

### Description

The  $\mu$ PD77C25 and  $\mu$ PD77P25 Digital Signal Processors (DSP) are significant upgrades to the  $\mu$ PD7720—the original member of NEC's DSP family.  $\mu$ PD77C25 is the mask ROM version;  $\mu$ PD77P25 has an OTP ROM or a UVEPROM. All versions are CMOS and identical in function. Unless contextually excluded, references in this data sheet to 77C25 include 77P25

The 77C25 executes instructions twice as fast as the 77C20A/7720A. Additional instructions allow the 77C25 to execute common digital filter routines more efficiently and at more than twice the speed of a 7720 implementation.

In addition to doubled execution speed, the 77C25 has four times the instruction ROM space and twice the data ROM and RAM space of the 7720. Real savings are now possible, especially where one 77C25 can do the work of and replace two or more 7720s.

The external clock frequency (8.3 MHz maximum) remains the same as for 77C20A/7720A while the internal instruction execution speed is doubled. For most applications, the 77C25 is plug-in compatible with the 77C20A/7720A/77P20.

The feature that distinguishes digital signal processing chips from general-purpose microcomputers is the on-chip multiplier necessary for high-speed signal processing algorithms. The 77C25 multiplier is very sophisticated, especially for a low-cost DSP chip; both multiplier inputs can be loaded simultaneously from two separate memory areas. These loading operations are only two of nine operations that can occur during one 122-ns instruction cycle.

For a typical DSP filter application involving many successive multiplications, the 77C25 provides a new multiplication product for addition to a sum of products every 122 nanoseconds. Additionally, during the same instruction, memory data pointers are manipulated, and even a return from subroutine may be executed. Table 1 compares 77C25 with 77C20A.

### Features

- Low-power CMOS: 25 mA typical current use (77C25)
- Fast instruction execution: 122 ns with 8.192-MHz clock
- All instructions execute in one instruction cycle

- Drop-in compatible with 77C20A/7720A/77P20
  - 16-bit data word
  - Multioperation instructions for fast program execution: any part, any combination, or all of the following operations may constitute one instruction that executes in 122 ns.
    - Load one multiplier input
    - Load the other multiplier input
    - Multiply (automatic)
    - Load product to output registers (automatic)
    - Add product to accumulator
    - Move RAM column data pointer
    - Move RAM row pointer
    - Move data ROM pointer
    - Return from subroutine
  - Modified Harvard architecture with three separate memory areas
    - Instruction ROM (2048 x 24 bits)
    - Data ROM (1024 x 16 bits)
    - Data RAM (256 x 16 bits)
  - 16 x 16-bit multiplier; 31-bit product with every instruction
  - Dual 16-bit accumulators
  - External maskable interrupt
  - Four-level stack for subroutines and/or interrupt
  - Multiple I/O capabilities
    - Serial: 8 or 16-bit (244 ns/bit)
    - Parallel: 8 or 16-bit
    - DMA
  - Compatible with most microprocessors, including:
    - $\mu$ PD8080
    - $\mu$ PD8085
    - $\mu$ PD8086/88
    - $\mu$ PD780 (Z80®)
    - $\mu$ PD78xx family
  - Packages: 28-pin DIP, 32-pin SOP, 44-pin PLCC
  - Single + 5-volt power supply
- Z80 is a registered trademark of Zilog Corporation.

### Applications

- Portable telecommunications equipment
- Digital filtering
- High-speed data modems
- Fast Fourier transforms (FFT)

- Speech synthesis and analysis
- Dual-tone multifrequency (DTMF) transmitters/receivers
- Equalizers
- Adaptive control
- Numerical processing

**Performance Benchmarks**

- Second-order digital filter (biquad): 1.1 μs
- Sin/cos of angles: 2.58 μs
- μ-law or A-law to linear conversion: 0.24 μs
- FFT
  - 32-point complex: 0.35 ms
  - 64-point complex: 0.8 ms

**Ordering Information**

Part Number	Package	ROM	Operating Temperature Range
μPD77C25C-xxx	28-pin plastic DIP	Mask	-40 to +85°C
C25GW-xxx	32-pin SOP		
C25L-xxx	44-pin PLCC		
μPD77P25C	28-pin plastic DIP	OTP	-10 to +70°C
P25D	28-pin ceramic DIP	UVEPROM	
P25GW	32-pin SOP	OTP	
P25L	44-pin PLCC	OTP	

**Table 1. Comparison of 77C25 With 77C20A**

	77C25/77P25	77C20A/77P20
Technology	CMOS/CMOS	CMOS/NMOS
Instruction cycle	122 ns	244 ns
Instruction ROM	2048 x 24 bits	512 x 23 bits
Data ROM	1024 x 16 bits	510 x 13 bits
Data RAM	256 x 16 bits	128 x 16 bits
Fixed-point multiplier	16 bits x 16 bits → 31 bits	16 bits x 16 bits → 31 bits
ALU	16-bit fixed-point	16-bit fixed-point
Accumulator	2 x 16 bits	2 x 16 bits
Host CPU interface	8-bit bus	8-bit bus
Serial interface	One input and one output 4 MHz	One input and one output 2 MHz
Temporary registers	Two	One

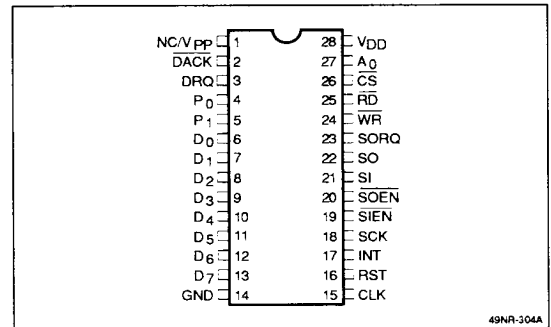
**Table 1. Comparison of 77C25 With 77C20A (cont)**

	77C25/77P25	77C20A/77P20
Additional instructions	JDPLN0	—
	JDPLNF	—
	Modification of RAM column data pointer M8-MF	—
DMA mode	Fully implemented	Partially implemented
Package	28-pin DIP	28-pin DIP
	44-pin PLCC	44-pin PLCC
	32-pin SOP	—
Power supply	5 V	5 V
Power consumption	50 mA (max) @ 8.192 MHz	40 mA (max) @ 8.192 MHz
Power saving mode (when idle)	Yes	No

Since the 77C25 executes an instruction in one external clock cycle (versus two cycles of the same 8.192-MHz clock for 77C20A), the 77C25 may be substituted for a 77C20A (or 7720A or 77P20) in a circuit without modification of that circuit. Hardware/software that implements data transfers—both serial and parallel—between the 77C25 and other devices in an existing 7720 design should use the handshake protocol described in the 77C25 User's Manual.

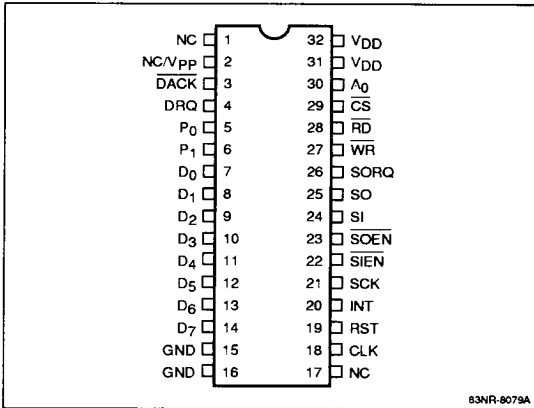
**Pin Configurations**

**28-Pin DIP**

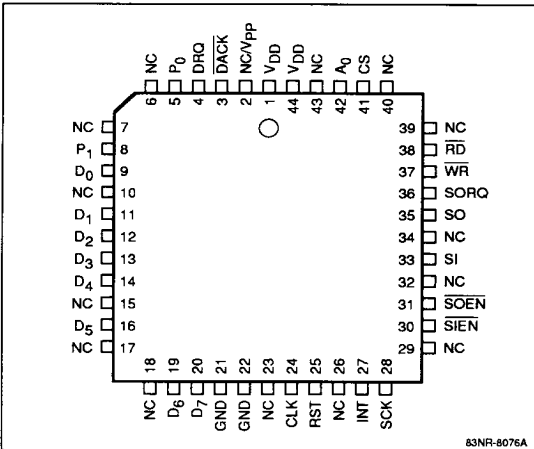


### Pin Configurations (cont)

#### 32-Pin SOP



#### 44-Pin PLCC



### Pin Identification

Symbol	Function
$A_0$	Status/data register select input
CLK	Single-phase master clock input
$\overline{CS}$	Chip select input
$D_0$ - $D_7$	Three-state I/O data bus
DACK	DMA request acknowledge input
DRQ	DMA request output
INT	Interrupt input
$P_0, P_1$	General-purpose output control lines
$\overline{RD}$	Read control signal input
RST	Reset input
SCK	Serial data I/O clock input
SI	Serial data input
$\overline{SIEN}$	Serial input enable input
SO	Three-state serial data output
SOEN	Serial output enable input
SORQ	Serial data output request
WR	Write control signal input
GND	Ground
$V_{DD}$	+5 V power supply
NC/Vpp	77C25: no connection 77P25: +12.5 V programming 77P25: +5 V for normal operation

3b

### PIN FUNCTIONS

#### $A_0$ (Status Data Register Select)

This input selects data register for read/write (low) or status register for read (high).

#### CLK

This is the single-phase master clock input.

#### $\overline{CS}$ (Chip Select)

This input enables data transfer through the data port with  $\overline{RD}$  or WR.

#### $D_0$ - $D_7$ (Data Bus)

This three-state I/O data bus transfers data between the data register or status register and the external data bus.

#### DACK (DMA Request Acknowledge)

This input indicates to the 77C25 that the data bus is ready for a DMA transfer ( $\overline{DACK} = \overline{CS}$  and  $A_0 = 0$ ).

3b-3

**DRQ (DMA Request)**

This output signals that the 77C25 is requesting a data transfer on the data bus.

**INT (Interrupt)**

A low-to-high transition on this pin executes a call instruction to location 100H if interrupts were previously enabled.

**P<sub>0</sub>, P<sub>1</sub>**

These pins are general-purpose output control lines.

 **$\overline{RD}$  (Read Control Signal)**

This input latches data from the data or status register to the data port where it is read by an external device.

**RST (Reset)**

This input initializes the 77C25 internal logic and sets the PC to 0.

**SCK (Serial Data I/O Clock)**

When this input is high, a serial data bit is transferred.

**SI (Serial Data Input)**

This pin inputs 8- or 16-bit serial data words from an external device such as an A/D converter.

 **$\overline{SIEN}$  (Serial Input Enable)**

This input enables the shift clock to the serial input register.

**SO (Serial Data Output)**

This three-state port outputs 8- or 16-bit data words to an external device such as a D/A converter.

 **$\overline{SOEN}$  (Serial Output Enable)**

This input enables the shift clock to the serial output register.

**SORQ (Serial Data Output Request)**

This output specifies to an external device that the serial data register has been loaded and is ready for output. SORQ is reset when the entire 8- or 16-bit word has been transferred.

 **$\overline{WR}$  (Write Control Signal)**

This input writes data from the data port into the data register.

**GND**

This is the connection to ground.

