MODULE 1

BASICS 8086 ARCHITECTURE

- **The microprocessor is a programmable integrated device that has computing and decision making capability**
- A microprocessor is an IC chip which can act as CPU of a digital computer
- The microprocessor communicates and operates in the binary numbers 0 and 1
- Microprocessor reads binary instruction from a storage device called memory,
- Accepts binary data as input and processes data according to those instruction, and provides results as output.

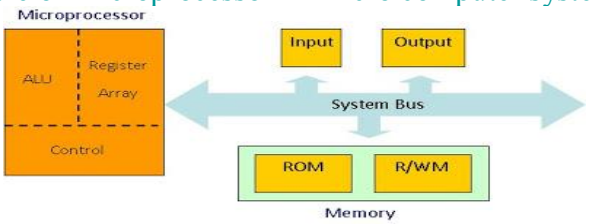
Binary Digits

- The microprocessor operates in binary digits 0 and 1 known as bits
- These digits 0 and 1 are also synonymous with low and high voltages
- Each microprocessor recognizes and processes a group of bits called the word
- Microprocessors are classified according to their word length.
  (for example a processor with an 8-bit word is known as an 8 bit microprocessor, and a processor with a 32-bit word is known as a 32-bit microprocessor)

Evolution of Intel Microprocessor

- In 1971, the first microprocessor Intel 4004 was invented,.It has word length of 4 bits
- Then later 8 bit microprocessor was developed by Intel in 1972 and it was named as 8008(It got lot of performance issues)
- The upgraded version of 8008 ,8080 was introduced in 1974 (Its main drawback was this that it required three power supplies)
- The modified version of 8080 is 8085.It has a word length of 8 bits.
- In 1979, intel developed microprocessor 8086 which is a 16 bit microprocessor. After the development of 8086 a number of microprocessors have been developed by various companies such as 8088, 80186, 80188, 80386, 80486, 80487, Pentium 1 to Pentium 5 etc.

Role of Microprocessor in Micro computer system



- **Memory:** Memory is a medium that stores binary information called instructions and data.
- Provides the instructions and data to microprocessor.
- Stores results and data for the microprocessor.

Input/Output Device: An input device is the one which transfers information from outside world to the computer.

- Examples:Keyboard,mouse

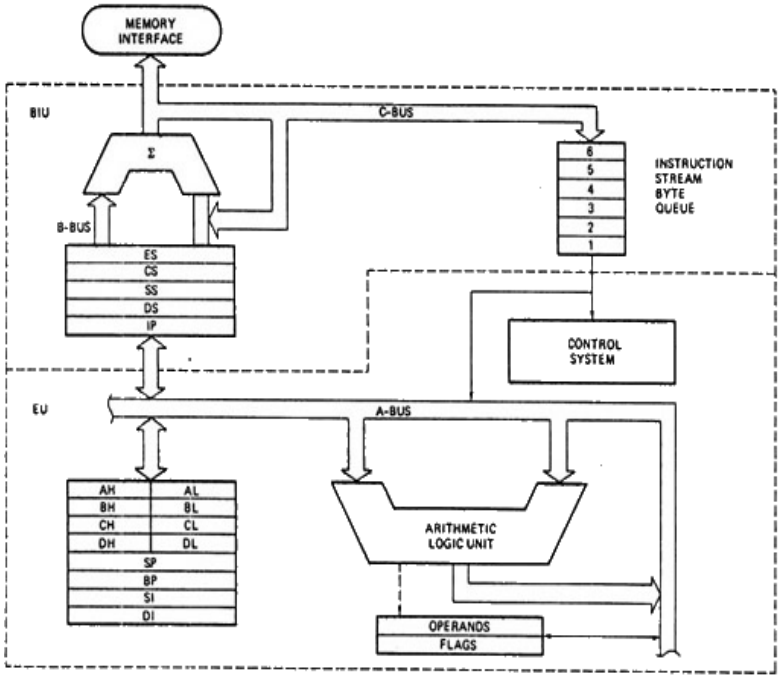
- An output device is the one which transfers information from the computer to the outside world.
  Printer,Monitor,Speaker,Plotter etc

- **MICROPROCESSOR:** A Miroprocessor is a semiconductor electronic clock driven device manufactured by using LSI/VLSI technique
- The microprocessor unit reads instructions from the memory a perform the tasks specified.
- It communicates with input/output devices either to accept or it send data.
- The timing of the communication process is controlled by the group of circuits called the control unit.

