82C235 Single Chip AT

1 Introduction

The 82C235 is a VLSI device that incorporates most of the motherboard logic required to build a low-cost, highly-integrated, IBM PC AT-compatible computer. It is designed to be used in conjunction with other Chips and Technologies controllers such as the 82C45X VGA Controller, the 82C601 Multifunction Controller, the 82C765 Floppy Disk Controller, and the 82C710 Integrated Floppy Disk and Multifunction Controller. When used with these devices, the 82C235 acts as the heart of a highly integrated system that significantly reduces motherboard size, component count, and the need for many I/O channel slots.

1.1 Features

The 82C235 provides the following features:

- 80286 control logic and clocks that support CPU speeds of up to 12.5MHz with zero (or one) wait-states
- A 146818-compatible real time clock with 114 bytes of CMOS RAM
- Two 8237-compatible DMA controllers
- Two 8259-compatible interrupt controllers
- An 8254-compatible programmable interval timer
- An 8255-compatible programmable peripheral interface
- An 82284-compatible clock generation and READY interface
- An 82288-compatible bus controller
- A DRAM controller that supports up to 8MB of DRAM (up to 16MB with the addition of an external '538-type decoder)
- A memory controller that provides shadow RAM and support for either 8-bit or 16-bit BIOS ROM
- A DRAM refresh controller
- 32 EMS page registers (LIM EMS 4.0-compatible)
- Interface logic for an 80287 numeric coprocessor
- Interface logic for an 8042 keyboard controller
- Fast Gate A20 and Fast CPU Reset logic
- Power management features
- Compact packaging in a single 160-pin plastic flat pack (160PFP)

Figure 1.1 is a system block diagram of the 82C235.

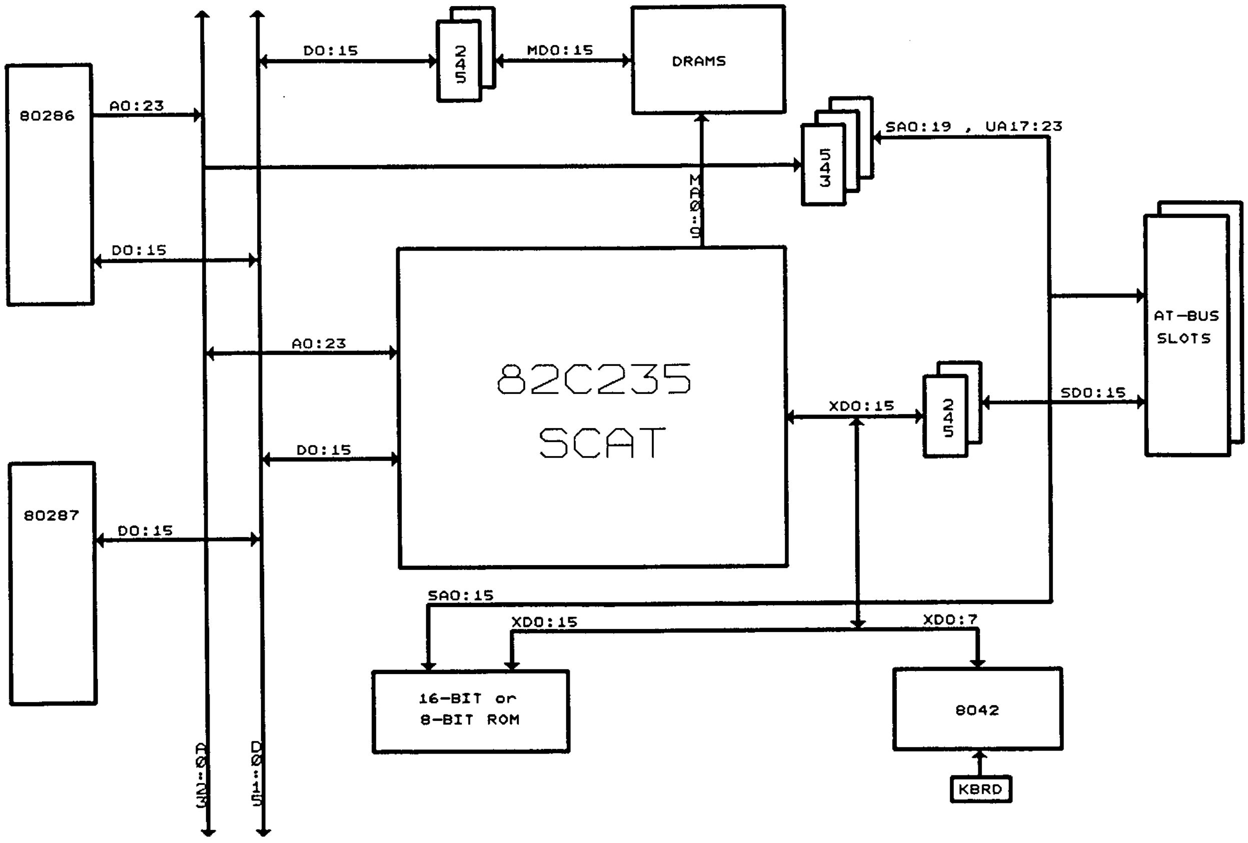


Figure 1.1 82C235 System Block Diagram

2 Functional Description

This section of the data manual provides functional descriptions of the following sections of the 82C235:

- Clock generation
- Bus control
- Local bus arbitration
- DMA controller
- Interrupt controller
- Programmable interrupt timer
- Keyboard/mouse interface
- Memory interface
- Numeric processor interface
- Real-time clock interface
- I/O channel interface
- Power management

2.1 Clock Generation

An 82C235 clock generator produces the following signals (refer to Figure 1.2):

- **PROCCLK**--(processor clock) is an output clock for the 80286 processor. PROCCLK can be set (with internal configuration register 46H, bits 3-2) for the following values:

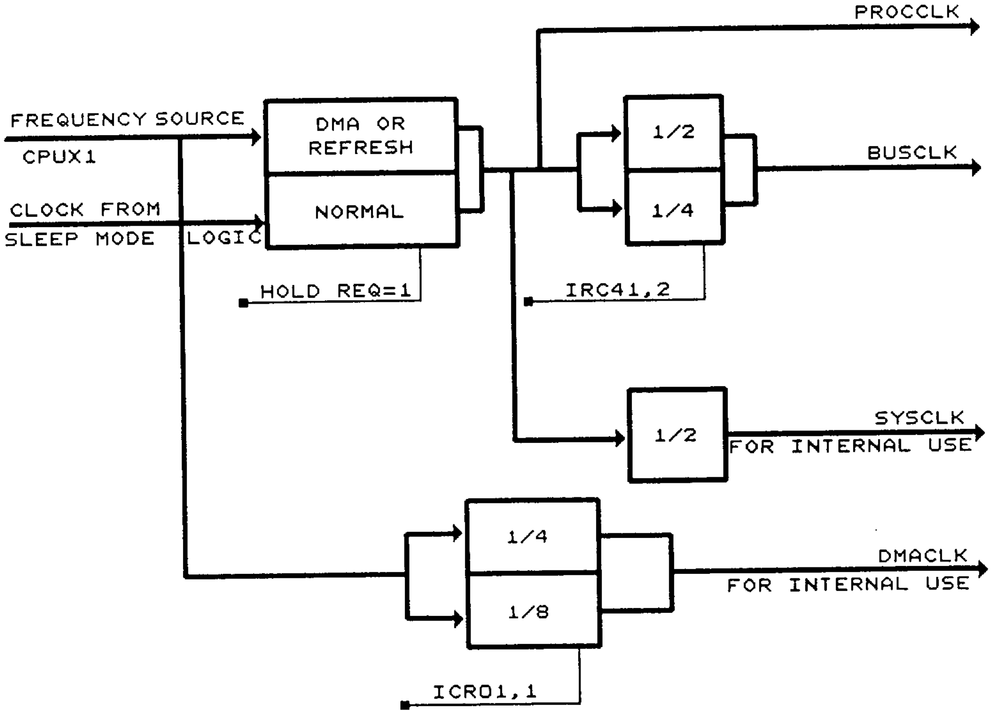


Figure 1.2 Clock Generation

PROCCLK = CPUX1
PROCCLK = CPUX1/2
PROCCLK = CPUX1/4
PROCCLK = CPUX1/8

PROCCLK can also be halted in order to implement power management techniques (refer to section 2.12 on power management for further information). The frequency chosen for PROCCLK must be twice the frequency of the 80286 processor; e.g., for an 80286 with a speed of 12MHz, PROCCLK must equal 24MHz. PROCCLK is derived in one of two ways: from an external crystal oscillator connected to CPUX1 (in which case CPUX2 is left unconnected), or from an external crystal connected across CPUX1 and CPUX2.

o **BUSCLK**--(bus clock) is an output clock for the I/O channel. BUSCLK can be set (with internal configuration register 41H, bit 2) for the following values:

BUSCLK = PROCCLK/2
BUSCLK = PROCCLK/4

o **OSC**--(oscillator) is a 14.31818 MHz output signal used by the I/O channel. The oscillator can be derived either from an external crystal connected to pins OSCX1 and OSCX2, or from an external oscillator connected to OSCX1 (OSCX2 is left unconnected).

o **DMACLK**--(DMA clock) is an internal clock used by DMA controllers to time DMA operations. DMACLK can be set (with internal configuration register 01H, bit 1) to the following values:

DMACLK = CPUX1/2
DMACLK = CPUX1/4

o **SYSCLK**--(system clock) is an internal clock used by 82C235 logic. The frequency of SYSCLK is:

SYSCLK = PROCCLK/2

2.2 Reset Strap Option

The 82C235 features a reset strap option. This option is employed to provide strap options without having to use pins. The 82C235 uses DACK* lines to implement the feature. The DACK* lines are normally outputs, but during power-up situations they are used as input signals. These lines are sampled when POWERGOOD goes high. Later, they convert to outputs for normal functions. A 4.7K ohm pull-down resistor pulls the signals low for strap option sampling, while a 4.7K ohm pull-up resistor pulls signal levels high for the same purpose. The following table describes the pins that are used in the strap option:

Pin #	Normal Pin	Strap Pin	Comments
89	DACK0*	SENSE0	Reflects at ICR 45-0
47	DACK1*	SENSE1	Reflects at ICR 45-1
45	DACK2*	SENSE2	Reflects at ICR 45-2
43	DACK4*	Reserved	
91	DACK5*	EXRTC*	Internal or external RTC
93	DACK6*	Reserved	
96	DACK7*	16BITROM*	16-bit or 8-bit ROM

The sense lines are general purpose and can be used as per the specific system requirements.

2.3 Bus Control

This section describes the data and address buses shown in Figure 1.1, the 82C235 Block Diagram.

82C235 Pins	Bus Name	Description
D15-D0	Local Data	A 16-bit bidirectional bus connecting the 82C235, the 80286, and the optional 80287 to local memory or to the memory data (MD) buffers.
MD15-MD0	Memory Data	A 16-bit bidirectional bus connecting local DRAM and the memory data (MD) buffers.
XD15-XD0	X-Data	A 16-bit bidirectional bus connecting the 82C235, ROM, XD-bus peripherals, and S-Data (SD) bus buffer.
SD15-SD0	S-Data	A 16-bit bidirectional bus connecting the I/O channel and the S-Data (SD) buffer.
A23-A00	Local Address	A 24-bit address bus, driven by either the 80286 or the 82C235.
SA19-SA0	S-Address	A latched, local address bus connecting the I/O channel to the XD-bus peripherals.
UA23-UA17	Unlatched Address	An unlatched address bus connected to the I/O channel. **NOTE**--This bus must be latched except during quick mode.
MA9-MA0	Multiplexed Address	A multiplexed address bus that is driven by the 82C235 to address DRAM.

2.3.1 Bus Controller

The internal bus controller, which is functionally similar to an 82288 bus controller, provides command generation and timing control for the AT-compatible I/O channel. To allow the processor to run faster than the I/O channel, DMA commands, timing, and accesses to the I/O channel can be programmed to run slower than local bus cycles.

2.3.2 Local Bus Arbitration

Although the local bus is normally controlled by the 80286, the 82C235's internal DMA controller or refresh controller can request control by issuing a HOLD request to the 80286. When this occurs, the 80286 relinquishes control and issues HLDA (hold acknowledgement) to the 82C235.

2.4 DMA Controller

Memory refresh and DMA functions are implemented within the 82C235 as independent bus masters. The 82C235 arbitrates these functions with internal logic, and gains control of the local bus via the HOLD/HLDA protocol of the 80286. The 82C235 contains two DMA controllers that are compatible with the Intel 8237. Each controller is a four- channel DMA device that can generate the memory addresses and control signals necessary to transfer information between a peripheral device and memory directly. This allows high-speed information transfer with little CPU intervention.

The two DMA controllers are internally cascaded to provide four DMA channels for transfers to 8-bit peripherals (DMA1) and three channels for transfers to 16-bit peripherals (DMA2). DMA2 Channel 0 provides the cascade interconnection for the two DMA devices, thereby maintaining IBM PC AT compatibility.

Figure 1.3 shows how the two controllers are cascaded. DMA channels 0-3 are used for 8-bit transfers, while channels 5-7 are used for 16-bit transfers. DMA operations are allowed within the full range of 16MB memory through the use of DMA page registers.

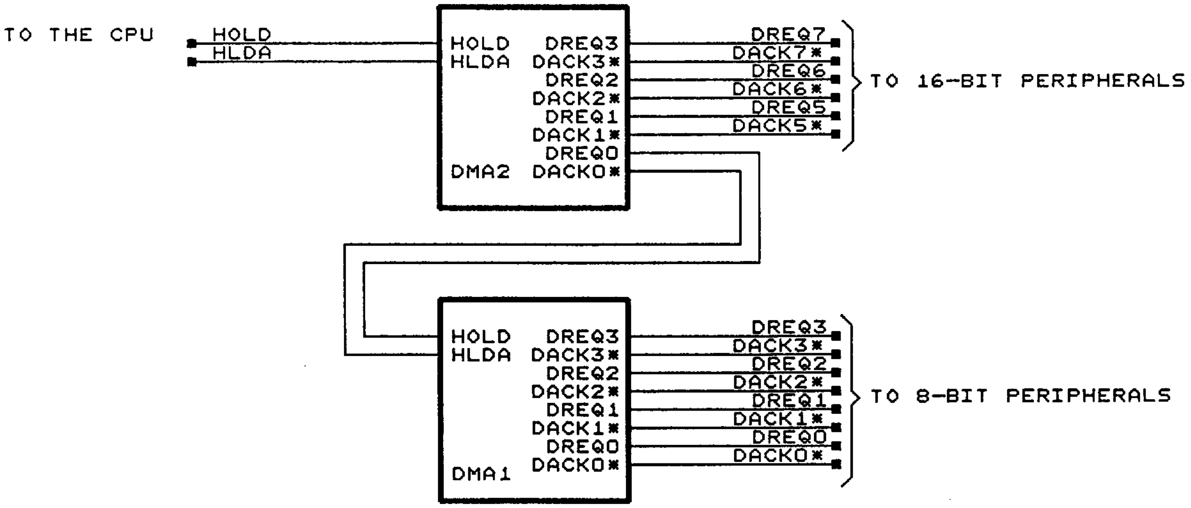


Figure 1.3 Cascaded DMA Controllers

The DMA clock, which controls DMA transfer rate and timing, is programmable. Its frequency is selectable for a rate equal to CPUX1/2 or CPUX1/4. (This is under control of internal configuration register 01H, bit 1). Wait-states during DMA operations are programmed with internal configuration register 01H, bits 5-2.

Refresh, which occurs at 15us intervals (nominally), is determined by counting down on the 1.19 MHz (OSCX1/12) clock. The refresh period is set by programming timer channel 1 of the 8254 timer/counter.

Table 1.1 shows use of DMA levels on the I/O channel. The DMA requests are shown by priority, starting with the highest level.

Table 1.1 DMA on the I/O Channel

DMA Level	System Board	I/O Channel
DRQ0	Not Used	Available
DRQ1	Not Used	Available
DRQ2	Not Used	Diskette Drive
DRQ3	Not Used	Available
DRQ5	Not Used	Available
DRQ6	Not Used	Available
DRQ7	Not Used	Available

DMA cycle length control is provided internally in the 82C235 allowing independent control for both 8-bit and 16-bit cycles. This is done throughout the programmable registers, which can extend command signals or insert wait states.

Each DMA channel has a pair of 16-bit counters and a reload register for each counter. The 16-bit counters allow the DMA to transfer blocks as large as 65,536 words. The register associated with each counter allows the channel to re-initialize without reprogramming. The following description of the DMA subsystem pertains to both DMA1 and DMA2 unless otherwise noted.

2.4.1 DMA Operation

During normal operation of the 82C235, the DMA subsystem is in either an Idle condition, a Program condition, or an Active condition. In the Idle condition the DMA controller executes cycles consisting of only one state. The Idle state, SI, is the default condition, and the DMA remains in this condition unless the device has been initialized and one of the DMA requests is active or the CPU attempts to access one of the internal registers.

When a DMA request becomes active, the device enters the Active condition and issues a hold request to the system. Once in the Active condition, the 82C235 generates the necessary memory addresses and command signals to accomplish a memory-to-I/O, I/O-to-memory, or a memory-to-memory transfer. Memory-to-I/O and I/O-to-memory transfers take place in one cycle while memory-to-memory transfers require two cycles. During transfers between memory and I/O, data is presented on the system bus by either memory or the requesting device and the transfer is completed in one cycle. Memory-to-memory transfers, however, require that the DMA store data from the read operation in an internal register. The contents of this register is then written to memory on the subsequent cycle.

During transfers between memory and I/O, two commands are activated during the same cycle. In the case of a memory-to-I/O transfer, the 82C235 asserts both DMAMEMR* and IOW* allowing data to be transferred directly to the requesting device from memory. Note that the 82C235 neither latches data from nor drives data out on this type of cycle.

The number of clock cycles required to transfer a word of data may be varied by programming the DMA or optionally extended by the peripheral device. During an Active cycle the DMA sequences through a series of states. Each state is one DMA clock cycle length, and the number of states in a cycle varies depending on how the device is programmed and the type of cycle being performed. The states are labeled S0-S4. They are explained in detail in the section called Active Condition.

2.4.2 Idle Condition

When no device is requesting service, the DMA is in an Idle condition that maintains the state machine in the SI state. During this time, the 82C235 samples the DREQ input pins every clock cycle. The internal select from the top level decoder and HLDA are also sampled at the same time to determine if the CPU is attempting to access the internal registers. When either these situations occurs, the DMA exits the Idle condition. Note that the Program condition has priority over the Active condition since a CPU cycle has already started.

2.4.3 Program Condition

The Program condition is entered whenever HLDA is inactive and internal select is active. The internal select is derived from the top-level decode described earlier. During this time, address lines A0-A3 become inputs if DMA1 is selected, or A1-A4 become inputs if DMA2 is selected. **NOTE**--When DMA2 is selected, A0 is ignored. These address inputs are used to select the DMA controller registers that are to be read or written to. Due to the large number of internal registers in the DMA subsystem, an internal flip-flop is used to supplement the addressing of the count and address registers. This bit is used to select between the high and low bytes of these registers. The flop-flop toggles each time a read or write occurs to any of the word count or address registers in the DMA. The internal flop-flop is cleared by a hardware RESET or a Master Clear command and may be set or cleared by the CPU issuing the appropriate command.

Special commands are supported by the DMA subsystem in the Program condition to control the device. These commands do not make use of the data bus but are derived from a set of addresses, the internal select, and IOW* or IOR*. These commands are Master Clear, Clear Register, Clear Mode Register Counter, Set, and Clear Byte Pointer Flip-Flop.

The 82C235 enables programming when ever HLDA has been inactive for one DMA clock cycle. It is the responsibility of the system to ensure that programming and HLDA are mutually exclusive. Erratic operation of the 82C235 can occur if a request for service occurs on an unmasked channel that is being programmed. The channel should be masked or the DMA disabled to prevent the 82C235 attempting to service a device with a channel which is partially programmed.

2.4.4 Active Condition

The 82C235 DMA subsystem enters the Active condition whenever a software request occurs or a DMA request on an unmasked channel occurs and the device is not in the Program condition. The 82C235 then begins a DMA transfer cycle.

In a read cycle for example, after receiving a DREQ, the 82C235 issues an HRQ to the system. Until an HLDA is returned, the DMA remains in an idle condition. On the next clock cycle, the DMA exits Idle and enters state S0. During S0, the device resolves priority and issues DACK on the highest priority channel requesting service. The DMA then proceeds to state S1, where the multiplexed addresses are output and latched. State S2 is then entered, at which time the 82C235 asserts DMAMEMR*. The device then transitions into S3, where the IOW* command is asserted. The 82C235 then remains in S3 until the wait-state counter has decremented to zero and IOCHRDY is true. Note that at least one additional S3 occurs unless Compressed Timing is selected. Once a ready condition is detected, the DMA enters S4, where both commands are de-asserted. In Burst Mode and Demand Mode (discussed below), subsequent cycles begin in S2 unless the intermediate addresses require updating. In these subsequent cycles, the lower addresses are changed in S2. The DMA can be programmed on a channel-by-channel basis to operate in one of four modes. The four modes are described below.

o **Single transfer Mode**--This mode directs the DMA to execute only one transfer cycle at a time. DREQ must be held active until DACK becomes active. If DREQ is held active throughout the cycle, the 82C235 de-asserts HRQ and releases the bus after the transfer is complete. After HLDA is inactive, the 82C235 again asserts HRQ and executes another cycle on the same channel unless a request from a higher priority channel is received. In this mode, the CPU is allowed to execute at least one bus cycle between transfers.

Following each transfer, the word count is decremented and the address is incremented or decremented. When the word count decrements from 0000H to FFFFH, the terminal count bit in the status register is set and a T/C pulse is generated. If the auto-initialization option is enabled, the channel re-initializes itself. If Auto-initialize is not selected, the DMA sets the DMA request bit mask and suspends transferring on the channel.

Block Transfer Mode--When Block Transfer Mode is selected, the 82C235 begins transfers in response to either a DREQ or a software request. This continues until a terminal count (FFFFH) is reached, at which time T/C is pulsed and the status register terminal count bit is set. In this mode DREQ need only be held active until DACK is asserted. Auto-initialization is operational in this mode also.

Demand Transfer Mode--In Demand Transfer mode, the DMA begins transfers in response to the assertion of DREQ and continues until either terminal count is reached or DREQ becomes inactive. This mode is normally used for peripherals that have limited buffering availability. The peripheral can initiate a transfer and continue until its buffer capacity is exhausted. The peripheral may re-establish service by again asserting DREQ. During idle periods between transfers, the CPU is released to operate and can monitor the operation by reading intermediate values from the address and word count registers. Once DREQ is de-asserted, higher priority channels are allowed to intervene. Reaching terminal count results in the generation of a T/C pulse, the setting of the terminal count bit in the status register, and auto-initialization (if enabled.)

Cascade Mode--This mode is used to interconnect more than one DMA controller, to extend the number of DMA channels while preserving the priority chain. In Cascade mode, the master DMA controller does not generate address or control signals. The DREQ and DACK signals of the master are used to interface the HRQ and HLDA signals of the slave DMA devices. Once the master has received an HLDA from the CPU in response to a DREQ caused by the HRQ from a slave DMA Controller, the master DMA controller ignores all inputs except HLDA from the CPU and DREQ on the active channel. This prevents conflicts between the DMA devices.

Figure 1.3 shows the cascade interconnection between the two levels of DMA devices. Note that Channel 0 of DMA2 is internally connected for Cascade mode to DMA1. Additional devices can be cascaded to the available channels in either DMA1 or DMA2 since cascade is not limited to two levels for DMA controllers.

When programming cascaded controllers, begin with the device which is actually generating HRQ to the system (first level device) and then proceed to the second level devices. RESET causes the DACK outputs to become active low and are placed into an inactive state. To allow the internal cascade between DMA1 and DMA2 to function correctly, the active low state of DACK should not be modified. This is because the 82C235 has an invertor between DACK0 of DMA2 and HLDA of DMA1. The first level device's DMA request mask bit prevents second level cascaded devices from generating unwanted hold requests during the initialization process.

2.4.5 DMA Transfers

Four types of transfer modes are provided in the 82C235 DMA subsystem. These transfer types are:

o **Read Transfer**--Read transfers move data from memory to an I/O device by generating the memory address and asserting DMAMEMR* and IOW* during the same cycle.

- **Write Transfer**--Write transfers move data from an I/O device to memory by generating the memory address and asserting IOR* and DMAMEMW.

- **Memory-to-Memory Transfer**--The memory-to-memory transfer is used to move a block of memory from one location in memory to another. DMA channels 0 and 1 may be programmed to operate as memory-to-memory channels by setting a bit in the Command Register. Once programmed to perform a memory-to-memory transfer, the process can be started by generating either a software or an external request to channel 0. Once the transfer is initiated, Channel 0 provides the address for the source block during the memory read portion of the cycle. Channel 1 generates the address for the memory write cycle. During the read cycle, a byte of data is latched in the internal Temporary Register of the 82C235. The contents of this register are then output on the XD0-7 data lines during the write portion of the cycle and subsequently written to memory. Channel 0 may be programmed to maintain the same source address on every cycle. This allows the CPU to initialize large blocks of memory with the same value. The 82C235 continues to perform transfer cycles until Channel 1 reaches terminal count.

- **Verify Transfer**--The verify transfer is a pseudo-transfer that is useful for diagnostics. In this type of transfer, the DMA operates as if it is performing a Read or Write Transfer by generating HRQ, addresses, and DACK, but does so without asserting a command signal. Since no transfer actually takes place, IOCHRDY is ignored during Verify transfer cycles.

2.4.6 Auto-Initialization

Each of the four DMA channel Mode Registers contains a bit that causes the channel to re-initialize after reaching terminal count. During this process, referred to as Auto-Initialization, the Base Address and Base Word Count Registers, which were originally written by the CPU, are reloaded into the Current Address and Current Word Count Registers (both the base and current registers are loaded during a CPU write cycle). The base register remains unchanged during DMA Active cycles and can only be changed by the CPU. If the channel is programmed to auto-initialize, the request mask bit is not set upon reaching terminal count. This allows the DMA to continue operation without CPU intervention.

During memory-to-memory transfers, the Word Count Registers of both Channel 0 and Channel 1 must be programmed with the same starting value for full auto-initialization. If Channel 0 reaches terminal count before Channel 1, then Channel 0 and Channel 1 must be programmed with the same starting value for full auto-initialization. If Channel 0 reaches terminal count before Channel 1, then Channel 0 reloads the starting address and word count, and continues transferring data from the beginning of the source block. Should Channel 1 reach terminal count first, it reloads the current registers and Channel 0 remains un-initialized.

2.4.7 DREQ Priority

The 82C235 supports two schemes for establishing DREQ priority. The first is fixed priority, which assigns priority based on channel position. In this method Channel 0 is assigned the highest priority. Priority assignment then progresses in order down through the channels, with Channel 3 receiving the lowest priority.

The second type of priority assignment is rotating priority. In this scheme the ordering of priority from Channel 0 to Channel 3 is maintained but the actual assignment of priority changes. The channel most recently serviced is assigned the lowest priority and, since the order of priority assignment remains fixed, the remaining three channels rotate accordingly. The rotating priority assignment is illustrated in Figure 3.6.

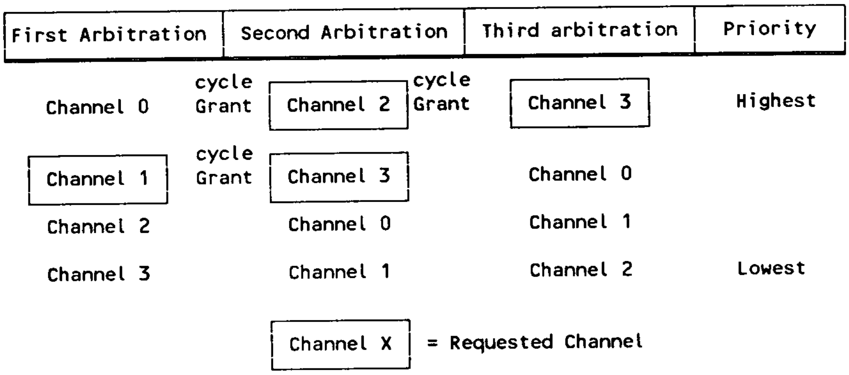


Figure 3.6. Rotating Priority Scheme

In instances where multiple requests occur at the same time, the 82C235 issues an HRQ but does not freeze the priority logic until HLDA is returned. Once HDLA becomes active, the priority logic is frozen and DACK is asserted on the highest requesting channel. Priority is not reevaluated until HLDA is deactivated.

2.4.8 Address Generation

The DMA Page Register is a set of 16 8-bit registers in the 82C235 that are used to generate the high order addresses during DMA cycles. Only eight of the registers are actually used, but all sixteen are included to maintain IBM PC/AT compatibility. Each DMA channel has a register associated with it with the exception of Channel 0 of DMA2, which is used for internal cascading to DMA1. Assignment of each of these registers is shown in Figure 3.7 along with its Read/Write address.

During Demand and Block Transfers, the 82C235 generates multiple sequential transfers. For most of these transfers the information in the external address latches remains the same, eliminating the need to be re-latched. Since the need to update the latches occurs only when a carry or borrow from the lower 8-bits of the Address Counter exists, the 82C235 updates the latch contents only when necessary. The 82C235 therefore executes S1 cycles only when necessary, resulting in an overall through-put improvement.

