8086 Microprocessors

Angel Deborah S

SSN College of Engineering

angeldeborahs@ssn.edu.in

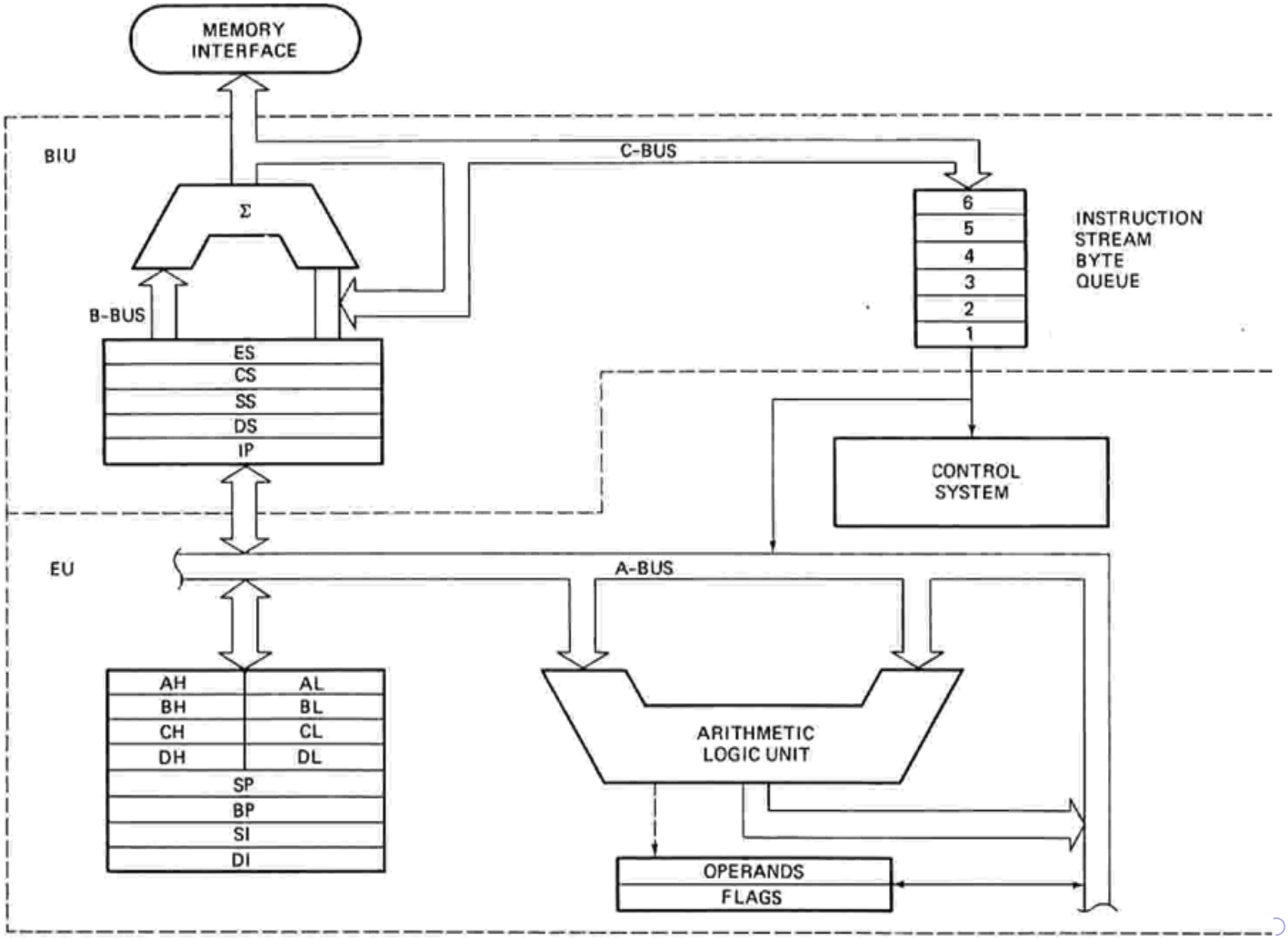
December 15, 2017

Overview

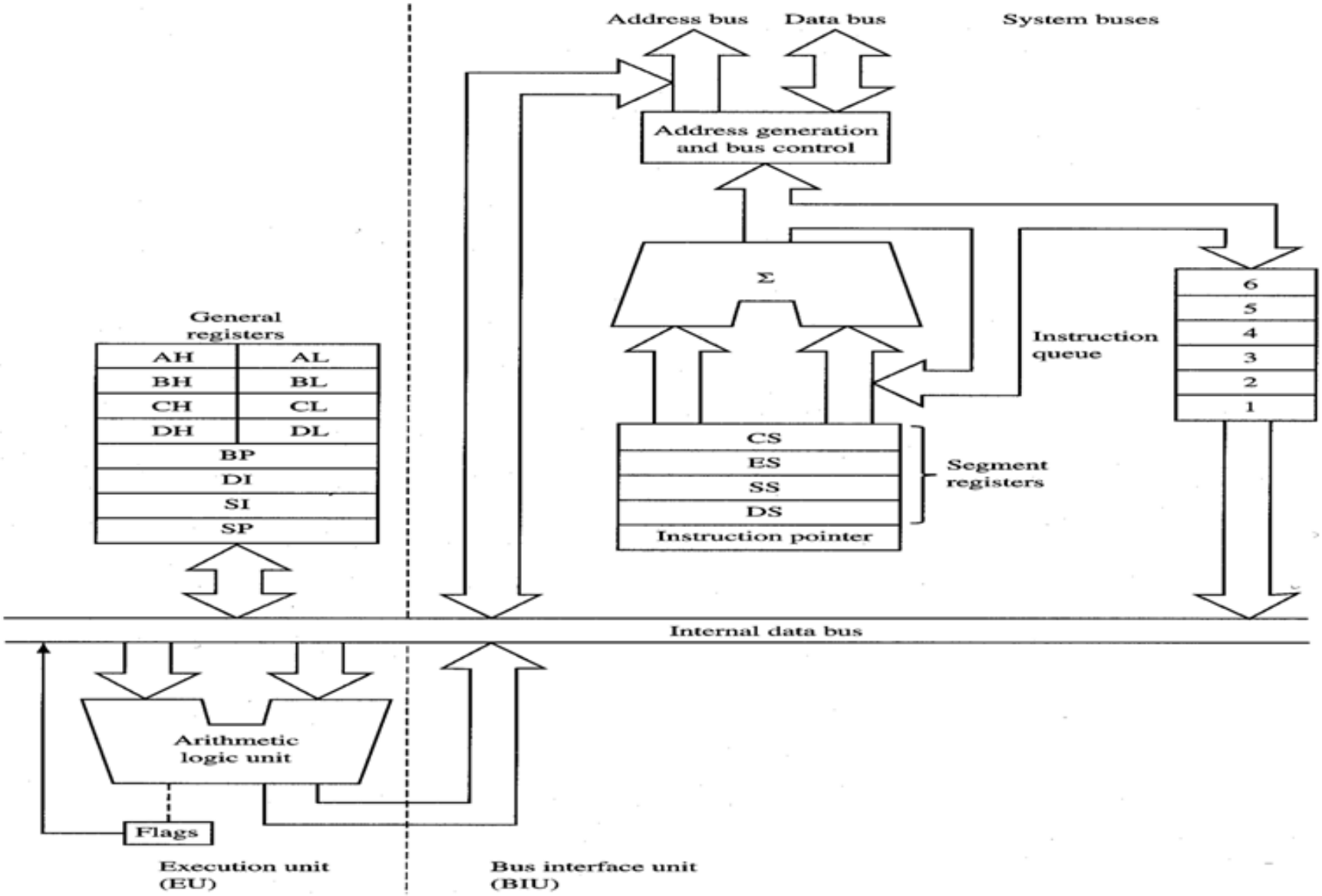
Features of 8086

- It is a 16 bit microprocessors.
- 8086 has a 20 bit address bus, can access upto 2^{20} memory locations (1 MB) .
- It provides fourteen 16-bit registers.
- It has multiplexed address and data bus AD0-AD15 and A16-A19.
- Multiplexed bus reduces number of pins but slow down the transfer of data.
- Perform bit , byte, word and block operations
- Two modes- minimum (single processor), maximum (multiprocessor)
- Supports multiprogramming
- Instruction Queue
- Different addressing modes

