2.3 INTRODUCTION TO MULTIPROGRAMMING

8086 and 8088 can be configured in two modes of operation, the minimum mode and the maximum mode. The minimum models used for a small system with a single processor, a system in which the 8086/8088 generates all the necessary bus control signals directly (thereby minimizing the required bus control logic). The maximum mode is for medium-size to large systems, which often include two or more processors

MULTIPROCESSOR SYSTEMS

Multiprocessor Systems refer to the use of multiple processors that execute instructions simultaneously and communicate using mailboxes and semaphores

Maximum mode of 8086 is designed to implement 3 basic multiprocessor configurations:

1. Coprocessor(8087)
2. Closely coupled (dedicated I/O processor:8089)
3. Loosely coupled (Multiuse)

Coprocessors and closely coupled configurations are similar - both the CPU and the external processor share:

- Memory
- I/O system
- Bus & bus control logic Clock generator

COPROCESSOR CONFIGURATION

WAIT instruction allows the processor to synchronize itself with external hardware, eg., waiting for 8087 math co-processor. When the CPU executes WAIT state, TEST input is asserted (low), the waiting state is completed and execution will resume.

ESC instruction:

ESC oppose, operand,

Oppose: Immediate value recognizable to a coprocessor as an instruction oppose

Operand: Name of a register or a memory address (in any mode)

When the CPU executes the ESC instruction, the processor accesses the memory operand by placing the address on the address bus. If a coprocessor is configured to share the system bus, it will recognize the ESC instruction and therefore will get the code and the operand

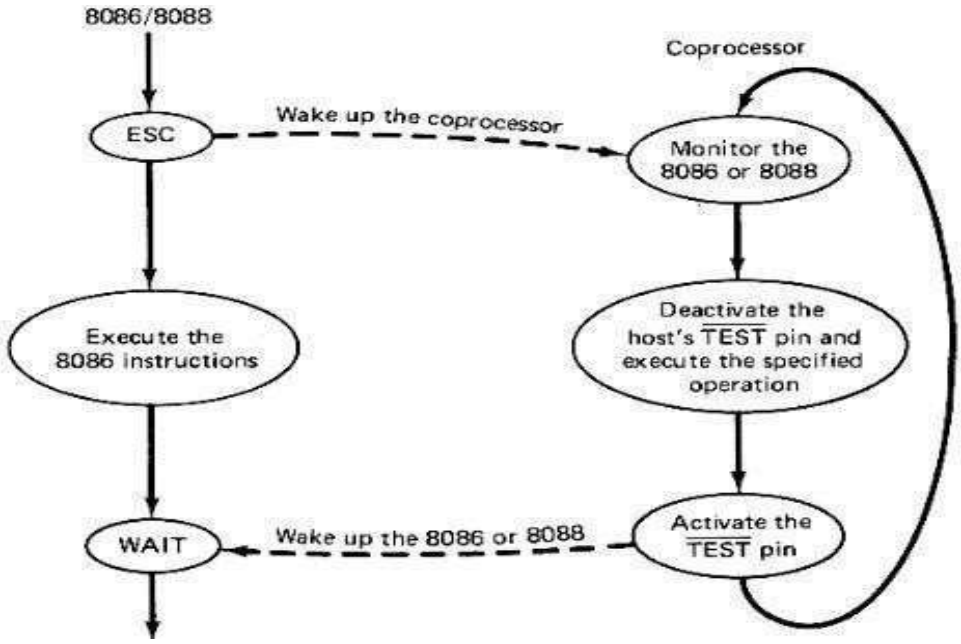


Figure 2.3.1 Synchronization between the 8086 and its coprocessor

[Source: "Microcomputer Systems: The 8086 / 8088 Family - Architecture, Programming and Design" by Yu-Cheng Liu, Glenn A.Gibson]

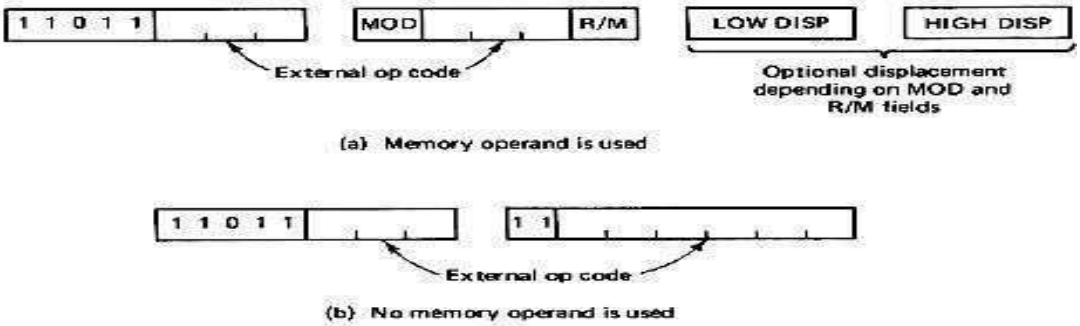


Figure 2.3.2 Machine code formats for the ESC instruction

[Source: "Microcomputer Systems: The 8086 / 8088 Family - Architecture, Programming and Design" by Yu-Cheng Liu, Glenn A.Gibson]

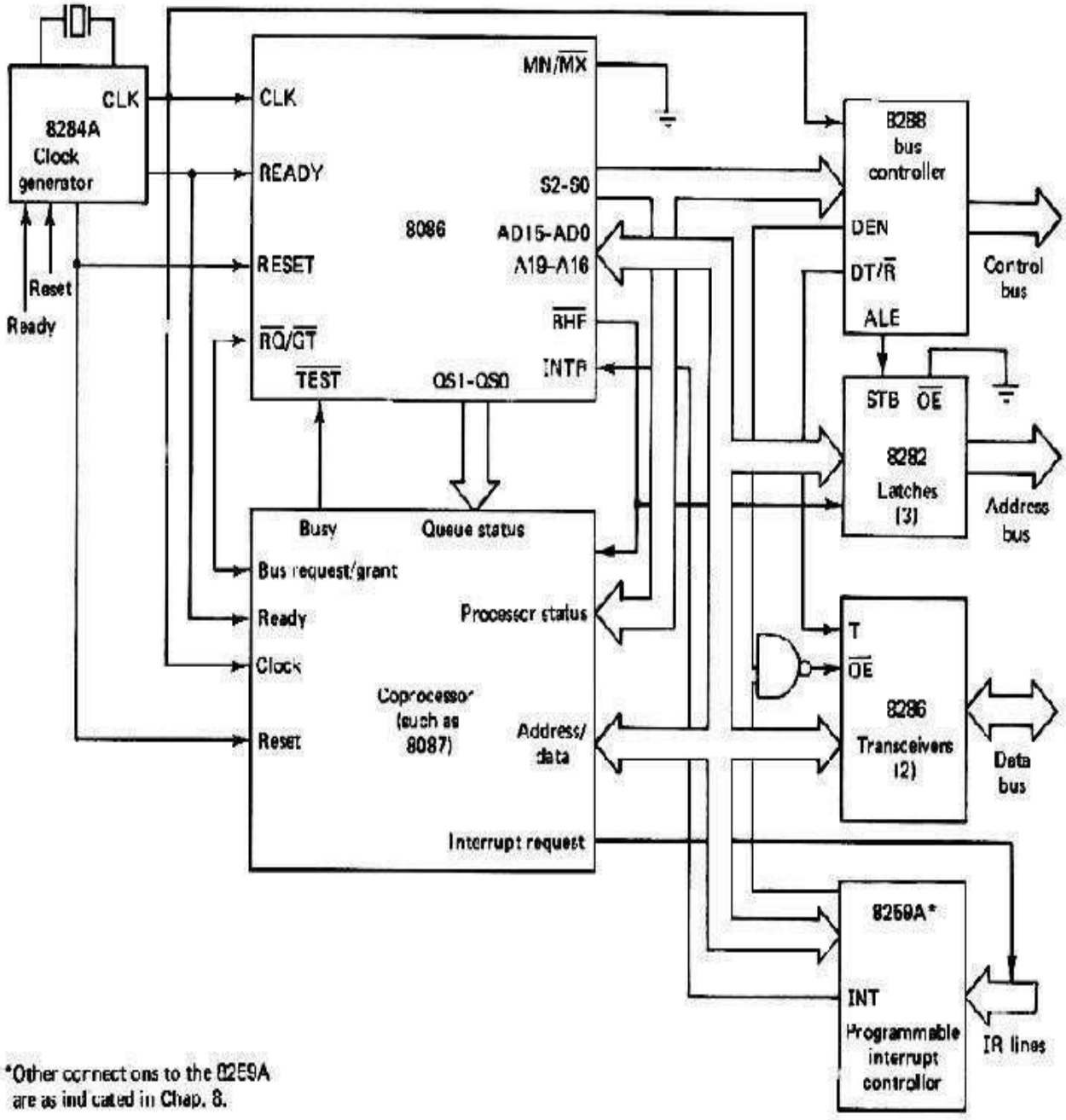


Figure 2.3.3 Coprocessor Configuration

[Source: "Microcomputer Systems: The 8086 / 8088 Family - Architecture, Programming and Design" by Yu-Cheng Liu, Glenn A.Gibson]

CLOSELY COUPLED CONFIGURATION

Closely Coupled processor may take control of the bus independently - 8089 shares CPU's clock and bus control logic.