Address	Register Function
080H	Unused
081H	8-bit DMA Channel 2 (DACK2)
082H	8-bit DMA Channel 3 (DACK3)
083H	8-bit DMA Channel 1 (DACK1)
084H	Unused
085H	Unused
086H	Unused
087H	8-bit DMA Channel 0 (DACK0)
088H	Unused
089H	16-bit DMA Channel 2 (DACK6)
08AH	16-bit DMA Channel 3 (DACK7)
08BH	16-bit DMA Channel 1 (DACK5)
08CH	Unused
08DH	Unused
08EH	Unused
08FH	Refresh Cycle

Figure 3.7. DMA Address Extension Map

2.4.9 Compressed Timing

The DMA subsystem in the 82C235 can be programmed to transfer a word in as few as three DMA clock cycles. The normal DMA cycle consists of three states: S2, S3, and S4 (this assumes Demand or Block Transfer Mode). Normal transfers require four DMA clock cycles since S3 is executed twice due to the one wait-state insertion. In systems capable of supporting high through-put, the 82C235 can be programmed to omit one S3 and assert both commands in S2. S2 begins the cycle by generating the address and asserting both commands. One S3 cycle is executed and the cycle terminates in S4. If Compressed Timing is selected, T/C is output in S2 and S1 cycles are executed as necessary to update the address latch. Note that Compressed Timing is not allowed for memory-to-memory transfers.

2.4.10 Register Descriptions

This section describes the registers used during DMA functions.

2.4.10.1 Current Address Register

Each DMA channel has a 16-bit Current Address Register that holds the address used during transfers. Each channel can be programmed to increment or decrement this register whenever a transfer is completed. This register can be read or written by the CPU in consecutive 8-bit bytes. If Auto-Initialization is selected, this register is reloaded from the Base Address Register upon reaching terminal count in the Current Word Count Register. Channel 0 can be prevented from incrementing or decrementing by setting the Address Hold Bit in the Command Register.

2.4.10.2 Current Word Count Register

Each channel has a Current Word Count Register that determines the number of transfers to perform. The actual number of transfers performed is one greater than the value programmed into the register. The register is decremented after each transfer until it goes from zero to FFFFH. When this roll-over occurs, the 82C235 generates T/C, suspends operation on that channel, sets the appropriate Request

Mask Bit or Auto-Initialize, and continues.

2.4.10.3 Base Word Count Register

This register preserves the initial value of the Current Word Count Register. It is also a write-only register that is loaded by writing to the Current Word Count Register. This register is loaded in the Current Word Count Register during Auto-Initialization.

2.4.10.4 Command Register

This register controls the overall operation of a DMA subsystem. The register can be read or written to by the CPU and is cleared by either a RESET or a Master Clear command.

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
DAK	DRQ	EW	RP	CT	CD	AH	M-M

- **DAK**--DACK active level is determined by bit 7. Programming a one in this bit position makes DACK an active high signal.

- **DRQ**--DREQ active level is determined by bit 6. Writing a one in this bit position causes DREQ to become active low.

- **EW**--Extended Write is enabled by writing a one to bit 5, causing the write commands to be asserted one DMA cycle earlier during a transfer. The read and write commands both begin in state S2 when enabled.

- **RP**--Writing a one to bit 4 causes the 82C235 to utilize a rotating priority scheme for honoring DMA requests. The default condition is fixed priority.

- **CT**--Compressed timing is enabled by writing a one to bit 3 of this register. The default 0 condition causes the DMA to operate with normal timing.

- **CD**--Bit 2 is the master disable for the DMA controller. Writing a one to this location disables the DMA subsystem (DMA1 or DMA2). This function is normally used whenever the CPU needs to reprogram one of the channels to prevent DMA cycles from occurring.

- **AH**--Writing a one to bit 1 enables the address hold feature in Channel 0 when performing memory-to-memory transfers.

- **M-M**--A one in the bit 0 position enables Channel 0 and Channel 1 to be used for memory-to-memory transfers.

2.4.10.5 Mode Register

Each DMA channel has a Mode Register associated with it. All four Mode Registers reside at the same I/O address. Bits 0 and 1 of the Write Mode Register command determine which channel's Mode Register is written to. The remaining six bits control the mode of the selected channel. Each channel's Mode Register can be read by sequentially reading the Mode Register location. A Clear Mode Register Counter command is provided to allow the CPU to restart the mode read process at a known point.

During mode read operation, bits 0 and 1 are one.

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
M1	M0	DEC	AI	TT1	TT0	CS1	CS0

o **M1-M0**--Mode selection for each channel is accomplished by bits 6 and 7.

M1	M0	MODE
0	0	Demand Mode
0	1	Single Cycle Mode
1	0	Block Mode
1	1	Cascade Mode

o **DEC**--Determines direction of the address counter. A one in bit 5 decrements the address after each transfer.

o **A1**--The Auto-Initialization function is enabled by writing a one in bit 4 of the Mode Register.

o **TT1-TT0**--Bits 2 and 3 control the type of transfer that is to be performed.

TT1	TT0	TYPE
0	0	Verify Transfer
0	1	Write Transfer
1	0	Read Transfer
1	1	Illegal

o **CS1-CS0**--Channel Select bits 1 and 0 determine which channel's Mode Register is written to. Read back of a mode register results in bits 1 and 0 both being ones.

CS1	CS0	CHANNEL
0	0	Channel 0 Select
0	1	Channel 1 Select
1	0	Channel 2 Select
1	1	Channel 3 Select

2.4.10.6 Request Register

This is four bit register used to generate software requests (DMA service can be requested either externally or under software control). Request Register bits can be set or reset independently by the CPU. The Request Mask has no effect on software generated requests. All four bits are read in one operation and appear in the lower four bits of the byte. Bits 4 through 7 are read as ones. All four request bits are cleared to zero by a RESET.

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
X	X	X	X	X	RB	RS1	RS0

(Write Operation)

- o **RB**--The request bit is set by writing a one to bit 2. RS1-RS0 select which bit (channel) is to be manipulated.

- o **RS<0:1>**--Channel Select 0 and 1 determine which channel's Mode Register is written to. Read back for the mode register results in bits 0 and 1 both being ones.

RS1	RS0	CHANNEL
0	0	Channel 0 Select
0	1	Channel 1 Select
1	0	Channel 2 Select
1	1	Channel 3 Select

Format for the Request Register read operation is shown below.

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
1	1	1	1	RC3	RC2	RC1	RC0

(Read Operation)

- o **RC<0:3>**--During a Request Register read, the state of the request bit associated with each channel is returned in bits 0 through 3 of the byte. The bit position corresponds to the channel number.

2.4.10.7 Request Mask Register

The Request mask register is a set of four bits that are used to inhibit external DMA requests from generating transfer cycles. This register can be programmed in two ways. Each channel can be independently masked by writing to the Write Single Mask Bit location. The data format for this operation is shown below.

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
X	X	X	X	X	MB	MS1	MS0

(Set/Reset Operation)

- o **MB**--Bit 2 sets or resets the request mask bit for the channel selected by MS1 and MS0. Writing a one in this bit position sets the mask, inhibiting external requests.
- o **MS<0:1>**--These two bits select the specific mask bit that is to be set or reset.

MCS1	MS0	CHANNEL
0	0	Channel 0 Select
0	1	Channel 1 Select
1	0	Channel 2 Select
1	1	Channel 3 Select

Alternatively, all four mask bits can be programmed in one operation by writing to the Write All Mask Bits address. The data format for this function and the Read All Mask Bits function is shown below.

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
X	X	X	X	MB3	MB2	MB1	MB0

- o **MB<0:3>**--Each bit position in the field represents the mask bit of a channel. The mask bit number corresponds to the channel number associated with the mask bit. All four mask bits are set following a RESET or a Master Clear command. Individual channel mask bits are set as a result of terminal count being reached, if Auto-Initialize is disabled. The entire register can be cleared, enabling all four channels, by performing a Clear Mask Register operation.

2.4.10.8 Status Register

The status of all four channels can be determined by reading the Status Register. Information is available to determine if a channel has reached terminal count and whether an external service request is pending. Bit 0-3 of this register are cleared by a RESET, a Master Clear, or each time a Status Read takes place. Bits 4-7 are cleared by a RESET, a Master Clear, or the pending request being de-asserted. Bits 4-7 are not affected by the state of the Mask Register Bits. The Channel number corresponds to the bit position.

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
DRQ3	DRQ2	DRQ1	DRQ0	TC3	TC2	TC1	TC0

(Read Only Register)

2.4.10.9 Temporary Register

The Temporary Register is used as a temporary holding register for data during memory-to-memory transfers. The register is loaded during the first cycle of a memory-to-memory transfer from XD<0:7>. During the second cycle of the transfer, the data in the Temporary Register is output on the XD<0:7> pins. Data from the last memory-to-memory transfer remains in the register unless a RESET or a Master Clear occurs.

2.4.11 Special Commands

Five special commands are provided to make the task of programming the device easier. These commands are activated as a result of a specific address and assertion of either an IOR* or IOW*. Information on the data lines is ignored by the 82C235 whenever an IOW* activated command is issued. Thus data returned on IOR* activated commands is invalid. Descriptions of the five special commands follow:

o **Clear Byte Pointer Flip-Flop**--This command is normally executed prior to reading or writing to the address or word count register. This initializes the flop-flop to point to the low byte of the register and allows the CPU to read or write the register bytes in correct sequence.

o **Set Byte Pointer Flip-Flop**--Setting the Byte Pointer Flip-Flop allows the CPU to adjust the pointer to the high byte of an address or word count register.

o **Master Clear**--This command has the same effect as a hardware RESET. The Command Register, Status Register, Request Register, Temporary Register, Mode Register Counter, and Byte Pointer Flip-Flop are cleared and the Request Mask Register is set. Immediately following Master Clear or RESET, the DMA is in the Idle Condition.

o **Clear Request Mask Register**--This command enables all four DMA channels to accept requests by clearing the mask bits in the register.

o **Clear Mode Register Counter**--In order to allow access to four Mode Registers while only using one address, an additional counter is used. After clearing the counter, all four Mode Registers may be read by doing successive reads to the Read Mode Register address. The order in which the register is read is Channel 0 first and Channel 3 last.

2.5 Interrupt Controller

The 82C235 incorporates two programmable interrupt controllers that are compatible with the Intel 8259A. The controllers accept requests from peripherals, resolve priority on pending interrupts and interrupts in service, issue an interrupt request to the CPU, and provide a vector that is used as an index by the CPU to determine which interrupt service routine to execute.

A variety of priority assignment modes are provided, which can be reconfigured at any time during system operation, allowing the complete interrupt subsystem to be restructured, based on the system requirements.

The controllers are cascaded in a fashion compatible with the IBM PC AT (refer to Figure 1.4).

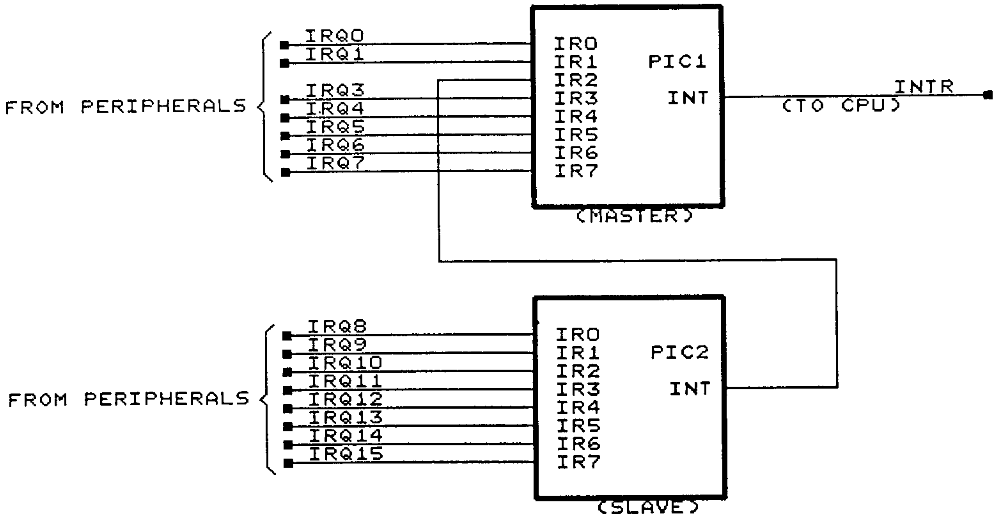


Figure 1.4 Cascaded Interrupt Controllers

Table 1.2 shows interrupt levels used by the system board and I/O channel. The interrupts are shown by priority, starting with the highest level.

Table 1.2 Interrupts on the I/O Channel

Interrupt	System Board	I/O Channel
NMI	Parity Check	IOCHCK
IRQ0	Timer	Not Available
IRQ1	Keyboard	Not Available
*IRQ8	Real-Time Clock	Not Available
IRQ9	Not Used	Available
IRQ10	Not Used	Available
IRQ11	Not Used	Available
IRQ12	Not Used	Available
IRQ13	Co-Processor	Not Available
IRQ14	Not Used	Available
IRQ15	Not Used	Available
IRQ3	Serial Port 2	Available
IRQ4	Serial Port 1	Available
IRQ5	Parallel Port 2	Available
IRQ6	Hard Disk and Diskette Drives	Available
IRQ7	Parallel Port 1	Available

* In internal RTC mode, IRQ8 is an internal signal.

The two devices are interconnected and must be programmed to operate in Cascade Mode (see Figure 1.4) for proper operation of all 16 interrupt channels. INTC1 is located at addresses 020H-021H and is configured for Master operation (defined below) in Cascade Mode. INTC2 is a Slave device (defined below) and is located at 0A0h-0A1H. The Interrupt Request output signal from INTC2 (INT) is internally connected to the interrupt request input Channel 2 IR2) of INTC1. The address decoding and Cascade interconnection matches that of the IBM PC AT.

Two additional interconnections are made to the interrupt request inputs of the interrupt controllers. The output of Time 0 in the Counter/Timer subsystem is connected to Channel 0 (IR0) of INTC1. An Interrupt request from the Real Time Clock is connected to Channel 0 (IR0) of INTC2. Figure 3.9 lists the 16 interrupt channels and their interrupt request sources.

The following description of the Interrupt Subsystem pertains to both INTC1 and INTC2 unless otherwise noted. Wherever register addresses are used, the address for the INTC1 register is listed first, and the address for the INTC2 register follows in parenthesis, e.g., 020H (0A0H).

2.5.1 Controller Operation

Figure 1.5 is a block diagram of the major elements in the interrupt controller. The Interrupt Request Register (IRR) is used to store requests from all the channels that are requesting service. Interrupt Request Register bits are labeled using the Channel Name IR<0:7> The In-Service register (ISR) contains all the channels that are currently being serviced (more than one channel can be in service at a time). In-Service Register bits are labeled IS<0:7>. The Interrupt Mask Register (IMR) allows the CPU to disable any or all of the interrupt channels. The Priority Resolver evaluates inputs from the above three registers, issues an interrupt request, and latches the corresponding bit into the In-Service Register. During interrupt acknowledge cycles, a master controller outputs a code to the slave device that is compared in the Cascade Buffer/Comparator with a three bit ID code previously written. If a match occurs in the slave controller, it generates an interrupt vector. The contents of the Vector Register are used to provide the CPU with an interrupt vector during Interrupt Acknowledge (INTA) cycles.

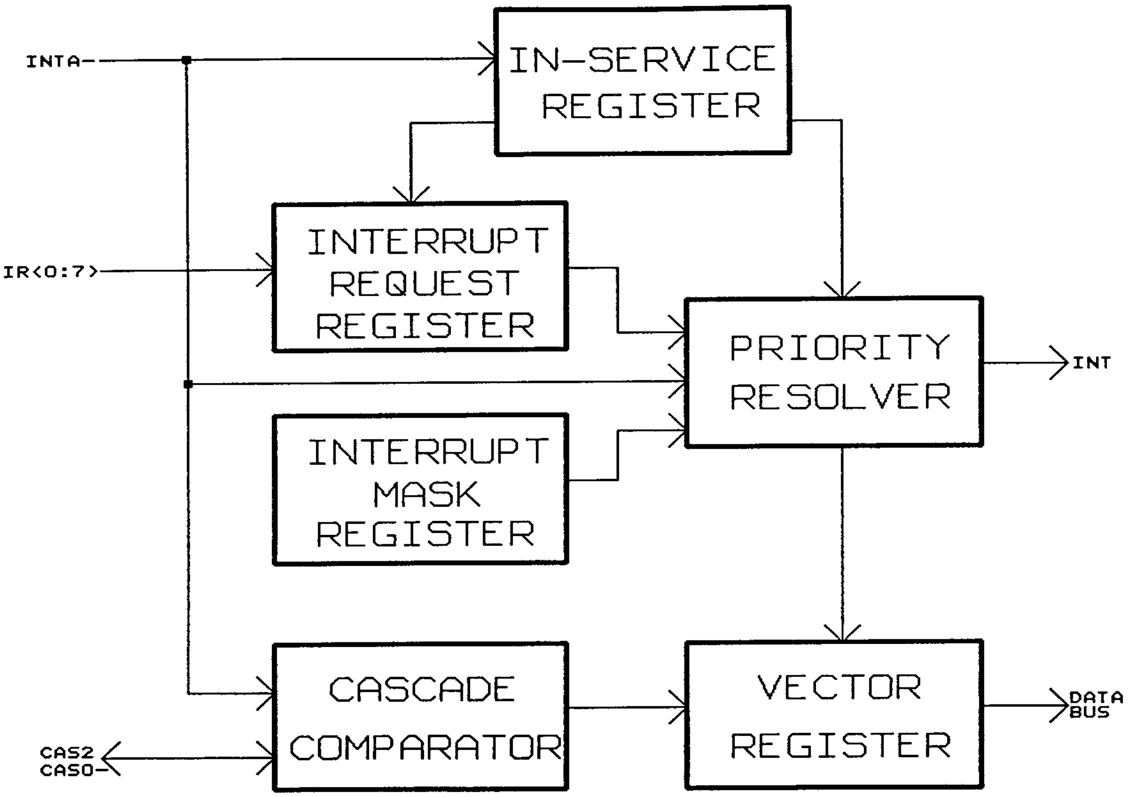


Figure 1.5 Interrupt Controller Block Diagram

2.5.2 Interrupt Sequence

The 82C235 allows the CPU to perform an indirect jump to a service routine in response to a request for service from a peripheral device (see Figure 1.6). The indirect jump is based on a vector that is provided by the 82C235 on the second of two CPU generated INTA cycles (the first INTA cycle is used for resolving priority; the second cycle is used for transferring the vector to the CPU). The events that occur during an interrupt sequence are as follows:

1 One or more of the interrupt requests (IR7-IR0) becomes active, setting the corresponding IRR bit(s).

2 The interrupt controller resolves priority based on the state of IRR, IMR, and ISR and asserts the INTR output if appropriate.

3 The CPU accepts the interrupt and responds with an INTA cycle.

4 During the first INTA cycle, the highest priority ISR bit is set and the corresponding IRR bit is reset. The internal Cascade address is generated and LD<0:7> outputs remain tri-stated.

5 The CPU executes a second INTA cycle, during which the 82C235 drives an 8-bit vector onto the data pins LD<0:7>, which is in turn latched by the CPU. The format of this vector is shown in

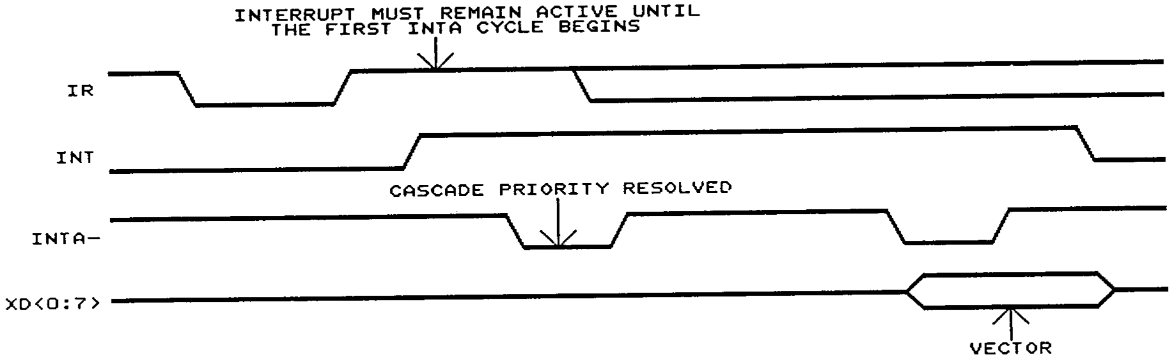

INTERRUPT MUST REMAIN ACTIVE UNTIL THE FIRST INTA CYCLE BEGINS
IR
INT
CASCADE PRIORITY RESOLVED
INTA–
XD<0:7>
VECTOR

Figure 1.6 Interrupt Sequence

the following table. Note that V<3:7> in the table are programmable by writing to Initialization Control Word 2 (see Initialization Command Words section below and Figure 1.7).

	D7	D6	D5	D4	D3	D2	D1	D0
IR7	V7	V6	V5	V4	V3	1	1	1
IR6	V7	V6	V5	V4	V3	1	1	0
IR5	V7	V6	V5	V4	V3	1	0	1
IR4	V7	V6	V5	V4	V3	1	0	0
IR3	V7	V6	V5	V4	V3	0	1	1
IR2	V7	V6	V5	V4	V3	0	1	0
IR1	V7	V6	V5	V4	V3	0	0	1
IR0	V7	V6	V5	V4	V3	0	0	0

6 At the end of the second INTA cycle, the ISR bit is cleared if the Automatic End-Of-Interrupt mode is selected (see End-Of-Interrupt section below). Otherwise, the ISR bit must be cleared by an End-of-Interrupt (EOI) command from the CPU at the end of the interrupt service routine.

If no interrupt request is present at the beginning of the first INTA cycle (i.e., a spurious interrupt), INTC1 issues an interrupt level 7 vector during the second INTA cycle.

2.5.3 End-of-Interrupt (EOI)

EOI is defined as the condition that causes an ISR bit to be reset. Determination of which ISR bit is to be reset can be done by a CPU command (specific EOI) or the Priority Resolver can be instructed to clear the highest priority IST bit (non-specific EOI).

The 82C235 can determine the correct ISR bit to reset when operated in modes that do not alter the fully nested structure, since the current highest priority ISR bit is necessarily the last level acknowledged and serviced. In conditions where the fully nested structure is not preserved, a specific EOI must be generated at the end of the interrupt service routine. An ISR bit that is masked, in Special Mask Mode by a IMR bit, is not cleared by a non-specific EOI command. Optionally, the interrupt controller can generate an Automatic End-of-Interrupt (AEOI) on the trailing edge of the second INTA cycle.

2.5.4 Priority Assignment

Assignment of priority is based on an interrupt channel's position relative to the other channels in the interrupt controller. After the initialization sequence, IR0 has the highest priority, IR7 has the lowest, and priority assignment is fixed (Fixed Priority Mode). Priority assignment can be rotated either manually (Specific Rotation Mode) or automatically (Automatic Rotation Mode) by programming Operational Command Word 2 (OCW2).

2.5.4.1 Fixed Priority Mode

This is the default condition that exists unless rotation (either manual or automatic) is enabled, or the controller is programmed for Polled Mode. In Fixed Priority Mode, interrupts are fully nested with priority assigned as shown:

Priority Status	Lowest							Highest
	7	6	5	4	3	2	1	0

Nesting allows interrupts with higher priorities to generate interrupt requests prior to the completion of the interrupt in service. When an interrupt is acknowledged, priority is resolved, the highest priority request's vector is placed on the bus, and the ISR bit for that channel is set. This bit remains set until an EOI (automatic or CPU generated) is issued to that channel. While the ISR bit is set, all interrupts of equal or lower priority are inhibited. Note that a higher priority interrupt that occurs during an interrupt service routine, is acknowledged only if the CPU has internally re-enabled interrupts.

2.5.4.2 Specific Rotation Mode

Specific Rotation allows the system software to re-assign priority levels by issuing a command that redefines the highest priority channel.

Before Rotation

Priority Status	Lowest							Highest
	7	6	5	4	3	2	1	0

(Specific Rotation command issued with Channel 5 specified)

After Rotation

Priority Status	Lowest							Highest
	5	4	3	2	1	0	7	6

2.5.4.3 Automatic Rotation Mode

In applications in which a number of equal priority peripherals are requesting interrupts, Automatic Rotation may be used to equalize the priority assignment. In this mode a peripheral, after being serviced, is assigned the lowest priority. All peripherals connected to the controller are serviced at least once in eight interrupt requests to the CPU from the controller. Automatic rotation occurs, if enabled, due to the occurrence of EOI (automatic or CPU generated).

Before Rotation (IR4 is highest priority request being serviced.)

ISR Status Bit	IS7	IS6	IS5	IS4	IS3	IS2	IS1	IS0
	0	1	0	1	0	0	0	0
Priority Status	Lowest							Highest
	7	6	5	4	3	2	1	0

After Rotation (IR4 service completed.)

ISR Status Bit	IS7	IS6	IS5	IS4	IS3	IS2	IS1	IS0
	0	1	0	0	0	0	0	0
Priority Status	Lowest							Highest
	4	3	2	1	0	7	6	5

2.5.5 Programming the Interrupt Controller

Two types of commands are used to control the 82C235 interrupt controllers: Initialization Command Words (ICWs) and Operational Command Words (OCWs).

2.5.5.1 Initialization Command Words

The initialization process consists of writing a sequence of four bytes to each interrupt controller. The initialization sequence is started by writing the first Initialization Command Word (ICW1) to address 020H (0A0H) with a one on bit 4 of the data byte. The interrupt controller interprets this as the start of the initialization sequence and does the following:

1 The Initialization Command Word Counter is reset to zero.

2 ICW1 is latched into the device.

3 Fixed Priority Mode is selected.

4 IR7 is assigned the highest priority.

5 The Interrupt Mask Register is cleared.

6 The Slave Mode Address is set to seven.

7 Special Mask Mode is disabled.

8 The IRR is selected for Status Read operations.

The next three I/Os write to address 021H (0A1H) will load ICW2-ICW4. See Figure 1.7 for a flow chart of the initialization sequence. The initialization sequence can be terminated at any point (all four bytes must be written for the controller to be properly initialized) by writing to address 020H (0A0H) with a zero in data bit 4. Note, this causes OCW2 or OCW3 to be written.

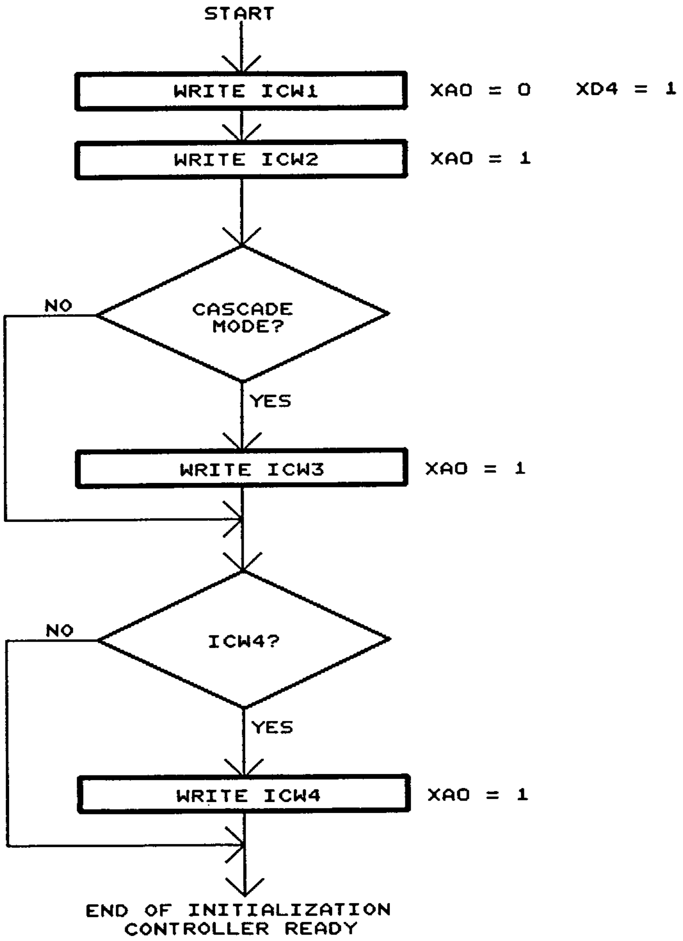


Figure 1.7 Initialization Sequence

ICW1 (Address 020H (0A0H))

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
X	X	X	SI	LTM	X	SM	X

(Write Only Register)

o SI--Bit 4 indicates to the interrupt controller that an Initialization Sequence is starting and must be a one to write ICW1.

o LTM--Bit 3 selects level or edge triggered inputs to the IRR. If a one is written to LTM, a high-level on the IRR input generates an interrupt request. The IR must be active until the first INTA cycle is started to generate the proper interrupt vector (an IR7 vector is generated if the IRR input is de-asserted early), and the IR must be removed prior to EOI to prevent a second interrupt from occurring.

o SM--Bit 1 selects between Single Mode and Cascade Mode. Single Mode is used whenever only one interrupt controller (INTC1) is used and is not recommended for this device. Cascade Mode allows the two interrupt controllers to be connected through IR2 of INTC1. INTC1 allows INTC2 to generate its own interrupt vectors if Cascade Mode is selected and the highest priority IR pending is from an INTC2 input. INTC1 and INTC2 must be programmed for Cascade Mode for both devices to operate.

ICW2 (Address 021H (0A1H))

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
V7	V6	V5	V4	V3	X	X	X

(Write only Register)

o V<3:7>--These bits are the upper five bits of the interrupt vector and are programmable by the CPU. The lower three bits of the vector are generated by the Priority Resolver during INTA (see Figure 1.7). INTC1 and INTC2 need not be programmed with the same value in ICW2.

ICW3 Format for INTC1 (Address 021H))

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
S7	S6	S5	S4	S3	S2	S1	S0

(Write Only Register)

o S<0:7>--Select which IR inputs have Slave Mode controllers connected. ICW3 in INTC1 must be written with 04H for INTC2 to function.

