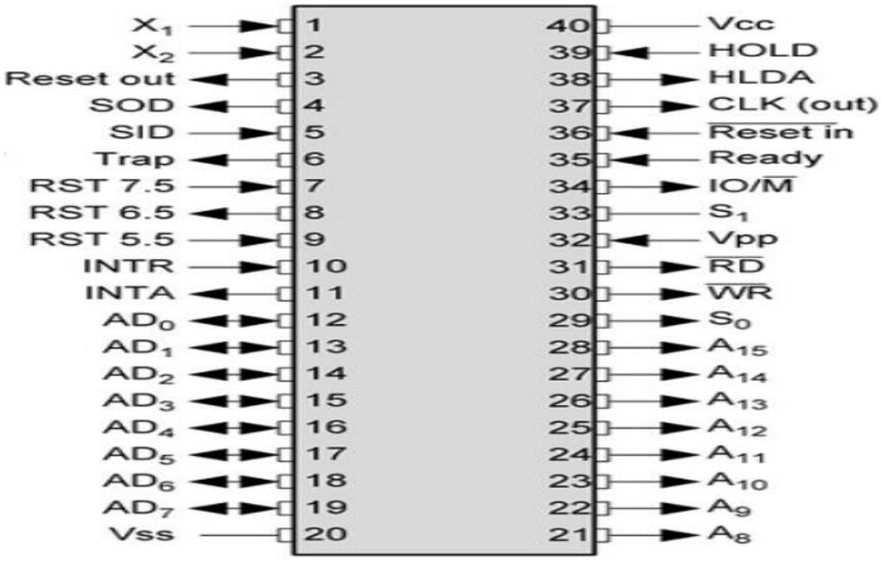
Microprocessor and Assembly programming

1) Pin diagram of 8086 & functionalities of each pin ***

Ans:



Functionality of each pin: The 8085-microprocessor ha 40 pins, each with a specific function. Main functionality of each pin:

i. Vcc (+5V): Power supply pin, provides the operating voltage to the microprocessor.

ii. Address Bus (A15-A8): These pins are part of the 16-bit address bus used to address memory locations and I/O devices.

iii. Address Bus (A7-A0): Additional address bus pins, completing the 16-bit address bus.

iv. Data Bus (AD7-AD0): These pins constitute the 8-bit bidirectional data bus used for transferring data between the microprocessor and memory or I/O devices.

v. Status Signals:
   a) S0, S1: Status signals indicating the current state of the microprocessor.
   b) S2, S3: More status signals indicating the current state of the microprocessor.

vi. Interrupt Control Signals:
   a) INTR: Interrupt request input.
   b) TRAP: Trap input, for higher priority interrupts.
   c) RST7.5, RST6.5, RST5.5: Restart interrupts.

vii. Serial Input/Output:
   a) SID: Serial data input.
   b) SOD: Serial data output.

