Lecture 3

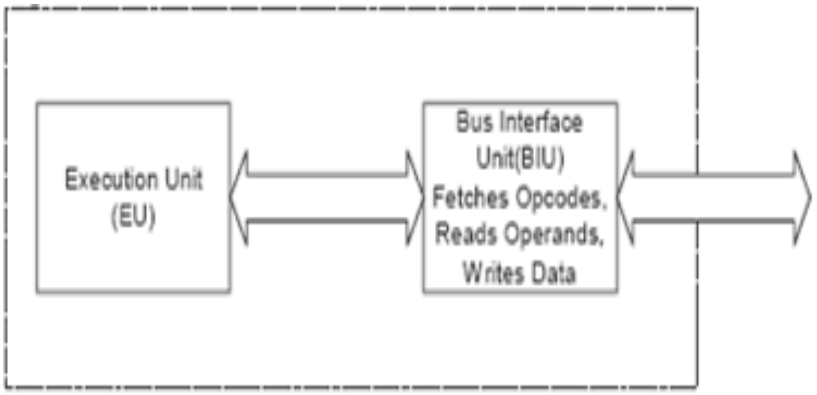
8086 Microprocessor:

Architecture of 8086:

- The architecture of 8086 includes
  - Arithmetic Logic Unit (ALU)
  - Flags
  - General registers
  - Instruction byte queue
  - Segment registers

EU & BIU:

- The 8086 CPU logic has been partitioned into two functional units namely Bus Interface Unit (BIU) and Execution Unit (EU)
- The major reason for this separation is to increase the processing speed of the processor
- The BIU has to interact with memory and input and output devices in fetching the instructions and data required by the EU
- EU is responsible for executing the instructions of the programs and to carry out the required processing



BUS INTERFACE UNIT (BIU):

The BIU performs all bus operations for EU .

Fetching instructions

Responsible for executing all external bus cycles.

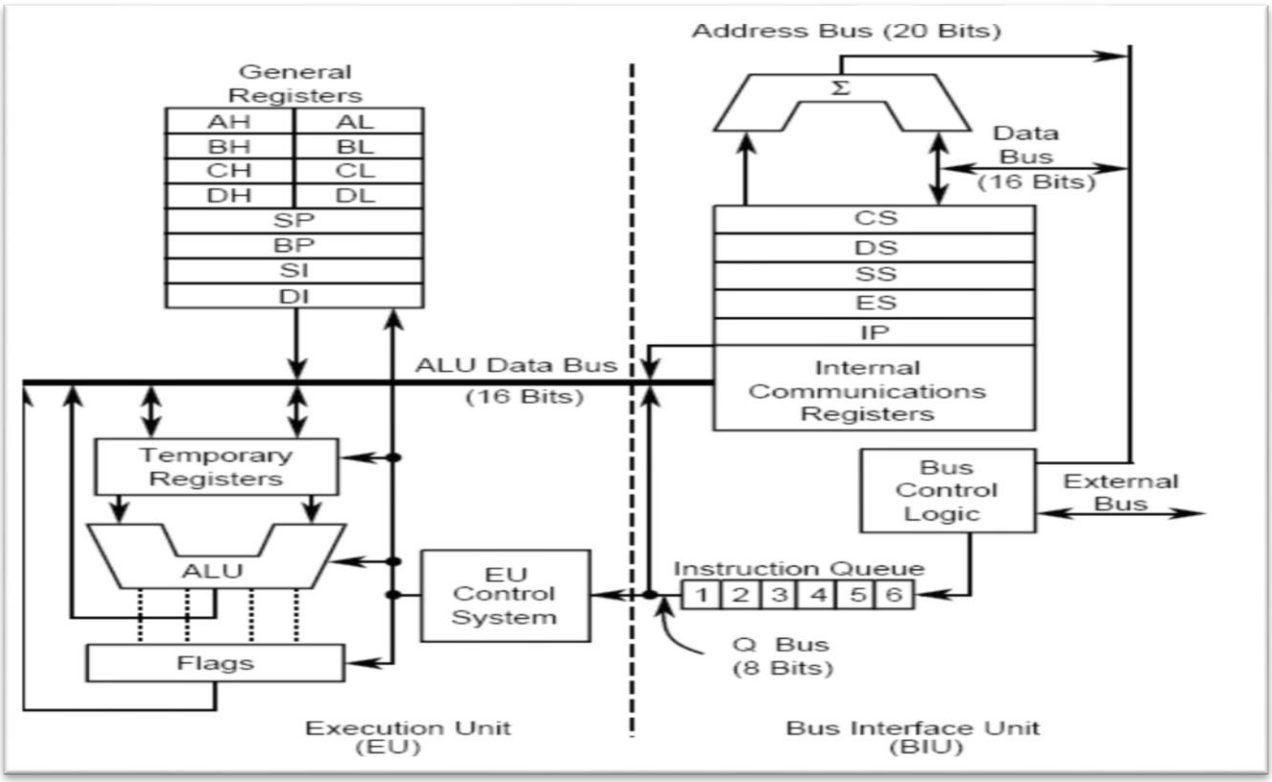
Read operands and write result.

