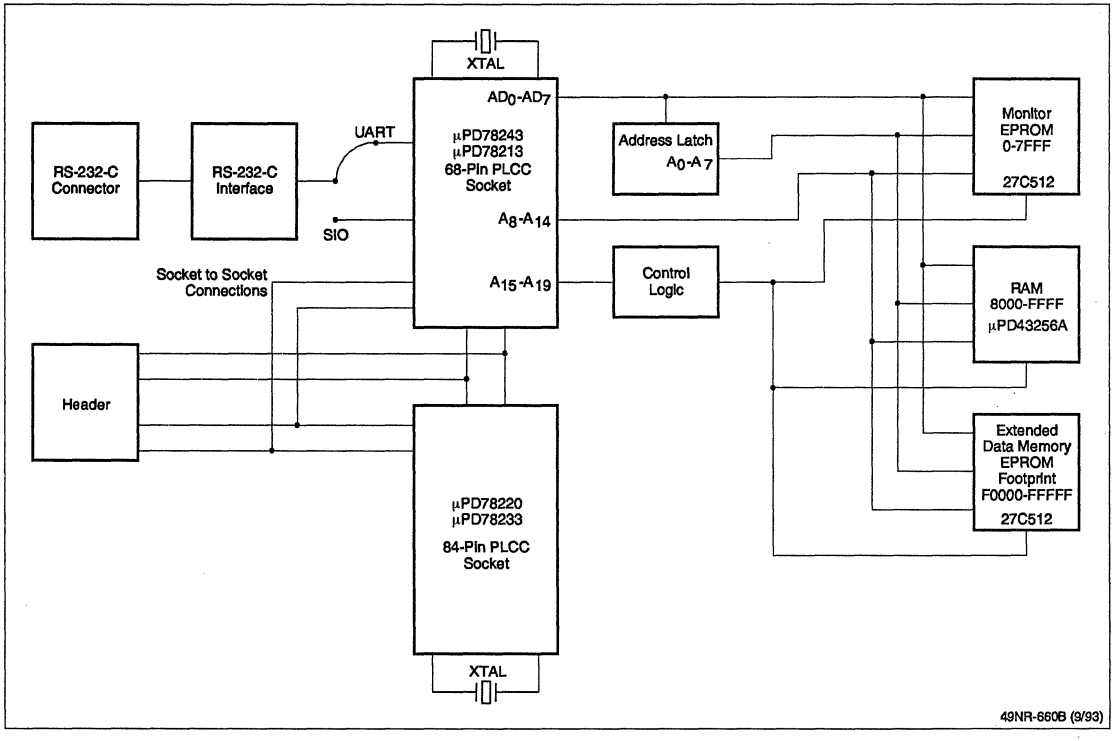
Ordering Information (Also, see selection guide.)

Part Number	Description
DDB-78K2-21x	Basic development board for μPD78218A/214 families designs
DDB-78K2-22x	Basic development board for μPD78224 family designs
DDB-78K2-23x	Basic development board for μPD78238 family designs

6m

50267

DDB-78K2 Block Diagram



NEC
NEC Electronics Inc.

EB-78230-PC
Evaluation Board
for the μPD78238 Family

September 1993

Description

The EB-78230-PC is an evaluation board for the NEC μPD78233 and μPD78234. These devices are 8-bit, single-chip microcontrollers. The EB-78230-PC provides a simple way to evaluate the capabilities of the μPD78233 and μPD78234 in an application without having to build a prototype. If it is necessary to connect the EB-78230-PC directly to a target system, the IE-78230 emulator probes can be purchased separately.

The EB-78230-PC features 32K bytes of static RAM for evaluation programs, an RS-232-C communication port, and a powerful on-board monitor. Evaluation programs can be downloaded from a host computer or created directly on the board using the line assembler. Programs can be executed in real time with or without breakpoints or one instruction at a time. Commands are available to display or change memory, general registers or special function registers, and to disassemble your code.

A controller program controls the EB-78230-PC directly from the console of an IBM PC®, PC/XT®, PC/AT® or compatible host computer using an RS-232-C serial interface.

Features

- 12-MHz max operating frequency
- μPD78233 evaluation board
- 32K bytes static RAM
- Real-time and single-step execution
- Four parallel or sequential breakpoints
- Display/change memory and general registers
- Display/change special function registers
- User program upload/download capability
- Symbolic debugging support
- Line assembler and disassembler
- RS-232-C serial interface for host computer
- Host control software for IBM PC, PC/XT, PC/AT or compatibles
- Connection to a target system using in-circuit emulator probes

IBM PC, PC/XT, and PC/AT are registered trademarks of International Business Machines Corporation.

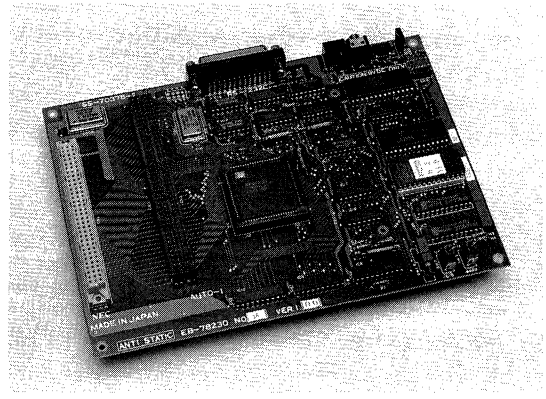
Equipment and Documentation Supplied

- EB-78230-PC evaluation board
- EB-78230-PC user's manual
- System disk for IBM PC
- AC/DC converter power supply
- Battery holder and mounting hardware

Ordering Information (Also, see selection guide.)

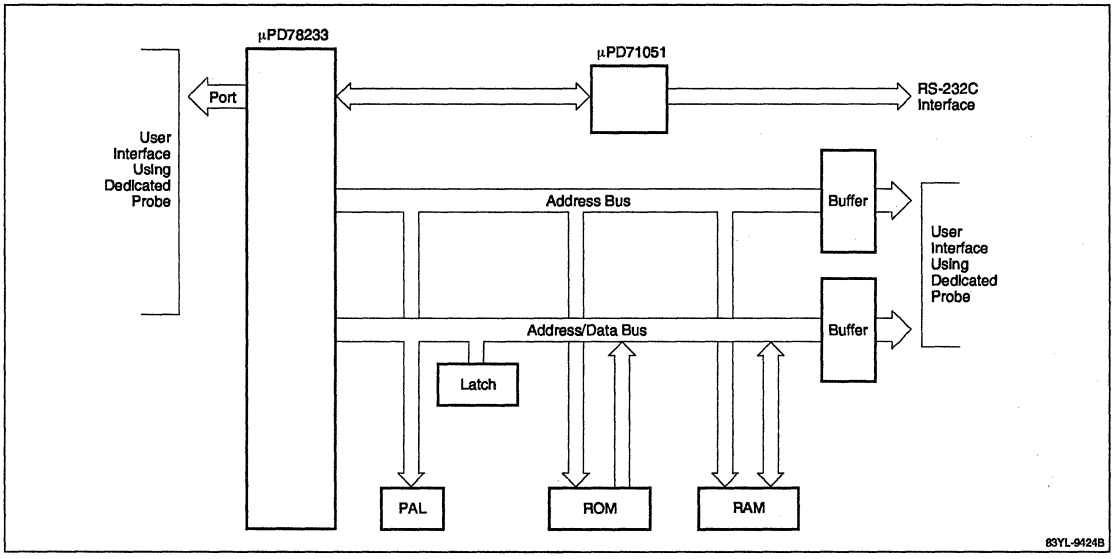
Part Number	Description
EB-78230-PC	μPD78233 Evaluation Board (IBM PC Based)
EP-78230GC-R	Emulator probe for 80-pin QFP (optional) (Includes one EV-9200GC-80 socket adapter)
EP-78230GJ-R	Emulator probe for 94-pin QFP (optional) (Includes one EV-9200G-94 socket adapter)
EP-78230LQ-R	Emulator probe for 84-pin PLCC package (optional)
EV-9200GC-80	Five socket adapters; converts 80-pin LCC probe tip to 80-pin plastic QFP (14x14) device footprint (optional).
EV-9200G-94	Five socket adapters; converts 94-pin LCC probe tip to 94-pin QFP device footprint (optional).

EB-78230-PC Evaluation Board



6n

EB-78230-PC Block Diagram



NEC
NEC Electronics Inc.

EB-78240-PC
Evaluation Board
for the μPD78214/218A/244 Families

September 1993

Description

The EB-78240-PC is an evaluation board for the NEC μPD78214/218A/244 families of 8-bit, single-chip microcontrollers. The EB-78240-PC provides a simple way to evaluate the capabilities of these devices in an application without having to build a prototype. If it is necessary to connect the EB-78240-PC directly to a target system, the IE-78240 emulator probes can be purchased separately.

The EB-78240-PC features 32K bytes of static RAM for evaluation programs, an RS-232-C communication port, and a powerful on-board monitor. Evaluation programs can be downloaded from a host computer or created directly on the board using the line assembler. Programs can be executed in real time with or without breakpoints or one instruction at a time. Commands are available to display or change memory, general registers or special function registers, and to disassemble your code.

A controller program controls the EB-78240-PC directly from the console of an IBM PC®, PC/XT®, PC/AT® or compatible host computer using an RS-232-C serial interface.

Features

- 12-MHz max operating frequency
- μPD78243 evaluation board
- 32K bytes static RAM
- Real-time and single-step execution
- Four parallel or sequential breakpoints
- Display/change memory and general registers
- Display/change special function registers
- User program upload/download capability
- Symbolic debugging support
- Line assembler and disassembler
- RS-232-C serial interface for host computer
- Host control software for IBM PC, PC/XT, PC/AT, or compatibles
- Connection to a target system using in-circuit emulator probes

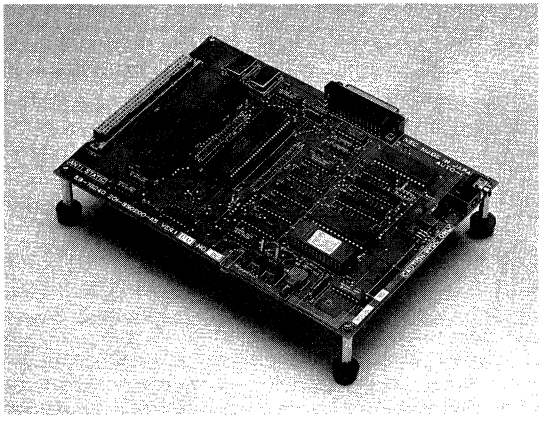
Equipment and Documentation Supplied

- EB-78240-PC evaluation board
- EB-78240-PC user's manual
- System disk for IBM PC
- AC/DC converter power supply
- Battery holder and mounting hardware

Ordering Information (Also, see selection guide.)

Part Number	Description
EB-78240-PC	μPD78243 evaluation board (IBM PC Based)
EP-78240CW-R	Emulator probe for 64-pin shrink DIP package (optional)
EP-78240GC-R	Emulator probe for 64-pin QFP (optional) (Includes one EV-9200GC-64 socket adapter)
EP-78240GJ-R	Emulator probe for 74-pin QFP (optional) (includes one EV-9200G-74 socket adapter)
EP-78240GQ-R	Emulator probe for 64-pin QUIP package (optional)
EP-78240LP-R	Emulator probe for 68-pin PLCC package (optional)
EV-9200GC-64	Five socket adapters; converts 64-pin LCC probe tip to 64-pin QFP device footprint (optional).
EV-9200G-74	Five socket adapters; converts 74-pin probe tip to 74-pin QFP device footprint (optional).

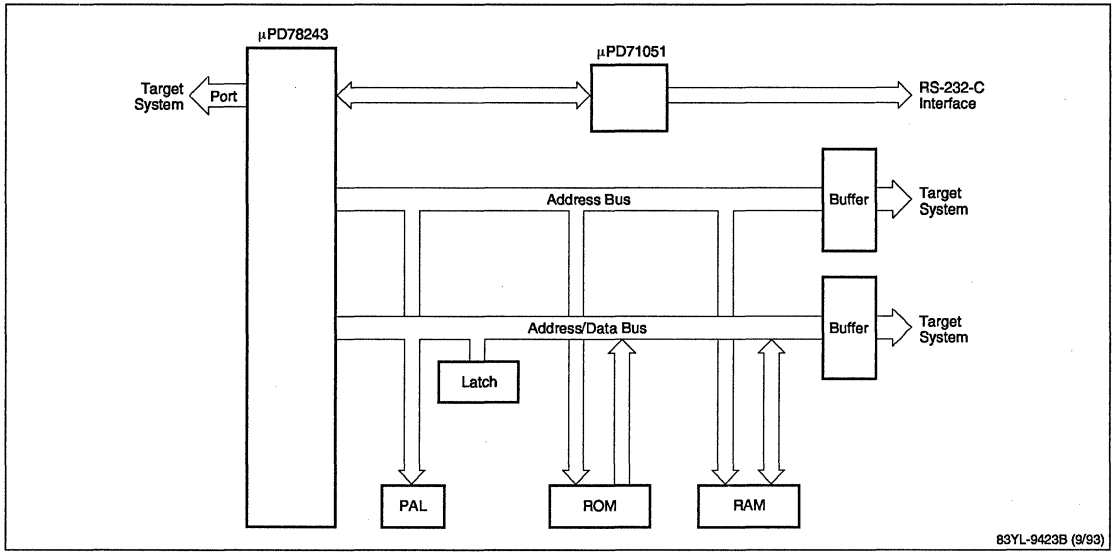
EB-78240-PC Evaluation Board



6o

IBM PC, PC/XT, and PC/AT are registered trademarks of International Business Machines Corporation.

EB-78240-PC Block Diagram



NEC
NEC Electronics Inc.

CC78K2
C Compiler for the
μPD78K2 Product Line

September 1993

Description

The CC78K2 C compiler is an ANSI standard C cross-compiler for the NEC μPD78K2 product line of microcontrollers. The CC78K2 (figure 1) converts ANSI standard C source code into NEC format object module or assembly language source files. During compilation, an optional optimizer can be invoked to optimize the object code for size and/or execution speed.

In addition, CC78K2 supports extended functions for μPD78K2 code generation. These extended functions allow the C compiler to take advantage of many powerful features in the μPD78K2 microcontrollers to decrease object code size and improve program execution speed.

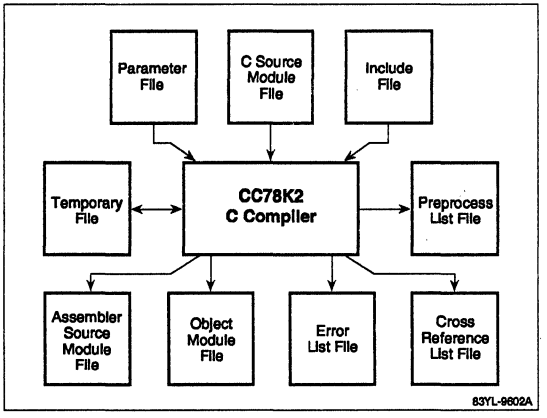
The relocatable object file produced by the CC78K2 can be converted into an absolute object file by the linker program and object converter program contained in the RA78K2 relocatable assembler package. The resulting ASCII hexadecimal format absolute object file then can be debugged using an NEC in-circuit emulator or evaluation board.

Features

- ANSI standard C compiler
- Extended functions for optimized μPD78K2 code generation
- Various optimization options for code size and/or execution speed
- Legal C code verification
- Outputs NEC format object module or assembly source file
- Run-time error checking
- Outputs debug information
- ROMable object file creation
- User selectable and directable output files, list, and full cross-reference files
- Extensive error reporting
- Built-in help facility
- Runs under MS-DOS® operating system

MS-DOS is a registered trademark of Microsoft Corporation

Figure 1. CC78K2 Functional Diagram



Ordering Information

Part Number	System	Description
CC78K2-D52	MS-DOS	5-1/4-inch, double-density floppy diskette

CC78K2 Extended Functions

- Register variables can be stored in the registers and Saddr area
- Saddr area usage for variables
- Direct peripherals access with SFR names
- Saddr area usage for function arguments and automatic variables
- Functions can be called using the CALLT table
- Functions can be stored in the CALLF area
- Bit data type
- In-line assembly language
- Interrupt functions
  - — Generate interrupt vector table
  - — Disable/enable interrupts
- 1-megabyte extended data memory support

6p

Compiler Options

The CC78K2 C compiler supports the following options during compilation:

- Target chip selection
- Parameter file specification
- Macro name definition
- Include files search path specification
- Symbol length extension
- Symbol name conversion to uppercase
- Outputs debug information
- Generates
  - — Object file
  - — Assembler source file (with/without C source)
  - — Cross-reference list file
  - — Error list file (with/without C source)
  - — Preprocess list file
- Listing format control
- ROMable processing
- Optimization option selection
- Run-time error check selection
- Temporary directory specification
- Warning level selection
- Outputs compilation status information

C Library Functions

The CC78K2 C compiler library includes most of the important C library functions that apply to PROM-based embedded systems. All library functions reside in the library files supplied. Header files that declare the set of library functions are also included.

The following library functions are available:

I/O Functions

sprintf	sscanf		

Character Functions

isalpha	isupper	islower	isdigit
isalnum	isxdigit	isspace	ispunct
isprint	isgraph	iscntrl	isascii
toupper	tolower	_toupper	_tolower
toascii			

String Functions

strlen	strcpy	strncpy	strcat
strncat	strcmp	strncmp	strchr
strrchr	strpbrk	strspn	strcspn
strstr	strtok	strtol	strtoul
atoi	atol	itoa	ltoa
ultoa			

Memory Functions

malloc	calloc	realloc	free
brk	sbrk	memcpy	memmove
memcmp	memchr	memset	

Program Control Functions

setjmp	longjmp	abort	atexit
exit			

Mathematical Functions

abs	labs	rand	srand
div	ldiv		

Special Functions

qsort	bsearch	strerror	va_start
va_arg	va_end		

License Agreement

CC78K2 is sold under terms of a license agreement included with the compiler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users for one year.

Documentation

For further information on source program formats, C compiler, and actual program examples, refer to the following manuals supplied with the compiler. Additional copies may be obtained from NEC Electronics Inc.

- CC78K Series C Compiler for Language
- CC78K Series C Compiler for Operation

NEC
NEC Electronics Inc.

RA78K2
Relocatable Assembler Package for the μPD78K2 Product Line

September 1993

Description

The RA78K2 relocatable assembler package converts symbolic source code for the μPD78K2 product line of 8-bit, single-chip microcontrollers into executable absolute address object code. The RA78K2 package consists of six separate programs: assembler (RA78K2), linker (LK78K2), hexadecimal format object converter (OC78K2), librarian (LB78K2), list converter (LCNV78K2), and structured assembler (ST78K2).

RA78K2 translates a symbolic source module into a relocatable object module. The assembler verifies that each instruction assembled is valid for the target microcontroller specified at assembly time and produces a listing file and a relocatable object module.

LK78K2 combines multiple relocatable object modules and library modules and converts them into a load module. OC78K2 converts a load module into an ASCII hexadecimal format absolute object code file.

LB78K2 allows commonly used relocatable object modules to be stored in one file and linked into multiple programs, greatly increasing programming efficiency. When a library file is included in the input of the linker, the linker extracts from the library file only those modules required to resolve external references and links them with the other modules.

LCNV78K2 allows relocatable list files to be converted into absolute list files.

The ST78K2 structured assembler preprocessor is a companion program to the RA78K2 relocatable assembler for the NEC μPD78K2 product line of microcontrollers. ST78K2 converts a source code file containing C-like structured assembly statements into a pure assembly language source file, which then can be assembled with RA78K2.

Features

- Absolute address object code output
- User selectable and directable output files
- Macro definitions
- Branch optimization
- Conditional assembly
- Extensive error reporting
- Powerful librarian
- C-like structured assembly statements
- Runs under MS-DOS® operating system

MS-DOS is a registered trademark of Microsoft Corporation.

Ordering Information

Part Number	System	Description
RA78K2-D52	MS-DOS	5-1/4-inch, double-density floppy diskette

Program Syntax

An RA78K2 source module consists of a series of code, data, or bit segments. Each segment consists of statements composed of up to four fields: symbol, mnemonic, operand, and comment.

The symbol field may contain a label whose value is the instruction or data address or a name that represents an instruction address, data address, or constant. The mnemonic field may contain an instruction or assembler directive. The operand field contains the data or expression for the specified instruction or directive. The comment field allows explanatory comments to be added to a program.

Character constants are translated into 7-bit ASCII codes. Numeric constants may be specified as binary, octal, decimal, or hexadecimal. Arithmetic expressions may include the operators +, -, *, /, MOD, OR, AND, NOT, XOR, EQ, NE, LT, LE, GT, GE, SHR, SHL, LOW, HIGH, ., ( ), and character constants.

Macro Definition

RA78K2 allows the definition of macro code sequences with up to five parameters, LOCAL symbols, and special repeated code sequences. The macro code sequence is different than a subroutine call in that the invocation of a macro in the source code results in the direct replacement of the macro call with the defined code sequence.

Assembler Directives

Assembler directives give instructions to the assembler, but they are not translated into machine code during assembly. Basic assembler directives include

storage definition and allocation directives (DB, DW, DS, DBIT); symbol directives (EQU, SET); and the location counter control directive ORG. Program control directives include segment directives (CSEG, DSEG, BSEG, ENDS); linkage directives (NAME, PUBLIC, EXTRN, EXTBIT); macro directives (MACRO, LOCAL, REPT, IRP, EXITM, ENDM); automatic BR instruction directive (BR); and assembly termination directive (END).