ICW3 Format for INTC2 (Address 0A1H))

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0	0	ID2	ID1	ID0

(Write Only Register)

- **ID2-ID0**--Determine the Slave Mode address the controllers will respond to during the cascaded INTA sequence. ICW3 in INTC2 should be written with 02H for cascade Mode operation. Note that b<3:7> should be zero.

ICW4 (Address 021H (0A1H))

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
X	X	X	EMI	X	X	AEOI	X

(Write Only Register)

- **EMI**--Bit 4 enables multiple interrupts from the same channel in fixed Priority Mode. This allows INTC2 to fully nest interrupts, when Cascade Mode with Fixed Priority Mode are both selected, without being blocked by INTC1. Correct handling of this mode requires the CPU to issue a non-specific EOI command to INTC2 and to check its In-Service Register for zero when exiting an interrupt service routine. If zero, a non-specific EOI command should be sent to INTC1. If non-zero, no command is issued.

- **AEOI**--Auto End-of-Interrupt is enabled when ICW4 is written with a zero in both. The interrupt controller performs a non-specific EOI on the trailing edge of the second INTA cycle. Note that this function should not be used in a device with fully nested interrupts unless the device is a cascade Master.

2.5.5.2 Operational Command Words

Operational Command Word One (OCW1) is located at address 021H (0A1H) and may be written any time the controller is in Initialization Mode. Operational Command Words Two and Three (OCW2 and OCW3) are located at address 020H (0A0H). Writing to address 020H (0A0H) with a zero in bit 4 places the controller in operational mode and loads OCW2 (if data bit 3 = 0) or OCW3 (if data bit 3 = 1).

OCW1 (Address 021H (0A1H))

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
M7	M6	M5	M4	M3	M2	M1	M0

(Read/Write Register)

- **M<0:7>**--These bits control the state of the Interrupt Mask Register. Each Interrupt Request can be masked by writing a one in the appropriate bit position (M0 controls IR0 etc.). Setting an IMR bit has no affect on lower priority requests. All IMR bits are cleared by writing ICW1.

OCW2 (Address 020H (0A0H))

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
R	SL	EOI	SI	0	L2	L1	L0

(Write Only Register)

- o **R**--This bit in conjunction with SL and EOI selects operational function. Writing a one in bit 7 causes one of the rotate functions to be selected.

R	SL	EOI	FUNCTION
1	0	0	Rotate on auto EOI enable*
1	0	1	Rotate on non-specific EOI
1	1	0	Specific Rotate Command
1	1	1	Rotate on specific EOI

* This function is disabled by writing a zero to all three bit positions.

- o **SL**--This bit in conjunction with R and EOI selects operational function. Writing a one in this bit position causes a specific or immediate function to occur. All specific commands require L2-L0 to be valid except no operation.

R	SL	EOI	FUNCTION
0	1	0	No Operation
0	1	1	Specific EOI Command
1	1	0	Specific Rotate Command
1	1	1	Rotate on specific EOI

- o **EOI**--This bit in conjunction with R and SL selects operational function. Writing a one in this bit position causes a function related to EOI to occur.

R	SL	EOI	FUNCTION
0	0	1	Non-specific EOI Command
0	1	1	Specific EOI Command
1	0	1	Rotate on non-specific EOI
1	1	1	Rotate on specific EOI

- o **SI**--Writing a zero in this bit position takes the interrupt controller out of initialize mode and writes OCW2 or OCW3.

- **L<0:2>**--These three bits are internally decoded to select which interrupt channel is to be affected by the Specific command. L<0:2> must be valid during three of the four specific cycles (see SL above).

OCW3 (Address 020H (0A0H))

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
0	ESMM	SMM	SI	1	PM	RR	RIS

(Write Only Register)

- **ESMM**--Writing a one in this bit position enables the Set/Reset Special Mask Mode function controlled by bit 5 (SMM). ESMM allows the other functions in OCW3 to be accessed and manipulated without affecting the Special Mask Mode state.

- **SMM**--If ESMM and SMM both are written with a one, the Special Mask Mode is enabled. Writing a one to ESMM and a zero to SMM disables Special Mask Mode. During Special Mask Mode, writing a one to any bit position inhibits interrupts and a zero enables interrupts on the associated channel by causing the Priority Resolver to ignore the condition for the ISR.

- **SI**--See SI above.

- **PM**--Polled Mode is enabled by writing a one to bit 2 of OCW3 causing the 82C235 to perform the equivalent of an INTA cycle during the next I/O read operation to the controller. The byte read during this cycle sets bit 7 if an interrupt is pending. If bit 7 of the byte is set, the level of the highest pending request is encoded on bits 2-0. The IRR remains frozen until the read cycle is completed, at which time the PM bit is reset.

- **RR**--When the RR bit (bit 1) is one, reading the Status Port at address 020H (0A0H) causes the contents of IRR or ISR (determined by RIS) to be placed on XD7-XD0. Asserting PM forces RR reset.

- **RIS**--This bit selects between the IRR and the ISR during Status Read operations if RR = 1. IRR is selected if this bit is set to one. ISR is selected if this bit is set to zero.

2.6 Programmable Interval Timer

The programmable interval timer, which is equivalent to the Intel 8254 Programmable Interval Timer/Counter, is programmable through external I/O ports 0040H through 0043H (refer to Figure 1.8).

The inputs of the three channels are connected to a 1.19MHz clock. The 1.19MHz clock is internally generated by dividing OSCX1 (14.31818MHz) by 12. The output of the three channels are as follows:

- o Channel 0 is a general purpose and software interrupt timer. The output of this channel is connected directly to the IRQ0 pin of the internal programmable interrupt controller.
- o The output of Channel 1 is used internally by the 82C235 to generate refresh requests.
- o The output of Channel 2 supports tone generation for the audio speaker.

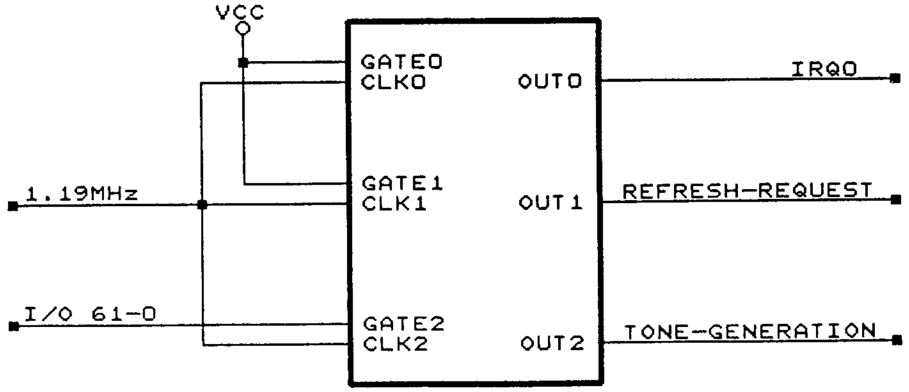


Figure 1.8 Programmable Timer Control

The Counter/Timer (CTC) in the 82C235 is general purpose, and can be used to generate accurate time delays under software control. The CTC contains three 16-bit counters (Counter 0-3) that can be programmed to count in binary or binary coded decimal (BCD). Each counter operates independently of the other two and can be programmed for operation as a timer or a counter.

All three of the counters are controlled from a common set of control logic. The control logic decodes control information written to the CTC and provides the controls necessary to load, read, configure, and control each counter. Counter 0 and Counter 1 can be programmed for all six modes, but Mode 1 and Mode 5 have limited usefulness due to the lack of an external hardware trigger signal. Counter 2 can be operated in any of six modes listed below.

o Mode 0 - Interrupt on terminal count

o Mode 1 - Hardware re-triggerable one-shot

o Mode 2 - Rate generator

o Mode 3 - Square wave generator

o Mode 4 - Software triggered strobe

o Mode 5 - Hardware re-triggerable strobe

All three counters in the CTC are driven from a common clock input pin (TMRCLK) that is derived by dividing OSCX1 (14.31818MHz) by 12. Counter zeros output (Out0) is connected to IRQ0 of INTC1 (see the Interrupt Controller Functional Description) and may be used as an interrupt to the system for time keeping and task switching. Counter 1 is programmed to generate pulses for use by the refresh generator. The third counter (Counter 2) is a full function Counter/Timer. This channel can be used as an interval timer, a counter, or as a gated rate/pulse generator (which is normally used as a speaker tone generator).

2.6.1 Counter Description

Each counter in the CTC contains a Control Register, a Status Register, a 16-bit Counting Element (CE), a pair of 8-bit Counter Input Latches (CIL and CIH), and a pair of 8-bit Counter Output Latches (COL and COH). Each counter also has a clock input for loading and decrementing the CE, a mode defined GATE input for controlling the counter (only GAT2 is controlled by I/O port 61, bit 0), and an OUT signal. The OUT signal's state and function are controlled by the Counter Mode and condition of the CE (see Mode Definitions).

The Control Register stores the mode and command information used to control the counter. The Control Register may be loaded by writing a byte, containing a pointer to the desired counter, to the Write Control Word address (043H). The remaining bits in the byte contain the mode, the type of command, and count format information.

The Status register allows the software to monitor counter condition and read back the contents of the Control Register.

The Counting Element is a loadable 16-bit synchronous down-counter. The CE is loaded or decremented on the falling edge of TMRCLK. The CE contains the maximum count when a zero is loaded; this is equivalent to 655536 in binary operation or 10000 in BCD. The CE does not stop when it reaches zero. In Modes 2 and 3, the CE is reloaded and in all other modes it wraps around to FFFFH in binary operation or 9999 in BCD.

The CE is indirectly loaded by writing one or two bytes (optional) to the Counter Input Latches, which are in turn loaded into the CE. This allows the CE to be loaded or reloaded in one TMRCLK cycle.

The CE is also read indirectly by reading the contents of the Counter Output Latches. COL and COH are transparent latches that can be read while transparent or latched (see Latch Counter Command).

2.6.2 Programming the CTC

After power-up, the condition of CTC Control Registers, counter registers, CE, and the output of all counters is undefined. Each counter must be programmed before it can be used.

Counters are programmed by writing a Control Word and then an initial count. The Control Register of a counter is written to by writing to the Control Word address (see the following table). The Control Word is a write-only location.

Address	Function
040H	Counter 0 Read/Write
041H	Counter 1 Read/Write
042H	Counter 2 Read/Write
043H	Counter Register Write Only

Control Word (043H)

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
F3	F2	F1	F0	M2	M1	M0	BCD

(Write Only Register)

- **F<0:3>**--Bits 4-7 determine the command to be performed as shown in Table 1.3.
- **M<0:2>**--Bits 1-3 determine the counter's mode during Read/Write Counter Commands (see Read/Write Counter Command) or select the counter during a Read-Back Command (see Read-Back command). Bits 1-3 become "don't care" during Latch Counter Commands.
- **BCD**--Bit 0 selects binary coded decimal counting format during Read/Write Counter Commands. When bit 0 is set to zero, the count is binary; when bit 0 is set to one, the count is BCD. Note that during Read-Back Command this bit must be zero.

2.6.3 Read/Write Counter Command

When writing to a counter, two conventions must be observed:

- Each counter's Control Word must be written before the initial count is written.
- Writing the initial count must follow the format specified in the Control Word (least significant byte only, most significant byte only, or least significant byte, then most significant byte).

Table 1.3 Command Table

F3	F2	F1	F0	Command
0	0	0	0	Latch Counter 0 (see Counter Latch Command)
0	0	0	1	Read/Write Counter 0 LSB Only
0	0	1	0	Read/Write Counter 0 MSB Only
0	0	1	1	Read/Write Counter 0 LSB then MSB
0	1	0	0	Latch Counter 1 (see Counter Latch Command)
0	1	0	1	Read/Write Counter 1 LSB Only
0	1	1	0	Read/Write Counter 1 MSB Only
0	1	1	1	Read/Write Counter 1 LSB then MSB
1	0	0	0	Latch Counter 2 (see Counter Latch Command)2
1	0	0	1	Read/Write Counter 2 LSB Only
1	0	1	0	Read/Write Counter 2 MSB Only
1	0	1	1	Read/Write Counter 2 LSB then MSB
1	1	X	X	Read-Back Command (see Counter Read-Back Command)

MSB = most significant byte
LSB = least significant byte

A new initial count can be written into the counter any time after programming without rewriting the Control Word, as long as the programmed format is observed.

During Read/Write Counter Commands M<0:2> are defined as follows:

M2	M1	M0	Function
0	0	0	Select Mode 0
0	0	1	Select Mode 1
X	1	0	Select Mode 2
X	1	1	Select Mode 3
1	1	0	Select Mode 4
1	1	1	Select Mode 5

2.6.4 Latch Counter Command

When a Latch Counter Command is issued, the counter's output latches (COL and COH) latch the current state of the CE. COL and COH remain latched until read by the CPU, or the counter is reprogrammed. The output latches then return to a "transparent" condition. In this condition, the latches are enabled and the contents of the CE may be read directly.

Latch Counter Commands may be issued to more than one counter before reading the first counter to which the command was issued. Also, multiple Latch Counter Commands issued to the same counter without reading the counter cause all but the first command to be ignored.

2.6.5 Read-Back Command

The Read-Back Command allows the user to check the count value, mode, and state of the OUT signal and Null Count Flag of the selected counter(s).

The format of the Read-Back Command is:

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
1	1	LC	LS	C2	C1	C0	0

- o **LC**--Writing a zero in bit 5 causes the selected counter(s) to latch the state of the CE in COL and COH.
- o **LS**--Writing a zero in bit 4 causes the selected counter(s) to latch the current condition of its Control Register, Null Count, and Output into the Status Register. The next read of the Counter results in the contents of the Status Register being read (see Status Read).
- o **C<0:2>**--Writing a one in bit 3 causes Counter 3 to latch one or both of the registers specified by LC and LS. The same is true for bits 2 and 1 except that they enable Counters 1 and 0 respectively.

Each counter's latches remain latched until either the latch is read or the counter is reprogrammed.

If LS = LC = 0, status is returned on the next read from the counter. The next one or two reads (depending on whether the counter is programmed to transfer one or two bytes) from the counter result in the count being returned.

Status Byte

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
OUT	NC	F1	F0	M2	M1	M0	BCD

- o **OUT**--Bit 7 contains the state of the OUT signal of the counter.
- o **NC**--Bit 6 contains the condition of the Null count Flag. This flag is used to indicate that the contents of the CE are valid. NC is set to a one during a write to the Control Register or the counter. NC is cleared to a zero whenever the counter is loaded from the counter input registers.
- o **F<0:1>**--Bits 4-5 contain the F0 and F1 Command bits, which were written to the Command Register of the counter during initialization. This information is useful when determining whether the high byte, the low byte, or both must be transferred during counter read/write operations.
- o **M<1:2>**--These bits reflect the mode of the counter and are interpreted in the same manner as in Write Command operations.
- o **BCD**--Bit 0 indicates the CE is operating in BCD format.

2.6.6 Counter Operation

Due to the previously stated restrictions in Counter 0 and Counter 1, Counter 2 is used as the example in describing counter operation, but the description of Mode 0, 2, 3, and 4 is relevant to all counters.

The following terms are defined for describing CTC operation.

- **TMRCLK pulse**--A clock equivalent to OSCX1 (14.31818MHz) divided by 12.
- **trigger**--The rising edge of the GATE2 input.
- **counter load**--The transfer of the 16-bit value in CIL and CIH to the CE.
- **initialized**--A Control Word written and the Counter Input Latches loaded.

Counter 2 operates in one of the following modes.

2.6.6.1 Mode 0 - Interrupt on Terminal Count

Writing the Control Word causes OUT2 to go low and remain low until the CE reaches zero, at which time it returns to high and remains high until a new count or Control Word is written. Counting is enabled when GATE2 = 1. Disabling the count has no effect on OUT2. The CE is loaded with the first TMRCLK pulse after the Control Word and initial count are loaded. When both CIL and CIH are written, the CE is loaded after CIH is written (see Write Operations). This TMRCLK pulse does not decrement the count (for an initial count of N, OUT2 does not go high until N+1 TMRCLK pulses after initialization). Writing a new initial count to the counter reloads the CE on the TMRCLK pulse and counting continues from the new count.

If an initial count is written with GATE2 = 0, it is still loaded on the next TMRCLK pulse but counting does not begin until GATE2 = 1. Therefore, Out2 goes high N TMRCLK pulses after GATE2 = 1.

2.6.6.2 Mode 1 - Hardware Re-Triggerable One-Shot

Writing the Control Word causes OUT2 to go high initially. Once initialized, the counter is armed and a trigger causes OUT2 to go low on the next TMRCLK pulse. OUT2 then remains low until the counter reaches zero. An initial count of N results in a one-shot pulse N TMRCLK cycles long.

Any subsequent triggers while OUT2 is low causes the CE to be reloaded, extending the length of the pulse. Writing a new count to CIL and CIH does not affect the current one-shot unless the counter is re-triggered.

2.6.6.3 Mode 2 - Rate Generator

Mode 2 functions as a divide-by-N counter, with OUT2 as the carry. Writing the Control Word during initialization sets OUT2 high.

When the initial count is decremented to one, OUT2 goes low on the next TMRCLK pulse. The following TMRCLK pulse returns OUT2 high, reloads the CE, and the process is repeated. In Mode 2, the counter continues counting (if GATE2 = 1) and generates an OUT2 pulse every N TMRCLK cycles. Note that a count of one is illegal in Mode 2.

GATE2 = 0 disables counting and forces OUT2 high immediately. A trigger reloads the CE on the TMRCLK pulse. Thus GATE2 can be used to synchronize the counter to external events.

Writing a new count while counting does not affect current operation unless a trigger is received. Otherwise, the new count is loaded at the end of the current counting cycle.

2.6.6.4 Mode 3 - Square Wave Generator

Mode 3 is similar to Mode 2 in every respect except for the duty cycle of OUT2. OUT2 is set high initially and remains high for the first half of the count. When the first half of the initial count expires, OUT2 goes low for the remainder of the count.

If the counter is loaded with an even count, the duty cycle of OUT2 is 50% (high = low = N/2). For odd count values, OUT2 is high one TMRCLK cycle longer than it is low. Therefore, high = (N+1)/2 and low = (N-1)/2.

2.6.6.5 Mode 4 - Software Triggered Strobe

Writing the Control Word causes OUT2 to go initially. Expiration of the initial count causes OUT2 to go low for one TMRCLK cycle. GATE2 = 0 disables counting but has no effect on OUT2. Also, a trigger does not reload the CE.

The counting sequence is started by writing the initial count. The CE is loaded on the TMRCLK pulse after initialization. The CE begins decrementing one TMRCLK pulse later. OUT2 goes low for one TMRCLK cycle, (N+1) cycles after the initial count is written. If a new initial count is written during a counting sequence, it is loaded into the CE on the next TMRCLK pulse and the sequence continues from the new count. This allows the sequence to be re-triggerable by software.

2.6.6.6 Mode 5 - Hardware Triggered Strobe

Writing the Control Word causes OUT2 to go high initially. Counting is started by trigger. The expiration of the initial count causes OUT2 to go low for one TMRCLK cycle. GATE 2 = 0 disables counting.

The CE is loaded during counting, the current counting sequence is not affected unless a trigger occurs. A trigger causes the counter to be reloaded from CIL and CIH, making the counter re-triggerable.

2.6.7 GATE2

In Modes 0, 2, 3, and 4 GATE2 is level sensitive and is sampled on the rising edge of TMRCLK. In Modes 1, 2, 3, and 5 the GATE2 input is rising-edge sensitive. This rising edge sets an internal flip-flop whose output is sampled on the next rising edge of TMRCLK. The flip-flop resets immediately after being sampled. Note that in Modes 2 and 3 the GATE2 input is both edge- and level-sensitive (see Table 1.4).

Table 1.4 Gate Pin Function

Mode	Condition		
	Low	Rising	High
0	Disables Counting	—	Enables Counting
1	—	a) Initiates Counting b) Resets Out Pin	—
2	a) Disables Counting b) Forces Out Pin High	Initates Counting	Enables Counting
3	a) Disables Counting b) Forces Out Pin High	Initiates Counting	Enables Counting
4	Disables Counting	—	Enables Counting
5	—	Initiates Counting	—

2.7 Keyboard Controller Interface

The 82C235 uses an external 8042 to handle keyboard operations (refer to Figure 1.9). The clock for the 8042 may be derived from BUSCLK or OSC and should have a frequency that is between 6 MHz and 10 MHz when used with standard keyboard controllers. The 8042 interfaces with the 82C235 through IRQ1 and a chip select line. The 8042 also provides two output signals: GATEA20 and CPURESET. These signals are brought into the 82C235 and combined internally with the FAST GATEA20 and FAST CPU RESET functions. The keyboard controller also supports the keylock and CGA strap functions.

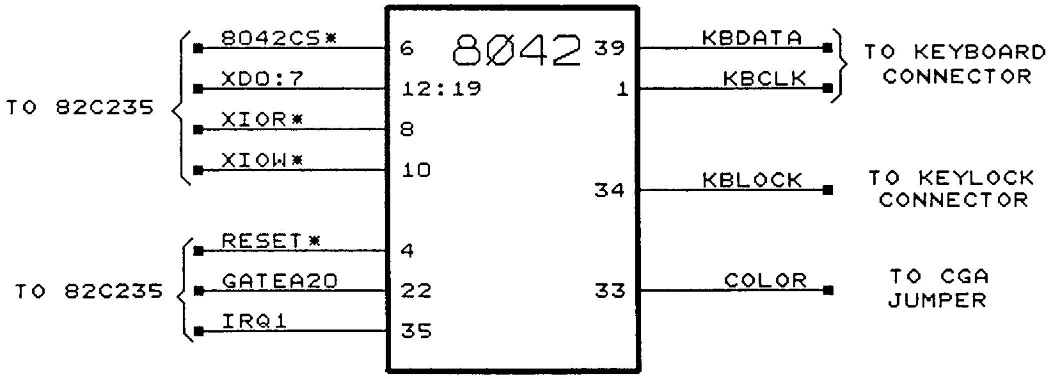


Figure 1.9 Keyboard Controller

2.8 Memory Interface

This section of the data manual discusses three types of memory interface: ROM/shadow RAM interface, DRAM interface, and extended/expanded memory interface.

2.8.1 ROM/Shadow RAM Interface

The 82C235 contains a programmable ROM chip select (ROMCS*) that becomes active when an access is made to the address range programmed by internal configuration register 48H or accesses between F00000H and FFFFFFH. The programmable range, which may be as large as 256KB (in 32KB blocks), is between 0C0000H and 0FFFFFH.

ROMs, which may be configured for either 8- or 16-bit mode, must be connected to the XD-bus. To select 16-bit mode, the DACK7 line must be pulled low with a 4.7K ohm resistor; to select 8-bit mode, the line must be pulled high with a 4.7K ohm resistor. The 82C235 asserts MEMCS16* during all ROM cycles if 16-bit mode is selected.

The 82C235 supports a feature called shadow RAM. This feature is invoked by copying an image of the BIOS (which is in ROM) into an area of RAM. The purpose of this feature is to allow operating systems and software applications to make faster accesses to the shadowed BIOS (rather than ROM which is much slower).

The degree of performance improvement derived from the use of shadow RAM depends primarily on the difference in access times between ROM and DRAM cycles. At higher system speeds, the difference can be significant. Additionally, shadow RAM allows use of a single 8-bit ROM (thus reducing circuit board size and component count) without sacrificing system performance.

Shadow RAM can be implemented using internal configuration registers 48H-4CH. Refer to Procedure 1.1 for the programming sequence that will enable shadow RAM. **NOTE**--The program that transfers the BIOS from ROM to RAM should reside in low RAM.

Procedure 1.1 Shadow RAM Programming Sequence

1. Disable the interrupts.
2. Copy the ROM BIOS into low ($\leq$ 640KB) RAM.
3. Disable ROMCS* using internal configuration register 48H.
4. Enable the shadow RAM using internal configuration registers 4AH-4CH.
5. Copy the BIOS from low DRAM into the area of memory reserved for shadow RAM.
6. Make the shadow RAM read only using by internal configuration register 49H.
7. Re-enable the interrupts.

2.8.2 DRAM Interface

The 82C235 includes a DRAM controller that supports directly up to 8MB of memory in four 18-bit (including two bits of parity) banks. By using an external '538-type decoder, up to eight 18-bit banks can be supported for a total of 16MB of memory. The 82C235 supports different memory configurations, which can be programmed using internal configuration register 4DH. To access any one of the four banks, the 82C235 asserts one of the four RAS signals (RAS0* through RAS3*) that corresponds to a particular bank, plus CASL* or CASH* to specify the low or high byte of that bank. For write operations, the MWE* signal is asserted. The 82C235 multiplexes the A01-20 address lines for DRAM accesses during assertion of RAS* and CAS* according to Table 1.4.

Table 1.5 Multiplexing the LA20-LA1 Address Lines

	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
RAS*	A19	A17	A8	A7	A6	A5	A4	A3	A2	A1
CAS*	A20	A18	A16	A15	A14	A13	A12	A11	A10	A9

NOTES-- A22-A21 are used for RAS0*-RAS3* generation for 2MB banks.
A20-A19 are used for RAS0*-RAS3* generation for 512KB banks.
A0 and BHE* are used for CASL* and CASH* generation.

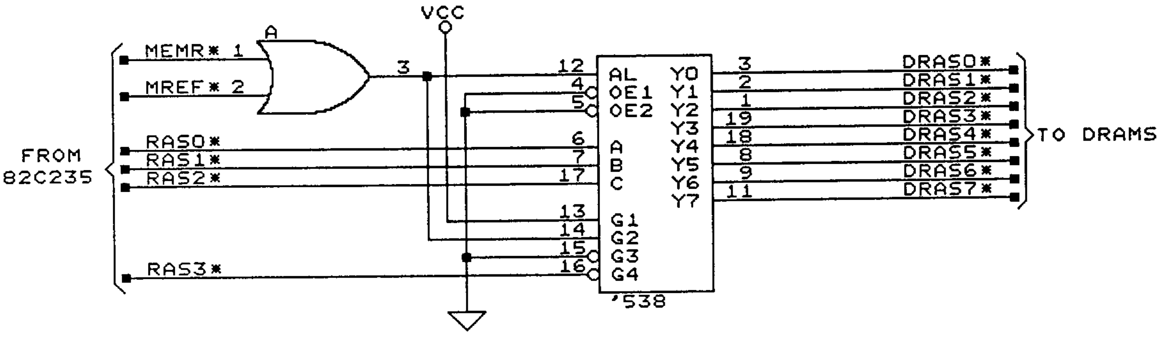
The upper addresses (A23-A19) are used to generate the RAS* signal for each bank. The A00 address line, in conjunction with BHE* signal, is used for CAS* generation for high byte or low byte or word. The addresses are sourced by the CPU or DMA controller inside the 82C235 or by EMS page registers inside the 82C235, or by a master on the I/O channel.

The 82C235 supports odd parity for each byte. Each time a byte or a word is written, an odd parity bit is generated and written along with the byte. When a read occurs, the written parity bit is compared against calculated parity from the read byte. If a mismatch occurs during this read operation, a parity error is reported and an NMI is generated indicating a problem with memory. The NMI generation for parity errors can be disabled using bit 6 of internal configuration register 46H or bit 2 of I/O port 61H. If the system designer decides to not implement the parity bit (because of cost or other reasons), NMI generation due to parity errors should be disabled as described.

Memory is accessed on the MD-bus (local bus) except when the address exceeds the installed memory size or the address references a range disabled by the RAM DISABLE bit using bit 5 of internal configuration register 4EH. When either of these conditions exist, the bus cycle is performed on the I/O channel. Additional memory can be configured on the I/O channel.

Local DRAM timing can be programmed using bits 6-4 of internal configuration register 4BH. Local DRAM timing can be run at zero wait-states, one wait-state, or RAS* delayed by PROCCLK/2 and one wait-state. The 82C235 also allows users to disable memory in the range 040000H-09FFFFH by using bit 5 of internal configuration register 4EH.

The 82C235 allows the system designer to utilize up to eight banks of memory using either 256K or 1M DRAM. This flexibility permits the installation of up to 16MB of total system memory. In order to support eight banks, an external 3-8 (F538) type decoder must be used. This mode is selected by programming an appropriate configuration using bits 3-0 of internal configuration register 4BH and enabling encoded RAS* mode by bit 7 of internal configuration register 4EH. This implementation is shown in Figure 1.10. During encoded RAS* mode, the multiplexed addresses, CAS* signals, and WE* are maintained in the same manner as they would be in normal (8MB) mode.



ENCODED RAS MODE IS ENABLED BY
1. ENABLING ICR4EH-7 AND
2. SELECTING ONE OF THE RAS ENCODE MODE USING ICR48H-3:0

Figure 1.10 RAS* Generation in Encoded RAS* Mode

2.8.3 Expanded/Extended Memory

The 82C235 supports the LIM EMS 4.0 specification. This specification allows operating systems and software applications to access memory above the DOS (1MB) limitation through implementation of a page mapping scheme that is managed by an EMS driver. Using I/O page registers, the 82C235 is able to re-map memory using 32 16KB pages.

The internal configuration register 4FH, bit 0, selects an I/O base for the I/O page registers to use for EMS implementation. To access the I/O page registers, enable internal configuration register 4FH, bit 6; to enable EMS memory mapping, enable internal configuration register 4FH, bit 7. Table 1.6 describes how I/O page registers are defined.

The 82C235 has 32 16KB EMS page registers. The first 24 EMS page frames are physically mapped into the area from 256KB to 640KB (040000H-09FFFFH). The remaining eight pages are mapped into the range 0D0000H-0EFFFFH. These are the physical addresses at which the EMS page frames appear. Logically mapped addresses are those addresses in extended memory that are pointed to by the I/O page registers. Figure 1.11, which shows how EMS is implemented, indicates that the physical addresses (A23-14) for each page frame are changed to the logical addresses specified by the I/O page register pairs. The converted addresses are then able to point to any location in RAM.