8086-Details

- It is a 16 bit microprocessor.
- It operates on +5 volts .
- It is available in 40 pin DIP
- It provides a 16-bit ALU.
- The size of address bus is 20-Bit Bus, i e it can access up to 1 MB of Memory-(2^{20} =1,048,576 bytes)
- It supports two operating modes- maximum mode and minimum mode
- It operates in 5 MHz/8MHz/10MHz/12MHz clock frequency
- Memory is divided into two banks-even bank and odd bank.
- Address range of memory is from 0000H to FFFFH.

ARCHITECTURE of 8086



8086 Microprocessor is divided into two units. They are

- **Bus Interface Unit (BIU)**
- **Execution unit (EU)**

Bus interface unit(BIU)

- The functions of BIU is that it fetches instructions
- Reads data from memory or I/O ports

- Write data to memory or I/O ports and buffers them in a queue
- Queue is 6 bytes long, it is referred to as instruction pipe line
- BIU fills this space with instructions waiting for execution by the EU
- **The BIU fetches and stores instructions in the queue while the EU executes present instructions, This is called pipelining.**

Register Organisation of 8086

The intel 8086 contains fourteen 16-bit registers. They are grouped as :-
1) General Purpose Registers
2) Pointer & Index Registers
3) Segment Registers
4) Instruction Pointer and Flags

1.General Purpose Register(GPRs):

- Intel 8086 has four 16 bit GPRs ,Each is further divided in to two 8-bit registers (higher order and lower order)
  1) Accumulator AX- AH,AL
  2) Base Register BX-BH,BL
  3) Counter Register CX-CH,CL
  4) Data Register DX-DH,DL
- They are also used as special purpose registers

2.Pointer & Index Register

- They are used for computing the memory address ,These registers are 4 namely

1) Stack Pointer (SP)
2) Base Pointer (BP)
3) Source Index (SI)
4) Destination Index (DI)

3.Segment Registers: There are 4 namely

1) Code Segment Register (CS)-Used to store instructions of a program
2) Data Segment Register (DS)-Store program data,variables & constants
3) Stack Segment Register (SS)-If stack is used then SS is used
4) Extra Segment Register (ES)-Stores some data strings

4.Instruction Pointer (IP) & Flag

- The IP points to the next instruction in current code segment.
- 8086 has 9 flag which are subdivided in to status & control flags.

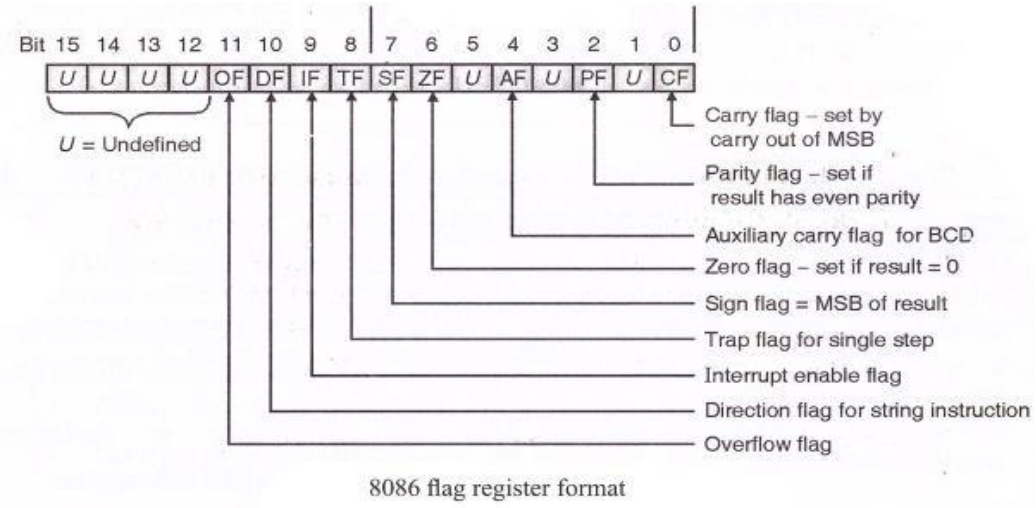
▶ **ALU:** It performs 16 bit or 8bit arithmetic operations like addition, subtraction ,multiplication, division, data conversion and logical operations like NOT,OR,AND,XOR.