Assembler Controls

The RA78K2 assembler (figure 1) has two types of controls. The primary controls, which are specified in the assembler command line or at the beginning of the source module, are as follows:

- Processor selection
- Output object creation selection
- Output list file selection
- Listing format controls
- Optimization selection
- Work file drive specification

The general controls, specified in the source program, are as follows:

- Inclusion of other source files
- Page eject
- Generation/suppression of listing
- Listing titles
- Conditional assembly controls

The listing file may contain the complete assembly listing or only lines with errors, and a symbol or cross-reference table. The symbol table shows all defined symbols in alphabetical order, their types, attributes, and the values initially assigned to them.

The cross-reference table contains all defined symbols and the numbers of all statements that refer to them. The object file contains the relocatable object module. The format of this module is an NEC proprietary relocatable object module format.

If the optimization option is chosen, the assembler will generate the most efficient code by converting, where possible, three-byte absolute branches into two-byte relative branches.

Figure 1. Relocatable Assembler Functional Diagram

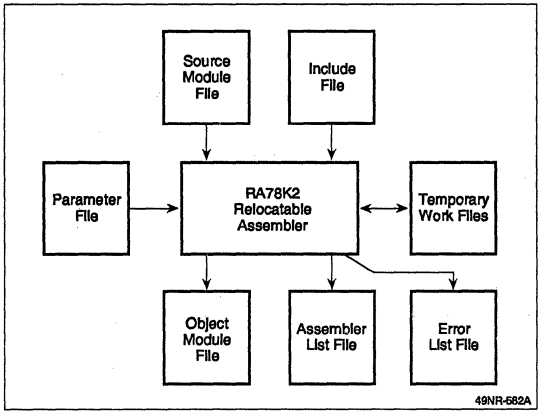
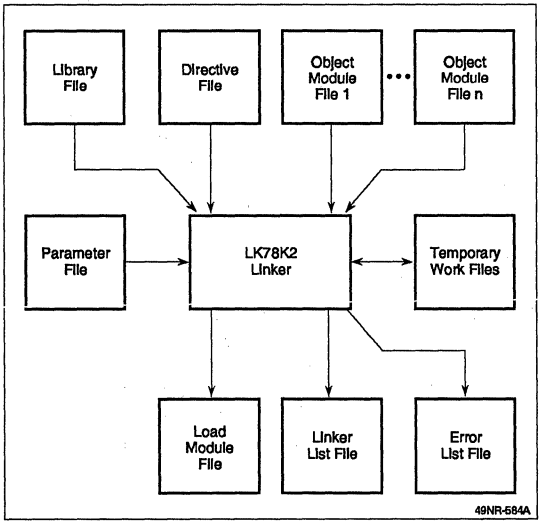


Figure 2. Linker Functional Diagram



Linker

The LK78K2 linker (figure 2) combines several relocatable object modules, resolving PUBLIC/EXTRN references between modules, to create a load module. This output module contains both absolute object code and symbol information. The linker will also search library files for required modules to resolve external references. The linker controls for LK78K2 can be specified in either the command line or in a parameter file.

The programmer can specify the starting address and order for code/data/stack segments, and protect areas of memory from being assigned.

Object Converter

The OC78K2 object converter (figure 3) outputs two files: an absolute load file in ASCII hexadecimal format, which can be downloaded to a PROM programmer, and a symbol file for the symbolic debugger. The programmer can also specify an error list file for error logging.

Librarian

The LB78K2 librarian (figure 4) creates and maintains library files containing relocatable object modules. This reduces the number of files to be linked together by storing several modules in a single file. This provides an easy way to link frequently used modules into programs. Modules can be added to, deleted from, or replaced within a library file, or the contents of the library file can be listed.

List Converter

The LCNV78K2 list converter (figure 5) converts a relocatable assembly list file into an absolute assembly list file, which contains absolute addresses and symbol values.

Figure 3. Object Converter Functional Diagram

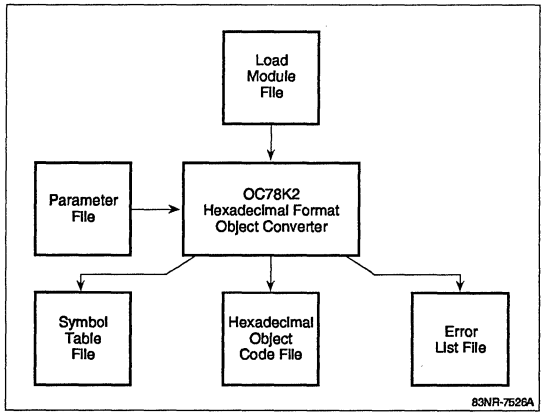


Figure 4. Librarian Functional Diagram

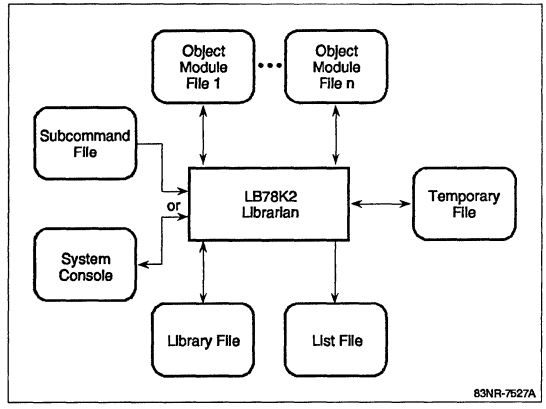
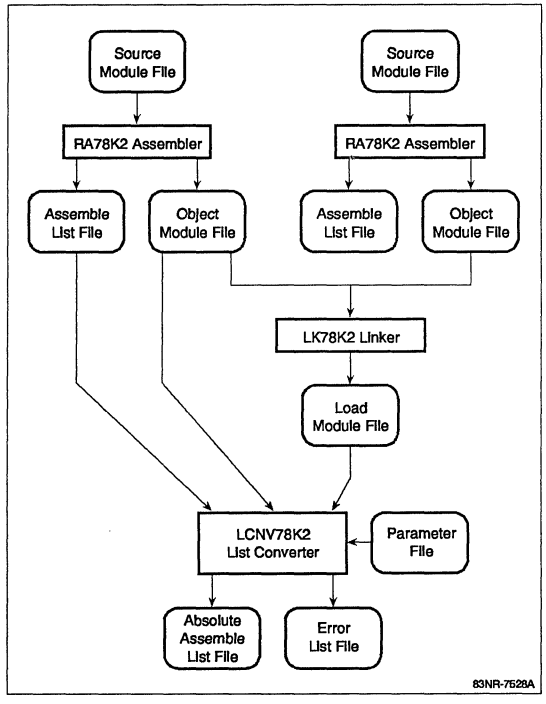


Figure 5. List Converter Functional Diagram



6q

Structured Assembler

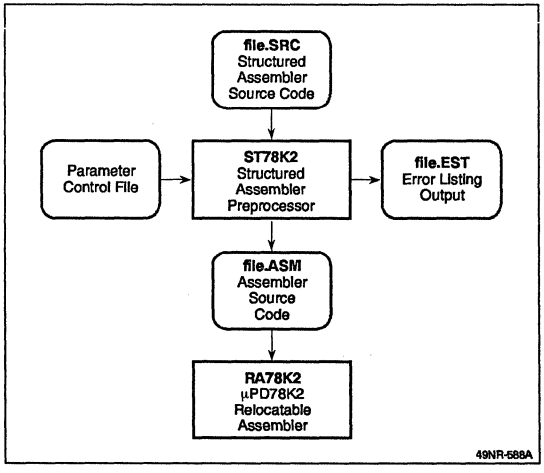
The ST78K2 (figure 6) converts a structured assembly statement into one or more μPD78K2 assembly language instructions that perform the desired operation. Since ST78K2 converts only structured assembly statements and not μPD78K2 assembly language instructions, a structured source program can include a combination of μPD78K2 structured assembly statements and assembly language.

ST78K2 enables the assembly language programmer to use some of the structures and syntax of higher-level languages such as the C language. This improves program readability and reliability, and increases programmer productivity.

Features of the ST78K2

- Control structures for conditions, looping, and switch-case
- Preprocessor directives for conditional code generation
- C-like representation of comparison operations
- C-like representation of assignment/arithmetic operations
- Increment and decrement operators
- Allows use of all μPD78K2 mnemonics, registers, and features

Figure 6. Structured Assembler Preprocessor Functional Diagram



Summary of Structured Language

A line of source code for ST78K2 contains either a structured assembly statement or a μPD78K2 assembly language statement. μPD78K2 assembly language statements (μPD78K2 instructions, RA78K2 directives, or RA78K2 controls) pass through ST78K2 without change.

Structured assembly statements consist of preprocessor directives, assignment statements, and control statements. These statements are entered one per line, and are terminated by a line feed character. An optional comment may follow a semicolon at the end of the statement; all text following a semicolon is ignored by ST78K2.

Preprocessor directives cause ST78K2 to include or omit portions of code. Assignment statements cause ST78K2 to generate one or more μPD78K2 assembly language instructions to alter the contents of a register or variable. Control statements cause ST78K2 to generate the necessary instructions to test conditions and change control flow based on those conditions.

Preprocessor Directives

ST78K2 preprocessor directives set and test variables, allowing conditional processing of code; include external files; and map instructions to μPD78K2 CALLT table reference instructions. Table 1 lists the preprocessor directives and their functions.

Table 1. Preprocessor Directives and Functions

Directive	Function
#define NAME value	Defines the variable NAME, set to the supplied value.
#ifdef ABC <statements> #else <statements> #endif	If ABC has been defined as above, or on the command line with the -D option, the first set of statements is processed and the second set ignored; if ABC has not been defined, or defined as zero, the first set of statements is ignored and the second set is processed.
#include "filename"	The named file is read from disk and processed as if included in the source.
#defcallt @LABEL CALL !label #endcallt	Whenever the instruction "CALL !label" is encountered in the source program, it is replaced by "CALLT [@LABEL]". The label must be defined in the CALLT table.

Assignment, Increment, and Decrement Statements

ST78K2 provides the ability to represent an assignment, or an assignment with an arithmetic operation, in C language syntax:

destination < assign-op> source

The assignment operators (table 2) allow either simple assignment or the combination of an assignment with an arithmetic operation on the source and destination.

Examples:

```
A = B          ;Move contents of B register to A
A + = [HL]     ;Add contents of memory at HL to A,
               ;store in A
```

Where an assignment requires an intermediate register to hold the value being assigned, the register is designated by naming it in parentheses following the assignment operation.

Examples:

```
DATA1 = B (A)  ;Store contents of B into memory at
               ;DATA1, using A as temporary
               ;storage
BC & = HL (XA) ;and BC with HL, store in BC,
               ;use XA as temp
```

The increment and decrement operators (+ + and --) operate on a single operand.

Table 2. Assignment Operators With Examples and Functions

Operator	Example	Function
=	A = B	A ← B
< - >	A < - > B	Contents of A and B are exchanged
+ =	A + = B	A ← A + B
-=	A -= B	A ← A - B
*=	AX *= B	AX ← AX * B
/=	AX /= C	AX ← AX / C
&=	A &= B	A ← A & B (logical AND)
I=	A I= B	A ← A I B (logical OR)
^=	A ^= B	A ← A ^ B (logical XOR)
> > =	A> > = B	(CY←A_0,A_{n-1}←A_n,...,A_{max}←0) x B times
< < =	A< < = B	(CY←A_{max},A_{n+1}←A_n,...A_0←0) x B times
+ +	A+ +	A ← A + 1
--	A--	A ← A - 1

Control Statements

Control statements (table 3) allow conditions to be tested. Based on the results of the test, blocks of code are allowed to be executed or skipped. Reserved words in the control statement define the start and end of blocks of code and expressions to be evaluated.

Example:

```
if (A = = [HL])   ;The condition is tested
   P5 = B (A)     ;If A equals the content of memory
   A = [HL]       ;at HL, this code is executed
else
   A + = [HL]     ;Otherwise this code is executed
   A - = B
   P5 = A
endif
```

Table 3. Control Statement Directives

Control Statement	Function
if - elseif - else - endif	Test variable expressions
if_bit - elseif_bit - else - endif	Test bit expressions
switch - case - default - ends	Select based on variable
for - next	Loop, test variable
while - endw	Loop, test variable
repeat - until	Loop, test variable
while_bit - endw	Loop, test bit
repeat - until_bit	Loop, test bit
break	Exit control block
continue	Skip to top of block
goto LABEL	Branch to label

Variable and Bit Expressions

Variable expressions for tests consist of a single value, comparison between two variables, or a logical combination of comparisons. Bit expressions test individual bits. Table 4 shows examples of comparisons. The allowable expressions using variables are shown in table 5.

Table 4. Examples of Variable Expression Comparisons

Comparison	Meaning
if (A)	True if A is non-zero
if (A < B)	True if A is less than B
if ((A < B) && (A > C))	True if A is less than B and greater than C
if_bit (P3.2)	True if bit 2 of P3 is 1
if_bit (!P3.2)	True if bit 2 of P3 is 0

Table 5. Expressions and Examples

Expression	Example
Primary	(A)
Term	(A < = B)
Term && Term	((A< B) && (A> C)) (logical AND)
Term II Term	((A= = C) II (A= = B)) (logical OR)

A primary value for a variable expression is a register name or defined symbol. A term consists of two primary values compared with a binary operator. Table 6 lists the supported binary operators and their meanings.

Table 6. Binary Operators

Binary Operator	Meaning
= =	Equal
!=	Not equal
>	Greater than
> =	Greater than or equal to
<	Less than
< =	Less than or equal to

Bit expressions test individual bits of registers, ports, or memory locations. Table 7 shows the acceptable forms of bit expressions.

Table 7. Bit Expressions and Examples

Bit Expression	Example
Bit_primary	(P2.1)
!Bit_primary	(!CY)
Bit_primary && Bit_primary	(A.0 && CY)
Bit_primary II Bit_primary	(P2.2 II CY)

A Bit_primary can be either a reserved word bit identifier, such as a bit of a register or port (P2.1, CY), or a bit definition symbol (SB0 EQU P2.2).

ST78K2 Operation and Controls

ST78K2 is invoked by specifying the name of the source file, followed by optional controls.
Example:

```
C> ST78K2 ABC.SRC -DXYZ = 3
```

ST78K2 reads the specified source file and produces an output assembly language file, which can be input to RA78K2. The output file contains all lines provided in the input source file, plus those generated by ST78K2. Lines containing no statements for the structured assembler are passed through unchanged. Lines with structured assembly statements are placed in the output preceded by a semicolon. RA78K2 treats these lines as comments. These commented lines are then followed by the code generated by ST78K2.

The controls for ST78K2 are specified in the preprocessor command line or in a parameter file invoked in the command line. Table 8 lists the ST78K2 preprocessor controls and functions.

Table 8. ST78K2 Preprocessor Controls

Control	Function
-Ofilename	Specify name of output assembly source file
-Ffilename	Specify name of parameter file to be read
-Efilename	Specify name of error listing file
-Dsymbol[= value]	Define a symbol (like #define in code)
-I[d:][directory]	Define path for include file
-WTn1,n2,n3	Define TAB settings for generated code
-SCcharacter	Defines word symbol last character

The -O option allows the name of the output file to be specified. If not specified, the output file name defaults to the name of the input source file with the extension .ASM.

The -F option allows a parameter file to be specified, which will be read by ST78K2. This parameter file can contain a list of controls to be given to ST78K2, instead of or in addition to those specified on the command line.

The -E option specifies the name of the error listing file. The error file contains the file name, error number, description of error and the line containing the error. If the -E option is not specified, the error file name defaults to the name of the input source file with the extension .EST.

The -D control allows a symbol to be defined on the command line, with an optional value provided. If a symbol is defined but no value specified, the value defaults to 1. If the source file contains a #define

directive which specifies a variable with the same name as the -D control, the value on the command line will override the value in the #define directive.

The -I control specifies a drive or directory other than the current drive and directory to search for include files.

The -WT control specifies the number of TAB characters to insert before labels, instruction mnemonics, and instruction operands generated by ST78K2. This allows clear separation of assembly language instructions coded in the source file from those generated by ST78K2.

The -SCcharacter control specifies the character used as the last character in a word symbol. The character must be a letter of the alphabet or the @, _ or ?. This allows ST78K2 to distingush between word and byte operations. Symbols ending in this character are treated as word symbols and will generate a word operation (e.g. MOVW). If the -SC option is not specified, ST78K2 assumes that a symbol ending with the character "P" or "p" is a word symbol.

Emulator Controller Program

Absolute object files produced by the RA78K2 relocatable assembler package can be debugged by using the appropriate NEC standalone in-circuit emulator. NEC emulator controller programs allow communication with the emulator through an RS-232C serial line. An emulator controller program can run on the IBM PC®, PC/XT®, or PC AT® under MS-DOS and is provided with the in-circuit emulator at no extra charge.

These emulator controller programs provide the following features:

- Uploading and downloading of object and symbol files
- Symbolic debugging capability
- Complete emulator control from host console
- On-line help facilities
- Macro command file capabilities
- Host system directory and file display
- Disk storage of debug session
- Storage of last 20 commands for recall

License Agreement

RA78K2 is sold under terms of a license agreement included with the assembler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users for one year.

Documentation

For further information on source program formats, assembler operation, and actual program examples, refer to the following manuals supplied with the RA78K2 package. Additional copies may be obtained from NEC Electronics Inc.

- RA78K Series Assembler Package, Language Manual
- RA78K Series Assembler Package, Operation Manual
- RA78K Series Structured Assembler Preprocessor, User's Manual

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

NEC
NEC Electronics Inc.

IE-78310A-R
In-Circuit Emulator
for the μPD78312A Family

September 1993

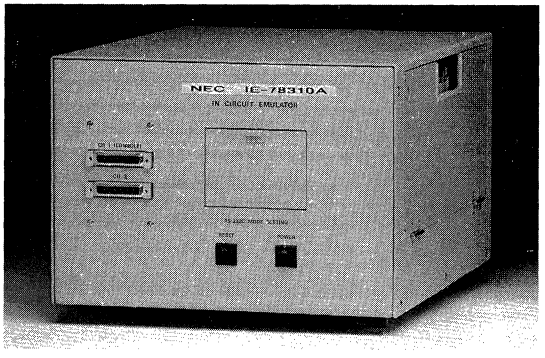
Description

The IE-78310A-R is an in-circuit emulator providing both hardware emulation and software debugging capabilities for the μPD78312A family of single-chip microcontroller. Real-time and single-step emulation, combined with sophisticated memory mapping features, breakpoints, and trace capabilities, create a powerful debugging environment. A line assembler/disassembler, full register and memory control, symbolic debugging, and complete upload/download capabilities simplify the task of debugging hardware and software.

Features

- 12-MHz operating frequency
- Real-time and single-step emulation capability
- User-specified breakpoints
  - — Logical OR of up to four sets of break conditions
    - Opcode fetch count
    - External sense clips condition
    - Emulation time
    - Logical AND of addresses, data values, CPU controls, and loop count
- Sophisticated trace capabilities
  - — Instruction, frame, or macro service display
  - — 2K x 44-bit trace buffer
  - — Address, control, data, and port trace features
- Powerful memory mapping feature
  - — 64K bytes of RAM mappable in 256-byte blocks
  - — Up to 16K bytes of high-speed internal RAM for μPD78312A ROM emulation

IE-78310A-R with Emulator Probe



- Line assembler/disassembler
- Symbolic debugging
  - — 2000 symbols available
  - — IEEE-796 bus memory expansion slot for 32K additional symbols
- CMOS latch-up warning and protection
- Eight external sense clips
- Self-diagnostic command
- Stand-alone mode or system mode with host control program

Equipment Supplied

The IE-78310A-R package includes the following:

- IE-78310A-R emulator frame
- Self-check board
- Emulation board
- Break board
- Control/interface board
- Shrink DIP target probe
- VSP target probe
- External sense clip unit
- System diskette

Ordering Information (Also, see selection guide.)

Part Number	Description
IE-78310A-R	In-circuit emulator for μPD78312A family
EP-78310CW	Emulator probe for 64-pin shrink DIP package(shipped with IE-78310A-R)
EP-78310GQ	Emulator probe for 64-pin QUIP package(shipped with IE-78310A-R)
EP-78310L	Emulator probe for 68-pin PLCC package (optional)
EP-78310GF	Emulator probe for 64-pin QFP package (optional)
RA78K3-D52	Relocatable assembler package for 78K3 product line of microcontrollers (optional)
CC78K3-D52	C compiler package for 78K3 product line of microcontrollers (optional)

NEC
NEC Electronics Inc.

IE-78327-R
In-Circuit Emulator
for the μPD78322 Family

September 1993

Description

The IE-78327-R is an in-circuit emulator providing both hardware emulation and software debugging capabilities for the μPD78322 family of single-chip microcontrollers. Real-time and single-step emulation, combined with sophisticated memory-mapping features, breakpoints, and trace capabilities, create a powerful debugging environment. A line assembler/disassembler, full register and memory control, symbolic debugging, and complete upload/download capabilities simplify the task of debugging hardware and software.

The IE-78327-R-EM is an optional emulation board available to upgrade the 75X or 78K product lines of in-circuit emulators to have functions equivalent to IE-78327-R.