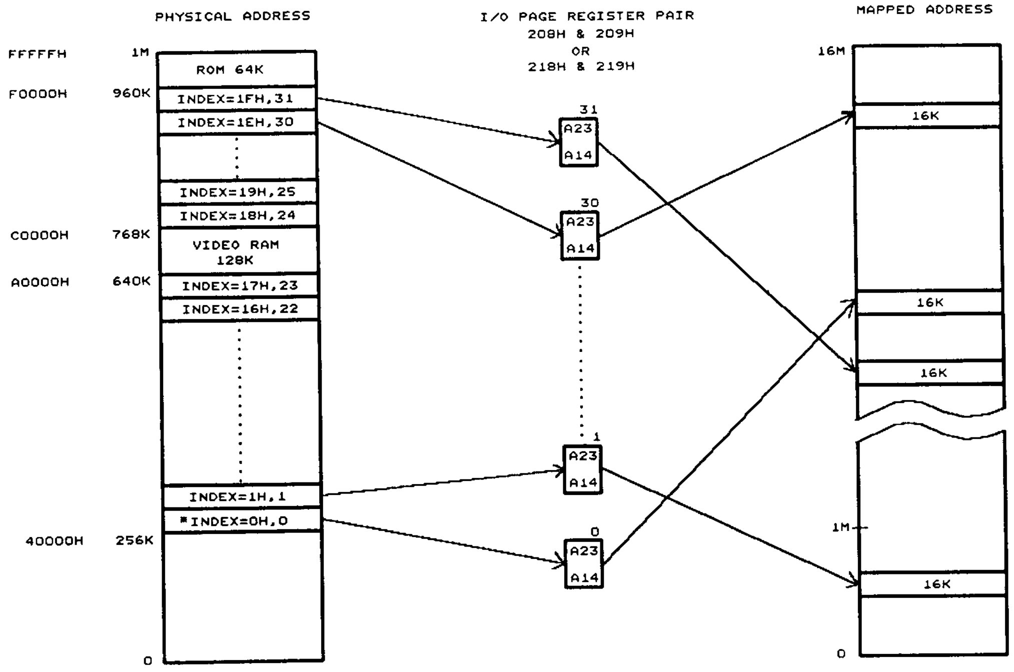
The EMS paging and I/O page registers are normally managed by application software or an EMS driver that allows applications to access more RAM than that allowed by DOS. More information on software implementation can be obtained from various Intel and Microsoft publications.

The 82C235 also supports a combination of expanded and extended memory (refer to Figure 1.12). Extended memory is memory at physical addresses above 1MB. Internal configuration register 4EH, bits

Table 1.6 I/O Page Registers Defined

EMS Page Register	Bit	Name	Description
I/O Base + 0 (Read/Write)	7-0	A21-A14	Addresses to be used instead of CPU addresses. * See note below.
I/O Base + 1 (Read/Write)	7	Page Enable	1=page addressed as EMS; 0=page not addressed as EMS (default).
	6-2	Reserved	
	0-1	A23-A22	Addresses to be used instead of CPU addreses. * See note below.
I/O Base + 2 (Read/Write)	7	AI Enable	This bit enables automatic incrementing of the EMS page index value upon an I/O Base + 1 register access. Thus, all 32 I/O page register pairs (word) can be written without having to rewrite index values. 1=automatic incrementing enable; 0=automatic incrementing disable (default).
	6-5	Reserved	
	4-0	Index 4-0	These bits determine which of the 32 EMS page register sets is accessed at the I/O Base and I/O Base + 1 locations.

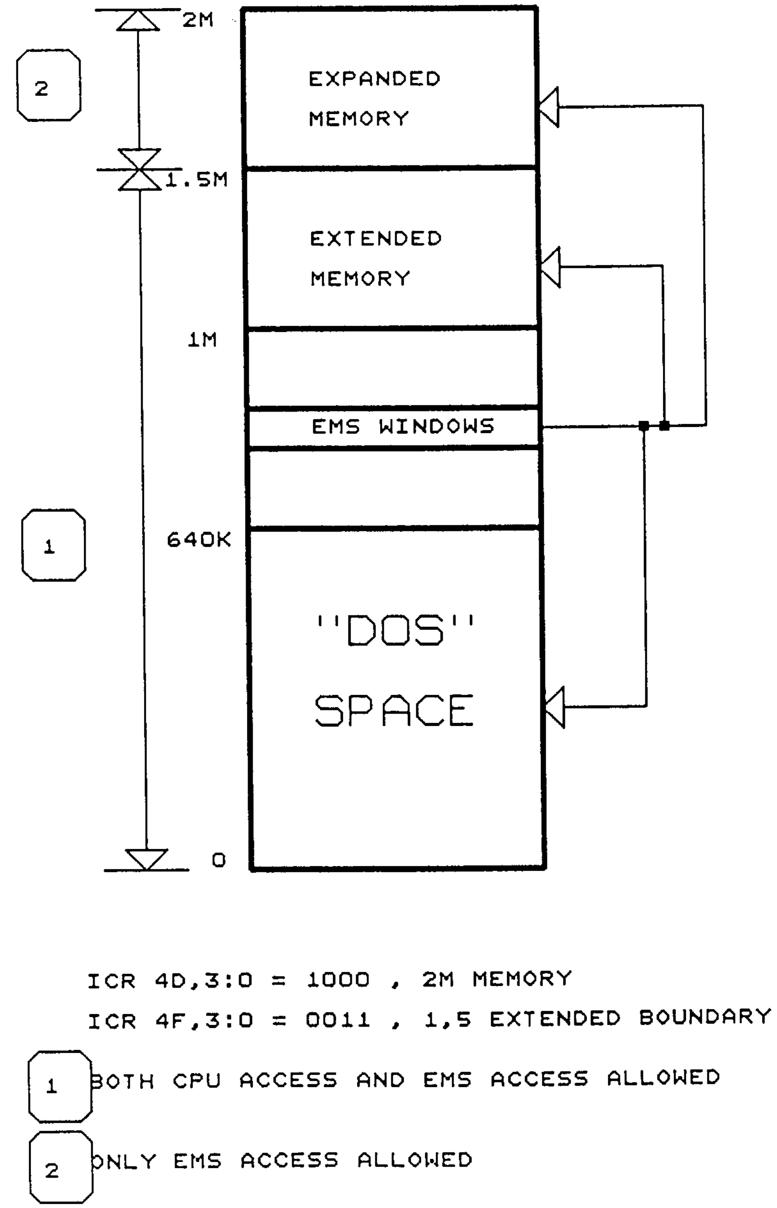
NOTE--The values written to registers I/O Base + 0 and I/O Base + 1 become the absolute values on the memory address lines to which they correspond. Therefore, the EMS address may point anywhere to the available system memory. No hardware checking is performed for conflicts with low (DOS) or other memory areas. It is the responsibility of the EMS driver to select appropriate values for the page register contents.



* INDEX IS PROGRAMMED USING I/O PORT 20AH OR 21AH

Figure 1.11 Expanded Memory Subsystem

3-0, define the extended boundary. Starting from 1MB up to the EMS boundary is the extended memory space. This area can be directly addressed by the CPU and can be used as extended memory. The extended memory is typically used for operating systems and software applications that do not limit their operations to 1MB of memory.



ICR 4D,3:0 = 1000 , 2M MEMORY

ICR 4F,3:0 = 0011 , 1,5 EXTENDED BOUNDARY

1 BOTH CPU ACCESS AND EMS ACCESS ALLOWED

2 ONLY EMS ACCESS ALLOWED

Figure 1.12 EMS/Extended Memory

2.9 Numeric Co-processor Interface

The 82C235 contains interface logic that supports use of an 80287 numeric coprocessor. The logic provides the following features:

- Decodes chip select to the 80287 (NPCS*)
- Generates a dedicated reset (NPRST)
- Detects and latches coprocessor error status
- Generates an interrupt (IRQ13) to the system when an error occurs

The clock for the 80287 can be derived externally from the BUSCLK or PROCCLK outputs.

2.10 RTC Interface

The 82C235 contains an internal, MC146818-compatible, real-time clock (RTC) with a total of 128 bytes of CMOS RAM. The internal RTC may be disabled so that an external RTC can be used. Internal/external RTC mode is selected by the state of the EXRTC* signal (DACK5*, pin 91) during power-up. If this pin is pulled low with a 4.7K ohm resistor, external RTC mode is selected. A 4.7K ohm pull-up resistor puts the 82C235 into internal RTC mode.

The definition of two 82C235 signal pins change when selecting RTC mode. The following table lists the differences between internal and external mode.

Table 1.7 Internal/External RTC Mode

Item	Internal RTC Mode	External RTC Mode
EXRTC* (DACK5*)	4.7K ohm pull-up resistor.	4.7K ohm pull-down resistor.
82C235 V_{cc} pins	Connect to the battery circuit.	Connect to the power supply.
MFP4 (pin 140)	32KHz input from an external oscillator.	IRQ08 is input from the RTC.
MFP5 (pin 141)	Power Sense input.	RTCCS* output to the RTC.

The internal RTC combines a complete time-of-day clock with alarm, one hundred year calendar, a programmable period interrupt, and 114 bytes of low-power static RAM. Provisions are made to enable the device to operate in a low-power (battery powered) mode and protect the contents of both the RAM and clock during system power-down.

2.10.1 Register Access

Reading and writing to the 128 locations in the Real Time Clock is accomplished by first placing the Index Address of the location you wish to access by writing the index to I/O port 70H. The address is then latched into the Index Address Register. The Index Address Register is then used as a pointer to the specific byte in the Real Time Clock, which may be read or written using I/O port 71H.

2.10.2 Address Map

Table 1.8 shows the internal register/RAM organization of the Real Time Clock portion of the 82C235. The 128 addressable locations in the Real Time Clock are divided into ten bytes that normally contain the time, calendar and alarm data, four control and status bytes, and 114 general purpose RAM bytes. All 128 bytes are readable by the CPU. The CPU may also write to all locations except Registers C, D, Bit 7 of Register A, and Bit 7 of the Seconds Byte which is always zero.

Table 1.8 Address Map for the Real Time Clock

Index	Function
00	SECONDS
01	SECONDS ALARM
02	MINUTES
03	MINUTES ALARM
04	HOURS
05	HOURS ALARM
06	DAY OF WEEK
07	DATE OF MONTH
08	MONTH
09	YEAR
0A	REGISTER A
0B	REGISTER B
0C	REFISTER C
0D	REGISTER D
0E	USER RAM
0F	USER RAM
.	
.	
.	
7E	USER RAM
7F	USER RAM

2.10.3 Time Calendar and Alarm Bytes

The CPU can obtain the time and calendar information by reading the appropriate locations in the Real Time Clock. Initialization of the time, calendar, and alarm information is accomplished by writing to these locations. Information is stored in these locations in binary-coded decimal (BCD) format.

Before initialization of the internal registers can be performed, the SET bit in Register B should be set to a one to prevent Real Time Clock updates from occurring. The CPU then initializes the first ten locations in BCD format. The SET bit should then be cleared to allow updates. Once initialized and enabled, the Real Time Clock performs clock/calendar updates at a 1Hz rate.

Table 1.9 shows the format for the ten clock, calendar, and alarm locations. The 24/12 bit in Register B determines whether the hour locations are updated using a 1-12 or 0-23 format. In 12 hour format, the high order bit of the hours byte in both the time and alarm bytes indicates PM when it is a one.

During updates, which occur once per second, the ten bytes of time, calendar, and alarm information are unavailable to be read or written by the CPU for a period of 2ms. These ten locations cannot be written to during this time. Information read while the Real Time CLock is performing update is undefined. The Update Cycle section describes how Update Cycle/PCU contention problems can be avoided.

Table 1.9 Time, Calendar, and Alarm Data Format

Index Register Address	Function	BCD Range
0	Seconds	00-59
1	Seconds Alarm	00-59
2	Minutes	00-59
3	Minutes Alarm	00-59
4	Hours (12 hour mode)	01-12 (AM) 81-92 (PM)
	Hours (24 hour mode)	00-23
5	Hours Alarm (12 hour mode)	01-12 (AM) 81-92 (PM)
	Hours Alarm (24 hour mode)	00-23
6	Day of Week	01-07
7	Day of Month	01-31
8	Month	01-12
9	Year	00-99

The alarm bytes can be programmed to generate an interrupt at a specific time or they can be programmed to generate a periodic interrupt.

2.10.4 Static RAM

The 114 bytes of RAM from Index Address OEH to 7FH are not affected by the Real Time Clock. These bytes are accessible during the update cycle and may be used for whatever the designer wishes. Typical applications use this as non-volatile storage for configuration and calibration parameters since this device is normally battery powered when the system is turned off.

2.10.5 Control and Status Registers

The 82C235 contains four registers used to control the operation and monitor the status of the Real Time Clock. These registers are located at Index Address 0AH-0DH and are accessible by the CPU at all times.

Register A (0AH)

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
UIP	DV2	DV1	DV0	RS3	RS2	RS1	RS0

(Read/Write register except UIP)

- o **UIP**--Update in progress flag is a status bit used to indicate when an update cycle is about to take place. A one indicates that an update cycle is taking place or is imminent. UIP goes active (High) 244us prior to the start of an update cycle and remains active for an additional 2ms while the update is taking place. The UIP bit is read only and is not affected by Reset. Writing a one to the SET bit in Register B inhibits any update cycle and then clears the UIP status bit.

- o **DV<0:2>**--These three bits are used to control the Divider/Prescaler on the Real-Time Clock. While the 82C235 RTC can operate at frequencies higher than 32.768 KHz, this is not recommended for battery powered operation due to the increased power consumption at these higher frequencies.

DV2	DV1	DV0	OSCI Freq	Mode
0	0	0	4.194304 MHz	Operate
0	0	1	1.048576 MHz	Operate
0	1	0	32.768 KHz	Operate
1	1	X	Reset Divider	

- o **RS<0:3>**--These four bits control the Periodic Interrupt rate. The Periodic Interrupt is derived from the Divider/Prescaler in the Real-Time Clock and is separate from the Alarm Interrupt. Both the alarm and periodic interrupts do use the same interrupt channel in the Interrupt Controller. Use of the Periodic Interrupt allows the generation of interrupts at rates higher than once per second. Table 1.10 shows the interrupt rates for which the Real-Time Clock can be programmed.

Register B (0BH)

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
SET	PIE	AIE	UIE	0	0	24/12	DSE

(Read Only Register)

- o **SET**--Writing a zero to this bit enables the Update Cycle and allows the Real Time Clock to function normally. When set to a one, the Update Cycle is inhibited and any cycle in progress is aborted. The SET bit is not affected by the RESET input pin.

- o **PIE**--The Periodic Interrupt Enable Bit controls the generation of interrupts based on the value programmed into the RS3-RS0 bits of Register A. This allows the user to disable this function without affecting the programmed rate. Writing a one to this bit enables the generation of periodic interrupts. This bit is cleared to a zero by Reset.

Table 1.10 Periodic Interrupt Rate

Rate Selection				Time Base	
RS3	RS2	RS1	RS0	4.194304 MHz 1.048576 MHz	32.768 KHz
0	0	0	0	None	None
0	0	0	1	30.517 μs	3.90526 ms
0	0	1	0	61.035 μs	7.8125 ms
0	0	1	1	122.070 μs	122.070 μs
0	1	0	0	244.141 μs	244.141 μs
0	1	0	1	488.281 μs	488.281 μs
0	1	1	0	976.562 μs	976.562 μs
0	1	1	1	1.953125 ms	1.953125 ms
1	0	0	0	3.90625 ms	3.90625 ms
1	0	0	1	7.8125 ms	7.8125 ms
1	0	1	0	15.625 ms	15.625 ms
1	0	1	1	31.25 ms	31.25 ms
1	1	0	0	62.5 ms	62.5 ms
1	1	0	1	125 ms	125 ms
1	1	1	0	250 ms	250 ms
1	1	1	1	500 ms	500 ms

o **AIE**--The generation of alarm interrupts is enabled by setting this bit to a one. Once this bit is enabled the Real Time Clock generates an alarm whenever a match occurs between the programmed alarm and clock information. If the "don't care" condition is programmed into one or more of the Alarm Registers, this enables the generation of periodic interrupts at rates of one second or greater. This bit is cleared by Reset.

o **UIE**--Update-ended interrupt enable bit enables the UF (update end flag) bit in Register C to assert interrupt request. This bit is cleared upon Reset.

o **24/12**--24/12 control bit is used to establish the format of both the Hours and Hours Alarm bytes. If this bit is a one, the Real Time Clock interprets and updates the information in these two bytes using the 24-hour mode. This bit can be read or written by the CPU and is not affected by Reset.

o **DSE**--The Real Time Clock can be instructed to handle daylight savings time changes by setting this bit to a one. This enables two exceptions to the normal time-keeping sequence to occur. Setting this bit to a zero disables the execution of these two exceptions.

Register C (0CH)

MSB LSB

b7	b6	b5	b4	b3	b2	b1	b0
IRQF	PF	AF	UF	0	0	0	0

(Read Only Register)

o **IRQF**–The Interrupt Request Flag is set to a one when any of the conditions that can cause an interrupt is true and the interrupt enable for that condition is true. The condition that causes this bit to be set also generates an interrupt. The logic expression for this flag is:

IRQF = PF & PIE

+ AF & AIE

+ UF & UIE

This bit and all other active bits in this register are cleared by reading this register or by activating the PS* input pin. Writing to this register has no affect on the contents.

o **PF**–The Period Interrupt Flag is set to a one when a transition, which is selected by RS3-RS0, occurs in the divider chain. This bit becomes active, independent of the condition of the PIE control bit. The PF bit then generates an interrupt and sets IRQF if PIE is a one.

o **AF**–A one appears in the AF bit when ever a match has occurred between the time register and alarm register during an update cycle. This flag is also independent of its enable (AIE) and generates an interrupt if AIE is true.

o **UF**–The update-ended flag bit is set after each cycle. When the UIE bit is a one, the one in UF causes the IRQF bit to be a one, asserting IRQ. UF is cleared by a Register C read or by a Reset.

REGISTER D (0DH)

MSB LSB

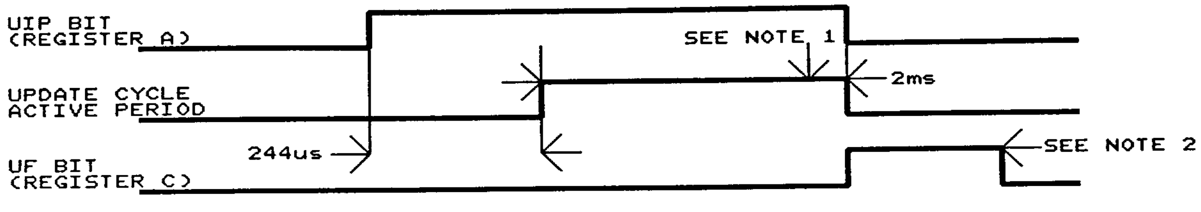
b7	b6	b5	b4	b3	b2	b1	b0
VRT	0	0	0	0	0	0	0

(Read Only Register)

o **VRT**–The Valid RAM and Time Bit indicates the condition of the contents of the Real Time Clock. This bit is cleared to a zero whenever the PS input pin is low. This pin is normally derived from the power supply, which supplies Vcc to the device and allows the user to determine whether the registers have been initialized since power was applied to the device. PS* has no effect on this bit and it can only be set by reading Register D. All unused register bits will be zero when read and are not writable.

2.10.6 Update Cycle

During normal operation, the Real Time Clock performs an update cycle once every second. The performance of an update cycle is contingent upon the divider bits DV<0:2> not being cleared, and the



NOTE:
1. REGISTER 0-9 ARE UNAVAILABLE TO BE READ OR WRITTEN DURING THIS TIME
2. UF BIT CLEARED BY CPU READ OF REGISTER C.

Figure 1.13 Update Cycle

SET bit in Register B being cleared. The function of the update cycle is to increment the clock/calendar registers and compare them to the Alarm Registers. If a match occurs between the two sets of registers, an alarm is issued and an interrupt is issued if the alarm and interrupt control bits are enabled.

During the time that an update is taking place, the lower ten registers are unavailable to the CPU. This is done to prevent the possible corruption of data in the registers or the reading of incorrect data. To avoid contention problems between the Real Time Clock and the CPU, a flag is provided in Register A to alert the user of an impending update cycle. This Update In Process Bit (UIP) is asserted 244us before the actual start of the cycle and is maintained until the cycle is complete. Once the cycle is complete, the UIP bit is cleared and the Update Flag (UF) in Register C is set. Figure 1.13 illustrates the update cycle. CPU access is always allowed to Register A through D during update cycles.

Two methods for reading and writing to the Real Time Clock are recommended. Both of these methods allow the user to avoid contention between the CPU and the Real Time Clock for access to the time and date information.

The first method is to read Register A, determine the state of the UIP bit. If the UIP is zero, perform the read or write operation. For this method to work successfully, the entire read or write operation (including any interrupt service routines which might occur) must not require more that 244us to complete from the beginning of the read of Register A to the completion of the last read or write operation to the Clock Calendar Registers.

The second method of accessing the lower ten registers is to read Register C once and disregard the contents. Continue reading this register until the UF bit is a one. This bit becomes true immediately after an update is completed. The user then has until the start of the next update cycle to complete a read or write operation.

2.10.7 Power-Up/Down

Most applications require the Real Time Clock to remain active whenever the system power is turned off. To accomplish this the user must provide an alternate source of power to the 82C235. This alternate source of power is normally provided by connecting a battery to the Vcc supply pin of the device. A means should be provided for switching between the system power supply and the battery. A circuit such as the one shown in Figure 1.14 may be used to eliminate power drain on the battery when the entire 82C235 is active. The circuit shown here allows for reliable transitions between system and battery power without undue battery power drain.

The user should also ensure that the Vin maximum specification is never exceeded when powering the system up or down. Failure to observe this specification may result in damage to the device.

A pin is provided on the device to protect the contents of the Real Time Clock and reduce power consumption whenever the system is powered down. This pin (PWRGOOD) should be low whenever the system power supply is not within specifications for proper operation of the system. This signal may be generated by circuitry in either the power supply or on the system board. The PWRGOOD input disables all unnecessary inputs during the time the system is powered down to prevent noise on the inactive pins from causing increased Icc. This pin must therefore be inactive for the remainder of the device to operate properly when system power is applied.

Another pin is provided to initialize the device whenever power is applied to the 82C235. This pin (PS*) does not alter the RAM or Clock/Calendar contents but it does initialize the necessary control register bits. Assertion of PS* disables the generation of interrupts and sets a flag indicting that the contents of the device may not be valid. A recommended circuit for controlling the PS* input is also shown in Figure 1.14.

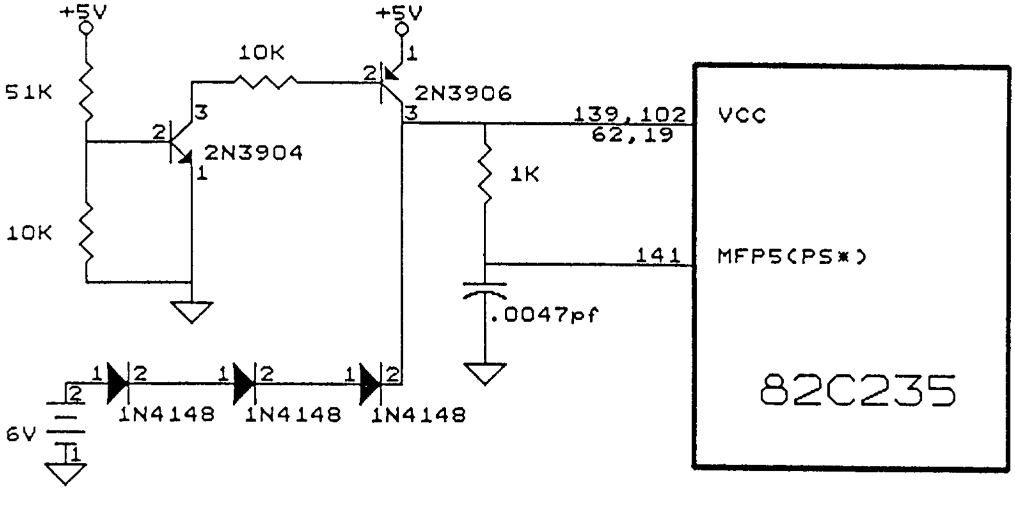


Figure 1.14 Power Conversion and Reset Circuitry

2.11 I/O Channel Interface

The 82C235 supports an IBM PC AT-compatible I/O channel. It incorporates a state machine that produces IO channel bus cycles for all CPU cycles not claimed by internal memory address decode logic.

The state machine synthesizes the address strobe signal (ALE), the bus command signals (XIOR*, XIOW*, XMEMR*, and XMEMW*), and MODA0 and MODA20. It monitors the state of the IOCS16* and MEMCS16* signals to determine if the device on the bus is capable of 16-bit operations for I/O and memory respectively. If a 16-bit operation is attempted with an 8-bit device, the 82C235 performs conversions of 16-bit CPU operations to paired 8-bit bus cycles.

The 82C235 supports XD-bus peripherals. Internal configuration resistor 44H controls the SDIRH and SDIRL signals for different peripherals.

Basic I/O channel cycle operation occurs as follows:

1 During processor state T_s, internal decodes are examined to determine if a bus cycle should be performed.

2 During the following T_s state, the 82C235 bus controller initiates the bus cycle by generating ALE. The bus controller then times out the COMMAND DELAY interval and asserts the appropriate command output.

3 In response to the asserted command, the internal wait-state logic times out for the number of processor BUSCLK's before beginning to monitor the IOCHRDY input. When the IOCHRDY input sampling is active, the bus controller deactivates the command. If the cycle in progress does not require conversion cycles, the 82C235 generates processor READY, and the CPU cycle terminates.

4 If the cycle in progress is a conversion cycle, the conversion logic sets MODA0 to logic one and begins to time out the ALE DELAY interval. At the end of this interval, the bus controller is re-triggered to produce another cycle as described in (2) and (3). Note that another ALE is not generated for the second cycle.

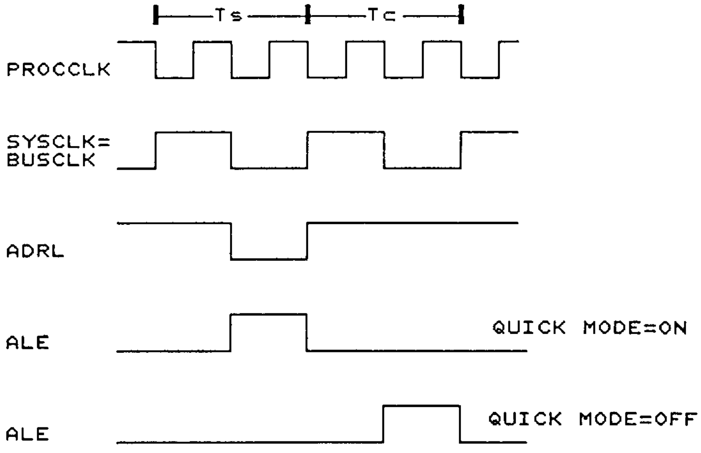
5 At the end of the second cycle, the 82C235 generates processor READY, and the CPU cycle terminates.

The I/O channel clock signal, BUSCLK, may be programmed to have a frequency of either PROCCLK/2 or PROCCLK/4 under control of bit 2 of internal configuration register 41H. All bus timings are with respect to BUSCLK. See Figure 1.15 for the relationship between PROCCLK, ADRL, BUSCLK, and ALE for different modes. Also see Figure 1.16 for the I/O cycle relationship with the programmed bus clock. Detailed timing diagrams of typical I/O channel cycles are included in the timing section.

The 82C235 supports DMA masters on the I/O channel. A device on an I/O channel can become master and control I/O channel operations. The MASTER* line is sampled during DMA to determine if an external bus master wishes to control the I/O channel. If the line is active, the 82C235 expects that the external address buffers have reversed direction. The 82C235 address lines also reverse direction (they normally drive out during DMA), so the internal DRAM controller can obtain the address. The memory and I/O control signals also reverse while MASTER* is active. The 82C235 generates MEMCS16* or IOCS16* as required for DRAM and internal register accesses. X-Bus peripherals (other than 82C235 controlled ROM) must emit MEMCS16* or IOCS16* themselves.

