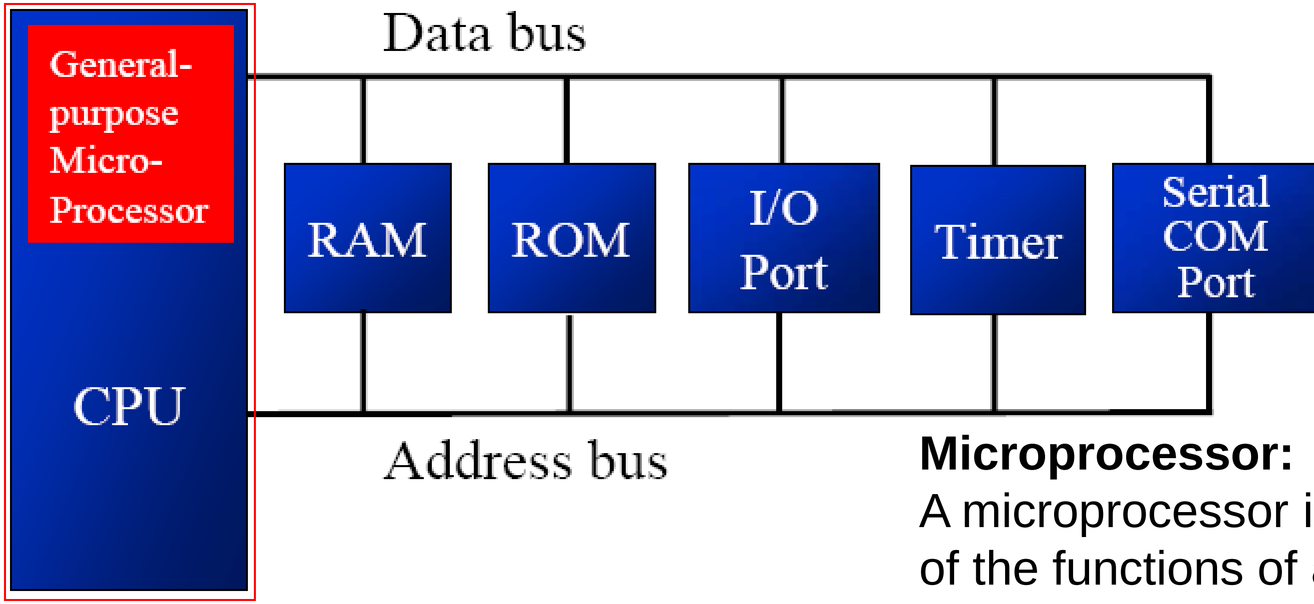
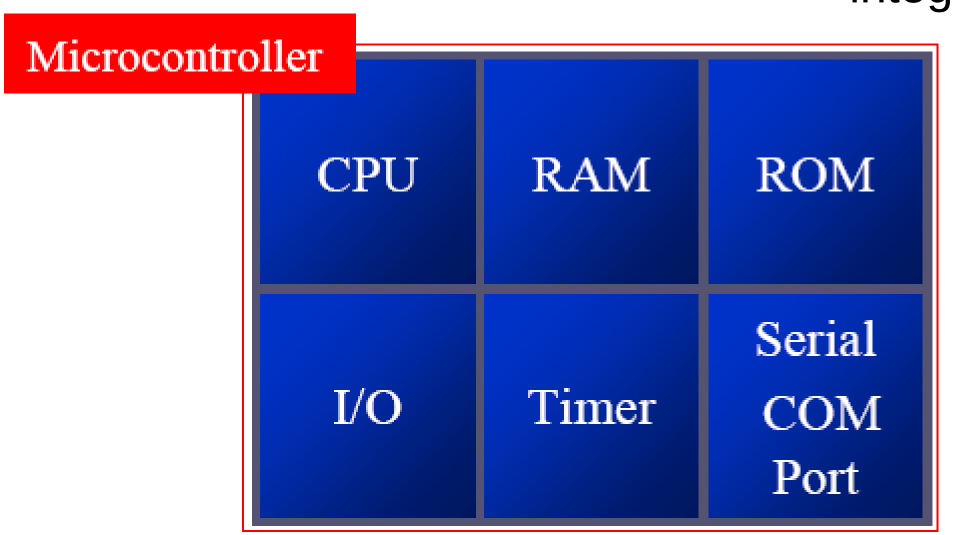
MODULE 1:
OVERVIEW OF MICROPROCESSORS



Microprocessor:
A microprocessor incorporates most or all of the functions of a computer's central processing unit (CPU) on a single integrated circuit (IC, or microchip).



Microcontroller:
If you put all the major components of a Single Board Computer onto a single chip it will be called as a Microcontroller.

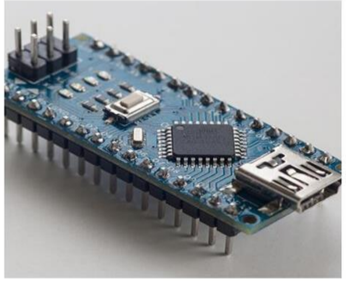
Introduction to Microprocessors

- A Microprocessor is a programmable, multipurpose, clock -driven, register-based electronic device that reads binary instructions from a storage device called memory, accepts binary data as input and processes data according to those instructions and provides results as output.

- Computer's Central Processing Unit (CPU) built on a **single Integrated Circuit (IC)** is called a **microprocessor**

- A Microprocessor has many components like transistors, registers, and diodes which come together to perform. The ability of the chip has become more complex with technology evolution. The functionality has become better and the speed has become faster.

- Most devices need to have a microprocessor to function. It is the element that brings intelligence to a device. Be it a computer or mobile phone, all devices need an interface to handle data that only a microprocessor provides.

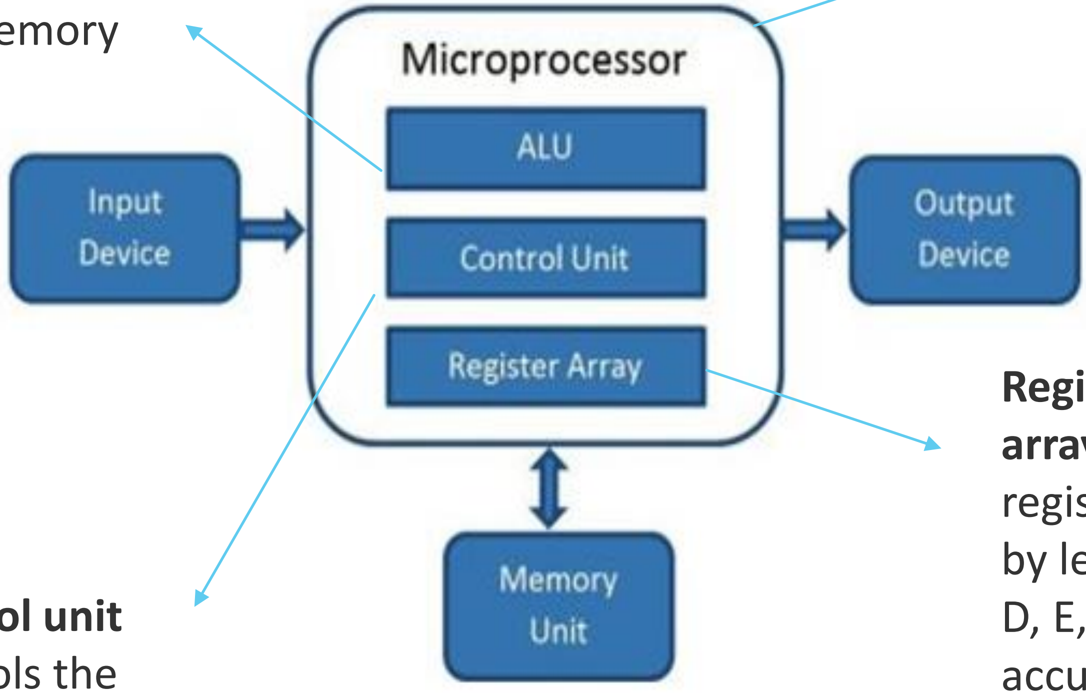


Generalized Block Diagram

ALU performs arithmetic and logical operations on the data received from an input device or memory