**V<sub>DD</sub> (Power Supply)**

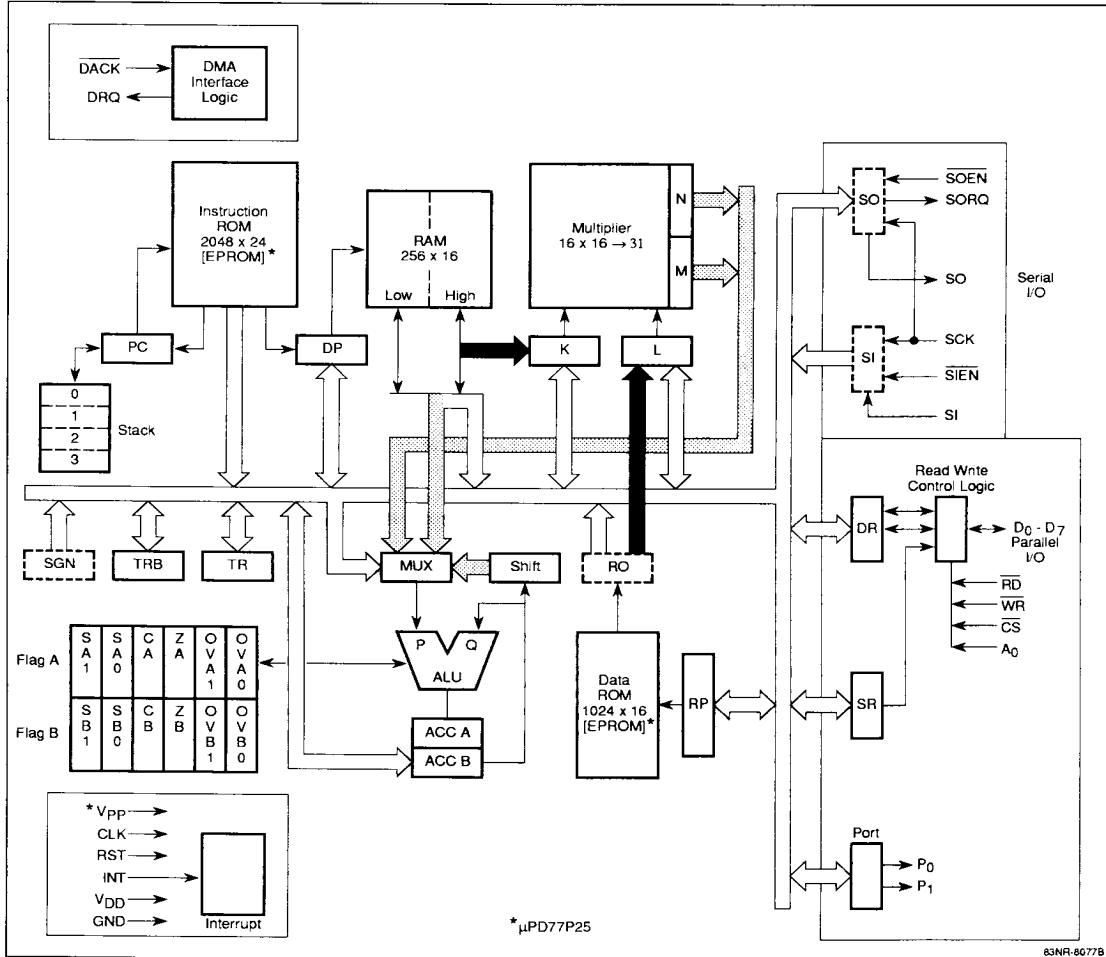
This pin is the + 5-volt power supply.

**NC/V<sub>PP</sub>**

This pin is not internally connected in the 77C25. In the 77P25, this pin inputs the programming voltage (V<sub>PP</sub>) when the part is being programmed.

This pin must be connected to V<sub>DD</sub> for normal 77P25 operation.

### Block Diagram



3b

### DATA BUSES

The primary bus (unshaded in the block diagram) makes a data path between all of the registers (including I/O), memory, and the processing sections. This bus is referred to as the IDB (internal data bus). The multiplier input registers K and L can be loaded not only from the IDB but alternatively via buses (darkened in the block diagram) directly from RAM to the K register and directly from data ROM to the L register. Output from the multiplier in the M and N registers is

typically added via buses (shaded in the block diagram) to either accumulator A or B as part of a multi-operation instruction.

### MEMORY

Memory is divided into three types: instruction ROM, data ROM, and data RAM. The 2048 x 24-bit words of instruction ROM are addressed by an 11-bit program counter that can be modified by an external reset, interrupt, call, jump, or return instruction.

3b-5

The data ROM is organized in 1024 x 16-bit words that are addressed through a 10-bit ROM pointer (RP register). The RP may be modified simultaneously with arithmetic instructions so that the next value is available for the next instruction. The data ROM is ideal for storing the necessary coefficients, conversion tables, and other constants for signal and math processing.

The data RAM is 256 x 16-bit words and is addressed through an 8-bit data pointer (DP register). The DP has extensive addressing features that operate simultaneously with arithmetic instructions, eliminating additional time for addressing or address modification.

### ARITHMETIC CAPABILITIES

One of the unique features of the 77C25 architecture is its arithmetic facilities. With a separate multiplier, ALU, and multiple internal data paths, the 77C25 is capable of carrying out a multiply, an add or other arithmetic operation, and a data move between internal registers in a single instruction cycle.

#### ALU

The ALU is a 16-bit two's complement unit capable of executing 16 distinct operations on data routed via the P and Q ALU inputs.

#### Accumulators (ACCA/ACCB)

Associated with the ALU are two 16-bit accumulators, each with its own set of flags, which are updated at the end of each arithmetic instruction. Table 2 shows the ACC A/B flag registers. In addition to zero result, sign, carry, and overflow flags, the 77C25 incorporates auxiliary overflow and sign flags (SA1, SB1, OVA1, OVB1). These flags enable the detection of an overflow condition and maintain the correct sign after as many as three successive additions or subtractions.

**Table 2. ACC A/B Flag Registers**

Flag A	SA1	SA0	CA	ZA	OVA1	OVA0
Flag B	SB1	SB0	CB	ZB	OVB1	OVB0

#### Sign Register (SGN)

When OVA1 is set, the SA1 bit will hold the corrected sign of the overflow. The SGN register will use SA1 to automatically generate saturation constants 7FFFH(+) or 8000H(-) to permit efficient limiting of a calculated value. The SGN register is not affected by arithmetic operations on accumulator B, but flags SB1, SB0, CB, ZB, OVB1, and OVB0 are affected.

#### Multiplier

Thirty-one bit results are developed by a 16 x 16-bit two's complement multiplier in 122 ns. The result is automatically latched to two 16-bit registers, M and N, at the end of each instruction cycle. The sign bit and 15 higher bits are in M and the 15 lower bits are in N; the LSB in N is zero. A new product is available for use after every instruction cycle, providing significant advantages in maximizing processing speed for real-time signal processing.

#### Stack

The 77C25 contains a four-level program stack for efficient program usage and interrupt handling.

#### Interrupt

The 77C25 supports a single-level interrupt. Upon sensing a high level on the INT terminal, a subroutine call to location 100H is executed. The EI bit of the status register automatically resets to 0, disabling the interrupt facility until it is reenabled under program control.

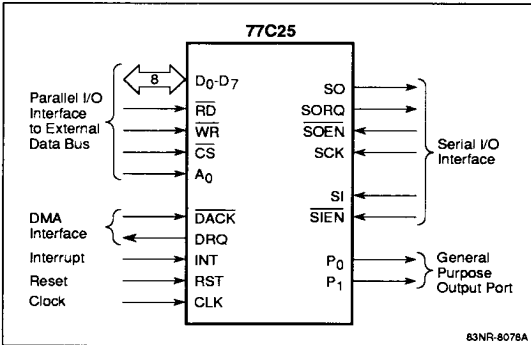
#### INPUT/OUTPUT

The 77C25 has three communication ports as shown in figure 1: two serial and one 8-bit parallel, each with its own control lines for interface handshaking. Parallel port operation is software-configurable to be in either polled mode or DMA mode. A general-purpose, two-line output port rounds out a full complement of interface capability.

#### Serial I/O

The two shift registers (SI, SO) are software-configurable to single- or double-byte transfers. The shift registers are externally clocked (SCK) to provide a simple interface between the 77C25 and serial peripherals such as A/D and D/A converters, codecs, or other 77C25's. Figure 2 shows serial I/O timing

**Figure 1. 77C25 Communication Ports**



### Parallel I/O

The 8-bit parallel I/O port may be used for transferring data or reading the 77C25 status as shown in table 3. Data transfer is handled through a 16-bit data register (DR) that is software-configurable for double- or single-byte data transfers. The port is ideally suited for operating with 8080, 8085, and 8086 processor buses and may be used with other processors and computer systems.

### DMA Mode Option

Parallel data transfers may be controlled (optionally) via DMA control lines DRQ and DACK. DMA mode allows high-speed transfers and reduced processor overhead. When in DMA mode, DACK input resets DRQ output when data transfer is completed.

**Note:** The RQM bit of the status register is affected by read/write operations in DMA mode the same as non-DMA mode. (In 7720 operation, RQM is not affected when in DMA mode.)

**Table 3. Parallel R/W Operation**

CS	A <sub>0</sub>	WR	RD	Operation
1	X	X	X	No effect on internal operation; D <sub>0</sub> -D <sub>7</sub> are at high impedance levels.
X	X	1	1	
0	0	0	1	Data from D <sub>0</sub> -D <sub>7</sub> is latched to DR (Note 1)
0	0	1	0	Contents of DR are output to D <sub>0</sub> -D <sub>7</sub> (Note 1)
0	1	0	1	Illegal (SR is read only)
0	1	1	0	Eight MSBs of SR are output to D <sub>0</sub> -D <sub>7</sub>
0	X	0	0	Illegal (may not read and write simultaneously)

**Notes:**

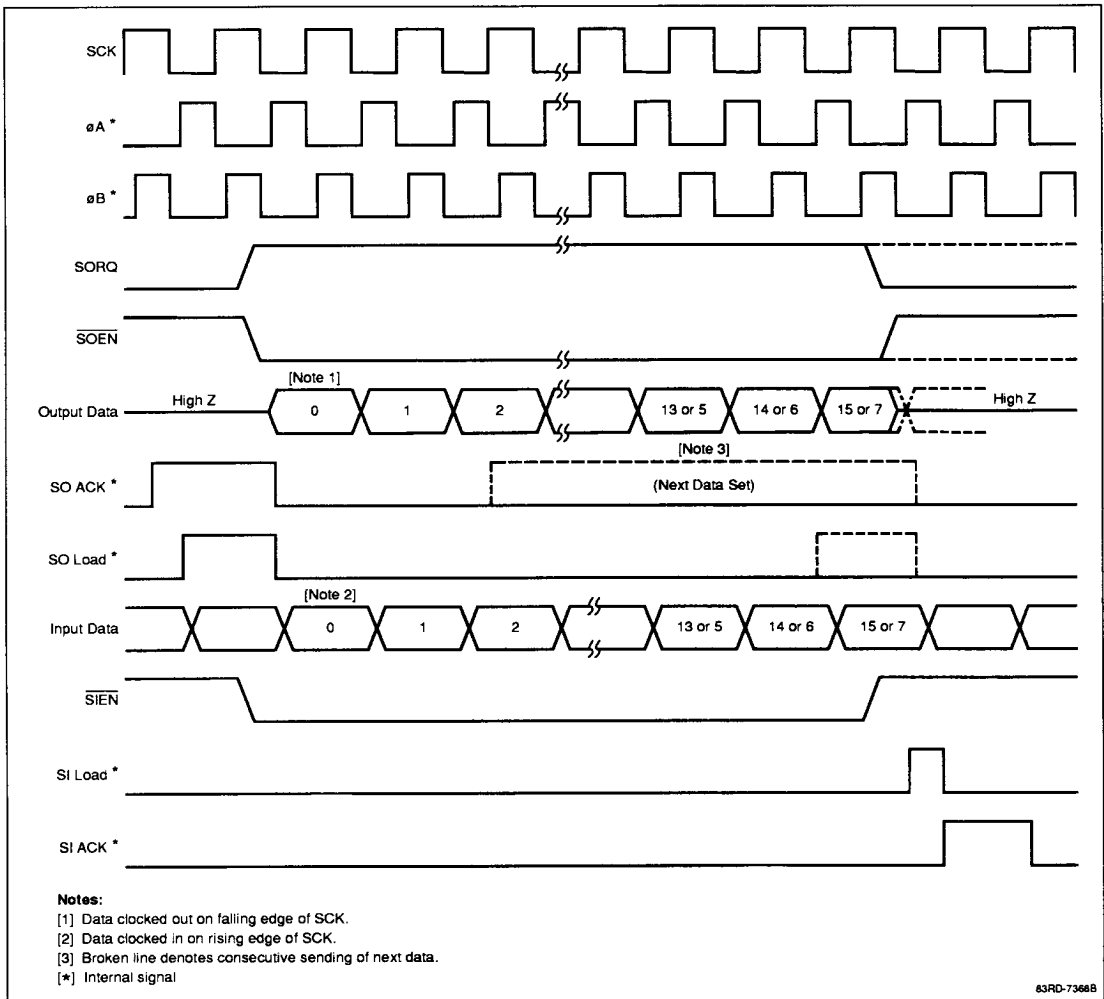
- (1) Eight MSBs or LSBs of data register (DR) are used, depending on DR status bit (DRS). The condition of DACK = 0 is equivalent to A<sub>0</sub> = CS = 0.

### Status Register

The status register, (figure 3, table 4) is a 16-bit register in which the 8 most significant bits may be read by the system's microprocessor for the latest parallel data I/O status. The RQM and DRS bits can only be affected by parallel data moves. The other bits can be written to (or read) by the 77C25 load immediate (LD) or move (MOV) instruction. The EI bit is automatically reset when an interrupt is serviced.