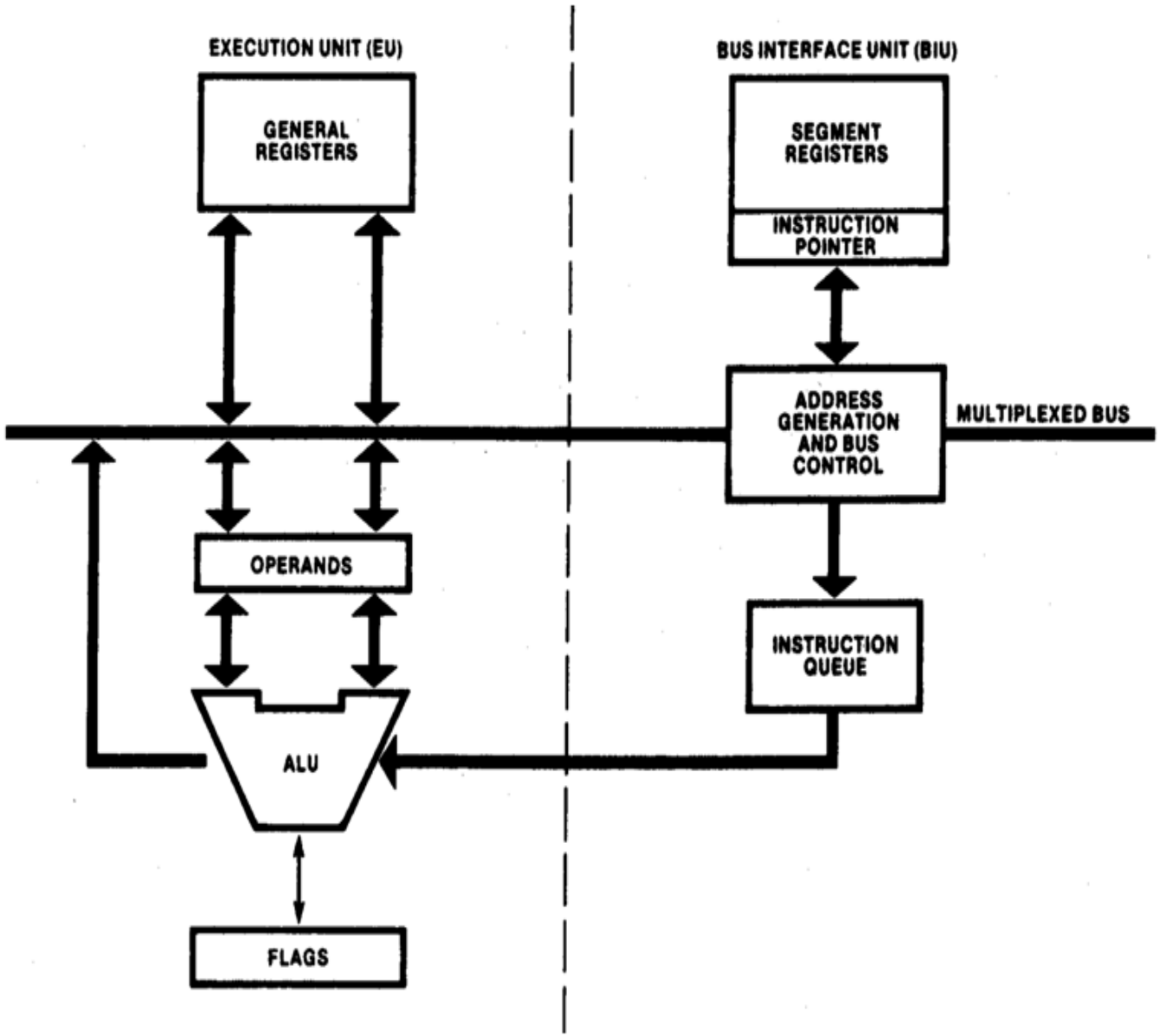
Block Diagram of 8086



Block Diagram of 8086



Execution and bus interface units



Execution Unit

It contains:

- Control circuitry
- Instruction Decoder
- Arithmetic Logic Unit
- Flag Register
- General Purpose Registers
- Pointers and Index Registers.