Communication through I/O Port/Bus

- ✓ controls the computer and performs all calculations.
- ✓ ability to perform various operations quickly



Register array consists of registers identified by letters like B, C, D, E, H, L, and accumulator.

Control unit controls the instructions and flow of data within the computer

stores programs and data required by the processor

Working of a Microprocessor

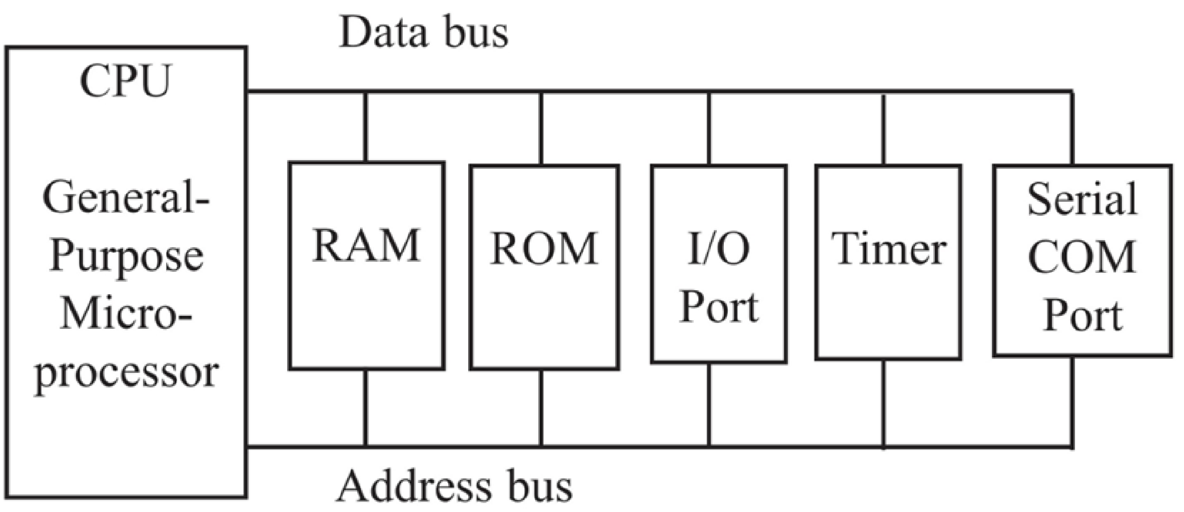
There are three steps that a microprocessor follows –

- **Fetch** – The instructions are in storage from where the processor fetches them.
- **Decode** – It then decodes the instruction to assign the task further. During this, the arithmetic and logic unit also performs to register the data temporarily.
- **Execute** – The assigned tasks undergo execution and reach the output port in binary form.

Between these processes, the register stores the temporary data and ALU (Arithmetic and Logic Unit) performs the computing functions.

Memory

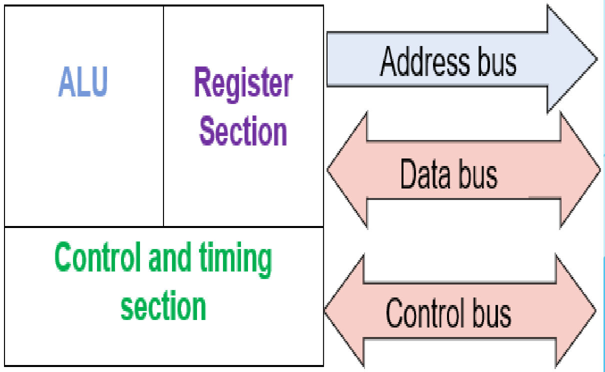
- **RAM** − Random Access Memory is **volatile memory** that gets erased when power is switched off. All data and instructions are stored in RAM.
- **ROM** − Read Only Memory is **non-volatile memory** whose data remains intact even after power is switched off. Microprocessor can read from it any time it wants but cannot write to it.



It is a processor in which memory and I/O output component is connected externally.

Basic Microprocessor Terms

- **Instructions Per Cycle** – A way to measure CPU's instruction speed in a single clock.
- **Instruction Set** – These are the commands that a processor understands to work between hardware and software.
- **Bus –** Set of conductors for data transmission, information control, and tasks addressed in a microprocessor. They are of three types – data bus, address bus, and control bus.
- **Word Length** – Refers to the number of bits processed at a time.
- **Clock Speed / Clock Rate** – The ability of microprocessors to perform tasks in a second.
- **Bandwidth** – Refers to the total bits in a single instruction.
- **Data Types** – Data type microprocessor supports like binary, ASCII, etc.
- **SIMD** – Single Instruction Multiple Data
- **PGA** – Pin Grid Array
- **FPU** – Floating Point Unit
- **ALU** – Arithmetic and Logic Unit
- **MMX** – MultiMedia eXtensions
- **MMU** – Memory Management Unit
- **Bit-** Binary Digit



Evolution of Microprocesors

Transistor was invented in 1948 (23 December 1947 in Bell lab). IC was invented in 1958 (Fair Child Semiconductors) By Texas Instruments J Kilby. The first microprocessor was invented by INTEL(**INT**egrated **EL**ectronics)

Name & Year	Bit Size	Description	Limitations/Improvements
4004 (1971)	4	1st Microprocessor having 2300 Transistors. With evolution in semiconductor Technology more transistors were then used in process	Very few instructions per second. Used for simple arithmetic and logical operations
8008 (1972)	8	1st 8 bit Microprocessor. It was useful for arithmetic and logic operations on 8-bit words	All 8 bit microprocessors have limited memory, addressing capacity, slow speed of execution, addressing modes etc.
8080 (1974)	8	World's first general purpose microprocessor. It was an 8-bit machine, with an 8-bit data path to memory. It was used in the first personal computer. Incomplete CPU. Clock speed of 2 MHz and 60k instruction per second. Clock speed of 500kHz and 50k instructions per second.	
8085 (1976)	8	Functionally complete CPU. Most Popular. On-chip clock generator present. Optimum set of registers and powerful performance gave good performance. 769230 instruction per second with 3 MHz speed.	
8086 (1978)	16	1st 16 bit microprocessor which overcomes most of the limitations of previous generation. It had a wider data path of 16-bits and larger registers along with an instruction cache or queue ,rich instruction set and more important had segmented memory scheme. It also had multiply and divide instructions	