EXECUTION UNIT (EU)

Execution unit contains the complete infrastructure required to execute an instruction.

م.م اثير هادي عيسى الرماحي

Architecture Diagram



❖ **Execution Unit:**

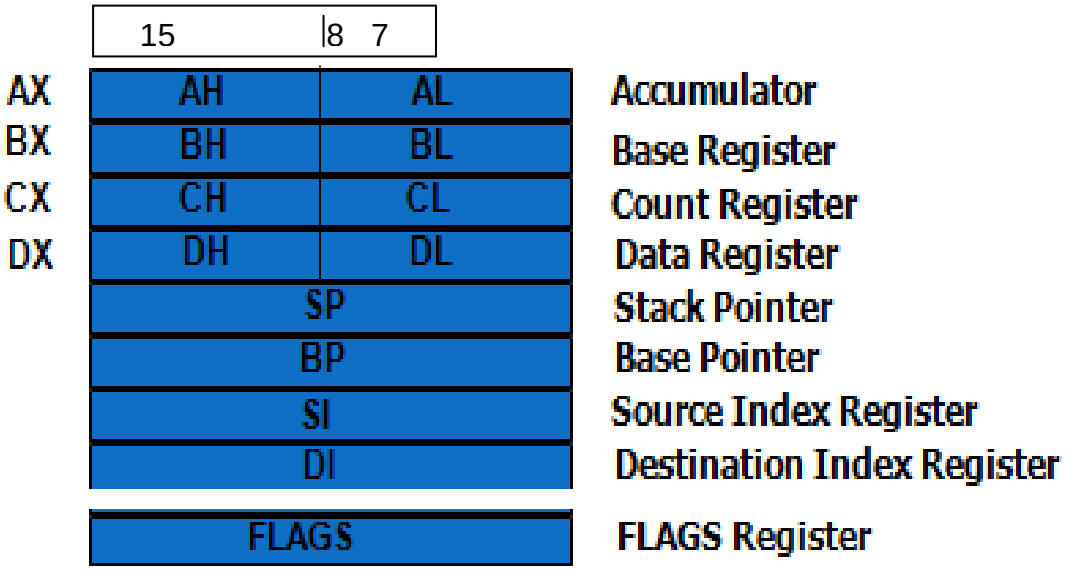
- The Execution Unit (EU) has
  - Control unit
  - Instruction decoder
  - Arithmetic and Logical Unit (ALU)
  - General registers
  - Flag register
  - Pointers
  - Index registers
- Control unit is responsible for the co-ordination of all other units of the processor
- ALU performs various arithmetic and logical operations over the data
- The instruction decoder translates the instructions fetched from the memory into a series of actions that are carried out by the EU

➢ **Execution Unit - Registers**

- General registers are used for temporary storage and manipulation of data and instructions
- Accumulator register consists of two 8-bit registers AL and AH, which can be combined together and used as a 16-bit register AX
- Accumulator can be used for I/O operations and string manipulation

م.م اثير هادي عيسى الرماحي

- Base register consists of two 8-bit registers BL and BH, which can be combined together and used as a 16-bit register BX
- BX register usually contains a data pointer used for based, based indexed or register indirect addressing. Similar to 8085 H-L register.
- Count register consists of two 8-bit registers CL and CH, which can be combined together and used as a 16-bit register CX
- Count register can be used as a counter in string manipulation and shift/rotate instructions
- Data register consists of two 8-bit registers DL and DH, which can be combined together and used as a 16-bit register DX
- Data register can be used to hold 16 bit result in 16 in 16x16 multiplication.