▶ It also performs register increment, decrement and shift operations.

Execution Unit: The functions of execution unit are:

1) To tell BIU where to fetch the instructions or data from.
2) To decode the instructions.
3) To execute the instructions.
4) EU has 16-bit ALU, which can perform arithmetic and logical operations on 8-bit as well as 16-bit.
5) EU contains he GPRs, pointer and index registers and flags.

Flag register (Status register)



8086 flag register format

Conditional flags/Status flags:

There are six conditional flags. These flags are set or reset by the EU on the basis of the result of the arithmetic and logical operation.

They are :-

i.Carry flag – This flag is set,if there is a carry in case of addition or borrow in case of subtraction

ii.**Parity flag(Pf)** –This flag is set,if the result contains even number of 1s

iii**. Auxiliary Carry Flag**:This flag is used for BCD operation.

iv.**Zero Flag**:This flag is set,if the result of arithmetic or logical operations is zero

v.**Sign flag:**it is set if arithmetic operation causes negative results.

vi.**Overflow flag** - This flag sets if an overflow occurs.

This flag sets to 1 if the result of a signed operation is too large to fit in the number of bits available to represent it.

Control Flags: There are 3 control flags in 8086 processor:-

1) Trap flag
2) Interrupt Enable flag
3) Direction flag

- Trap flag - it permits the operation of a processor in single-step mode
- Interrupt flag - if the flag is set maskable hardware interrupts will be enabled,if set to 0,such interrupts will be ignored.
- Direction flag –This is used by string manipulation instructions.
- *If this flag is set,the contents of index registers(SI and DI) will be decremented after each operation*

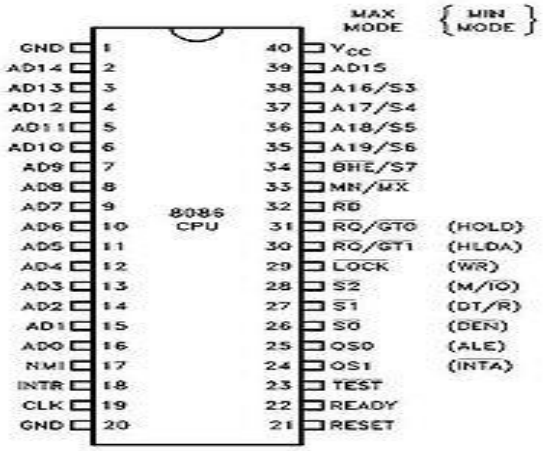
PIN Diagram of 8086(HARDWARE ARCHITECTURE)

- 8086 available in 40 pin DIP
- 8086 operates in minimum mode(single processor) or maximum mode (multi processor)
- Functions of some pins are different in minimum mode and maximum mode

Thus the 8086 signals are categorized in to 3 groups

i. The signals having common functions in both minimum and maximum mode
ii. The signals that have special functions for minimum mode

iii. The signals having special functions for maximum mode



1.Signals common for min and max mode

- The 8086 require +5V (*VCC at pin 40*)and one clock pulse(pin 19) for operation
- Pin number 1 and 20 are grounded
- 20 pins are used as address lines.
- In this 20 pins, 16 pins are used as both address and data lines.(*multiplexing facility reduces the number of pins*).

- **Address/data bus -(*Pin 2 to 16*)**

AD0-AD15: Bidirectional Address/data Bus: Thsese are lower order address bus multiplexed with data bus.

A16-A19/S3-S6:These are multiplexed a These are the 4 address/status buses.