Evolution of Microprocesors (contd…)

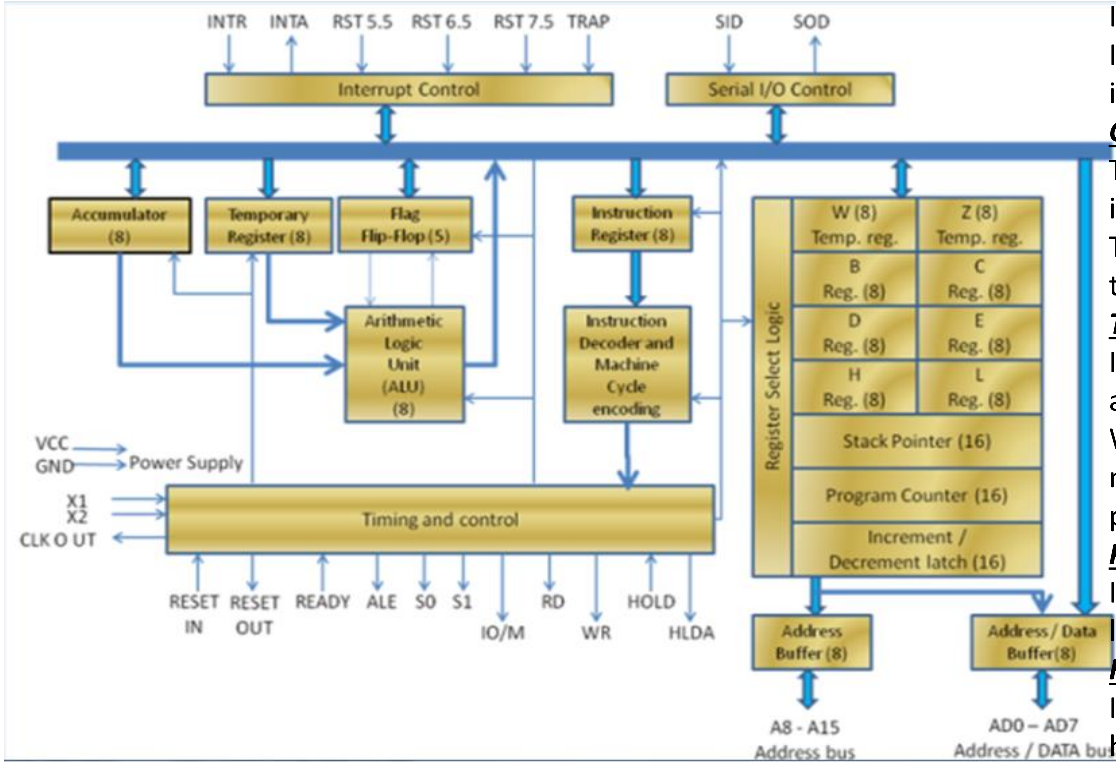
Name & Year	Bit Size	Description	Limitations/Improvements
8088 (1980)	16	Cheaper version of 8086 and 8-bit external bus. Used in IBM PC.	Both 8086/88 didn't have memory management and protection capability.
80286 (1982)	16	Addressable memory of 16 MB instead of just 1 MB and contains two modes-real mode and first generation 16-bit protected mode	Limitation on maximum segment size .
80386 (1985)	32	Intel's first 32-bit machine. Due to its 32-bit architecture it was able to compete against the complexity and power of microcomputers and mainframes introduced just a few years earlier. It was the first processor to support multitasking and contained the 32-bit protected mode.	
80387 (1986)	32	Variant of 80386. It supported high speed environment. Integrated Math Co-processor	
80486 (1989)	32	Introduced the concept of cache technology and instruction pipelining which increased processor speed and optimized instruction set.	

Architecture of 8085 and 8086 Microprocessors

The main features of 8085 μp are:

- It is a 8 bit microprocessor.
- It is manufactured with N-MOS technology.
- It has 16-bit address bus and hence can address up to $2^{16} = 65536$ bytes (64KB) memory locations through A^0-A^{15} .
- The first 8 lines of address bus and 8 lines of data bus are multiplexed $AD^0 – AD^7$.
- Data bus is a group of 8 lines $D^0 – D^7$.
- It supports external interrupt request.
- A 16 bit program counter (PC)
- A 16 bit stack pointer (SP)
- Six 8-bit general purpose register arranged in pairs: BC, DE, HL.
- It requires a signal +5V power supply and operates at 3.2 MHZ single phase clock.
- It is enclosed with 40 pins DIP (Dual in line package).

8085 Microprocessor: A brief study



Data bus carries the data to be stored. It is bidirectional, whereas address bus carries the location to where it should be stored and it is unidirectional. It is used to transfer the data & Address I/O

Its bit position is shown in the following table –

D7	D6	D5	D4	D3	D2	D1	D0
S	Z		AC		P		CY

Accumulator

It is an 8-bit register used to perform arithmetic, logical, I/O & LOAD/STORE operations. It is connected to internal data bus & ALU.

General purpose register

There are 6 general purpose registers in 8085 processor, i.e. B, C, D, E, H & L. Each register can hold 8 bit data. These registers can work in pair to hold 16 bit data and their pairing combination is like B C, D E & H L.

Temporary register

It is an 8 bit register, which holds the temporary data of arithmetic and logical operations.

W and Z are two 8 bit temporary registers of 8085 microprocessor, which is not accessible to the user. (non programmable)

Program counter

It is a 16 bit register used to store the memory address location of the next instruction to be executed.

Flag register

It is an 8 bit register having five 1 bit flip flops , which holds either 0 or 1 depending upon the result stored in the accumulator.