**3b**

Figure 2. Serial I/O Timing





**Figure 3. Status Register**

15	14	13	12	11	10	9	8
RQM	USF1	USF0	DRS	DMA	DRC	SOC	SIC
MSB							
7	6	5	4	3	2	1	0
EI	0	0	0	0	0	P1	P0
LSB							

**Table 4. Status Register Flags**

Flag	Description
RQM (Request for master)	A read or write from DR to IDB sets RQM = 1. An external read (write) resets RQM = 0.
USF1 and USF0 (User flags 1 and 0)	General-purpose flags that may be read by an external processor for user-defined signaling and 0)
DRS (DR status)	For 16-bit DR transfers (DRC = 0). DRS = 1 after the first 8 bits have been transferred. DRS = 0 after all 16 bits have been transferred.
DMA (DMA enable)	DMA = 0 (Non-DMA transfer mode) DMA = 1 (DMA transfer mode)
DRC (DR control)	DRC = 0 (16-bit mode) DRC = 1 (8-bit mode)
SOC (SO control)	SOC = 0 (16-bit mode) SOC = 1 (8-bit mode)
SIC (SI control)	SIC = 0 (16-bit mode) SIC = 1 (8-bit mode)
EI (Enable interrupt)	EI = 0 (interrupts disabled) EI = 1 (interrupts enabled)
P1, P0 (Ports 0 and 1)	P0 and P1 directly control the state of output pins P <sub>0</sub> and P <sub>1</sub>

### Temporary Registers

The 77C25 has two 16-bit temporary registers.

### INSTRUCTIONS

The 77C25 has three types of instructions: OP/RT (operation/return), JP (jump), and LD (load immediate). Each type takes the form of a 24-bit word and executes in 122 ns.

### Instruction Timing

To control the execution of instructions, the external 8-MHz clock is divided into phases for internal execution. The various elements of the 24-bit instruction word are executed in a set order. Multiplication automatically begins first. Also, data moves from source to destination before other elements of the instruction.

Data being moved on the internal data bus (IDB) is available for use in ALU operations (if P-select field of

the instruction specifies IDB). However, if the accumulator specified in the ASL field is also specified as the destination of the data move, the ALU operation becomes a NOP, as the data move supersedes the ALU operation.

Pointer modifications occur at the end of the instruction cycle after their values have been used for data moves. The result of multiplication is available at the end of the instruction cycle for possible use in the next instruction. If a return is specified as part of an OP instruction, it is executed last.

An assembly language OP instruction may consist of what looks like one to six lines of assembly code, but all of these lines are assembled together into one 24-bit instruction word. Therefore, the order of the six lines makes no difference in the order of execution described above. However, for understanding 77C25 operation and to eliminate confusion, assembly code should be written in the order described; that is: data move, ALU operations, data pointer modifications, and then return.

### OP/RT Instructions

Figure 4 illustrates the OP/RT (operation/return) instruction field specification. This is really one instruction type capable of executing all ALU functions listed in table 6.

The ALU functions operate on the value specified by the P-select field (table 5).

The RT indicates an option in bit D<sub>22</sub> that causes a return from subroutine or interrupt service.

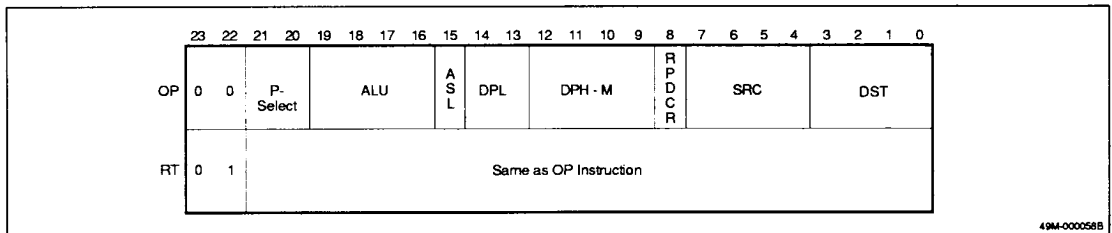
Besides the arithmetic functions, this instruction can also (1) modify the RAM data pointer DP, (2) modify the data ROM pointer RP, and (3) move data along the on-chip data bus from a source register to a destination register. Tables 7, 8, 9, and 10 show the ASL, DPL, DPH, and RPDCR fields, respectively. The possible source and destination registers are listed in tables 11 and 12.

**Table 5. P-Select Field**

Mnemonic	D <sub>21</sub>	D <sub>20</sub>	ALU Input
RAM	0	0	RAM
IDB	0	1	* Internal data bus
M	1	0	M register
N	1	1	N register

\* Any value on the on-chip data bus. Value may be selected from any of the source registers listed in table 11.

**Figure 4. OP/RT Instruction Field**



**Table 6. ALU Field**

Mnemonic	D <sub>19</sub>	D <sub>18</sub>	D <sub>17</sub>	D <sub>16</sub>	ALU Function	SA1, SB1	SA0, SB0	CA, CB	ZA, ZB	OVA1, OV B1	OVA0, OV B0
NOP	0	0	0	0	No operation	—	—	—	—	—	—
OR	0	0	0	1	OR	x	Δ	0	Δ	0	0
AND	0	0	1	0	AND	x	Δ	0	Δ	0	0
XOR	0	0	1	1	Exclusive OR	x	Δ	0	Δ	0	0
SUB	0	1	0	0	Subtract	Δ	Δ	Δ	Δ	Δ	Δ
ADD	0	1	0	1	Add	Δ	Δ	Δ	Δ	Δ	Δ
SBB	0	1	1	0	Subtract with borrow	Δ	Δ	Δ	Δ	Δ	Δ
ADC	0	1	1	1	Add with carry	Δ	Δ	Δ	Δ	Δ	Δ
DEC	1	0	0	0	Decrement ACC	Δ	Δ	Δ	Δ	Δ	Δ
INC	1	0	0	1	Increment ACC	Δ	Δ	Δ	Δ	Δ	Δ
CMP	1	0	1	0	Complement ACC (one's complement)	x	Δ	0	Δ	0	0
SHR1	1	0	1	1	1-bit right shift	x	Δ	Δ	Δ	0	0
SHL1	1	1	0	0	1-bit left shift	x	Δ	Δ	Δ	0	0
SHL2	1	1	0	1	2-bit left shift	x	Δ	0	Δ	0	0
SHL4	1	1	1	0	4-bit left shift	x	Δ	0	Δ	0	0
XCHG	1	1	1	1	8-bit exchange	x	Δ	0	Δ	0	0

**Symbols:**

- Δ May be affected, depending on the results
- Previous status can be held
- 0 Reset
- x Indefinite

**Table 7. ASL Field**

Mnemonic	D <sub>15</sub>	ACC Selection
ACCA	0	ACCA
ACCB	1	ACCB

**Table 8. DPL Field**

Mnemonic	D <sub>14</sub>	D <sub>13</sub>	Low DP Modify (DP <sub>3</sub> -DP <sub>0</sub> )
DPNOP	0	0	No operation
DPINC	0	1	Increment DPL
DPDEC	1	0	Decrement DPL
DPCLR	1	1	Clear DPL

**Table 9. DPH Field**

Mnemonic	D <sub>12</sub>	D <sub>11</sub>	D <sub>10</sub>	D <sub>9</sub>	High DP Modify
M0	0	0	0	0	Exclusive OR of DPH (DP <sub>7</sub> -DP <sub>4</sub> ) with the mask defined by the 4 bits (D <sub>12</sub> -D <sub>9</sub> ) of the DPH field
M1	0	0	0	1	
M2	0	0	1	0	
M3	0	0	1	1	
M4	0	1	0	0	
M5	0	1	0	1	
M6	0	1	1	0	
M7	0	1	1	1	
M8	1	0	0	0	
M9	1	0	0	1	
MA	1	0	1	0	
MB	1	0	1	1	
MC	1	1	0	0	
MD	1	1	0	1	
ME	1	1	1	0	
MF	1	1	1	1	

**Table 10. RPDCR Field**

Mnemonic	D <sub>8</sub>	RP operation
RPNOP	0	No operation
RPDEC	1	Decrement RP

**Table 11. SRC Field**

Mnemonic	D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	Source Register
NON/TRB	0	0	0	0	TRB (Note 1)
A	0	0	0	1	ACCA (Accumulator A)
B	0	0	1	0	ACCB (Accumulator B)
TR	0	0	1	1	TR temporary register
DP	0	1	0	0	DP data pointer
RP	0	1	0	1	RP ROM pointer
RO	0	1	1	0	RO ROM output data
SGN	0	1	1	1	SGN sign register
DR	1	0	0	0	DR data register
DRNF	1	0	0	1	DR no flag (Note 2)
SR	1	0	1	0	SR status register

**Table 11. SRC Field (cont)**

Mnemonic	D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	Source Register
SIM	1	0	1	1	SI serial in MSB (Note 3)
SIL	1	1	0	0	SI serial in LSB (Note 4)
K	1	1	0	1	K register
L	1	1	1	0	L register
MEM	1	1	1	1	RAM

**Notes:**

- (1) Contents of TRB register are also output if NON is specified.
- (2) DR to IDB, RQM not set. in DMA, DRQ not set.
- (3) First bit in goes to MSB, last bit to LSB.
- (4) First bit goes to LSB, last bit to MSB (bit reversed).

**Jump Instructions**

Figure 5 shows the JP instruction field specification. Bits D<sub>21</sub>, D<sub>20</sub>, and D<sub>19</sub> of the BRCH field identify the three types of instructions: unconditional jump (100), subroutine call (101), and conditional jump (010). Table 13 lists the instruction mnemonics for the complete BRCH field, bits D<sub>21</sub>-D<sub>13</sub>.

All the instructions in table 13—if unconditional or if the specified condition is true—take their next program execution address from the next address field (NA) in figure 5. Otherwise, PC = PC + 1.

**Load Data (LD) Instructions**

Figure 6 shows the LD instruction field specification.

The load data instruction will take the 16-bit value contained in the immediate data field (ID) and place it in the register specified by the destination field (DST). This is the same as the DST field (table 12) in the OP/RT instruction.

**Table 12. DST Field**

Mnemonic	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>	Destination Register
@NON	0	0	0	0	No register
@A	0	0	0	1	ACCA (Accumulator A)
@B	0	0	1	0	ACCB (Accumulator B)
@TR	0	0	1	1	TR temporary register
@DP	0	1	0	0	DP data pointer
@RP	0	1	0	1	RP ROM pointer
@DR	0	1	1	0	DR data register
@SR	0	1	1	1	SR status register
@SOL	1	0	0	0	SO serial out LSB (Note 1)

**Table 12. DST Field (cont)**

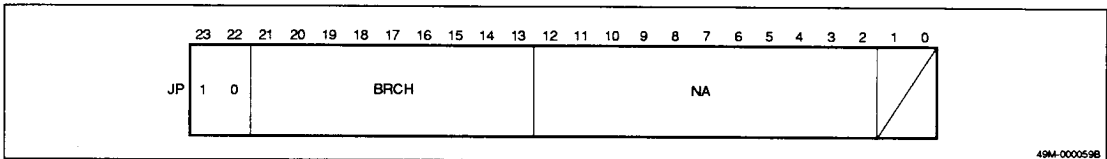
Mnemonic	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>	Destination Register
@SOM	1	0	0	1	SO serial out MSB (Note 2)
@K	1	0	1	0	K (Mult)
@KLR	1	0	1	1	IDB → K, ROM → L (Note 3)
@KLM	1	1	0	0	Hi RAM → K, IDB → L (Note 4)
@L	1	1	0	1	L register
@TRB	1	1	1	0	TRB register

Mnemonic	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>	Destination Register
@MEM	1	1	1	1	RAM

**Notes:**

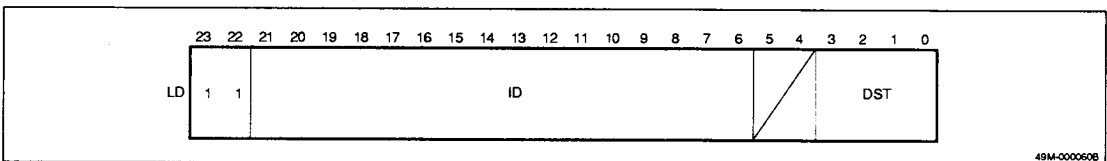
- (1) LSB is first bit out.
- (2) MSB is first bit out.
- (3) Internal data bus to K, and ROM to L register.
- (4) Contents of RAM address specified by DP<sub>6</sub> = 1 is placed in K register, IDB is placed in L (that is: 1, DP<sub>5</sub>, DP<sub>4</sub>, DP<sub>3</sub>-DP<sub>0</sub>).

