Architecture of INTEL 8086

- 8086 has a pipelined architecture
- In pipelined architecture, the processor will have a number of functional units and execution time of each functional unit overlaps.
- Each functional unit works independently most of the time.
- The architecture of 8086 can be internally divided into two separate functional units: Bus Interface Unit (BIU) and Execution Unit (EC)

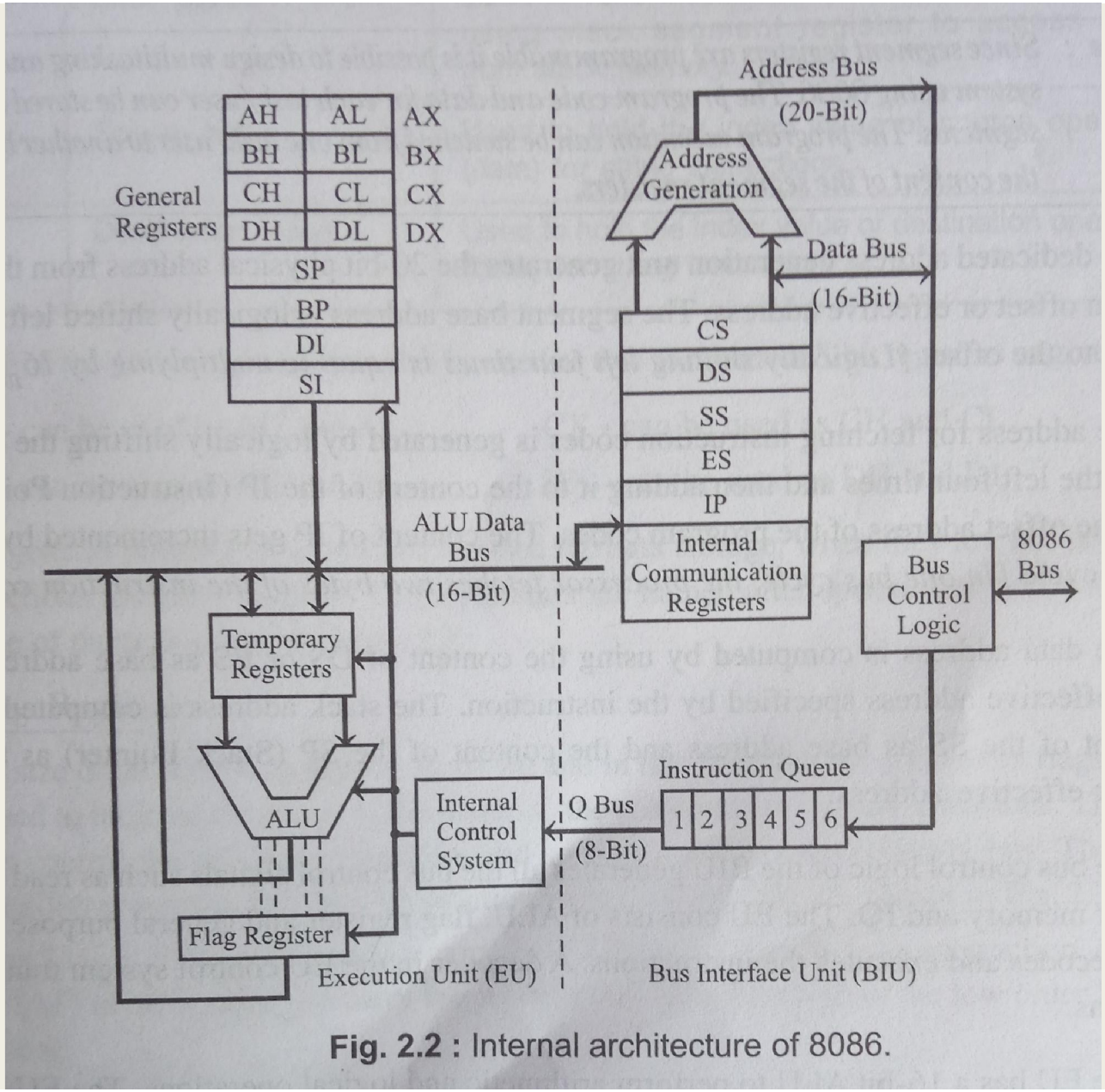

Fig. 2.2 : Internal architecture of 8086.