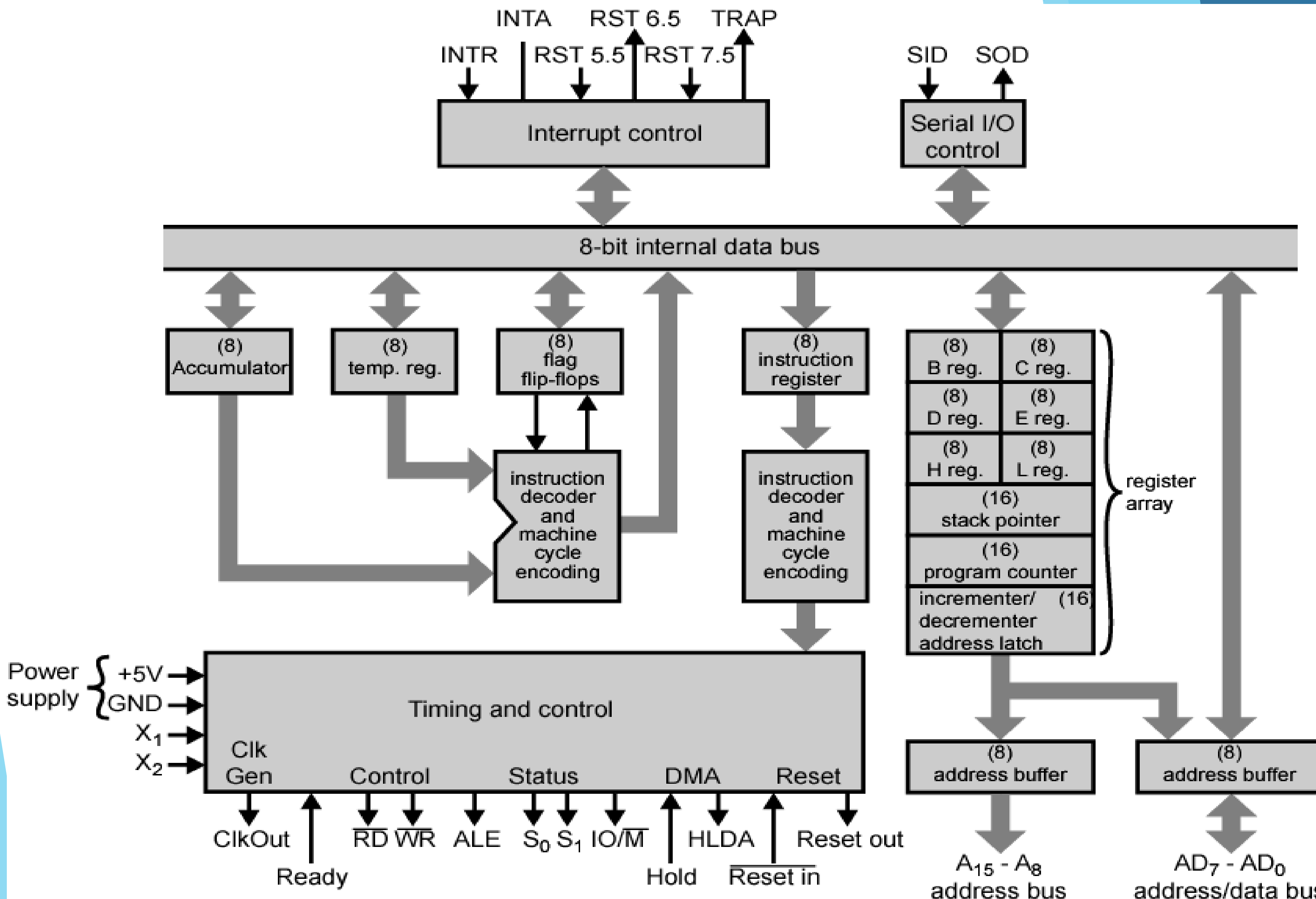
Stack pointer

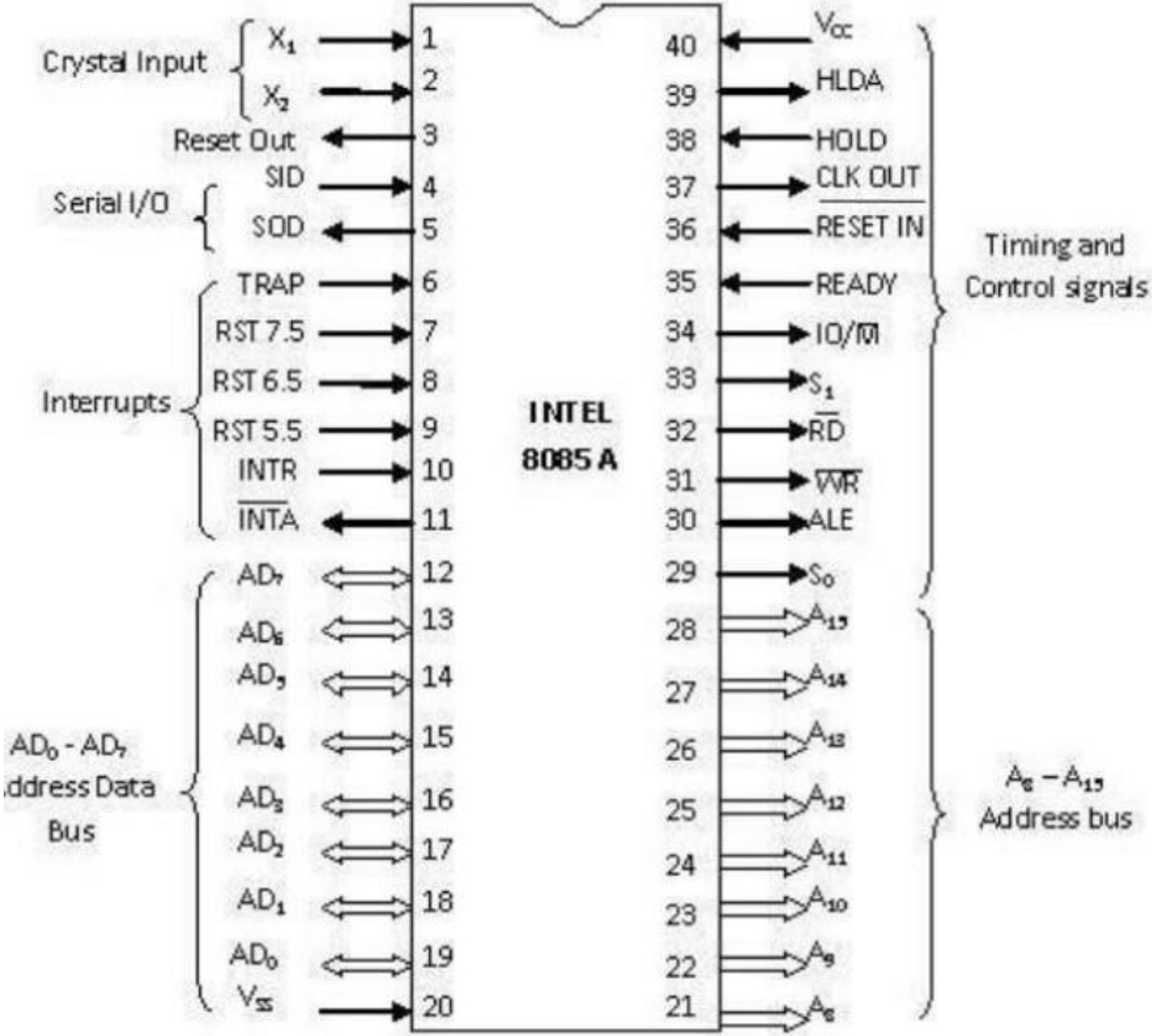
It is also a 16 bit register works like stack, which is always incremented/decremented by 2 during push & pop operations. It holds the address of the top location of the stack.

