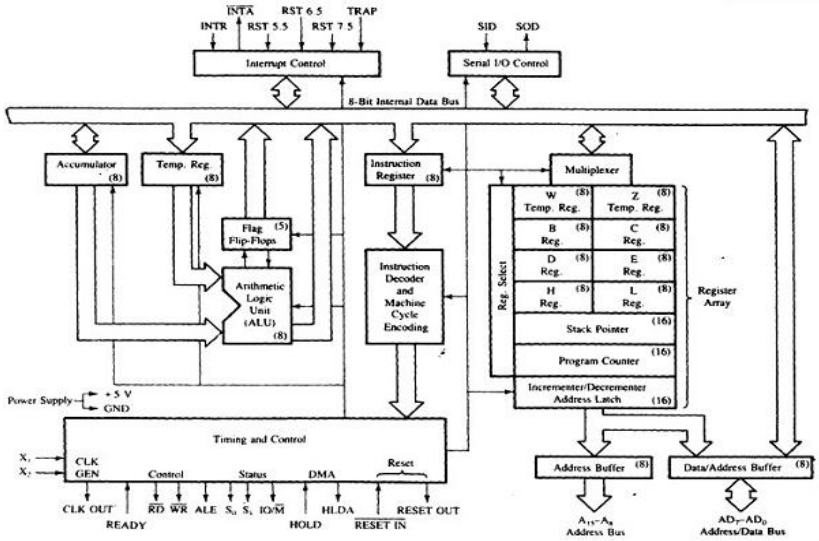
MODULE 1 EVOLUTION OF MICROPROCESSOR

A microprocessor is a computer processor which incorporates the functions of a computer's central processing unit (CPU) on a single integrated circuit (IC), or at most a few integrated circuits The microprocessor is a multipurpose, clock driven, register based, digital-integrated circuit which accepts binary data as input, processes it according to instructions stored in its memory, and provides results as output. Microprocessors contain both combinational logic and sequential digital logic. Microprocessors operate on numbers and symbols represented in the binary numeral system.

8085 ARCHITECTURE



ALU: The Arithmetic and Logic Unit**,** ALU performs the arithmetic and logical operations:

- Addition
- Subtraction
- Logical AND
- Logical OR
- Logical EXCLUSIVE OR
- Complement (Logical NOT)
- Increment (add 1)
- Decrement (subtract 1)
- Left shift, Rotate left, Rotate right
- Clear, etc.

Timing and Control Unit: The timing and control unit is the section of the CPU.

- It is used to generate timing and control signals which are necessary for the execution of instructions.
- It is used to control data flow between CPU and peripherals (including memory).
- It is used to provide status, control and timing signals which are required for the operation of memory and I/O devices.
- It is used to control the entire operations of the microprocessor and peripherals connected to it.

Registers: Registers are used for temporary storage and manipulation of data and instructions by the microprocessor. Data remain in the registers till they are sent to the I/O devices or memory. Intel 8085 microprocessor has the following registers:

- One 8-bit accumulator (ACC) i.e. register A
- Six general purpose registers of 8-bit, these are B,C, D, E, H and L

- One 16-bit stack pointer, SP
- One 16-bit Program Counter, PC
- Instruction register
- Temporary register

In addition to the above mentioned registers the 8085 microprocessor contains a set of five flip-flops which serve as flags (or status flags). A flag is a flip-flop which indicates some conditions which arises after the execution of an arithmetic or logical instruction.

1. **Accumulator (ACC):** The accumulator is an 8-bit register associated with the ALU. The register 'A' is an accumulator in the 8085. It is used to hold one of the operands of an arithmetic and logical operation. The final result of an arithmetic or logical operation is also placed in the accumulator.
2. **General-Purpose Registers:** The 8085 microprocessor contains six 8-bit general purpose registers. They are: B, D, C, E, H and L register.
   To hold data of 16-bit a combination of two 8-bit registers can be employed.
   The combination of two 8-bit registers is called **register pair**. The valid register pairs in the 8085 are: D-E, B-C and H-L. The H-L pair is used to act as a memory pointer.
3. **Program Counter (PC):** It is a 16-bit special purpose register. It is used to hold the address of memory of the next instruction to be executed. It keeps the track of the instruction in a program while they are being executed. The microprocessor increments the content of the next program counter during the execution of an instruction so that at the end of the execution of an instruction it points to the next instructions address in the program.
4. **Stack Pointer (SP):** It is a 16-bit special function register used as memory pointer. A stack is nothing but a portion of RAM. In the stack, the contents of only those registers are saved, which are needed in the later part of the program.
   The stack pointer (SP) controls the addressing of the stack. The Stack Pointer contains the address of the top element of data stored in the stack.
5. **Instruction Register:** The instruction register holds the opcode (operation code or instruction code) of the instruction which is being decoded and executed.
6. **Temporary Register:** It is an 8-bit register associated with the ALU. It holds data during an arithmetic/logical operation. It is used by the microprocessor. It is not accessible to programmer.
7. **Flags:** The Intel 8085 microprocessor contains five flip-flops to serve as a status flags. The flip-flops are reset or set according to the conditions which arise during an arithmetic or logical operation. The five status flags of Intel 8085 are: Carry Flag (CS), Parity Flag (P), Auxiliary Carry Flag (AC), Zero Flag(Z) Sign Flag(S). If a flip-flop for a particular flag is set, then it indicates 1. When it is reset, it indicates 0.