The REFRESH* line is also bi-directional in the master environment. If the master holds the I/O channel more than 15us, it should initiate a refresh cycle. It accomplishes this by holding the REFRESH* line active until the 82C235 refresh state machine asserts XMEMR*.

$$\mathrm{BUSCLK}=\frac{\mathrm{PROCCLK}}{2}$$



$$\mathrm{BUSCLK}=\frac{\mathrm{PROCCLK}}{4}$$

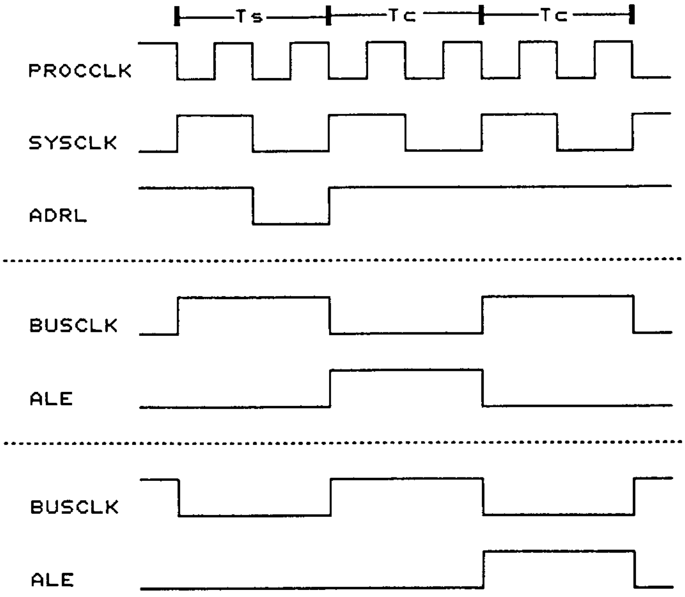
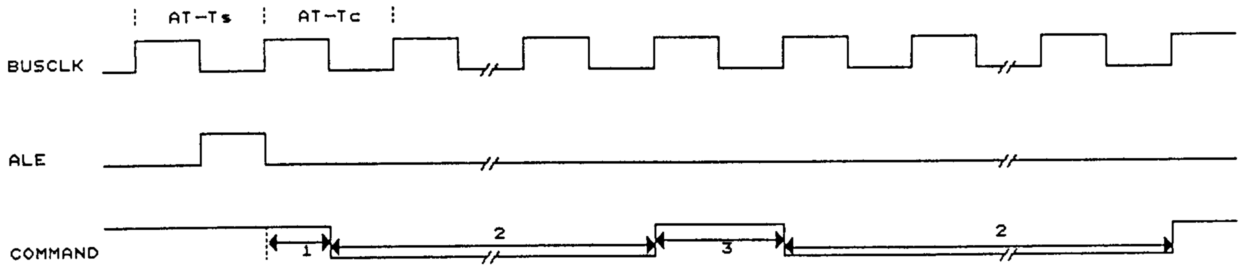


Figure 1.15 I/O Channel Timing Relationships in Different Modes



		16-BIT MEM	16-BIT I/O	8-BIT MEM	8-BIT I/O
1	CMD DLY FROM ALE	0	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
2	CMD WIDTH	2	$4\frac{1}{2}$	$4\frac{1}{2}$	$4\frac{1}{2}$
3	CMD DLY FROM FIRST CMD DURING CONVERSION	X	X	$1\frac{1}{2}$	$1\frac{1}{2}$

* NUMBERS ARE IN BUSCLK PERIODS

Figure 1.16 I/O Channel Default Cycle for 8-Bit/16-Bit, XD/SD Bus Cycle

2.12 Power Management

Power management circuitry reduces system power consumption by slowing or stopping the processor clock (PROCCLK) during idle periods. A non-static CPU design allows the processor clock to be slowed down, while a static CPU design allows the processor to be stopped completely.

Internal configuration register 46H, bits 3-2, are used to reduce the frequency of PROCCLK by dividing the CPUX1 input clock signal by two, four, or eight. Note, however, that PROCCLK returns to its normal frequency (CPUX1) during DMA or refresh operations (refer to Figure 1.17). PROCCLK reflects the frequency programmed into internal configuration register 46H, bits 1-0. The default value of this field halts PROCCLK. Figure 1.18 indicates how PROCCLK behaves during sleep mode implementation.

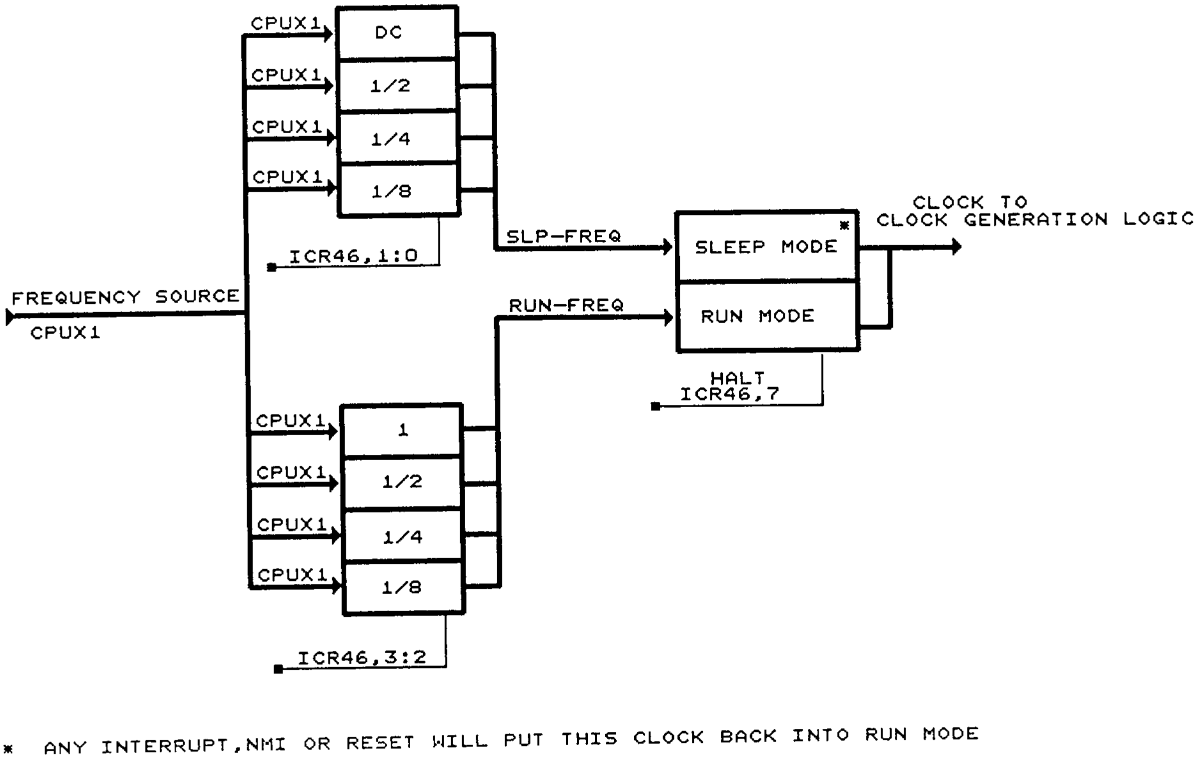
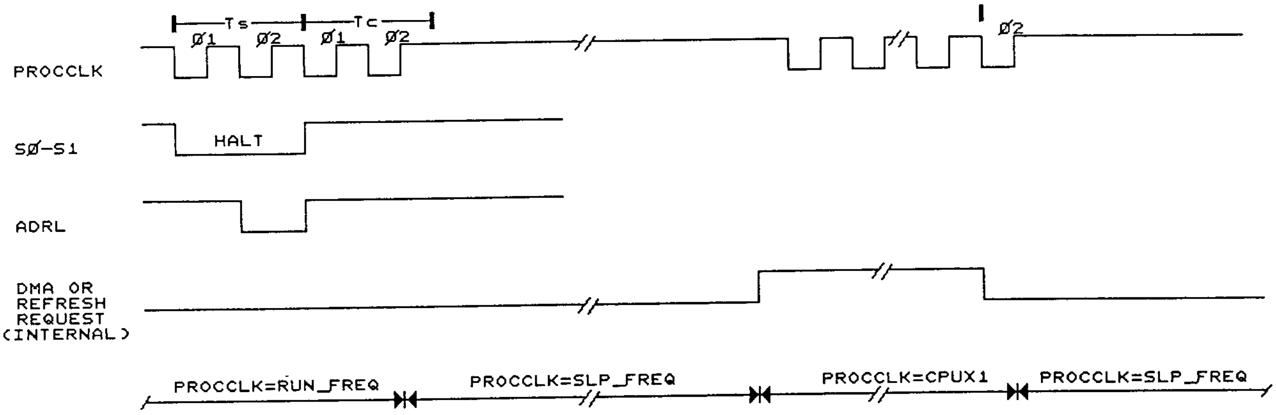


Figure 1.17 Clock Management

If a static CMOS processor is used in the system, PROCCLK can be halted completely using internal configuration register 46H, bit 7, to enable *sleep mode*. If sleep mode is enabled, a HALT instruction will put the 82C235 into sleep mode and CPU power consumption will be reduced to standby levels. Sleep mode is aborted by all interrupts, maskable or not, and reset conditions.



ICR46H, 7=1
ICR46H,3-2=ØØ
ICR46H,1-Ø=ØØ

Figure 1.18 Sleep Mode

3 Address Mapping

This section describes 82C235 I/O and memory maps.

3.1 I/O Address Maps

The following tables indicate the I/O ports used in the 82C235. If an address is not shown here, it may not be used. (Refer to an IBM PC AT technical reference manual for reserved locations.)

DMA Controller 1	I/O Port Address	R/W	Description
	0000H	R/W	DMA Channel 0 current address
	0001H	R/W	DMA Channel 0 current word count
	0002H	R/W	DMA Channel 1 current address
	0003H	R/W	DMA Channel 1 current word count
	0004H	R/W	DMA Channel 2 current address
	0005H	R/W	DMA Channel 2 current word count
	0006H	R/W	DMA Channel 3 current address
	0007H	R/W	DMA Channel 3 current word count
	0008H	R/W	Command/Status Register
	0009H	R/W	DMA Request Register
	000AH	R/W	DMA Single Bit Mask Register
	000BH	R/W	DMA Mode Register
	000CH	R/W	DMA Clear Byte Pointer
	000DH	R/W	DMA Master Clear
	000EH	R/W	Clear Mask Register
	000FH	R/W	DMA Write All Mask Register Bit

Interrupt Controller 1	I/O Port Address	R/W	Description
	0020H	W	INTC ICW1
		W	INTC OCW2
		W	INTC OCW3
		R	INTC Interrupt Request Register (IIR)
		R	INTC In-Service Register (ISR)
		R	INTC Polling Data Byte
	0021H	W	INTC ICW2
		W	INTC ICW3
		W	INTC ICW4
		W	INTC OCW1
		R	INTC Interrupt Mask Register (IMR)

Internal Configuration Registers (ICRs)	I/O Port Address	R/W	Description
	0022H	WO	Internal Configuration Index Register
	0023H	R/W	Internal Configuration Data Register

	Index	R/W	Description
	01H	R/W	DMA Wait-state Control Register
	40H	RO	Version Register
	41H	R/W	Channel Control Register
	42H	R/W	Reserved
	43H	R/W	Reserved
	44H	R/W	Peripheral Control Register
	45H	R/O	Miscellaneous Status Register
	46H	R/W	Power Management Register
	47H	---	Reserved
	48H	R/W	ROM Enable Register
	49H	R/W	RAM Write Protect Register
	4AH	R/W	Shadow RAM Enable Register 1
	4BH	R/W	Shadow RAM Enable Register 2
	4CH	R/W	Shadow RAM Enable Register 3
	4DH	R/W	DRAM Configuration Register
	4EH	R/W	Extended Boundary Register
	4FH	R/W	EMS Control Register

Timer/Counter Registers	I/O Port Address	R/W	Description
	0040H	R/W	Timer 0 Count Load/Read
	0041H	R/W	Timer 1 Count Load/Read
	0042H	R/W	Timer 2 Count Load/Read
	0043H	W	Timer Control Word

Miscellaneous I/O Registers	I/O Port Address	R/W	Description
	0061H	R/W	Control/Status Port
	0070H	W	Real-Time Clock Index and NMI Mask
	0071H	R/W	Real-Time Clock Data Port
	0092H	R/W	System Control Port

DMA Page Registers	Address	R/W	Description
	0081H	W	DMA Channel 2 Page Register
	0082H	W	DMA Channel 3 Page Register
	0083H	W	DMA Channel 1 Page Register
	0087H	W	DMA Channel 0 Page Register
	0089H	W	DMA Channel 6 Page Register
	008AH	W	DMA Channel 7 Page Register
	008BH	W	DMA Channel 5 Page Register

Interrupt Controller 2	Address	R/W	Description
	00A0H	W	INTC ICW1
		W	INTC OCW2
		W	INTC OCW3
		R	INTC Interrupt Request Register (IIR)
		R	INTC In-Service Register (ISR)
		R	INTC Polling Data Byte
	00A1H	W	INTC ICW2
		W	INTC ICW3
		W	INTC ICW4
		W	INTC OCW1
		R	INTC Interrupt Mask Register (IMR)

DMA Controller 2	Address	R/W	Description
	00C0H	R/W	Channel 0 base and current address
	00C2H	R/W	Channel 0 base and current word count
	00C4H	R/W	Channel 1 base and current address
	00C6H	R/W	Channel 1 base and current word count
	00C8H	R/W	Channel 2 base and current address
	00CAH	R/W	Channel 2 base and current word count
	00CCH	R/W	Channel 3 base and current address
	00CEH	R/W	Channel 3 base and current word count
	00D0H	R/W	Read Status Register/Write Comnd Reg
	00D2H	R/W	Write Request Register
	00D4H	R/W	Write Single Mask Register Bit
	00D6H	R/W	Write Mode Register
	00D8H	R/W	Clear Byte Pointer Filp-Flop
	00DAH	R/W	Read Temporary Register/Write Mstr Clr
	00DCH	R/W	Clear Mask Register
	00DEH	R/W	Write All Mask Register Bits
	000FH	R/W	DMA Write All Mask Register Bit

Coprocessor Registers	Address	R/W	Description
	00F0H	W	Clear Co-processor Busy
	00F1H	W	Reset Co-processor

EMS Page Registers	Address	R/W	Description
	02x8H	R/W	EMS Page Register
	02x9H	R/W	EMS Page Register
	02xAH	R/W	EMS Page Register

Note--x = 0 or 1 depending on ICR 4F, bit 0

3.2 Memory Address Map

The following table indicates the memory addresses used in the 82C235. For more detailed information about memory addressing, refer to the IBM PC AT Technical Reference Manual.

Address	R/W	Description
000000-09FFFFH	R/W	System RAM
0A0000-0BFFFFH	R/W	Video Memory or Shadow RAM
0C0000-0EFFFFH	R/W	BIOS extension or Shadow RAM
0F0000-0FFFFFH	R/W	ROM or Shadow RAM
100000-FBFFFFH	R/W	Expanded or Extended Memory
FC0000-FEFFFFH	R/O	BIOS extension
FF0000-FFFFFFH	R/W	ROM

4 Configuration Register Definitions

This section of the data manual describes two types of registers: configuration registers that are internal to the 82C235 and those that are IBM PC AT-compatible registers.

4.1 Internal Configuration Registers

Internal configuration registers (ICRs) control the enhanced options of the 82C235. ICRs are accessed by addressing an index register at port 22H and by immediately writing (or reading) data to (or from) port 23H. The 82C235 contains ICRs at index values 40H through 4FH. In the definitions that follow, power-on default states of the registers are given. All ICRs are read/write except as noted. All *reserved* bits must be written as zero unless noted (the results are unpredictable). The value read from a reserved bit is not defined.

Index	Register	Bit	Description/Bit State
01H	DMA Wait-state Control		The clock and wait-state control subsystem performs two functions: control of the DMA command width and selection of the DMA clock rate.
		7-6	Reserved.
		5-4	16-bit DMA WAIT-STATE 1:0. These bits control the number of wait-states inserted during 16-bit DMA transfers. The following table describes the settings and operation. These bits default to a logic 00 setting. 00 = One wait-state (default) 01 = Two wait-states 10 = Three wait-states 11 = Four wait-states
		3-2	8-bit DMA WAIT-STATE 1:0. These bits control the number of wait-states inserted during 8-bit DMA transfers. The following table describes the settings and operation. These bits default to a logic 00 setting. 00 = One wait-state (default) 01 = Two wait-states 10 = Three wait-states 11 = Four wait-states
		1	DMA XMEMR EXT DIS. The assertion of XMEMR* is delayed by one DMA clock cycle later than XIOR* in the IBM PC AT. This may not be desirable in some systems. 0 = Enables extended XMEMR* function (default). 1 = Disables extended XMEMR* function (starts XMEMR* at the same time as XIOR*).
		0	DMA CLOCK SEL. This bit allows the user to program the DMA clock to operate at either CPUX1/4 or CPUX1/2. 0 = CPUX1/4. Drives both 8- and 16-bit operations (default). 1 = CPUX1/2. **NOTE--**If this bit is changed during operation, an internal synchronizer controls the actual switching of the clock to prevent a short clock pulse from causing a DMA malfunction.

Index	Register	Bit	Description/Bit State
40H	Version		Read Only.
		7-4	Family type 3-0 identifies the specific part within a pin-compatible family. The family type of the 82C235 is zero.
		3-0	Device revision code 3-0 identifies the revision level of the 82C235 silicon.
41H	Clock Control		
		7	Reserved.
		6	QUICK MODE. When enabled, ALE is generated for all cycles. Non-channel cycles generate ALE but no command appear. The channel cycles in this mode do not have any additional wait-states. This mode is used in slow systems (< = 10 MHz) where BUSCLK is forced to PROCCLK/2. 0 = quick mode disabled (default). 1 = quick mode enabled.
		5-3	Reserved.
		2	BUSCLK SEL. 0 = selects BUSCLK as PROCCLK/4 (default). 1 = selects BUSCLK as PROCCLK/2.
		1-0	RFSH CMDW 1-0. These bits specify the command pulse width during a refresh cycle. These pulse widths are derived from OSC. 00 = 140ns 01 = 210ns 10 = 280ns (default) 11 = 350ns
42H			Reserved.
43H			Reserved.

Index	Register	Bit	Description/Bit State
44H	Peripheral Control		This register controls whether peripherals are on the XD-bus or the SD-bus. The SDIRH and SDIRL SD-bus controls are affected by bits in this register.
		7	Reserved.
		6	CHIPS VIDEO ON THE XD-BUS. When this bit is high, a CHIPS video controller is resident on the XD-bus. The I/O address ranges covered for the video controller are 0102H-0104H, 03B0H-03DFH, and 46E8H. Memory address ranges are between A0000H and BFFFFH. 0 = CHIPS video controller is on the SD-bus (default). 1 = CHIPS video controller is on the XD-bus.
		5	GAME PORT ON THE XD-BUS. When this bit is high, game port is resident on the XD-bus. The I/O address range covered for the game port is 0200H-0207H. 0 = game port on the SD-bus (default). 1 = game port on the XD-bus.
		4	SERIAL PORT CHANNEL 2 ON THE XD-BUS. When this bit is high, serial port 2 is resident on the XD-bus. The I/O address range covered for serial port 2 is 02F8H-02FFH. 0 = serial port 2 on the SD-bus (default). 1 = serial port 2 on the XD-bus.
		3	SERIAL PORT CHANNEL 1 ON THE XD-BUS. When this bit is high, serial port 1 is resident on the XD-bus. The I/O address range covered for serial port 1 is 03F8H-03FFH. 0 = serial port 1 on the SD-bus (default). 1 = serial port 1 on the XD-bus.
		2	PARALLEL PORT ON THE XD-BUS. When this bit is high, a parallel port is resident on the XD-bus. The address range covered for the parallel port is 0378H-037FH. 0 = parallel port on the SD-bus (default). 1 = parallel port on the XD-bus.
		1	HDC/FDC ON THE XD-BUS. When this bit is high, the hard drive and diskette drive controllers are resident on the XD-bus. The I/O address range covered for HDC/FDC is 01F0H-01F7H and 03F0H-03F7H. 0 = HDC/FDC on the SD-bus (default). 1 = HDC/FDC on the XD-bus.
		0	Reserved.

Index	Register	Bit	Description/Bit State
45H	Miscellaneous Status		Read Only.
		7	NMI DISABLE. This bit indicates the current state of the NMI disable bit of I/O port 70, bit 7. Bit 7 of I/O port 70 is write only. 0 = NMI enabled. 1 = NMI disabled.
		6	GATEA20 FROM 8042. This bit indicates the current state of the GATEA20 input from the 8042 keyboard controller. 0 = GATEA20 is low. 1 = GATEA20 is high.
		5	BUSY TO 80286. This bit indicates the current state of the latched coprocessor busy status. 0 = coprocessor busy. 1 = coprocessor not busy.
		4	INTERNAL RTC INSTALLED. This bit indicates the state of the EXRTC*(DACK5*) pin during reset. 0 = external RTC used. 1 = internal RTC used.
		3	Reserved.
		2-0	SENSE LINES 2-0. These lines (2-0) sense the power-up state of DACK lines (2-0), respectively. A logic zero is latched if a 4.7K ohm pull-down resistor is used. A logic one is latched if a 4.7K ohm pull-up resistor is used. These lines can be used for power-on setup parameters. 0 = corresponding DACK* line pulled low. 1 = corresponding DACK* line pulled high.

Index	Register	Bit	Description/Bit State
46H	Power Management		This register controls power management features. It controls sleep mode functions and PROCCLK frequency under software control. The PROCCLK frequency can be reduced when CPU speed is not critical. This results in power savings.
		7	SLEEP ENABLE. This bit enables sleep mode. If this bit is set, sleep mode will be entered upon execution of a HALT instruction. 0 = sleep mode disabled (default). 1 = sleep mode enabled.
		6	AUX PARITY DISABLE (read/write). This bit, in conjunction with the parity disable bit (PARITY DISABLE) of I/O port 61H, is used to disable parity error contributions to the NMI. The bit is logically ORed with the PARITY DISABLE bit. Thus, system parity is disabled if either this bit (AUX PARITY DIS) or the PARITY DISABLE bit is set to a logical one. 0 = auxiliary parity check is enabled (default). 1 = auxiliary parity check is disabled.
		5-4	Reserved.
		3-2	RUN FREQ. These bits select the frequency of the PROCCLK signal to the 80286 CPU when in normal run mode. 00 = CPUX1 (default) 01 = CPUX1/2 10 = CPUX1/4 11 = CPUX1/8
		1-0	SLEEP FREQ. These bits select the frequency of the PROCCLK signal to the 80286 CPU when in sleep mode. 00 = high-level DC (default) 01 = CPUX1/2 10 = CPUX1/4 11 = CPUX1/8
47H			Reserved. **DO NOT WRITE TO THIS REGISTER.**

Index	Register	Bit	Description/Bit State
48H	ROM Enable		This register is used to select the address ranges to be included in the ROMCS* output. The power-on default address range is 0F0000H-0FFFFFH. The range from FC0000H to FFFFFFH is always enabled.
		7	0F8000H-0FFFFFH ROM ENABLE. This bit asserts ROMCS* for accesses to these ranges. 0 = disables ROMCS* for the above range. 1 = enables ROMCS* for the above range (default).
		6	0F0000H-0F7FFFH ROM ENABLE. This bit asserts ROMCS* for accesses to these ranges. 0 = disables ROMCS* for the above range. 1 = enables ROMCS* for the above range (default).
		5	0E8000H-0EFFFFH ROM ENABLE. This bit asserts ROMCS* for accesses to these ranges. 0 = disables ROMCS* for the above range (default). 1 = enables ROMCS* for the above range.
		4	0E0000H-0E7FFFH ROM ENABLE. This bit asserts ROMCS* for accesses to these ranges. 0 = disables ROMCS* for the above range (default). 1 = enables ROMCS* for the above range.
		3	0D8000H-0D8FFFFH ROM ENABLE. This bit asserts ROMCS* for accesses to these ranges. 0 = disables ROMCS* for the above range (default). 1 = enables ROMCS* for the above range.
		2	0D0000H-0D7FFFH ROM ENABLE. This bit asserts ROMCS* for accesses to these ranges. 0 = disables ROMCS* for the above range (default). 1 = enables ROMCS* for the above range.
		1	0C8000H-0CFFFFH ROM ENABLE. This bit asserts ROMCS* for accesses to these ranges. 0 = disables ROMCS* for the above range (default). 1 = enables ROMCS* for the above range.
		0	0C0000H-0C7FFFH ROM ENABLE. This bit asserts ROMCS* for accesses to these ranges. 0 = disables ROMCS* for the above range (default). 1 = enables ROMCS* for the above range.

Index	Register	Bit	Description/Bit State
49H	RAM Write Protect		This register is used to write protect portions of RAM addresses.
		7	0F8000H-0FFFFFH RAM READ ONLY. This bit makes this address range read only. 0 = enables read/write operations for the above range (default). 1 = enables read only operations for the above range.
		6	0F0000H-0F7FFFH RAM READ ONLY. This bit makes this address range read only. 0 = enables read/write operations for the above range (default). 1 = enables read only operations for the above range.
		5	0E8000H-0EFFFFH RAM READ ONLY. This bit makes this address range read only. 0 = enables read/write operations for the above range (default). 1 = enables read only operations for the above range.
		4	0E0000H-0E7FFFH RAM READ ONLY. This bit makes this address range read only. 0 = enables read/write operations for the above range (default). 1 = enables read only operations for the above range.
		3	0D8000H-0DFFFFH RAM READ ONLY. This bit makes this address range read only. 0 = enables read/write operations for the above range (default). 1 = enables read only operations for the above range.
		2	0D0000H-0D7FFFH RAM READ ONLY. This bit makes this address range read only. 0 = enables read/write operations for the above range (default). 1 = enables read only operations for the above range.
		1	0C8000H-0CFFFFH RAM READ ONLY. This bit makes this address range read only. 0 = enables read/write operations for the above range (default). 1 = enables read only operations for the above range.
		0	0C0000H-0C7FFFH RAM READ ONLY. This bit makes this address range read only. 0 = enables read/write operations for the above range (default). 1 = enables read only operations for the above range.

Index	Register	Bit	Description/Bit State
4AH	Shadow RAM Enable 1		This register controls whether shadow RAM is enabled or disabled for individual 16KB segments of the address range from 0A0000H to 0BFFFFH.
		7	0BC000H:0BFFFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		6	0B8000H:0BBFFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		5	0B4000H:0B7FFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		4	0B0000H:0B3FFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		3	0AC000H:0AFFFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		2	0A8000H:0ABFFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		1	0A4000H:0A7FFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		0	0A0000H:0A3FFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.

Index	Register	Bit	Description/Bit State
4BH	Shadow RAM Enable 2		This register controls whether shadow RAM is enabled or disabled for individual 16KB segments of the address range from 0C0000H to 0DFFFFH.
		7	0DC000H:0DFFFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		6	0D8000H:0DBFFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		5	0D4000H:0D7FFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		4	0D0000H:0D3FFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		3	0CC000H:0CFFFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		2	0C8000H:0CBFFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		1	0C4000H:0C7FFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		0	0C0000H:0C3FFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.

Index	Register	Bit	Description/Bit State
4CH	Shadow RAM Enable 3		This register controls whether shadow RAM is enabled or disabled for individual 16KB segments of the address range from 0E0000H to 0FFFFFH.
		7	0FC000H:0FFFFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		6	0F8000H:0FBFFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		5	0F4000H:0F7FFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		4	0F0000H:0F3FFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		3	0EC000H:0EFFFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		2	0E8000H:0EBFFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		1	0E4000H:0E7FFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.
		0	0E0000H:0E3FFFH SHADOW RAM. 0 = disables this 16KB segment (default). 1 = enables shadow RAM in this 16KB segment.

Index	Register	Bit	Description/Bit State
4DH	DRAM Configuration		
		7	Reserved.
		6-4	LOCAL DRAM TIMING 2-0. These bits define the local DRAM timing as shown in the following table.

Bit 6 5 4	Timing
0 0 0 =	zero processor wait-states
0 0 1 =	one processor wait-state (default)
0 1 0 =	one processor wait-state with RAS* delay

RAS* signals are delayed by PROCCLK/2 in this mode.

3-0 CFG3-0. These bits specify the planar DRAM configuration as shown in following table. The default value after reset is 0001 (512KB total DRAM).

Bit 3210	B0	B1	B2	B3	Total Memory	Low Memory	EMS/Extended Memory
0000	0	0	0	0	0	0	0
0001	512KB	0	0	0	512KB	512KB	0
0010	512KB	128KB	0	0	640KB	640KB	0
0011	512KB	512KB	0	0	1MB	640KB	384KB
0100	512KB	512KB	0	0	1MB	1MB	0
0101	512KB	512KB	512KB	0	1.5MB	1MB	512KB
0110	512KB	512KB	512KB	512KB	2MB	1MB	1MB
0111	512KB	512KB	512KB	512KB	4MB*	1MB	3MB
1000	2MB	0	0	0	2MB	1MB	1MB
1001	2MB	2MB	0	0	4MB	1MB	3MB
1010	2MB	2MB	2MB	0	6MB	1MB	5MB
1011	2MB	2MB	2MB	2MB	8MB	1MB	7MB
1100	2MB	2MB	2MB	2MB	10MB*	1MB	9MB
1101	2MB	2MB	2MB	2MB	12MB*	1MB	11MB
1110	2MB	2MB	2MB	2MB	14MB*	1MB	13MB
1111	2MB	2MB	2MB	2MB	16MB*	1MB	15MB**

* This configuration is implemented using RAS* encoded mode, which allows a system to have eight banks of DRAM.

** DRAM in the range FC0000H to FFFFFFH is usable only as a target for EMS. ROM is always enabled for normal CPU accesses in this range.

Index	Register	Bit	Description/Bit State
4EH	Extended Boundary		
		7	RAS ENCODE ENABLE. This bit changes the four RAS* lines into three encoded RAS* lines and one RAS* timing line. These four lines are used with an external 3-8 decoder to support eight banks of DRAMs. 0 = RAS* encode disabled (default). 1 = RAS* encode enabled.
		6	Reserved.
		5	RAM DISABLE 040000H-09FFFFH. This bit disables the 82C235's internal DRAM controller for accesses in the range 040000H-09FFFFH. These accesses are directed to the I/O channel. 0 = enables 040000H-09FFFFH DRAM range (default). 1 = disables 040000H-09FFFFH DRAM range.
		4	Reserved.
		3-0	EXTENDED BOUNDARY. These bits specify the upper boundary of the extended memory within the total memory defined by the internal configuration register 4DH (see Figure 1.9).

Bit 3210	Extended Memory Boundary
0000	No Boundary (default)
0001	1MB
0010	1.25MB
0011	1.5MB
0100	2MB
0101	4MB
0110	6MB
0111	8MB
1000	10MB
1001	12MB
1010	14MB
1011	No Boundary
1100	No Boundary
1101	No Boundary
1110	No Boundary

Index	Register	Bit	Description/Bit State
4FH	EMS Control Register		
		7	GLOBAL EMS ENABLE. This bit enables the EMS memory. 0 = EMS is disabled (default). 1 = EMS is enabled.
		6	EMS I/O ENABLE. This bit enables access to the EMS I/O ports. 0 = EMS I/O port access is disabled (default). 1 = EMS I/O port access is enabled.
		5-1	Reserved.
		0	I/O BASE. This bit specifies which I/O page registers are used to read or write EMS page register contents. 0 = EMS page registers are 0208H, 0209H, and 020AH (default). 1 = EMS page registers are 0218H, 0219H, and 021AH.