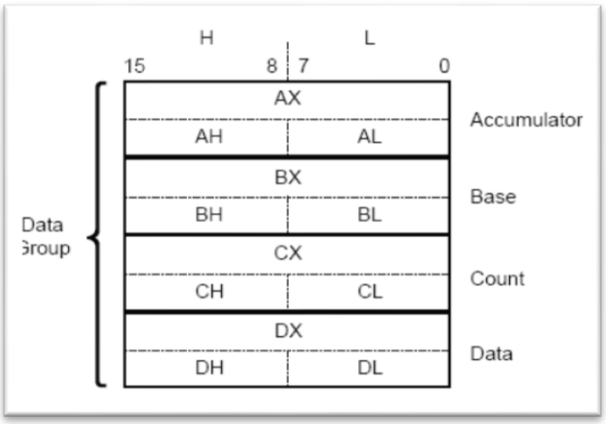


Pointer Registers

- Stack pointer and BP are used to access data in the stack segment.
- SP is used as an offset from the current SS during execution of instructions that involve the stack segment in external memory.
- BP is used in based addressing mode.

Index Register

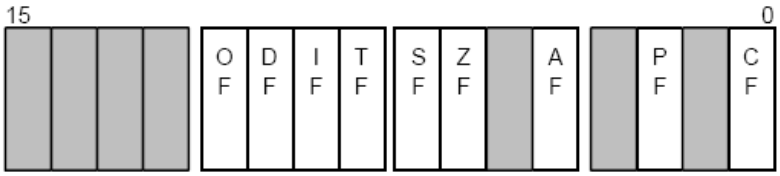
- Source index register (SI) and Destination Index Registers are used in indexed addressing.



م.م اثير هادي عيسى الرماحي

➢ Execution Unit – Flags:

Register Name: Processor Status Word
Register Mnemonic: PSW (FLAGS)
Register Function: Posts CPU status information.



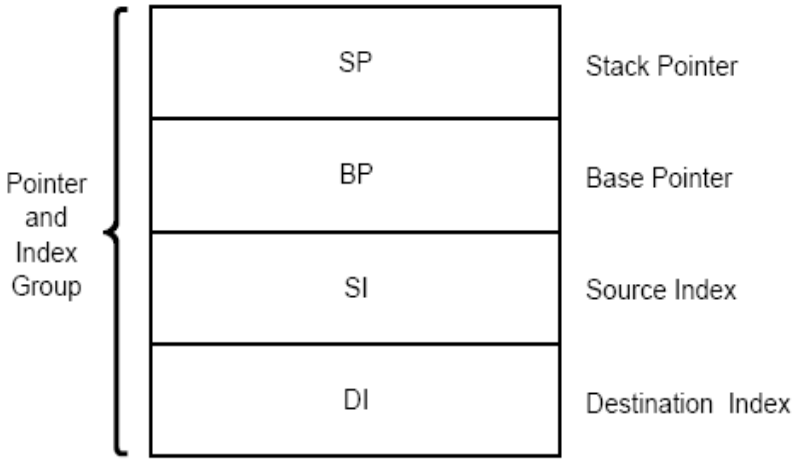
- Overflow Flag (OF) - set if the size of the exceeds the capacity of the destination location.
- Direction Flag (DF) – It is used with string operations. When set , it causing string instructions to auto – decrement or to process strings from right to left.
- Interrupt-enable Flag (IF) - setting this bit enables maskable interrupts . When IF = 0 , all maskable interrupt are disable.
- Single-step Flag (Trap F) – put 8086 in the single step mode.
- Sign Flag (SF) - set if the most significant bit of the result is one.
- Zero Flag (ZF) - set if the result is zero.
- Auxiliary carry Flag (AF) - set if there was a carry from or borrow to bits 0-3 in the AL register.
- Parity Flag (PF) - set if parity (the number of "1" bits) in the low-order byte of the result is even.
- Carry Flag (CF) - set if there was a carry from or borrow to the most significant bit during last result calculation