Instruction register and decoder

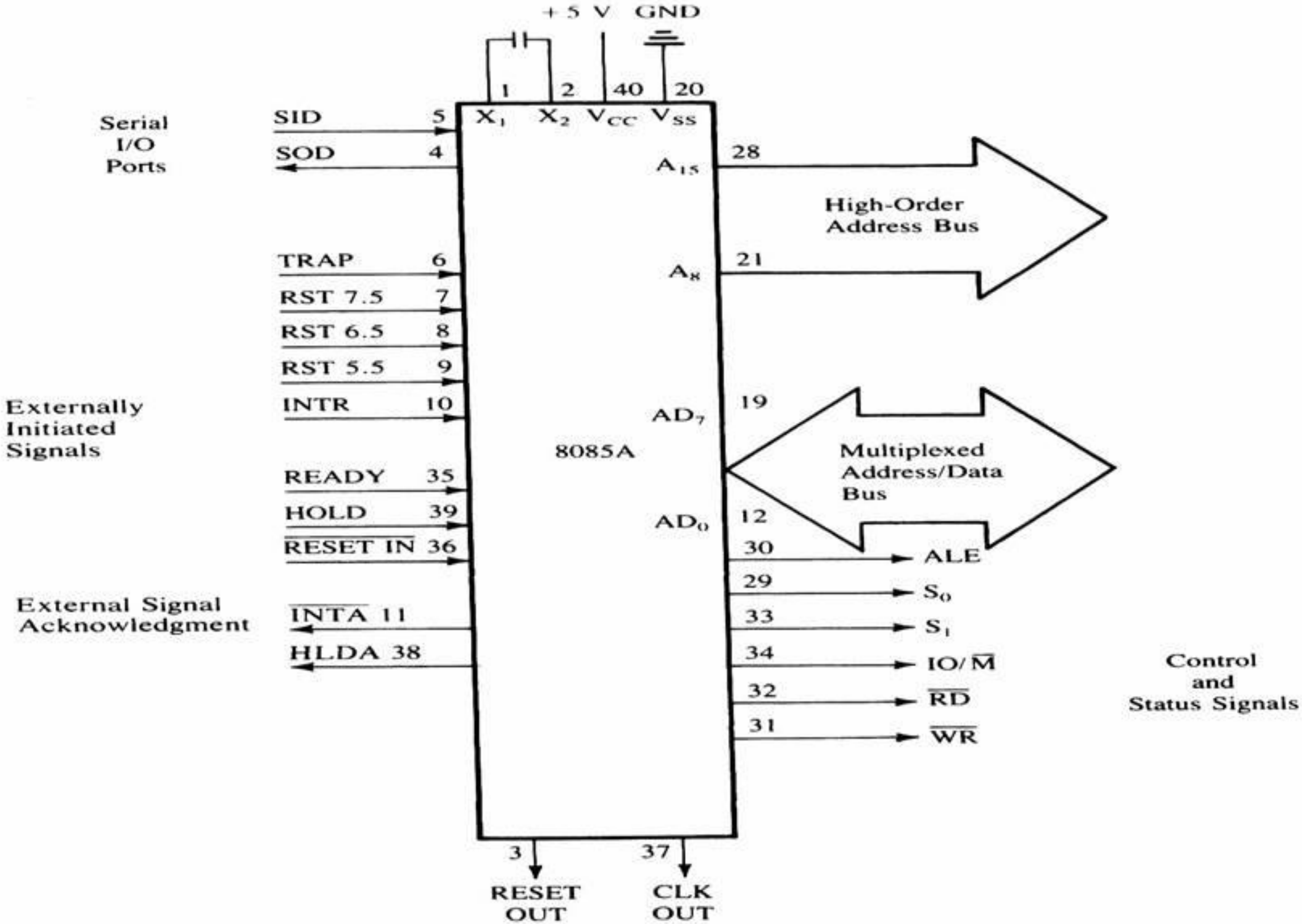
It is an 8 bit register.

When an instruction is fetched from memory then it is stored in the Instruction register. It decodes the information present in the Instruction register.


INTA
RST 6.5
TRAP
INTR
RST 5.5
RST 7.5
SID
SOD
Interrupt control
Serial I/O control
8-bit internal data bus
(8) Accumulator
(8) temp. reg.
(8) flag flip-flops
(8) instruction register
(8) B reg.
(8) C reg.
(8) D reg.
(8) E reg.
(8) H reg.
(8) L reg.
(16) stack pointer
(16) program counter
incrementer/ decrementer address latch (16)
register array
instruction decoder and machine cycle encoding
instruction decoder and machine cycle encoding
Power supply
+5V
GND
X_1
X_2
Timing and control
Clk Gen
Control
Status
DMA
Reset
ClkOut
Ready
$\overline{RD}$ $\overline{WR}$
ALE
S_0 S_1 IO/$\overline{M}$
Hold
HLDA
$\overline{\text{Reset in}}$
Reset out
(8) address buffer
(8) address buffer
A_{15} - A_8 address bus
AD_7 - AD_0 address/data bus



Intel 8085 Pin Configuration



Signals and I/O Pins

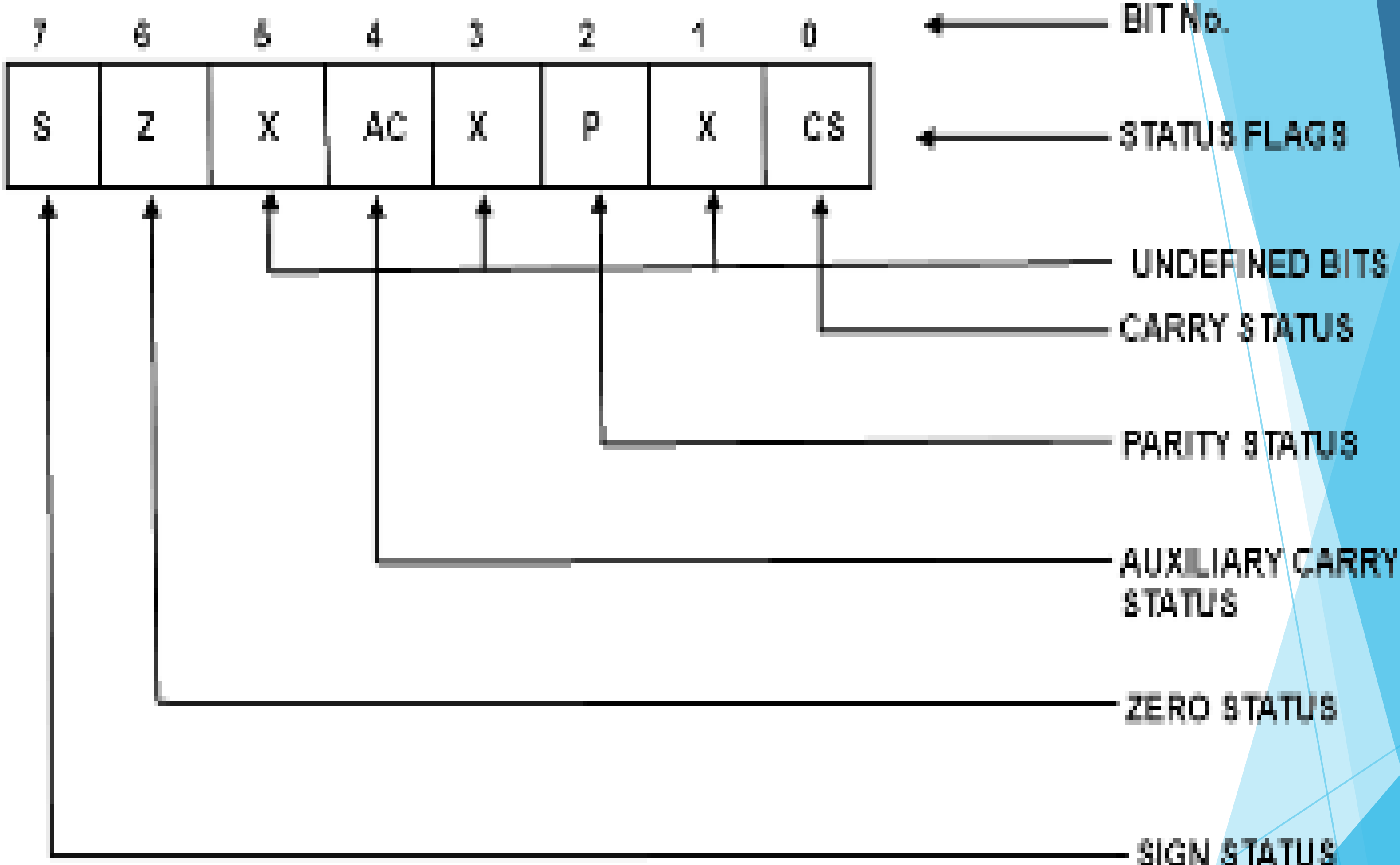


Figure 2: Status Flags of Intel 8085

Features of 8086 Microprocessor:

1. Intel 8086 was launched in 1978.
2. It was the first 16-bit microprocessor.
3. This microprocessor had major improvement over the execution speed of 8085.
4. It is available as 40-pin Dual-Inline-Package (DIP).