Bus Interface Unit

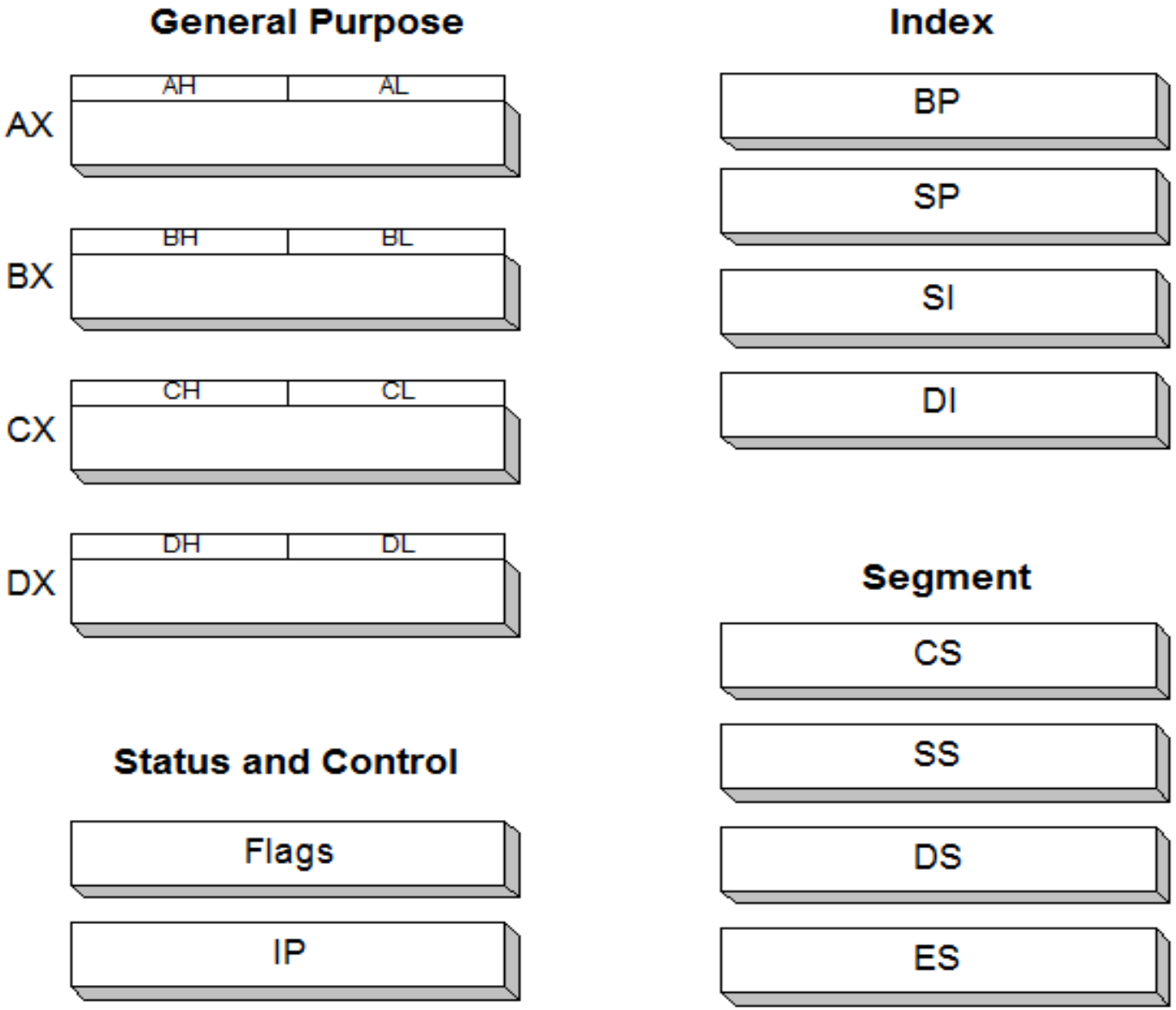
Functions of BIU:

- Sends address of the memory or I/O.
- Fetches instruction from memory.
- Reads data from port/memory
- Writes data into port/memory.
- Supports instruction queuing.
- Provides the address relocation facility.