- communication with host CPU is by way of shared memory
- host sets up a message (command) in memory
- independent processor interrupts host on completion
- Two 8086's cannot be closely coupled
- Coprocessor cannot take control of the bus, it does everything through the CPU
  -

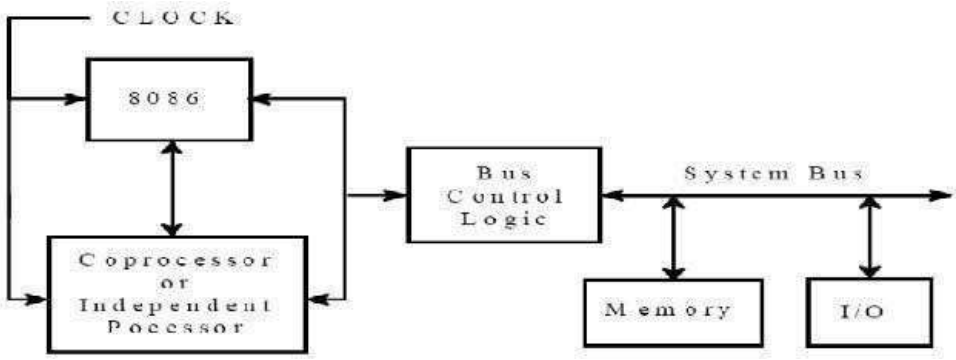


Figure 2.3.4 closely coupled configuration

[Source: "Microcomputer Systems: The 8086 / 8088 Family - Architecture, Programming and Design" by Yu-Cheng Liu, Glenn A.Gibson]

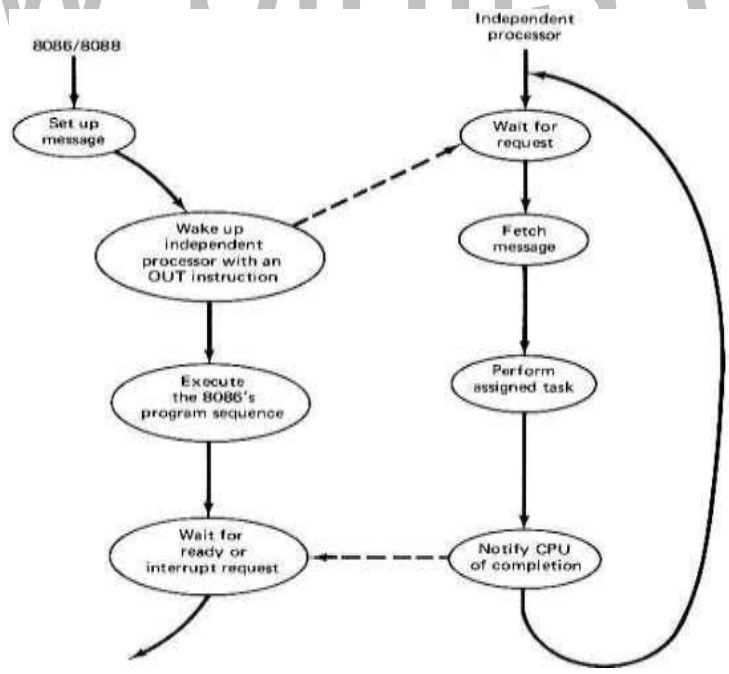


Figure 2.3.5 Inter processor communication through shared memory

[Source: "Microcomputer Systems: The 8086 / 8088 Family - Architecture, Programming and Design" by Yu-Cheng Liu, Glenn A.Gibson]

LOOSELY COUPLED CONFIGURATIONS:

A loosely coupled configuration provides the following advantages:

1. High system throughput can be achieved by having more than once.
2. The system can be expanded in a modular form. Each bus master module is an independent unit and normally resides on a separate PC board. Therefore, a bus master module can be added or removed without affecting the other modules in the system.
3. A failure in one module normally does not cause a breakdown of the entire system and the faulty module can be easily detected and replaced.
4. Each bus master may have a local bus to access dedicated memory or I/O devices so that a greater degree of parallel processing can be achieved. More than one bus master module may have access to the shared system bus

Extra bus control logic must be provided to resolve the bus arbitration problem. The extra logic is called bus access logic and it is its responsibility to make sure that only one bus master at a time has control of the bus.

Simultaneous bus requests are resolved on a **priority basis:** There are three schemes for establishing priority as shown in Figure 2.3.6

1. Daisy chaining.
2. Polling.
3. Independent requesting

DAISY CHAINING:

- Need a bus controller to monitor bus busy and bus request signals
- Sends a bus grant to a Master and each Master either keeps the service or passes it on
- Controller synchronizes the clocks
- Master releases the Bus Busy signal when finished

POLLING:

- Controller sends address of device to grant bus access
- Can use priority resolution here:

- Highest priority is granted first, if it does not respond, then a lower priority is granted, and so on until someone accepts
- (ie: one request line, 3-bit grant line).

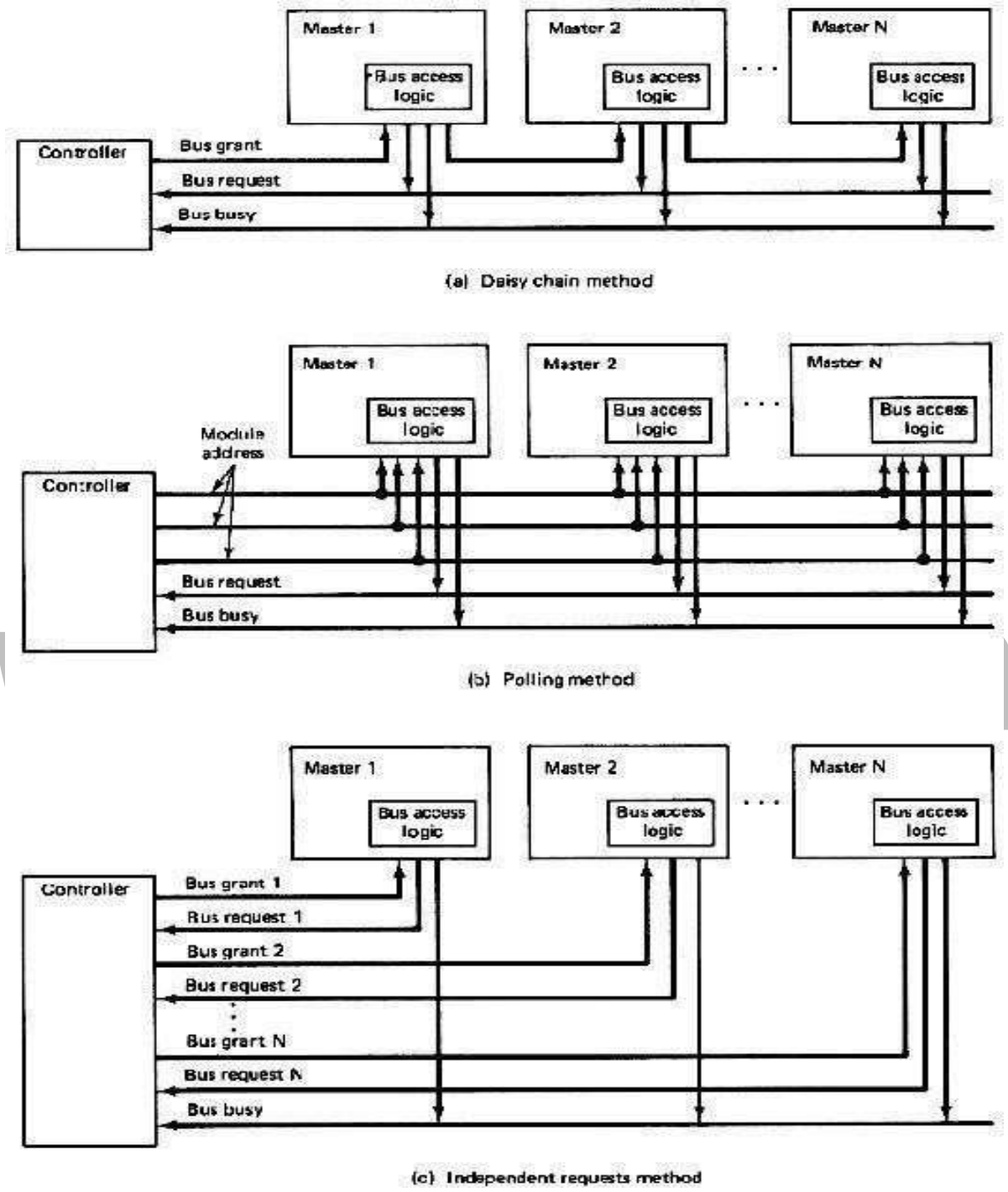


Figure 2.3.6 Bus allocation schemes

[Source: "Microcomputer Systems: The 8086 / 8088 Family - Architecture, Programming and Design" by Yu-Cheng Liu, Glenn A.Gibson]

INDEPENDENT:

- Each master has a request and grant line
- Now just a question of priority
- Could have fixed priority, rotating priority, etc.
- usually fixed because memory is desired to be the highest priority
- Synchronization of the clocks must be performed once a Master is recognized
- Master will receive a common clock from one side and pass it to the controller which will derive clock for transfer
- Can accurately predict calculations (since memory is always the highest priority)

2.1 8086 SIGNALS

The 8086 Microprocessor is a 16-bit CPU available in 3 clock rates, i.e. 5, 8 and 10MHz, packaged in a 40 pin CERDIP or plastic package. The 8086 Microprocessor operates in single processor or multiprocessor configurations to achieve high performance. The pin configuration is as shown in fig1. Some of the pins serve a particular function in minimum mode (single processor mode) and others function in maximum mode (multiprocessor mode) configuration.