Data and Address Bus

- The Intel 8085 is an 8-bit microprocessor. Its **data bus** is 8-bit wide and therefore, 8 bits of data can be transmitted in parallel from or to the microprocessor.
- The Intel 8085 requires an **address bus** of 16-bit wide as the memory addresses are of 16-bits.
- The 8 most significant bits of the address are transmitted by the address bus, A-bus (pins A_8 ? A_{15}).
- The 8 least significant bits of the address are transmitted by data/address bus, AD-bus (pins AD_0 ? AD_7).

Interrupt control : Whenever a microprocessor is executing a main program and if suddenly an interrupt occurs, the microprocessor shifts the control from the main program to process the incoming request. After the request is completed, the control goes back to the main program. There are 5 interrupt signals in 8085 microprocessors: INTR, TRAP, RST 7.5, RST 6.5, RST 5.5

Serial Input/output control : It controls the serial data communication by using Serial input data and Serial output data.

8085 microprocessor	8086 microprocessor
The data bus is of 8 bits.	The data bus is of 16 bits.
The address bus is of 16 bits.	The address bus is of 20 bits.
The memory capacity is 64 KB. Also 8085 Can Perform Operation upto 2^8 i.e. 256 numbers. A number greater than this is to taken multiple times in 8 bit data bus.	The memory capacity is 1 MB. Also 8086 Can Perform Operation upto 2^{16} i.e. 65,536 numbers.
The input/output port addresses are of 8 bits.	The input/output port addresses are of 8 bits.
The operating frequency is 3.2 MHz.	The operating frequency is 5 MHz, 8MHZ,10MHZ.
8085 MP has Single Mode of Operation.	8086 MP has Two Modes Of Operation. 1. Minimum Mode = Single CPU PROCESSOR 2. Maximum Mode = Multiple CPU PROCESSOR.
It not have multiplication and division instructions.	It have multiplication and division instructions.
It does not support pipe-lining.	It supports pipe-lining as it has two independent units Execution Unit (EU) and Bus Interface Unit (BIU).
It does not support instruction queue.	It supports instruction queue.
Memory space is not segmented.	Memory space is segmented.
It consists of 5 flags(Sign Flag, Zero Flag, Auxiliary Carry Flag, Parity Flag, Carry Flag).	It consists of 9 flags(Overflow Flag, Direction Flag, Interrupt Flag, Trap Flag, Sign Flag, Zero Flag, Auxiliary Carry Flag, Parity Flag, Carry Flag)

8086 MICROPROCESSORS

Register Organization of 8086

- 8086 has a powerful set of registers containing general purpose and special purpose registers.
- All the registers of 8086 are 16-bit registers.
- The general purpose registers, can be used either 8-bit registers or 16-bit registers. The general purpose registers are either used for holding the data, variables and intermediate results temporarily or for other purpose like counter or for storing offset address for some particular addressing modes etc.
- The special purpose registers are used as segment registers, pointers, index registers or as offset storage registers for particular addressing modes.
- Fig shows register organization of 8086. We will categorize the register set into four groups as follows:

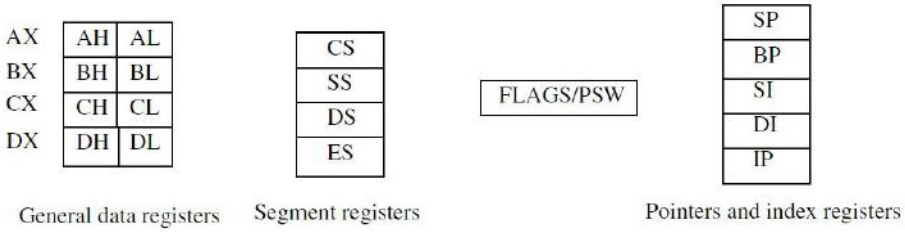


Fig.1.4 Register organization of 8086 Microprocessor

General Data Registers: The registers AX, BX, CX, and DX are the general 16-bit registers.

- **AX Register:** Accumulator register consists of two 8-bit registers AL and AH, which can be combined together and used as a 16- bit register AX. AL in this case contains the low-order byte of the word, and AH contains the high order byte. Accumulator can be used for I/O operations, rotate and string manipulation.

- **BX Register:** This register is mainly used as a base register. It holds the starting base location of a memory region within a data segment. It is used as offset storage for forming physical address in case of certain addressing mode.