**Figure 5. JP Instruction Field Specification**



49M-000059B

**Figure 6. LD Instruction Field Specification**



49M-000050B

**Table 13. BRCH Field**

Mnemonic	D <sub>21</sub> -D <sub>19</sub>	D <sub>18</sub> -D <sub>16</sub>	D <sub>15</sub> -D <sub>13</sub>	Conditions
JMP	1 0 0	0 0 0	0 0 0	No condition
CALL	1 0 1	0 0 0	0 0 0	No condition
JNCA	0 1 0	0 0 0	0 0 0	CA = 0
JCA	0 1 0	0 0 0	0 1 0	CA = 1
JNCB	0 1 0	0 0 0	1 0 0	CB = 0
JCB	0 1 0	0 0 0	1 1 0	CB = 1
JNZA	0 1 0	0 0 1	0 0 0	ZA = 0
JZA	0 1 0	0 0 1	0 1 0	ZA = 1
JNZB	0 1 0	0 0 1	1 0 0	ZB = 0
JZB	0 1 0	0 0 1	1 1 0	ZB = 1
JNOVA0	0 1 0	0 1 0	0 0 0	OVA0 = 0
JOVA0	0 1 0	0 1 0	0 1 0	OVA0 = 1
JNOVB0	0 1 0	0 1 0	1 0 0	OVB0 = 0
JOVB0	0 1 0	0 1 0	1 1 0	OVB0 = 1
JNOVA1	0 1 0	0 1 1	0 0 0	OVA1 = 0
JOVA1	0 1 0	0 1 1	0 1 0	OVA1 = 1

Mnemonic	D <sub>21</sub> -D <sub>19</sub>	D <sub>18</sub> -D <sub>16</sub>	D <sub>15</sub> -D <sub>13</sub>	Conditions
JNOVB1	0 1 0	0 1 1	1 0 0	OVB1 = 0
JOVB1	0 1 0	0 1 1	1 1 0	OVB1 = 1
JNSA0	0 1 0	1 0 0	0 0 0	SA0 = 0
JSA0	0 1 0	1 0 0	0 1 0	SA0 = 1
JNSB0	0 1 0	1 0 0	1 0 0	SB0 = 0
JSB0	0 1 0	1 0 0	1 1 0	SB0 = 1
JNSA1	0 1 0	1 0 1	0 0 0	SA1 = 0
JSA1	0 1 0	1 0 1	0 1 0	SA1 = 1
JNSB1	0 1 0	1 0 1	1 0 0	SB1 = 0
JSB1	0 1 0	1 0 1	1 1 0	SB1 = 1
JDPL0	0 1 0	1 1 0	0 0 0	DPL = 0
JDPLN0	0 1 0	1 1 0	0 0 1	DPL ≠ 0
JDPLF	0 1 0	1 1 0	0 1 0	DPL = FH
JDPLNF	0 1 0	1 1 0	0 1 1	DPL ≠ FH
JNSIAK	0 1 0	1 1 0	1 0 0	SIACK = 0
JSIACK	0 1 0	1 1 0	1 1 0	SIACK = 1

**Table 13. BRCH Field (cont)**

Mnemonic	D <sub>21</sub> -D <sub>19</sub>	D <sub>18</sub> -D <sub>16</sub>	D <sub>15</sub> -D <sub>13</sub>	Conditions
JNSOAK	0 1 0	1 1 1	0 0 0	SO ACK = 0
JSOAK	0 1 0	1 1 1	0 1 0	SO ACK = 1
JNRQM	0 1 0	1 1 1	1 0 0	RQM = 0
JRQM	0 1 0	1 1 1	1 1 0	RQM = 1

### ELECTRICAL SPECIFICATIONS

#### Absolute Maximum Ratings

T<sub>A</sub> = 25°C unless otherwise specified

Supply voltage, V <sub>DD</sub>	-0.5 to +7.0 V
V <sub>PP</sub> (77P25)	-0.5 to +13.5 V
Input voltage, V <sub>I</sub>	-0.5 to V <sub>DD</sub> + 0.5 V
V <sub>RST</sub> (77P25)	-0.5 to +13 V
Output voltage, V <sub>O</sub>	-0.5 to V <sub>DD</sub> + 0.5 V
Storage temperature, T <sub>STG</sub>	-65 to 150°C
Operating temperature, T <sub>OPT</sub>	
77C25/77C25-10	-40 to +80°C
77P25 (Normal operation)	-10 to +70°C
77P25 (PROM mode)	+20 to +30°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage.

### Recommended Operating Conditions

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Supply voltage	V <sub>DD</sub>	4.5	5.0	5.5	V	Normal operation
		5.7	6.0	6.25	V	Programming*
	V <sub>PP</sub> *	4.5	5.0	5.5	V	Reading and normal operation
		12	12.5	12.8	V	Programming
Input voltage, low	V <sub>IL</sub>	-0.3		0.8	V	
Input voltage, high	V <sub>IH</sub>	2.2		V <sub>DD</sub> + 0.3	V	
CLK input voltage, low	V <sub>ILC</sub>	-0.3		0.5	V	
CLK input voltage, high	V <sub>IHC</sub>	3.5		V <sub>DD</sub> + 0.3	V	
Input voltage for setting PROM mode	V <sub>RST</sub> *	11.5	12.0	12.5	V	Reading and writing
Operating temperature	T <sub>OPT</sub>	-40	+25	+85	°C	77C25/77C25-10
		-10	+25	+70	°C	Normal operation*
		+20	+25	+30	°C	PROM mode*

\* For μPD77P25

## μPD77C25/77P25

### DC Characteristics, Normal

$T_A = -40$  to  $+85^\circ\text{C}$  (77C25/77C25-10),  $-10$  to  $70^\circ\text{C}$  (77P25);  $V_{DD} = 4.5$  to  $5.5$  V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Output voltage, low	$V_{OL}$			0.45	V	$I_{OL} = 2.0$ mA
Output voltage, high	$V_{OH}$	$0.7 V_{DD}$			V	$I_{OH} = 400$ μA
Input leakage current, low	$I_{LIL}$			-10	μA	$V_{IN} = 0$ V
Input leakage current, high	$I_{LIH}$			10	μA	$V_{IN} = V_{DD}$
Output leakage current, low	$I_{LOL}$			-10	μA	$V_{OUT} \approx 0.47$ V
Output leakage current, high	$I_{LOH}$			10	μA	$V_{OUT} = V_{DD}$
Supply current (77C25)	$I_{DD}$		25	50	mA	$f_{CLK} = 8.192$ MHz
			15	25	mA	$f_{CLK} = 8.192$ MHz; RST = 1
Supply current (77P25)	$I_{DD}$		35	60	mA	$f_{CLK} = 8.192$ MHz
			20	35	mA	$f_{CLK} = 8.192$ MHz; RST = 1
			$I_{PP}$	1	mA	

### DC Characteristics, PROM Mode

$T_A = +20$  to  $+30^\circ\text{C}$ ;  $V_{DD} = 5.75$  to  $6.25$  V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input leakage current	$I_{RST}$			30	μA	$V_{RST} = 12.0 \pm 0.5$ V
Supply current	$I_{CC}$			60	mA	
	$I_{PP}$			30	mA	

### Capacitance

$T_A = 25^\circ\text{C}$ ;  $V_{DD} = 0$  V

Parameter	Symbol	Typ	Max	Unit	Conditions
CLK, SCK capacitance	$C_\phi$	20		pF	$f_c = 1$ MHz
Input capacitance	$C_{IN}$	20		pF	
Output capacitance	$C_{OUT}$	20		pF	

### AC Characteristics

$T_A = -40$  to  $85^\circ\text{C}$  (77C25/77C25-10),  $-10$  to  $+70^\circ\text{C}$  (77P25);  $V_{DD} = 4.5$  to  $5.5$  V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
<b>Clock</b>						
CLK cycle time 77C25/77P25 77C25-10	$t_{CYC}$	120 100	122 100	2000 2000	ns ns	Measuring at 2 V
CLK pulse width 77C25 77P25 77C25-10	$t_{CC}$	55 60 45			ns ns ns	
CLK rise time	$t_{CR}$			10	ns	Measuring at 1 and 3 V
CLK fall time	$t_{CF}$			10	ns	
SCK cycle time 77C25/77P25 77C25-10	$t_{CYS}$	240 200	244 200		ns ns	
SCK high pulse width 77C25/77P25 77C25-10	$t_{SSH}$	100 80			ns ns	
SCK low pulse width 77C25/77P25 77C25-10	$t_{SSL}$	100 80			ns ns	
SCK rise time	$t_{SR}$			20	ns	
SCK fall time	$t_{SF}$			20	ns	
<b>Host Interface Timing</b>						
$A0$ , $\overline{CS}$ , DACK setup time for $\overline{RD}$	$t_{SAR}$	0			ns	
$A0$ , $\overline{CS}$ , DACK hold time for $\overline{RD}$	$t_{HRA}$	0			ns	
$\overline{RD}$ pulse width 77C25/77P25 77C25-10	$t_{WRD}$	120 100			ns ns	
$A0$ , $\overline{CS}$ , DACK setup time for $\overline{WR}$	$t_{SAW}$	0			ns	
$A0$ , $\overline{CS}$ , DACK hold time for $\overline{WR}$	$t_{HWA}$	0			ns	
$\overline{WR}$ pulse width 77C25/77P25 77C25-10	$t_{WWR}$	120 100			ns ns	
Data setup time for $\overline{WR}$ 77C25/77P25 77C25-10	$t_{SDW}$	100 80			ns ns	
Data hold time for $\overline{WR}$	$t_{HWD}$	0			ns	
$\overline{RD}$ , $\overline{WR}$ recovery time 77C25/77P25 77C25-10	$t_{RW}$	100 80			ns ns	
DACK hold time for DRQ	$t_{HRQA}$	$0.5t_{CYC}$			ns	
$\overline{RD}$ , $\overline{WR}$ setup time for CLK	$t_{SRWC}$	50			ns	Note 1
$\overline{RD}$ , $\overline{WR}$ hold time for CLK	$t_{HCRW}$	50			ns	Note 1
<b>Host Interface Switching</b>						
$\overline{RD}$ ↓ → data delay time 77C25/77P25 77C25-10	$t_{DRD}$			100 80	ns ns	

3b

**AC Characteristics (cont)**

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
RD ↑ → data float time 77C25/77P25 77C25-10	t <sub>FRD</sub>	10 10		65 50	ns ns	
CLK ↑ → DRQ delay time 77C25/77P25 77C25-10	t <sub>DCRQ</sub>			80 65	ns ns	
DACK ↓ → DRQ delay time 77C25/77P25 77C25-10	t <sub>DARQ</sub>			110 90	ns ns	
CLK ↑ → P <sub>0</sub> , P <sub>1</sub> delay time 77C25/77P25 77C25-10	t <sub>DCP</sub>			100 80	ns ns	

**Interrupt Reset Timing**

RST setup time for CLK 77C25/77P25 77C25-10	t <sub>SRSC</sub>	50 40			ns ns	Note 1
RST hold time for CLK 77C25/77P25 77C25-10	t <sub>HCRS</sub>	50 40			ns ns	Note 1
RST pulse width	t <sub>RST</sub>	2t <sub>CYC</sub> 3t <sub>CYC</sub>			ns ns	System reset Enter power saving state
INT setup time for CLK 77C25/77P25 77C25-10	t <sub>SINC</sub>	50 40			ns ns	Note 1
INT hold time for CLK 77C25/77P25 77C25-10	t <sub>HCIN</sub>	50 40			ns ns	Note 1
INT pulse width	t <sub>INT</sub>	3t <sub>CYC</sub>			ns	
INT recovery time	t <sub>RINT</sub>	2t <sub>CYC</sub>			ns	

**Interrupt Reset Switching**

CLK ↑ → reset state delay time 77C25/77P25 77C25-10	t <sub>DCRS</sub>			100 80	ns ns	
-----------------------------------------------------------	-------------------	--	--	-----------	----------	--

**Serial Interface Timing**

SIEN, S1 setup time for SCK 77C25/77P25 77C25-10	t <sub>SSIS</sub>	50 40			ns ns	
SIEN, S1 hold time for SCK 77C25/77P25 77C25-10	t <sub>HSSI</sub>	30 20			ns ns	
SOEN setup time for SCK 77C25/77P25 77C25-10	t <sub>SSES</sub>	50 40			ns ns	
SOEN hold time for SCK 77C25/77P25 77C25-10	t <sub>HSSE</sub>	30 25			ns ns	
CLK setup time for SCK 77C25/77P25 77C25-10	t <sub>SCS</sub>	50 40			ns ns	Note 1



### AC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
CLK hold time for SCK 77C25/77P25	t <sub>HSC</sub>	50			ns	Note 1
77C25-10		40			ns	
SCK setup time for CLK 77C25/77P25	t <sub>SSC</sub>	50			ns	Note 1
77C25-10		40			ns	
SCK hold time for CLK 77C25/77P25	t <sub>HCS</sub>	50			ns	Note 1
77C25-10		40			ns	

### Serial Interface Switching

SCK ↑ → SORQ delay time 77C25/77P25	t <sub>DSSQ</sub>	30		150	ns	
77C25-10		20		120	ns	
SCK ↓ → SO delay time 77C25/77P25	t <sub>DLSO</sub>			60	ns	
77C25-10				50	ns	
SCK ↓ → SO hold time 77C25/77P25	t <sub>HLSO</sub>	0			ns	
77C25-10		0			ns	
SCK ↓ → SO float time 77C25/77P25	t <sub>FSSO</sub>			60	ns	
77C25-10				50	ns	

#### Notes:

- (1) Setup and hold requirement for asynchronous signal only guarantees recognition at next CLK.