Features

- 16-MHz operating frequency
- Real-time and non-real-time emulation
- Sophisticated break events:
  - Logical OR of up to six events with pass count
    Logical AND of addresses, data values, CPU status, and external sense clips five to eight data (four bus cycle events)
    External sense clips one to four data
    Executed instruction address (four addresses)
  - Sequential enable for bus cycle events
- Sophisticated trace capabilities
  - Three trace modes: unconditional, qualified, and sectional
  - Traces main and internal CPU bus activity and external sense clip activity or time between trace frames
  - Store specified memory/register/SFR contents in trace buffer
  - 8K x 88-bit trace buffer
  - Instruction or frame display
  - Trace search capability
  - Specify trigger point at beginning, middle, or end of trace buffer
- Internal data RAM sampling
  - Up to 2000 three-word samples
  - Sample frequency: 0.4, 0.6, 0.8, or 1 to 10,000 μs
- Program Coverage
  - Display map of memory space containing object code
  - Display map of object code that has been executed
  - Display percentage of executed instructions to total instructions in an area
- Emulation timer and instruction counter
- Debug activities during real-time emulation
  - Display trace buffer
  - Display data sampled from internal RAM
  - Modify trace conditions
  - Modify trace event conditions
  - Restart trace
- Powerful memory mapping:
  56K bytes of RAM for internal ROM, turbo access manager memory, or off-chip memory emulation
- Line assembler/disassembler
- Symbolic debugging: 2000 symbols maximum
- Save/restore in-circuit emulator settings to/from disk
- CMOS latch-up warning and protection
- Eight external sense clips on emulator probe
- Host control program for IBM PC®, PC/XT®, PC/AT®, or compatible
- Centronics parallel interfaces for optional high-speed program download and printer

6s

Equipment Supplied

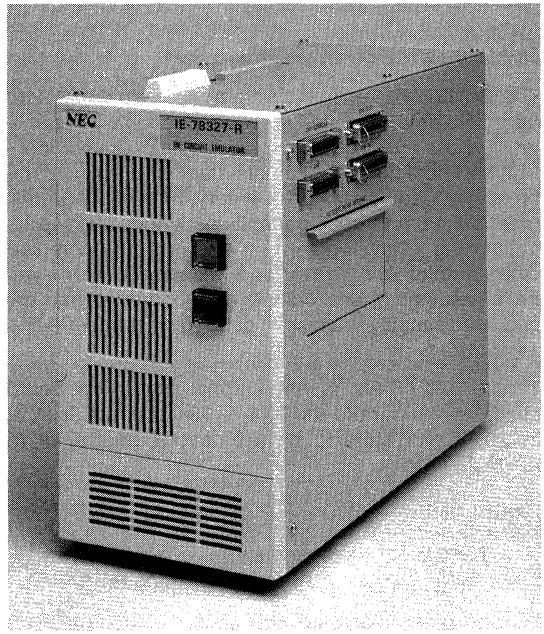
The IE-78327-R package includes the following:

- IE-78327-R emulator frame
- IE-78327-R-EM emulation board
- IE-78327-R-BK break board
- Control/trace board
- Controller software program

Ordering Information (Also, see selection guide.)

Part Number	Description
IE-78327-R	In-circuit emulator for μPD78322 family
IE-78327-R-EM	Emulation board included in IE-78327 package (separately purchased to upgrade 75x or 78K product line emulators to have functions equivalent to IE-78327-R)
IE-78330-R-BK	Break board (separately purchased to upgrade 75x or 78K product line emulators to have functions equivalent to IE-78327-R)
EP-78320GF-R	Emulator probe for 80-pin LCC (optional)
EP-78320GJ-R	Emulator probe for 74-pin QFP (optional) (includes one EV-9200G-74 socket adapter)
EP-78320L-R	Emulator probe for 68-pin PLCC package (optional)
EP-78327CW-R	Emulator probe for 64-pin plastic shrink DIP (optional)
EP-78327GF-R	Emulator probe for 64-pin plastic QFP (optional)
EV-9200G-64	Five socket adapters; converts 64-pin LCC probe tip to 64-pin QFP device footprint.
EV-9200G-74	Five socket adapters; converts 74-pin probe tip to 74-pin QFP device footprint (optional).
EV-9200G-80	Five socket adapters; converts 80-pin LCC probe tip to 80-pin QFP (14x20) device footprint (optional).
RA78K3-D52	Relocatable assembler for K3 product line of microcontrollers (optional)
CC78K3-D52	C compiler for K3 product line of microcontrollers (optional)

IE-78327-R In-Circuit Emulator



NEC
NEC Electronics Inc.

IE-78350-R
In-Circuit Emulator
for the μPD78352/356 Families

September 1993

Description

The IE-78350-R is an in-circuit emulator providing both emulation and software debugging capabilities for the NEC μPD78352 and μPD78356 families of single-chip microcontrollers. Real-time and single-step emulation, combined with sophisticated memory-mapping features, break points, and trace capabilities, create a powerful debugging environment.

The IE-78350-R-EM1 is a separately sold, I/O emulation board used with the IE-78350-R development system for μPD78350 or μPD78352 16/8-bit single-chip microcontrollers.

The IE78355-R-EM1 is a separately sold, I/O emulation board used with the IE-78350-R development system for μPD78355, μPD78P355, and μPD78356 16/8-bit single-chip microconputers.

Features

- 25-MHz max operating frequency
- Real-time and non-real-time emulation
- Sophisticated break events:
  - Logical OR of up to six events
    Pass count
    External sense clips (5-8) data (four bus cycle events)
    External sense clips (1-4) data
    Executed instruction address (four addresses)
    Logical AND of addresses, data values, and CPU status
  - Sequential enable for bus cycle events
- Sophisticated trace capabilities
  - Three trace modes: unconditional, qualified, and sectional
  - Traces main and internal CPU bus activity and external sense clip activity or time between trace frames
  - Store specified memory/register/SFR contents in trace buffer
  - 8K x 88-bit trace buffer
  - Instruction or frame display
  - Trace search capability
  - Specify trigger point at beginning, middle, or end of trace buffer
- Internal data RAM sampling
  - Up to 2000 three-word samples
  - Sample frequency: 0.4, 0.6, 0.8, or 1 to 10,000 μs
- Program Coverage
  - Display map of memory space containing object code
  - Display map of object code that has been executed
  - Display percentage of executed instructions to total instructions in an area
- Emulation timer and instruction counter
- Debug activities during real-time emulation
  - Display trace buffer
  - Display data sampled from internal RAM
  - Modify trace conditions
  - Modify trace event conditions
  - Restart trace
- Powerful memory mapping:
  56K bytes of RAM for internal ROM, turbo access manager memory, or off-chip memory emulation
- Line assembler/disassembler
- Symbolic debugging: 2000 symbols maximum
- Save/restore in-circuit emulator settings to/from disk
- CMOS latch-up warning and protection
- Eight external sense clips on emulator probe
- Host control program for IBM PC®, PC/XT®, PC/AT®, or compatible
- Centronics parallel interfaces for optional high-speed program download and printer

6t

Ordering Information (Also, see selection guide.)

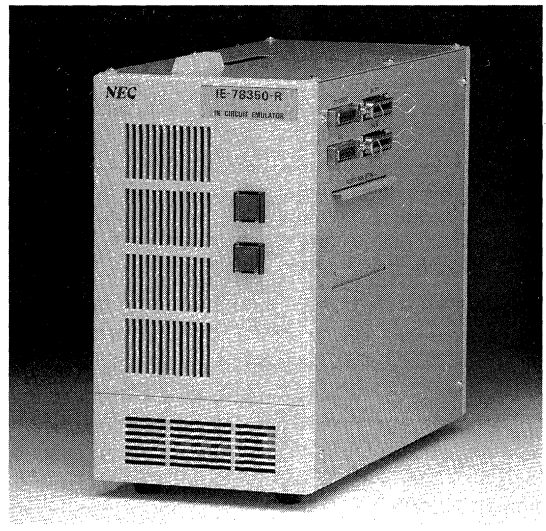
Part Number	Description
IE-78350-R	In-circuit emulator for μPD78352/356 families
IE-78350-R-EM1	I/O emulation board for IE-78350-R (optional) combined with IE-78350-R to emulate μPD78352 family
IE-78355-R-EM1	I/O emulation board for IE-78350-R (optional) combined with IE-78350-R to emulate μPD78356 family
IE-78350-R-EM	Emulation board for IE-78350-R (included with IE-78350-R)
EP-78240GC-R	Emulator probe for 64-pin QFP (optional) (includes one EV-9200GC-64 socket adapter)
EP-78355GC-R	Emulation probe for 100-pin QFP (optional) (includes one EV-9500GC-100 socket adapter)
EV-9200GC-64	Five socket adapters; converts 64-pin LCC probe tip to 64-pin QFP device footprint (optional)
EV-9500GC-100	One socket adapter; 100-pin PGA to 100-pin QFP (optional)
EV-9501GC-100	One socket adapter; 120-pin LCC to 100-pin PGA receptacle (optional)
RA78K3-D52	Relocatable assembler for 78K3 product line (optional)
CC78K3-D52	C compiler for 78K3 product line

Equipment Supplied

The IE-78350-R package includes the following:

- IE-78350-R emulator frame
- IE-78350-R-EM emulation board
- Break board
- Control/trace board (fixed in the IE-78350-R)
- System diskette

IE-78350-R In-Circuit Emulator



NEC
NEC Electronics Inc.

EB-78320-PC
Evaluation Board
for the μPD78322 Family

September 1993

Description

The EB-78320-PC is an evaluation board for the NEC μPD78322 family of 8-/16-bit, single-chip microcontrollers. The EB-78320-PC provides a simple way to evaluate the capabilities of the μPD78322 family in an application without having to build a prototype.

The EB-78320 features 32K bytes of static RAM for evaluation programs, an RS-232C communication port, and a powerful on-board monitor. Evaluation programs can be downloaded from a host computer or created directly on the board using the line assembler. Programs can be executed in real-time with or without breakpoints or one instruction at a time. Commands are available to display or change memory, general or special function registers, and to disassemble code.

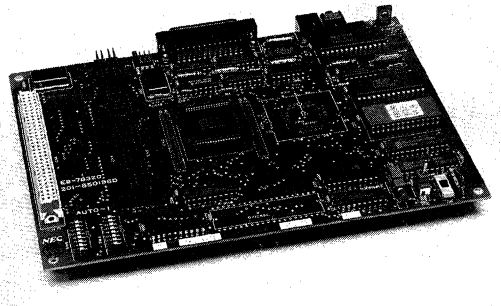
A controller program controls the EB-78320 directly from the console of an IBM PC®, PC/XT®, PC AT®, or compatible host computer using an RS-232C serial interface.

Features

- 16-MHz maximum operating frequency
- μPD78320 evaluation board
- 32K bytes of static RAM
- Real-time and single-step execution
- Four parallel or sequential breakpoints
- Display/change memory and general registers
- Display/change special function registers
- User program upload/download capability
- Symbolic debugging support
- Line assembler and disassembler
- RS-232C serial interface for host computer
- Host control software for IBM PC®, PC/XT®, PC AT®, or compatibles

IBM PC, PC/XT and PC AT are registered trademarks of International Business Machines Corporation.

EB-78320-PC Evaluation Board



Ordering Information (Also, see selection guide.)

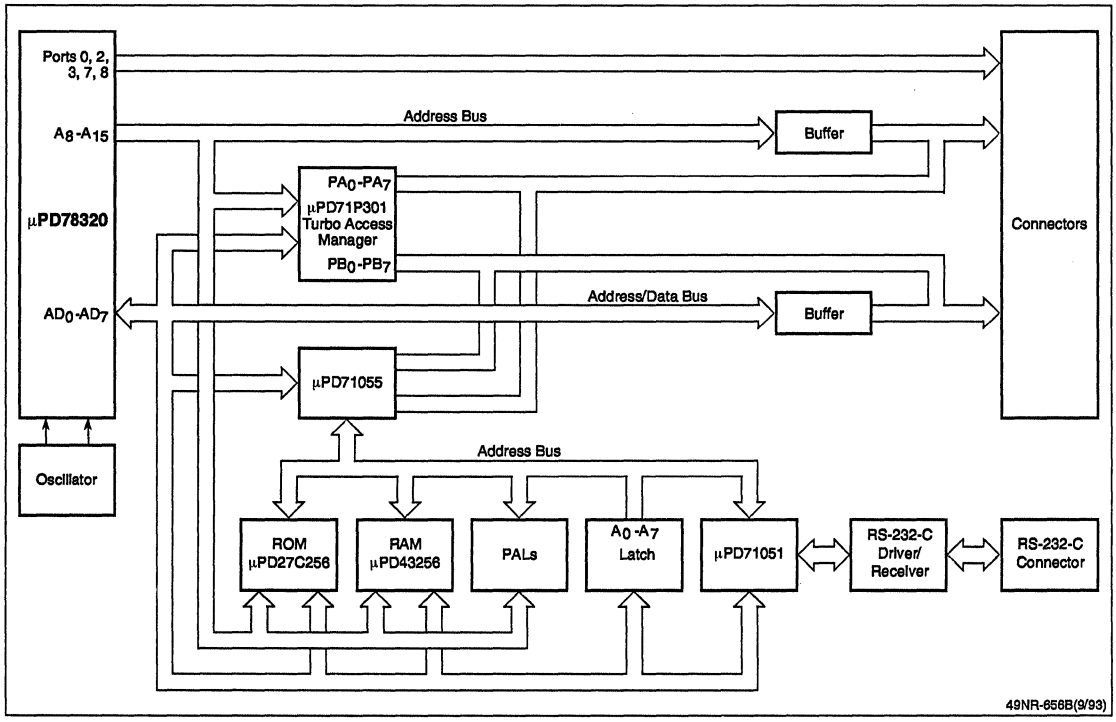
Part Number	Description
EB-78320-PC	μPD78320 evaluation board (IBM PC based)
EP-78320GF-R	Emulator probe for 80-pin QFP package (optional)
EP-78320GJ-R	Emulator probe for 74-pin QFP package (optional)
EP-78320L-R	Emulator probe for 68-pin PLCC package (optional)
EV-9200G-74	Five socket adapters; converts 74-pin probe tip to 74-pin QFP device footprint (optional).
EV-9200G-80	Five socket adapters; converts 80-pin to 80-pin QFP (14x20)device footprint (optional).

Equipment Supplied

The EB-78320-PC package consists of the following:

- EB-78320-PC evaluation board
- EB-78320-PC user's manual
- System disk for IBM PC
- AC/DC converter power supply
- Battery holder and mounting hardware
- Warranty policy and registration card

Block Diagram



NEC
NEC Electronics Inc.

EB-78350-PC Evaluation Board for the μPD78352 Family

September 1993

Description

The EB-78350-PC is an evaluation board for the NEC μPD78352 family of 8-/16-bit, single-chip microcontrollers. The EB-78350-PC provides a simple way to evaluate the capabilities of the μPD78352 family in an application without having to build a prototype.

The μPD78350/352 can be emulated by connecting a separately purchased probe to the EB-78350-PC.

The EB-78350-PC features 32K bytes of static RAM for evaluation programs, an RS-232C communication port, and a powerful on-board monitor. Evaluation programs can be downloaded from a host computer or created directly on the board using an on-line assembler. Programs can be executed in real time with or without breakpoints or one instruction at a time. Commands are available to display or change memory, general or special function registers, and to disassemble code.

A controller program controls the EB-78350-PC directly from the console of an IBM PC®, PC/XT®, PC/AT®, or compatible host computer using an RS-232C serial interface.

Features

- 25-MHz maximum operating frequency
- μPD78350/352 evaluation board
- 32K bytes of static RAM
- Real-time and single-step execution
- Four parallel or sequential breakpoints
- Display/change memory and general registers
- Display/change special function registers
- User program upload/download capability
- Symbolic debugging support
- Line assembler and disassembler
- RS-232C serial interface for host computer
- Host control software for IBM PC, PC/XT, PC/AT or compatibles

IBM PC, PC/XT and PC/AT are registered trademarks of International Business Machines Corporation.

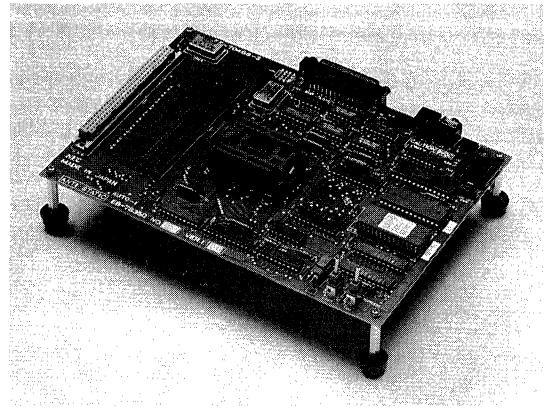
Equipment and Documentation Supplied

- EB-78350-PC evaluation board
- EB-78350-PC user's manual
- System disk for IBM PC
- AC/DC converter power supply
- Battery holder and mounting hardware
- Warranty policy and registration card

Ordering Information (Also, see selection guide.)

Part Number	Description
EB-78350-PC	μPD78350/352 evaluation board
EP-78240GC-R	Emulator probe for 64-pin QFP (optional) (includes one EV-9200GC-64 socket adapter)
EV-9200GC-64	Five socket adapters; converts 64-pin LCC probe tip to 64-pin QFP device footprint (optional).

EB-78350-PC Evaluation Board



NEC Electronics Inc.

CC78K3
C Compiler for the μPD78K3 Product Line

September 1993

Description

The CC78K3 C compiler is an ANSI standard C cross-compiler for the NEC μPD78K3 product line of microcontrollers. The CC78K3 (figure 1) converts ANSI standard C source code into NEC format object module or assembly language source files. During compilation, an optional optimizer can be invoked to optimize the object code for size and/or execution speed.

In addition, CC78K3 supports extended functions for μPD78K3 code generation. These extended functions allow the C compiler to take advantage of many powerful features in the μPD78K3 microcontrollers to decrease object code size and improve program execution speed.

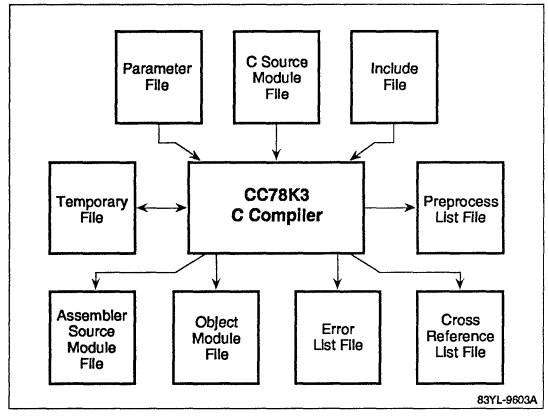
The relocatable object file produced by the CC78K3 can be converted into an absolute object file by the linker program and object converter program contained in the RA78K3 relocatable assembler package. The resulting ASCII hexadecimal format absolute object file then can be debugged using an NEC in-circuit emulator or evaluation board.

Features

- ANSI standard C compiler
- Extended functions for optimized μPD78K3 code generation
- Various optimization options for code size and/or execution speed
- Legal C code verification
- Outputs NEC format object module or assembly source file
- Run-time error checking
- Outputs debug information
- ROMable object file creation
- User selectable and directable output files, list, and full cross-reference files
- Extensive error reporting
- Built-in help facility
- Runs under MS-DOS® operating system

MS-DOS is a registered trademark of Microsoft Corporation

Figure 1. CC78K3 Functional Diagram



Ordering Information

Part Number	System	Description
CC78K3-D52	MS-DOS	5-1/4-inch, double-density floppy diskette

CC78K3 Extended Functions

- Register variables can be stored in the registers and Saddr area
- Saddr area usage for variables
- Direct peripherals access with SFR names
- Saddr area usage for function arguments and automatic variables
- Functions can be called using the CALLT table
- Functions can be stored in the CALLF area
- Bit data type
- In-line assembly language
- Interrupt functions
  - — Generate interrupt vector table
  - — Disable/enable interrupts
- Vector/CALLT table address change

6w

Compiler Options

The CC78K3 C compiler supports the following options during compilation:

- Target chip selection
- Parameter file specification
- Macro name definition
- Include files search path specification
- Symbol length extension
- Symbol name conversion to uppercase
- Outputs debug information
- Generates
  - — Object file
  - — Assembler source file (with/without C source)
  - — Cross-reference list file
  - — Error list file (with/without C source)
  - — Preprocess list file
- Listing format control
- ROMable processing
- Optimization option selection
- Run-time error check selection
- Temporary directory specification
- Warning level selection
- Outputs compilation status information

C Library Functions

The CC78K3 C compiler library includes most of the important C library functions that apply to PROM-based embedded systems. All library functions reside in the library files supplied. Header files that declare the set of library functions are also included.

The following library functions are available:

I/O Functions

sprintf	sscanf		

Character Functions

isalpha	isupper	islower	isdigit
isalnum	isxdigit	isspace	ispunct
isprint	isgraph	iscntrl	isascii
toupper	tolower	_toupper	_tolower
toascii			

String Functions

strlen	strcpy	strncpy	strcat
strncat	strcmp	strncmp	strchr
strrchr	strpbrk	strspn	strcspn
strstr	strtok	strtol	strtoul
atoi	atol	itoa	ltoa
ultoa			

Memory Functions

malloc	calloc	realloc	free
brk	sbrk	memcpy	memmove
memcmp	memchr	memset	

Program Control Functions

setjmp	longjmp	abort	atexit
exit			

Mathematical Functions

abs	labs	rand	srand
div	ldiv		

Special Functions

qsort	bsearch	strerror	va_start
va_arg	va_end		

License Agreement

CC78K3 is sold under terms of a license agreement included with the compiler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users for one year.

Documentation

For further information on source program formats, C compiler, and actual program examples, refer to the following manuals supplied with the compiler. Additional copies may be obtained from NEC Electronics Inc.

- CC78K Series C Compiler for Language
- CC78K Series C Compiler for Operation

NEC
NEC Electronics Inc.