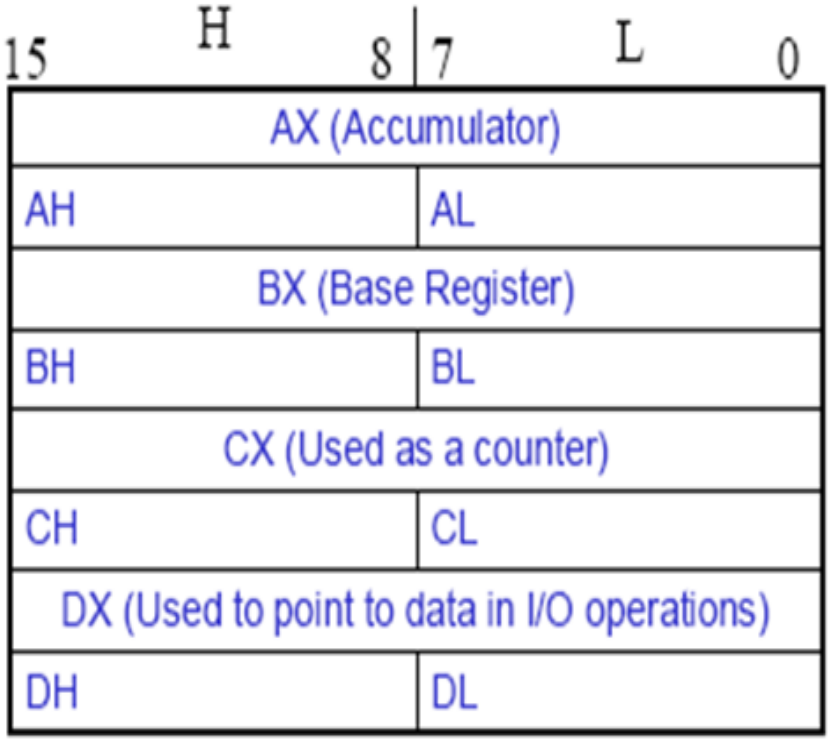
To implement these functions the BIU contains:

- Instruction queue
- segment registers
- Instruction pointer
- Address summer
- Bus control logic

8086 Registers



General Purpose Registers



- Normally used for storing temporary results
- Each of the registers is 16 bits wide (AX, BX, CX, DX)
- Can be accessed as either 16 or 8 bits AX, AH, AL

General Purpose Registers

AX

- Accumulator Register
- Preferred register to use in arithmetic, logic and data transfer instructions because it generates the shortest Machine Language Code
- Must be used in multiplication and division operations
- Must be used in I/O operations

BX

- Base Register
- Serves as an address register

General Purpose Registers

CX

- Count register
- Used as a loop counter
- Used in shift and rotate operations

DX

- Data register
- Used in multiplication and division
- Also used in I/O operations

Pointer and Index Registers

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

- All 16 bits wide, L/H bytes are not accessible
- 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

Pointer and Index Registers

- Contain the offset addresses of memory locations
- Can be used in arithmetic and other operations
- SP: Stack pointer
  - Used with SS to access the stack segment
- BP: Base Pointer
  - Primarily used to access data on the stack
  - Can be used to access data in other segments
- SI: Source Index register
  - Required for some string operations
  - When string operations are performed, the SI register points to memory locations in the data segment which is addressed by the DS register. Thus, SI is associated with the DS in string operations.