- **CX Register:** It is used as default counter or count register in case of string and loop instructions.

- **DX Register:** Data register can be used as a port number in I/O operations and implicit operand or destination in case of few instructions. In integer 32-bit multiply and divide instruction the DX register contains high-order word of the initial or resulting number.

Segment Registers: To complete 1Mbyte memory is divided into 16 logical segments. The complete 1Mbyte memory segmentation is as shown in fig 1.5. Each segment contains 64Kbyte of memory. There are four segment registers.

- **Code segment (CS)** is a 16-bit register containing address of 64 KB segment with processor instructions. The processor uses CS segment for all accesses to instructions referenced by instruction pointer (IP) register. CS register cannot be changed directly. The CS register is automatically updated during far jump, far call and far return instructions. It is used for addressing a memory location in the code segment of the memory, where the executable program is stored.

- **Stack segment (SS)** is a 16-bit register containing address of 64KB segment with program stack. By default, the processor assumes that all data referenced by the stack pointer (SP) and base pointer (BP) registers is located in the stack segment. SS register can be changed directly using POP instruction. It is used for addressing stack segment of memory. The stack segment is that segment of memory, which is used to store stack data.

- **Data segment (DS)** is a 16-bit register containing address of 64KB segment with program data. By default, the processor assumes that all data referenced by general registers (AX, BX, CX, DX) and index register (SI, DI) is located in the data segment. DS register can be changed directly using POP and LDS instructions. It points to the data segment memory where the data is resided.

- **Extra segment (ES)** is a 16-bit register containing address of 64KB segment, usually with program data. By default, the processor assumes that the DI register references the ES segment in string manipulation instructions. ES register can be changed directly using POP and LES instructions. It also refers to segment which essentially is another data segment of the memory. It also contains data.

Pointers and Index Registers: The pointers contain within the particular segments. The pointers IP, BP, SP usually contain offsets within the code, data and stack segments respectively

- **Stack Pointer (SP)** is a 16-bit register pointing to program stack in stack segment.

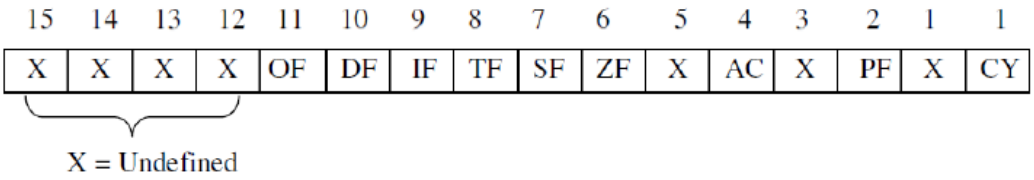
- **Base Pointer (BP)** is a 16-bit register pointing to data in stack segment. BP register is usually used for based, based indexed or register indirect addressing.

- **Source Index (SI)** is a 16-bit register. SI is used for indexed, based indexed and register indirect addressing, as well as a source data addresses in string manipulation instructions.

- **Destination Index (DI)** is a 16-bit register. DI is used for indexed, based indexed and register indirect addressing, as well as a destination data address in string manipulation instructions

Flag Registers: Flags Register determines the current state of the processor.
They are modified automatically by CPU after mathematical operations, this allows to determine the type of the result, and to determine conditions to transfer control to other parts of the program. The 8086 flag

register as shown in the fig, 9 active flags and they are divided into two categories: 1. Conditional Flags 2. Control Flags



Conditional flags are as follows:

- **Carry Flag (CY):** This flag indicates an overflow condition for unsigned integer arithmetic. It is also used in multiple-precision arithmetic.

- **Auxiliary Flag (AC):** If an operation performed in ALU generates a carry/barrow from lower nibble (i.e. D0 – D3) to upper nibble (i.e. D4 – D7), the AC flag is set i.e. carry given by D3 bit to D4 is AC flag. This is not a general-purpose flag; it is used internally by the Processor to perform Binary to BCD conversion.

- **Parity Flag (PF):** This flag is used to indicate the parity of result. If lower order 8-bits of the result contains even number of 1's, the Parity Flag is set and for odd number of 1's, the Parity flag is reset.

- **Zero Flag (ZF):** It is set; if the result of arithmetic or logical operation is zero else it is reset.

- **Sign Flag (SF):** In sign magnitude format the sign of number is indicated by MSB bit. If the result of operation is negative, sign flag is set.