RA78K3
Relocatable Assembler Package for the μPD78K3 Product Line

September 1993

Description

The RA78K3 relocatable assembler package converts symbolic source code for the μPD78K3 product line of 8/16-bit, single-chip microcontrollers into executable absolute address object code. The RA78K3 package consists of six separate programs: assembler (RA78K3), linker (LK78K3), hexadecimal format object converter (OC78K3), librarian (LB78K3), list converter (LCNV78K3), and structured assembler (ST78K3).

RA78K3 translates a symbolic source module into a relocatable object module. The assembler verifies that each instruction assembled is valid for the target microcontroller specified at assembly time and produces a listing file and a relocatable object module.

LK78K3 combines multiple relocatable object and library modules and converts them to a load module. OC78K3 converts a load module into an ASCII hexadecimal format absolute object code file.

LB78K3 allows commonly used relocatable object modules to be stored in one file and linked into multiple programs, greatly increasing programming efficiency. When a library file is included in the input of the linker, the linker extracts from the library file only those modules required to resolve external references and links them with the other modules.

LCNV78K3 allows relocatable list files to be converted into absolute list files.

The ST78K3 structured assembler preprocessor is a companion program to the RA78K3 relocatable assembler for the NEC μPD78K3 product line of microcontrollers. ST78K3 converts a source code file containing C-like structured assembly statements into a pure assembly language source file, which then can be assembled with RA78K3.

The RA78K3 assembler package also includes an ECC generator program (ECCGEN), which generates and applies Error Correcting Code (ECC) to the hexadecimal object module file.

MS-DOS is a registered trademark of Microsoft Corporation.

Features

- Absolute address object code output
- User-selectable and directable output files
- Macro definitions
- Branch optimization
- Conditional assembly
- Extensive error reporting
- Powerful librarian
- C-like structured assembly statements
- Runs under MS-DOS® operating system

Ordering Information

Part Number	System	Description
RA78K3-D52	MS-DOS	5-1/4-inch, double-density floppy diskette

Program Syntax

An RA78K3 source module consists of a series of code, data, or bit segments. Each segment consists of statements composed of up to four fields: symbol, mnemonic, operand, and comment.

The symbol field may contain a label, whose value is the instruction or data address, or a name that represents an instruction address, data address, or constant. The mnemonic field may contain an instruction or assembler directive. The operand field contains the data or expression for the specified instruction or directive. The comment field allows explanatory comments to be added to a program.

Character constants are translated into 7-bit ASCII codes. Numeric constants may be specified as binary, octal, decimal, or hexadecimal. Arithmetic expressions may include the operators +, -, *, /, MOD, OR, AND, NOT, XOR, EQ, NE, LT, LE, GT, GE, SHR, SHL, LOW, HIGH, ., ( ), and character constants.

Macro Definition

RA78K3 allows the definition of macro code sequences with up to five parameters, LOCAL symbols, and special repeated code sequences. The macro code sequence differs from a subroutine call: the invocation of a macro in the source code results in the direct replacement of the macro call with the defined code sequence.

Assembler Directives

Assembler directives give instructions to the assembler. They are not translated into machine code during assembly. Basic assembler directives include storage definition and allocation directives (DB, DW, DS, DBIT); symbol directives (EQU, SET); and location counter control directive (ORG). Program control directives include segment directives (CSEG, DSEG, BSEG, ENDS); linkage directives (NAME, PUBLIC, EXTRN, EXTBIT); macro directives (MACRO, LOCAL, REPT, IRP, EXITM, ENDM); automatic BR instruction directive (BR); register assignment directive (RSS); and assembly termination directive (END).

Assembler Controls

The RA78K3 assembler (figure 1) has two types of controls. Primary controls are specified in the assembler command line or at the beginning of the source module as follows:

- Processor selection
- Output object creation selection
- Output list file selection
- Listing format controls
- Optimization selection
- Work file drive specification

General controls are specified in the source program as follows:

- Inclusion of other source files
- Page eject
- Generation/suppression of listing
- Listing titles
- Conditional assembly controls

The listing file contains either the complete assembly listing or only the lines with errors, and a symbol or cross-reference table. The symbol table shows all defined symbols in alphabetical order with the types, attributes, and the values initially assigned to them.

The cross-reference table contains all defined symbols and the numbers of all statements referring to them. The object file contains the relocatable object module. This is an NEC proprietary relocatable object module format.

If the optimization option is chosen, the assembler will generate the most efficient code by converting, where possible, three-byte absolute branches into two-byte relative branches.

Linker

The LK78K3 linker (figure 2) combines several relocatable object modules, resolving PUBLIC/EXTRN references between modules, to create a load module. This output module contains both absolute object code and symbol information. The linker will also search library files for required modules to resolve external references. The linker controls for LK78K3 can be specified in either the command line or in a parameter file.

Figure 1. Relocatable Assembler Functional Diagram

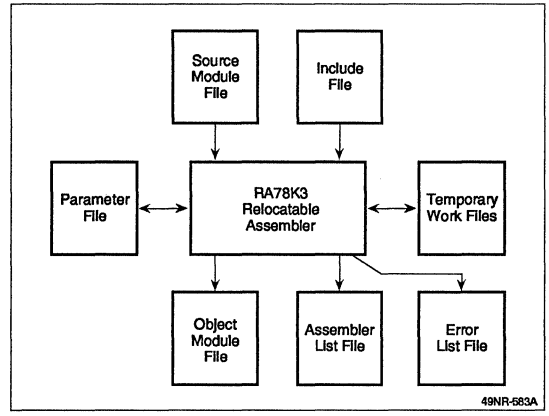
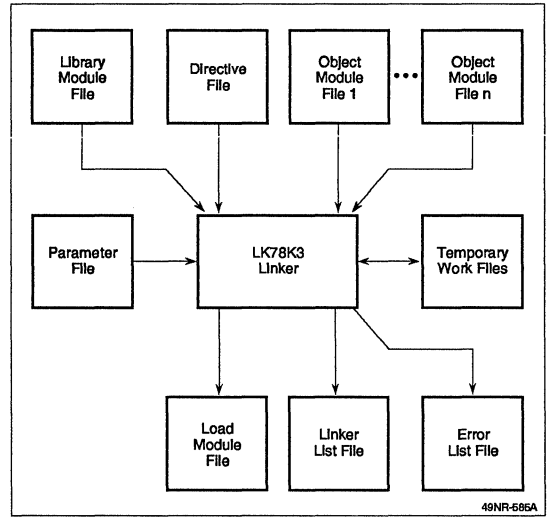


Figure 2. Linker Functional Diagram

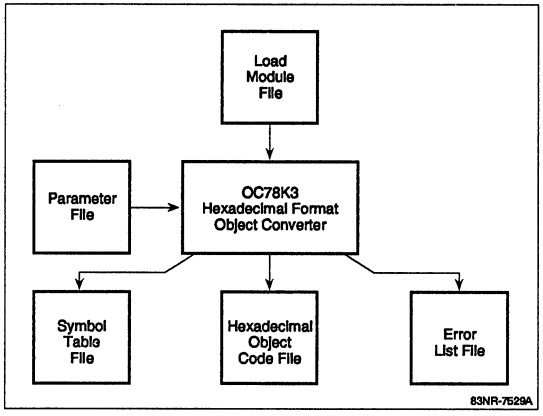


The programmer can specify the starting address and order for code/data/stack segments and protect areas of memory from being assigned.

Object Converter

The OC78K3 object converter (figure 3) outputs two files: an absolute load file in ASCII hexadecimal format, which can be downloaded to a PROM programmer, and a symbol file for the symbolic debugger. The programmer can also specify an error list file for error logging.

Figure 3. Object Converter Functional Diagram



Librarian

The LB78K3 librarian (figure 4) creates and maintains library files containing relocatable object modules. This reduces the number of files to be linked together by storing several modules in a single file. This provides an easy way to link frequently used modules into programs. Modules can be added to, deleted from, or replaced within a library file; or the contents of the library file can be listed.

List Converter

The LCNV78K3 list converter (figure 5) converts a relocatable assembly list file into an absolute assembly list file, which contains absolute addresses and symbol values.

Figure 4. Librarian Functional Diagram

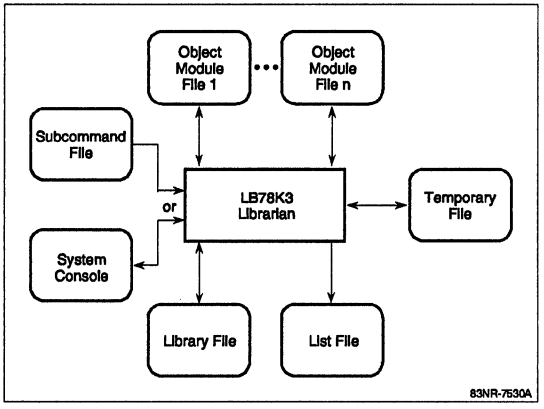
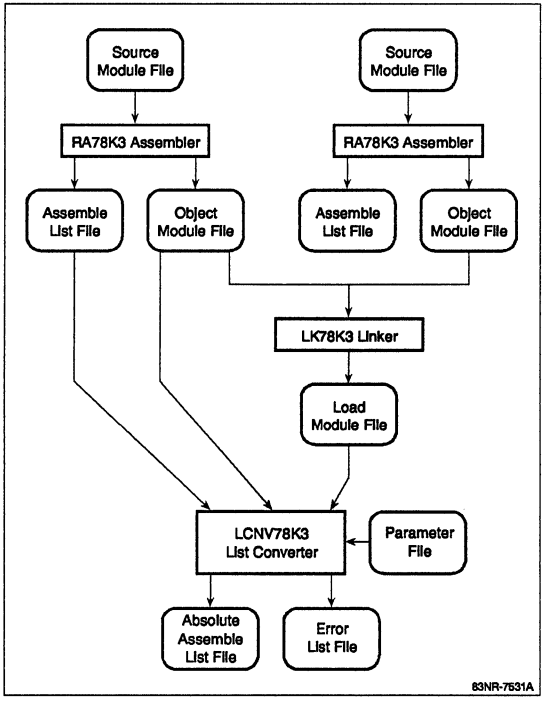


Figure 5. List Converter Functional Diagram



6x

Structured Assembler

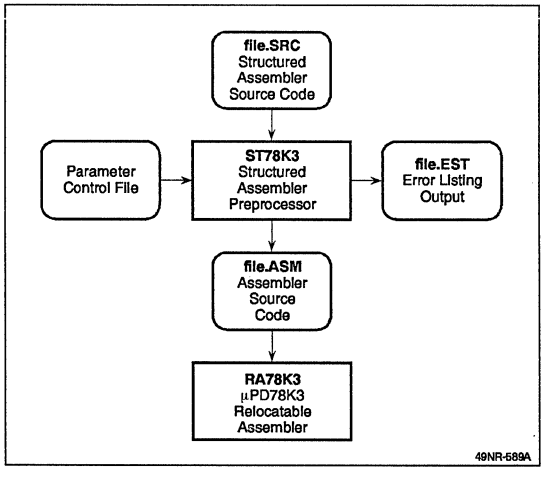
The ST78K3 (figure 6) will convert a structured assembly statement into one or more μPD78K3 assembly language instructions that perform the desired operation. Because ST78K3 converts only the structured assembly statements and not μPD78K3 assembly language instructions, a structured source program can include a combination of μPD78K3 structured assembly statements and assembly language.

ST78K3 enables the assembly language programmer to use some of the structures and syntax of higher-level languages such as the C language. This improves program readability and reliability, and increases programmer productivity.

Features of the ST78K3

- Control structures for conditions, looping, and switch-case
- Preprocessor directives for conditional code generation
- C-like representation of comparison operations
- C-like representation of assignment/arithmetic operations
- Increment and decrement operators
- Allows use of all μPD78K3 mnemonics, registers, and features

Figure 6. Structured Assembler Preprocessor Functional Diagram



Summary Of Structured Language

A line of source code for ST78K3 contains either a structured assembly statement or a μPD78K3 assembly language statement. μPD78K3 assembly language statements (μPD78K3 instructions, RA78K3 directives, or RA78K3 controls) pass through ST78K3 without change.

Structured assembly statements consist of preprocessor directives, assignment statements, and control statements. These statements are entered one per line, and are terminated by a line feed character. An optional comment may follow a semicolon at the end of the statement; all text following a semicolon is ignored by ST78K3.

Preprocessor directives cause ST78K3 to include or omit portions of code. Assignment statements cause ST78K3 to generate one or more μPD78K3 assembly language instructions to alter the contents of a register or variable. Control statements cause ST78K3 to generate the necessary instructions to test conditions and change control flow based on those conditions.

Preprocessor Directives

ST78K3 preprocessor directives set and test variables, allowing conditional processing of code; include external files; and map instructions to μPD78K3 CALLT table reference instructions. Table 1 lists the preprocessor directives and their functions.

Table 1. Preprocessor Directives and Functions

Directive	Function
#define NAME value	Defines the variable NAME, set to the supplied value.
#ifdef ABC < statements> #else < statements> #endif	If ABC has been defined as above, or on the command line with the -D option, the first set of statements is processed and the second set ignored; if ABC has not been defined, or defined as zero, the first set of statements is ignored and the second set is processed.
#include "filename"	The named file is read from disk and processed as if included in the source.
#defcallt @LABEL CALL !label #endcallt	Whenever the instruction "CALL !label" is encountered in the source program, it is replaced by "CALLT [@LABEL]". The label must be defined in the CALLT table.

Assignment, Increment, and Decrement Statements

ST78K3 provides the ability to represent an assignment, or an assignment with an arithmetic operation, in C language syntax:

destination < assign-op> source

The assignment operators (table 2) allow either simple assignment or the combination of an assignment with an arithmetic operation on the source and destination.

Examples:

```
A = B       ;Move contents of B register to A
A + = [HL]  ;Add contents of memory at HL to A,
            ;store in A
```

Where an assignment requires an intermediate register to hold the value being assigned, the register is designated by naming it in parentheses following the assignment operation.

Examples:

```
DATA1 = B (A)   ;Store contents of B into memory at
                ;DATA1, using A as temporary storage
BC & = HL (XA)  ;and BC with HL, store in BC,
                ;use XA as temp
```

The increment and decrement operators (+ + and --) operate on a single operand.

Table 2. Assignment Operators with Examples and Functions

Operator	Example	Function
=	A = B	A ← B
< - >	A < - > B	Contents of A and B are exchanged
+ =	A + = B	A ← A + B
-=	A -= B	A ← A - B
*=	AX *= B	AX ← AX * B
/=	AX /= C	AX ← AX / C
&=	A &= B	A ← A & B (logical AND)
|=	A |= B	A ← A | B (logical OR)
^=	A ^= B	A ← A ^ B (logical XOR)
> > =	A> > = B	(CY←A_0,A_{n-1}←A_n,...,A_{max}←0) x B times
< < =	A< < = B	(CY←A_{max},A_{n+1}←A_n,...,A_0←0) x B times
+ +	A+ +	A ← A + 1
- -	A- -	A ← A - 1

Control Statements

Control statements (table 3) allow conditions to be tested. Based on the results of the test, blocks of code are allowed to be executed or skipped. Reserved words in the control statement define the start and end of blocks of code and expressions to be evaluated.

Example:

```
if ( A = = [HL])    ;The condition is tested.
   P5 = B (A)       ;If A equals the content of memory
   A = [HL]         ;at HL, this code is executed.
else
   A + = [HL]       ;Otherwise, this code is executed.
   A – = B
   P5 = A
endif
```

6x

Table 3. Control Statements and Function

Control Statement	Function
if - elseif - else - endif	Test variable expressions
if_bit - elseif_bit - else - endif	Test bit expressions
switch - case - default - ends	Select based on variable
for - next	Loop, test variable
while - endw	Loop, test variable
repeat - until	Loop, test variable
while_bit - endw	Loop, test bit
repeat - until_bit	Loop, test bit
break	Exit control block
continue	Skip to top of block
goto LABEL	Branch to label

Variable and Bit Expressions

Variable expressions for tests consist of a single value, comparison between two variables, or a logical combination of comparisons. Bit expressions test individual bits. Table 4 shows examples of comparisons. The allowable expressions using variables are shown in table 5.

Table 4. Examples of Variable Expression Comparisons

Comparison	Meaning
if (A)	True if A is non-zero
if (A < B)	True if A is less than B
if ((A < B) && (A > C))	True if A is less than B and greater than C
if_bit (P1.2)	True if bit 2 of P1 is 1
if_bit (!P1.2)	True if bit 2 of P1 is 0

Table 5. Expressions and Examples

Expression	Example
Primary	(A)
Term	(A < = B)
Term && Term	((A< B) && (A> C)) (logical AND)
Term II Term	((A= = C) II (A= = B)) (logical OR)

A primary value for a variable expression is a register name or defined symbol. A term consists of two primary values compared with a binary operator. Table 6 lists the supported binary operators and their meanings.

Table 6. Binary Operators

Binary Operator	Meaning
= =	Equals
!=	Not equal
>	Greater than
> =	Greater than or equal
<	Less than
< =	Less than or equal

Bit expressions test individual bits of registers, ports, or memory locations. Table 7 shows the acceptable forms of bit expressions.

Table 7. Bit Expressions and Examples

Bit Expression	Example
Bit_primary	(P0.1)
!Bit_primary	(!CY)
Bit_primary && Bit_primary	(A.0 && CY)
Bit_primary II Bit_primary	(P0.2 II CY)

A Bit_primary can be either a reserved word bit identifier, such as a bit of a register or port (P0.1, CY), or a bit definition symbol (SB0 EQU P0.2).

ST78K3 Operation and Controls

ST78K3 is invoked by specifying the name of the source file, followed by optional controls. For example:

C> ST78K3 ABC.SRC -DXYZ= 3

ST78K3 reads the specified source file and produces an output assembly language file, which can be input to RA78K3. The output file contains all lines provided in the input source file, plus those generated by ST78K3. Lines containing no statements for the structured assembler are passed through unchanged. Lines with structured assembly statements are placed in the output preceded by a semicolon. RA78K3 treats these lines as comments. These commented lines are then followed by the code generated by ST78K3.

The controls for ST78K3 are specified in the preprocessor command line or in a parameter file invoked in the command line. Table 8 lists the ST78K3 preprocessor controls and functions.

Table 8. ST78K3 Preprocessor Controls

Control	Function
-Ofilename	Specify name of output assembly source file
-Ffilename	Specify name of parameter file to be read
-Efilename	Specify name of error listing file
-Dsymbol[=value]	Define a symbol (like #define in code)
-I[d:][directory]	Define path for include file
-WTn1,n2,n3	Define TAB settings for generated code
-SCcharacter	Defines word symbol last character

The -O option allows the name of the output file to be specified. If not specified, the output file name defaults to the name of the input source file with the extension .ASM.

The -F option allows a parameter file to be specified, which will be read by ST78K3. This parameter file can contain a list of controls to be given to ST78K3, instead of or in addition to those specified on the command line.

The -E option specifies the name of the error listing file. The error file contains the file name, error number, description of error and the line containing the error. If the -E option is not specified, the error file name defaults to the name of the input source file with the extension .EST.

The -D control allows a symbol to be defined on the command line, with an optional value provided. If a symbol is defined but no value is specified, the value defaults to 1. If the source file contains a #define directive which specifies a variable with the same name as the -D control, the value on the command line will override the value in the #define directive.

The -I control specifies a drive or directory other than the current drive and directory to search for include files.

The -WT control specifies the number of TAB characters to insert before labels, instruction mnemonics, and instruction operands generated by ST78K3. This allows clear separation of assembly language instructions coded in the source file from those generated by ST78K3.

The -SCcharacter control specifies the character used as the last character in a word symbol. The character must be a letter of the alphabet or the @, _ or ?. This allows ST78K3 to distinguish between word and byte operations. Symbols ending in this character are treated as word symbols and will generate a word operation (e.g. MOVW). If the -SC operation is not specified, ST78K3 assumes that a symbol ending with the character "P" or "p" is a word symbol.

Emulator Controller Program

Absolute object files produced by the RA78K3 relocatable assembler package can be debugged with the appropriate NEC standalone in-circuit emulator. NEC emulator controller programs allow communication with the emulator through an RS-232C serial line. An emulator controller program can run on the IBM PC®, PC/XT®, or PC AT® under MS-DOS and is provided with the in-circuit emulator at no extra charge.

These emulator controller programs provide the following features:

- Uploading/downloading of object and symbol files
- Symbolic debugging capability
- Complete emulator control from host console
- On-line help facilities
- Macro command file capabilities
- Host system directory and file display
- Disk storage of debug session
- Storage of last 20 commands for recall

License Agreement

RA78K3 is sold under terms of a license agreement included with the assembler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users for one year.

Documentation

For further information on source program formats, assembler operation, and actual program examples, refer to the following manuals supplied with the RA78K3 package. Additional copies may be obtained from NEC Electronics Inc.