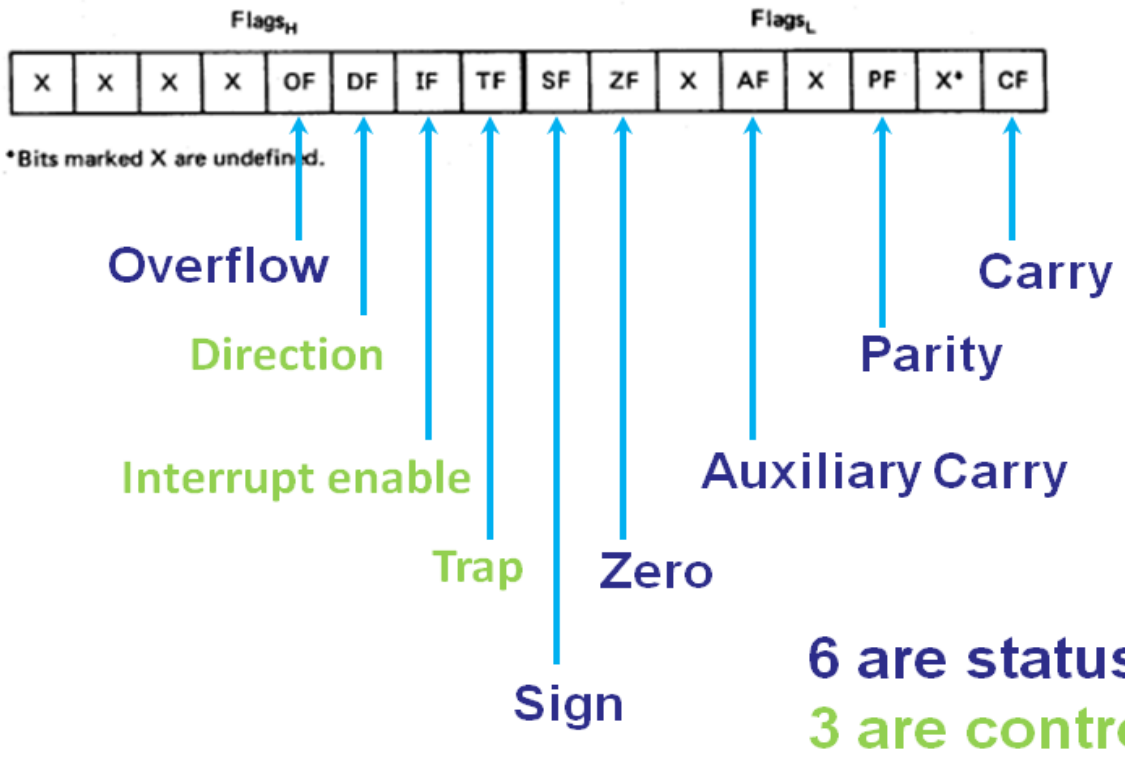
Pointer and Index Registers

- SI: Source Index register
  - Required for some string operations
  - When string operations are performed, the SI register points to memory locations in the data segment which is addressed by the DS register. Thus, SI is associated with the DS in string operations.
- DI: Destination Index register
  - Required for some string operations
  - When string operations are performed, the DI register points to memory locations in the data segment which is addressed by the ES register. Thus, DI is associated with the ES in string operations.
- The SI and the DI registers may be used to access data stored in arrays

Segment Register

- CS, DS, SS and ES
- Stores the memory addresses of instructions and data
- Memory Organization
  1. Each byte in memory has a 20 bit address starting with 0 to 220-1 or 1 meg of addressable memory
  2. Addresses are expressed as 5 hex digits from 00000 - FFFFF
  3. Problem: But 20 bit addresses are TOO BIG to fit in 16 bit registers!
  4. Solution: Memory Segment
     - Block of 64K (65,536) consecutive memory bytes
     - A segment number is a 16 bit number
     - Segment numbers range from 0000 to FFFF
     - Within a segment, a particular memory location is specified with an offset
     - An offset also ranges from 0000 to FFFF