Bit Mnemonic	Bit Name	Reset State	Function
OF	Overflow Flag	0	If OF is set, an arithmetic overflow has occurred.
DF	Direction Flag	0	If DF is set, string instructions are processed high address to low address. If DF is clear, strings are processed low address to high address.
IF	Interrupt Enable Flag	0	If IF is set, the CPU recognizes maskable interrupt requests. If IF is clear, maskable interrupts are ignored.
TF	Trap Flag	0	If TF is set, the processor enters single-step mode.
SF	Sign Flag	0	If SF is set, the high-order bit of the result of an operation is 1, indicating it is negative.
ZF	Zero Flag	0	If ZF is set, the result of an operation is zero.
AF	Auxiliary Flag	0	If AF is set, there has been a carry from the low nibble to the high or a borrow from the high nibble to the low nibble of an 8-bit quantity. Used in BCD operations.
PF	Parity Flag	0	If PF is set, the result of an operation has even parity.
CF	Carry Flag	0	If CF is set, there has been a carry out of, or a borrow into, the high-order bit of the result of an instruction.

م.م اثير هادي عيسى الرماحي

➢ **Execution Unit – Pointers**

- Stack Pointer (SP) is a 16-bit register pointing to program stack
- Base Pointer (BP) is a 16-bit register pointing to data in stack segment. BP register is usually used for based, based indexed or register indirect addressing.
- Source Index (SI) is a 16-bit register. SI is used for indexed, based indexed and register indirect addressing, as well as a source data addresses in string manipulation instructions.
- Destination Index (DI) is a 16-bit register. DI is used for indexed, based indexed and register indirect addressing, as well as a destination data addresses in string manipulation instructions.



❖ **Bus Interface Unit:**

The BIU has

- Instruction stream byte queue
- A set of segment registers
- Instruction pointer

➢ **BIU – Instruction Byte Queue**

- 8086 instructions vary from 1 to 6 bytes
- Therefore fetch and execution are taking place concurrently in order to improve the performance of the microprocessor
- The BIU feeds the instruction stream to the execution unit through a 6 byte prefetch queue
- Execution and decoding of certain instructions do not require the use of buses
- While such instructions are executed, the BIU fetches up to six instruction bytes for the following instructions (the subsequent instructions)
- The BIU store these prefetched bytes in a first-in-first out register by name instruction byte queue
- When the EU is ready for its next instruction, it simply reads the instruction byte(s) for the instruction from the queue in BIU

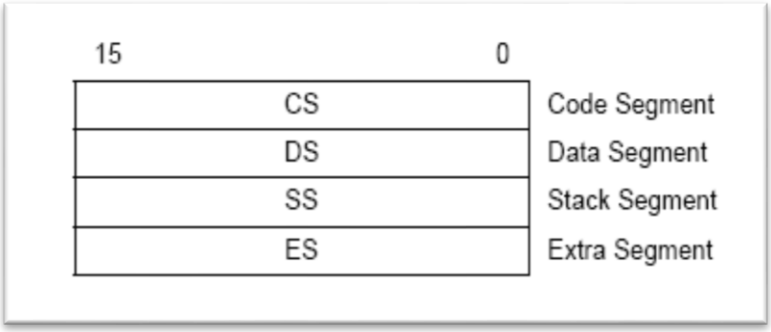
م.م اثير هادي عيسى الرماحي

- **Segment: Offset Notation:**
  - The total addressable memory size is 1MB
  - Most of the processor instructions use 16-bit pointers the processor can effectively address only 64 KB of memory
  - To access memory outside of 64 KB the CPU uses special segment registers to specify where the code, stack and data 64 KB segments are positioned within 1 MB of memory
  - A simple scheme would be to order the bytes in a serial fashion and number them from 0 (or 1) to the end of memory
  - The scheme used in the 8086 is called segmentation
  - Every address has two parts, a SEGMENT and an OFFSET (Segmnet:Offset)
  - The segment indicates the starting of a 64 kilobyte portion of memory, in multiples of 16
  - The offset indicates the position within the 64k portion
  - Absolute address = (segment * 16) + offset
- **Segment Registers:**

