16Bit Microprocessor : 8086

Features of 8086

- 8086 is a 16bit processor. It's ALU, internal registers works with 16bit binary word
- 8086 has a 16bit data bus. It can read or write data to a memory/port either 16bits or 8 bit at a time
- 8086 has a 20bit address bus which means, it can address upto 2^{20} = 1MB memory location
- Frequency range of 8086 is 6-10 MHz

Data Read/Write process from /To Memory

Word Read

- Each of 1 MB memory address of 8086 represents a byte wide location
- 16bit words will be stored in two consecutive Memory location

- If first byte of the data is stored at an **even address** , 8086 can read the entire word in one operation.
  - For example if the 16 bit data is stored at even address 00520$_H$ is 2607

    MOV BX, [00520]
    8086 reads the first byte and stores the data in BL and reads the 2nd byte and stores the data in BH

    BL← (00520)
    BH← (00521)

- If the first byte of the data is stored at an ODD address, 8086 needs two operation to read the 16 bit data
  - For example if the 16 bit data is stored at even address 00521$_H$ is F520

    MOV BX, [00521]
    In first operation , 8086 reads the 16 bit data from the 00520 location and stores the data of 00521 location in register BL and discards the data of 00520 location

In 2nd operation, 8086 reads the 16 bit data from the 00522 location and stores the data of 00522 location in register BH and discards the data of 00523 location

BL← (00521)
BH← (00522)

Byte Read:

MOV BH, [Addr]

For Even Address:

Ex: MOV BH, [00520]

8086 reads the first byte from 00520 location and stores the data in BH and reads the 2nd byte from the 00521 location and ignores it

BH ← [00520]

For Odd Address

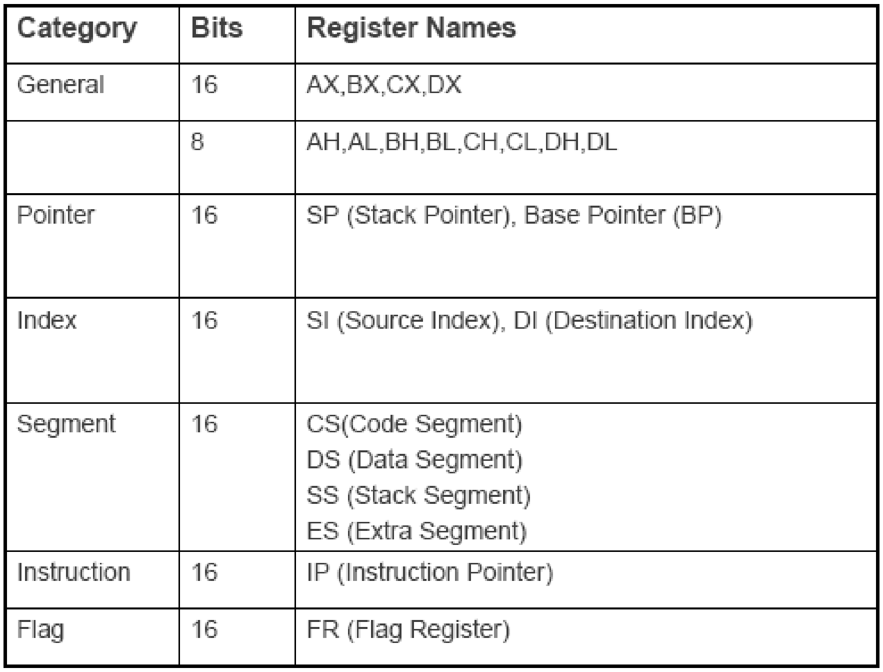
MOV BH, [Addr]

Ex: MOV BH, [00521]

8086 reads the first byte from 00520 location and ignores it and reads the 2nd byte from the 00521 location and stores the data in BH