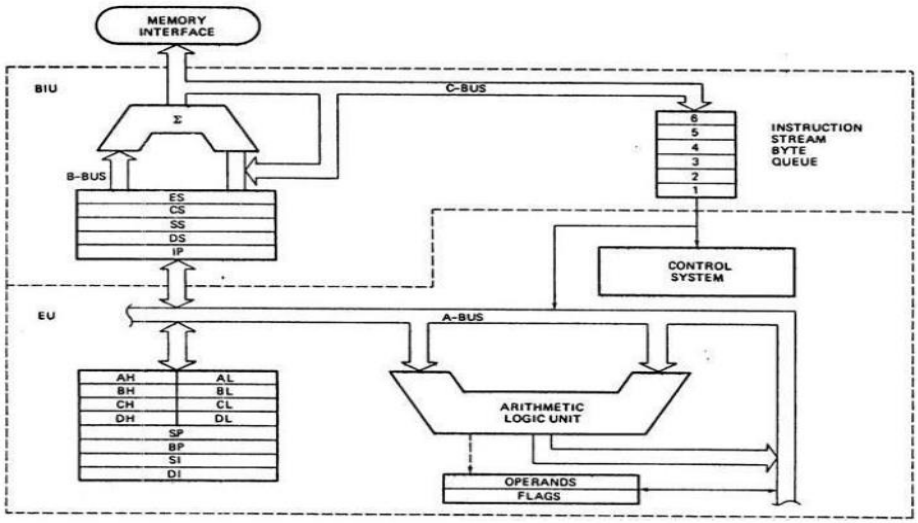
Control Flags: Control flags are set or reset deliberately to control the operations of the execution unit. Control flags are as follows:

- **Trap Flag (TF):** It is used for single step control. It allows user to execute one instruction of a program at a time for debugging. When trap flag is set, program can be run in single step mode.

- **Interrupt Flag (IF):** It is an interrupt enable/disable flag. If it is set, the maskable interrupt of 8086 is enabled and if it is reset, the interrupt is disabled. It can be set by executing instruction sit and can be cleared by executing CLI instruction.

- **Direction Flag (DF):** It is used in string operation. If it is set, string bytes are accessed from higher memory address to lower memory address. When it is reset, the string bytes are accessed from lower memory address to higher memory address.

8086 Architecture



- The 8086 is mainly divided into mainly two blocks
    1. Execution Unit (EU)
    2. Bus interface Unit (BIU)

- Dividing the work between these two will speed up the processing

Bus Interface Unit:

It provides the interface of 8086 to external memory and I/O devices via the System Bus. It performs various machine cycles such as memory read, I/O read etc. to transfer data between memory and I/O devices.

BIU performs the following functions-

- It generates the 20 bit physical address for memory access.
- It fetches instructions from the memory.
- It transfers data to and from the memory and I/O.
- Maintains the 6 byte prefetch instruction queue (**supports pipelining**).

BIU mainly contains the **4 Segment registers**, the **Instruction Pointer**, an **Instruction Queue** and an **Address Generation Circuit**.

Instruction Pointer (IP):

- It is a 16 bit register. It holds offset of the next instructions in the Code Segment.
- IP is incremented after every instruction byte is fetched.
- IP gets a new value whenever a branch instruction occurs.
- CS is multiplied by 10H to give the 20 bit physical address of the Code Segment.
- Address of the next instruction is calculated as CS x 10H + IP.

Address Generation Circuit:

- The BIU has a Physical Address Generation Circuit.
- It generates the 20 bit physical address using Segment and Offset addresses using the formula:
  Physical Address = Segment Address x 10H + Offset Address

Instruction Queue:

- It is a 6 byte queue (FIFO).
- Fetching the next instruction (by BIU from CS) while executing the current instruction is called pipelining.
- Gets flushed whenever a branch instruction occurs.

Segment Registers:

- **Code Segment register:** CS holds the base address for the Code Segment. All programs are stored in the Code Segment and accessed via the IP.
- **Data Segment registers:** DS holds the base address for the Data Segment.
- **Stack Segment register:** SS holds the base address for the Stack Segment.
- **Extra Segment register:** ES holds the base address for the Extra Segment.

The Execution Unit (EU): The main components of the EU are General purpose registers, the ALU, Special purpose registers, Instruction Register and Instruction Decoder and the Flag/Status Register.

1. Fetches instructions from the Queue in BIU, decodes and executes arithmetic and logic operations using the ALU.
2. Sends control signals for internal data transfer operations within the microprocessor.
3. Sends request signals to the BIU to access the external module.
4. It operates with respect to T-states (clock cycles) and not machine cycles.

- **Instruction Register and Instruction Decoder:** The EU fetches an opcode from the queue into the instruction register. The instruction decoder decodes it and sends the information to the control circuit for execution.

- **Arithmetic Logic Unit (16 bit):** Performs **8 and 16 bit** arithmetic and logic operations.

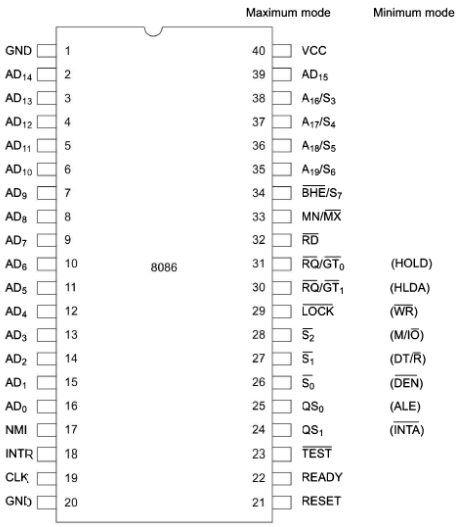
- **Flag Registers:** the 16 bit flag register reflects the result of ALU.