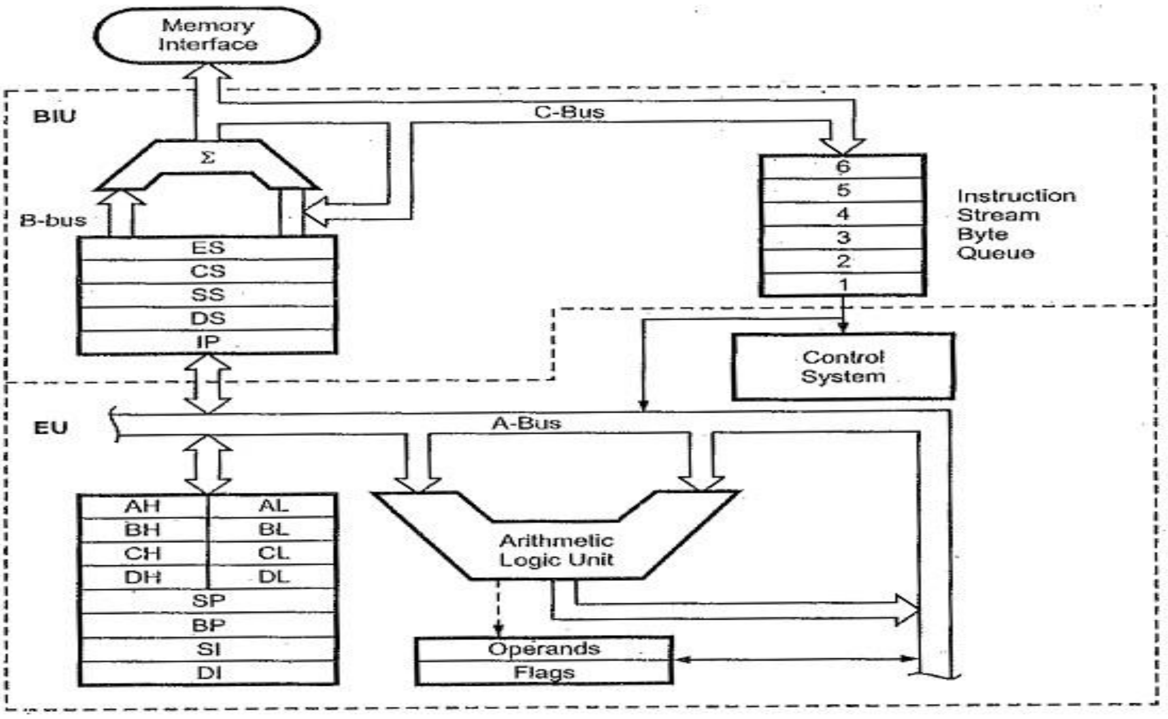
viii. Control Signals:
   a) ALE: Address latch enable, used to latch the address into external latch ICs.
   b) RD: Read control signal, used to indicate a read operation.
   c) WR: Write control signal, used to indicate a write operation.
   d) IO/M: Input/Output (I/O) and Memory control signal, differentiating between memory and I/O operations.

ix. Clock Signals:
   a) CLK: Clock input for synchronizing internal operations.
   b) RESET: Reset input, used to reset the microprocessor.

x. Ground (GND): Ground connection, serves as the reference point for the voltage levels in the system.

2) internal block diagram of 8086 & explanation. *

Ans:



i. Bus Interface Unit (BIU):
   a) Interfaces with system bus, manages memory access.
   b) Contains segment register, instruction pointer and bus controller.
   c) Generates physical address for memory access.

ii. Execution Unit (EU):
   a) CS, DS, SS, ES: Segment for memory addressing.
   b) Define memory segments for code, data, stack and additional operator.

iii. Segment Register:
   a) Generates internal control signals for instruction execution.
   b) Synchronizes operations with system clock for proper timing.

iv. External Bus Interface:
   a) Interfaces with external peripherals (memory, I/O devices).
   b) Facilitates data exchange between microprocessor and peripherals.

3) General purpose register, special purpose register and flag register. ***

Ans: **General Purpose Register:** There are 6 general purpose register (B, C, D, E, H, L). these registers used for a wide range of tasks including storing operands, temporary results, addresses and more. They can also be combined into pairs (BC, DE, HL) to held 16-bit data for performing 16-bit operations. Each holds 8 bits data.

Special Purpose Register: There are several special registers.

i. Program Counter (PC): holds the address of the next instruction to be executed.
ii. Stack pointer (SP): Points to the top of the memory stack, used for data storage.
iii. Flag register: Contains individual flags reflecting the status of the previous operation. Example: Zero flag, Carry flag
iv. Others: Instruction register, Memory Address register, Memory Buffer register.