The 8086 signals can be categorized in three groups. The first are the signals having common functions in minimum as well as maximum mode, the second are the signals which have special functions in minimum mode and third are the signals having special functions for maximum mode.

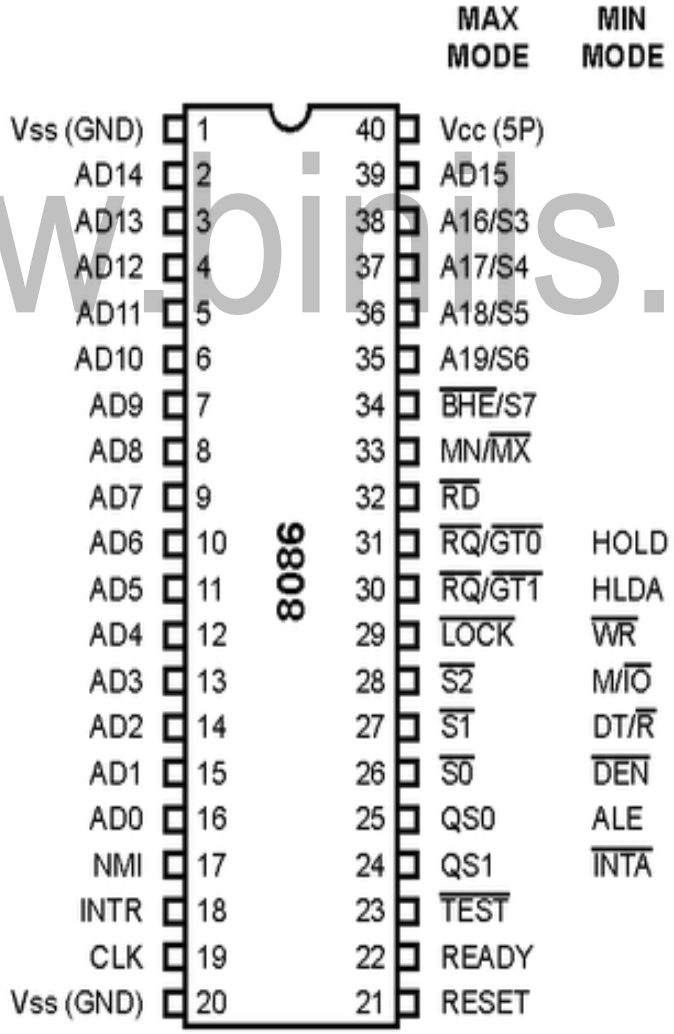


Figure 2.1.1 8086 signals

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M.Bhurchandi]

The following signal description are common for both the minimum and maximum modes. **AD15-AD0**: These are the time multiplexed memory I/O address and data lines. Address remains on the lines during T1 state, while the data is available on the data bus during T2, T3, TW and T4. Here T1, T2, T3, T4 and TW are the clock states of a machine cycle. TW Isa wait state. These lines are active high and float to a tractate during interrupt acknowledge and local bus hold acknowledge cycles.

A19/S6, A18/S5, A17/S4, and A16/S3: These are the time multiplexed address and status lines. During T1, these are the most significant address lines or memory operations. During I/O operations, these lines are low. During memory or I/O operations, status information is available on those lines for T2, T3, TW and T4

.The status of the interrupt enable flag bit (displayed on S5) is updated at the beginning of each clock cycle. The S4 and S3 combined indicate which segment register is presently being used for memory accesses as shown in Table 2.1.1. These lines float to tri-state off (restated) during the local bus hold acknowledge. The status line S6 is always low (logical). The address bits are separated from the status bits using latches controlled by the ALE signal.

S4	S3	Indication
0	0	Alternate Data
0	1	Stack
1	0	Code or none
1	1	Data

Table 2.1.1 S4 & S3 status

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M.Bhurchandi]

BHE/S7-Bus High Enable/Status:

The bus high enable signal is used to indicate the transfer of data over the higher order (D15-D8) data bus as shown in Table 2.1.2. It goes low for the data transfers over D15-D8 and is used to derive chip selects of odd address memory bank or peripherals. BHE is low during T1 for read, write and interrupt

Acknowledge cycles, when- ever a byte is to be transferred on the higher byte of the data bus. The status information is available during T2, T3 and T4. The signal is active low and is restated during ‘hold'. It is low during T1 for the first pulse of the interrupt acknowledges cycle.

BHE	A0	Indication
0	0	Whole word
0	1	Upper byte from or to odd address
1	0	Upper byte from or to even address
1	1	None

Table 2.1.2 Bus high enable status

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M.Bhurchandi]

RD-Read:

Read signal, when low, indicates the peripherals that the processor is performing a memory or I/O read operation. RD is active low and shows the state for T2, T3, and TW of any read cycle. The signal remains restated during the 'hold acknowledge'.

READY:

This is the acknowledgement from the slow devices or memory that they have completed the data transfer. The signal made available by the devices is synchronized by the 8284A clock generator to provide ready input to the 8086. The signal is active high.

INTR- interrupt Request:

This is a level triggered input. This is sampled during the last clock cycle of each instruction to determine the availability of the request. If any interrupt requests pending, the processor enters the interrupt acknowledge cycle. This can be internally masked by resetting the interrupt enable flag. This signal is active high and internally synchronized.

TEST:

This input is examined by a 'WAIT' instruction. If the TEST input goes low, execution will continue, else, the processor remains in an idle state. The input is synchronized internally during each clock cycle on leading edge of clock.

NMI-Non-mask able Interrupt:

This is an edge-triggered input which causes a Type2 interrupt. The NMI is not mask able internally by software. A transition from low to high initiates the interrupt response at the end of the current instruction. This input is internally synchronized.

RESET:

This input causes the processor to terminate the current activity and start execution from FFFF0H. The signal is active high and must be active for at least four clock cycles. It restarts execution when the RESET returns low. RESET is also internally synchronized.

CLK-Clock Input:

The clock input provides the basic timing for processor operation and bus control activity. It's an asymmetric square wave with 33% duty cycle. The range of frequency for different 8086 versions is from 5MHz to 10MHz.

VCC:

+5V power supply for the operation of the internal circuit. GND ground for the internal circuit.

MN/MX:

The logic level at this pin decides whether the processor is to operate in either minimum (single processor) or maximum (multiprocessor) mode.

THE FOLLOWING PIN FUNCTIONS ARE FOR THE MINIMUM MODE OPERATION OF 8086.

M/IO - Memory/IO:

This is a status line logically equivalent to S2 in maximum mode. When it is low, it indicates the CPU is having an I/O operation, and when it is high, it indicates that the CPU is having a memory operation. This line becomes active in the previous

T4 and remains active till final T4 of the current cycle. It is restated during local bus "hold acknowledge".

INTA -Interrupt Acknowledge:

This signal is used as a read strobe for interrupt acknowledge cycles. In other words, when it goes low, it means that the processor has accepted the interrupt. It is active low during T2, T3 and TW of each interrupt acknowledge cycle.

ALE-Address latch Enable:

This output signal indicates the availability of the valid address on the address/data lines, and is connected to latch enable input of latches. This signal is active high and is never restated.

DT /R -Data Transmit/Receive:

This output is used to decide the direction of data flow through the Trans receivers (bidirectional buffers). When the processor sends out data, this signal is high and when the processor is receiving data, this signal is low. Logically, this is equivalent to S1 in maximum mode. Its timing is the same as M/I/O. This is restated during 'hold acknowledge'.

DEN-Data Enable

This signal indicates the availability of valid data over the address/data lines. It is used to enable the Trans receivers (bidirectional buffers) to separate the data from the multiplexed address/data signal. It is active from the middle ofT2 until the middle of T4 DEN is restated during 'hold acknowledge' cycle.

HOLD, HLDA-Hold/Hold Acknowledge:

When the HOLD line goes high, it indicates to the processor that another master is requesting the bus access. The processor, after receiving the HOLD request, issues the hold acknowledge signal on HLDA pin, in the middle of the next clock cycle after completing the current bus (instruction) cycle. At the same time, the processor floats the local bus and control lines.

When the processor detects the HOLD line low, it lowers the HLDA signal. HOLD is an asynchronous input and it should be externally synchronized. If the

DMA request is made while the CPU is performing a memory or I/O cycle, it will release the local bus during T 4 provided:

1. The request occurs on or before T 2 state of the current cycle.
2. The current cycle is not operating over the lower byte of a word (or operating on an odd address).
3. The current cycle is not the first acknowledge of an interrupt acknowledge sequence.
4. A Lock instruction is not being executed.