5. 8086 is a 16bit processor. It's ALU, internal registers works with 16bit binary word.
6. 8086 has a 16bit data bus. It can read or write data to a memory/port either 16bits or 8 bit at a time.
7. 8086 has a 20bit address bus which means, it can address upto 220 = 1MB memory location.
8. Frequency range of 8086 is 6-10 MHz

1. Bus Interface Unit (BIU)
2. Execution Unit (EU)

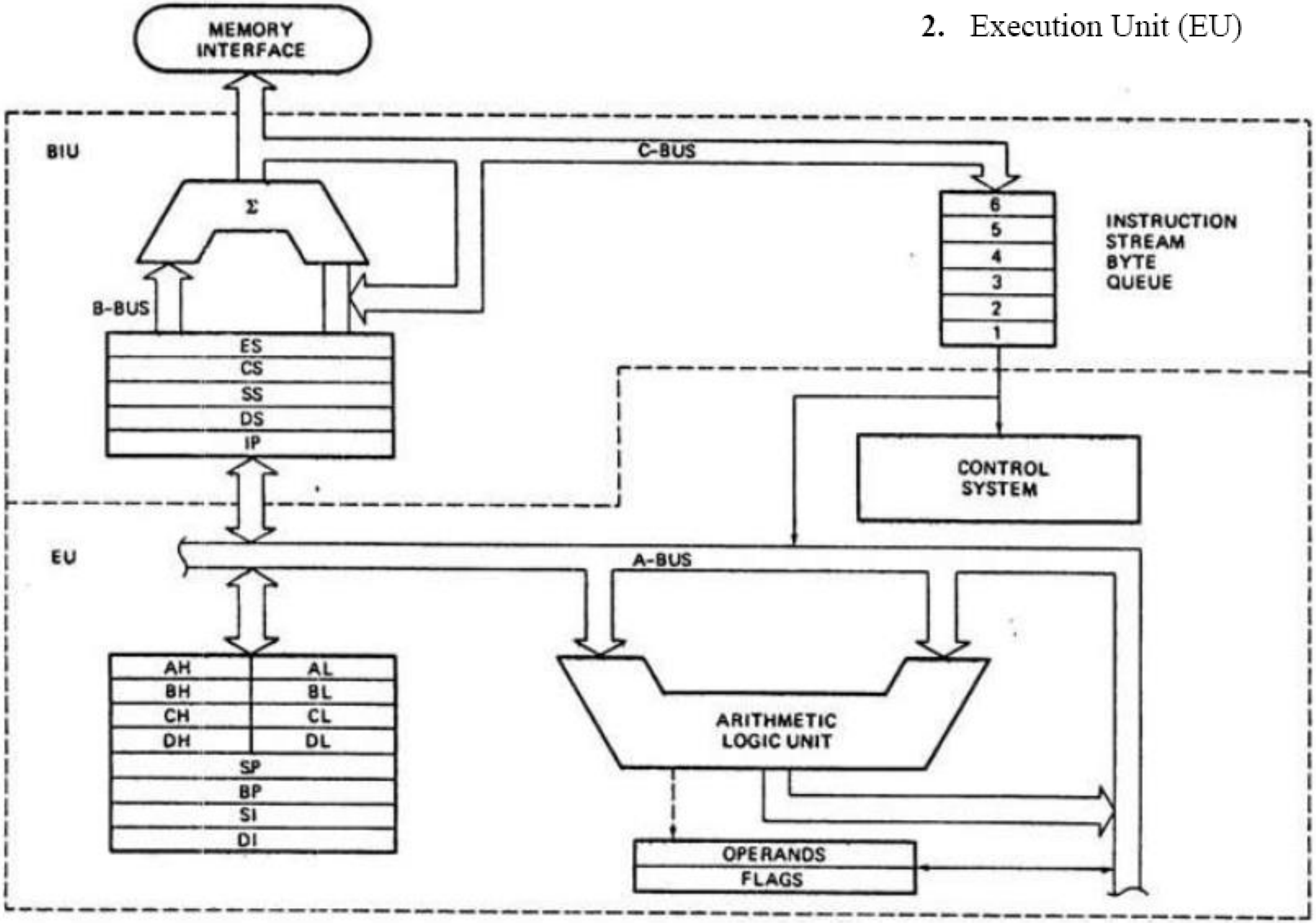
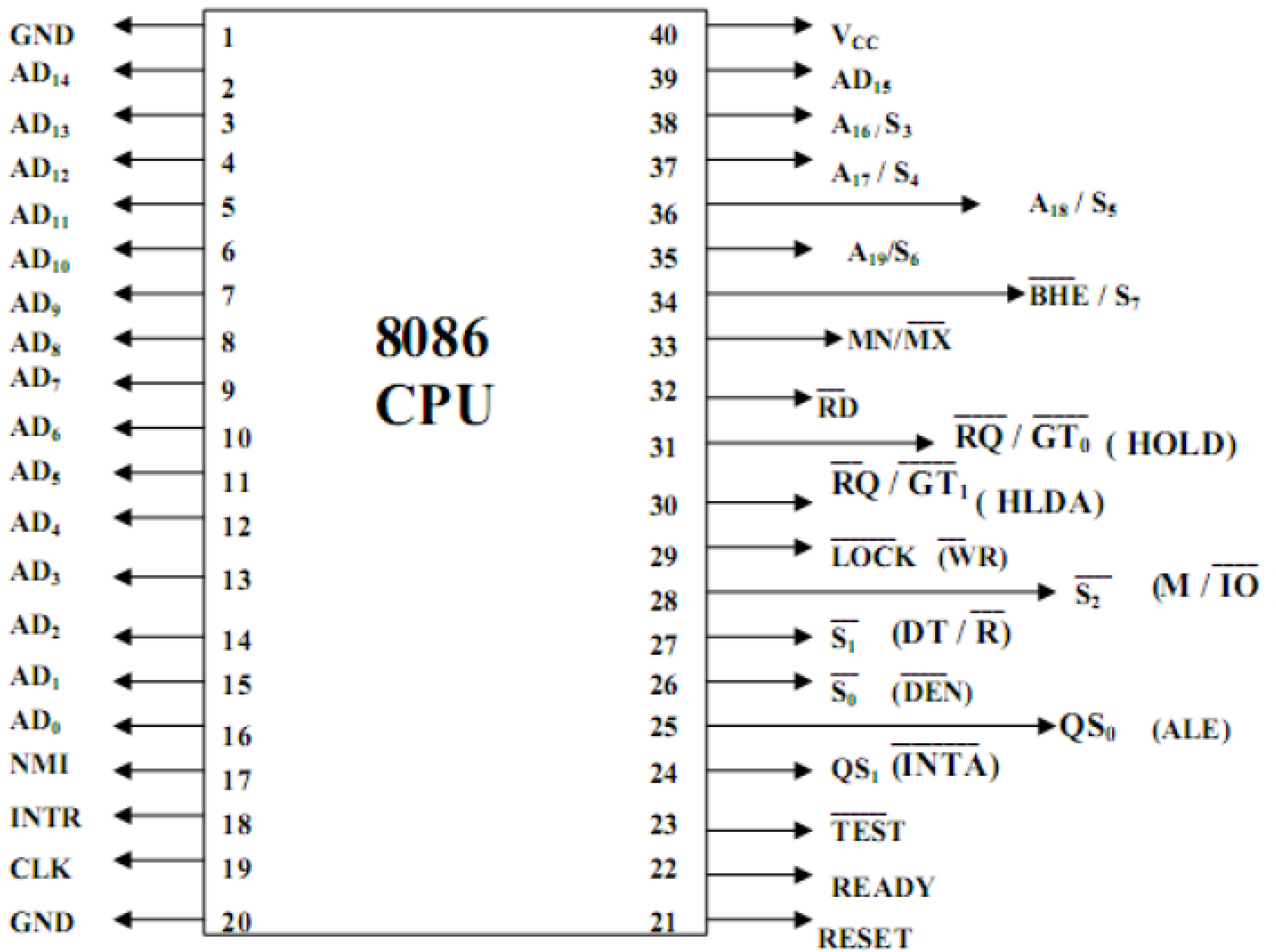
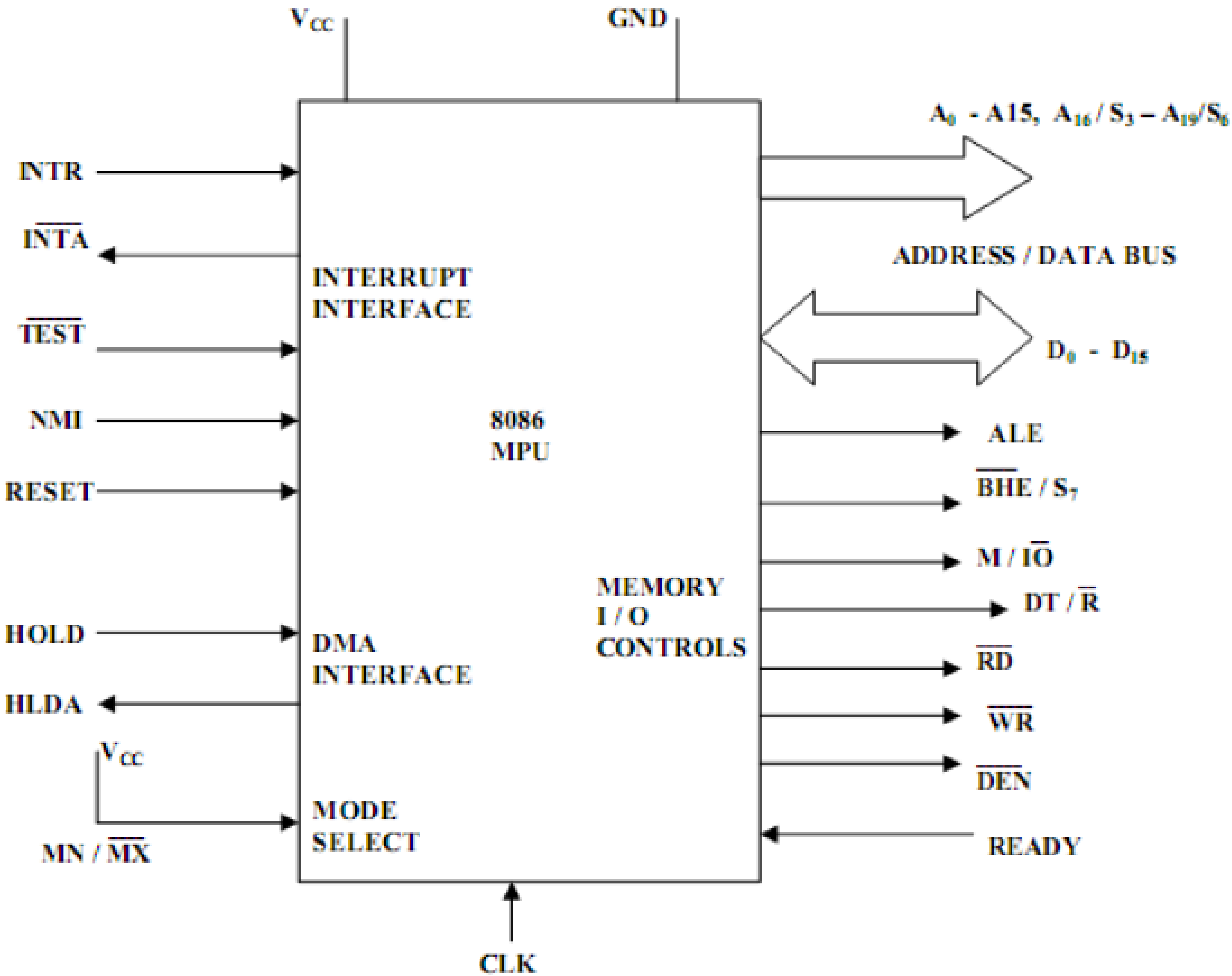