- S7/BHE – (*Pin 34*)

BHE stands for Bus High Enable. It is used to indicate the transfer of data using data bus D8-D15

S_4	S_3	Indications
0	0	Alternate Data
0	1	Stack
1	0	Code or None
1	1	Data

- RD – (*Pin 32*)

It is used to read signal for Read operation

- Ready - (*pin 22*)

When it is high, it indicates that the device is ready to transfer data. When it is low, it indicates wait state

- RESET-(*pin 21*)

It is used to restart the execution. It causes the processor to immediately terminate its present activity

- INTR-(*pin 18*)

It is an interrupt request signal

- NMI-(*pin 17*)

It stands for non-maskable interrupt.

- MN/$\overline{MX}$ –(*pin 33*)

It stands for Minimum/Maximum. When it is high, it works in the minimum mode and vice-versa

- Minimum Mode Signals
- INTA-(*pin 24*)

It is an interrupt acknowledgement signal. When the microprocessor receives this signal, it acknowledges the interrupt

ALE - (*pin 25*)

It stands for address enable latch. This signal indicates the availability of a valid address on the address/data lines.

DEN-(*pin 26*)

It stands for Data Enable.It is used to enable transreceivers to seperate data from multiplexed AD Bus

- DT/$\bar{R}$-(*pin 27*)

It stands for Data Transmit/Receive signal. It decides the direction of data flow through the transreceiver. When it is high, data is transmitted out and vice-a-versa.

- M/$\overline{IO}$-(*pin 28*)

This signal is used to distinguish between memory and I/O operations. When it is high, it indicates I/O operation and when it is low indicates the memory operation.

- $\overline{WR}$-(*pin 29*)

It stands for write signal. It is used to write the data into the memory or the output device depending on the status of M/IO signal.

- HLDA-(*pin 30*)

It stands for Hold Acknowledgement signal . This signal acknowledges the HOLD signal.

- HOLD-(*pin 31*)

This signal indicates to the processor that external devices are requesting to access the address/data buses.

Maximum Mode Signals

- QS1 and QS0-(*pin 24 and 25*):These are queue status signals. These signals provide the status of instruction queue. Their conditions are shown in the following table

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

- $\overline{S0}$, $\overline{S1}$, $\overline{S2}$ –(*pin 26, 27, and 28*)

These are the status signals that provide the status of operation

S_2	S_1	S_0	Status
0	0	0	Interrupt acknowledgement
0	0	1	I/O Read
0	1	0	I/O Write
0	1	1	Halt
1	0	0	Opcode fetch
1	0	1	Memory read
1	1	0	Memory write
1	1	1	Passive

- $\overline{LOCK}$-(*pin 29*)

When this signal is active, it indicates to the other processors not to ask the CPU to leave the system bus.

- $\overline{RQ}/\overline{GT1}$ and $\overline{RQ}/\overline{GT0}$-(*pin 30 and pin 31*)

These are the Request/Grant signals used by the other processors requesting the CPU to release the system bus. When the signal is received by CPU, then it sends acknowledgment. $\overline{RQ}/\overline{GT0}$ has a higher priority than $\overline{RQ}/\overline{GT1}$.

- **Minimum Mode Configuration**:In minimum mode configuration,MN/MX is connected to Vcc.
- In minimum mode,the pins used are $\overline{INTA}$,ALE,$\overline{DEN}$,DT/$\overline{R}$,M/$\overline{IO}$,$\overline{WR}$,HOLD,HLDA.
- It contain a clock generator 8284 which generates clk pulses.
- Three latches 8282 are used to generate 20 bit address and $\overline{BHE}$

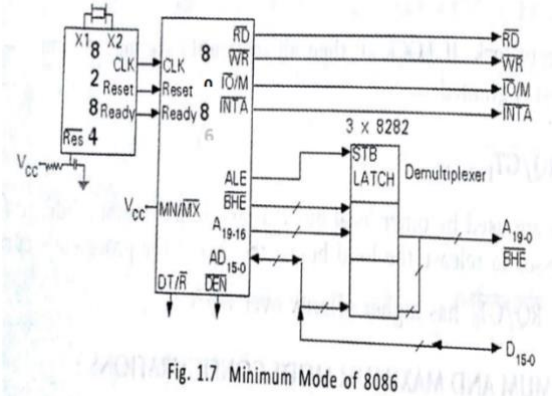


Fig. 1.7 Minimum Mode of 8086

- **MAXIMUM MODE CONFIGURATION:**In Maximum mode configuration,MN/$\overline{MX}$ is connected to gnd.
- The signals used are MN/$\overline{MX}$,QS1,QS0,$\overline{S0}$,$\overline{S1}$,$\overline{S2}$,LOCK,$\overline{RQ}/\overline{GT1}$,$\overline{RQ}/\overline{GT0}$ respectively