- **Timing and control unit:** it derives necessary control signals to execute instruction opcode received from queue depending upon information available in decoding unit.

Memory Segmentation:

- The memory in 8086 is organized as segments. In this scheme, the complete physical memory is divided into number of logical segments.
- Each segment is 64 K Bytes in size and is addressed by one of the segment registers.
- The CPU of 8086 is able to address 1 MB of memory.
- The complete 1 MB is divided into 16 segments each of 64 K bytes.
- The address of segments may be addressed as 0000H to F000H
- The offset address values are from 0000H to FFFFH so that physical address ranges from 00000H to FFFFFH. In the above case , segments is called non – overlapping segments.
- Suppose a segment start at particular address and its maximum size is 64 K bytes. But, if another segment starts before this 64 K bytes location of first segment, then the segment is said to be overlapping segment.
- The advantages include:
    - Allow the memory capacity to be 1 Mb although actual address to be handled are of 16 bits.
    - Allow the placing of code, data and stack portion of same program in different segments of memory for data and code protection.
    - Permits a program and/ or its data to be put into different areas of memory each time it is executed i.e. provision for relocation is done.

PIN OUT DIAGRAM OF 8086



- 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.
- 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.

Pins	Description
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 is await state.
A19/S6,A18/S5, A17/S4,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.
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). t goes low for the data transfers over D15-D8 and is used to derive chip selects of odd address memory bank or peripherals.
RD	Read signal, when low, indicates the peripherals that the processor is performing a memory or I/O read operation.
READY	This is the acknowledgement from the slow devices or memory that they have completed the data transfer.
INTR	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 request is pending, the processor enters the interrupt acknowledge cycle.
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.
NMI	This is an edge-triggered input which causes a Type2 interrupt. The NMI is not maskable internally by software. A transition from low to high initiates the interrupt response at the end of the current instruction.
RESET	This input causes the processor to terminate the current activity and start execution from FFFF0H
CLK	The clock input provides the basic timing for processor operation and bus control activity.
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.
M/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.
INTA	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.
ALE	This output signal indicates the availability of the valid address on the address/data lines and is connected to latch enable input of latches.
DT/$\bar{R}$: (Data transmit/Receive)	This output is used to decide the direction of data flow through the transreceivers (bidirectional buffers).
$\overline{\text{DEN}}$ (Data Enable)	This signal indicates the availability of valid data over the address/data lines. It is used to enable the transreceivers (bidirectional buffers) to separate the data from the multiplexed address/data signal
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.

S2, S1, S0 - Status Lines	These are the status lines which reflect the type of operation, being carried out by the processor.
$\overline{LOCK}$:	This output pin indicates that other system bus masters will be prevented from gaining the system bus, while the signal is low.
QS1, QS0-Queue Status	These lines give information about the status of the code prefetch queue. These are active during the CLK cycle after which the queue operation is performed
$\overline{RQ}/\overline{GT_0}$ (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.

$\overline{S}_2$	$\overline{S}_1$	$\overline{S}_0$	*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

QS_1	QS_0	*Indication*
0	0	No operation
0	1	First byte of opcode from the queue
1	0	Empty queue
1	1	Subsequent byte from the queue

PHYSICAL MEMORY ORGANIZATION OF 8086

In an 8086 based system, the 1Mbytes memory is physically organised as an odd bank and an even bank, each of 512 Kbytes, addressed in parallel by the processor. Byte data with an even address is transferred on D_7-D_0, while the byte data with an odd address is transferred on $D_{15}-D_8$ buss lines. The processor provides two enable signals, $\overline{BHE}$ and A_0 for selection of either even or odd or both the banks. The instruction stream is fetched from memory as words and is addressed internally by the processor as necessary. In other words, if the processor fetches a word (consecutive two bytes) from memory, there are different possibilities, like:

1. Both the bytes may be data operands
2. Both the bytes may contain opcode bits
3. One of the bytes may be opcode while the other may be data

- All of the above operations are handled by the internal decoder circuit.
- 8086 is a 16 bit microprocessor so it can access two bytes of data in one memory or I/O read or write operation. But commercially available chips are only one byte size.
- A map of memory 8086 system starts from 00000H and ends at FFFFFH.