  The memory of 8086 is divided into 4 segments namely
  - Code segment (program memory)
  - Data segment (data memory)
  - Stack memory (stack segment)
  - Extra memory (extra segment)
  - Code Segment (CS) register is a 16-bit register containing address of 64 KB segment with processor instructions
  - The processor uses CS segment for all accesses to instructions referenced by instruction pointer (IP) register
  - Stack Segment (SS) register is a 16-bit register containing address of 64KB segment with program stack
  - By default, the processor assumes that all data referenced by the stack pointer (SP) and base pointer (BP) registers is located in the stack segment
  - By default, the processor assumes that all data referenced by general registers (AX, BX, CX, DX) and index register (SI, DI) is located in the data segment
  - Extra Segment (ES) register is a 16-bit register containing address of 64KB segment, usually with program data
  - By default, the processor assumes that the DI register references the ES segment in string manipulation instructions
- **Different Areas in Memory:**
  - Program memory – Program can be located anywhere in memory
  - Data memory – The processor can access data in any one out of 4 available segments

م.م اثير هادي عيسى الرماحي

- Stack memory – A stack is a section of the memory set aside to store addresses and data while a subprogram executes
- Extra segment – This segment is also similar to data memory where additional data may be stored and maintained
- Data Segment (DS) register is a 16-bit register containing address of 64KB segment with program data



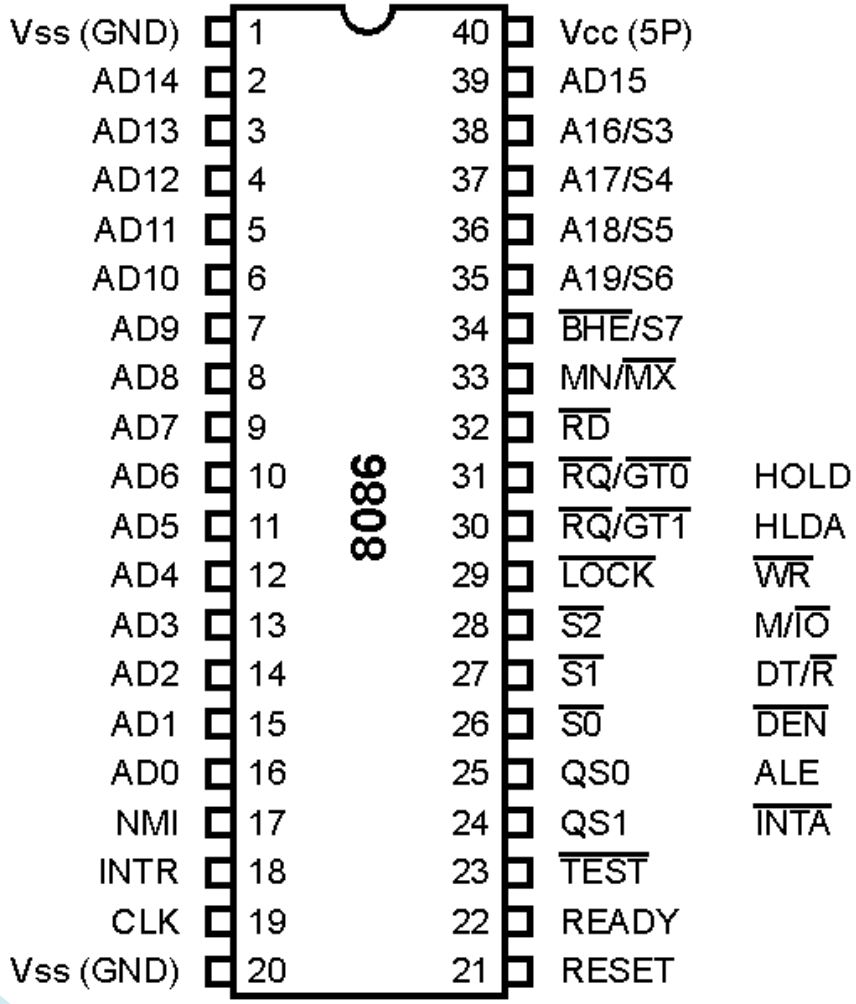
If a location 109F0 of Code Segment is to be addressed to fetch An instruction, the physical address will be calculated as follows

CSR = 010A

IP = F950

Effective Address = 109F0

Pin Diagram of Intel 8086:



م.م اثير هادي عيسى الرماحي