- RA78K Series Assembler Package, Language Manual
- RA78K Series Assembler Package, Operation Manual
- RA78K Series Structured Assembler Preprocessor, User's Manual
- ECC Generator, User's Manual

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

NEC

Section 7
Soldering

μPD78C00 Product Line; Soldering and Packaging Information

Part Number	Package	Package Drawing	Recommended Soldering Code (Note 1)
μPD78C14 Family			
78C10ACW	64-pin SDIP	P64C-70-750A,C	WS60-00-1
78C10AGF-3BE	64-pin QFP	P64GF-100-3B8, 3BE-1	IR30-00-1, VP15-00-1
78C10AGF(A)-3BE			
78C10AGQ-36	64-pin QUIP	P64GQ-100-36	WS60-00-1
78C10AGQ(A)-36			
78C10AL	68-pin PLCC	P68L-50A1-1	IR30-00-1, VP15-00-1
78C10AL(A)			
78C11ACW-xxx	64-pin SDIP	P64C-70-750A, C	WS60-00-1
78C11AGF-xxx-3BE	64-pin QFP	P64GF-100-3B8, 3BE-1	IR30-00-1, VP15-00-1, WS60-00-1
78C11AGF(A)-xxx-3BE			IR30-00-1, VP15-00-1
78C11AGQ-xxx-36	64-pin QUIP	P64GQ-100-36	WS60-00-1
78C11AGQ(A)-xxx-36			
78C11AGQ-xxx-37	64-pin QUIP (straight)	P64GQ-100-37	—
78C11AL-xxx	68-pin PLCC	P68L-50A1-1	IR30-00-1, VP15-00-1
78C11AL(A)-xxx			
78C12ACW-xxx	64-pin SDIP	P64C-70-750A,C	WS60-00-1
78C12AG-xxx-37	64-pin QUIP (straight)	P64GQ-100-37	—
78C12AG-xxx-36	64-pin QUIP	P64GQ-100-36	WS60-00-1
78C12AG(A)-xxx-36			
78C12AGF-xxx-3BE	64-pin QFP	P64GF-100-3B8, 3BE-1	IR30-00-1, VP15-00-1, WS60-00-1
78C12AL-xxx	68-pin PLCC	P68L-50A1-1	IR30-00-1, VP15-00-1
78C12AL(A)-xxx			
78C14AG-xxx-AB8	64-pin QFP	P64GC-80-AB8-2	IR30-107-1, VP15-107-1
78C14CW-xxx	64-pin SDIP	P64C-70-750A,C	WS60-00-1
78C14G-xxx-36	64-pin QUIP	P64GQ-100-36	WS60-00-1
78C14G-xxx-37	64-pin QUIP (straight)	P64GQ-100-37	—
78C14G-xxx-1B	64-pin QFP	P64G-100-12, 1B-1	IR30-107-1, VP15-107-1
78C14GF-xxx-3BE	64-pin QFP	P64GF-100-3B8, 3BE-1	IR30-107-1, VP15-107-1, WS60-107-1
78C14L-xxx	68-pin PLCC	P68L-50A1-1	IR30-00-1, VP15-00-1

μPD78C00 Product Line; Soldering and Packaging Information (cont)

Part Number	Package	Package Drawing	Recommended Soldering Code (Note 1)
μPD78C14 Family (cont)			
78CP14CW	64-pin SDIP	P64C-70-750A,C	WS60-00-1
78CP14G-36	64-pin QUIP	P64GQ-100-36	WS60-00-1
78CP14G-37	64-pin QUIP (straight)	P64GQ-100-37	WS60-00-1
78CP14GF-3BE	64-pin QFP	P64GF-100-3B8, 3BE-1	—
78CP14L	68-pin PLCC	P68L-50A1-1	VP15-162-1
78CP14DW	64-pin CER SDIP w/window	P64DW-70-750A	WS60-00-1
78CP14R	64-pin CER QUIP w/window	P64RQ-100-A	WS60-00-1
78CP14G(A)-36	64-pin QUIP	P64GQ-100-36	WS60-00-1
μPD78C18 Family			
78C17CW	64-pin SDIP	P64C-70-750A, C	WS60-00-1
78C17GF-3BE	64-pin QFP	P64GF-100-3B8, 3BE-1	IR30-107-1, VP15-107-1, WS60-00-1
78C17GF(A)-3BE			IR30-207-1, VP15-207-1, WS60-207-1
78C17GQ-36	64-pin QUIP	P64GQ-100-36	WS60-00-1
78C17GQ(A)-36			
μPD78C18CW-xxx	64-pin SDIP	P64C-70-750A, C	WS60-00-1
78C17GF-xxx-3BE	64-pin QFP	P64GF-100-3B8, 3BE-1	IR30-107-1, VP15-107-1, WS60-107-1
78C17GF(A)-xxx-3BE			
78C17GQ-xxx-36	64-pin QUIP	P64GQ-100-36	WS60-00-1
78C17GQ(A)-xxx-36			
78CP18CW	64-pin SDIP	P64C-70-750A, C	WS60-00-1
78CP18GF-3BE	64-pin QFP	P64GF-100-3B8, 3BE-1	IR30-107-1, VP15-107-1
78CP18GF(A)-3BE			
78CP18GQ-36	64-pin QUIP	P64GQ-100-36	WS60-00-1
78CP18GQ(A)-36			
78CP18DW	64-pin SDIP w/window	P64DW-70-750A	—
78CP18KB (Note 2)	64-pin ceramic LCC w/window	X64KW-100A-1	—

Notes:

(1) See soldering conditions table at the end of this section for further information on NEC's soldering codes.

(2) Not intended for soldering; if soldering code is not listed, contact NEC.

μPD78K0 Product Line; Soldering and Packaging Information

Part Number	Package	Package Drawing	Recommended Soldering Code (Note 1)
μPD78002 Family			
78001BCW-xxx	64-pin plastic shrink DIP	P64C-70-750A, C	WS60-00-1
78002BCW-xxx			
78001BGC-xxx-AB8	64-pin plastic QFP	P64GC-80-AB8-2	IR30-107-1, VP15-107-1, WS60-107-1
78002BGC-xxx-AB8			
μPD78002Y Family			
78001BYCW-xxx	64-pin plastic shrink DIP	P64C-70-750A, C	WS60-00-1
78002BYCW-xxx			
78001BYGC-xxx-AB8	64-pin plastic QFP	P64GC-80-AB8-2	IR30-107-1, VP15-107-1, WS60-107-1
78002BYGC-xxx-AB8			
μPD78014 Family			
78011BCW-xxx	64-pin plastic shrink DIP	P64-70-750A, C	WS60-00-1
78012BCW-xxx			
78013CW-xxx			
78014CW-xxx			
78P014CW			
78011BGC-xxx-AB8	64-pin plastic QFP	P64GC-80-AB8-2	IR30-107-1, VP15-107-1, WS60-107-1
78012BGC-xxx-AB8			
78013GC-xxx-AB8			
78014GC-xxx-AB8			
78P014GC	64-pin plastic QFP	P64GC-80-AB8-2	IR30-162-1, VP15-162-1
78P014DW	64-pin ceramic shrink DIP w/ window	P64DW-70-750A	Pin partial heating
μPD78014Y Family			
78011BYCW-xxx	64-pin plastic shrink DIP	P64C-70-750A, C	WS60-00-1
78012BYCW-xxx			
78013YCW-xxx			
78014YCW-xxx			
78P014YCW			
78011BYGC-xxx-AB8	64-pin plastic QFP	P64GC-80-AB8-2	IR30-107-1, VP15-107-1, WS60-107-1
78012BYGC-xxx-AB8			
78013YGC-xxx-AB8			
78014YGC-xxx-AB8			
78P014YGC	64-pin plastic QFP	P64GC-80-AB8-2	IR30-162-1, VP15-162-1
78P014YDW	64-pin ceramic shrink DIP w/ window	P64DW-70-750A	Pin partial heating

μPD78K0 Product Line; Soldering and Packaging Information (cont)

Part Number	Package	Package Drawing	Recommended Soldering Code (Note 1)
μPD78044 Family			
78042GF-xxx-3B9	80-pin plastic QFP	P80GF-80-3B9-1	IR35-207-1, WS60-207-1, VP15-207-1
78043GF-xxx-3B9			
78044GF-xxx-3B9			
78P044GF-3B9	80-pin plastic QFP	P80GF-80-3B9-1	(Note 2)
78P044KL-S	80-pin ceramic LCC w/window	X80KW-80A	Soldering not recommended
μPD78054 Family			
78052GC-xxx-3B9	80-pin plastic QFP	S80GC-65-SB9-1	Note 2
78053GC-xxx-3B9			
78054GC-xxx-3B9			
78P054GC-3B9			
78052GK-xxx-BE9	80-pin plastic TQFP	P80GK-50-BE9-1	Note 2
78053GK-xxx-BE9			
78054GK-xxx-BE9			
78P054GK-BE9			
78P054KK-T	80-pin ceramic LCC w/window	X80KW-65A	Note 2
μPD78064 Family			
78062GC-xxx-7EA	100-pin plastic QFP (14 x 14 mm)	P100GC-50-7EA	Note 2
78063GC-xxx-7EA			
78064GC-xxx-7EA			
78P064GC-7EA			
78062GF-xxx-3BA	100-pin plastic QFP (14 x 20 mm)	P100GF-65-3BA	Note 2
78063GF-xxx-3BA			
78064GF-xxx-3BA			
78P064GF-3BA			
78P064KL-T	100-pin ceramic LCC w/window (14 x 20 mm)	Note 3	Note 2

Notes:

(1) See soldering conditions table at the end of this section for further information on NEC's soldering codes.

(2) Please contact NEC Electronics.

(3) Under development

μPD78K2 Product Line; Soldering and Packaging Information

Part Number	Package	Package Drawing	Recommended Soldering Code (Note 1)
μPD78214 Family			
78212CW-xxx	64-pin SDIP	P64C-70-750A,C	WS60-00-1
78213CW			
78214CW-xxx			
78P214CW			
78212GC-xxx	64-pin plastic QFP	P64GC-80-AB8-2	IR30-162-1, VP15-162-1
78213GC			
78214GC-xxx			
78P214GC	64-pin plastic QFP	P64GC-80-AB8-2	IR30-162-1, VP15-162-1 (Note 2)
78212GJ-xxx	74-pin plastic QFP	S74GJ-100-5BJ-1	IR30-00-1, VP15-00-1
78213GJ			
78214GJ-xxx			
78P214GJ	74-pin plastic QFP	S74GJ-100-5BJ-1	IR30-107-1, VP15-107-1
78213G36	64-pin plastic QUIP	P64GQ-100-36	WS60-00-1
78214Gxxx36			
782P14GQ			
78213L	68-pin PLCC	P68L-50A1-1	VP15-162-1
78214L-xxx			
78P214L	68-pin PLCC	P68L-50A1-1	VP15-107-1
78P214DW	64-pin ceramic shrink DIP w/ window	P64DW-70-750A1	Pin partial heating
μPD78218A Family			
78217ACW	64-pin plastic SDIP	P64C-70-750A,C	WS60-00-1
78218ACW			
78P218ACW			
78217AGC	64-pin plastic QFP	P64C-80-AB8-2	IR30-162-1, VP15-162-1
78218AGC			
78P218AGC			
78P218ADW	64-pin ceramic SDIP w/window	P64DW-70-750A1	Pin partial heating
μPD78224 Family			
78220L	84-pin PLCC	P84L-50A3-1	VP15-162-1
78224L-xxx			
78P224L			
78220GJ-5BG	94-pin plastic QFP	S94GJ-80-5BG-1	IR30-107-1, VP15-107-1
78224GJ-xxx-5BG			
78P224GJ-5BG			Pin partial heating

μPD78K2 Product Line; Soldering and Packaging Information

Part Number	Package	Package Drawing	Recommended Soldering Code (Note 1)
μPD78238 Family			
78233GC	80-pin plastic QFP	S80GC-65-3B9-1	IR30-162-1, VP15-162-1
78234GC-xxx			
78237GC			
78238GC-xxx			
78P238GC			
78233GJ	94-pin plastic plastic QFP	S94GJ-80-5BG-1	IR30-107-1
78234GJ-xxx			
78237GJ			
78238GJ-xxx			
78P238GJ			
78233LQ	84-pin PLCC	P84L-50A3-1	VP15-107-1
78234LQ-xxx			
78237LQ			
78238LQ-xxx			
78P238LQ			
78P238KF	94-pin ceramic LCC w/window	X94KW-80A	Pin partial heating
μPD78244 Family			
78243CW	64-pin plastic SDIP	P64C-70-750A, C	WS60-00-1
78244CW-xxx			
78243GC-AB8	64-pin plastic QFP	P64GC-80-AB8-2	IR30-162-1, VP15-162-1
78244GC-xxx			

Notes:

(1) See soldering conditions table at the end of this section for further information on NEC's soldering codes.

(2) This soldering method is not applicable to the "K" specification product.

μPD78K3 Product Line; Soldering and Packaging Information

Part Number	Package	Package Drawing	Recommended Soldering Code (Note 1)
μPD78312A Family			
78310ACW	64-pin plastic shrink DIP	P64C-70-750A, C	—
78312ACW-xxx			
78P312ACW			
78310AGF-3BE	64-pin plastic QFP	P64GF-100-3B8, 3BE-1	IR30-162-1, VP15-162-1
78312AGF-xxx-3BE			
78P312AGF-3BE			
78310AGQ-36	64-pin plastic QUIP	P64GQ-100-36	IR30-162-1, VP15-162-1
78312AGQ-xxx-36			
78P312AGQ-36			
78310AL	68-pin plastic PLCC	P68L-50A1-1	IR30-00-1, VP15-162-1
78312AL-xxx			
78P312AL			VP15-00-1
78P312ADW	64-pin ceramic shrink DIP w/ window (350-mil)	P64DW-70-750A	—
78P312AR	64-pin ceramic QUIP w/window	P64RQ-100-A	—
μPD78322 Family			
78320GF	80-pin plastic QFP	P80GF-80-3B9-1	IR30-162-1
78320GF(A)			
78320GF(A1)			
78320GF(A2)			
78320L	68-pin PLCC	P68L-50A1-1	IR30-00-1, VP15-00-1
78320L(A)			
78320L(A1)			
78320L(A2)			
78322GF-xxx	80-pin plastic QFP	P80GF-80-3B9-1	IR30-162-1
78322GF(A)-xxx			
78322GF(A1)-xxx			
78322GF(A2)-xxx			
78322L-xxx	68-pin PLCC	P68L-50A1-1	IR30-00-1, VP15-00-1
78322L(A)-xxx			VP15-00-1
78322L(A1)-xxx			VP15-00-1
78322L(A2)-xxx			VP15-00-1
78P322GF	80-pin plastic QFP	P80GF-80-3B9-1	—
78P322L	68-pin PLCC	P68L-50A1-1	—
78P322KE	80-pin ceramic LCC with window	X80KW-80A	—
78P322KC	68-pin ceramic LCC with window	X68KW-50A	—

μPD78K3 Product Line; Soldering and Packaging Information

Part Number	Package	Package Drawing	Recommended Soldering Code (Note 1)
μPD78352 Family			
78350GC-3BE	64-pin plastic QFP (3.0-mm height)	P64GC-80-3BE	IR30-107-1 VP15-107-1 WS60-107-1
78P352AG-xxx-22	64-pin plastic QFP (1.7-mm height)	P64G-80-22-1	(Note 2)
78P352G-22			IR30-107-2 VP15-107-2
78P352KK	64-pin ceramic LCC with window	X80KW-80B	Not intended for soldering
μPD78356 Family			
78355GC-7EA	100-pin plastic QFP	P100GC-50-7EA	(Note 2)
78356GC-xxx-7EA			
78P356GC-7EA			
78P356K	120-pin ceramic LCC with window	X120KW-80A	(Note 2)

Note:

(1) See soldering conditions table at the end of this section for further information on NEC's soldering codes.

(2) Please contact NEC Electronics.

Soldering Conditions

Method (1)	Code (2)	Soldering Conditions	Exposure Limit (3)
Infrared reflow	IR30-00-1	Package peak temp: 230°C Time: 30 sec max (210°C min)	No limit
	IR30-107-1 IR30-107-2		Max no. of days: 7 (thereafter, 10 hours baking at 125°C is required)
	IR30-162-1		Max no. of days: 2 (thereafter, 16 hours baking at 125°C is required)
	IR30-207-1		Max no. of days: 7 (thereafter, 20 hours baking at 125°C is required)
	IR35-207-1	Package peak temp: 235°C Time: 30 sec max (210°C min)	Max no. of days: 7 (thereafter, 20 hours baking at 125°C is required)
Vapor phase	VP15-00-1	Package peak temp: 215°C Time: 40 sec max (200°C min)	No limit
	VP15-107-1 VP15-107-2		Max no. of days: 7 (thereafter, 10 hours baking at 125°C is required)
	VP15-162-1		Max no. of days: 2 (thereafter, 16 hours baking at 125°C is required)
	VP15-207-1		Max no. of days: 7 (thereafter, 20 hours baking at 125°C is required)
Wave soldering	WS60-00-1	Solder bath temp: 260°C max Time: 10 sec max	No limit
	WS60-107-1		Max no. of days: 7 (thereafter, 10 hours baking at 125°C is required)
	WS60-207-1		Max no. of days: 7 (thereafter, 20 hours baking at 125°C is required)
Pin partial heating (SDIP)		Temperature: 260°C max Time: 10 sec max (per device side)	
Pin partial heating (QFP)		Temperature: 300°C max Time: 3 sec max (per device side)	

Notes:

(1) Do not use different soldering methods together. However, on all devices the pin partial heating soldering method can be used alone or in combination with other soldering methods.

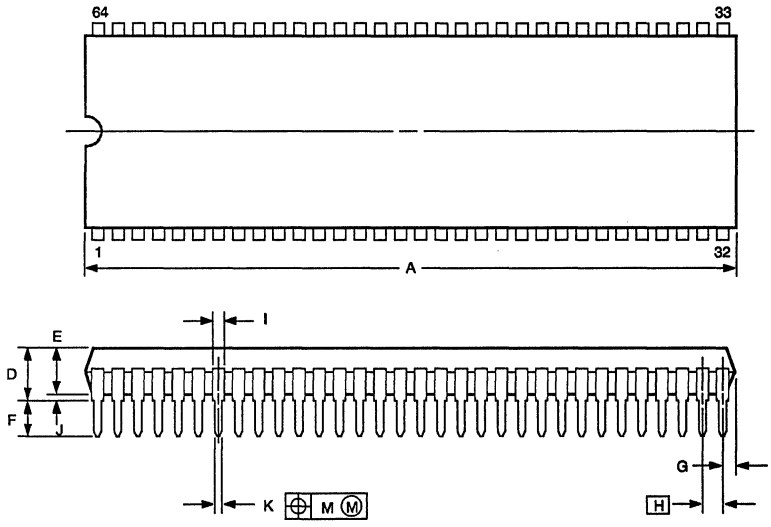
(2) The maximum number of soldering operations is one or two as indicated by the last digit of the soldering code: -1 or -2.

(3) Maximum no. of days refers to the number of days after unpacking the dry pack. Storage conditions are 25°C and 65% RH max.

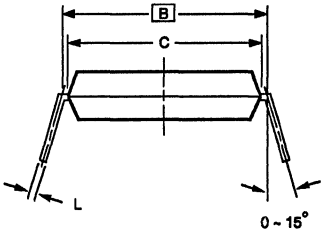
NEC

Section 8
Package Drawings

64-Pin Plastic Shrink DIP (P64C-70-750A, C)



Item	Millimeters	Inches
A	58.68 max	2.310 max
B	19.05 (TP)	.750 (TP)
C	17.0	.669
D	5.08 max	.200 max
E	4.31 max	.170 max
F	3.2 ±0.3	.126 ±.012
G	1.78 max	.070 max
H	1.778 (TP)	.070 (TP)
I	0.9 min	.035 min
J	0.51 min	.020 min
K	0.50 ±0.10	.020 ±.004
L	$0.25^{+0.10}_{-0.05}$	$.010^{+.004}_{-.002}$
M	0.17	.007



P64C-70-750A, C

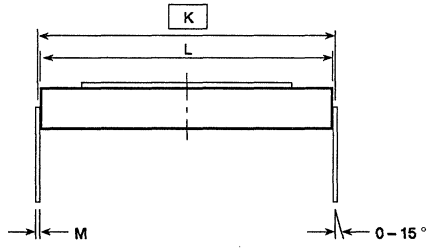
83YL-5560B (5/93)

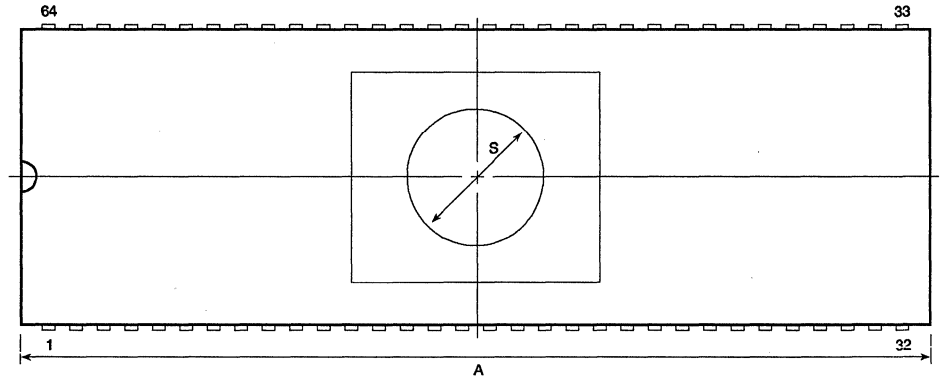
8

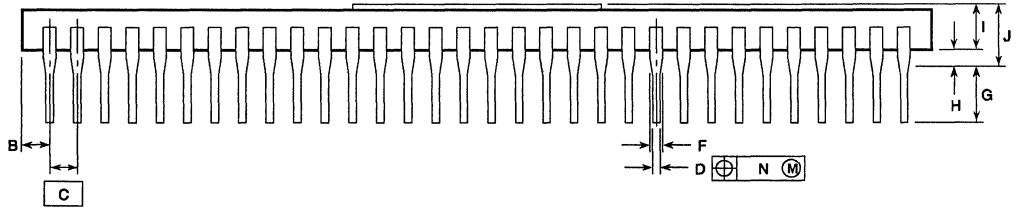
64-Pin Ceramic Shrink DIP (P64DW-70-750A)

Item	Millimeters	Inches
A	58.68 max	2.310 max
B	1.78 max	.070 max
C	1.778 (TP)	.070 (TP)
D	0.46 ± 0.05	.018 ± .002
F	0.8 min	.031 min
G	3.5 ± 0.3	.138 ± .012
H	1.0 min	.039 min
I	3.0	.118
J	5.08 max	.200 max
K*	19.05 (TP)	.750 (TP)
L	18.8	.740
M	0.25 ± 0.05	.010 $^{+.002}_{-.003}$
N	0.25	.010
S	8.89 dia	.350 dia

* Item K to center of leads when formed parallel.