So far we have presented the pin descriptions of 8086 in minimum mode.

THE FOLLOWING PIN FUNCTIONS ARE APPLICABLE FOR MAXIMUM MODE OPERATION OF 8086.

S2, S1, S0 -Status Lines:

These are the status lines which reflect the type of operation, being carried out by the processor. These become active during T4 of the previous cycle and remain active during T1 and T2 of the current bus cycle. The status lines return to passive state during T3 of the current bus cycle so that they may again become active for the next bus cycle during T4. Any change in these lines during T3 indicates the starting of a new cycle, and return to passive state indicates end of the bus cycle. These status lines are encoded in Table 2.1.3.

S 2	S 1	S 0	Indication
0	0	0	Interrupt acknowledge
0	0	1	Read I/O port
0	1	0	Write I/O port
0	1	1	Halt
1	0	0	Code Access
1	0	1	Read Memory

1	**1**	**0**	**Write Memory**
1	**1**	**1**	**Passive**

Table 2.1.3 Status lines S2, S1 & S0

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M.Bhurchandi]

LOCK: This output pin indicates that other system bus masters will be prevented from gaining the system bus, while the LOCK signal is low. The LOCK signal is activated by the 'LOCK' prefix instruction and remains active until the completion of the next instruction. This floats to tri-state off during "hold acknowledge". When the CPU is executing a critical instruction which requires the system bus, the LOCK prefix instruction ensures that other processors connected in the system will not gain the control of the bus. The 8086, while executing the prefixed instruction, asserts the bus lock signal output, which may be connected to an external bus controller.

QS1, QS0-Queue Status:

These lines give information about the status of the code prefect queue. These are active during the CLK cycle after which the queue operation is performed. These are encoded as shown in Table2.1.4.

QS1	QS2	Indication
0	**0**	**No operation**
0	**1**	**First byte opcode from the Queue**
1	**0**	**Empty Queue**
1	**1**	**Subsequent byte from the Queue**

Table 2.1.4.Queue Status

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M.Bhurchandi]

RQ/GT0, RQ/GT1-ReQuest/Grant:

These pins are used by other local bus masters, in maximum mode, to force the processor to release the local bus at the end of the processor's current bus cycle. Each of the pins is bidirectional with RQ/GT0 having higher priority than RQ/ GT1, RQ/GT pins have internal pull-up resistors and may be left unconnected. The request Grant sequence is as follows:

1. A pulse one clock wide from another bus master requests the bus access to8086.
2. During T4 (current) or T1 (next) clock cycle, a pulse one clock wide from 8086 to the requesting master, indicates that the 8086 has allowed the local bus to float and that it will enter the "hold acknowledge" state at next clock cycle. The CPU's bus interface unit is likely to be disconnected from the local bus of the system.
3. A one clock wide pulse from another master indicates to 8086 that the 'hold' request is about to end and the 8086 may regain control of the local bus at the next clock cycle. Thus each master to master exchange of the local bus is a sequence of 3 pulses. There must be at least one dead clock cycle after each bus exchange. The request and grant pulses are active low. For the bus requests those are received while 8086 is performing memory or I/O cycle, the granting of the bus is governed by the rules as discussed in case of HOLD, and HLDA in minimum mode.

2.2 BASIC CONFIGURATIONS, SYSTEM BUS TIMINGS, SYSTEM DESIGN USING 8086

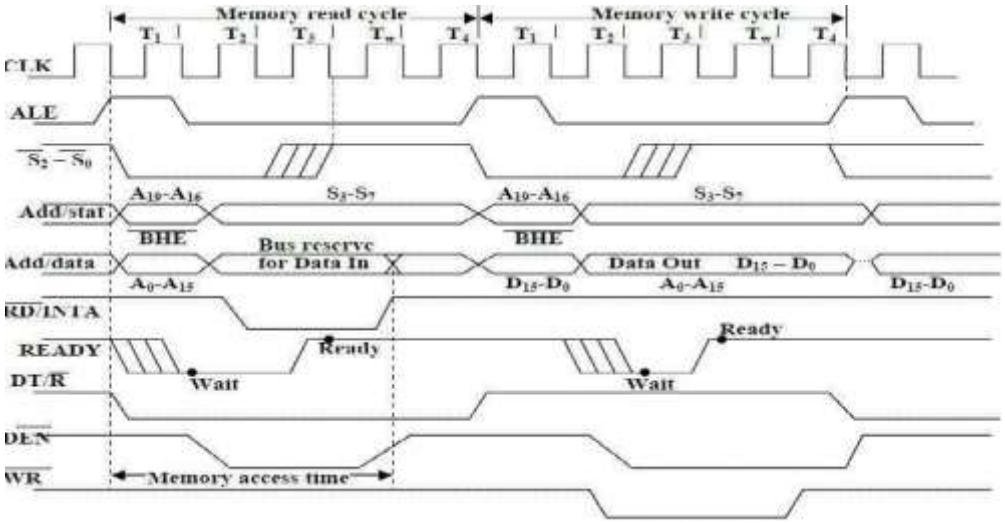
BASIC CONFIGURATION

READ WRITE TIMING DIAGRAM - GENERAL BUS OPERATION

The 8086 has a combined address and data bus commonly referred as a time multiplexed address and data bus. The main reason behind multiplexing address and data over the same pins is the maximum utilization of processor pins and it facilitates the use of 40 pin standard DIP package. The bus can be DE multiplexed using a few latches and Trans receivers, whenever required.

Basically, all the processor bus cycles consist of at least four clock cycles. These are referred to as T1, T2, T3, and T4. The address is transmitted by the processor during T1, It is present on the bus only for one cycle. The negative edge of this ALE pulse is used to separate the address and the data or status information as shown in Figure 2.2.1...

In maximum mode, the status lines S0, S1 and S2 are used to indicate the type of operation. Status bits S3 to S7 are multiplexed with higher order address bits and the BHE signal.



Address is valid during T1 while status bits S3 to S7 are valid during T2 through T4.s operation cycle

[Source: Advanced Microprocessors andMicrocontrollers by A.K Ray & K.M. Bhurchandi]

SYSTEM BUS TIMINGS:

MINIMUM MODE 8086 SYSTEM AND TIMINGS

In a minimum mode 8086 system, the microprocessor 8086 is operated in minimum mode by strapping its MN/MX* pin to logic1. In this mode, all the control signals are given out by the microprocessor chip itself. There is a single microprocessor in the minimum mode system. The remaining components in the system are latches, Tran's receivers, clock generator, memory and I/O devices.

The code fetch and read cycles are similar. Hence the timing diagram can be categorized in two parts, the first is the timing diagram for read cycle and the second is the timing diagram for write cycle.

Figure 2.2.2 shows the read cycle timing diagram. The read cycle begins in T1 with the assertion of the address latch enable (ALE) signal and also M/IO* signal. During the negative going edge of this signal, the valid address is latched on the local bus. The BHE* and A0 signals address low, high or both bytes.

From To to T4, the M/IO* signal indicates a memory or I/O operation. At T2 the address is removed from the local bus and is sent to the output. The bus is then restated. The read (RD*) control signal is also activated in T2 .The read (RD) signal causes the addressed device to enable its data bus drivers.

After RD* goes low, the valid data is available on the data bus. The addressed device will drive the READY line high, when the processor returns the read signal to high level, the addressed device will again tractate its bus drivers.

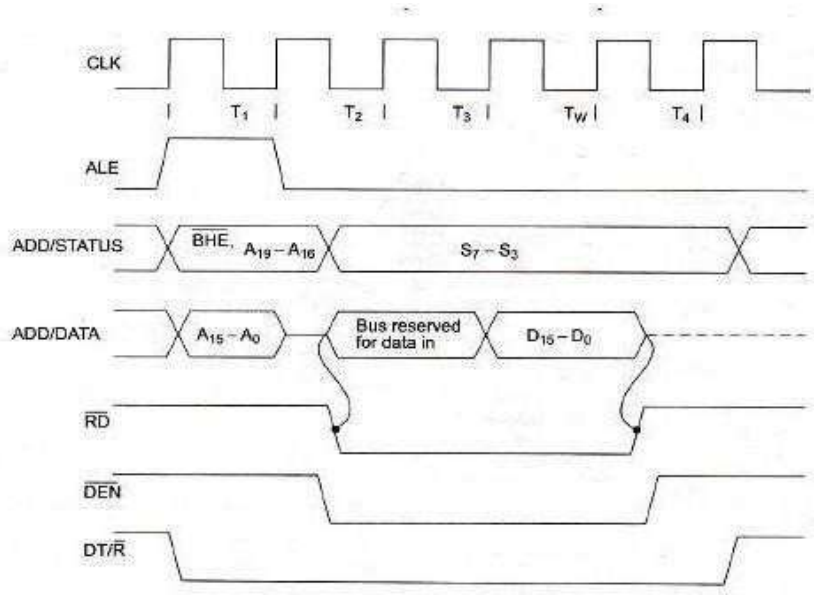


Figure 2.2.2 Read cycle timing diagram for minimum mode

[Source: AdvancedMicroprocessors and Microcontrollers by A.K Ray & K.M. Bhurchandi]

A write cycle also begins with the assertion of ALE and the emission of the address. The M/IO* signal is again asserted to indicate a memory or I/O operation. In T2 after sending the address in to the processor sends the data to be written to the addressed location. The data remains on the bus until middle of T4 state. The WR* becomes active at the beginning ofT2 (unlike RD* is somewhat delayed in T2 to provide time for floating). The BHE* and A0 signals are used to select the proper byte or bytes of memory or I/O word to be read or written. The M/IO*, RD* and WR* signals indicate the types of data transfer.