### PROM Program Timing

T<sub>A</sub> = 25 ± 5°C; V<sub>IHR</sub> = 12.0 ± 0.5 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
<b>Data Read</b>						
CE setup time for RST	t <sub>SRSC</sub>	2			μs	V <sub>DD</sub> = 5.0 ± 0.5 V
OE setup time for RST	t <sub>SRSE</sub>	2			μs	V <sub>PP</sub> = V <sub>DD</sub>
<b>Data Read Switching</b>						
Address to output delay	t <sub>DAD</sub>			200	ns	V <sub>DD</sub> = 5.0 ± 0.5 V
CE to output delay	t <sub>DCD</sub>			200	ns	V <sub>PP</sub> = V <sub>DD</sub>
OE to output delay	t <sub>DODR</sub>			75	ns	
OE high to output float	t <sub>FCD</sub>	0		60	ns	
Address to output hold	t <sub>HAD</sub>	0			ns	
<b>Data Write</b>						
CE setup time for RST	t <sub>SRSC</sub>	2			μs	V <sub>DD</sub> = 6.0 ± 0.25 V
CE setup time for address	t <sub>SAC</sub>	2			μs	V <sub>PP</sub> = 12.5 ± 0.3 V
CE setup time for data	t <sub>SDC</sub>	2			μs	
CE setup time for V <sub>pp</sub>	t <sub>SVPC</sub>	2			μs	
CE setup time for V <sub>DD</sub>	t <sub>SVDC</sub>	2			μs	
OE setup time for data	t <sub>SDO</sub>	2			μs	

**PROM Program Timing (cont)**

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Address hold time	$t_{HCA}$	2			μs	
Data hold time	$t_{HCD}$	2			μs	
Initial program pulse width	$t_{WC0}$	0.95	1.0	1.05	ms	
Overprogram pulse width	$t_{WC1}^*$	2.85		78.75	ms	

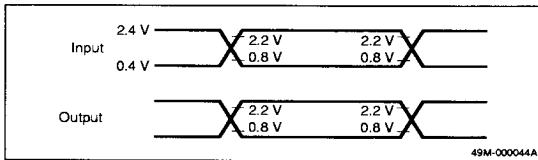
**Data Write Switching**

OE to output float time	$t_{FOD}$	0		130	ns	$V_{DD} = 6.0 \pm 0.25 V$
OE to output delay	$t_{DODW}$			150	ns	$V_{PP} = 12.5 \pm 0.3 V$

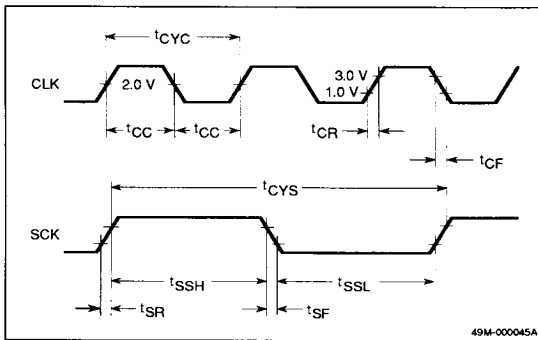
$t_{WC1} = 3nt_{WC0}$  assuming initial program pulse is applied n times.

**Timing Waveforms**

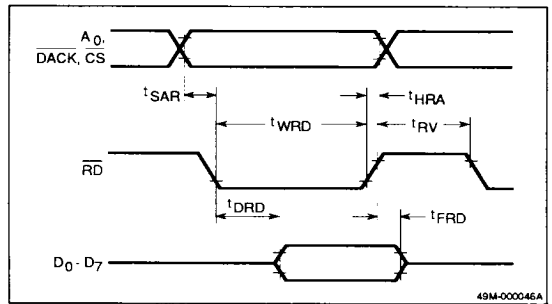
**Input/Output Voltage Reference Levels**