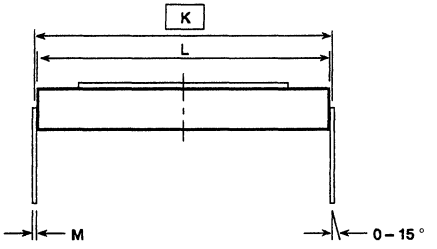
P64DW-70-750A

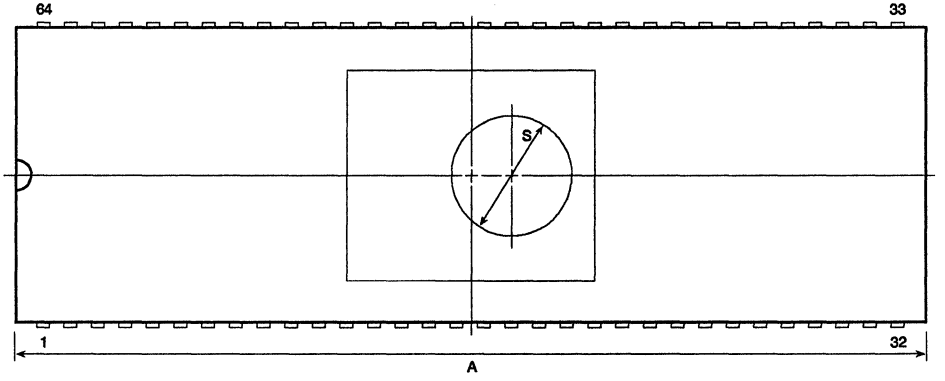
49NR-509B (5/93)

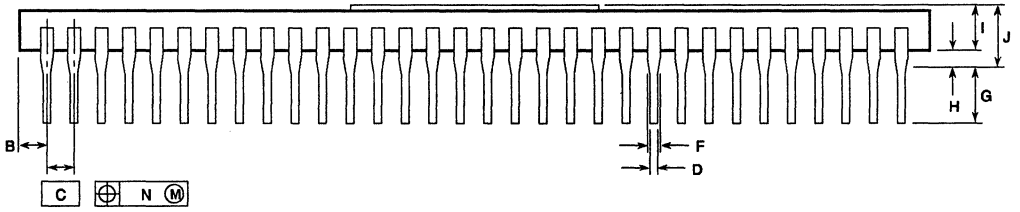
64-Pin Ceramic Shrink DIP (P64DW-70-750A1)

Item	Millimeters	Inches
A	58.68 max	2.310 max
B	1.78 max	.070 max
C	1.778 (TP)	.070 (TP)
D	0.46 ± 0.05	.018 ± .002
F	0.8 min	.031 min
G	3.5 ± 0.3	.138 ± .012
H	1.0 min	.039 min
I	3.0	.118
J	5.08 max	.200 max
K*	19.05 (TP)	.750 (TP)
L	18.8	.740
M	0.25 ± 0.05	.010 $^{+.002}_{-.003}$
N	0.25	.010
S	7.62 dia	.300 dia

* Item K to center of leads when formed parallel.







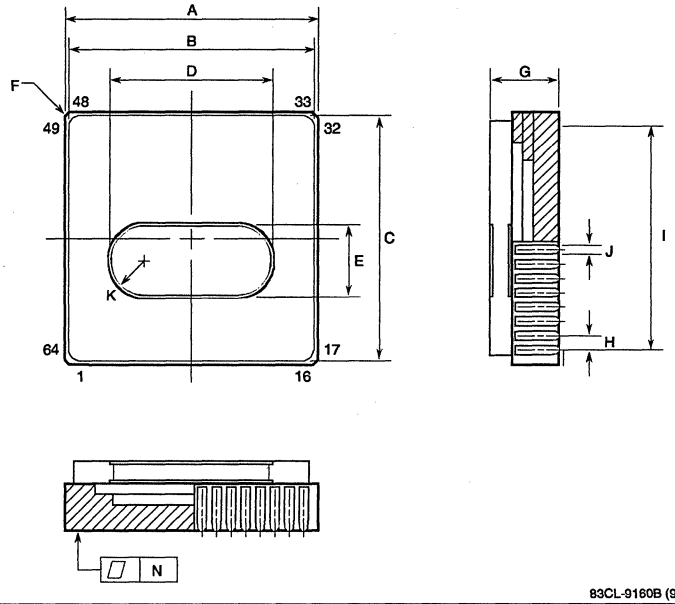
P64DW-70-750A1

49NR-691B (2/90)

8

64-Pin Ceramic LCC w/window (X80KW-80B)

Item	Millimeters	Inches
A	14.00 ± 0.18	.551 ± .007
B	13.60 ± 0.15	.535 ± .006
C	13.60 ± 0.15	.535 ± .006
D	9.0	.354 Typ.
E	4.0	.157 Typ.
F	C 0.3	C .012
G	3.185 ± 0.371	.125 ± .015
H	0.8 ± 0.1	.031 ± .004
I	12.0 ± 0.15	.472 ± .006
J	0.51 ± 0.1	.020 ± .004
K	R 2.0	R .079
N	0.08	.003

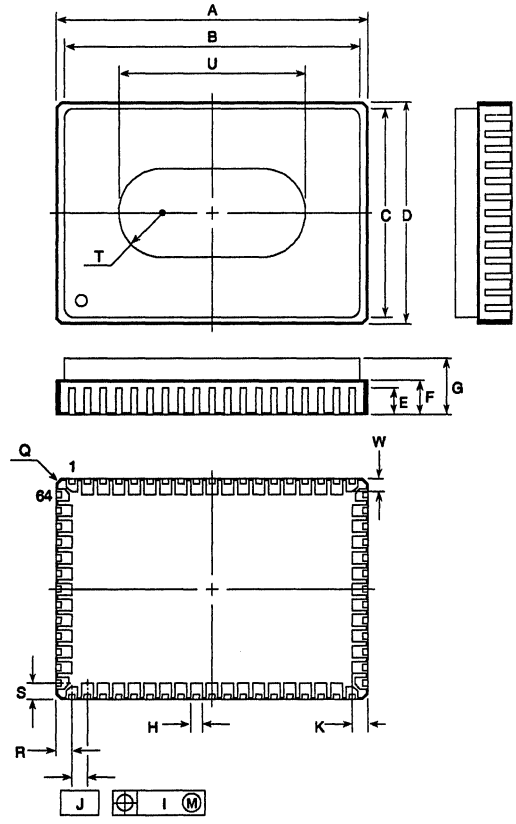


X80KW-80B

83CL-9160B (9/93)

64-Pin Ceramic LCC w/window (X64KW-100A-1)

Item	Millimeters	Inches
A	20.0 ± 0.4	.787 ± .016
B	19.0	.748
C	13.2	.520
D	14.0 ± 0.4	.550 ± .016
E	1.64	.065
F	2.14	.084
G	3.556 max	.140 max
H	0.70 ± 0.10	.028 ± .004
I	0.1	.004
J	1.0 (TP)	.039 (TP)
K	1.0 ± 0.2	.039 ± .008
Q	0.25 cor	.010 cor
R	1.0	.039
S	1.0	.039
T	3.0 rad	.118 rad
U	12.0	.472
W	0.8 ± 0.2	.031 ± .008



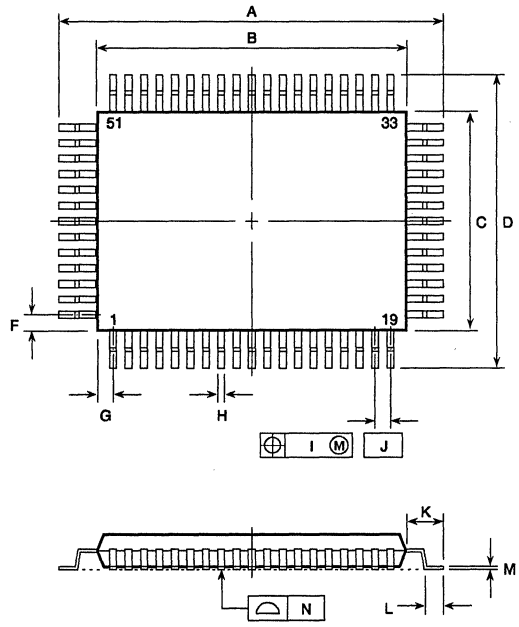
X64KW-100A-1

49NR-699B (5/90)

8

64-Pin Plastic QFP (P64G-100-12, 1B-1)

Item	Millimeters	Inches
A	24.7 ± 0.4	.972 $^{+.017}_{-.016}$
B	20.0 ± 0.2	.795 $^{+.009}_{-.008}$
C	14.0 ± 0.2	.551 $^{+.009}_{-.008}$
D	18.7 ± 0.4	.736 ± .016
F	1.0	.039
G	1.0	.039
H	0.40 ± 0.10	.016 $^{+.004}_{-.005}$
I	0.20	.008
J	1.0 (TP)	.039 (TP)
K	2.35 ± 0.2	.093 $^{+.008}_{-.009}$
L	1.2 ± 0.2	.047 $^{+.009}_{-.008}$
M	0.15 $^{+0.10}_{-0.05}$	.006 $^{+.004}_{-.003}$
N	0.15	.006
P	2.05 $^{+0.2}_{-0.1}$	.081 $^{+.008}_{-.005}$
Q	0.1 ± 0.1	.004 ± .004
S	2.45 max	.096 max



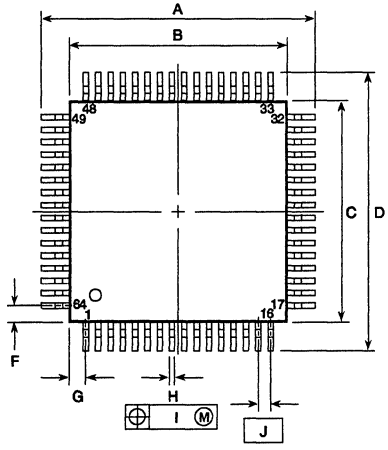
P64G-100-12, 1B-1

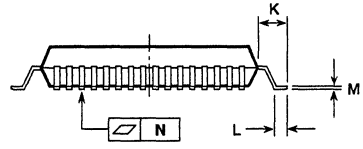
49NR-543B (5/93)

64-Pin Plastic QFP (P64GC-80-AB8-2)

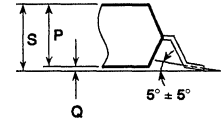
Item	Millimeters	Inches
A	17.6 ± 0.4	.693 ± .016
B	14.0 ± 0.2	.551 $^{+.009}_{-.008}$
C	14.0 ± 0.2	.551 $^{+.009}_{-.008}$
D	17.6 ± 0.4	.693 ± .016
F	1.0	.039
G	1.0	.039
H	0.35 ± 0.10	.014 $^{+.004}_{-.005}$
I	0.15	.006
J	0.8 (TP)	.031 (TP)
K	1.8 ± 0.2	.071 ± .008
L	0.8 ± 0.2	.031 $^{+.009}_{-.008}$
M	0.15 $^{+0.10}_{-0.05}$	.006 $^{+.004}_{-.003}$
N	0.15	.006
P	2.55	.100
Q	0.1 ± 0.1	.004 ± .004
S	2.85 max	.112 max

Note:
Each lead centerline is located within 0.15 mm (.006 inch) of its true position (TP) at maximum material condition.





Enlarged detail of lead end

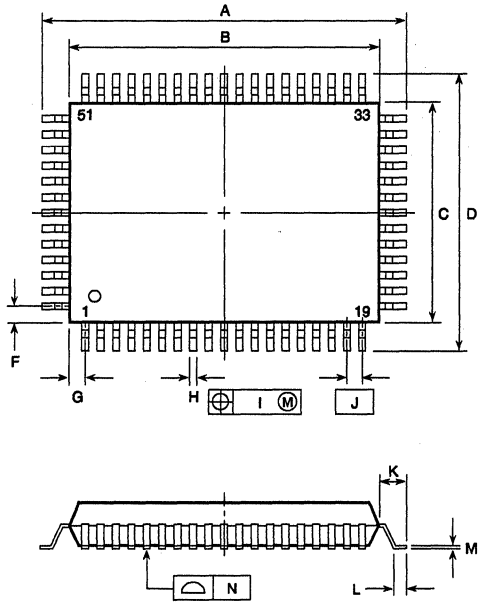


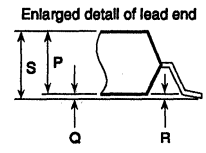
P64GC-80-AB8-2

49NR-669B (5/93)

64-Pin Plastic QFP (P64GF-100-3B8, 3BE-1)

Item	Millimeters	Inches
A	23.6 ± 0.4	.929 ± .016
B	20.0 ± 0.2	.795 $^{+.009}_{-.008}$
C	14.0 ± 0.2	.551 $^{+.009}_{-.008}$
D	17.6 ± 0.4	.693 ± .016
F	1.0	.039
G	1.0	.039
H	0.40 ± 0.10	.016 $^{+.004}_{-.005}$
I	0.20	.008
J	1.0 (TP)	.039 (TP)
K	1.8 ± 0.2	.071 $^{+.008}_{-.009}$
L	0.8 ± 0.2	.031 $^{+.009}_{-.008}$
M	0.15 $^{+0.10}_{-0.05}$	.006 $^{+.004}_{-.003}$
N	0.15	.006
P	2.7	.106
Q	0.1 ± 0.1	.004 ± .004
R	0.1 ± 0.1	.004 ± .004
S	3.0 max	.119 max



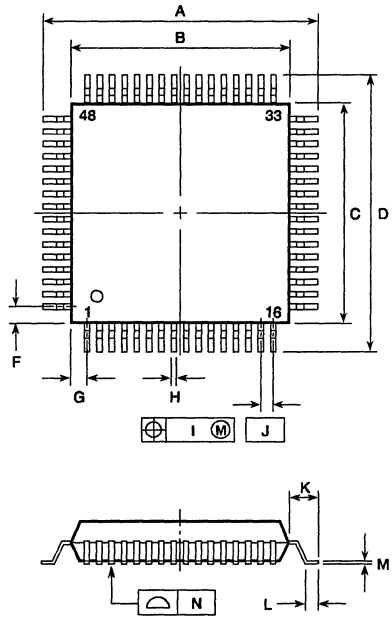


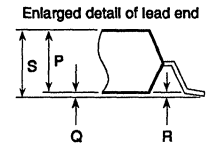
P64GF-100-3B8, 3BE-1

49NR-599B (6/91)

64-Pin Plastic QFP (3.0-mm height) (P64GC-80-3BE)

Item	Millimeters	Inches
A	17.6 ± 0.4	.693 ± .016
B	14.0 ± 0.2	.551 $^{+.009}_{-.008}$
C	14.0 ± 0.2	.551 $^{+.009}_{-.008}$
D	17.6 ± 0.4	.693 ± .016
F	1.0	.039
G	1.0	.039
H	0.35 ± 0.10	.014 $^{+.004}_{-.005}$
I	0.15	.006
J	0.8 (TP)	.031 (TP)
K	1.8 ± 0.2	.071 ± .008
L	0.8 ± 0.2	.031 $^{+.009}_{-.008}$
M	0.15 $^{+0.10}_{-0.05}$	.006 $^{+.004}_{-.003}$
N	0.15	.006
P	2.7	.106
Q	0.1 ± 0.1	.004 ± .004
R	0.1 ± 0.1	.004 ± .004
S	3.0 max	.118 max





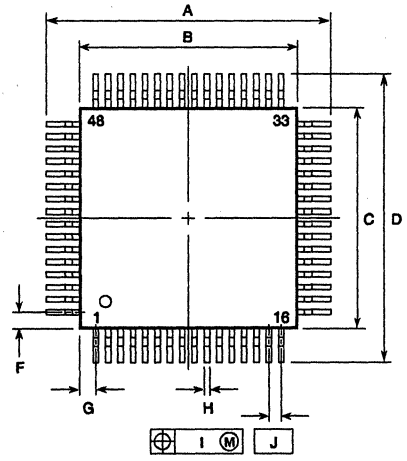
P64GC-80-3BE

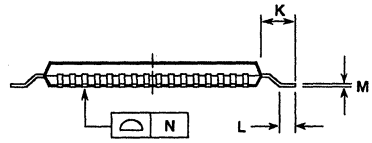
83CL-9159B (12/92)

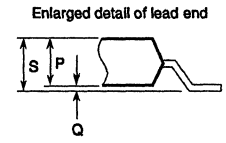
8

64-Pin Plastic QFP (1.7-mm height) (P64G-80-22-1)

Item	Millimeters	Inches
A	18.4 ± 0.4	.724 $^{+.017}_{-.016}$
B	14.0 ± 0.2	.551 $^{+.009}_{-.008}$
C	14.0 ± 0.2	.551 $^{+.009}_{-.008}$
D	18.4 ± 0.4	.724 $^{+.017}_{-.016}$
F	1.0	.039
G	1.0	.039
H	0.35 ± 0.10	.014 $^{+.004}_{-.005}$
I	0.15	.006
J	0.8 (TP)	.031 (TP)
K	2.2 ± 0.2	.087 $^{+.008}_{-.009}$
L	1.0 ± 0.2	.039 $^{+.009}_{-.008}$
M	0.15 $^{+0.10}_{-0.05}$	.006 $^{+.004}_{-.003}$
N	0.15	.006
P	1.5 ± 0.1	.059 ± .004
Q	0.0 ± 0.1	.000 ± .004
S	1.7 max	.067 max







P64G-80-22-1

49NR-670B (1/90)

64-Pin Ceramic QFP for Engineering Samples

Item	Millimeters	Inches
A	14.20	.559
B	12.0	.472
C	18.0	.708
D	20.0	.787
H	0.40	.016
J	1.0	.039
N	0.15	.006
S	2.25	.089

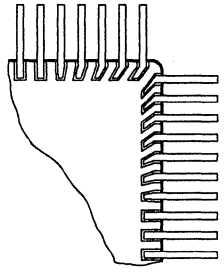
Notes:

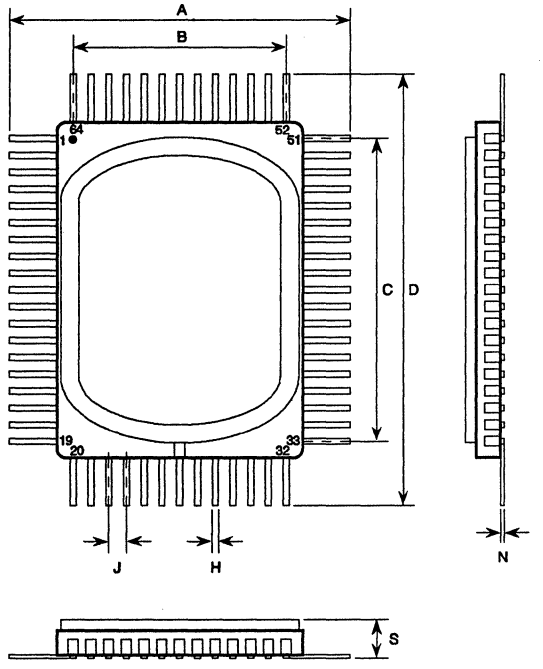
(1) The metal cover is connected to pin 26.

(2) The leads on the bottom surface are formed obliquely.

(3) The length of the leads is not defined since the cutting of the lead tips is not controlled during the manufacturing process.