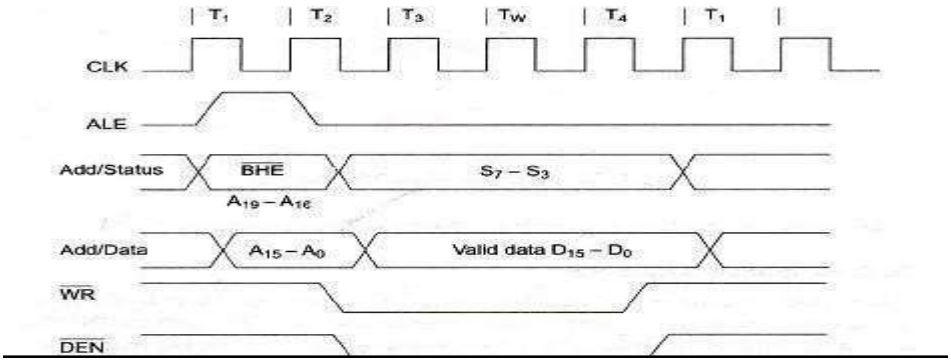


Figure 2.2.3 Write cycle timing diagram for minimum mode

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M. Bhurchandi]

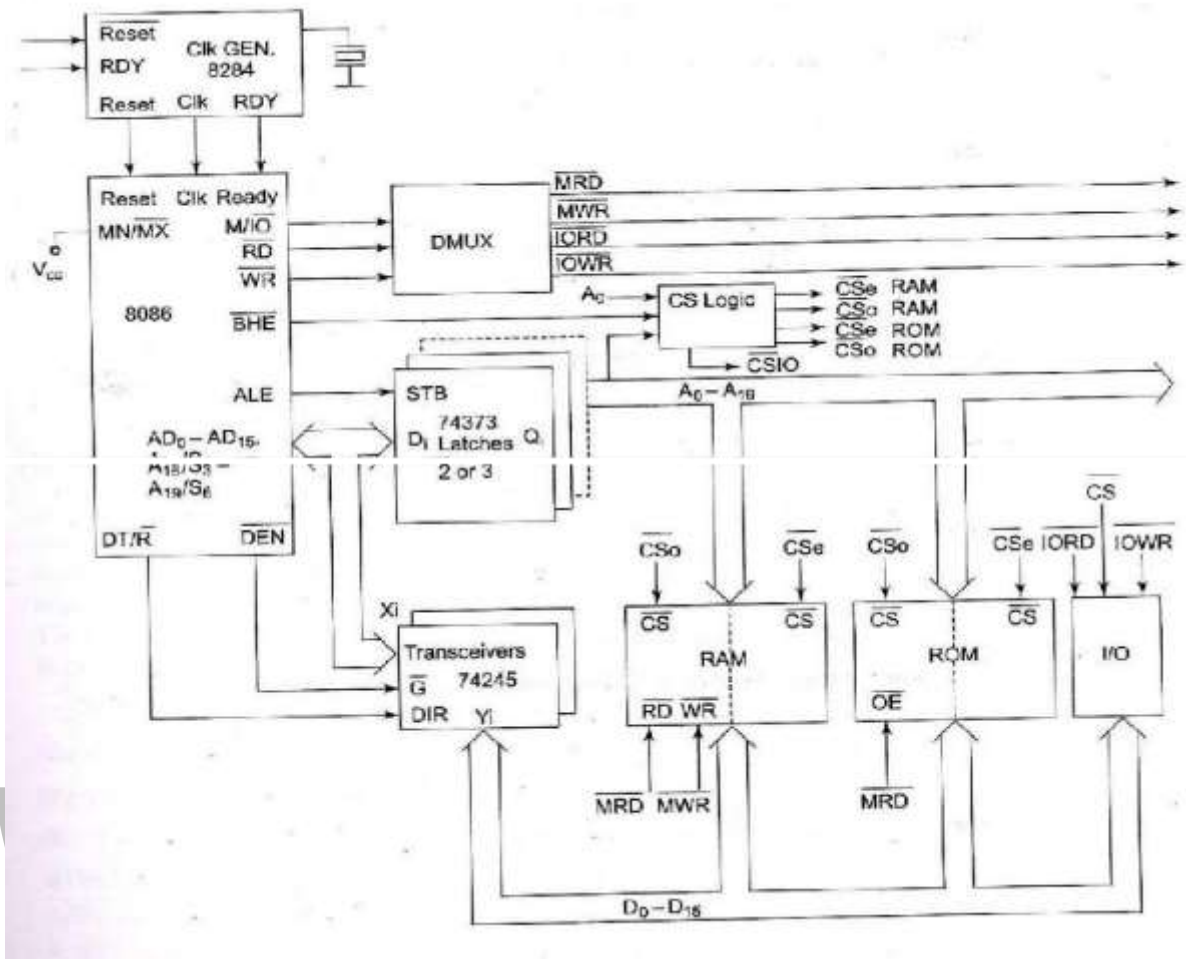


Figure 2.2.4 Minimum mode configuration

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M. Bhurchandi]

SYSTEM DESIGN USING 8086

MAXIMUM MODE 8086 SYSTEM AND TIMINGS

In the maximum mode, the 8086 is operated by strapping the MN/MX* pin to ground. In this mode, the processor derives the status signals S2*, S1* and S0*. Another chip called bus controller derives the control signals using this status information. In the maximum mode, there may be more than one microprocessor in the system configuration.

The basic functions of the bus controller chip IC8288, is to derive control signals likeRD* and WR* (for memory and I/O devices), DEN*, DT/R*, ALE, etc. using the information made available by the processor on the status lines. The bus controller chip has input lines S2*, S1* and S0* and CLK. The CPU drives these inputs to 8288. It derives the outputs ALE, DEN*, DT/R*, MWTC*, AMWC*, IORC*, IOWC* and AIOWC*. The AEN*, IOB and CEN pins are especially useful for multiprocessor systems. AEN* and IOB are generally grounded. CEN pin is usually tied to+5V.

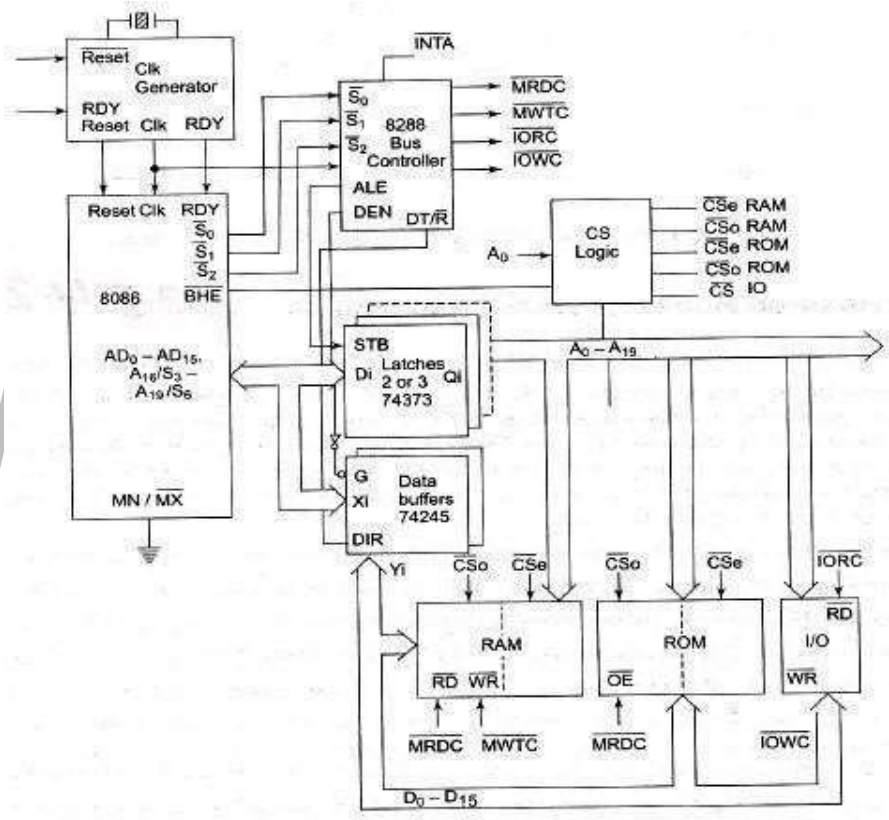


Figure 2.2.5 Maximum mode configuration

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M. Bhurchandi]

The significance of the MCE/PDEN* output depends upon the status of the IOB pin. If IOB is grounded, it acts as master cascade enable to control cascaded 8259A; else it acts as peripheral data enable used in the multiple bus configurations.