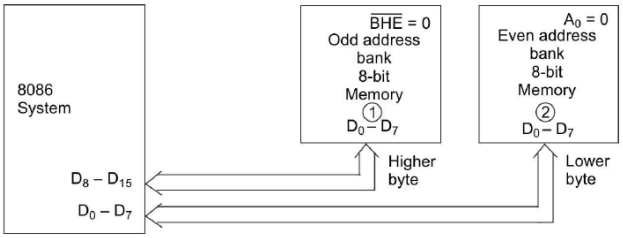


Fig. 1.7 *Physical Memory Organisation*

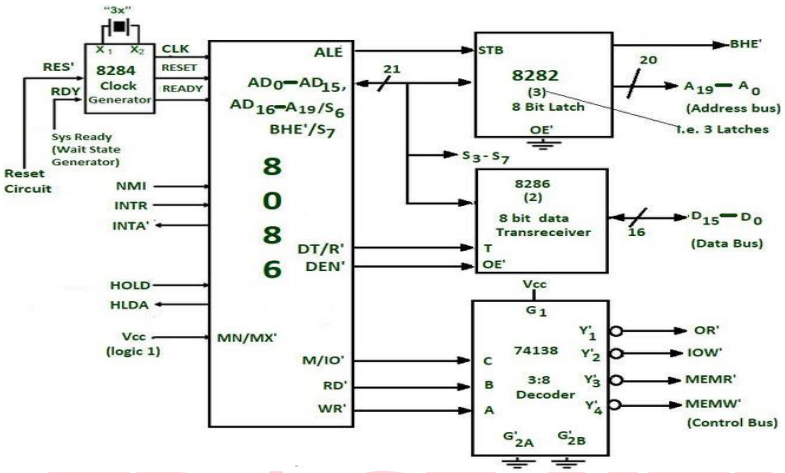
- Thus bits D0 – D7 of a 16 bit data will be transferred over D0 – D7 bus of 16 bit microprocessor to/ from 8 bit memory (1) and D8 – D15 will be transferred over D8 – D15 of 16 bit microprocessor to/ from memory (2).
- The lower byte is stored at 00000H and is transferred over D0 – D7, so 00000H must be in memory 2
- Higher byte is stored in 00001H and is transferred over D8 – D15, so 00001H must be in memory 1.
- All lower bytes are stored in even memory bank (memory bank 2) and all higher bytes are in odd memory bank (memory bank 1).
- Thus, complete memory map of 8086 is divided into odd and even memory banks.
- If 8086 transfers a 16-bit data to/ from memory both these banks must be selected for operation.
- Two signals A0 and $\overline{BHE}$ solve the problem of selection of memory banks.
- Certain locations are reserved for specific CPU purposes.

- The locations from FFFF0H to FFFFFH are reserved for operations including jump to initialization program and I/O processor initialization.
- The locations 00000H to 003FFH is reserved for interrupt vector table.

MINIMUM MODE OF OPERATION IN 8086

- The 8086 microprocessor operates in minimum mode when MN/MX' = 1.
- In minimum mode,8086 is the only processor in the system which provides all the control signals which are needed for memory operations and I/O interfacing.
- Here the circuit is simple but it does not support multiprocessing.

- The address bus of 8086 is 20 bits long. By this we can access 220 bytes memory i.e. 1MB. Out of 20 bits, 16 bits A0 to A15(or 16 lines) are multiplexed with a data bus.



8282 (8 bits) latches: The latches are buffered D FF. They are used to separate the valid address from the multiplexed Address/data bus by using the control signal ALE, which is connected to strobe (STB) of 8282. The ALE is active high signal. Here three such latches are required because the address is 20 bits.

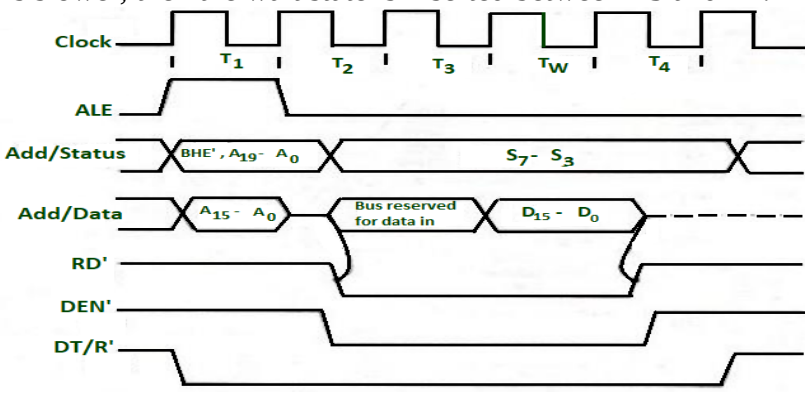
8286 (8 bits) transceivers: They are bidirectional buffers and also known as data amplifiers. They are used to separate the valid data from multiplexed add/data bus. Two such transceivers are needed because the data bus is 16 bits long. 8286 is connected to DT/R' and DEN' signals. They are enabled through the DEN signal. The direction of data on the data bus is controlled by the DT/R' signal. DT/R' is connected to T and DEN' is connected to OE'.