Enlarged detail of bottom



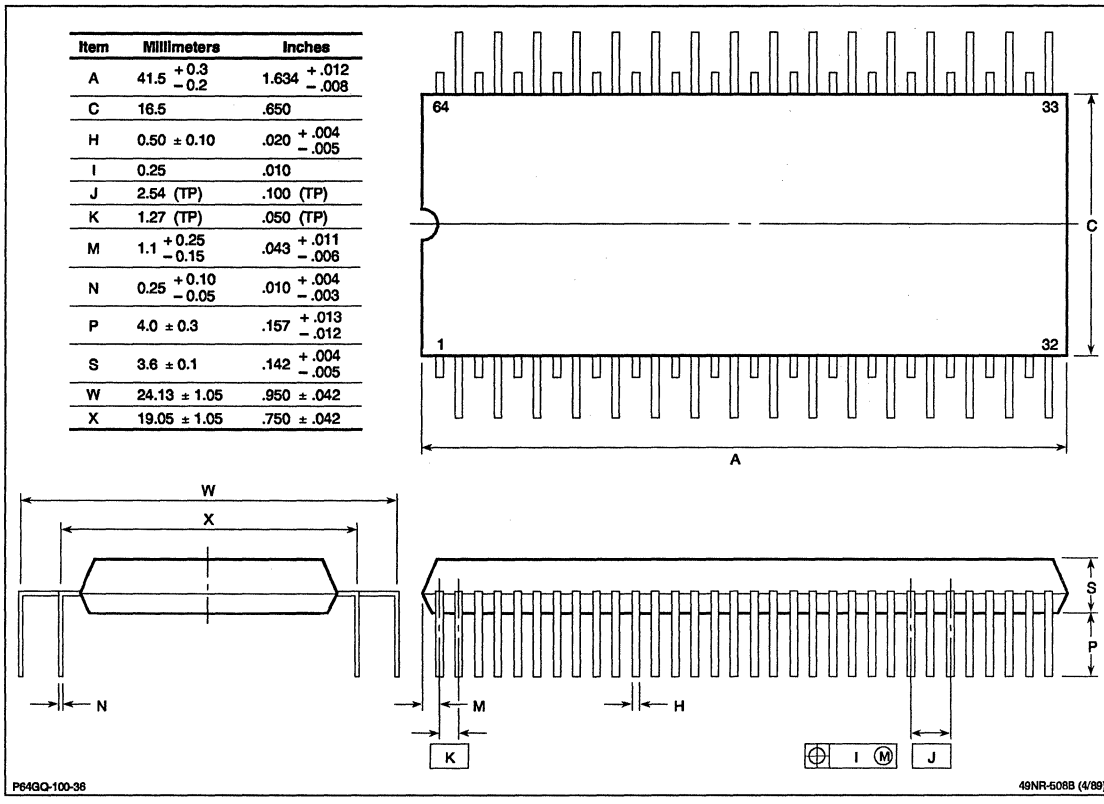


83YL-9269B (5/93)

8

64-Pin Plastic QUIP (P64GQ-100-36)

Item	Millimeters	Inches
A	41.5 $^{+0.3}_{-0.2}$	1.634 $^{+.012}_{-.008}$
C	16.5	.650
H	0.50 ± 0.10	.020 $^{+.004}_{-.005}$
I	0.25	.010
J	2.54 (TP)	.100 (TP)
K	1.27 (TP)	.050 (TP)
M	1.1 $^{+0.25}_{-0.15}$	.043 $^{+.011}_{-.006}$
N	0.25 $^{+0.10}_{-0.05}$	.010 $^{+.004}_{-.003}$
P	4.0 ± 0.3	.157 $^{+.013}_{-.012}$
S	3.6 ± 0.1	.142 $^{+.004}_{-.005}$
W	24.13 ± 1.05	.950 ± .042
X	19.05 ± 1.05	.750 ± .042



P64GQ-100-36B

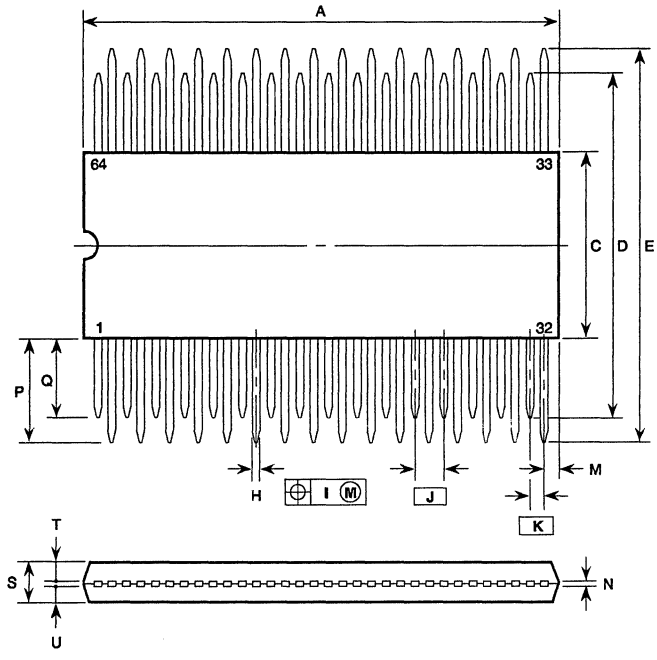
49NR-508B (4/89)

64-Pin Plastic QUIP (P64GQ-100-37)

Item	Millimeters	Inches
A	41.5 +0.3 −0.2	1.634 +.012 −.008
C	16.5	.650
D	30.0 ±0.4	1.181 ±.016
E	35.1 ±0.4	1.382 ±.016
H	0.50 ±0.10	.020 +.004 −.005
I	0.25	.010
J	2.54 (TP)	.100 (TP)
K	1.27 (TP)	.050 (TP)
M	1.1 +0.25 −0.15	.043 +.011 −.006
N	0.25 +0.10 −0.05	.010 +.004 −.003
P	9.3 ±0.2	.366 +.009 −.008
Q	6.75 ±0.2	.266 +.009 −.008
S	3.6 ±0.1	.142 +.004 −.005
T	1.8 ±0.1	.071 +.004 −.005
U	1.55 ±0.1	.061 ±.004

Note:

Each lead centerline is located within 0.25 mm (0.010 inch) of its true position (TP) at maximum material condition.



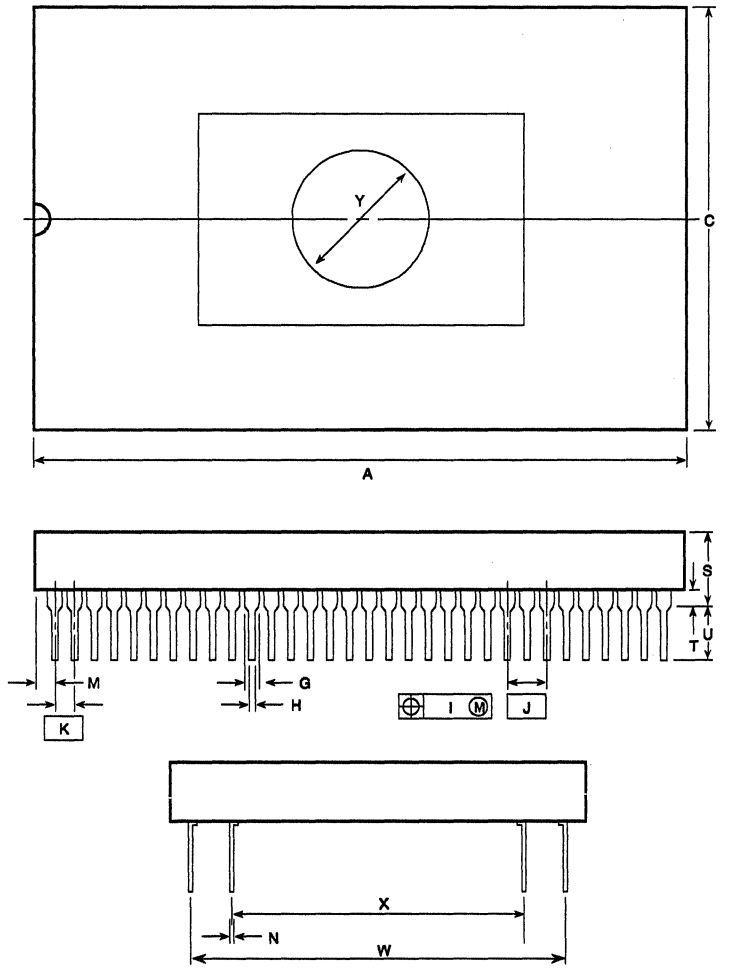
P64GQ-100-37

83YL-9268B (5/93)

8

64-Pin Ceramic QUIP w/window (P64RQ-100-A)

Item	Millimeters	Inches
A	41.91 max	1.650 max
C	26.67 ± 0.4	1.050 ± .016
G	0.92 min	.036 min
H	0.46 ± 0.05	.018 ± .002
I	0.25	.010
J	2.54 (TP)	.100 (TP)
K	1.27 (TP)	.050 (TP)
M	1.27 max	.050 max
N	0.25 ± 0.05	.010 $^{+.002}_{-.003}$
S	4.72 max	.186 max
T	1.0 min	.039 min
U	3.5 ± 0.3	.138 $^{+.012}_{-.013}$
W	24.13	.950
X	19.05	.750
Y	8.89 dia	.350 dia

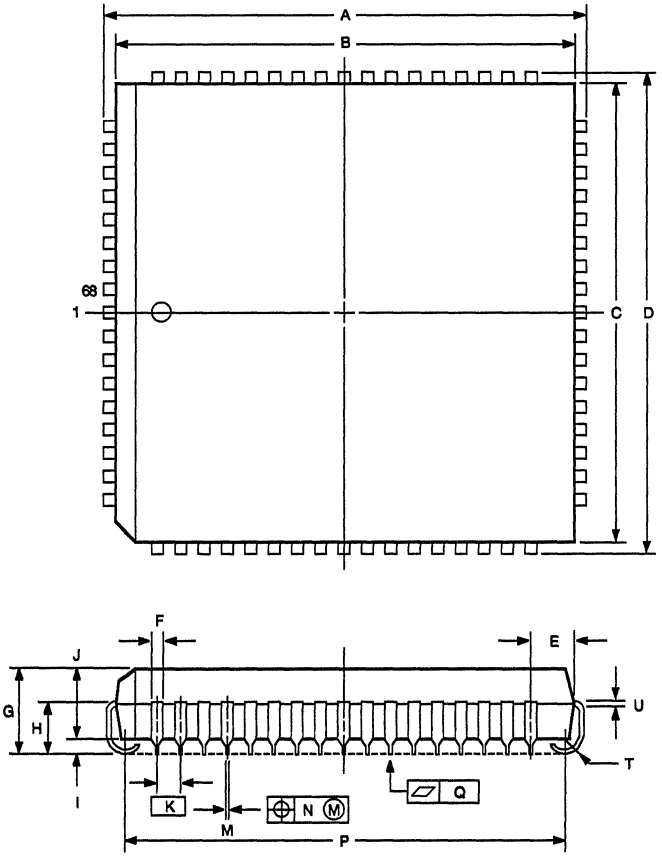


P64RQ-100-A

49NR-510B (9/93)

68-Pin PLCC (P68L-50A1-1)

Item	Millimeters	Inches
A	25.2 ±0.2	.992 ±.008
B	24.20	.953
C	24.20	.953
D	25.2 ±0.2	.992 ±.008
E	1.94 ±0.15	.076 $^{+.007}_{-.006}$
F	0.8	.024
G	4.4 ±0.2	.173 $^{+.009}_{-.008}$
H	2.8 ±0.2	.110 $^{+.009}_{-.008}$
I	0.9 min	.035 min
J	3.4	.134
K	1.27 (TP)	.050 (TP)
M	0.40 ±0.10	.016 $^{+.004}_{-.005}$
N	0.12	.005
P	23.12 ±0.20	.910 $^{+.009}_{-.008}$
Q	0.15	.006
T	0.8 radius	.031 radius
U	0.20 $^{+0.10}_{-0.05}$	.008 $^{+.004}_{-.002}$



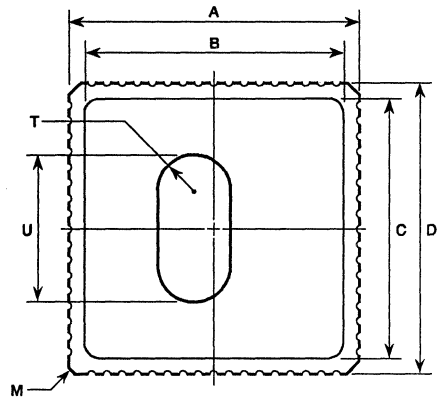
P68L-50A1-1

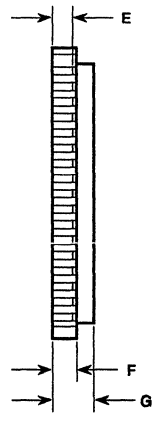
83YL-5581B (6/93)

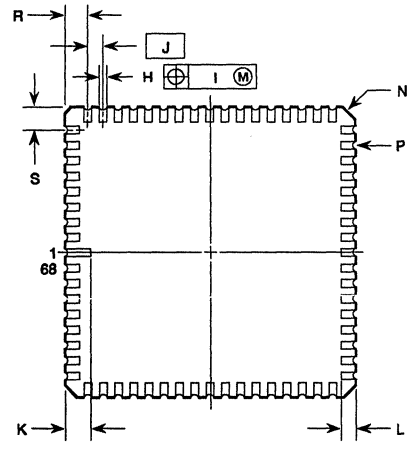
8

68-Pin Ceramic LCC w/window (X68KW-50A)

Item	Millimeters	Inches
A	24.13 ± 0.40	.95 ± .016
B	21.5	.846
C	21.5	.846
D	24.13 ± 0.40	.95 ± .016
E	1.65	.065
F	2.03	.08
G	3.50 max	.138 max
H	0.64 ± 0.10	.025 ± .004
I	0.12	.005
J	1.27	.05
K	2.16 ± 0.20	.085 ± .008
L	1.27 ± 0.20	.05 ± .008
M	C0.50	C.02
N	C1.02	C.04
P	R0.20	R.008
R	1.905	.075
S	1.905	.075
T	R3.0	R.118
U	12.0	.472





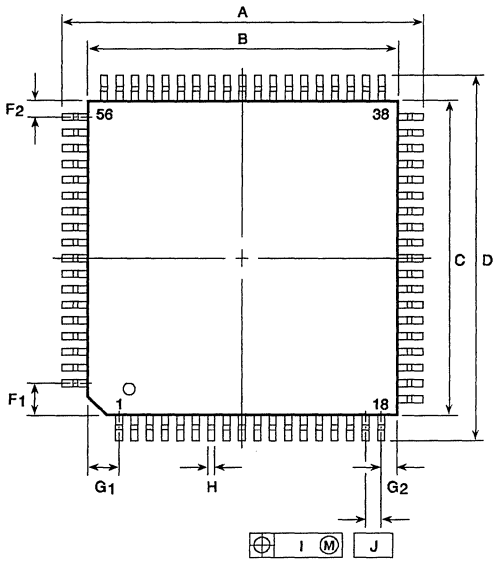


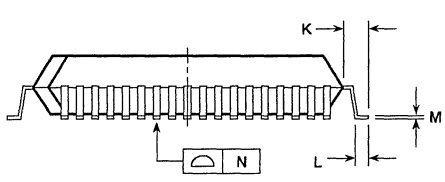
X68KW-50A

83FM-9390B

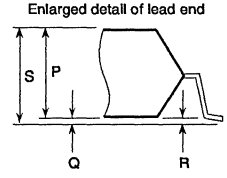
74-Pin Plastic QFP (Dwg No. S74GJ-100-5BJ-1)

Item	Millimeters	Inches
A	23.2 ± 0.4	.913 $^{+.017}_{-.016}$
B	20.0 ± 0.2	.787 $^{+.009}_{-.008}$
C	20.0 ± 0.2	.787 $^{+.009}_{-.008}$
D	23.2 ± 0.4	.913 $^{+.017}_{-.016}$
F_1	2.0	.079
F_2	1.0	.039
G_1	2.0	.079
G_2	1.0	.039
H	0.40 ± 0.10	.016 $^{+.004}_{-.005}$
I	0.20	.008
J	1.0 (TP)	.039 (TP)
K	1.6 ± 0.2	.063 ± .002
L	0.8 ± 0.2	.031 $^{+.009}_{-.008}$
M	0.15 $^{+0.10}_{-0.05}$	.006 $^{+.004}_{-.005}$
N	0.15	.006
P	3.7	.146
Q	0.1 ± 0.1	.004 ± .004
R	0.1 ± 0.1	.004 ± .004
S	4.0 max	.158 max





Enlarged detail of lead end



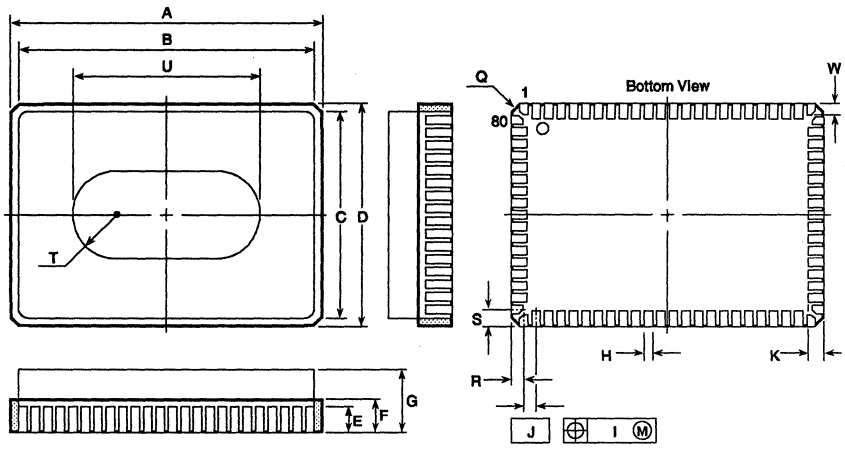
S74GJ-100-5BJ-1

49NR-347B (2/90)

8

80-Pin Ceramic LCC w/window (X80KW-80A)

Item	Millimeters	Inches
A	20.0 ± 0.4	.787 $^{+.017}_{-.016}$
B	19.0	.748
C	13.2	.520
D	14.2 ± 0.4	.559 ± .016
E	1.64	.065
F	2.14	.084
G	4.064 max	.160 max
H	0.51 ± 0.10	.020 ± .004
I	0.08	.003
J	0.8 (TP)	.031 (TP)
K	1.0 ± 0.2	.039 $^{+.009}_{-.008}$
Q	0.5 cor	.020 cor
R	0.8	.031
S	1.1	.043
T	3.0 rad	.118 rad
U	12.0	.472
W	0.75 ± 0.2	.030 $^{+.008}_{-.009}$

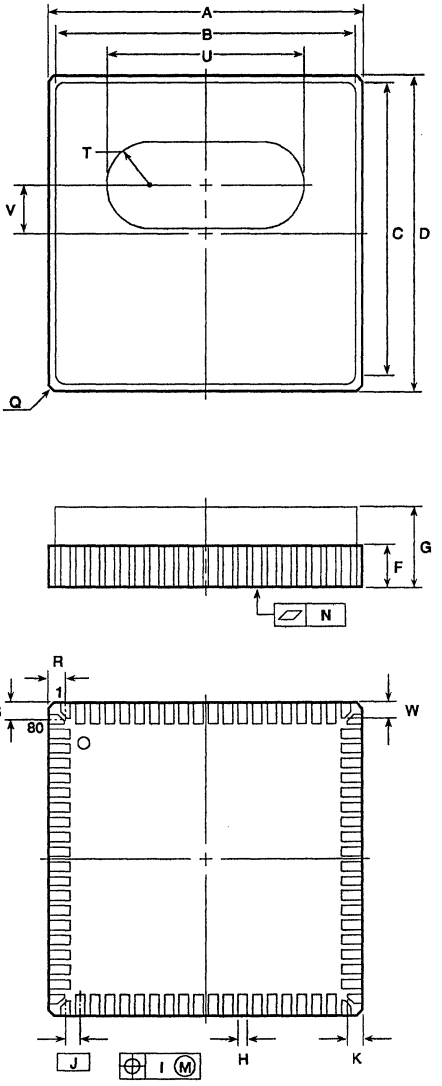


X80KW-80A

49NR-617B (11/89)

80-Pin Ceramic LCC w/window (X80KW-65A)

Item	Millimeters	Inches
A	14.0 ± 0.2	.551 ± .008
B	13.6	.535
C	13.6	.535
D	14.0 ± 0.2	.551 ± .008
F	1.84	.072
G	3.6 max	.142 max
H	0.45 ± 0.10	.018 $^{+0.004}_{-0.005}$
I	0.06	.003
J	0.65 (TP)	.024 (TP)
K	1.0 ± 0.15	.039 $^{+0.007}_{-0.006}$
N	0.1	.004
Q	0.3 cor	.012 cor
R	.825	.032
S	.825	.032
T	2.0 rad	.079 rad
U	9.0	.354
V	2.1	.083
W	0.75 ± 0.15	.030 $^{+0.006}_{-0.007}$



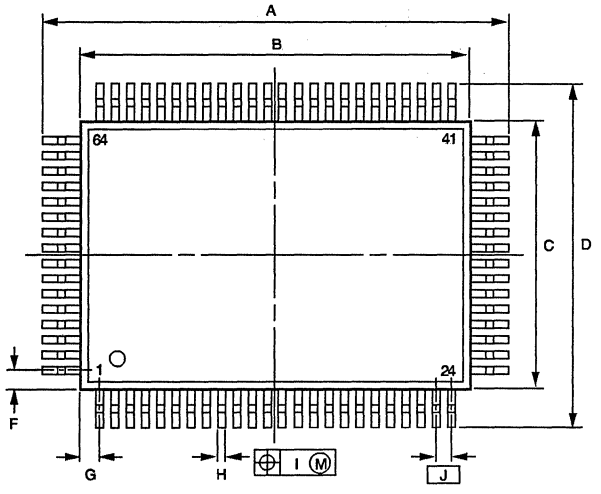
X80KW-65A

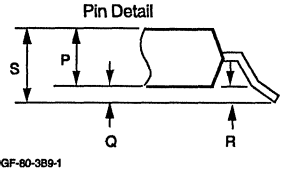
83YL-9617B (10/93)

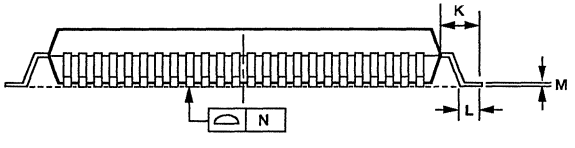
8

80-Pin Plastic QFP (P80GF-80-3B9-1)

Item	Millimeters	Inches
A	23.6 ±0.4	.929 ±.016
B	20.0 ±0.2	.787 +.009 −.008
C	14.0 ±0.2	.551 +.009 −.008
D	17.6 ±0.4	.693 ±.016
F	1.0	.039
G	0.8	.031
H	0.35 ±0.10	.014 +.004 −.005
I	0.15	.006
J	0.8 (TP)	.031 (TP)
K	1.8 ±0.2	.071 +.009 −.008
L	0.8 ±0.2	.031 +.009 −.008
M	0.15 +0.10 −0.05	.006 +.004 −.002
N	0.15	.006
P	2.7	.106
Q	0.1 ±0.1	.004 ±.004
R	0.1 ±0.1	.004 ±.004
S	3.0 max	.118 max