Flag register: There are 9 active flag registers. 6 conditional flags are:

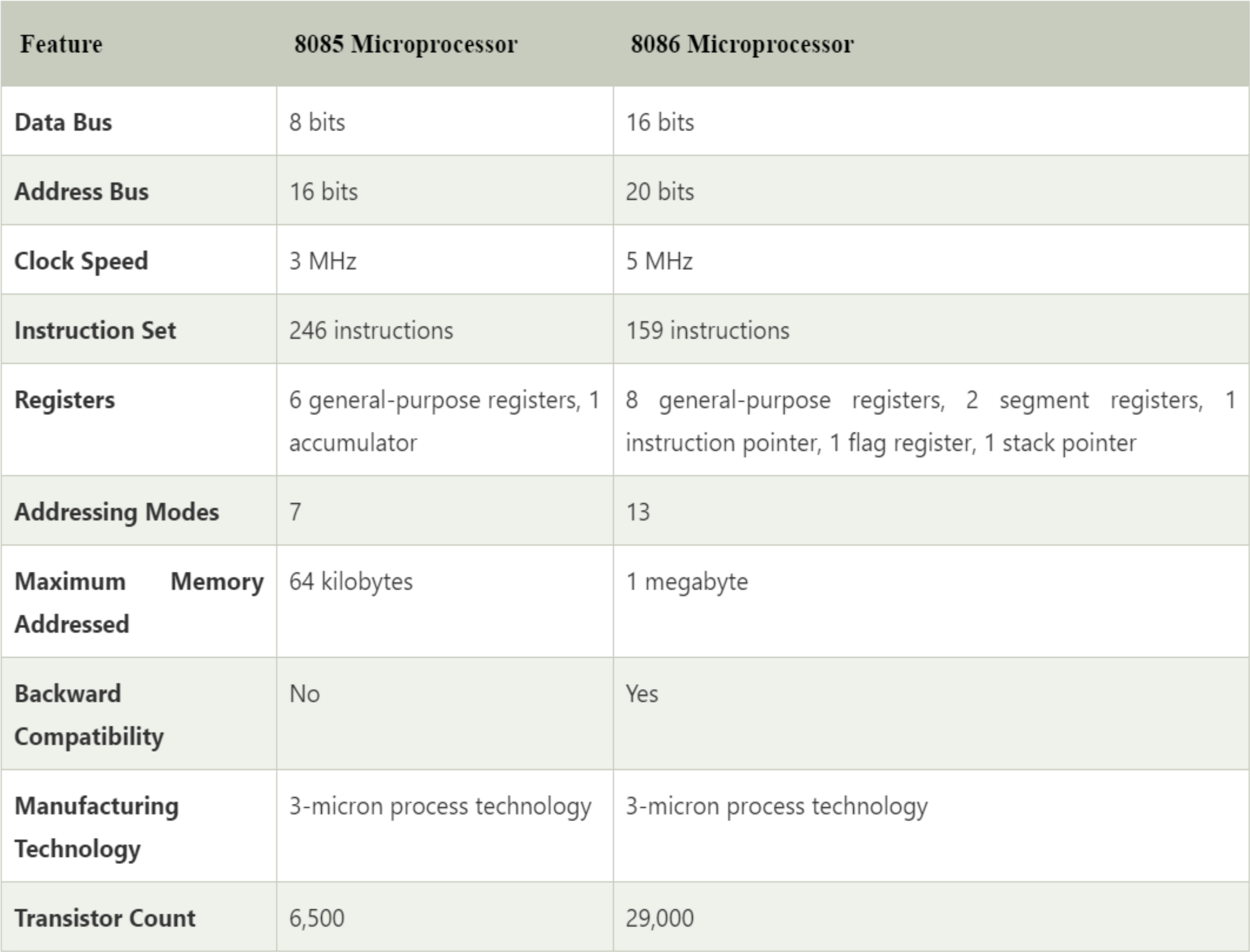
i. Carry flag: CF is set, if an arithmetic operation results in a carry. Otherwise, it is reset.
ii. Parity flag: PF is set if the result has even parity, PF is zero for odd parity. If the result has an even number of 1, flag is set, for odd no. of 1, flag is reset.
iii. Auxiliary flag: AF is used by BCD arithmetic instructions. In an arithmetic operation, when a carry is generated by digit 3 & passed to digit 4, the AF flag is set.
iv. Zero flag: ZF is set (1) if result is zero, ZF is zero (0) for nonzero result.
v. Sign flag: SF is set if most significant bit of the result is 1(negative). Cleared to zero, for non-negative result.
vi. Overflow flag: OF set, if there is an arithmetic overflow, that is, if the size of the result exceeds the capacity of destination location.