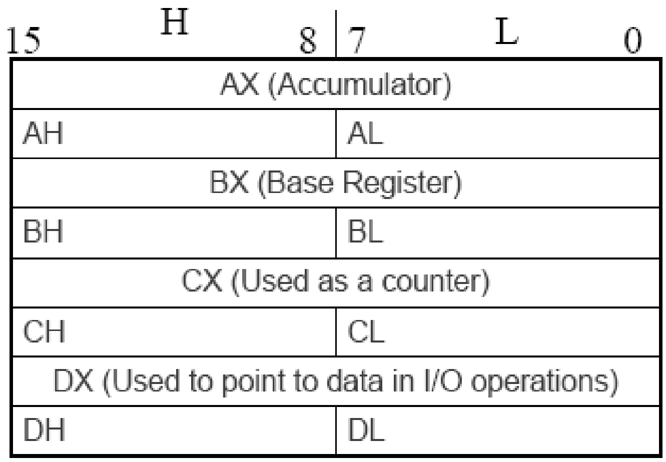
BH ← [00521]

Registers of 8086



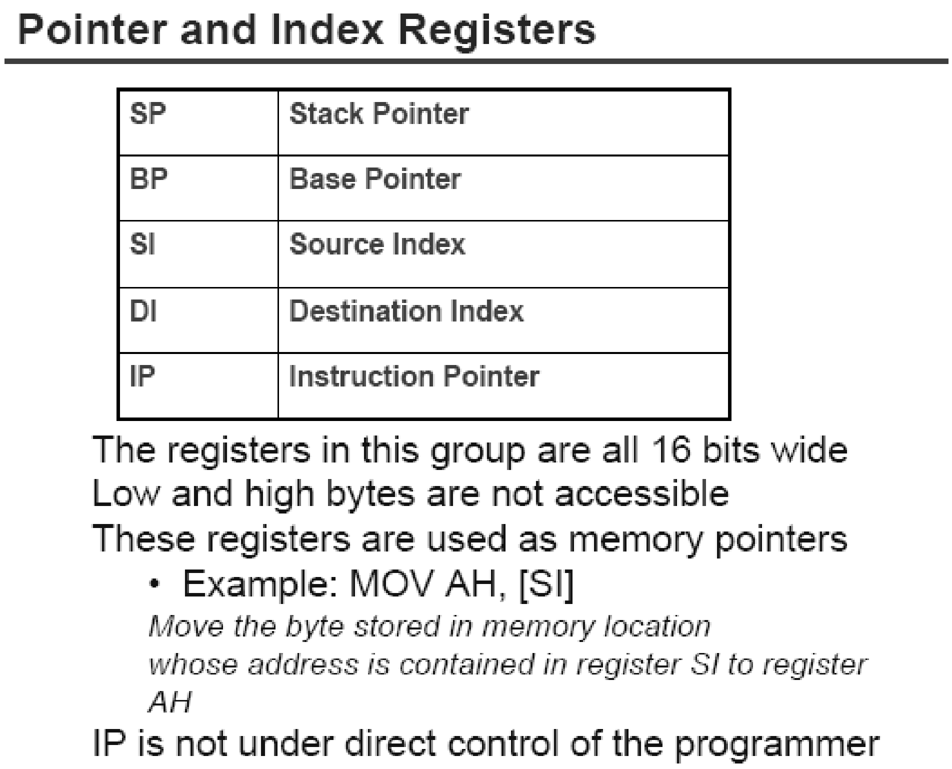
Category	Bits	Register Names
General	16	AX,BX,CX,DX
	8	AH,AL,BH,BL,CH,CL,DH,DL
Pointer	16	SP (Stack Pointer), Base Pointer (BP)
Index	16	SI (Source Index), DI (Destination Index)
Segment	16	CS(Code Segment) DS (Data Segment) SS (Stack Segment) ES (Extra Segment)
Instruction	16	IP (Instruction Pointer)
Flag	16	FR (Flag Register)

General Purpose Registers



15 H 8	7 L 0
AX (Accumulator)	
AH	AL
BX (Base Register)	
BH	BL
CX (Used as a counter)	
CH	CL
DX (Used to point to data in I/O operations)	
DH	DL

- **Data Registers** are normally used for storing temporary results that will be acted upon by subsequent instructions
- Each of the registers is 16 bits wide (AX, BX, CX, DX)
- General purpose registers can be accessed as either 16 or 8 bits e.g., AH: upper half of AX, AL: lower half of AX



Pointer and Index Registers

SP	Stack Pointer
BP	Base Pointer
SI	Source Index
DI	Destination Index
IP	Instruction Pointer

The registers in this group are all 16 bits wide

Low and high bytes are not accessible

These registers are used as memory pointers

- Example: MOV AH, [SI]

  Move the byte stored in memory location whose address is contained in register SI to register AH

IP is not under direct control of the programmer

Important 8086 Pin Diagram/Description

AD15±AD0

ADDRESS DATA BUS: These lines constitute the time multiplexed memory/IO address and data bus.