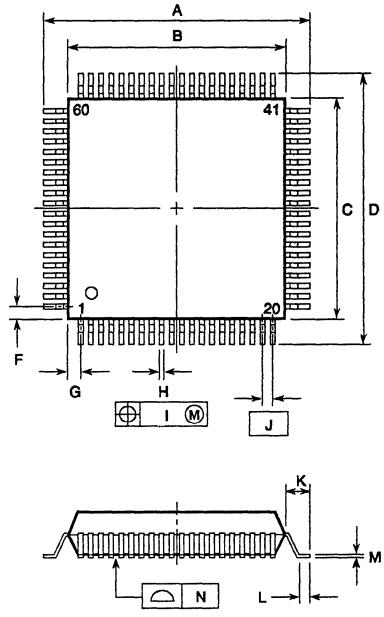


P80GF-80-3B9-1

83IH-5643B (10/91)

80-Pin Plastic QFP (S80GC-65-3B9-1)

Item	Millimeters	Inches
A	17.2 ± 0.4	.677 ± .016
B	14.0 ± 0.2	.551 $^{+.009}_{-.008}$
C	14.0 ± 0.2	.551 $^{+.009}_{-.008}$
D	17.2 ± 0.4	.677 ± .016
F	0.8	.031
G	0.8	.031
H	0.30 ± 0.10	.012 $^{+.004}_{-.005}$
I	0.13	.005
J	0.65 (TP)	.026 (TP)
K	1.6 ± 0.2	.063 ± .008
L	0.8 ± 0.2	.031 $^{+.009}_{-.008}$
M	0.15 $^{+0.10}_{-0.05}$	.006 $^{+.004}_{-.003}$
N	0.12	0.005
P	2.7	.106
Q	0.1 ± 0.1	.004 ± .004
R	0.1 ± 0.1	.004 ± .004
S	3.0 max	.119 max



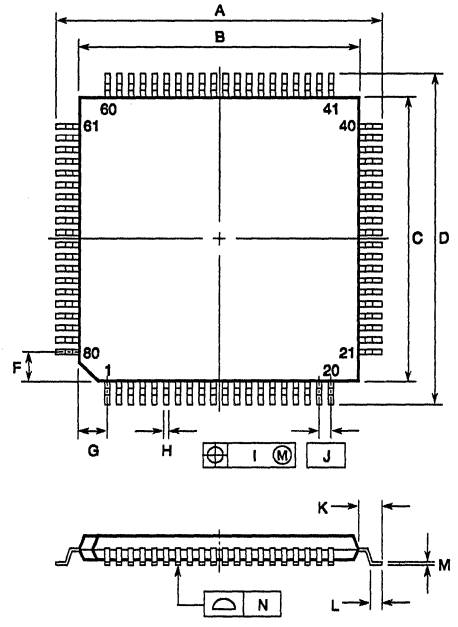
S80GC-65-3B9-1

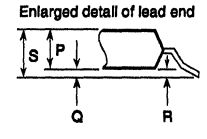
49NR-591B (10/93)

8

80-Pin Plastic TQFP (P80GK-50-BE9-1)

Item	Millimeters	Inches
A	14.0 ± 0.4	.551 ± .016
B	12.0 ± 0.2	.472 $^{+.009}_{-.008}$
C	12.0 ± 0.2	.472 $^{+.009}_{-.008}$
D	14.0 ± 0.4	.551 ± .016
F	1.25	.049
G	1.25	.049
H	0.20 ± 0.10	.008 ± .004
I	0.10	.004
J	0.5 (TP)	.020 (TP)
K	1.0 ± 0.2	.039 $^{+.009}_{-.008}$
L	0.5 ± 0.2	.020 $^{+.008}_{-.009}$
M	0.125 $^{+0.10}_{-0.05}$	.005 $^{+.004}_{-.002}$
N	0.10	.004
P	1.05	.041
Q	0.05 ± 0.05	.002 ± .002
R	0.05 ± 0.05	.002 ± .002
S	1.27 max	.050 max



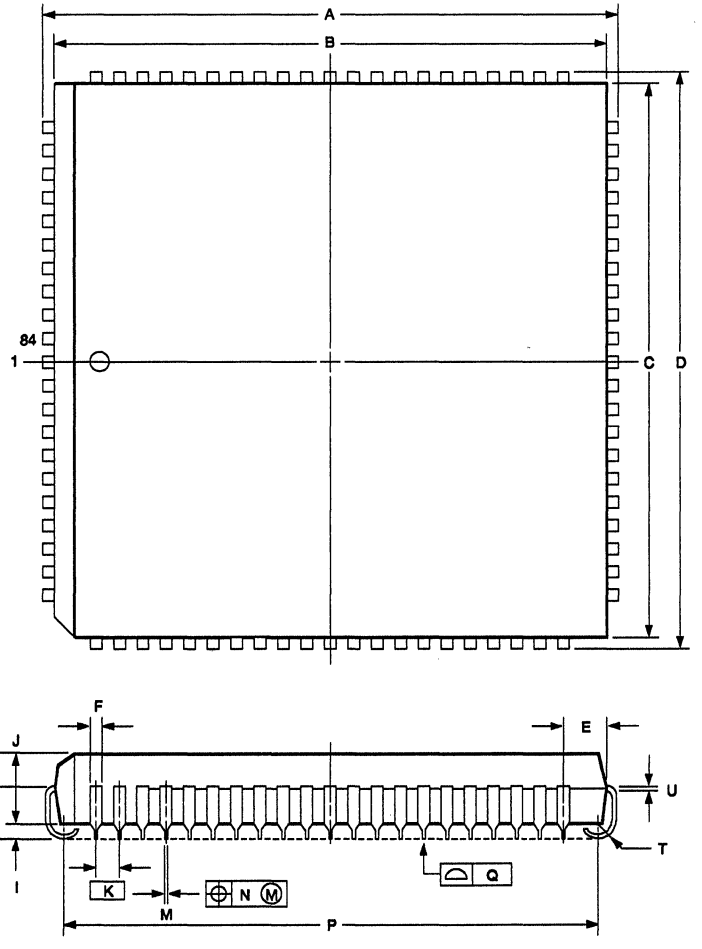


P80GK-50-BE9-1

49NR-728B (10/93)

84-Pin PLCC (P84L-50A3-1)

Item	Millimeters	Inches
A	30.2 ±0.2	1.189 ±.008
B	29.28	1.153
C	29.28	1.153
D	30.2 ±0.2	1.189 ±.008
E	1.94 ±0.15	.076 ±.006
F	0.6	.024
G	4.4 ±0.2	.173 ±.008
H	2.8 ±0.2	.110 ±.008
I	0.9 min	.035 min
J	3.4	.134
K	1.27 (TP)	.050 (TP)
M	0.40 ±0.10	.016 ±.004
N	0.12	.005
P	28.20 ±0.20	1.110 ±.008
Q	0.15	.006
T	0.8 radius	.031 radius
U	$0.20^{+0.10}_{-0.05}$	$.008^{+.004}_{-.002}$



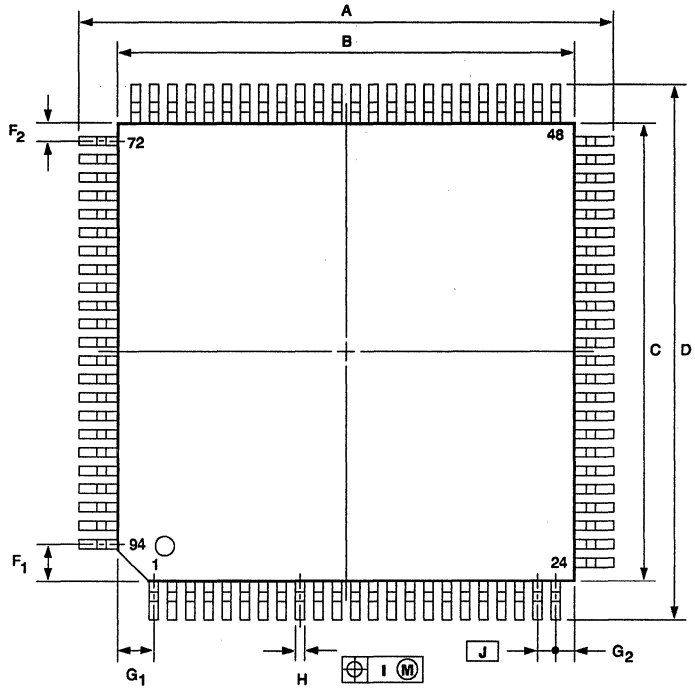
P84L-50A3-1

83YL-5806B (4/93)

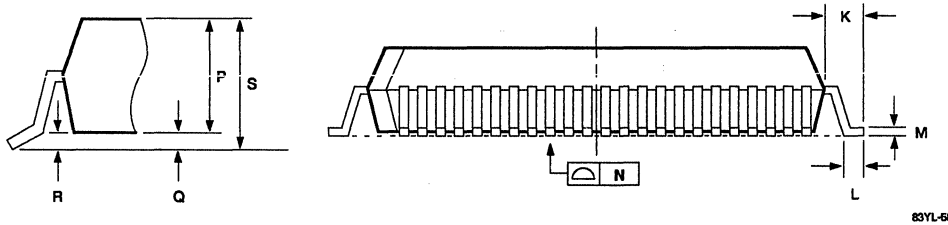
8

94-Pin Plastic QFP (S94GJ-80-5BG-1)

Item	Millimeters	Inches
A	23.2 ±0.4	.913 +.017 −.016
B	20.0 ±0.2	.787 +.009 −.008
C	20.0 ±0.2	.787 +.009 −.008
D	23.2 ±0.4	.913 +.017 −.016
F_1	1.6	.063
F_2	0.8	.031
G_1	1.6	.063
G_2	0.8	.031
H	0.35 ±0.10	.014 +.004 −.005
I	0.15	.006
J	0.8 (TP)	.031 (TP)
K	1.6 ±0.2	.063 ±.008
L	0.8 ±0.2	.031 +.009 −.008
M	0.15 +0.10 −0.05	.006 +.004 −.003
N	0.15	.006
P	3.7	.146
Q	0.1 ±0.1	.004 ±.004
R	0.1 ±0.1	.004 ±.004
S	4.0 max	.158 max



Detail of lead end

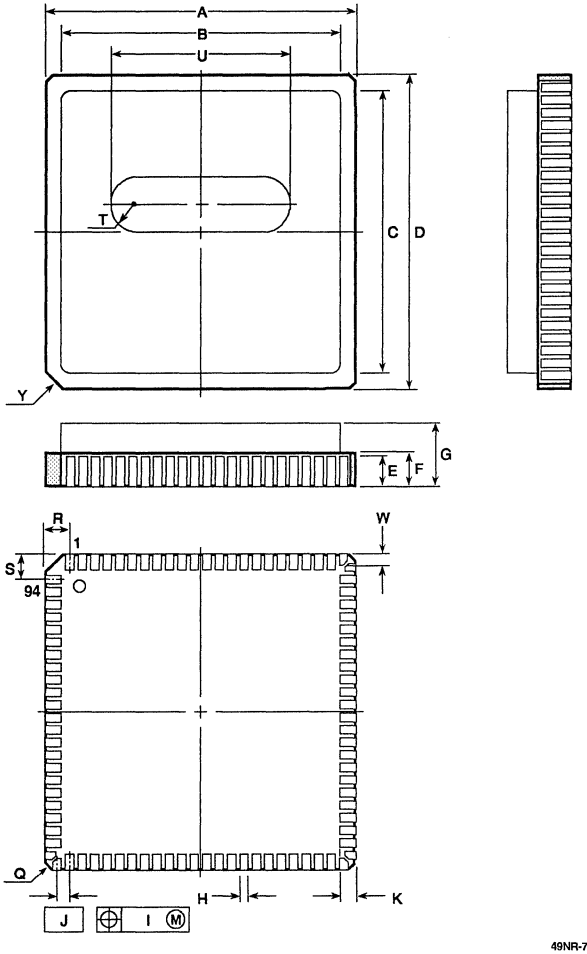


S94GJ-80-5BG-1

83YL-5810B (2/90)

94-Pin Ceramic LCC (X94KW-80A)

Item	Millimeters	Inches
A	20.0 ± 0.4	.787 ± .017
B	18.0	.709
C	18.0	.709
D	20.0 ± 0.4	.787 ± .017
E	1.94	.076
F	2.14	.084
G	4.064 max	.160 max
H	0.51 ± 0.10	.020 ± .004
I	0.08	.003
J	0.8 (TP)	.031 (TP)
K	1.0 ± 0.2	.039 ± .008
Q	0.3 cor	.012 cor
R	1.6	.063
S	1.6	.063
T	1.75 rad	.069 rad
U	11.5	.453
W	0.75 ± 0.2	.030 ± .008
Y	1.0 cor	.039 cor

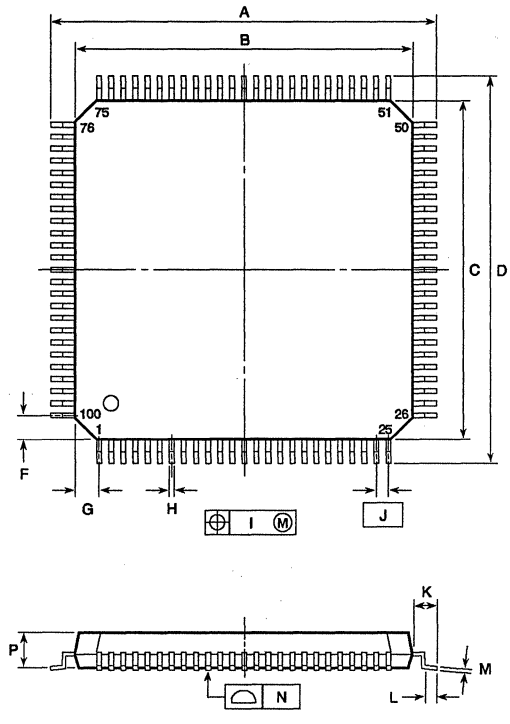


X94KW-80A

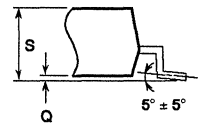
49NR-704B (3/90)

100-Pin Plastic QFP (P100GC-50-7EA)

Item	Millimeters	Inches
A	16.0 ± 0.4	.630 ± .016
B	14.0 ± 0.2	.551 $^{+.009}_{-.008}$
C	14.0 ± 0.2	.551 $^{+.009}_{-.008}$
D	16.0 ± 0.4	.630 ± .016
F	1.0	.039
G	1.0	.039
H	0.20 ± 0.10	.008 ± .004
I	0.08	.003
J	0.5 (T.P.)	.020 (T.P.)
K	1.0 ± 0.2	.039 $^{+.009}_{-.008}$
L	0.5 ± 0.2	.020 $^{+.008}_{-.009}$
M	0.15 ± 0.05	.006 ± .002
N	0.10	.004
P	1.45	.057
Q	0.1 ±0.1	.004 ± .004
S	1.7 max	.067 max



Enlarged detail of lead end

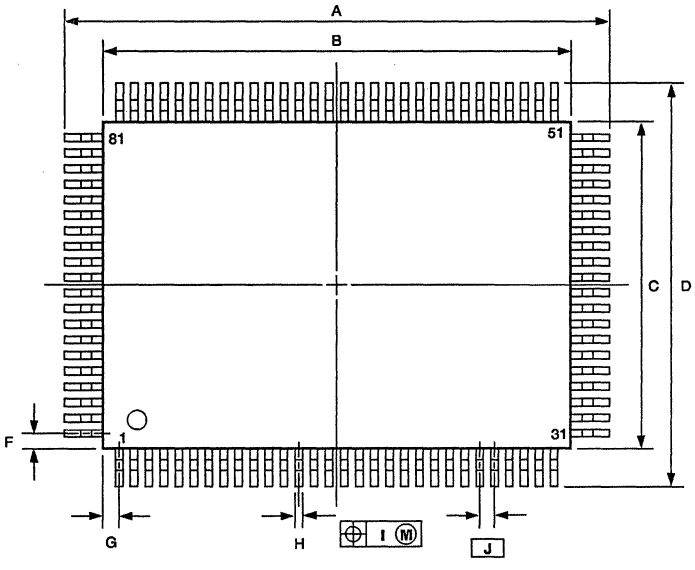


P100GC-50-7EA

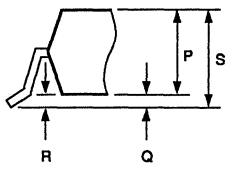
83RD-7598B (6/93)

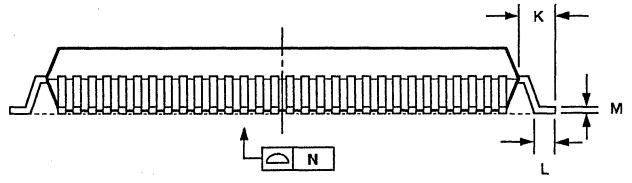
100-Pin Plastic QFP (P100GF-65-3BA)

Item	Millimeters	Inches
A	23.6 ±0.4	.929 ±.016
B	20.0 ±0.2	.787 $^{+.009}_{-.008}$
C	14.0 ±0.2	.551 $^{+.009}_{-.008}$
D	17.6 ±0.4	.693 ±.016
F	0.8	.031
G	0.6	.024
H	0.30 ±0.10	.012 $^{+.004}_{-.005}$
I	0.15	.006
J	0.65 (TP)	.026 (TP)
K	1.8 ±0.2	.071 $^{+.009}_{-.008}$
L	0.8 ±0.2	.031 $^{+.009}_{-.008}$
M	0.15 $^{+0.10}_{-0.05}$	.006 $^{+.004}_{-.002}$
N	0.15	.006
P	2.7	.106
Q	0.1 ±0.1	.004 ±.004
R	0.1 ±0.1	.004 ±.004
S	3.0 max	.118 max



Detail of lead end





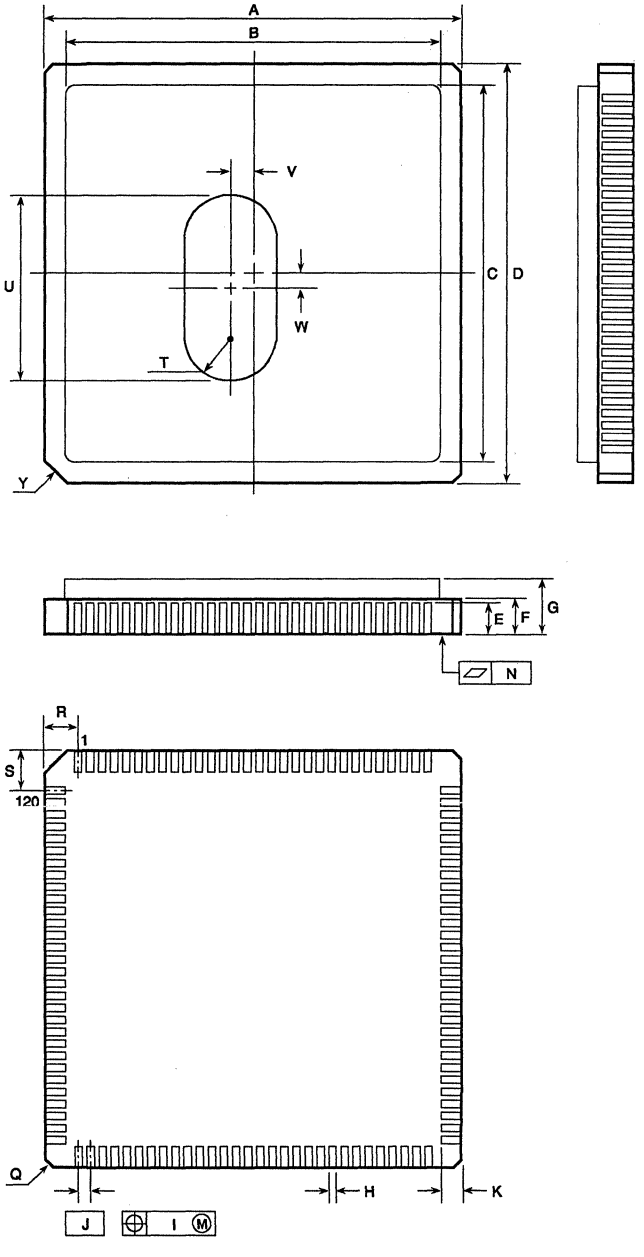
P100GF-65-3BA

83YL-5811B (9/93)

8

120-Pin Ceramic LCC (X120KW-80A)

Item	Millimeters	Inches
A	27.3 ± 0.27	1.075 ± .011
B	24.5	.965
C	24.5	.965
D	27.3 ± 0.27	1.075 ± .011
E	1.94	.076
F	2.14	.084
G	3.57 max	.141 max
H	0.51 ± 0.10	.020 ± .004
I	0.08	.003
J	0.8 (TP)	.031 (TP)
K	1.0 ± 0.15	.039 ± .006
N	0.10	.004
Q	0.3 cor	.012 cor
R	2.05	.081
S	2.05	.081
T	3.0 rad	.118 rad
U	12.0	.472
V	1.5	.060
W	1.0	.039
Y	1.0 cor	.039 cor



X120KW-80A

83YL-9409B (5/93)

NEC

NEC Electronics Inc.

Corporate Headquarters
475 Ellis Street
Mountain View, CA 94043
Telephone: 415-960-6000
Fax: 415-965-6130

For literature, call toll-free 8 a.m. to 4 p.m., Pacific time: 1-800-366-9782

50053-1