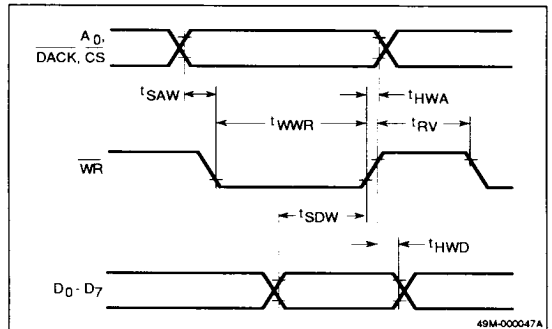
**Clock Timing**



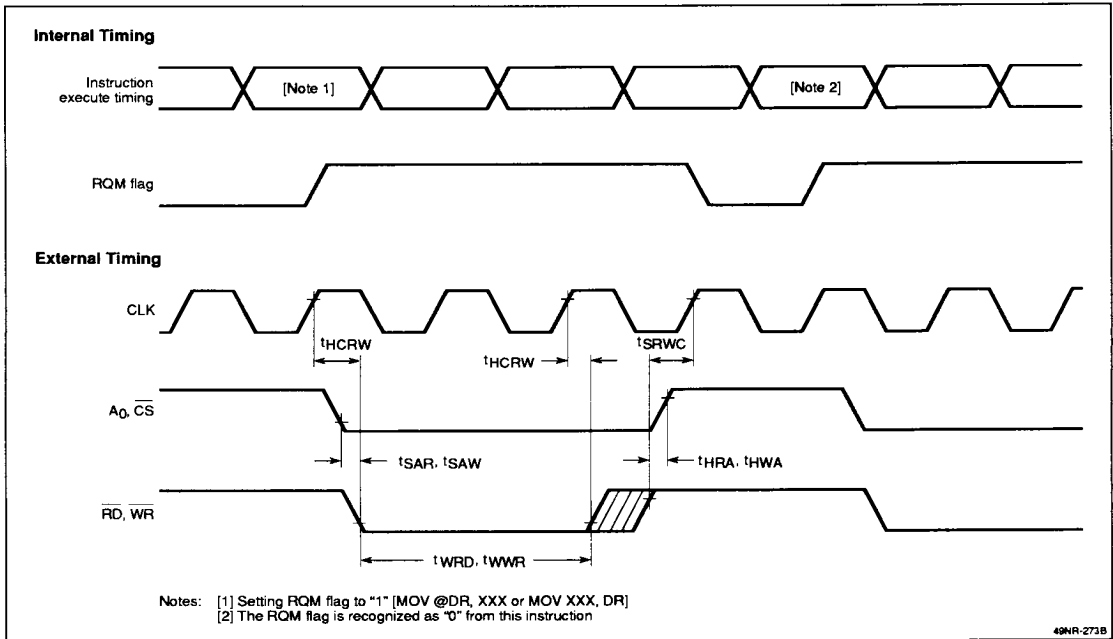
**Host Read Operation**



**Host Write Operation**

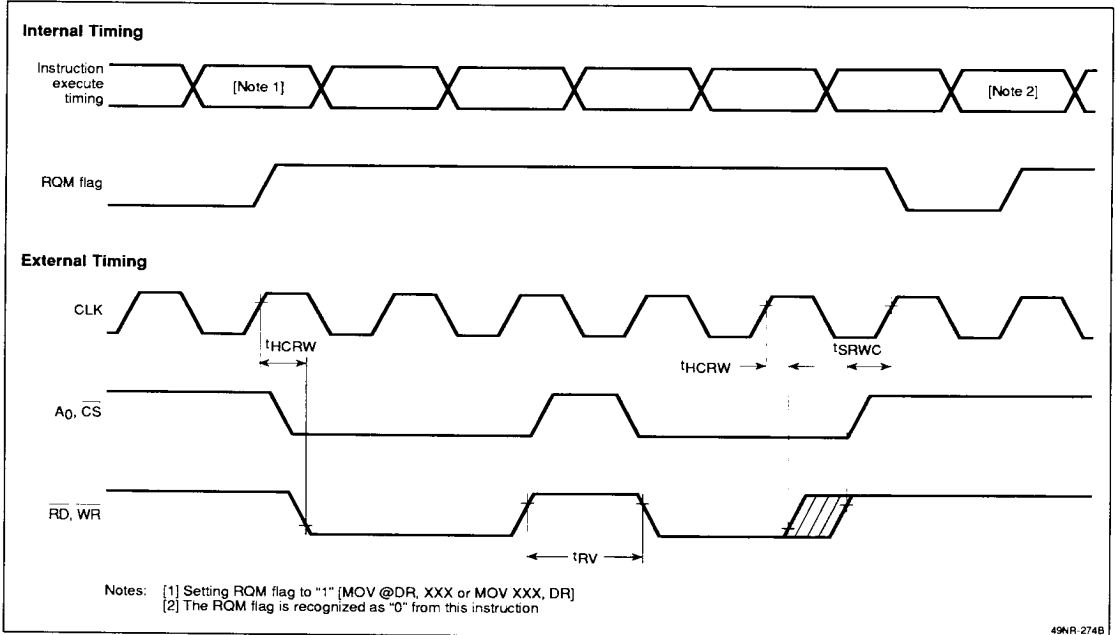


## Normal Operation, 8-Bit Mode

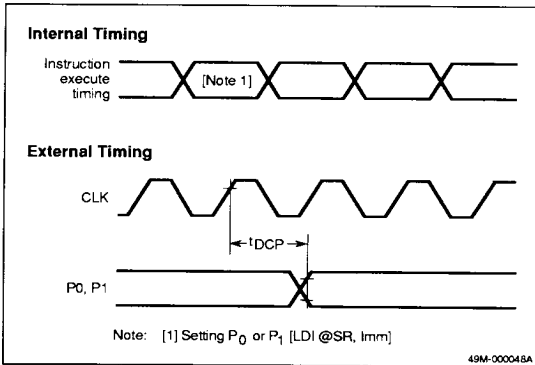


3b

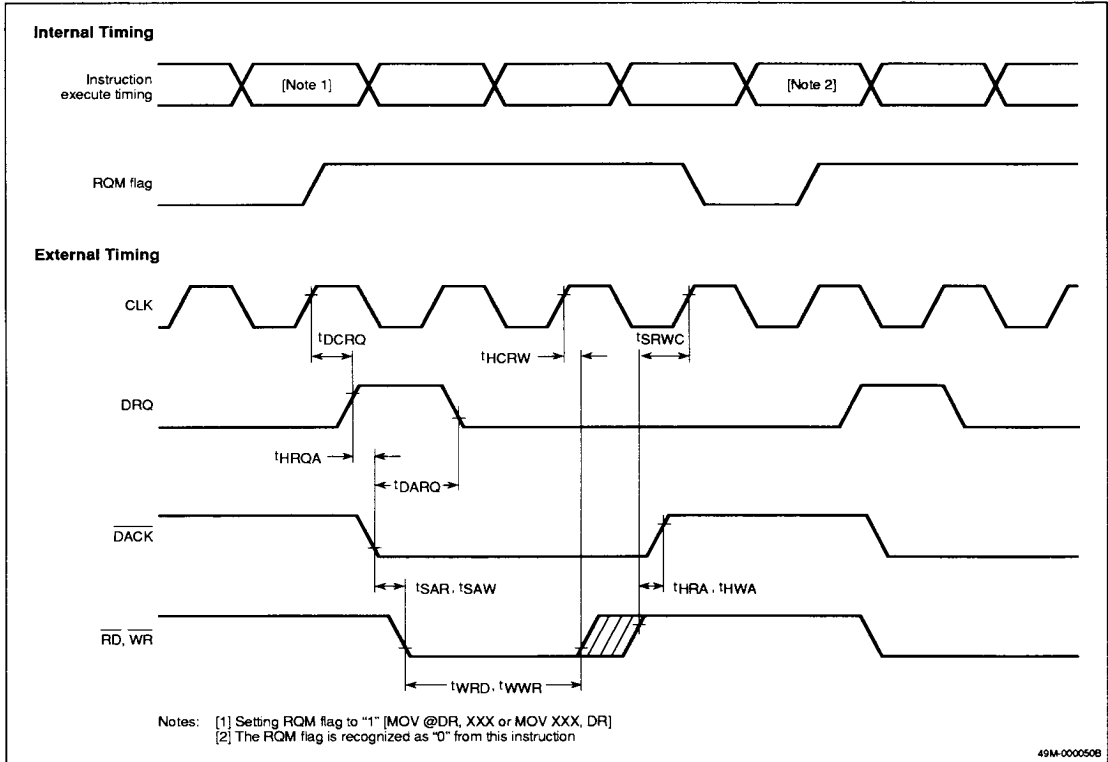
**Normal Operation, 16-Bit Mode**



**Port Operation**

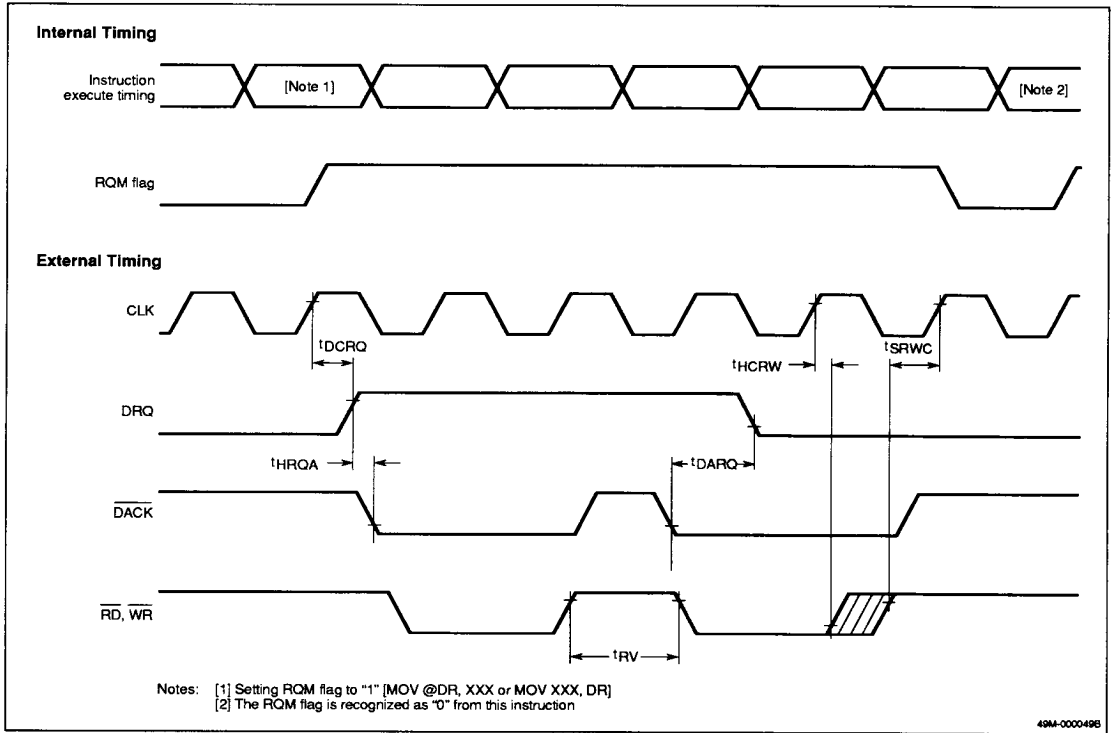


## DMA Operation, 8-Bit Mode

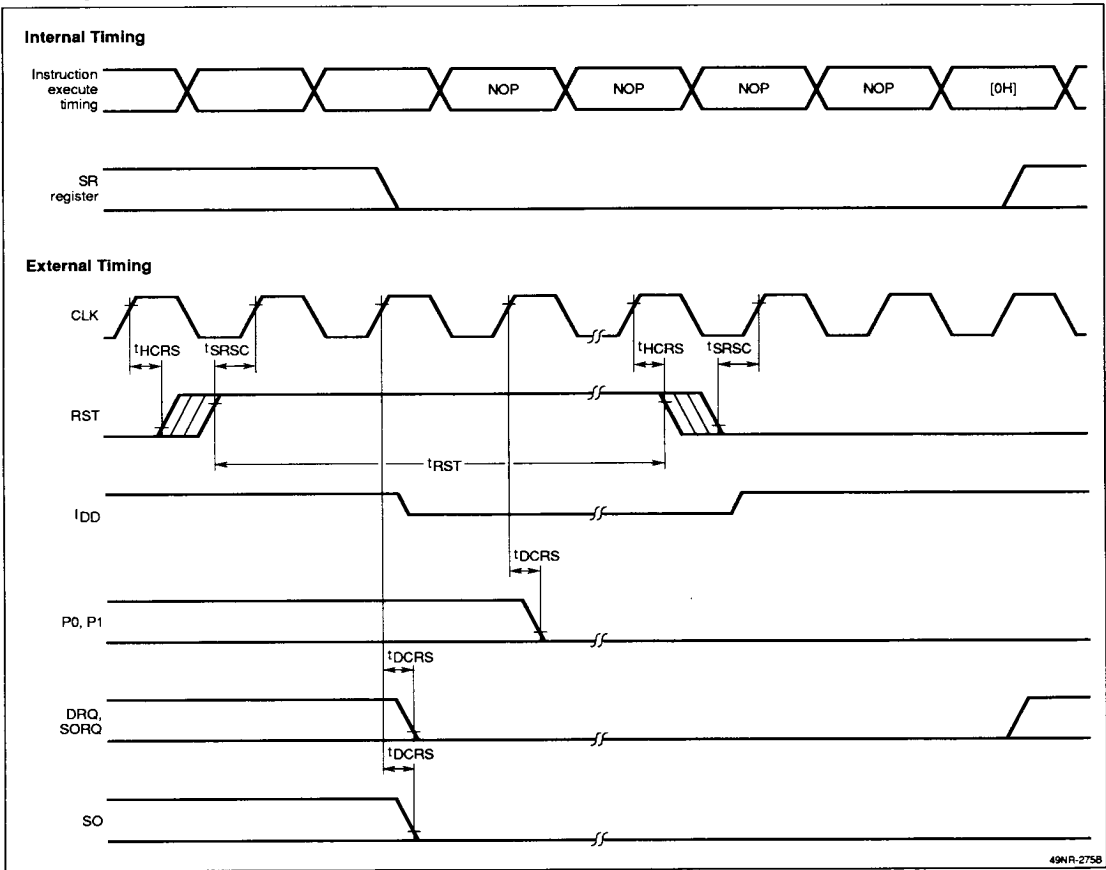


3b

DMA Operation, 16-Bit Mode

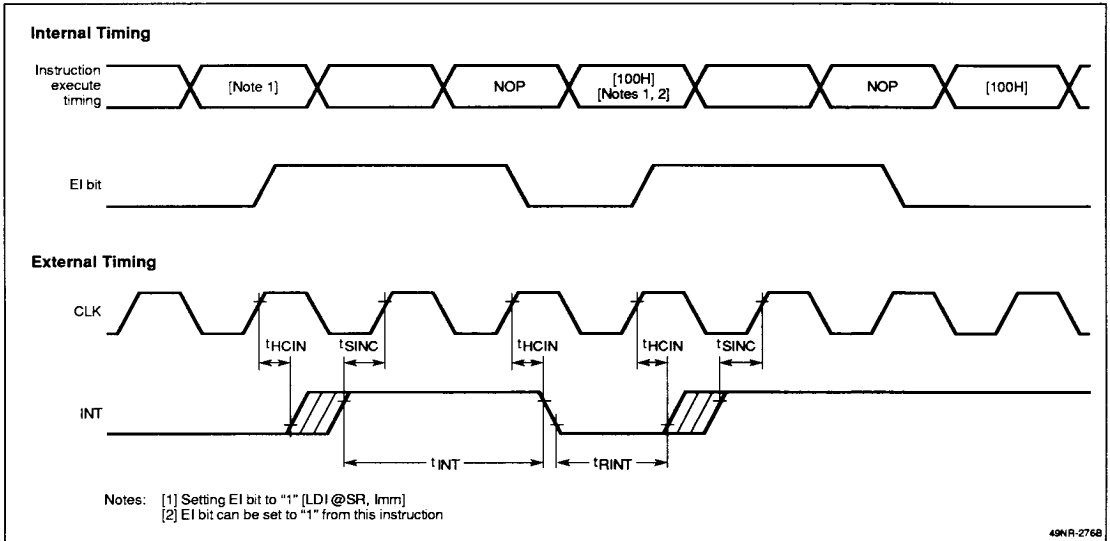


## Reset Operation

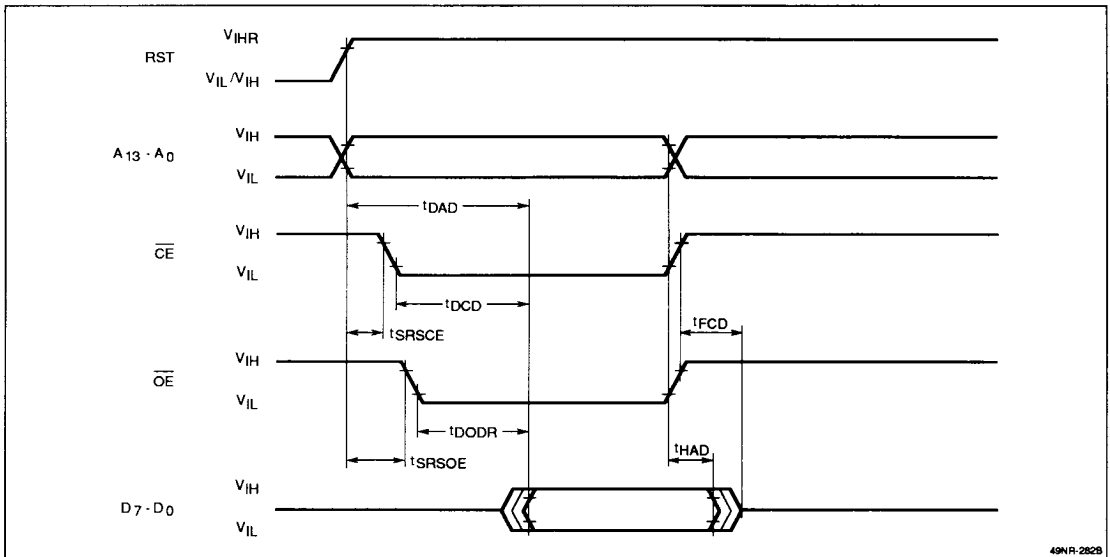


3b

**Interrupt Operation**

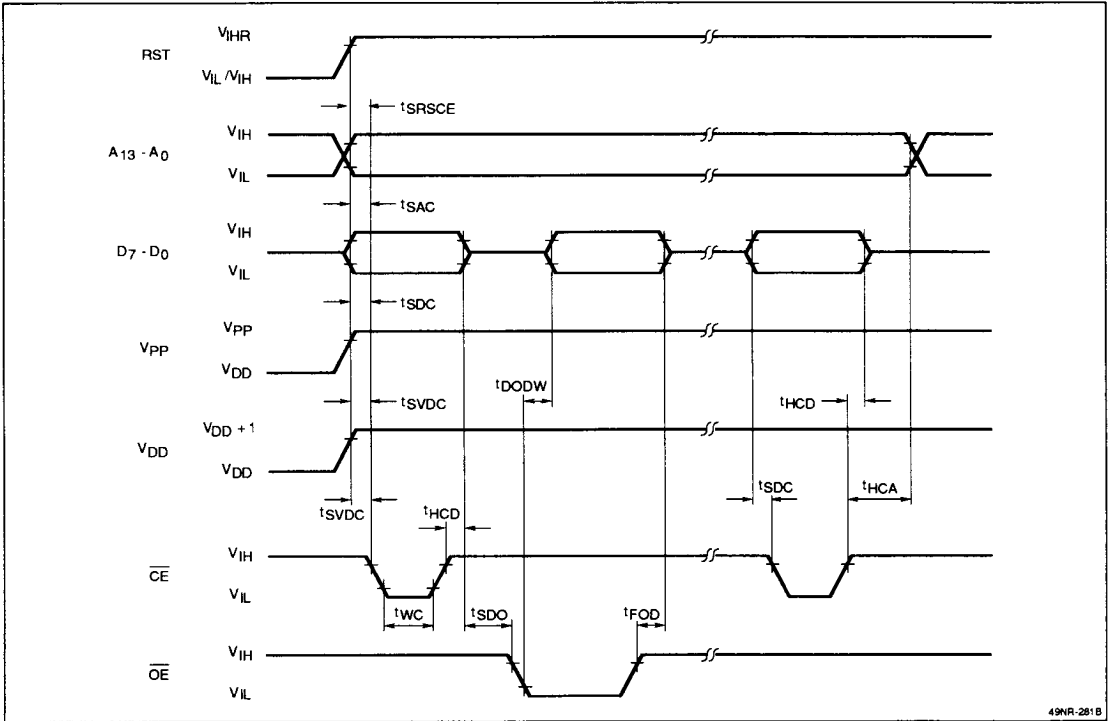


**PROM Read Timing**



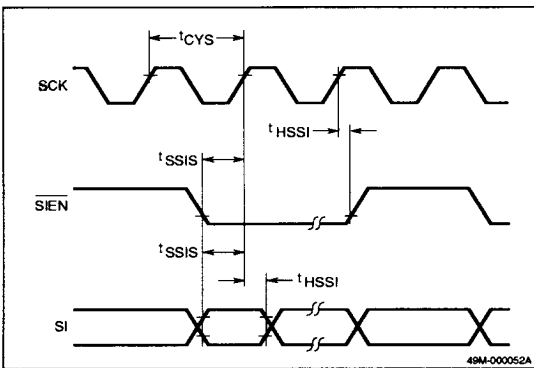


## PROM Program Timing



3b

Figure 7. Serial Input Operation



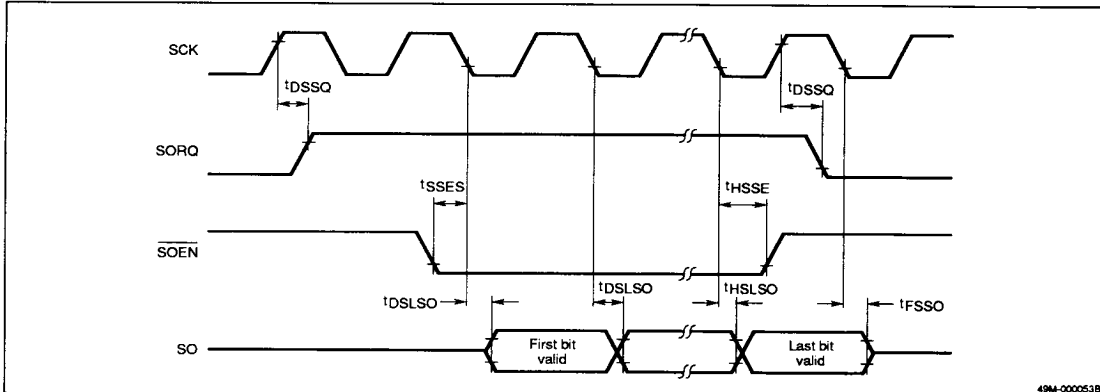
## SERIAL TIMING

### Serial Output Case 1: $\overline{SOEN}$ Asserted in Response to SORQ

Figure 8 shows timing for serial output when  $\overline{SOEN}$  is asserted in response to SORQ. If  $\overline{SOEN}$  is held inactive until after SORQ is asserted, and then  $\overline{SOEN}$  is asserted at least  $t_{SSES}$  before the falling edge of SCK, SO will become valid  $t_{DSLSO}$  after the falling edge of SCK for use by an external device at the subsequent rising edge of SCK.

Note that, although figure 8 shows  $\overline{SOEN}$  being asserted during a different SCK pulse than the one in which SORQ is asserted, it is permissible for these to occur during the same pulse of SCK as long as  $\overline{SOEN}$  is still asserted  $t_{SSES}$  before the falling edge of SCK.

Figure 8. Serial Output Case 1



49M-000053B

