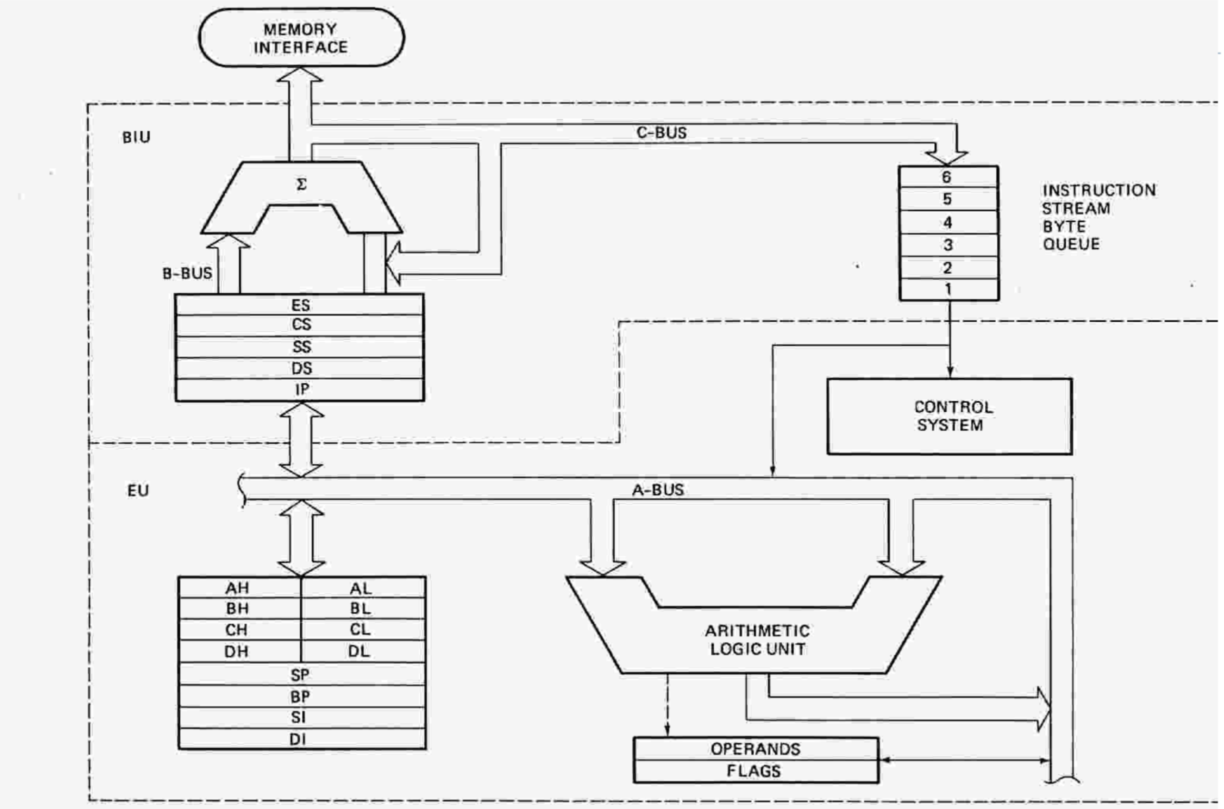
16-Bit 8086 Intel Processor Architecture