3 remaining flags are used to control certain operations:

a) Trap flag b) Interrupt flag c) Direction flag.

4) <u>Characteristics, features, difference, advantages & disadvantages of 8085 & 8086. ***</u>

Ans: <u>Features</u>:



Feature	8085 Microprocessor	8086 Microprocessor
Data Bus	8 bits	16 bits
Address Bus	16 bits	20 bits
Clock Speed	3 MHz	5 MHz
Instruction Set	246 instructions	159 instructions
Registers	6 general-purpose registers, 1 accumulator	8 general-purpose registers, 2 segment registers, 1 instruction pointer, 1 flag register, 1 stack pointer
Addressing Modes	7	13
Maximum Memory Addressed	64 kilobytes	1 megabyte
Backward Compatibility	No	Yes
Manufacturing Technology	3-micron process technology	3-micron process technology
Transistor Count	6,500	29,000

<u>Advantage and disadvantages of 8085 & 8086:</u>

<u>Advantage of 8085;</u>

i. Due to its smaller size and simpler design

ii. Suitable for battery-powered applications.
iii. Simpler instruction set and architecture.
iv. Extensive learning resources and support.

Disadvantages of 8085:

i. Slower and less memory access compared to 8086.
ii. Can be limiting for complex applications.
iii. Limits maximum memory usage.

Advantage of 8086:

i. Faster clock speed and wider data bus for faster processing.
ii. Supports complex operations like multiplication and division.
iii. Allows for addressing more memory.
iv. Improves responsiveness to external events.

Disadvantages of 8086:

i. More expensive due to increased complexity.
ii. Not ideal for battery-powered applications.
iii. Requires understanding of additional features.

Difference of 8085 & 8086:

i. The primary difference lies in their word size, leading to variations in data handling, memory addressing, and instruction complexity.
ii. The 8086 offers significantly higher performance with its faster clock speed, wider data bus, and pipeline architecture.
iii. The 8085 is simpler and more cost-effective, making it ideal for resource-constrained applications.
iv. The 8086 has a richer instruction set with features like memory-to-memory operations and multiplication/division, suitable for more demanding tasks.

5) Categorization of interrupt, ISR location, maskable or non-maskable. ***

Ans: An interrupt is a signal sent by hardware or software to temporarily halt the CPU's current execution and handle a specific event. It allows the CPU to respond promptly to critical events such as I/O requests, timer expirations, or errors. Interrupts can be initiated by external hardware devices or by executing special software instructions. Upon receiving an interrupt, the CPU transfers control to a predefined interrupt service routine (ISR) to handle the event before returning to normal execution.

There are two types of interrupts:

Software Interrupts:

i. Mainly Instructions: Initiated by executing specific software instructions.
ii. RST0 - RST7: Restart instructions that jump to predefined memory locations.