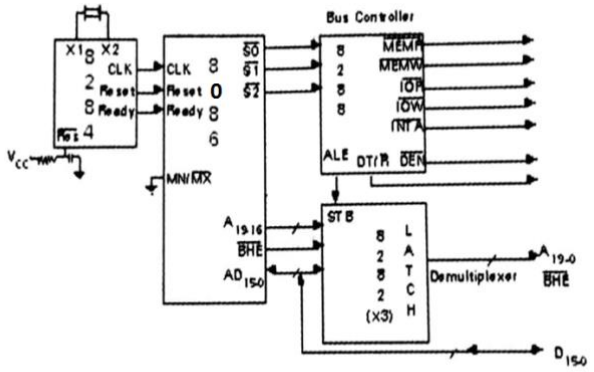


Fig. 1.8 Maximum Mode of 8086

Programming of 8086 Proccessor:

- An instruction has 2 parts.Op code and address.
- Op code tells the microprocessor what to do
- Address tells the microprocessor where to take action.

Addressing modes of 8086

I. Immediate addressing mode
II. Register addressing mode
III. Direct addressing mode

IV. Register indirect addressing mode
V. Based addressing with displacement
VI. Indexed addressing with displacement
VII. Based Indexed Addressing
VIII. Based Indexed Addressing with displacement
IX. I/O port Addressing
X. Implicit Addressing mode

1.Immediate operand addressing mode: In this addressing mode, immediate data is a part of the instruction itself is known as immediate addressing mode

Examples :-
i)MOV CX, 4929 H
ii)ADD AX, 2387 H
iii)MOV AL, FFH

2.Register addressing mode: In this mode, data is stored in a register

MOV CX, AX ; moves the contents of the 16-bit AX register into the 16-bit CX register)
ADD BX,DX

3.Direct addressing mode: In this addressing mode 16 bit m/m address is directly in the instruction

MOV AX, [1592H] : This instruction transfers contents of memory location 1592H into AX

Data resides in m/m location in DS whose address is calculated using 1592H as offset address and content of DS as segment address.

4.Register indirect addressing mode: In this addressing mode 16 bit EA (effective address) is provided by base register BX or index register SI OR DI

MOV AX, [BX] : Suppose the register BX contains offset 4895H,added to the data present in DS.

5.Based-addressing with displacement: In this addressing mode,16 bit EA is specified in two parts.One part is specified as the contents of BX or BP.The other part is specified in the instruction itself called displacement.

MOV AX, 1234H [BX] : The contents of m/m location whose EA is obtained by adding contents of BX and 16 bit displacement 1234H are transferred into AX.

6**. Indexed addressing with displacement**: In this addressing mode,16 bit EA is calculated by adding the contents of index registers SI or DI and displacement provided in the instruction itself

- MOV AX ,1234 H [SI].

7.Based Indexed Addressing: 16 bit EA is calculated by adding the contents of a base register(BX or BP) to the contents of an index register(SI or DI)

- MOV AX,[BX] [SI]

8. **Based Indexed Addressing with displacement**: EA is provided by adding the contents of base register(BX OR BP),the contents of an index register (SI or DI) and the displacement within the instruction.

- MOV AX,8000H [SI] [BX]

- Where 8000H is the displacement.

9. I/O Port Addressing: It is of 2 types:

A)Fixed I/O Port Addressing: In this mode, port address is specified with fixed port address.

- EX: IN AL,19H

B)Variable I/O Port Addressing: In this mode, the port address is specified indirectly in the instruction

- EX: IN AX,DX
- OUT DX,AX

10.Implicit Addressing Mode: In this mode instruction contains only the opcode. The instruction assumes that data is in predefined registers

Eg :- i) CMC-Complement Carry Flag

ii) HLT-Halt