**Serial Output Case 2: SOEN Active Before SORQ Is High**

Figure 9 shows output timing when SOEN is active before SORQ is high. If SOEN is held active before SORQ is high, data will be shifted out whenever it becomes available in the serial output register (assuming previous data is already shifted out). In this case, SORQ will rise  $t_{DSSQ}$  after a rising edge of SCK. The first SO bit occurs  $t_{DLSO}$  after the next falling edge of SCK for use by an external device at the subsequent rising edge of SCK.

Subsequent bits will be shifted out  $t_{DLSO}$  after subsequent falling edges of SCK for use at subsequent rising edges of SCK. The last bit to be shifted out will also follow this pattern, and will be held valid  $t_{FSSO}$  after the

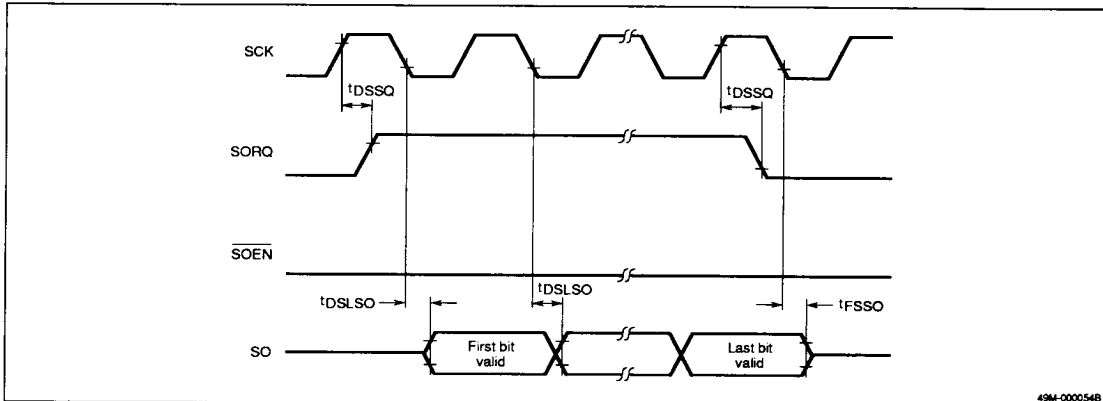
corresponding falling edge of SCK at which it is to be used. SORQ will be held  $t_{DSSQ}$  after this same rising edge of SCK, and then removed.

**Serial Output Case 3: SOEN Released During a Transfer**

If SOEN is released while SCK is in the middle of a transfer, as shown in figure 10, at least  $t_{HSSE}$  after the falling edge of SCK, then the next bit will be shifted out  $t_{DLSO}$  after the falling edge of SCK for use at the subsequent rising edge of SCK. SO will go inactive  $t_{FSSO}$  after the falling edge of SCK.

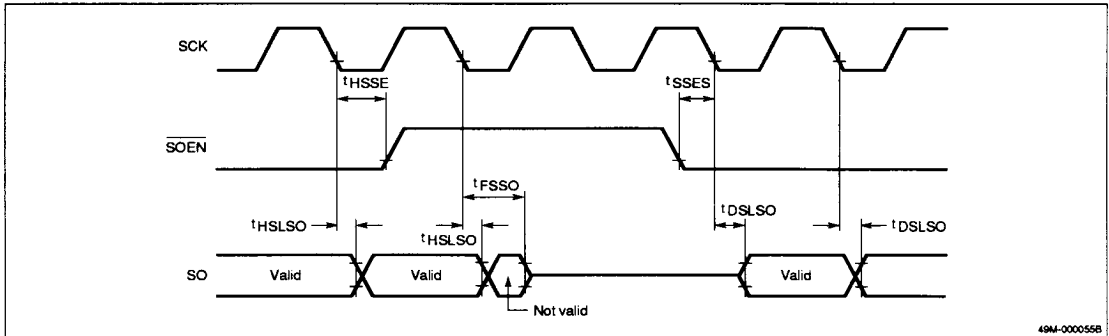
**Note:** For all its uses, SOEN must not change state within  $t_{SSSES}$  before or  $t_{HSSE}$  after the falling edge of SCK; otherwise the results will be indeterminate.

Figure 9. Serial Output Case 2



49M-000054B

**Figure 10. Serial Output Case 3**



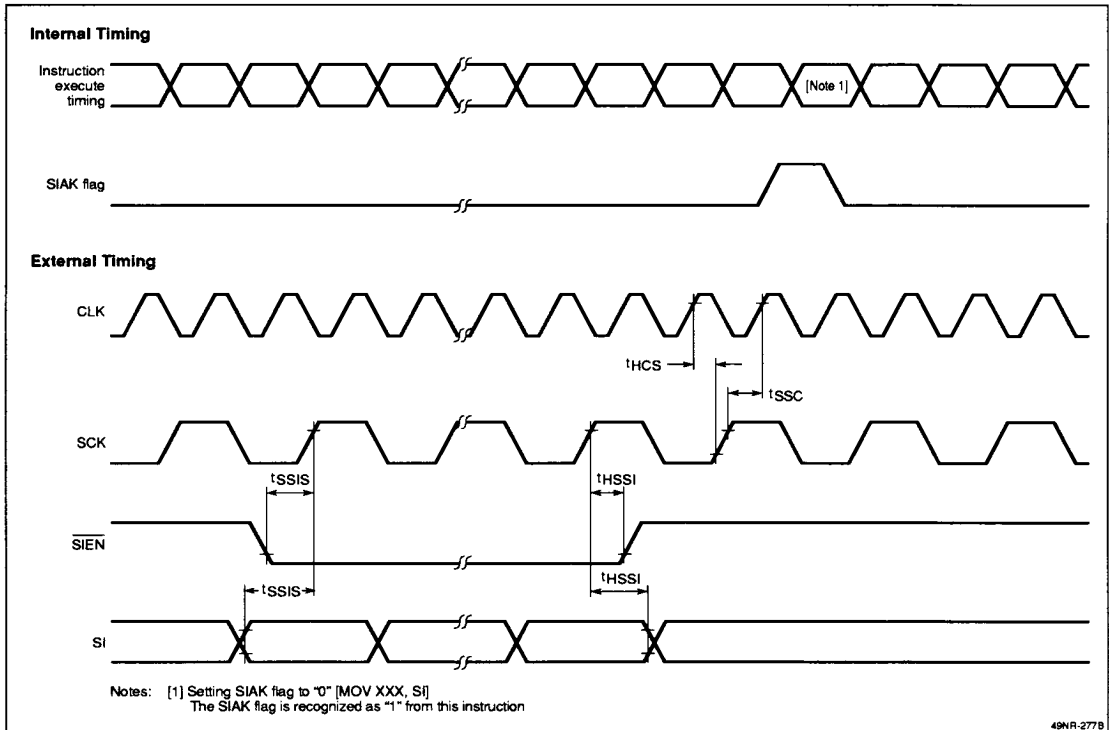
### Serial Input

Serial input timing (figure 11) is much simpler than serial output timing (figure 12). Data bits are shifted in on the rising edge of SCK if SIEN is asserted. Both SIEN

and SI must be stable at least  $t_{SSIS}$  before and  $t_{HSSI}$  after the rising edge of SCK; otherwise the results will be indeterminate.