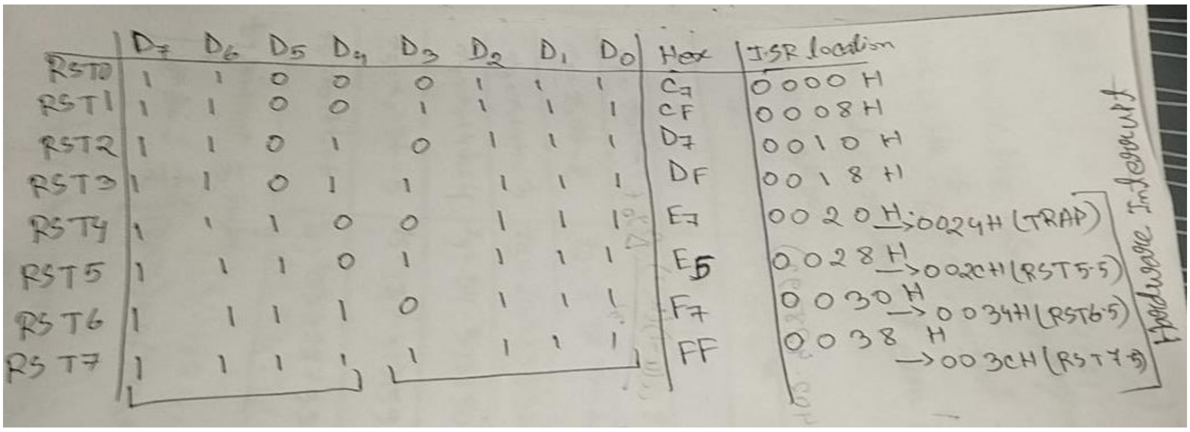
Hardware Interrupts:

i. Signal from Input/Output: Generated by external hardware devices.
ii. Generated by Hardware: Triggered directly by hardware components.

iii. RST7.5, RST6.5, RST5.5, TRAP: Special restart instructions or trap available in the Intel 8085.

iv. Maskable: Some interrupts can be disabled (maskable) which are RST7.5, RST5.5, while others cannot (non-maskable) which is TRAP.

<u>ISR location:</u> The Interrupt Service Routine (ISR) location refers to the specific memory address where the code responsible for handling an interrupt is stored. When an interrupt occurs, the CPU jumps to this predefined memory location to execute the ISR, which performs the necessary actions to handle the interrupt.



	D7	D6	D5	D4	D3	D2	D1	D0	Hex	ISR location
RST0	1	1	0	0	0	1	1	1	C7	0000 H
RST1	1	1	0	0	1	1	1	1	CF	0008 H
RST2	1	1	0	1	0	1	1	1	D7	0010 H
RST3	1	1	0	1	1	1	1	1	DF	0018 H
RST4	1	1	1	0	0	1	1	1	E7	0020 H → 0024H (TRAP)
RST5	1	1	1	0	1	1	1	1	E5	0028 H → 002CH (RST5.5)
RST6	1	1	1	1	0	1	1	1	F7	0030 H → 0034H (RST6.5)
RST7	1	1	1	1	1	1	1	1	FF	0038 H → 003CH (RST7.5)

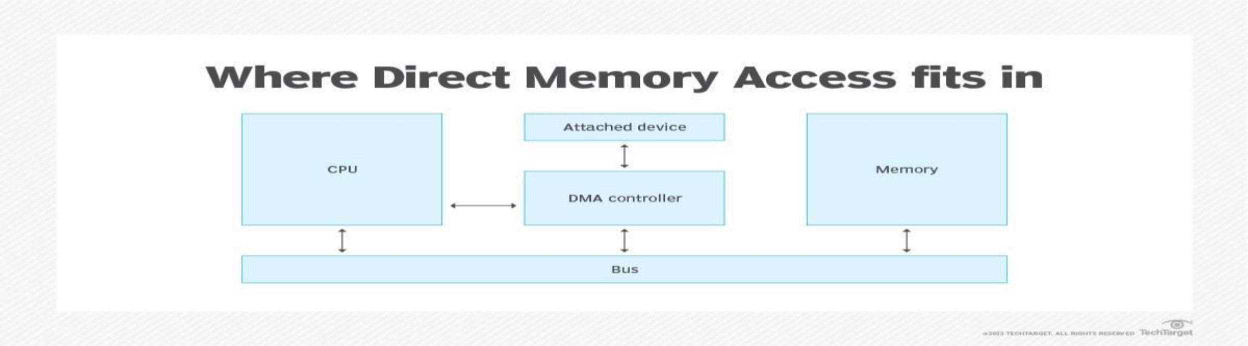
Hardware Interrupt

The ISR location is typically determined during system initialization or configuration and is specific to each type of interrupt. It is crucial for efficient interrupt handling and ensuring timely response to interrupt requests in a computer system.