4.2 IBM PC AT-Compatible Registers

I/O Port	Register	Bit	Description/Bit State
61H	Control Port/Status		This port controls several system-level functions. The port can be accessed through any *odd* I/O port address between (and including) 61H and 6FH.
		7	PARITY CHECK (read only). This bit indicates an error has occurred on the planar memory. 0 = no error occurred. 1 = an error occurred.
		6	IOCH CHK (read only). This bit indicates an I/O channel check has occurred (usually a parity error) on the system I/O channel. 0 = no error occurred. 1 = an error occurred.
		5	TMR 2 OUT (read only). This bit returns the condition of the timer 2 output.
		4	RFSH DETECT (read only). This bit toggles on each refresh cycle.
		3	CHCK DIS (read/write). This bit disables NMI generation for channel check errors. 0 = disables NMI (default). 1 = enables NMI.
		2	PARITY DISABLE (read/write). This bit, in conjunction with the auxiliary parity disable bit (AUX PARITY DISABLE) of internal configuration register 46H, is used to disable parity error controbutions to the NMI. This bit is logically ORed with AUX PARITY DISABLE. Thus, system parity is disabled if either this bit (PARITY DISABLE) or the AUX PARITY DISABLE bit is set to a logical one. 0 = parity is enabled (default). 1 = parity is disabled.
		1	SPKR DATA (read/write). This bit gates the output of channel 2 of the timer/counter. 0 = output is disabled (default). 1 = output is enabled.
		0	TMR 2 GATE (read/write). This bit controls operation of timer channel 2. 0 = channel 2 timer operation is disabled (default). 1 = channel 2 timer operation is enabled.

I/O Port	Register	Bit	Description/Bit State
70H It	RTC/CMOS Index and NMI Mask		This register is used to access to the RTC and its CMOS RAM. also masks NMIs from accessing the CPU. Bits 6-0 are not used when the external RTC option is selected.
		7	NMI DISABLE (write only). This bit disables the generation of NMIs. 0 = enables gerneration of NMIs (default). 1 = disables the generation of NMIs.
		6-0	RTCINDX6:0. These bits are used as index pointers to the 128 locations (114 user RAM, 14 RTC registers) contained in the internal real-time clock.
71H	RTC Data		This register is used to transfer data to and from the internal real-time clock. The RTC register is selected by bit 6:0 of I/O port 70H as described above. If an external RTC is used, this register must be provided externally.
		7-0	RTCDATA7:0.
92H	System Control		This register is used as a faster alternative to gating A20 and resetting the CPU rather than using the 8042 keyboard controller. This register is compatible with IBM PS/2 architecture.
		7-2	Reserved.
		1	ALT GATEA20. 0 = MODA20 is forced low. 1 = Address bit A20 (on the CPU) goes directly to the MODA20 pin.
		0	ALT CPU RESET. 0 to 1 transition = a reset pulse is provided on the CPURST pin to reset the CPU. After setting, the state is maintained after a CPU reset so the BIOS can determine if the reset was caused by a CPU RESET condition.

5 Pin Descriptions, Characteristics, and Listings

This section of the data manual contains information about the input and output signals from the 82C235. Specifically, the section begins with tables containing the signal names and their descriptions. These tables are separated into categories of signal types such as clock or control signals.

These tables are followed by an illustration of the 160-pin plastic flat pack that indicates how the pins are numbered on the device. The illustration is followed by two tables that list the signal names and their associated pin numbers, first in alphabetical order and then in numerical order.

5.1 Signal Descriptions (82C235 Flat Pack)

A hyphenated signal name specifies a set of signals. For example, RAS0-3* indicates four signals: RAS0*, RAS1*, RAS2*, and RAS3*. A signal name appended by an asterisk, such as RAS0*, specifies an active-low signal.

The following legend applies to the *Type* column in the tables.

Abbreviation	Meaning
I	Input
O	Output
B	Bi-directional

5.1.1 Clock Input and Output Signals

Pin	Type	Name	Description
122	I	OSC1	**Crystal Oscillator Input** is a 14.318MHz input either from an external crystal oscillator or from a parallel resonance, fundamental mode crystal that is also tied to OSC2.
121	O	OSC2	**Crystal Oscillator Output** is the output from the internal crystal oscillator. If an external crystal oscillator is used, leave OSC2 unconnected.
3	I	CPUX1	**CPU Clock Oscillator Input** is an input used to derive PROCCLK. CPUX1 is either from an external crystal oscillator or from a parallel resonance, fundamental mode crystal that is also tied to CPUX2.
4	O	CPUX2	**CPU Clock Oscillator Output** is an output from the internal crystal oscillator. If an external crystal oscillator is used, leave CPUX2 unconnected.
2	O	PROCCLK	**Processor Clock** is the clock output to the 80286 processor. The frequency of PROCCLK is twice the processor's internal clock frequency.
56	O	BUSCLK	**Bus Clock** is the bus clock output, programmable to one-half or one-fourth the frequency of CPUX1.

5.1.2 Local Bus Interface Signals

Pin	Type	Name	Description
5-18 20-21 23-30	B B B	A00-A13 A14-A15 A16-A23	**Address Bus Bits 00 to 23** are generated by the 80286 during processor cycles. During HLDA (bus hold acknowledge) the 82C235 drives the address bus. During I/O channel master operations a DMA controller or a microprocessor drives the address bus. A00 and A20 are driven by the 82C235 during I/O channel master operation.
58	B	MODA0	**Modified A00** is the internally latched state of address bit A00 that is to be routed to the XD-bus interface instead of A00. The 82C235 toggles this bit during conversion cycles. MODA0 is an input signal during master operation from the I/O channel.
160	B	D00	**Local Data Bus Bits 00 to 15** compose the local data bus that is connected directly to the processor.
158	B	D01	
156	B	D02	
154	B	D03	
152	B	D04	
150	B	D05	
148	B	D06	
146	B	D07	
159	B	D08	
157	B	D09	
155	B	D10	
153	B	D11	
151	B	D12	
149	B	D13	
147	B	D14	
145	B	D15	
32, 31	I	S1*, S0*	**Processor Status Signals 1 and 2** are active-low inputs from the 80286 that, along with M/IO*, define the type of cycle. S1* and S0* are tri-stated during a hold acknowledge state.
34	I	M/IO*	**Memory/IO select** is the input from 80286 that distinguishes between memory accesses and I/O accesses. If M/IO* is high during T_S, a memory cycle is in progress. If low, an I/O cycle is in progress.
33	B	BHE*	**Byte High Enable** is an input from the 80286 during processor cycles and an output during DMA cycles. BHE* and A00 indicate the type of bus transfer. BHE* is pulled high internally. **BHE* A00 Function** 0 0 Word Transfer 0 1 Odd Byte Transfer 1 0 Even Byte Transfer 1 1 Reserved
37	O	READY*	**READY** is an active-low signal that indicates the end of a cycle to the 80286. READY* is driven by the 82C235 and should be connected to the 80286's READY* line.
36	O	HOLD	**Hold Request** is an active-high output to the 80286 processor that requests bus access for a DMA or a refresh cycle. HOLD should be connected to the 80286 processor's HOLD pin.
35	I	HLDA	**Hold Acknowledge** is an active-high input from the 80286 processor that indicates when the bus has been given up by the processor. HLDA should be connected to the 80286 processor's HDLA pin.
98	B	MODA20	**Modified(Gated) A20** is gated A20 from the 82C235's gate A20 logic and should be used instead of CPU A20. MODA20 is an input during I/O channel master operations.

Pin	Type	Name	Description
143	0	NMI	**Non-Maskable Interrupt** is generated as a result of a parity error or an I/O channel error. A low-to-high transition on NMI causes a non-maskable interrupt at the end of the current cycle. NMI should be connected to the 80286 processor's NMI pin.
144	O	INTR	**Interrupt Request** is an active-high request to the 80286 that it suspend the current process and acknowledge the request. INTR should be connected to the 80286 processor's INTR pin.

5.1.3 Numeric Processor Interface Signals

Pin	Type	Name	Description
133	O	BUSY*	**Numeric Processor Busy Status** is an active-low output to the BUSY* input of the 80286 that echoes the state of BUSY* from the 80287.
127	I	NPBZ*	**Busy Status** is an active-low input that comes directly from the 80287 numeric processor's BUSY* pin. The signal indicates that the 80287 is currently executing a command.
131	O	NPCS*	**Numeric Processor Chip Select** is an active-low output that goes directly to the NPS1* input of the 80287 numeric processor. NPCS* is the address decode for the 80287 select, and is active for addresses 0F8H-0FFH.
128	I	ERROR*	**Error** is an active-low input from the 80287 numeric processor that indicates an 80287 error.
132	O	NPRST	**Numeric Processor Reset** is an active-high output that resets the 80287 numeric processor. NPRST is activated during power-up and write operations to I/O port 0F1H.

5.1.4 Memory Interface Signals

Pin	Type	Name	Description
108-117	O	MA0-MA9	**Multiplexed DRAM Address Bits MA 0 to 9** are outputs to the DRAMs, which must be appropriately buffered and terminated.
107-104	O	RAS0-3*	**Row Address Strobes 0 to 3** are active-low strobes used as RAS inputs to the banks of DRAM. Each bank is 18 bits wide (2 bits for parity). Each byte is addressed with a low or high CAS signal. For further details, refer to the DRAM section of this data sheet. RAS*0-3 perform different functions in an encoded RAS mode. Refer to the DRAM section of this document to see the implementation of this mode.
101	O	CASH*	**Column Address Strobe High** is an active-low output to all high byte DRAMs.
100	O	CASL*	**Column Address Strobe Low** is an active-low output to all low byte DRAMs.
119	B	PARH	**Parity High** is the parity bit from the high-order bytes of the DRAMs. If the bit is high, it indicates there are an odd number of ones in the high byte, which includes the parity bit itself. Thus, parity is odd.
118	B	PARL	**Parity Low** is the parity bit from the low-order bytes of the DRAMs. If the bit is high, it indicates there are an odd number of ones in the low byte, which includes the parity bit itself. Thus, parity is odd.

Pin	Type	Name	Description
103	O	MWE*	**Memory Write Enable** is an active-low output connected to all DRAMs. MWE* is normally low, but is high for read cycles. This signal is also used to control the memory data buffers; it controls the direction of the transceivers that buffer the ROM/DRAM array from the local data bus.
38	B	ROMCS*	**ROM Chip Select** is an active-low output to the EPROM(s). ROMCS* becomes active for the programmed address range. For further information, refer to the section on ROM interface (2.8.1).

5.1.5 I/O Channel Interface Signals

Pin	Type	Name	Description
60-61 63-76	B B	XD00-XD01 XD02-XD15	**16-Bit Data Bus** is the 16-bit system data bus. 16-bit CPU cycles are converted to one 16-bit or two 8-bit bus cycles depending on the states of IOCS16* and MEMCS16*. The I/O channel data bus (SD-bus) is buffered from this bus.
79	B	XIOR*	**I/O Read Command** is an active-low output used by I/O devices to put their data on the bus. This signal is used by on-board peripherals as well as the I/O channel. During I/O channel master operations XIOR* is an input command to the peripherals and the 82C235.
80	B	XIOW*	**I/O Write Command** is an active-low output used by I/O devices to capture data from the bus. This signal is used by XD-bus peripherals as well as the I/O channel. During I/O channel master operations XIOW* is an input command to the peripherals and the 82C235.
77	B	XMEMR*	**Memory Read Command** is an active-low output used by XD-bus video memory and I/O channel memory. During I/O channel master operations XMEMR* is an input command to the peripherals and the 82C235. During memory refresh cycles XMEMR* is always an output.
78	B	XMEMW*	**Memory Write Command** is an active-low output used by XD-bus video memory and I/O channel memory. During I/O channel master operations XMEMW* is an input command to the peripherals and the 82C235.
124	B	MREF*	**Memory Refresh Control** is an active-low output to the I/O channel that indicates a refresh cycle. During I/O channel master operations MREF* is an input to the peripherals and the 82C235.
57	O	ALE	**Address Latch Enable** is an active-high output used to latch valid addresses on the I/O channel. ALE is forced high during DMA cycles.
41	I	IOCHRDY	**I/O Channel Ready** is used by I/O channel or XD-bus devices to lengthen their R/W cycles. Normally, IOCHRDY is high; it is pulled low for extending the cycle time. This input should be driven by an open collector output driver.
42	I	IOCHCK*	**I/O Channel Check** is an active-low signal from the I/O channel used to trigger an NMI in the 80286 processor in the event of an unrecoverable I/O channel error.
54-50 49 84-86 88-87	I I I I	IRQ03-07 IRQ09 IRQ10-12 IRQ14-15	**Interrupt Requests 3-7, 9, 10-12, 14-15** are asynchronous inputs to the 82C235 interrupt controllers. These requests are prioritized with IRQ03 having the highest priority and IRQ15 the lowest. The request line is held active until acknowledged by the processor with an interrupt acknowledge cycle.

Pin	Type	Name	Description
90 48 46 44 92 94 96	I I I I I I I	DRQ0 DRQ1 DRQ2 DRQ3 DRQ5 DRQ6 DRQ7	**DMA Requests 0-3, 5-7** are asynchronous requests used by peripherals to request DMA services. These requests are prioritized with DRQ0 having the highest priority and DRQ7 the lowest. DRQ must be held active until the corresponding DMA acknowledge (DACK) line goes active.
89 47 45 43 91 93 96	O O O O O O O	DACK0* DACK1* DACK2* DACK3* DACK5* DACK6* DACK7*	**DMA Acknowledge 0-3, 5-7** are active-low acknowledge signals issued by the 82C235 after a DMA service request (via a DRQ line) and successful arbitration.
55	O	TC	**Terminal Count** is an active-high output pulse to the I/O channel that indicates the end of a DMA transfer.
81	O	LOMEGCS*	**Low Meg Chip Select** is an active-low output that is a decode of memory accesses below 1MB. This output is used to generate memory command signals for the lower 1MB address range.
97	I	MASTER*	**Master** is an active-low input from the I/O channel's 16-bit extension. MASTER* allows a microprocessor or a DMA controller residing on the I/O channel to control the system address, data, and control lines.
40	I	0WS*	**Zero Wait-state** is an active-low input from the I/O channel. This signal allows the present bus cycle to terminate without inserting any additional wait-states. 0WS should be driven with an open collector or a tri-state driver.
83	B	IOCS16*	**I/O 16-bit Chip Select** is an active-low signal. IOCS16*, an input from the I/O channel and XD-bus peripherals, indicates that the present cycle is a 16-bit I/O cycle. IOCS16* is an output for bus cycles controlled directly by the 82C235. IOCS16* should be driven with an open collector or tri-state driver.
82	B	MEMCS16*	**Memory 16-bit Chip Select** is an active-low signal. MEMCS16*, an input from the I/O channel and XD-bus peripherals, indicates that the present cycle is a 16-bit memory cycle. MEMCS16* is an output for bus cycles controlled directly by the 82C235. MEMCS16* should be driven with an open collector or tri-state driver.

5.1.6 Miscellaneous Signals

Pin	Type	Name	Description
137	O	ADRL*	**Address Latch** is an active-low output strobe that latches the addresses.
138	O	CPURST	**CPU Reset** is an active-high output that resets the 80286 processor. CPURST goes active at power-up, when PWRGOOD goes low, or during a software controlled *fast reset*.
134	O	XRST	**Peripheral Reset** is an active-high output that resets external peripherals during power-up.
123	O	SPKOUT	**Speaker Data** is a waveform (the output of channel 2 of the timer/counter gated by bit 1 of port 61H) that is routed to a driver circuit, a low-pass filter, and then to the speaker.

Pin	Type	Name	Description
136 135	O	SDIRH SDIRL	**Channel Data Bus Control** are outputs that control the direction of the data buffer between 82C235 and the I/O channel. When the signals are high (the default), data flows from the system to the I/O channel; when the signals are low, data flows from the I/O channel to the system.
39	I	PWRGOOD	**Power Good** is an active-high input from the power supply. When this signal is high, it indicates that all power supply voltages have reached their working levels. CPURST (pin 138) and XRST (pin 134) go high when PWRGOOD is low.
141	B	MFP5	**Multi Function Pin 5** is a bi-directional pin that has two functions: MFP5 is **Real Time Clock Chip Select** (RTCCS*) in external RTC mode. RTCCS* is an active-low output in this mode. RTCCS* is a decode of the I/O address range 070H-071H. MFP5 is **Power Sense** (PS) in internal RTC mode. PS is an active-high input in this mode that indicates the state of the battery. The signal should be connected to the battery back-up circuit.
140	I	MFP4	**Multi Function Pin 4** is an input that has two functions: MFP4 is **Interrupt Request 8** (IRQ08) in external RTC mode. IRQ08 is an interrupt request (with a priority of 8) from an external (type MC164818) RTC. MFP4 is **32KHz Oscillator Input** (OSCI) in internal RTC mode. OSCI should be connected to a square wave source with a frequency of 32.768 KHz. This input signal is used as a clock that times the internal RTC of the 82C235.
126	I	MFP3	**Multi Function Pin 3** is an input pin that is used to reset the CPU (CPURST2*). This signal, an active-low input from the 8042 keyboard controller, is used to generate the CPURST signal to the 80286 microprocessor. The signal is under software control.
130	O	MFP2	**Multi Function Pin 2** is an output pin that is used for **8042 Chip Select** (-8042CS). This signal, an active-low output to the 8042 keyboard controller, is a decode of the I/O address range 060H-06FH.
129	I	MFP1	**Multi Function Pin 1** is an input pin that generates **Gate A20** (GATEA20). This signal, an active-high input from the 8042 keyboard controller, is used by the 82C235 to keep its 0-1MB memory address range under software control when the 80286 processor is in real mode.
125	I	MFP0	**Multi Function Pin 0** is an input pin that acts as **Interrupt Request 1** (IRQ1). This signal, an active-high input from the 8042 keyboard controller, is an interrupt request (with a priority of 1) used by the 82C235's internal interrupt controller.
19 62 102 139		V_{CC}	**+5 Volts**
1 22 59 99 120 142		V_{SS}	**Ground**

5.1.7 Special Pins

The following signals, which are normally outputs, are used as inputs during a power-on reset to select various configuration options. During normal operations (shown in parenthesis), the input pins may have totally different functions than those which they have during power-on reset time. The value of the pull-down or pull-up resistor should be 4.7K ohms.

Pin	Type	Name	Description
96	I	16BITROM* (DACK7*)	**16-Bit ROM** is an active-low input during power-on reset. When this signal is pulled low, 16-bit ROM can be used; if this signal is pulled high, 8-bit ROM can be used. The 82C235 converts 16-bit ROM cycles into two 8-bit ROM cycles in 8-bit ROM mode operation.
91	I	EXRTC* (DACK5*)	**External Real Time Clock** is an active-low input during power-on reset. When this signal is pulled low, the internal RTC is disabled and an external RTC can be used. When this signal is pulled high, the internal RTC is used. The 82C235 changes the functions of the MFP4 and MFP5 pins accordingly.

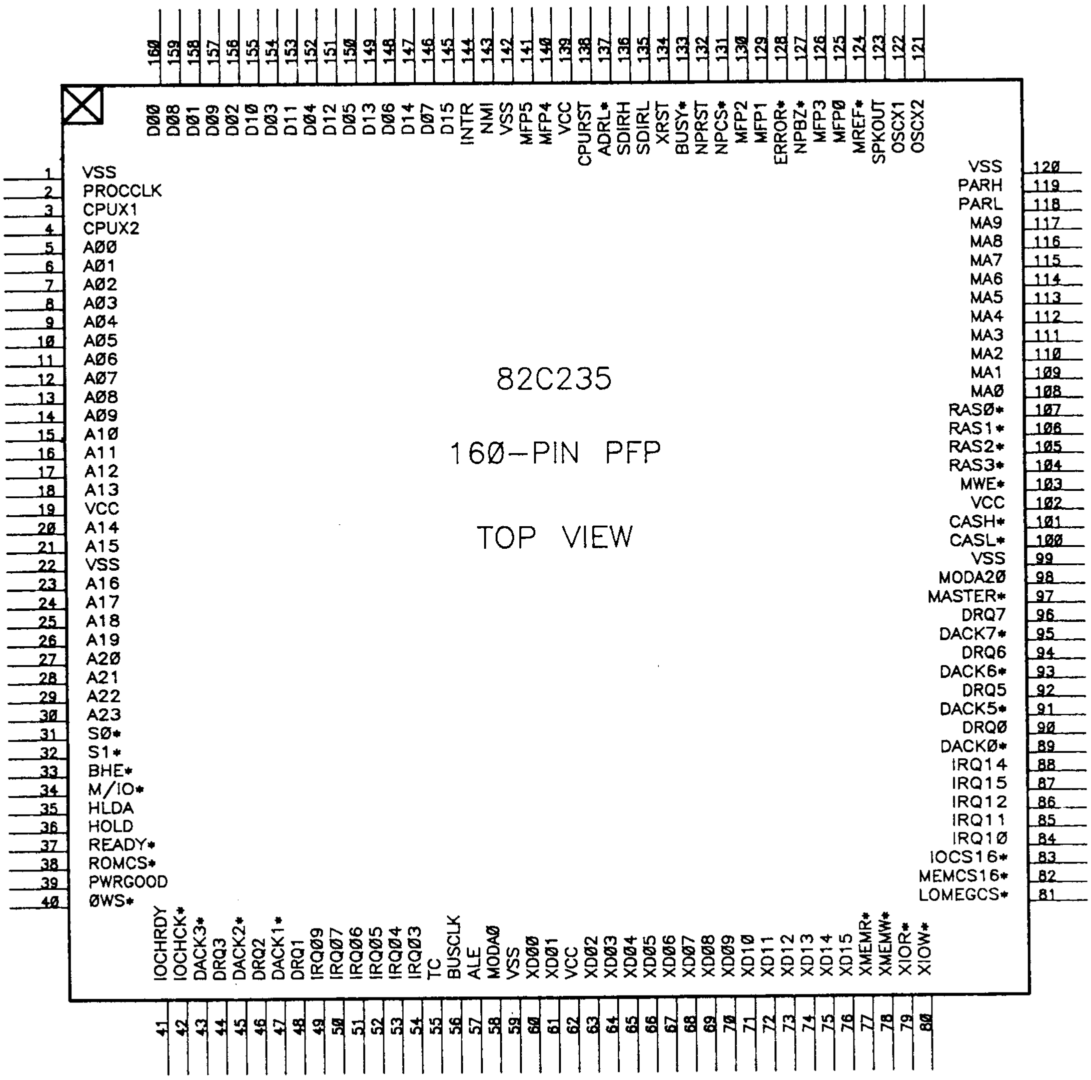


Figure 1.19 160-Pin Plastic Flat Pack

5.2 Alphabetical Listing of Pinouts

Signal Name	Pin	Signal Name	Pin	Signal Name	Pin	Signal Name	Pin
0WS*	40	DACK6*	93	D13	149	RAS3*	104
A00	5	DACK7*	96	D14	147	READY*	37
A01	6	DRQ0	90	D15	145	ROMCS*	38
A02	7	DRQ1	48	LOMEGCS*	81	S0*	31
A03	8	DRQ2	46	M/IO*	34	S1*	32
A04	9	DRQ3	44	MA0	108	SDIRH	136
A05	10	DRQ5	92	MA1	109	SDIRL	135
A06	11	DRQ6	94	MA2	110	SPKOUT	123
A07	12	DRQ7	96	MA3	111	TC	55
A08	13	ERROR*	128	MA4	112	V_{CC}	19
A09	14	HLDA	35	MA5	113	V_{CC}	62
A10	15	HOLD	36	MA6	114	V_{CC}	102
A11	16	INTR	144	MA7	115	V_{CC}	139
A12	17	IOCHCK*	42	MA8	116	V_{SS}	1
A13	18	IOCHRDY	41	MA9	117	V_{SS}	22
A14	20	IOCS16*	83	MASTER*	97	V_{SS}	59
A15	21	IRQ03	54	MEMCS16*	82	V_{SS}	99
A16	23	IRQ04	53	MFP0	125	V_{SS}	120
A17	24	IRQ05	52	MFP1	129	V_{SS}	142
A18	25	IRQ06	51	MFP2	130	XD00	60
A19	26	IRQ07	50	MFP3	126	XD01	61
A20	27	IRQ09	49	MFP4	140	XD02	63
A21	28	IRQ10	84	MFP5	141	XD03	64
A22	29	IRQ11	85	MODA0	58	XD04	65
A23	30	IRQ12	86	MODA20	98	XD05	66
ADRL*	137	IRQ14	88	MREF*	124	XD06	67
ALE	57	IRQ15	87	MWE*	103	XD07	68
BHE*	33	D00	160	NMI	143	XD08	69
BUSCLK	56	D01	158	NPBZ*	127	XD09	70
BUSY*	133	D02	156	NPCS*	131	XD10	71
CASH*	101	D03	154	NPRST	132	XD11	72
CASL*	100	D04	152	OX1	122	XD12	73
CPURST	138	D05	150	OX2	121	XD13	74
CX1	3	D06	148	PARH	119	XD14	75
CX2	4	D07	146	PARL	118	XD15	76
DACK0*	89	D08	159	PROCCLK	2	XIOR*	79
DACK1*	47	D09	157	PWRGOOD	39	XIOW*	80
DACK2*	45	D10	155	RAS0*	107	XMEMR*	77
DACK3*	43	D11	153	RAS1*	106	XMEMW*	78
DACK5*	91	D12	151	RAS2*	105	XRST	134

5.3 Numerical Listing of Pinouts

Pin	Signal Name	Pin	Signal Name	Pin	Signal Name	Pin	Signal Name
1	V_{SS}	41	IOCHRDY	81	LOMEGCS*	121	OX2
2	PROCCLK	42	IOCHCK*	82	MEMCS16*	122	OX1
3	CX1	43	DACK3*	83	IOCS16*	123	SPKOUT
4	CX2	44	DRQ3	84	IRQ10	124	MREF*
5	A00	45	DACK2*	85	IRQ11	125	MFP0
6	A01	46	DRQ2	86	IRQ12	126	MFP3
7	A02	47	DACK1*	87	IRQ15	127	NPBZ*
8	A03	48	DRQ1	88	IRQ14	128	ERROR*
9	A04	49	IRQ09	89	DACK0*	129	MFP1
10	A05	50	IRQ07	90	DRQ0	130	MFP2
11	A06	51	IRQ06	91	DACK5*	131	NPCS*
12	A07	52	IRQ05	92	DRQ5	132	NPRST
13	A08	53	IRQ04	93	DACK6*	133	BUSY*
14	A09	54	IRQ03	94	DRQ6	134	XRST
15	A10	55	TC	96	DACK7*	135	SDIRL
16	A11	56	BUSCLK	96	DRQ7	136	SDIRH
17	A12	57	ALE	97	MASTER*	137	ADRL*
18	A13	58	MODA0	98	MODA20	138	CPURST
19	V_{CC}	59	V_{SS}	99	V_{SS}	139	V_{CC}
20	A14	60	XD00	100	CASL*	140	MFP4
21	A15	61	XD01	101	CASH*	141	MFP5
22	V_{SS}	62	V_{CC}	102	V_{CC}	142	V_{SS}
23	A16	63	XD02	103	MWE*	143	NMI
24	A17	64	XD03	104	RAS3*	144	INTR
25	A18	65	XD04	105	RAS2*	145	D15
26	A19	66	XD05	106	RAS1*	146	D07
27	A20	67	XD06	107	RAS0*	147	D14
28	A21	68	XD07	108	MA0	148	D06
29	A22	69	XD08	109	MA1	149	D13
30	A23	70	XD09	110	MA2	150	D05
31	S0*	71	XD10	111	MA3	151	D12
32	S1*	72	XD11	112	MA4	152	D04
33	BHE*	73	XD12	113	MA5	153	D11
34	M/IO*	74	XD13	114	MA6	154	D03
35	HLDA	75	XD14	115	MA7	155	D10
36	HOLD	76	XD15	116	MA8	156	D02
37	READY*	77	XMEMR*	117	MA9	157	D09
38	ROMCS*	78	XMEMW*	118	PARL	158	D01
39	PWRGOOD	79	XIOR*	119	PARH	159	D08
40	0WS*	80	XIOW*	120	V_{SS}	160	D00

6 Physical Characteristics

6.1 Absolute Maximum Ratings

Symbol	Parameter	Min	Max	Units
V_{cc}	Supply Voltage	3.0	5.5	V
V_I	Input Voltage	-0.5	5.5	V
V_o	Output Voltage	-0.5	5.5	V
T_{op}	Operating Temperature	-25	85	°C
T_{STG}	Storage Temperature	-40	125	°C

NOTE-- Permanent device damage may occur if Absolute Maximum Ratings are exceeded. Functional operation should be restricted to the conditions described under Operating conditions.