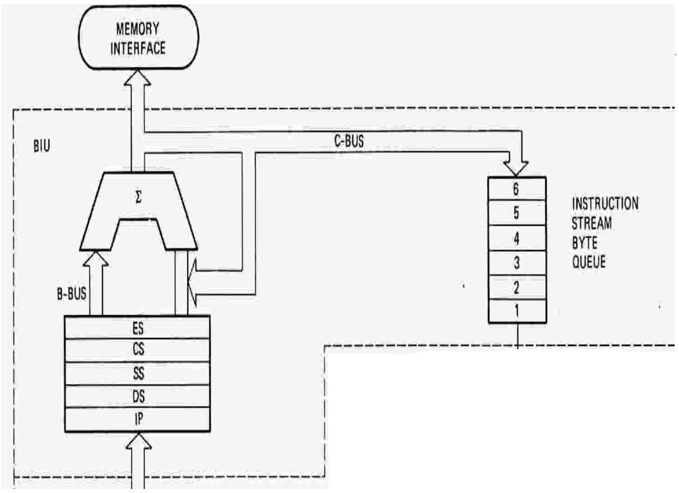
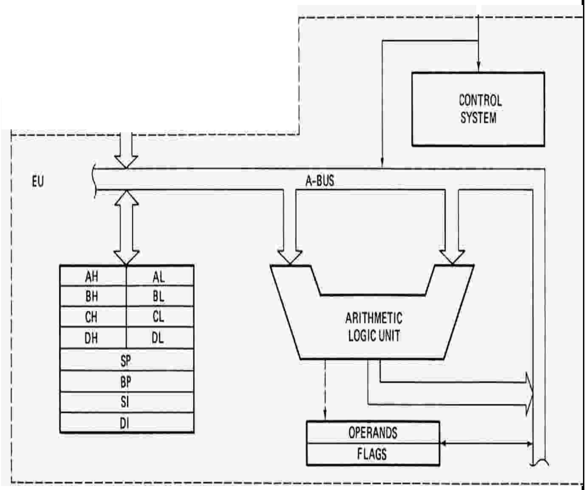
Microprocessor 8086 is internally divided into two separate functional units. These are:

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

Q. Why are there two partitions in 8086 Architecture?

=> For parallel processing/ pipelining of multiple instructions.

	1s	2s	3s	4s	5s
Instruction 1:	F	D	E		
Instruction 2:		F	D	E	
Instruction 3:			F	D	E

BIU (BUS Interfacing Unit)	**EU (Execution Unit)**
BIU  MEMORY INTERFACE BIU C-BUS Σ B-BUS ES CS SS DS IP 6 5 4 3 2 1 INSTRUCTION STREAM BYTE QUEUE	**EU**  CONTROL SYSTEM EU A-BUS AH AL BH BL CH CL DH DL SP BP SI DI ARITHMETIC LOGIC UNIT OPERANDS FLAGS
BIU facilitates communication between the EU & the memory or I/O circuits. **Responsible for transmitting addresses, data, and control signals on the buses.**	**The execution unit of Internal Architecture of 8086 tells the BIU from where to fetch instructions or data, decodes instructions and executes instructions.**
<u>Main components of BIU:</u> ● It provides a full 16-bit bi-directional data bus and 20-bit address bus. ● Registers (CS, DS, ES, SS, and IP) hold addresses of memory locations. ● IP (instruction pointer) contains the address of the next instruction to be executed by the EU.	**<u>Main components of EU:</u>** ● Control Circuitry ● Instruction Decoder ● Arithmetic Logic Unit (ALU) ● Flag Register ● General Purpose Registers ● Pointers and Index Registers ● Also has address bus (20 bits) and data bus (16 bits)
Main work - Fetch [Directly fetches data/instruction by calculating physical addresses from segments and offsets]	**Main Work - Decode and execute** [Directly decodes and executes instructions. Also facilitates fetching since in EU there are a few offset registers]

Operations of BIU

The bus interface unit is responsible for performing all external bus operations.

- It sends the address of the memory or I/O.
- It fetches instruction from memory.
- It reads data from port/memory.
- It Writes data into port/memory.
- It supports instruction queuing.
- It provides the address relocation facility.

Operations of EU

- Fetches instructions from the Queue in BIU, decodes and executes arithmetic and logic operations using the ALU.

- Sends control signals for internal data transfer operations within the microprocessor.

- Sends request signals to the BIU to access the external module.

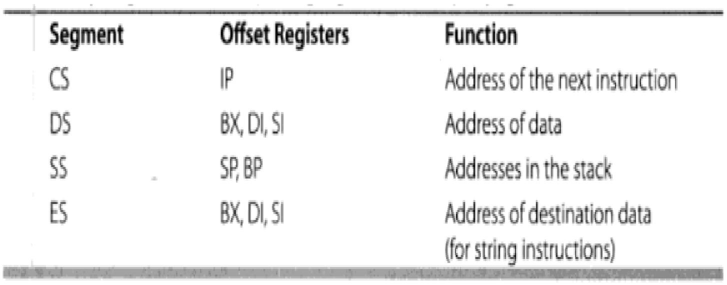
- It operates with respect to T-states (clock cycles) and not machine cycles.