6) <u>Direct memory access. *</u>

Ans: Direct Memory Access (DMA) is a capability provided by some computer bus architectures that enables data to be sent directly from an attached device, such as a disk drive, to the main memory on the computer's motherboard. The microprocessor, or central processing unit (CPU), is freed from involvement with the data transfer, speeding up overall computer operation. DMA enables devices (such as disk drives, external memory, graphics cards, network cards and sound cards) to share and receive data from the main memory in a computer. It does this while still allowing the CPU to perform other tasks. Without a process such as DMA, the computer's CPU becomes preoccupied with data requests from an attached device and is unable to perform other operations during that time. With DMA, a CPU initiates a data transfer with an attached device and can still perform other operations while the data transfer is in progress. DMA enables a computer to transfer data to and from devices with less CPU overhead.

The CPU gets the DMA controller to begin data transfers, and the DMA controller then sets the memory addresses and read/write lines to the system memory.

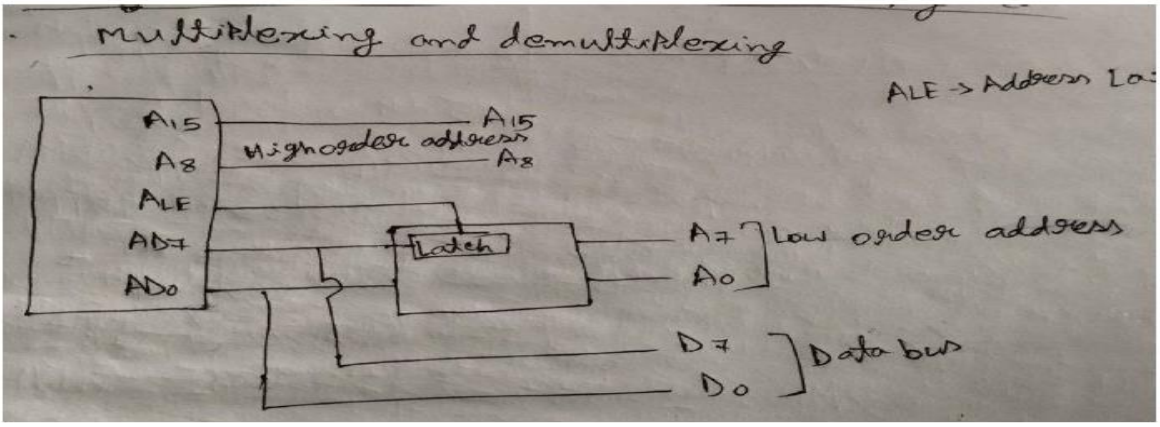

Where Direct Memory Access fits in
Attached device
CPU
DMA controller
Memory
Bus
©2023 TechTarget. All rights reserved TechTarget

7) Multiplexing and demultiplexing of address & data bus. *

Ans: Multiplexing and demultiplexing of the address and data bus is a technique used in microprocessor systems to reduce the number of physical lines required for communication between the CPU and external devices such as memory and input/output (I/O) devices.

Multiplexing of Address and Data Bus:

i. During certain operations, such as memory or I/O read/write cycles, the microprocessor multiplexes the address and data on the same bus lines.
ii. In the initial phase of a memory or I/O operation, the microprocessor places the lower-order bits of the memory address or I/O port address on the data bus.
iii. Then, in subsequent cycles, the microprocessor places data to be written to memory or I/O devices on the same data bus.



Demultiplexing of Address and Data Bus:

i. External memory or I/O devices connected to the microprocessor need to demultiplex the address and data.
ii. During the first cycle of a memory or I/O operation, external circuitry latches the lower-order bits of the address from the data bus.
iii. In subsequent cycles, when the data is available on the bus, the external devices interpret the latched address along with control signals to determine the memory location or I/O port to be accessed.

# Mosfet technology used in 85 & 86*

# Maskable & non-Maskable