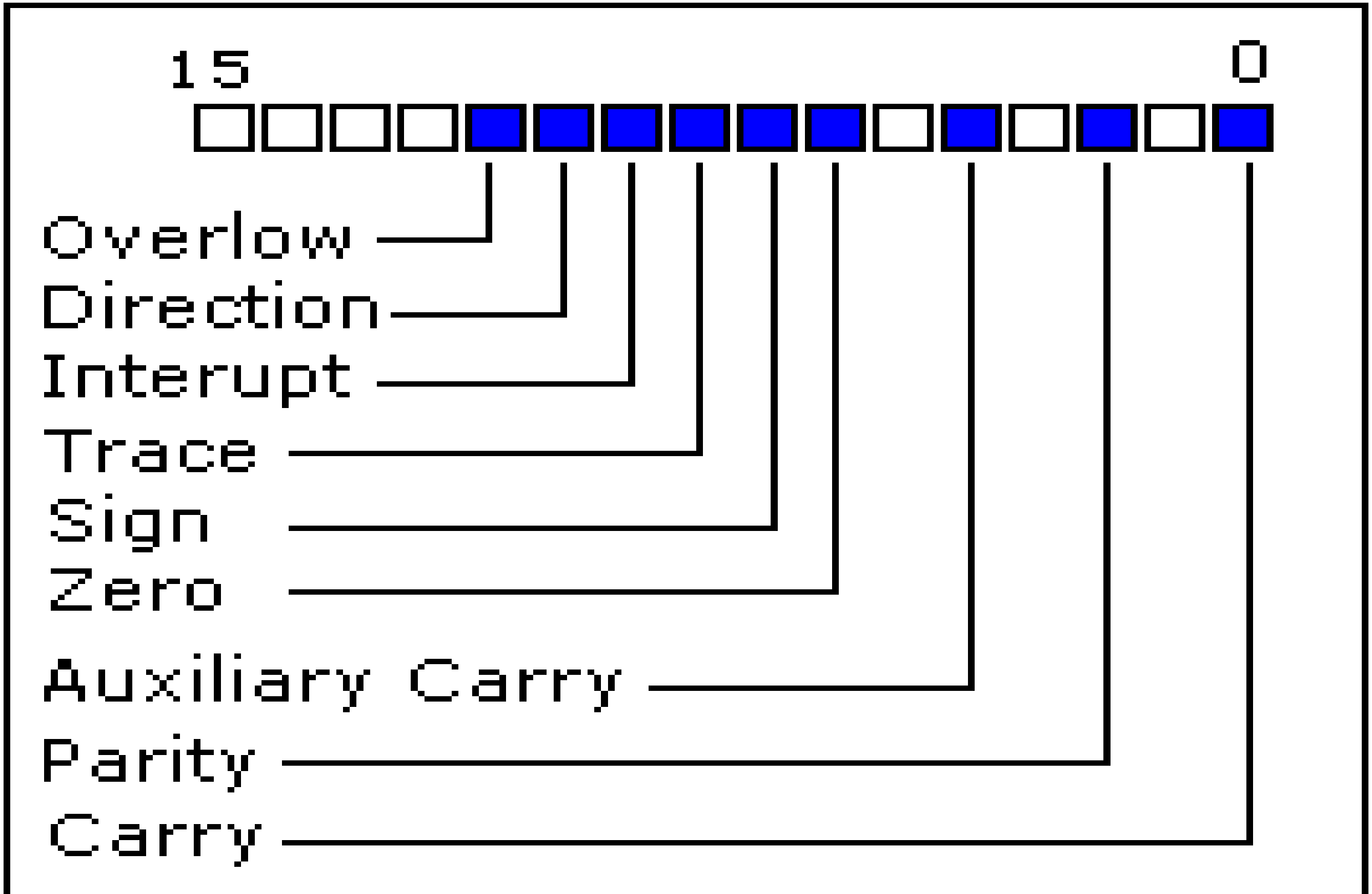
Fig. 1: Block Diagram of Intel 8086

Pin Diagram of 8086:

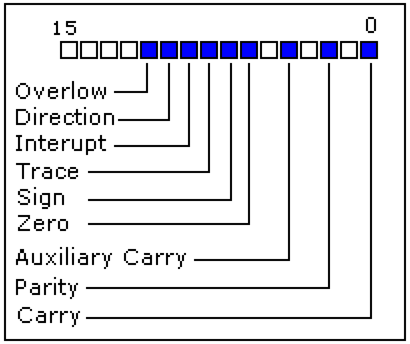



V_{CC}
GND
A_0 - A15, $A_{16}/S_3 - A_{19}/S_6$
INTR
$\overline{INTA}$
INTERRUPT
INTERFACE
ADDRESS / DATA BUS
$\overline{TEST}$
D_0 - D_{15}
NMI
8086
MPU
ALE
RESET
$\overline{BHE}/S_7$
$M/\overline{IO}$
MEMORY
I / O
CONTROLS
$DT/\overline{R}$
HOLD
DMA
INTERFACE
$\overline{RD}$
HLDA
$\overline{WR}$
V_{CC}
$\overline{DEN}$
MODE
SELECT
MN / $\overline{MX}$
READY
CLK

Flags of 8086 processor



Most Arithmetic and Logic Instructions affect the processor status register (or **Flags**)



As you may see there are 16 bits in this register, each bit is called a **flag** and can take a value of **1** or **0**.

- **Carry Flag (CF)** - this flag is set to **1** when there is an **unsigned overflow**. For example when you add bytes **255 + 1** (result is not in range 0...255). When there is no overflow this flag is set to **0**.
- **Zero Flag (ZF)** - set to **1** when result is **zero**. For none zero result this flag is set to **0**.
- **Sign Flag (SF)** - set to **1** when result is **negative**. When result is **positive** it is set to **0**. Actually this flag take the value of the most significant bit.
- **Overflow Flag (OF)** - set to **1** when there is a **signed overflow**. For example, when you add bytes **100 + 50** (result is not in range -128...127).
- **Parity Flag (PF)** - this flag is set to **1** when there is even number of one bits in result, and to **0** when there is odd number of one bits. Even if result is a word only 8 low bits are analyzed!
- **Auxiliary Flag (AF)** - set to **1** when there is an **unsigned overflow** for low nibble (4 bits).
- **Interrupt enable Flag (IF)** - when this flag is set to **1** CPU reacts to interrupts from external devices.
- **Direction Flag (DF)** - this flag is used by some instructions to process data chains, when this flag is set to **0** - the processing is done forward, when this flag is set to **1** the processing is done backward.

Difference Between RISC and CISC

1. RISC stands for Reduced Instruction Set Computer.	1. CISC stands for Complex Instruction Set Computer.
2. RISC processors have simple instructions taking about one clock cycle. The average clock cycle per instruction (CPI) is 1.5	2. CSIC processor has complex instructions that take up multiple clocks for execution. The average clock cycle per instruction (CPI) is in the range of 2 and 15.
3. Performance is optimized with more focus on software	3. Performance is optimized with more focus on hardware.
4. It has no memory unit and uses a separate hardware to implement instructions..	4. It has a memory unit to implement complex instructions.
5. It has a hard-wired unit of programming.	5. It has a microprogramming unit.
6. The instruction set is reduced i.e. it has only a few instructions in the instruction set. Many of these instructions are very primitive.	**6.** The instruction set has a variety of different instructions that can be used for complex operations.

7. The instruction set has a variety of different instructions that can be used for complex operations.	**7.** CISC has many different addressing modes and can thus be used to represent higher-level programming language statements more efficiently.
8. Complex addressing modes are synthesized using the software.	8. CISC already supports complex addressing modes
9. Multiple register sets are present	9. Only has a single register set
10. RISC processors are highly pipelined	10. They are normally not pipelined or less pipelined
11. The complexity of RISC lies with the compiler that executes the program	11. The complexity lies in the microprogram
12. Execution time is very less	12. Execution time is very high
17. RISC architecture is used in high-end applications such as video processing, telecommunications and image processing.	17. CISC architecture is used in low-end applications such as security systems, home automation, etc.

13. Code expansion can be a problem	13. Code expansion is not a problem
14. Decoding of instructions is simple.	14. Decoding of instructions is complex
15. It does not require external memory for calculations	15. It requires external memory for calculations
16. The most common RISC microprocessors are Alpha, ARC, ARM, AVR, MIPS, PA-RISC, PIC, Power Architecture, and SPARC.	16. Examples of CISC processors are the System/360, VAX, PDP-11, Motorola 68000 family, AMD and Intel x86 CPUs.

THANK YOU