6.2 Operating Conditions

Symbol	Parameter	Min	Max	Units
V_{cc}	Supply Voltage	4.75	5.25	V
T_A	Ambient Temperature	0	70	°C

6.3 Capacitive Characteristics

(T_A = 25°C, V_{CC} = 5V ± 5%)

Symbol	Parameter	Min	Max	Units	Test Conditions
C_{IN}	Input Capacitance	10		pF	f_C = 1 MHz
$C_{I/O}$	I/O Capacitance		20	pF	Unmeasured pins
C_{OUT}	Output Capacitance		20	pF	returned to GND

7 DC Characteristics

(T_A = 25°C, V_{CC} = 5V ± 5%)

Symbol	Parameter	Min	Max	Units	Test Conditions
V_{IL}	Input Low Voltage	-0.5	0.8	V	
V_{IH}	Input High Voltage	2.0			V_{CC} +0.5 (units=volts)
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 4.0 mA (note 1)
V_{OH}	Output High Voltage	2.4		V	I_{OH} = -4.0 mA
V_{CL}	Clock Output Low		0.4	V	I_{OL} = 4.0 mA
V_{CH}	Clock Output High	3.9		V	I_{OH} = -4.0 mA
I_{IL}	Input Leakage Current	-10	20	uA	V_I = Vcc to 0 V
I_{OL}	Output Leakage Current	-10	20	uA	V_O = Vcc to 0.45 V
I_{CC}	Vcc Supply Current		50	mA	CPUX1 = 24 MHz
I_{CCSB}	V_{CC} Standby Supply Current		50	uA	

NOTE 1--

I_{OL} = 2mA for NPCS*, NPRST, BUSY*, HOLD, XRST, CPURST, SPKOUT, SDIRH, SDIRL, TC, D0:15, MFP5, MFP2, NMI, INTR, MODA20, BHE*, READY*, ROMCS*

I_{OL} = 8mA for XMEMR*, XMEMW*, XIOR*, XIOW*, MREF*, MWE*, BALE, ADRL*

8 82C235 AC Characteristics
(T_a = 0-70 C°, V_{cc} = 5V ± 5%)

8.1 Reset Timings

Symbol	Parameter	Min	Max	Units
t1	PWRGOOD Active from V_{CC} High (V_{CC}=4.5V)	100		us
t2	CPURST, XRST Active from V_{CC} High		1	us
t3	CPURST, XRST Inactive from PWRGOOD High	64	66	t11
t4	PWRGOOD Rise Time (1.0V to 3.6V)		500	us

8.2 Clock Timings

Symbol	Parameter	Min	Max	Units
t11	CPUX1 Cycle Time	40		ns
t12	CPUX1 Rise Time (Oscillator Input)		8	ns
t13	CPUX1 Fall Time (Oscillator Input)		8	ns
t14	CPUX1 High Time (Oscillator Input)	13		ns
t15	CPUX1 Low Time (Oscillator Input)	11		ns
F_{cxl}	CPUX1 Frequency	DC	25	MHz
F_{oxl}	OSCX1 Frequency	DC	15	MHz
t21	CPUX1 to PROCCLK Delay		25	ns
t22	PROCCLK Rise Time (1.0V to 3.6V)		8	ns
t23	PROCCLK Fall Time (3.6V to 1.0V)		8	ns
t24	PROCCLK High Time (3.6V)	13		ns
t25	PROCCLK Low Time (1.0V)	11		ns
t26	PROCCLK Cycle Time	40		ns
t30	PROCCLK to BUSCLK Delay		25	ns

8.3 DRAM-Control Timings

Symbol	Parameter	Min	Max	Units
t41	Row Address Delay from CPU address	0	40	ns
t42	Column Address Delay from $PROCCLK_H$	3	40	ns
t43	RAS* Active Delay from $PROCCLK_L$	3	11^+	ns
t44	RAS* Inactive Delay from $PROCCLK_H$	3	20^+	ns
t45	CAS* Active Delay from PROCCLK	3	26^+	ns
t46	CAS* Inactive Delay from $PROCCLK_L$	6	33	ns
t48	MWE* Active Delay from $PROCCLK_L$	3	32	ns
t51	Read Data Setup Time to $PROCCLK_L$	5		ns
t52	Read Data Hold Time from $PROCCLK_L$	6		ns
t53	D Bus to Parity Delay	0	31^+	ns
t54	Write Parity Hold Time from $PROCCLK_L$	0	40	ns

8.4 8-Bit Bus and ROM Cycle Timings

Symbol	Parameter	Min	Max	Units
t61	ADRL Active Delay from PROCCLK$_L$	3	20	ns
t62	ADRL Inactive Delay from PROCCLK$_L$	3	20	ns
t63	ALE Active Delay from BUSCLK$_L$	-10	10	ns
t64	ALE Inactive Delay from BUSCLK$_H$	-10	12	ns
t65	Command Active Delay from BUSCLK	-10	16	ns
t66	Command Inactive Delay from BUSCLK$_H$	-10	15	ns
t67	SDIR Active Delay from PAROCCLK$_L$		16	ns
t71	XD Bus Setup Time from PROCCLK$_L$	35		ns
t72	XD Bus Hold Time from PROCCLK$_L$	5	46	ns
t73	IOCHRDY Setup Time to BUSCLK$_H$	11		ns
t75	ROMCS* Active Delay from BUSCLK$_L$	0	40	ns
t76	ROMCS* Inactive Delay from BUSCLK$_H$	0	20	ns
t77	MREF* Active Delay from HLDA$_H$	0	40	ns
t78	MREF* Inactive Delay from PROCCLK$_H$	0	40	ns
t79	Refresh Address Valid from OSC$_H$	0	50	ns
t80	HOLD Inactive Delay from PROCCLK$_H$	0	40	ns
t81	XMEMR* Active Delay from OSC$_H$	0	35	ns
t82	XMEMR* Inactive Delay from OSC$_H$	0	35	ns
t83	RAS* Active Delay from OSC$_H$	0	35	ns
t84	RAS* Inactive Delay from OSC$_H$	0	35	ns
t85	XD Bus to LD Bus delay	0	30	ns
t86	MEMCS16 Setup Time to BUSCLK$_H$	10		ns
t87	MEMCS16 Hold Time to BUSCLK$_H$	17		ns
t88	IOCS16 Setup Time to COMMAND$_L$	10		ns
t89	IOCS16 Hold Time to BUSCLK$_H$	40		ns

8.5 DMA Timings

Symbol	Parameter	Min	Max	Units
t90	DREQ Setup Time to PROCCLK$_L$	20		ns
t91	HLDA Setup Time to PROCCLK$_L$	20		ns
t92	Address Valid Delay from PROCCLK$_L$	0	35	ns
t93	DACK Active Delay from PROCCLK$_L$	0	40	ns
t94	DACK Inactive Delay from PROCCLK$_L$	0	40	ns
t95	Command Valid from PROCCLK$_L$	0	50	ns
t96	Command Active Delay from PROCCLK$_L$	0	50	ns
t97	Command Inactive Delay from PROCCLK$_L$	0	50	ns
t98	Command Invalid from PROCCLK$_L$	0	40	ns
t99	TC Active Delay from PROCCLK$_L$	0	50	ns
t100	TC Inactive Deplay from PROCCLK$_L$	0	50	ns

8.6 CPU Interface Timings

Symbol	Parameter	Min	Max	Units
t111	S0*, S1* Setup Time to PROCCLK $_L$	22		ns
t112	S0*, S1* Hold Time from PROCCLK $_L$	3		ns
t113	A Bus Setup Time to PROCCLK $_L$	0		ns
t114	A Bus Hold Time from PROCCLK $_L$	0		ns
t115	BHE* Setup Time to PROCCLK $_L$	0		ns
t116	BHE* Hold Time from PROCCLK $_L$	0		ns
t117	D Bus Active Time to PROCCLK $_L$	5		ns
t118	D Bus Inactive Time from PROCCLK $_L$	6		ns
t119	Write Data Delay from PROCCLK $_L$	0	30	ns
t120	Write Data Hold Time from PROCCLK $_L$	0		ns
t121	READY Active Delay from PROCCLK $_H$	0	36	ns
t122	READY Inactive Delay from PROCCLK $_H$	0	40	ns
t123	0WS Setup Time to BUSCLK $_L$	9		ns
t124	0WS Hold Time from BUSCLK $_L$	4		ns
t127	XD Bus Valid from BUSCLK $_H$		40	ns
t128	XD Bus Invalid from COMMAND $_H$	5	25	ns
t129	SA0 Valid from BUSCLK $_L$	-10	40	ns
t132	MEMCS16 Active Delay from PROCCLK $_L$	0	40	ns
t133	ADRL, ALE Active Delay from HLDA $_H$	0	40	ns
t134	ADRL, ALE Inactive Delay from HLDA $_L$	0	40	ns
t135	READY Active Delay from PROCCLK $_L$	0	18	ns
t136	READY Inactive Delay from PROCCLK $_L$	0	40	ns

8.7 Miscellaneous Timings

Symbol	Parameter	Min	Max	Units
t151	CPURST Active Delay from PROCCLK $_L$	5	22	ns
t152	CPURST Inactive Delay from PROCCLK $_L$	5	22	ns
t153	8042CS Active Delay from ADRL $_L$	0	40	ns
t163	NPBUSY Active Pulse Width	26		ns
t164	ERROR Hold Time to NPBUSY	30		ns
t165	ERROR Setup Time to NPBUSY	10		ns
t167	BUSY Active Delay from NPBUSY	0	23	ns
t168	BUSY Inactive Delay from NPBUSY	0	23	ns
t169	BUSY Inactive Delay from XIOW	-10	20	ns
t170	NPRST Active Delay from XIOW	-10	27	ns
t171	NPRST Inactive Delay from XIOW	-10	27	ns
t173	IOCHRDY Setup Time to OSC/BUSCLK	20		ns
t174	IOCHRDY Hold Time to OSC/BUSCLK	15		ns

NOTES--

$^+$ = Timing for 12 MHz, 0 ws, 40%/60% PROCCLK, and using DRAM with RAS PRECHARGE TIME (t_{RP}) is 60ns max.

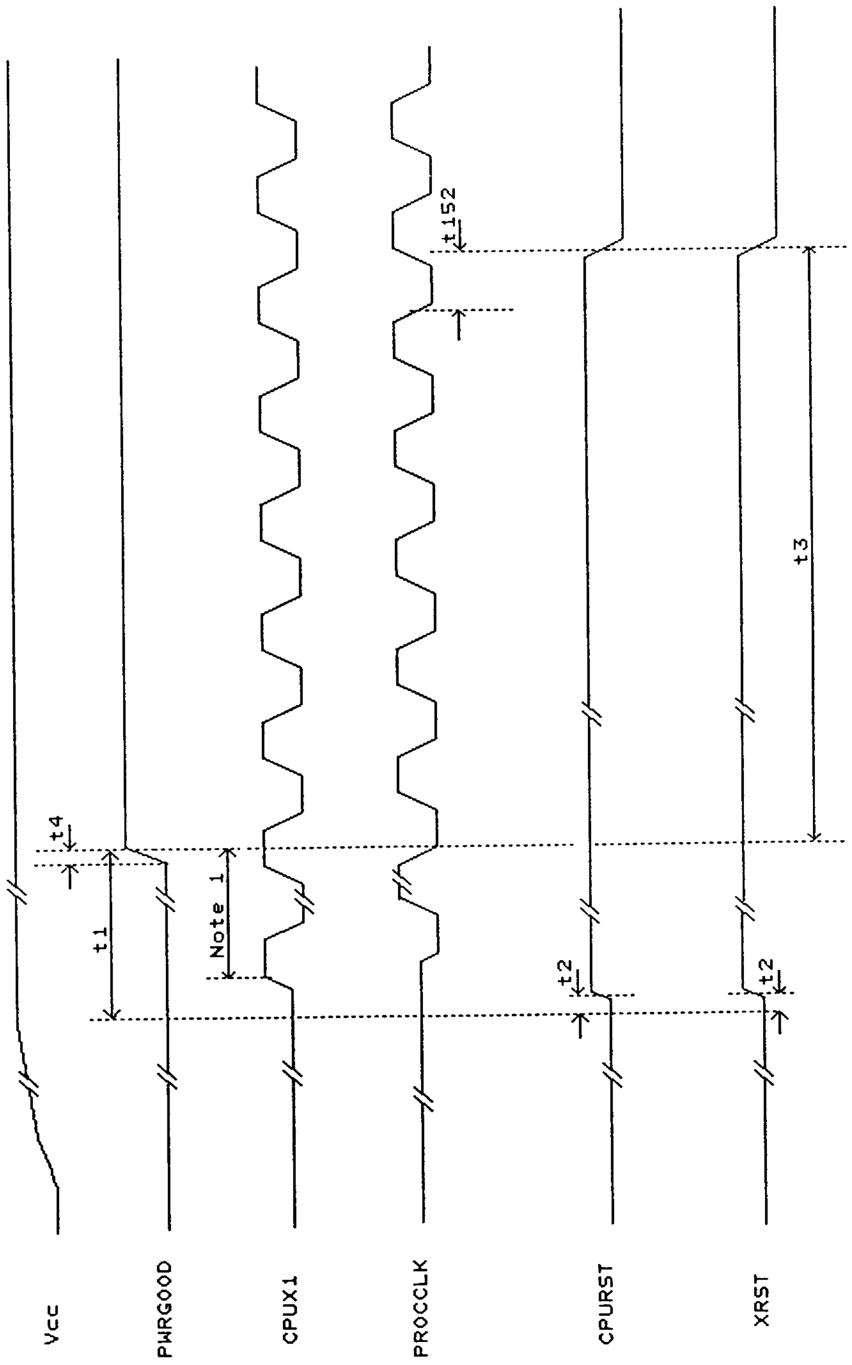
$_L$ = High to Low edge

$_H$ = Low to High edge

9 Timing Diagrams

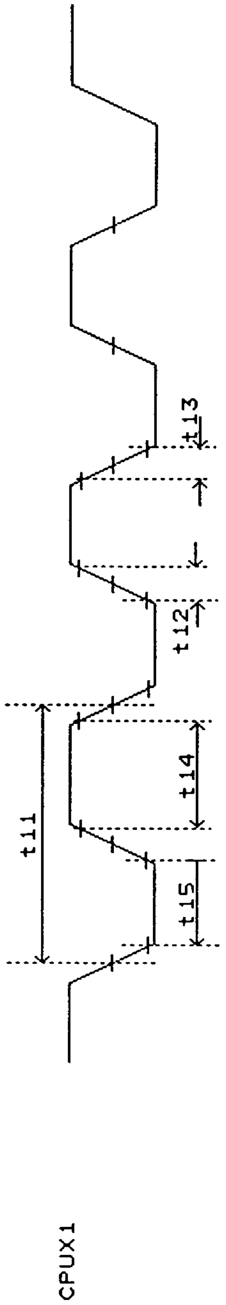
This section contains 82C235 timing diagrams.