- BIU
  - *fetches instructions*
  - *reads data from memory and I/O ports*
  - *Writes data to memory and I/O ports*
- BIU Contains
  - *Segment registers*
  - *Instruction Pointer*
  - *Instruction queue*
  - *Address generation unit*
  - *Bus control unit*
- EU executes instructions that have already been fetched by the BIU
- BIU and EU function independently

- Instruction queue is a FIFO (First-In-First-Out) group of registers
- The size of the queue is 6 bytes
- The BIU fetches instruction code from the memory and stores it in the queue
- The EU fetches instruction code from the queue
- The BIU has four number of 16-bit registers
- They are
  - *Code Segment (CS)*
  - *Data Segment (DS)*
  - *Stack Segment (SS)*
  - *Extra Segment (ES)*

- The 8086 memory space can be divided into segments of 64 KB
- The 4 segment registers are used to hold four segment base address
- Hence 8086 can directly address 4 segments of 64Kb at any time instant.
- This feature of 8086 allows the system designer to allocate separate areas for storing program codes and data
- The contents of segment registers are programmable

- The address generation unit generates the 20-bit physical address from the segment base and an offset or effective address
- The segment address is logically shifted left four times and added to the offset
- The address for fetching instruction codes is generated by logically shifting the contents of CS to the left four times and then adding it to the contents of the IP.
- The IP holds the offset address of the program codes
- The content of IP gets incremented by two after every bus cycle [In one bus cycle, the processor fetches two bytes of the instruction code]

- The data address is computed by using the contents of DS/ES as base address and an offset or effective address specified by the instruction
- The stack address is computed by using the content of the SS as base address and the content of the SP as the offset address or effective address
- The bus control logic of the BIU generates all the bus control signals such as read and write signals for memory and IO
- The EU consists of ALU, flag register and general purpose registers
- The EU decodes and executes the instructions
- A decoder in the EU control system translates the instructions
- The EU has a 16-bit ALU to perform arithmetic and logic operations.
- The EU has 8 numbers of 16-bit general purpose registers

TABLE - 2.7 : SPECIAL FUNCTIONS OF 8086 REGISTERS

Register	Name of the register	Special function
AX	16-bit Accumulator	Stores the 16-bit result of certain arithmetic and logical operations.
AL	8-bit Accumulator	Stores the 8-bit result of certain arithmetic and logical operations.
BX	Base register	Used to hold the base value in base addressing mode to access memory data.
CX	Count register	Used to hold the count value in SHIFT, ROTATE and LOOP instructions.
DX	Data register	Used to hold data for multiplication and division operations.
SP	Stack pointer	Used to hold the offset address of top of stack memory.
BP	Base pointer	Used to hold the base value in base addressing using stack segment register to access data from stack memory.
SI	Source index	Used to hold the index value of source operand (data) for string instructions.
DI	Destination index	Used to hold the index value of destination operand (data) for string instructions.

Some of the 16-bit registers can also be used as two numbers of 8-bit registers as given below:

AX - can be used as AH and AL ; CX - can be used as CH and CL

BX - can be used as BH and BL ; DX - can be used as DH and DL

Multiplexing

- Transferring different information at different well-defined times through the same lines.
- A group of such line is called a multiplexed bus
- So need lesser no. of pins to communicate
- Most of data lines are multiplexed with some or all address lines to form an address/data bus
- The status signals emitted by the microprocessor are sometimes multiplexed either with the data lines or with some of the address lines
- Necessary hardware is required to demultiplex those lines