Physical address calculation:

To do all the fetching/read/write from MEM/IO or to MEM/IO BIU calculates Physical address. To generate a 20 bit physical address Segment Registers (CS, DS, SS, ES) and Pointer and Index Registers (IP, SP, BP, SI, DI) are needed.

Formula:

Physical address = segment no x 10h + offset



Segment	Offset Registers	Function
CS	IP	Address of the next instruction
DS	BX, DI, SI	Address of data
SS	SP, BP	Addresses in the stack
ES	BX, DI, SI	Address of destination data (for string instructions)

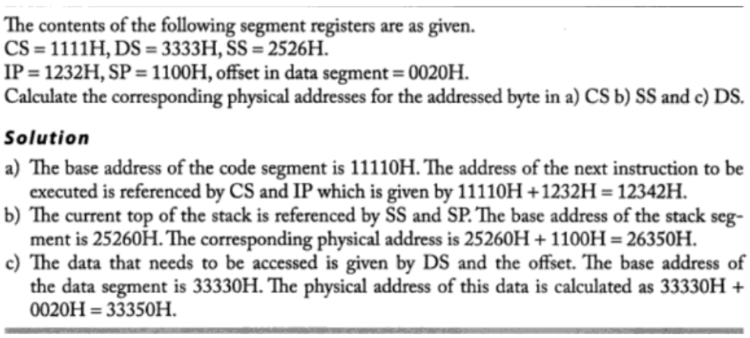
Example:

EU's ALU for execution:

This unit performs all arithmetic and logical computations.

Instructions those are fetched and decoded, are executed in the ALU

Thus, ALU has direct access to General Purpose Registers and Flags.

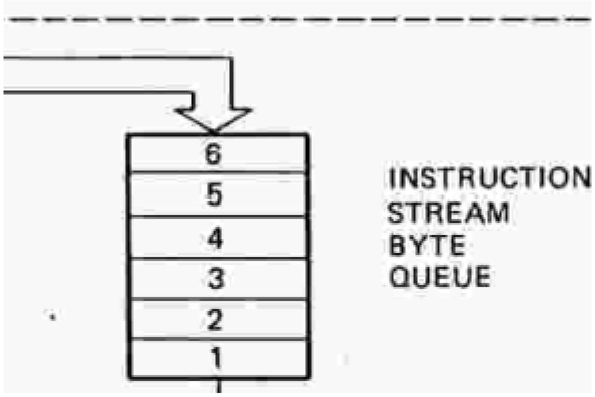


The contents of the following segment registers are as given.
CS = 1111H, DS = 3333H, SS = 2526H.
IP = 1232H, SP = 1100H, offset in data segment = 0020H.
Calculate the corresponding physical addresses for the addressed byte in a) CS b) SS and c) DS.

Solution

a) The base address of the code segment is 11110H. The address of the next instruction to be executed is referenced by CS and IP which is given by 11110H + 1232H = 12342H.
b) The current top of the stack is referenced by SS and SP. The base address of the stack segment is 25260H. The corresponding physical address is 25260H + 1100H = 26350H.
c) The data that needs to be accessed is given by DS and the offset. The base address of the data segment is 33330H. The physical address of this data is calculated as 33330H + 0020H = 33350H.

Instruction Queue/ Prefetch Queue



*Each slot can contain 1 byte of information. Therefore an instruction queue of 8086 microprocessors can contain 6 byte instructions.

Q. When does the instruction queue refill?
=> when 2 byte instructions are completed executing only then new instructions can come. Since the majority of the instruction size is more than 1 byte.

*Follows FIFO structure meaning the instruction that has been fetched first will be passed to the EU for execution first.

*Instruction Queue will fail its FIFO structure when **JMP or CALL** instruction is executed.

- Every processor has some **temporary registers** those are not visible to the programmers.For these registers, our data are not lost in between different kinds of operations,

  XCNG CX, BX
  MOV temp, CX
  MOV CX, BX
  MOV BX, temp

- **FLAG Registers:** A flag is a flip–flop which indicates some condition produced by the execution of an instruction or controls certain operations of the EU.