9.1 Hardware Reset Timing

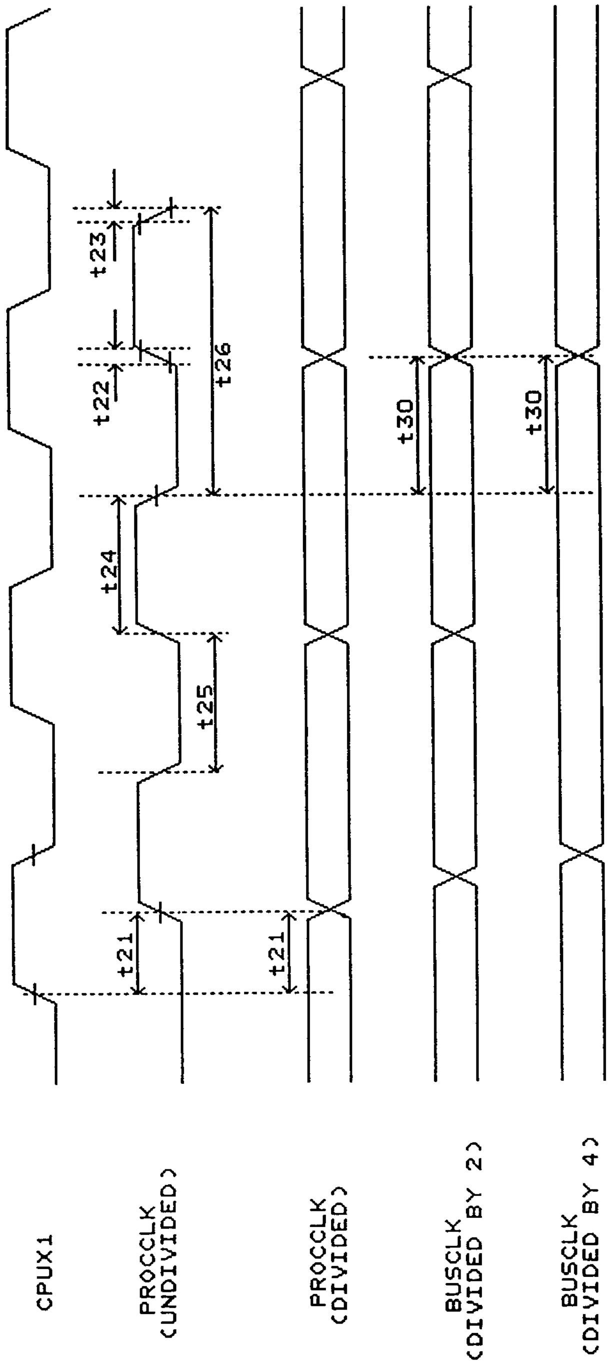


Note 1: CPUX1 MUST COMPLETE A MINIMUM OF 8 CYCLES PRIOR TO PWRGOOD

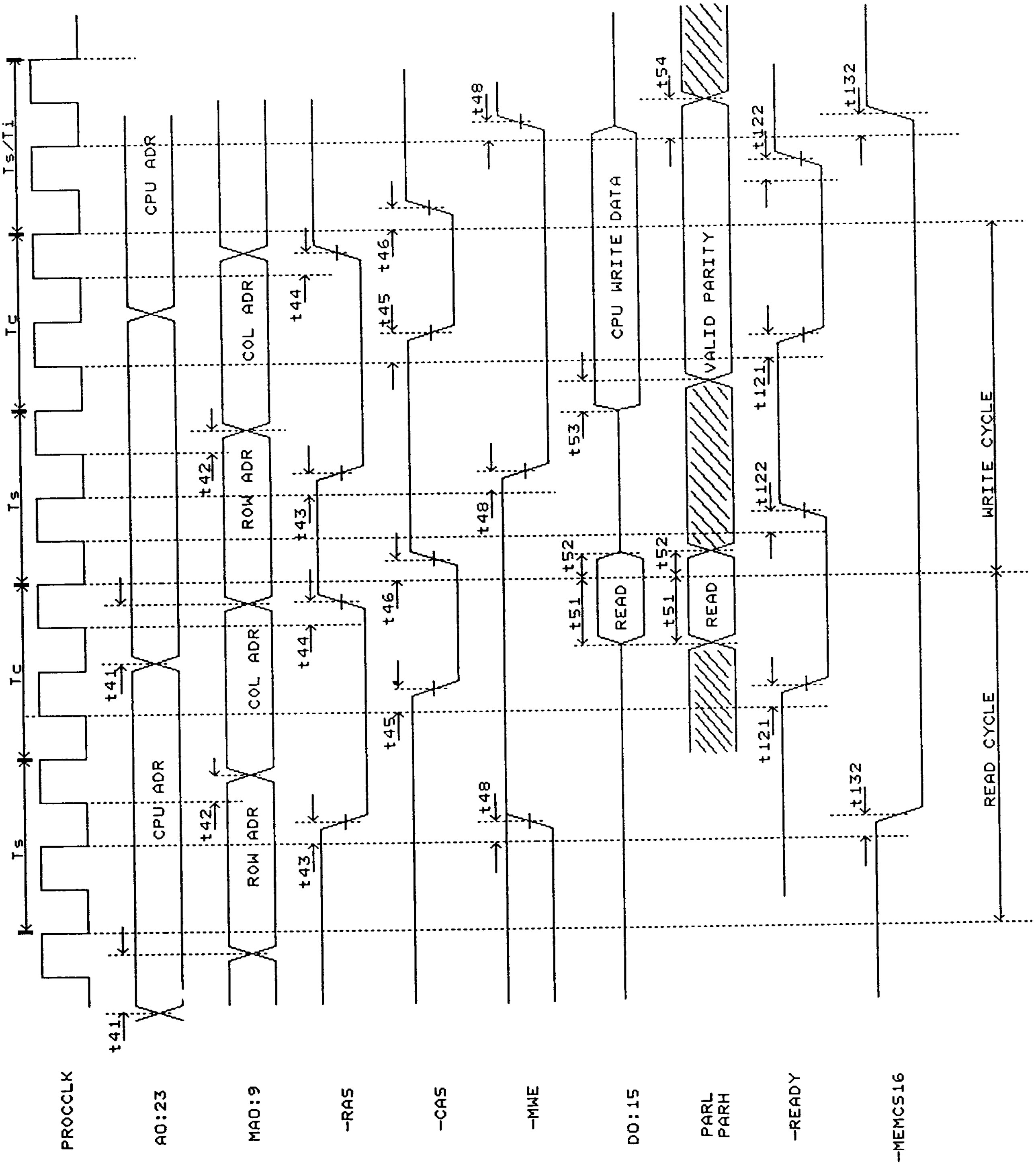
9.2 Clock Input Timing



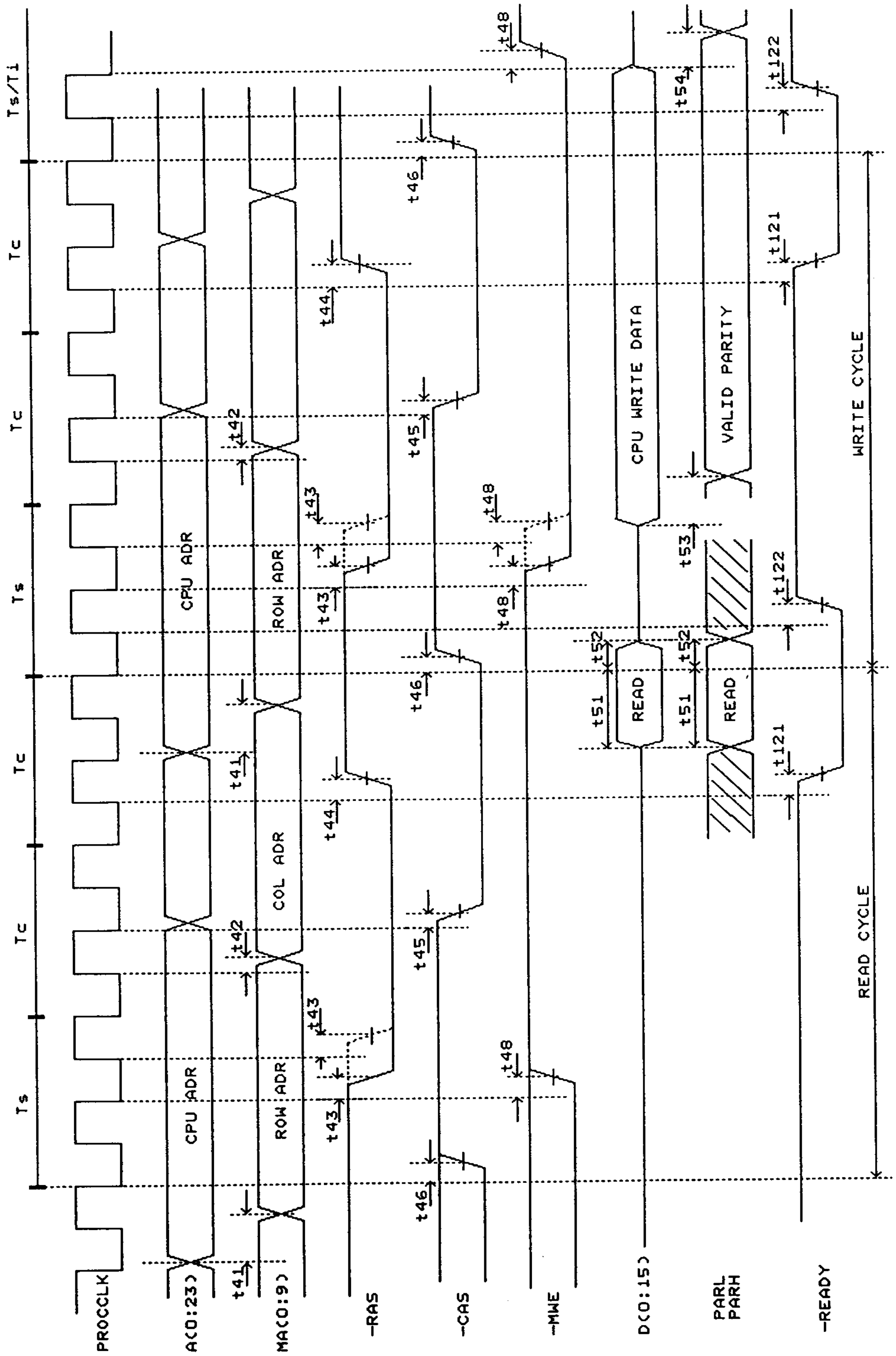
9.3 Clock Output Timing



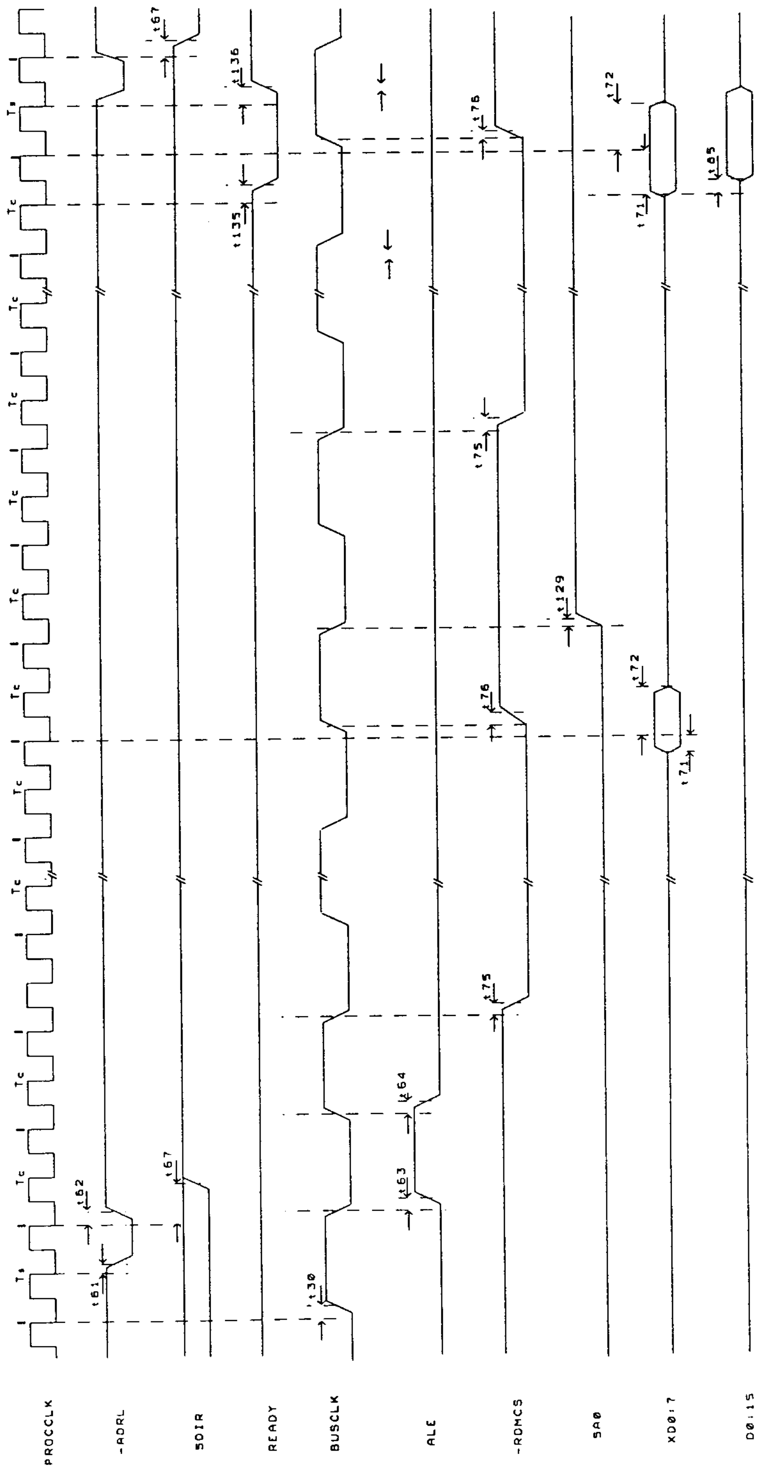
9.4 Zero Wait-state DRAM Timing



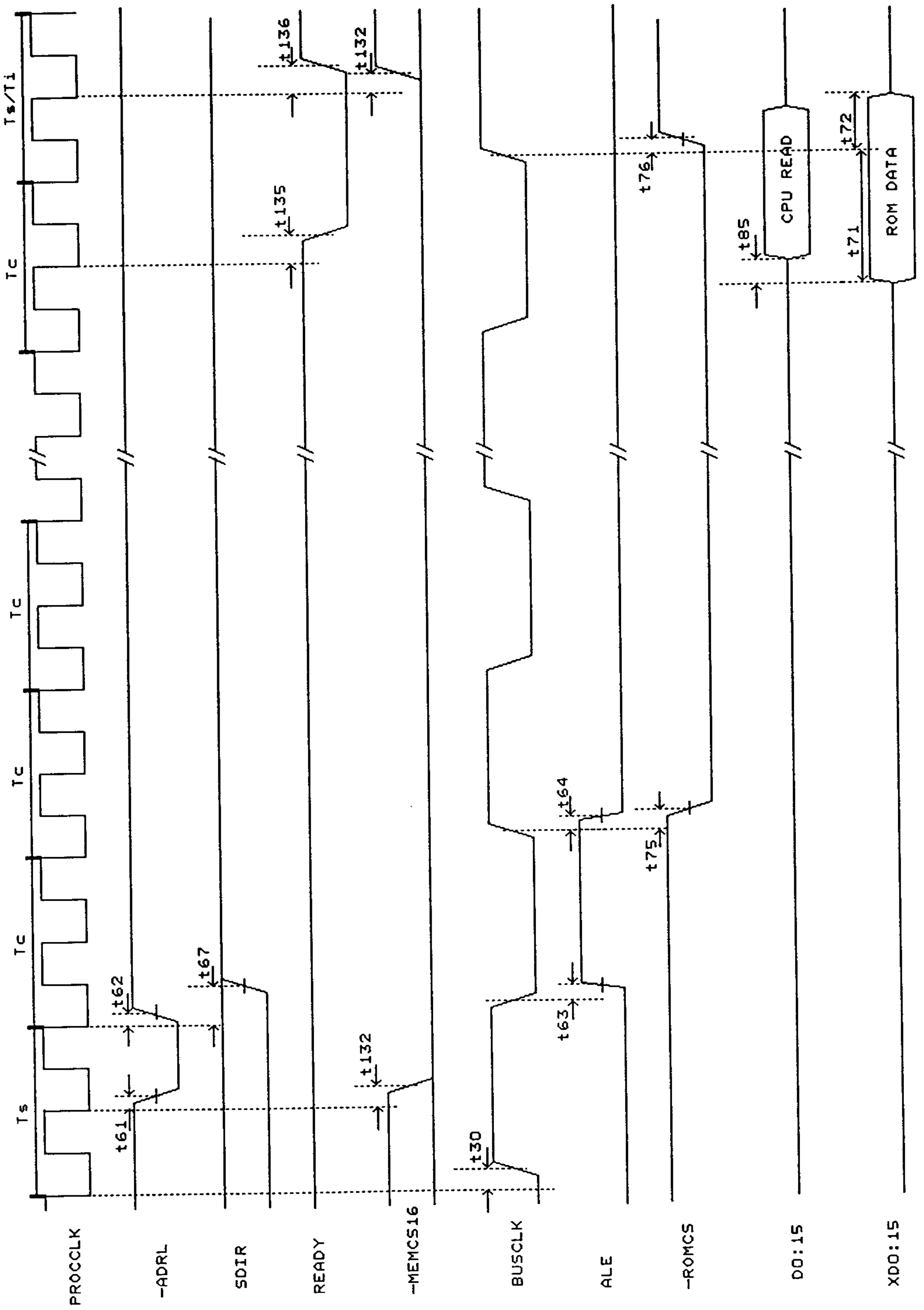
9.5 One Wait-state DRAM Timing



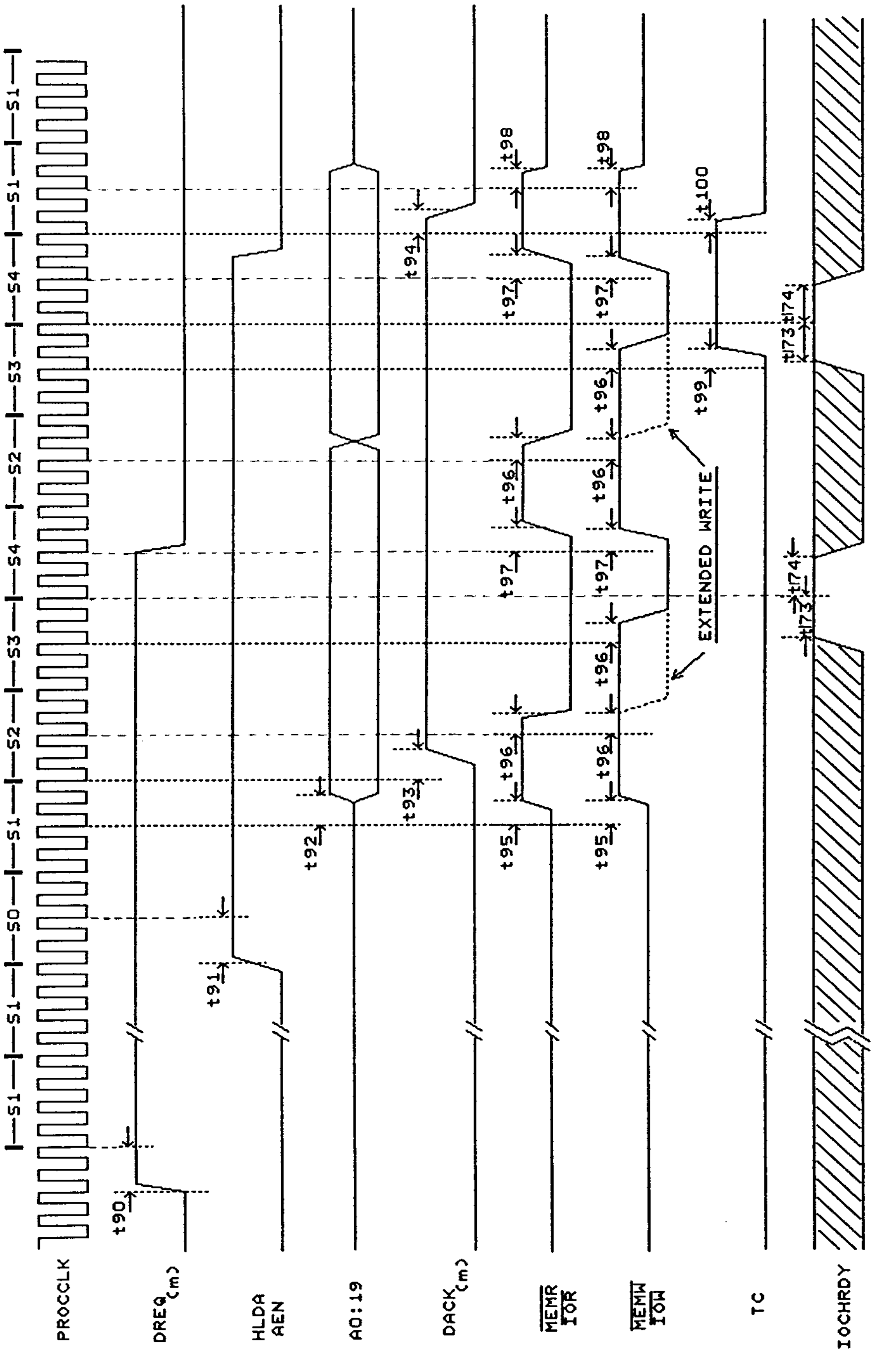
9.6 8-Bit ROM Cycle Timing



9.7 16-Bit ROM Cycle Timing

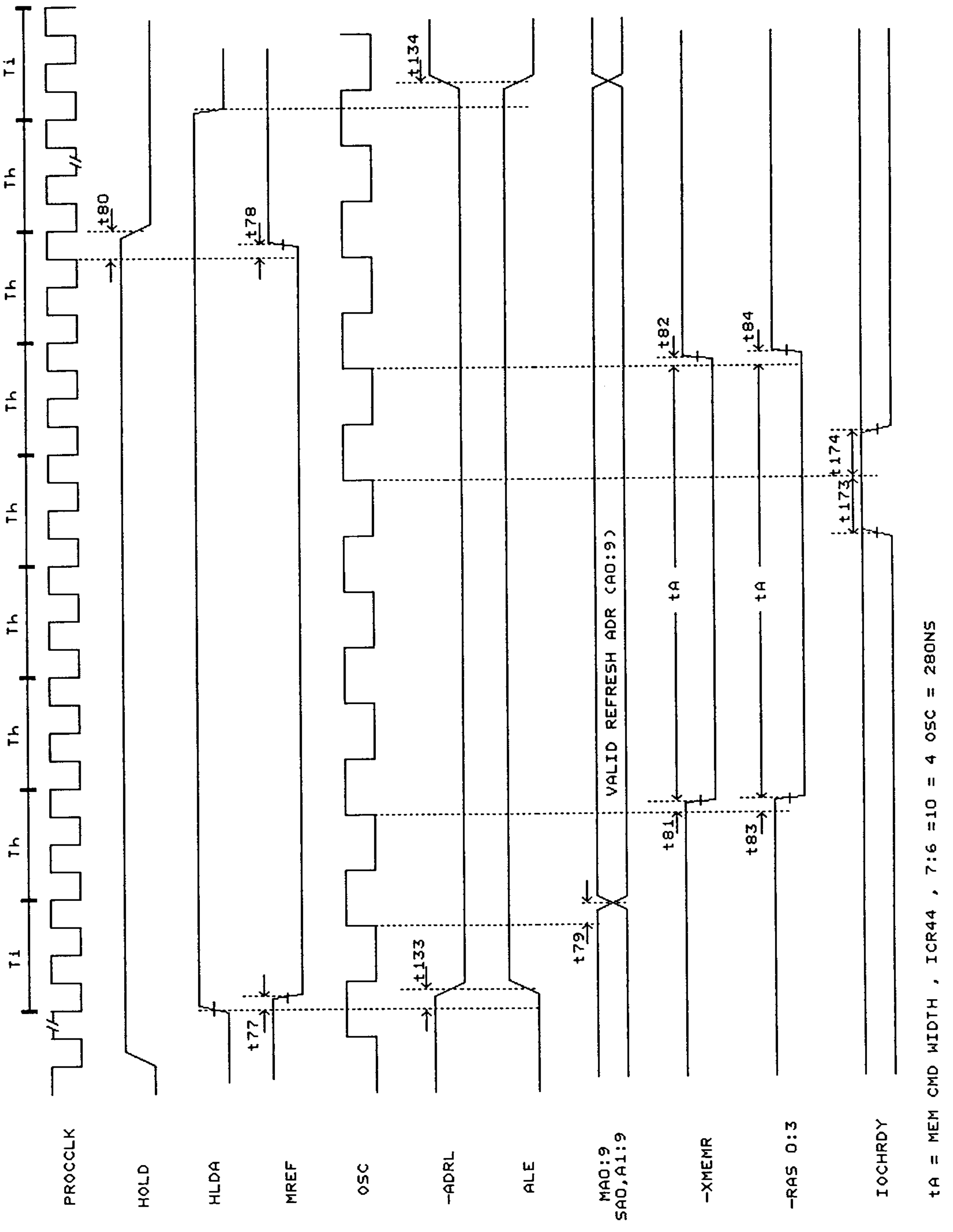


9.8 DMA Read/Write Cycle Timing



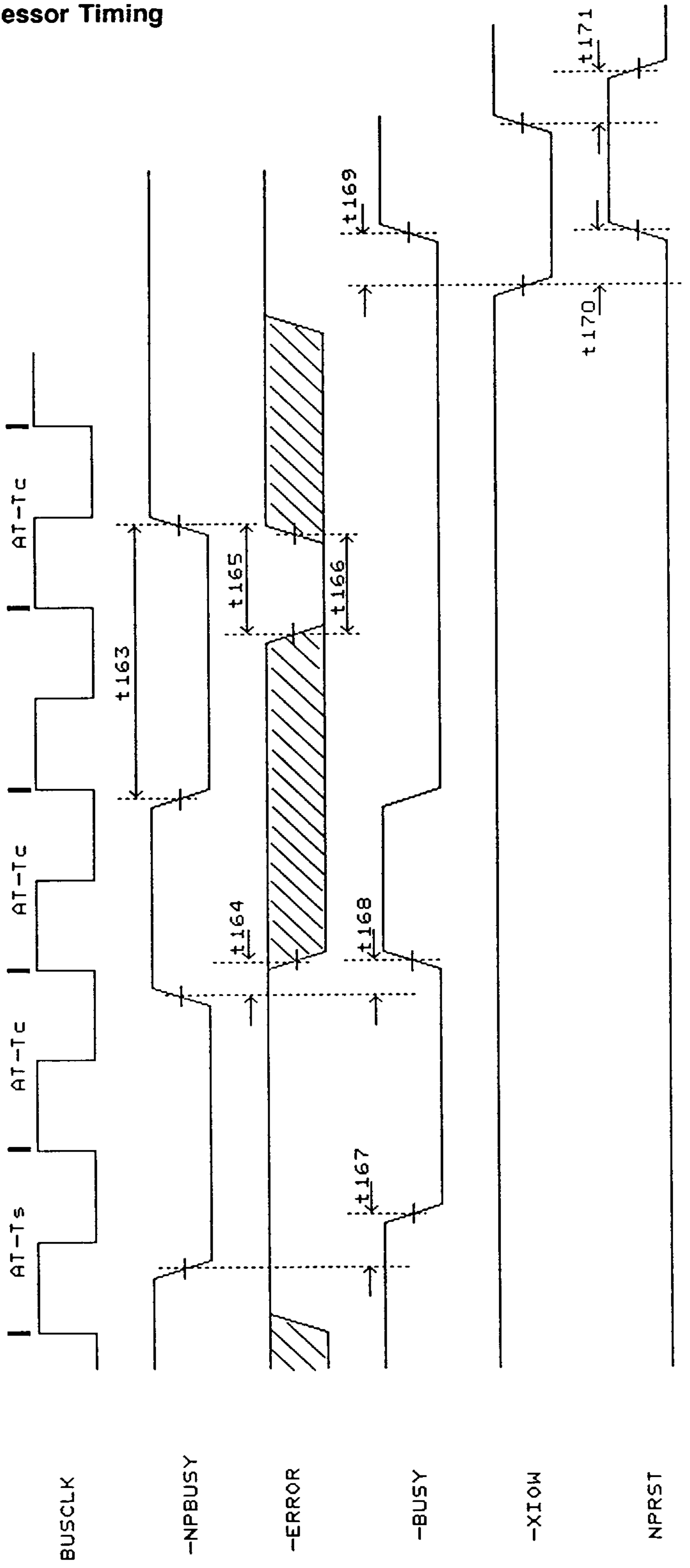
DMA W.S , ICR 41-1 = 1 = 2 CLOCK READ COMMAND PULSE
DMACLK = PROCCLK / 4

9.9 Memory Refresh Cycle Timing

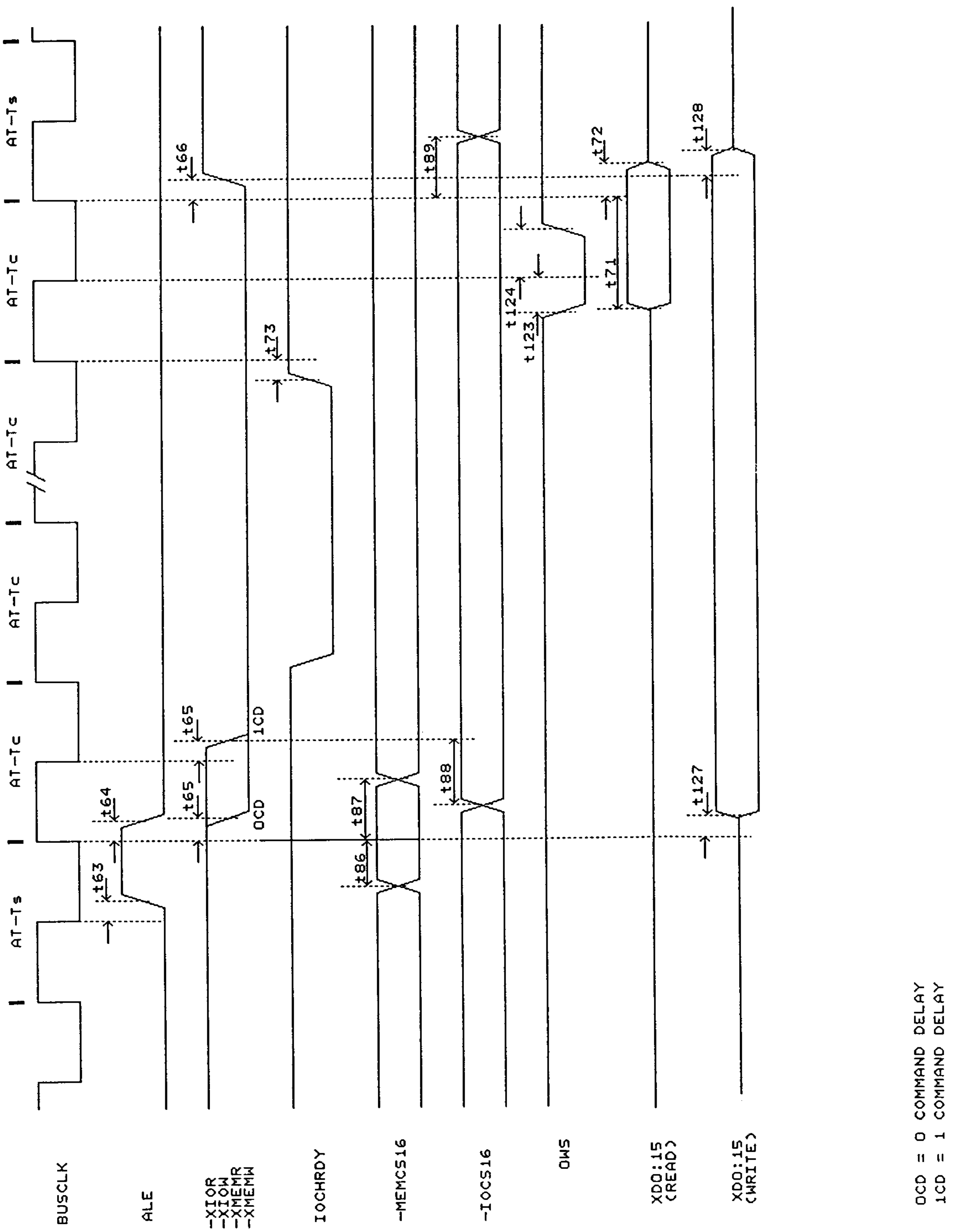


tA = MEM CMD WIDTH , ICR44 , 7:6 =10 = 4 OSC = 280NS

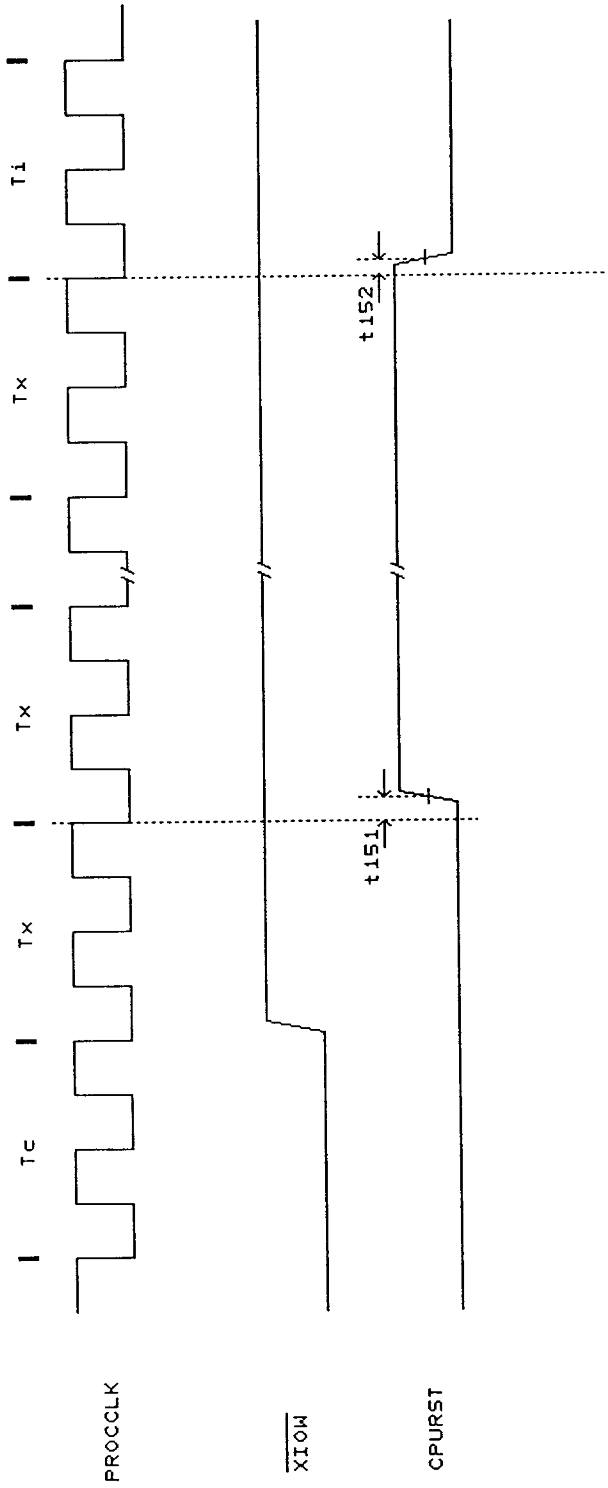
9.10 Numeric Processor Timing



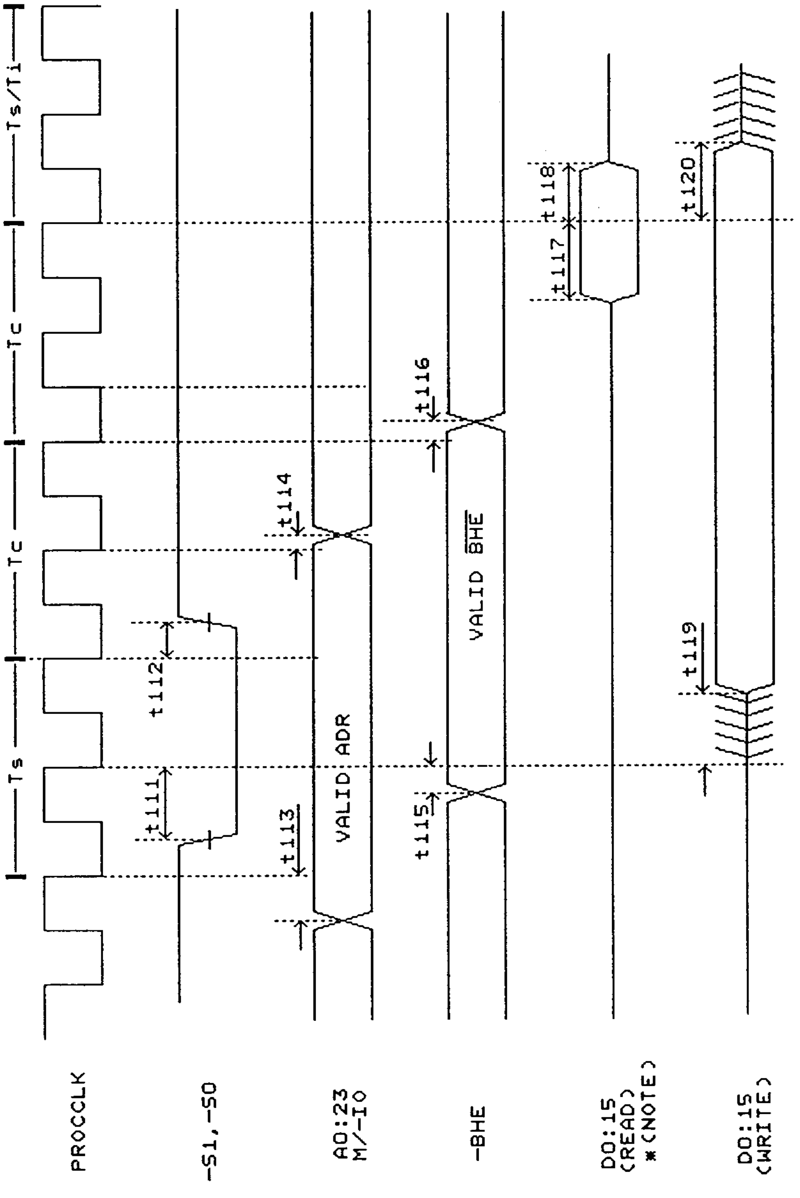
9.11 I/O Channel Timing



9.12 Software Reset Timing

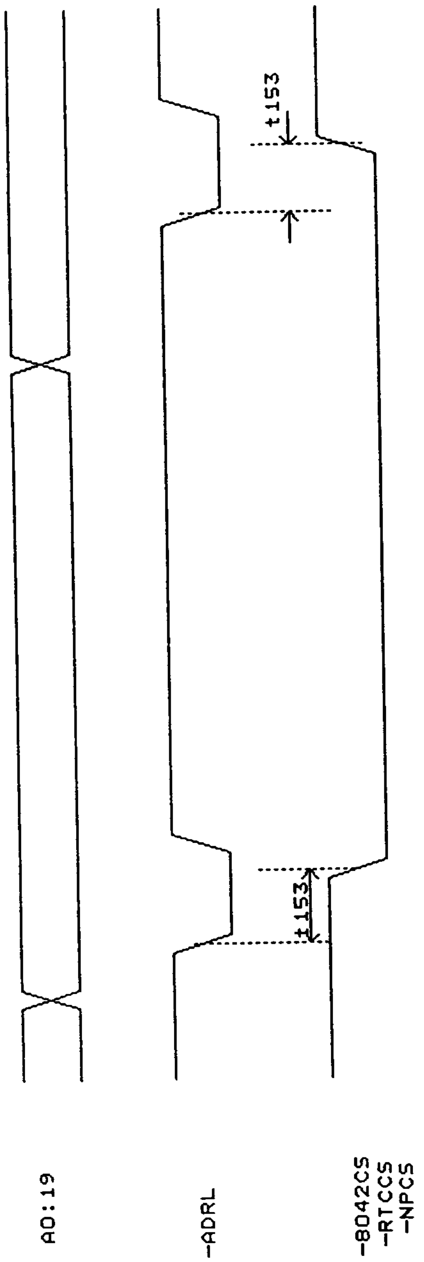


9.13 CPU Interface Timing



* NOTE: THESE D0:15 IS OUTPUT FROM 82C235

9.14 Peripheral Control Timing



10 160-Pin Plastic Flat Package (PFP) Physical Dimensions

DIMENSIONS: mm (in)

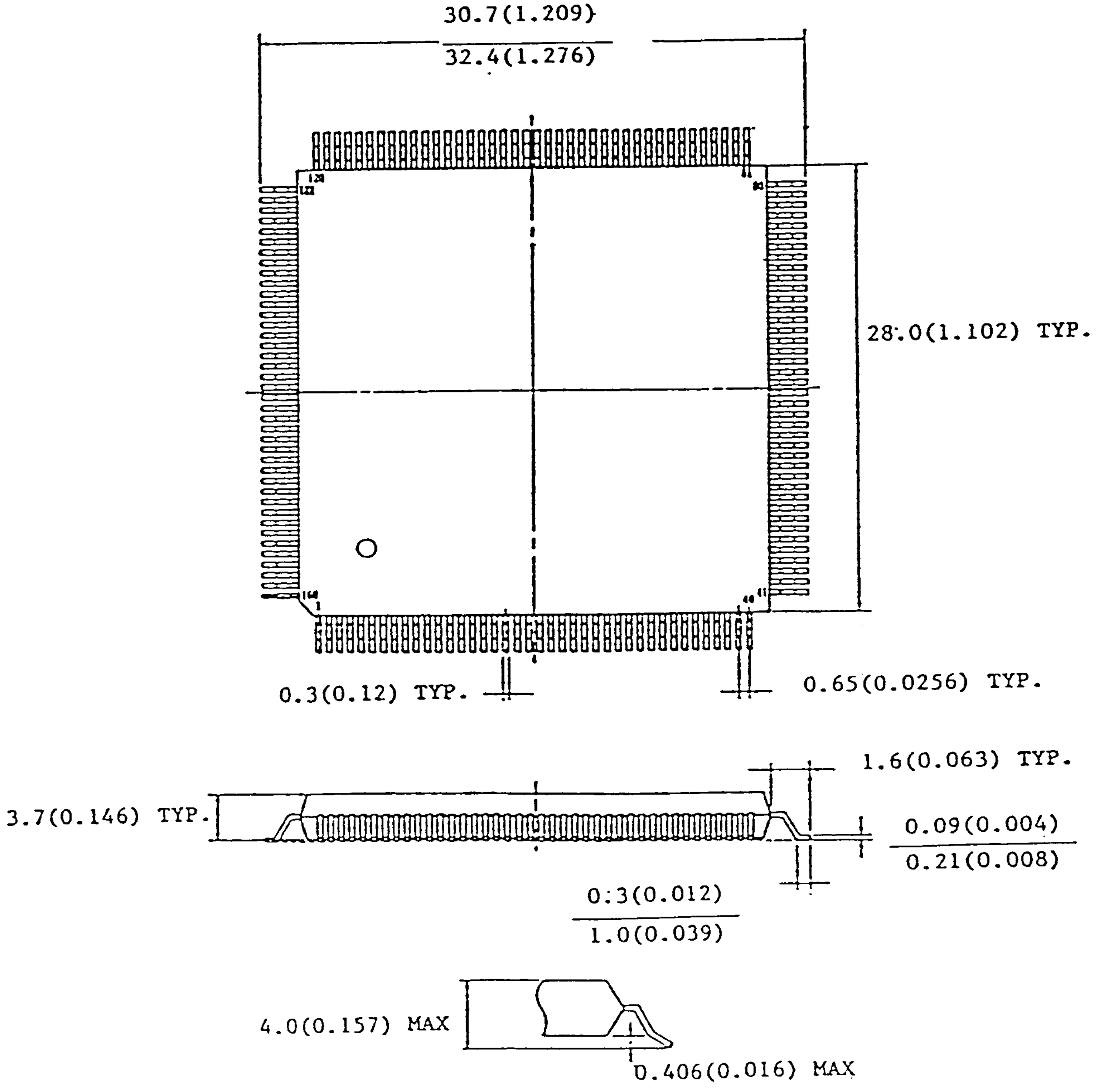


Figure 1.20 160-Pin PFP Physical Dimensions

11 Ordering Information

Order Number Item

F82C235 Plastic Flat Pack; 160 pins

DK82C235 82C235 Designer Kit. This kit allows a designer to evaluate and test the CHIPS 82C235 in a working environment. The kit, which is fully documented, has the following features:

- an 82C235-based motherboard
- a mounted 80286 microprocessor
- socket for optional 80287 numeric processor
- design-support TTL circuitry
- CHIPS BIOS
- 5-slot AT I/O channel (for video, communication, floppy disk, and hard disk controllers)
- test points on edge connectors for easy access with test equipment
- user work area on the motherboard for prototyping and further circuit development

The designer must supply the following:

- power supply
- keyboard
- video card and terminal
- disk controller and disk drive(s)

For details, contact your local Chips and Technologies sales engineer or distributor.