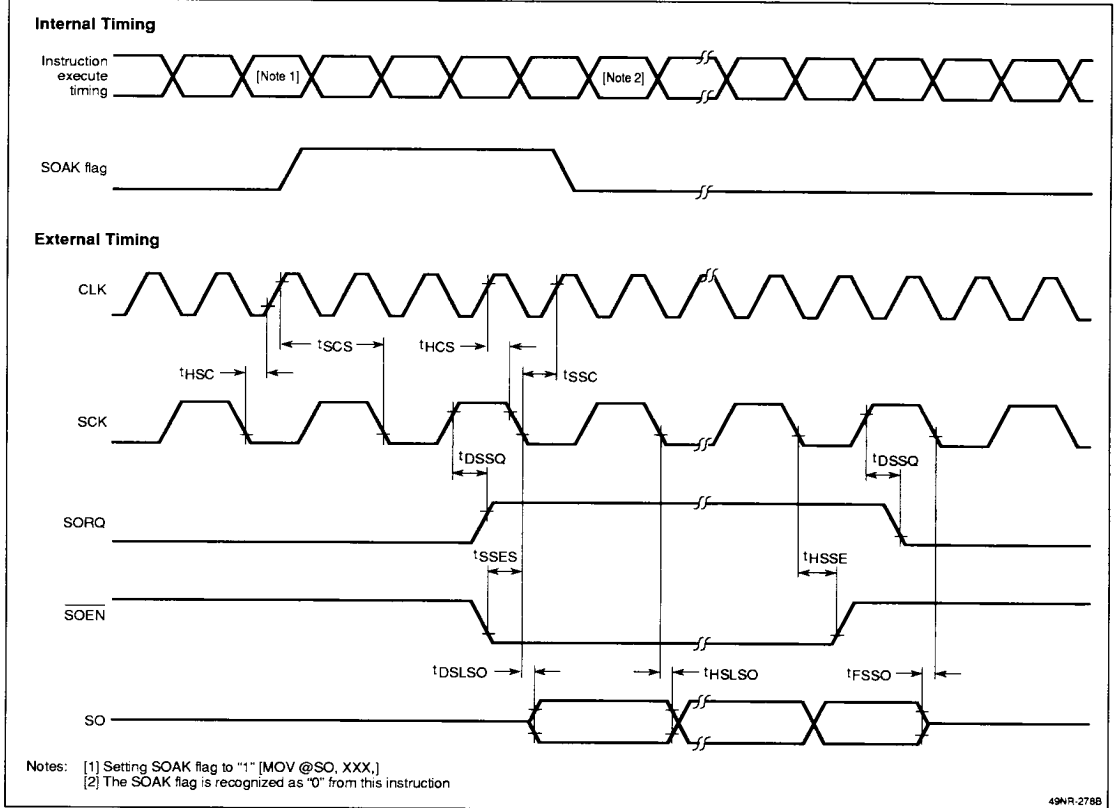
**3b**

**Figure 11. Serial Input Timing Example**



Notes: [1] Setting SIAK flag to "0" [MOV XXX, SI]  
The SIAK flag is recognized as "1" from this instruction

Figure 12. Serial Output Timing Example



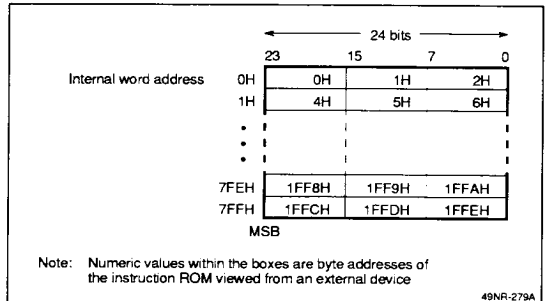
μPD77P25 PROM

The μPD77P25 has a PROM—one-time programmable (OTP) or ultraviolet erasable (UVE)—consisting of a 2K x 24-bit instruction ROM and a 1K x 16-bit data ROM.

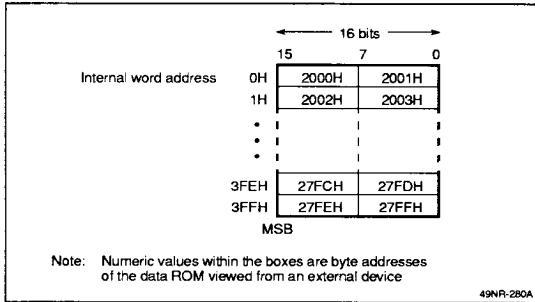
Data is written to or read from the PROM in 8-bit bytes. Because instruction words are 24 bits and data words are 16 bits, special byte addresses are assigned to the instruction ROM (0H-1FFH) and data ROM (2000H-27FFH) as shown in figures 13 and 14.

Each internal word address of the instruction ROM is equivalent to three byte addresses used by external devices plus one dummy byte address. For example, in figure 13, internal word address 0H corresponds to byte addresses 0H, 1H, and 2H plus dummy byte address 3H (not shown).

Figure 13. Instruction ROM



**Figure 14. Data ROM**



### UVEPROM Erasure

Data in a UVEPROM can be erased by exposure to light with a wavelength shorter than 400 nm. Usually, ultraviolet light with a 254-nm wavelength is used. The erasure process, which sets all data bits to 1's, must take place before data is programmed to a UVEPROM.

The total light quantity required to completely erase the written data is 15 Ws/cm<sup>2</sup>, equivalent to exposure to a UV lamp with a rating of 12,000 μW/cm<sup>2</sup> for about 20 minutes. A longer time may be necessary because of such factors as the age of the UV lamp and stains on the package window. The window must be positioned within 1 inch of the UV lamp.

If the UVEPROM is exposed to direct sunlight or fluorescent light for a long time, the data might be destroyed. To prevent this, mask the window with a cover or film after the erasure process.

### Data Programming Procedure

Following is the procedure for programming the 77P25. Table 15 shows the reassigned pin functions when writing/reading the PROM.

Since the area from byte address 2800H to 3FFFH is for internal testing, the area for the instruction ROM and data ROM must be set from byte address 0H to 27FH. Set the data to dummy byte addresses in the instruction ROM area to FFH in the normal programming.

- (1) Apply + 12.5 V to RST (pin 16), + 6 V to V<sub>DD</sub>, and + 12.5 V to V<sub>PP</sub>. This causes the PROM to enter program mode.
- (2) Specify the desired ROM byte address from address input pins A<sub>0</sub> to A<sub>13</sub>.
- (3) Program the data on the data bus (D<sub>0</sub>-D<sub>7</sub>) by applying 0 to  $\overline{CE}$  while  $\overline{OE}$  is 1 (program mode).

- (4) Output the programmed data to the data bus (D<sub>0</sub>-D<sub>7</sub>) by applying 0 to  $\overline{OE}$  while  $\overline{CE}$  is 1 (program verify mode).
- (5) Repeat steps 2 through 4, 25 times maximum until the data is properly written to the specified address.
- (6) After verifying that the data has been properly programmed, apply additional pulses by setting  $\overline{OE}$  to 1 (clear  $\overline{CE}$  to 0). The pulse width is 3X ms if the number of repetitions in steps 3 and 4 is X.

The above procedure completes writing one byte of data. If the data will not be properly programmed even after steps 2 to 4 have been repeated more than 25 times, the 77P25 is defective.

**Table 14. Pin Functions for PROM Programming/Reading**

Program Mode	Normal Mode	Function
A <sub>0</sub>	A <sub>0</sub>	Input address (viewed from external device) for programming/reading PROM (instruction ROM and data ROM).
A <sub>1</sub>	$\overline{WR}$	
A <sub>2</sub>	SORQ	
A <sub>3</sub>	SO	
A <sub>4</sub>	SI	
A <sub>5</sub>	$\overline{SOEN}$	
A <sub>6</sub>	$\overline{SIEN}$	
A <sub>7</sub>	SC1	
A <sub>8</sub>	INT	
A <sub>9</sub>	CLK	
A <sub>10</sub>	P <sub>1</sub>	
A <sub>11</sub>	P <sub>0</sub>	
A <sub>12</sub>	DRQ	
A <sub>13</sub>	$\overline{DACK}$	
D <sub>0</sub> -D <sub>7</sub>	D <sub>0</sub> -D <sub>7</sub>	Input/output data for PROM (instruction ROM and data ROM)
$\overline{CE}$	$\overline{CS}$	PROM program strobe signal (active low)
$\overline{OE}$	$\overline{RD}$	PROM read strobe signal (active low)
V <sub>PP</sub>	V <sub>PP</sub>	Power pin for programming PROM; apply -12.5 V for writing and +5 V for reading.
V <sub>DD</sub>	V <sub>DD</sub>	Power pin; apply +6 V for programming and -5 V for reading.
GND	GND	Ground pin
	RST	Sets PROM program or read mode. Mode is set when +12.5 V is applied.

Data Reading Procedure

- (1) Apply +12.5 V to RST, +5.0 V to VDD, and +5.0 V to VPP. This causes the PROM to enter read mode.
- (2) Specify the desired ROM byte address from the address input pins A0 to A13.
- (3) Data will be output to the data bus (D0-D7) by clearing OE and CE to 0.

Instruction ROM Code Protection

A word of the instruction ROM can be protected if data FEH is programmed to a dummy byte address. For example, byte addresses 0H, 1H, and 2H (word address 0H) are protected if FEH is programmed to dummy byte address 3H. Following is the procedure for protecting the instruction ROM.

- (1) Set data FFH to the dummy addresses; then perform the data program procedure.
- (2) Verify the programmed data by the data read procedure.
- (3) Set data FEH to the dummy addresses; again perform the data program procedure.

DEVELOPMENT TOOLS

For software development and assembly into object code, a relocatable assembler (RA77C25) is available. This software is available to run on MS-DOS®, CP/M®, VAX®/VMS®, and VAX/UNIX® systems.

For debugging, a hardware emulator (EVAKIT-77C25) provides in-circuit, real-time emulation of the 77C25. Features of the EVAKIT-77C25 include break/step emulation, symbolic debugging, and on-line assembly/disassembly of code.

The EVAKIT-77C25 connects via a probe to the target system for test and demonstration of the final system design. It also connects to the host development system via an RS-232 port. Using Kermit or NEC's EVA communications program, code can be downloaded or uploaded between development system and EVAKIT.

MS-DOS is a registered trademark of Microsoft Corporation. CP/M is a registered trademark of Digital Research, Incorporated. VAX and VMS are registered trademarks of Digital Equipment Corporation. UNIX is a registered trademark of UNIX System Laboratories, Incorporated.

By connecting to a PROM programmer, the EVAKIT is also used to prepare 77P25 PROMs intended for prototyping and small volume applications. A program adaptor, PA-77P25, is provided for use with the data I/O programmer.

Code submittal for the mask ROM μPD77C25 is accomplished by preparing a 27C256A or μPD77P25 PROM using the same programming device.

SYSTEM CONFIGURATION

Figures 15, 16, 17, and 18 show typical system applications for the 77C25.

Figure 15. Spectrum Analysis System

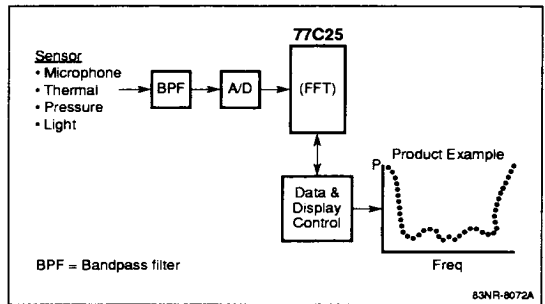
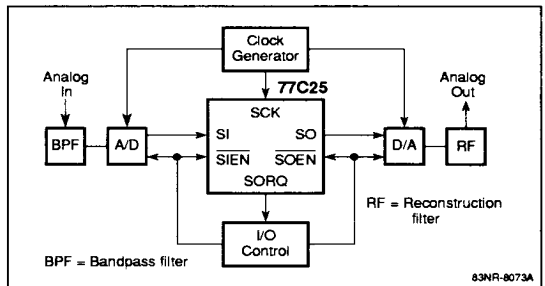
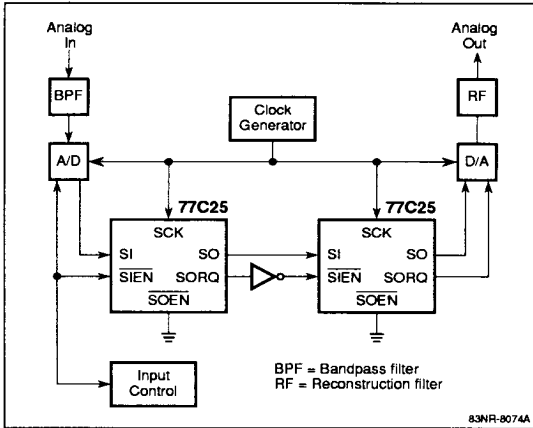


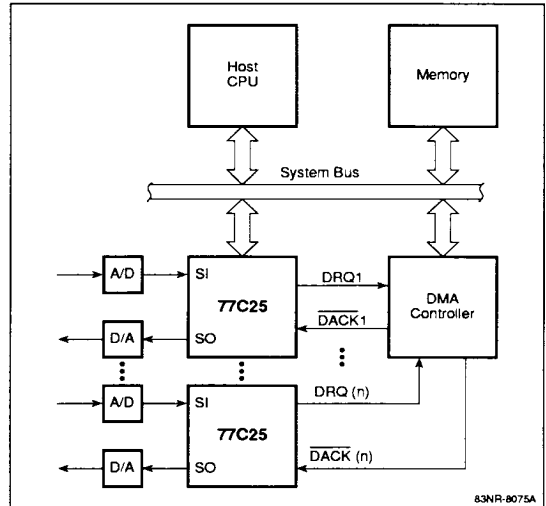
Figure 16. Analog-to-Analog Digital Processing System Using a Single 77C25



**Figure 17. Signal Processing System Using Cascaded 77C25's and Serial Communication**



**Figure 18. Signal Processing System Using 77C25's As a Complex Computer Peripheral**



3b