INTA* pin is used to issue two interrupt acknowledge pulses to the interrupt controller or to an interrupting device.

IORC*, IOWC* are I/O read command and I/Owrite command signals respectively.

These signals enable an IO interface to read or write the data from or to the addressed port. The MRDC*, MWTC* are memory read command and memory write command signals respectively and may be used as memory read and write signals. All these command signals instruct the memory to accept or send data from or to the bus.

For both of these write command signals, the advanced signals namely AIOWC* and AMWTC* are available. They also serve the same purpose, but are activated one clock cycle earlier than the IOWC* and MWTC* signals, respectively. The maximum mode system is shown in Figure 2.2.5.

The maximum mode system timing diagrams are also divided in two portions as read (input) and write (output) timing diagrams. The address/data and address/status timings are similar to the minimum mode. ALE is asserted in T1, just like minimum mode. The only difference lies in the status signals used and the available control and advanced command signals. The Figure 2.2.6 shows the maximum mode timings for the read operation while the Figure 2.2.7 shows the same for the write operation.

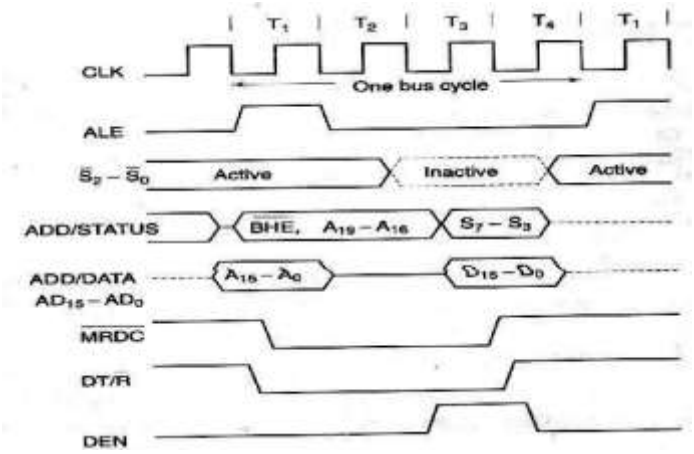


Figure 2.2.6 Memory Read Cycle

[Source: Advanced Microprocessors andMcroicontrollers by A.K Ray & K.M. Bhurchandi]

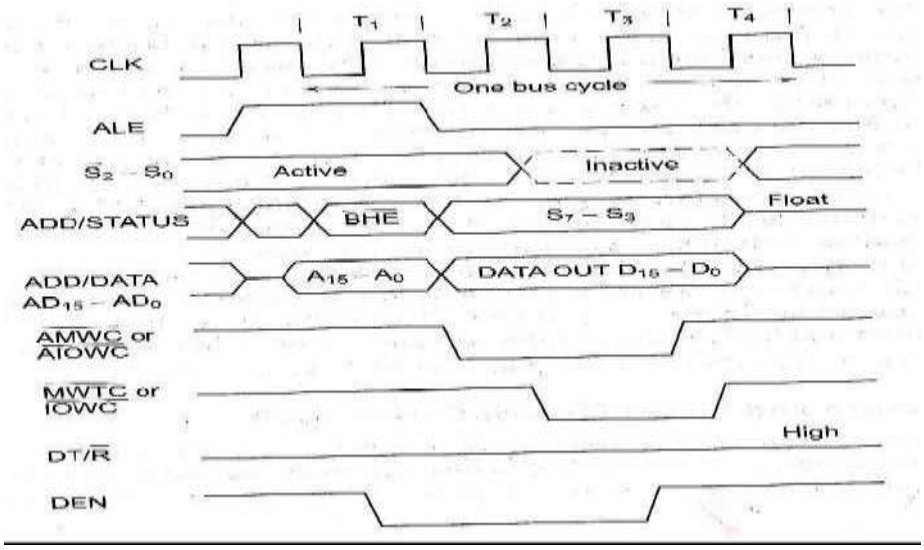


Figure 2.2.7 Memory write cycle

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K

2.4 INTRODUCTION TO ADVANCED PROCESSORS:

80286 MICROPROCESSOR

SALIENT FEATURES OF 80286

The 80286 is the first member of the family of advanced microprocessors with memory management and protection abilities. The 80286 CPU, with its 24-bit address bus is able to address 16 Mbytes of physical memory. Various versions of 80286 are available that runs on 12.5 MHz, 10 MHz and 8 MHz clock frequencies. 80286 is upwardly compatible with8086 in terms of instruction set.

80286 has two operating modes namely real address mode and virtual address mode.

In real address mode, the 80286 can address up to 1Mb of physical memory address like 8086.In virtual address mode, it can address up to 16 Mb of physical memory address space and 1GB of virtual memory address space.

The instruction set of 80286 includes the instructions of 8086 and 80186. 80286 has some extra instructions to support operating system and memory management. In real address mode, the 80286 is object code compatible with 8086. In protected virtual address mode, it is source code compatible with 8086. The performance of 80286 is five times faster than the standard 8086.

NEED FOR MEMORY MANAGEMENT

The part of main memory in which the operating system and other system programs are stored is not accessible to the users. It is required to ensure the smooth execution of the running process and also to ensure their protection. The memory management which is an important task of the operating system is supported by a hardware unit called memory management unit.

Swapping in of the Program

Fetching of the application program from the secondary memory and placing it in the physical memory for execution by the CPU.

Swapping out of the executable Program

Saving a portion of the program or important results required for further execution back to the secondary memory to make the program memory free for

Further execution of another required portion of the program.

CONCEPT OF VIRTUAL MEMORY

Large application programs requiring memory much more than the physically available 16 Mbytes of memory, may be executed by diving it into smaller segments. Thus for the user, there exists a very large logical memory space which is not actually available. Thus there exists a virtual memory which does not exist physically in a system. This complete process of virtual memory management is taken care of by the 80286 CPU and the supporting operating system.

INTERNAL ARCHITECTURE OF 80286

Register Organization of 80286.The 80286 CPU contains almost the same set of registers, as in 8086, namely

1. Eight 16-bit general purpose registers
2. Four 16-bit segment registers
3. Status and control registers
4. .Instruction Pointer

The register set of 80286 is shown in Fig. 4.1.

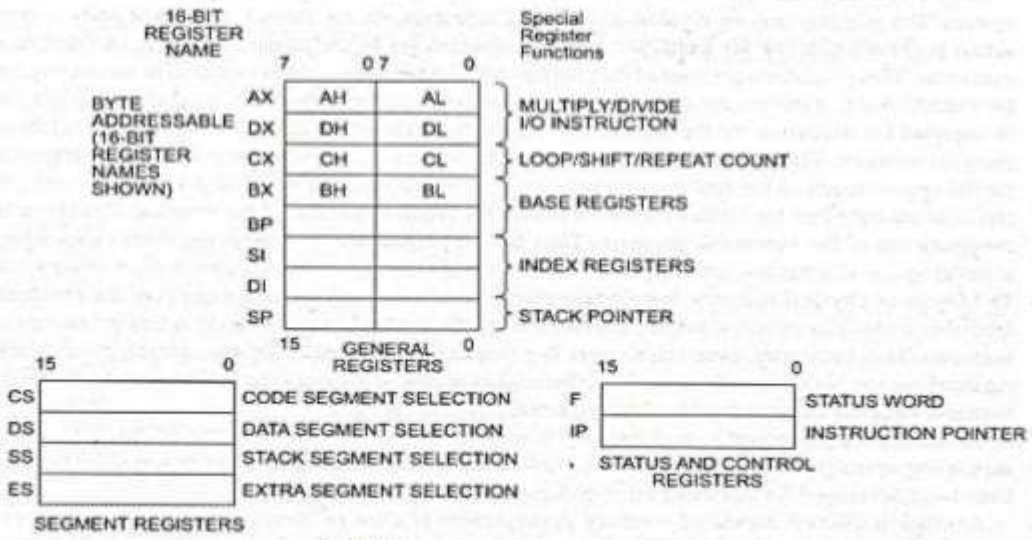


Fig.4.1 *Register Set of 80286 (Intel Corp.)*

The flag register reflects the results of logical and arithmetic instructions.

-	NT	IOPL	OF	DF	IF	TF	SF	ZF	-	AF	-	PF	-	CF

Figure 2.4.2 Flag register

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M. Bhurchandi]

D2, D4, D6, D7 and D11 are called as status flag bits. The bits D8 (TF) and D9 (IF) are used for controlling machine operation and thus they are called control flags. The additional fields available in 80286 flag registers are:

1. IOPL - I/O Privilege Field (bits D12 and D13)
2. NT - Nested Task flag (bit D14)
3. PE - Protection Enable (bit D16)
4. MP - Monitor Processor Extension (bit D17)
5. EM - Processor Extension Emulator (bit D18)
6. TS – Task Switch (bit D19)

Protection Enable flag places the 80286 in protected mode, if set. This can only be cleared by resetting the CPU. If the Monitor Processor Extension flag is set, allows WAIT instruction to generate a processor extension not present exception.