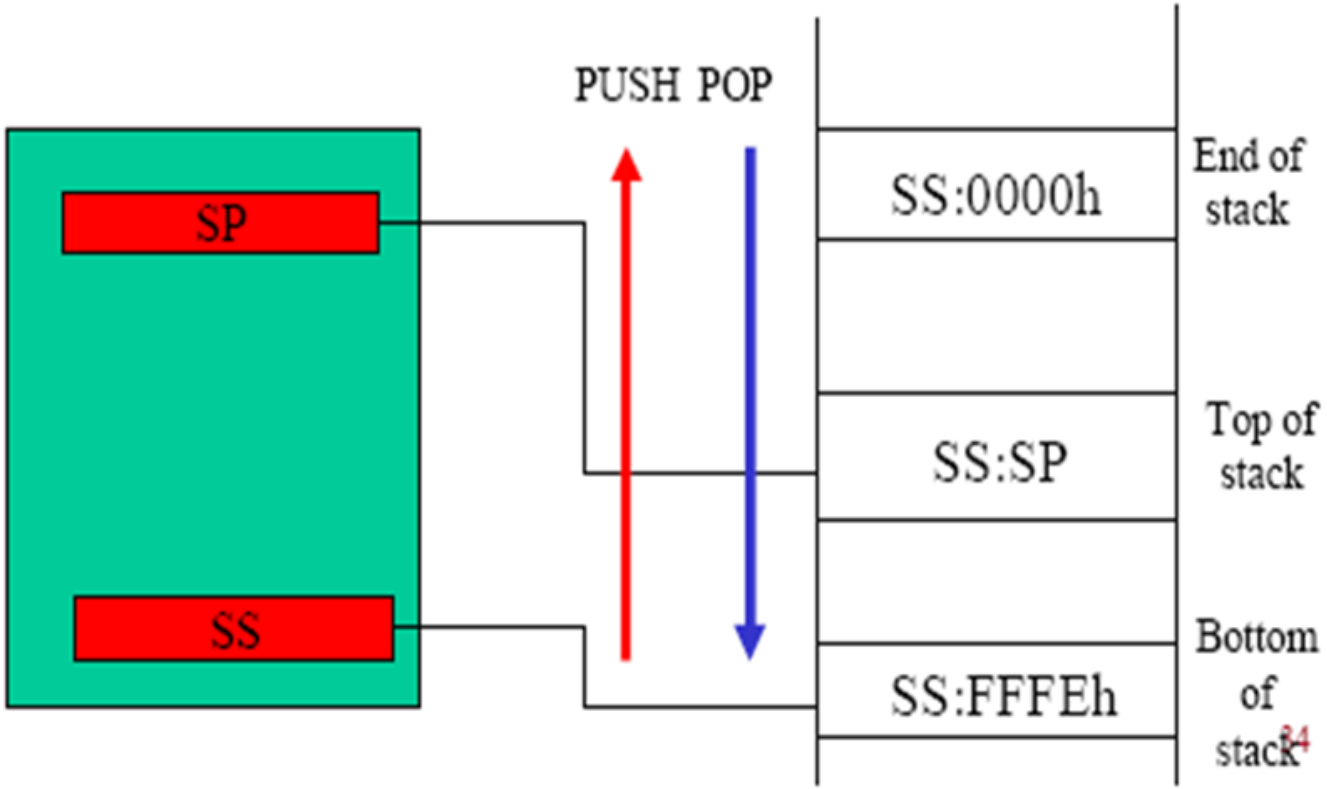
Flag Register



The Stack

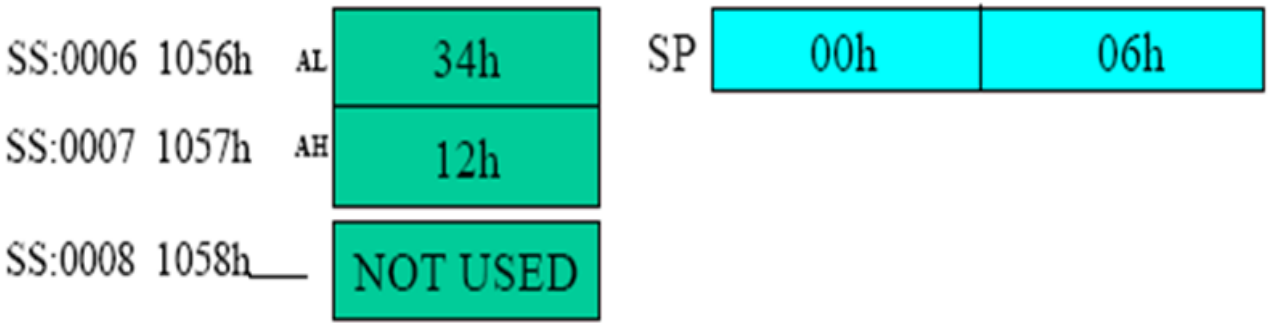
- The stack is used for temporary storage of information such as data or addresses.
- When a CALL is executed, the 8086 automatically PUSHes the current value of CS and IP onto the stack.
- Other registers can be pushed.
- Before return from the subroutine, POP instructions can be used to pop values back from the stack into the corresponding registers.

The Stack operation



Push Operation

- Given
  - SS = 0105h
  - SP = 0008h
  - AX = 1234h
  - What is the outcome of the PUSH AX instruction?
- A_{BOS} = 01050 + FFFEh = 1104h
- A_{TOS} = 01050 + 0008h = 1058h
- Decrement the SP by 2 and write AX into the word location 1056h.



Pop Operation

- What is the outcome of the following
  - POP AX
  - POP BX
  - if originally 1058h contained AABBh?

1056	34
1057	12
1058	BB
1059	AA

- Read into the specified register from the stack and increment the stack pointer for each POP operation
- At the first POP
  - –AX = 1234h SP = 0008h
- At the second POP
  - –BX = AABBh

Question? What is SP and Physical address now? →

SP = 000Ah and
Physical Address: 105A