- 8284 clock generator is used to provide the clock.
- M/IO'= 1, then I/O transfer is performed over the bus. and when M/IO' = 0, then I/O operation is performed.
- The signals RD' and write WR' are used to identify whether a read bus cycle or a write bus cycle is performing. When WR' = 0, then it indicates that valid output data on the data bus.
- RD' indicates that the 8086 is performing a read data or instruction fetch process is occurring .During read operations, one other control signal is also used, which is DEN (data enable) and it indicates the external devices when they should put data on the bus.
- Control signals for all operations are generated by decoding M/IO', RD', WR'. They are decoded by 74138 3:8 decoder.

<u>Timing diagram: The working of min mode can be easily understood by timing diagrams.</u>

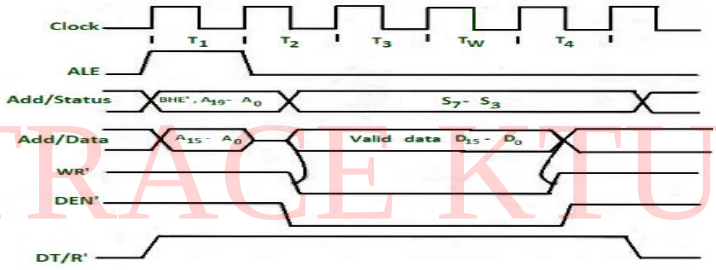
- All processors bus cycle is of at least 4 T-states (T1, T2, T3, T4). The address is given by processor in the T1 state. It is available on the bus for one T-state.
- In T2, the bus is tristated for changing the direction of the bus (in the case of a data read cycle.)
- The data transfer takes place between T3 and T4.

- If the addressed device is slower, then the wait state is inserted between T3 and T4.



Read Memory cycle

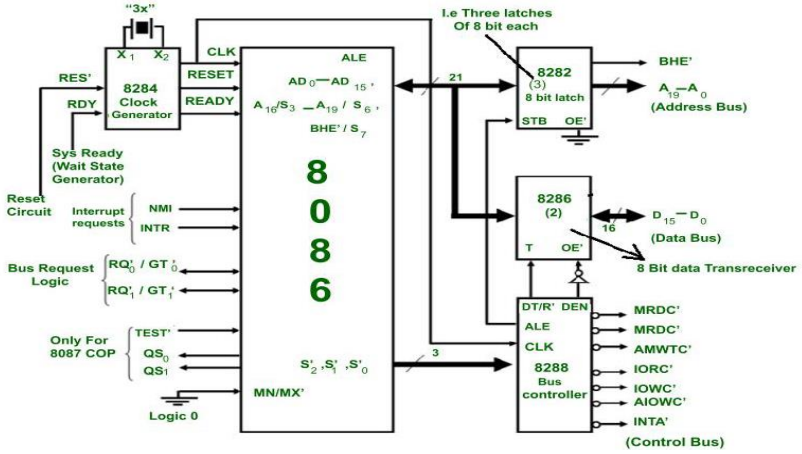
- At T1 state ALE =1 ,this indicates that a valid address is latched on the address bus and also M / IO'= 1, which indicates the memory operation is in progress.
- In T2, the address is removed from the local bus and is sent to the addressed device. Then the bus is tristated.
- When RD' = 0, the valid data is present on the data bus.
- During T2 DEN' =0, which enables transceivers and DT/R' = 0, which indicates that the data is received.
- During T3, data is put on the data bus and the processor reads it.
- The output device makes the READY line high. This means the output device has performed the data transfer process. When the processor makes the read signal to 1, then the output device will again tristate its bus drivers.



Memory Write cycle

MAXIMUM MODE OPERATION OF 8086

- In this we can connect more processors to 8086 (8087/8089)
- 8086 max mode is basically for implementation of allocation of global resources and passing bus control to other co – processor (i.e. second processor in the system), because two processors cannot access system bus at same instant.
- All processors execute their own program.
- The resources which are common to all processors are known as global resources.
- The resources which are allocated to a particular processor are known as local or private resources.



- When MN/ MX' = 0, 8086 works in max mode.

- Clock is provided by 8284 clock generator.
- 8288 bus controller- Address form the address bus is latched into 8282 8-bit latch. Three such latches are required because address bus is 20 bits. The ALE (Address latch enable) is connected to STB(Strobe) of the latch. The ALE for latch is given by 8288 bus controller.
- The data bus is operated through 8286 8-bit transceiver. Two such transceivers are required, because data bus is 16-bit. The transceivers are enabled the DEN signal, while the direction of data is controlled by the DT/R signal. DEN is connected to OE' and DT/ R' is connected to T. Both DEN and DT/ R' are given by 8288 bus controller.
- Control signals for all operations are generated by decoding S'2, S'1 and S'0 using 8288 bus controller.
- Bus request is done using RQ' / GT' lines interfaced with 8086. RQ0/GT0 has more priority than RQ1/GT1.
- INTA' is given by 8288, in response to an interrupt on INTR line of 8086.
- In max mode, the advanced write signals get enabled one T-state in advance as compared to normal write signals. This gives slower devices more time to get ready to accept the data, therefore it reduces the number of cycles.