Task Switch flag if set, indicates the next instruction using extension will generate exception 7, permitting the CPU to test whether the current processor extension is for the current task.

MACHINE STATUS WORD (MSW)

The machine status word consists of four flags – PE, MO, EM and TS of the four lower order bits D19 to D16 of the upper word of the flag register. The LMSW and SMSW instructions are available in the instruction set of 80286 to write and read The MSW in real address mode.

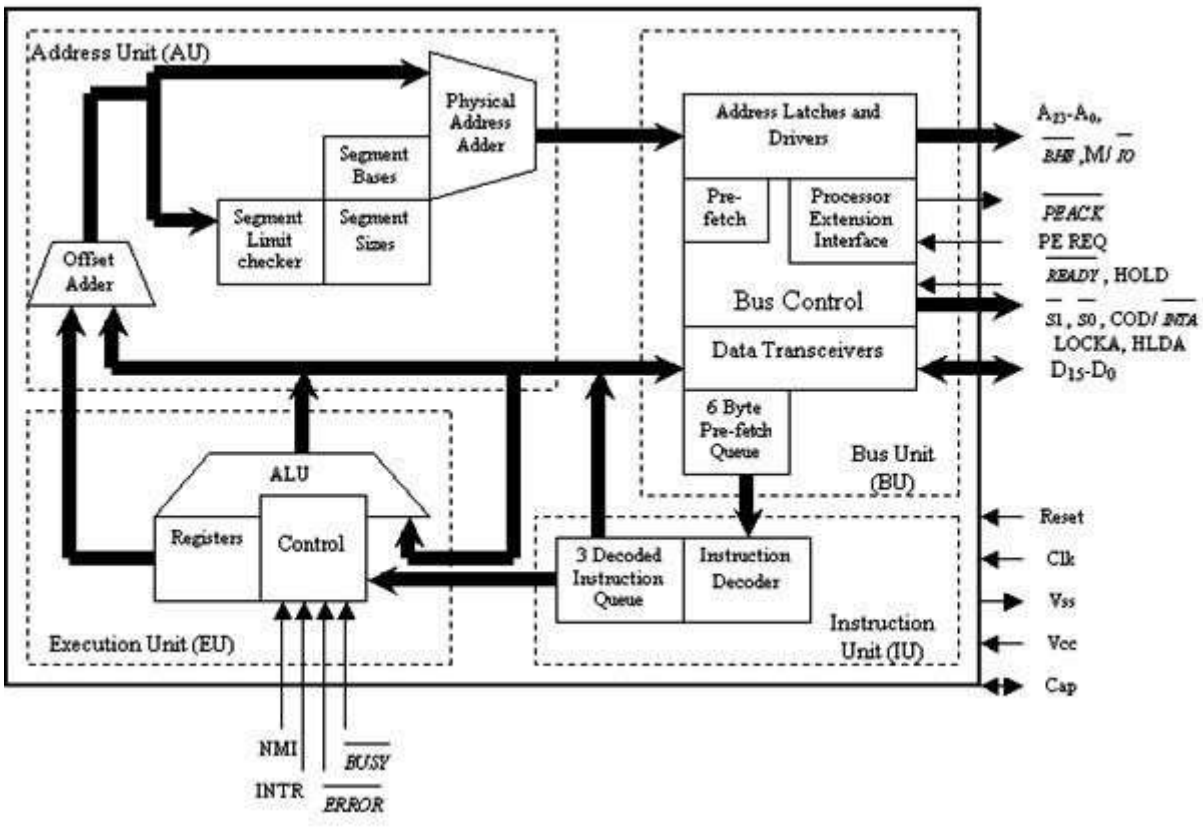


Figure 2.4.3 Internal Block diagram of 80286

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M. Bhurchandi]

The CPU contain four functional blocks

1. Address Unit (AU)
2. Bus Inuit (BU)
3. Instruction Unit (IU)
4. Execution Unit (EU)

The address unit is responsible for calculating the physical address of instructions and data that the CPU wants to access. Also the address lines derived by this unit may be used to address different peripherals. The physical address computed by the address unit is handed over to the bus unit (BU) of the CPU. Major function of the bus unit is to fetch instruction bytes from the memory. Instructions are fetched in advance and stored in a queue to enable faster execution of the instructions. The bus unit also contains a bus control module that controls the prefacer module. These perfected instructions are arranged in a 6-byte instructions queue. The 6-byte prefect queue forwards the instructions arranged in it to the **instruction unit** (IU). The instruction unit accepts instructions from the prefect queue and an instruction decoder

Decodes them one by one. The decoded instructions are latched onto a decoded instruction queue. The output of the decoding circuit drives a control circuit in the **execution unit,** which is responsible for executing the instructions received from decoded instruction queue. The decoded instruction queue sends the data part of the instruction over the data bus. The EU contains the register bank used for storing the data as scratch pad, or used as special purpose registers. The ALU, the heart of the EU, carries out all the arithmetic and logical operations and sends the results over the data bus or back to the register bank.

INTERRUPTS OF 80286

The Interrupts of 80286 may be divided into three categories,

1. External or hardware interrupts
2. INT instruction or software interrupts
3. Interrupts generated internally by exceptions

While executing an instruction, the CPU may sometimes be confronted with a special situation because of which further execution is not permitted. While trying to execute a divide by zero instruction, the CPU detects a major error and stops further execution. In this case, we say that an exception has been generated. In other words, an instruction exception is an unusual situation encountered during execution of an instruction that stops further execution. The return address from an exception, in most of the cases, points to the instruction that caused the exception.

As in the case of 8086, the interrupt vector table of 80286 requires 1Kbytes of space forstoring 256, four-byte pointers to point to the corresponding 256 interrupt service routines (lSR). Each pointer contains a 16-bit offset followed by a 16-bit segment selector to point to a particular ISR. The calculation of vector pointer address in the interrupt vector table from the (8-bit) INT type is exactly similar to 8086.

Like 8086, the 80286 supports the software interrupts of type 0 (INT 00) to type FFH (INT FFH).

MASKABLE INTERRUPT INTR:

This is a mask able interrupt input pin of which the INT type is to be provided by an external circuit like an interrupt controller. The other functional details of this

interrupt pin are exactly similar to the INTR input of 8086.

NON-MASKABLE INTERRUPT NMI:

It has higher priority than the INTR interrupt.Whenever this interrupt is received, a vector value of 02 is supplied internally to calculate the pointer to the interrupt vector table. Once the CPU responds to a NMI request, it does not serve any other interrupt request (including NMI). Further it does not serve the processor extension (coprocessor) segment overrun interrupt, till either it executes IRET or it is reset. To start with, this clears the IF flag which is set again With the execution of IRET, i.e. return from interrupt.

SIGNAL DESCRIPTION OF 80286

CLK: This is the system clock input pin. The clock frequency applied at this pin is divided by two internally and is used for deriving fundamental timings for basic operations of the circuit. The clock is generated using 8284 clock generator.

D15-D0: These are sixteen bidirectional data bus lines.

A23-A0: These are the physical address output lines used to address memory or I/O devices. The address lines A23 - A16 are zero during I/O transfers

BHE: This output signal, as in 8086, indicates that there is a transfer on the higher byte of the data bus (D15 – D8).

S1 , S0: These are the active-low status output signals which indicate initiation of a bus cycle and with M/IO and COD/INTA, they define the type of the bus cycle.

M/ IO: This output line differentiates memory operations from I/O operations. If this signal is it “0” indicates that an I/O cycle or INTA cycle is in process and if it is “1” it indicates that a memory or a HALT cycle is in progress.

COD/ INTA: This output signal, in combination with M/ IO signal and S1, S0 distinguishes different memory, I/O and INTA cycles.

LOCK: This active-low output pin is used to prevent the other masters from gaining the control of the bus for the current and the following bus cycles. This pin is activated by a "LOCK" instruction prefix, or automatically by hardware during XCHG, interrupt acknowledge or descriptor table access

READY This active-low input pin is used to insert wait states in a bus cycle, for

interfacing low speed peripherals. This signal is neglected during HLDA cycle.

HOLD and HLDA This pair of pins is used by external bus masters to request for the controlof the system bus (HOLD) and to check whether the main processor has granted the control (HLDA) or not, in the same way as it was in 8086.

INTR: Through this active high input, an external device requests 80286 to suspend the current instruction execution and serve the interrupt request. Its function is exactly similar to that of INTR pin of 8086.

NMI: The Non-Mask able Interrupt request is an active-high, edge-triggered input that is equivalent to an INTR signal of type 2. No acknowledge cycles are needed to be carried out. PEREG and PEACK (Processor Extension Request and Acknowledgement),

Processor extension refers to coprocessor (80287 in case of 80286 CPU). This pair of pins extends the memory management and protection capabilities of 80286 to the processor extension 80287. The PEREQ input requests the 80286 to perform a data operand transfer for a processor extension. The PEACK active-low output indicates to the processor extension that the requested operand is being transferred.