ALE

Address Latch Enable. A HIGH on this line causes the lower order 16bit address bus to be latched that stores the addresses and then, the lower order 16bit of the address bus can be used as data bus.

READY

READY is the acknowledgement from the addressed memory or I/O device that it will complete the data transfer.

INTR

INTERRUPT REQUEST: is a level triggered input which is sampled during the last clock cycle of each instruction to determine if the processor should enter into an interrupt acknowledge operation. A subroutine is vectored to via an interrupt vector lookup table located in system memory. It can be internally masked by software resetting the interrupt enable bit. INTR is internally synchronized. This signal is active HIGH.

$\overline{\text{INTA}}$

Interrupt Acknowledge from the MP

NMI
NON-MASKABLE INTERRUPT: an edge triggered input which causes an interrupt request to the MP. A subroutine is vectored to via an interrupt vector lookup table located in system memory. NMI is not maskable internally by software.

RESET: causes the processor to immediately terminate its present activity. The signal must be active HIGH for at least four clock cycles. It restarts execution

MN/$\overline{\textbf{MX}}$
MINIMUM/MAXIMUM: indicates what mode the processor is to operate in. The two modes are discussed in the following sections.

M/$\overline{\textbf{IO}}$: Differentiate between the Memory and I/O operation. A LOW on this pin indicated I/O operation and a HIGH indicated a Memory Operation

HOLD : The 8086 has a pin called HOLD. This pin is used by external devices to gain control of the busses.

HLDA :

When the HOLD signal is activated by an external device, the 8086 stops executing instructions and stops using the busses. This would allow external devices to control the information on the

8086 MINIMUM AND MAXIMUM MODES of operation

MN/$\overline{\text{MX}}$

- **Minimum mode** *The 8086 processor works in a single processor environment. All control signals for memory and I/O are generated by the microprocessor.*
- **Maximum mode** *is designed to be used when a coprocessor exists in the system.*
- *8086 works in a multiprocessor environment. Control signals for memory and I/O are generated by an external BUS Controller.*

Data Transfer Between CPU and the Memory

Memory Write:

Byte Transfer: move BYTEPTR ds : [SI], 37H

Word Transfer: move WORDPTR ds : [SI], 1237H

Memory Read:

Byte Transfer: move al, BYTEPTR ds : [SI]

Transfers data from the physical memory address calculated using ds and [SI] to register AL (Lower byte of AX Register)

Word Transfer: move ax, WORDPTR ds : [SI]

Transfers data from the physical memory address calculated using ds and [SI] to register AL (Lower byte of AX Register) and the next byte from the next memory location calculated as ds:[SI +1] is transferred to AH (Higher byte of AX Register)

Memory operation through ax Register

Write:
MOV AX , 1234H
MOV WORDPTR ds: [SI], ax

Ds: 0000H
SI: 0500H
Physical Address: 00000+0500= 00500 H

The instruction transfers
34 → 00500H
12 → 00501H

Read:

MOV ax, WORDPTR ds: [SI]

Ds: 0000H
SI: 0500H
Physical Address: 00000+0500= 00500 H

The instruction transfers
AL ← (00500)
AH ← (00501)

Data Transfer Between CPU and the Port

Port addresses in 8086 are assigned either 8bit port address or 16 bit address

For a Port with 8bit port address:

Read Operation:

IN Padr where Padr is the 8bit Port address
Ex: IN 20H
The instruction transfers data byte from the 8bit port address 20H to register AL

Write Operation:

OUT Padr where Padr is the 8bit Port address
Ex: OUT 20H
The instruction transfers data byte from AL to the 8bit port address 20H .

For a Port with 16bit port address:

DX register is used to hold the Port address

Read Operation:

Example:

```
Mov DX, 4000H
IN al, DX
```

The instruction transfers data byte from 16bit port address 4000H contained in DX register to AL.

Write Operation:

Example:

```
MOV AL, 10H
MOV DX, 4000H
OUT  DX, al
```

The instruction transfers data byte 10H from register AL to 16bit port address 4000H contained in DX