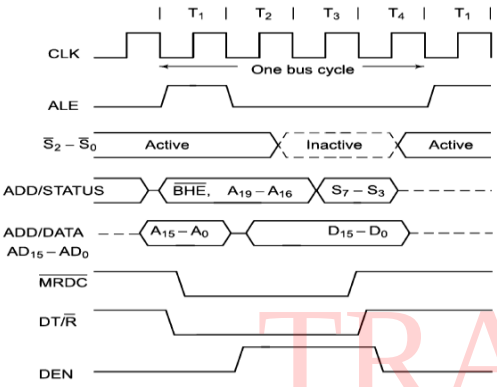


Fig. 1.16 (a) *Memory Read Timing in Maximum Mode*

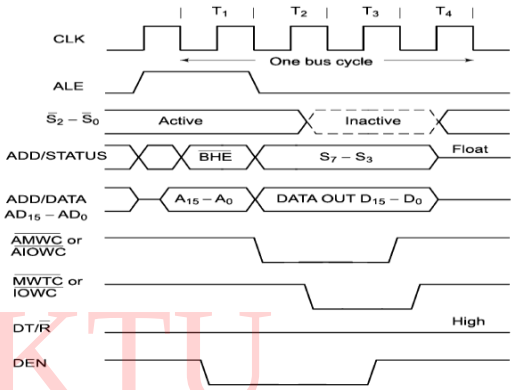


Fig. 1.16(b) *Memory Write Timing in Maximum Mode*

DIFFERENCE BETWEEN 8086 AND 8088 MICROPROCESSORS

Sl no	8086	8088
1	The data bus is of 16 bits.	The data bus is of 8 bits.
2	It has 3 available clock speeds (5 MHz, 8 MHz & 10 MHz)	It has 2 available clock speeds (5 MHz, 8 MHz)
3	The memory capacity is 512 kB.	The memory capacity is implemented as a single 1 MX 8 memory banks.
4	It has memory control pin (M/IO) signal.	It has complemented memory control pin (IO/M) signal of 8086.
5	It has Bank High Enable (BHE) signal.	It has Status Signal (SSO).
6	It can read or write either 8-bit or 16-bit word at the same time.	It can read only 8-bit word at the same time.
7	Input/Output voltage level is measured at 2.5 mA.	Input/Output voltage level is measured at 2.0 mA
8	It has 6 byte instruction queue.	It has 4 byte instruction queue as it can fetch only 1 byte at a time.
9	It draws a maximum supply current of 360 mA.	It draws a maximum supply current of 340 mA.

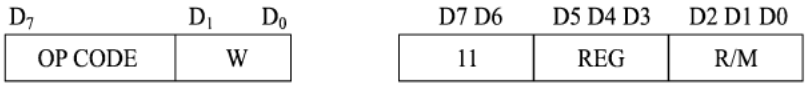
MACHINE LANGUAGE INSTRUCTION FORMAT

- A Machine language instruction format has one or more fields associated with it.

- The first field is called operation code field or opcode field, which indicates the type of operation to be performed by the CPU.
- It also contains other fields known as operand fields.
- The CPU executes the instruction using the information available in instruction fields.
- There are 6 general format of instruction fields in 8086. The length vary from 1 byte to 6 bytes.

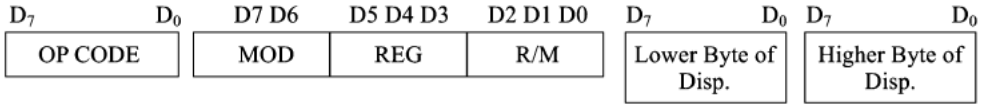
1. One byte Instruction This format is only one byte long and may have the implied data or register operands. The least significant 3-bits of the opcode are used for specifying the register operand, if any. Otherwise, all the 8-bits form an opcode and the operands are implied.

2. Register to Register This format is 2 bytes long. The first byte of the code specifies the operation code and width of the operand specified by w bit. The second byte of the code shows the register operands and R/M field, as shown below.



3. Register to/from Memory with no Displacement This format is also 2 bytes long and similar to the register to register format except for the MOD field as shown.

4. Register to/from Memory with Displacement This type of instruction format contains one or two additional bytes for displacement along with 2-byte the format of the register to/from memory without displacement. The format is as shown below.



5. Immediate Operand to Register In this format, the first byte as well as the 3-bits from the second byte which are used for REG field in case of register to register format are used for opcode. It also contains one or two bytes of immediate data. The complete instruction format is as shown below.

6. Immediate Operand to Memory with 16-bit Displacement This type of instruction format requires 5 or 6 bytes for coding. The first 2 bytes contain the information regarding OPCODE, MOD, and R/M fields. The remaining 4 bytes contain 2 bytes of displacement and 2 bytes of data as shown.