BUSY and **ERROR:** Processor extension BUSY and ERROR active-low input signals indicate the operating conditions of a processor extension to 80286. The BUSY goes low, indicating 80286 to suspend the execution and wait until the BUSY become inactive. In this duration, the processor extension is busy with its allotted job. Once the job is completed the processor extension drives the BUSY input high indicating 80286 to continue with the program execution. An active ERROR signal causes the 80286 to perform the processor extension interrupt while executing the WAIT and ESC instructions. The active ERROR signal indicates to 80286 that the processor extension has committed a mistake and hence it is reactivating the processor extension interrupt.

CAP: A 0.047 if, 12V capacitor must be connected between this input pin and ground to filter the output of the internal substrate bias generator. For correct operation of 80286 the capacitor must be charged to its operating voltage. Till this capacitor charges to its full capacity, the 80286 may be kept stuck to reset to avoid any spurious activity.

Vss: This pin is a system ground pin of 80286.

Vcc: This pin is used to apply +5V power supply voltage to the internal circuit of 80286. RESET The active-high RESET input clears the internal logic of 80286, and reinitializes it **RESET** The active-high reset input pulse width should be at least 16 clock cycles. The 80286 requires at least 38 clock cycles after the trailing edge of the RESET input signal,before it makes the first opcode fetch cycle.

Real Address Mode

- Act as a fast 8086
- Instruction set is upwardly compatible
- It address only 1 M byte of physical memory using A0-A19.
- In real addressing mode of operation of 80286, it just acts as a fast 8086.

The instruction set is upward compatible with that of 8086.

The 80286 addresses only 1Mbytes of physical memory using A0- A19. The Lines A20-A23 are not used by the internal circuit of 80286 in this mode. In real address mode, while addressing the physical memory, the 80286 uses BHE along with A0- A19. The 20-bit physical address is again formed in the same way as that in 8086.

The contents of segment registers are used as segment base addresses. The other registers, depending upon the addressing mode, contain the offset addresses. Because of extra pipelining and other circuit level improvements, in real address mode also, the 80286 operates at a much faster rate than 8086, although functionally they work in an identical fashion. As in 8086, the physical memory is organized in terms of segments of 64Kbyte maximum size.

An exception is generated, if the segment size limit is exceeded by the instruction or the data. The overlapping of physical memory segments is allowed to minimize the memory requirements for a task. The 80286 reserves two fixed areas of physical memory for system initialization and interrupt vector table. In the real mode the first 1Kbyte of memory starting from address 0000H to 003FFH is reserved for interrupt vector table. Also the addresses from FFFF0H to FFFFFH are reserved for system initialization.

The program execution starts from FFFFH after reset and initialization. The

interrupt vector table of 80286 is organized in the same way as that of 8086. Some of the interrupt types are reserved for exceptions, single-stepping and processor extension segment overrun, etc

When the 80286 is reset, it always starts the execution in real address mode. In real address mode, it performs the following functions: it initializes the IP and other registers of 80286, it prepares for entering the protected virtual address mode.

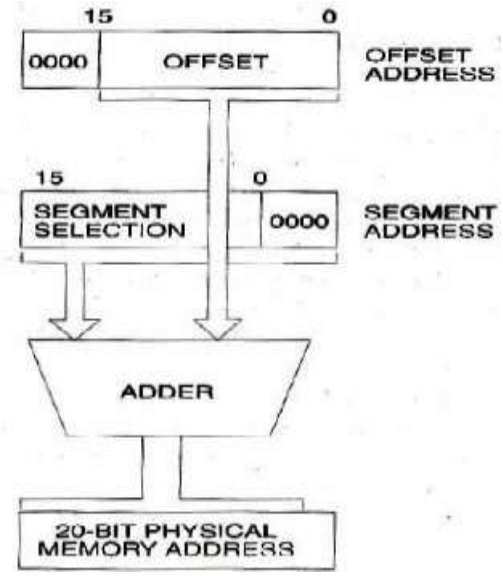


Figure 2.4.4 Real Address calculation

[Source: Advanced Microprocessors and Microcontrollers by A.K Ray & K.M. Bhurchandi]

PROTECTED VIRTUAL ADDRESS MODE (PVAM)

80286 is the first processor to support the concepts of virtual memory and memory management. The virtual memory does not exist physically it still appears to be available within the system. The concept of VM is implemented using Physical memory that the Pecan directly access and secondary memory that is used as a storage for data and program, which are stored in secondary memory initially.

The Segment of the program or data required for actual execution at that instant is fetched from the secondary memory into physical memory. After the execution of this fetched segment, the next segment required for further execution is again fetched from the secondary memory, while the results of the executed segment are stored back into the secondary memory for further references. This continues till the complete program is executed

During the execution the partial results of the previously executed portions are

again fetched into the physical memory, if required for further execution. The procedure of fetching the chosen program segments or data from the secondary storage into physical memory is called *swapping*. The procedure of storing back the partial results or data back on thesecondary storage is called *unswapping*. The virtual memory is allotted per task.

The 80286 is able to address 1 G byte (230 bytes) of virtual memory per task. The complete virtual memory is mapped on to the 16Mbyte physical memory. If a program larger than 16Mbyte is stored on the hard disk and is to be executed, if it is fetched in terms of data or program segments of less than 16Mbyte in size into the program memory by swapping sequentially as per sequence of execution.

Whenever the portion of a program is required for execution by the CPU, it is fetched from the secondary memory and placed in the physical memory is called *swapping in* of the program. A portion of the program or important partial results required for further execution, may be saved back on secondary storage to make the PM free for further execution of another required portion of the program is called *swapping out* of the executable program.

80286 uses the 16-bit content of a segment register as a selector to address a descriptor stored in the physical memory. The descriptor is a block of contiguous memory locations containing information of a segment, like segment base address, segment limit, segment type, privilege level, segment availability in physical memory; descriptor type and segment use another task.

2.5 I/O PROGRAMMING

On the 8086, all programmed communications with the I/O ports is done by the IN and OUT instructions.

IN and OUT instructions

Name Mnemonic and Format Description

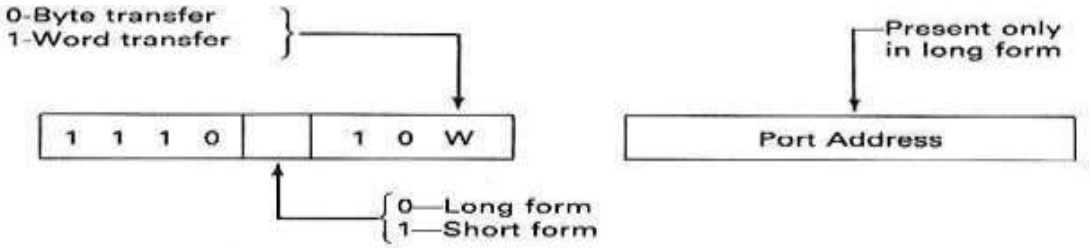
Long form, byte	IN AL, PORT	(AL) <- (PORT)
Long form, word	IN AX, PORT	(AX) <- (PORT+1: PORT)
Short form, byte	IN AL, DX	(AL) <- ((DX))
Short form, word	IN AX, DX	(AX) <- ((DX) + 1: (DX))
Long form, byte	OUT PORT, AL	(PORT) <- (AL)
Long form, word	OUT PORT, AX	(PORT+1: PORT) <- (AX)
Short form, byte	OUT DX, AL	((DX)) <- (AL)
Short form, word	OUT DX, AX	((DX)+1: (DX)) <- (AX)

Note: PORT is a constant ranging from 0 to 255

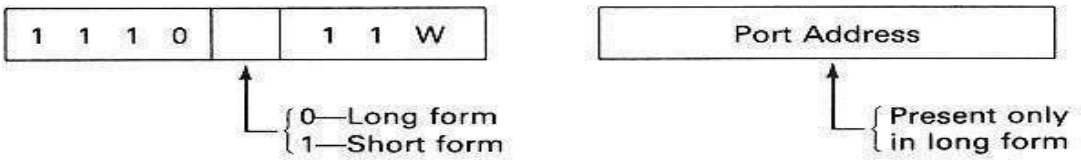
Flags: No flags are affected

Addressing modes: Operands are limited as indicated above. If the second operand is DX, then there is only one byte in the instruction and the contents of DX are used as the port address. Unlike memory addressing, the contents of DX are not modified by any segment register. This allows variable access to I/O ports in the range 0 to 64K.

The machine language code for the IN instruction is:



Although AL or AX is implied as the first operand in an IN instruction, either AL or AX must be specified so that the assembler can determine the W-bit. Similar comments apply to the OUT instruction except that for it the port address is the destination and is therefore indicated by the first operand, and the second operand is either AL or AX. Its machine code is:



Note that if the long form of the IN or OUT instruction is used the port address must be in the range 0000 to 00FF, but for the short form it can be any address in the range 0000 to FFFF (i.e. any address in the I/O address space). Neither IN nor OUT affects the flags.

The IN instruction may be used to input data from a data buffer register or the status from status register. The instructions

IN AX, 28H

MOV DATA_WORD, AX

Would move the word in the ports whose address are 0028 and 0029 to the memory location DATA_WORD.