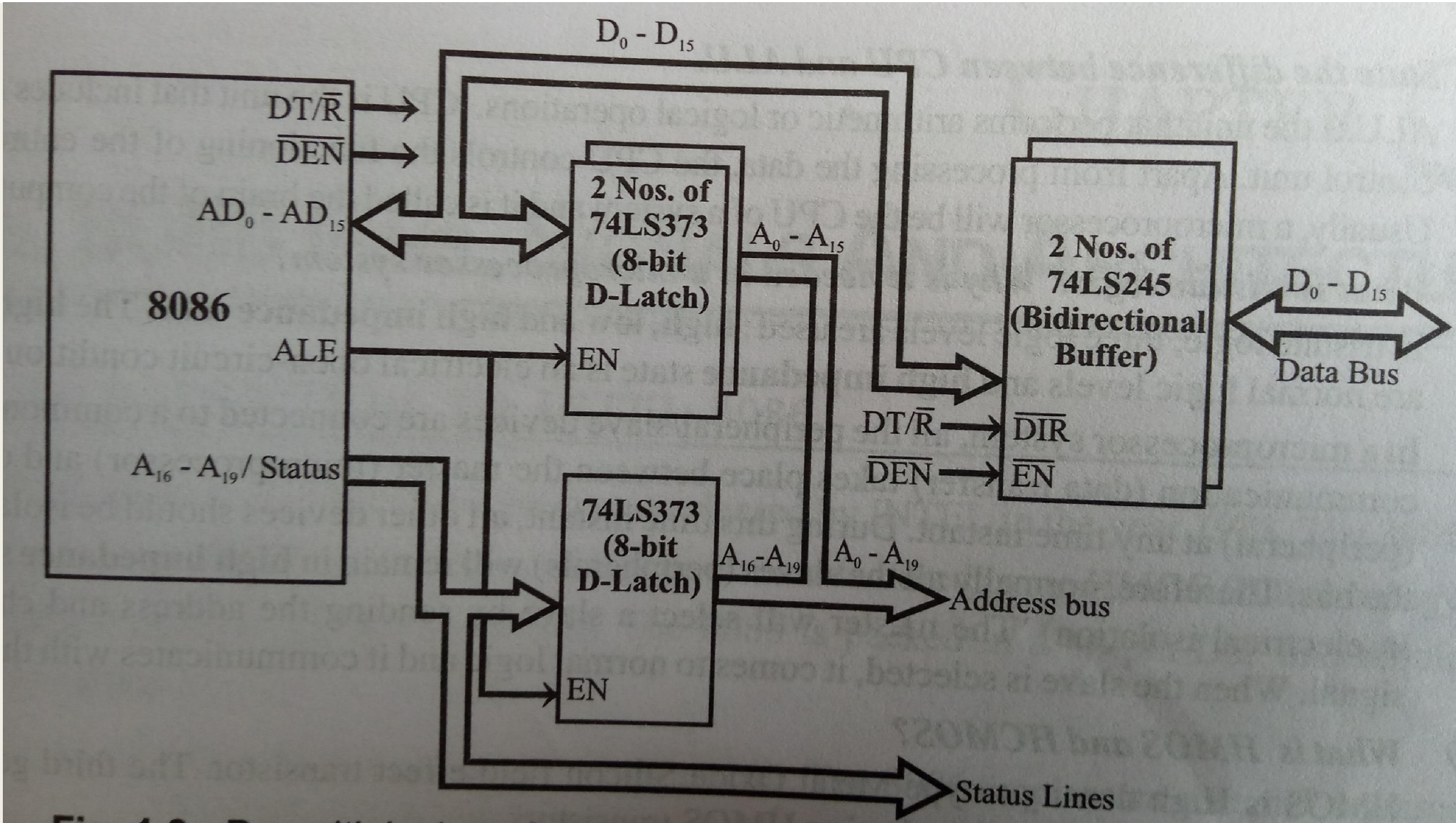

Fig. 1.3 : Demultiplexing of address and data lines in an 8086 processor.

Demultiplexing in 8086 Processor

- To demultiplex, the processor provides a signal called ALE (Address Latch Enable)
- The ALE is high and then low at the beginning of every bus cycle
- At the same time, the address is given out through AD_0 – AD_{15} and A_{16} – A_{19} /status lines
- Demultiplexing of address/data lines and address/status lines using 8-bit D-latch 74LS373 is shown in figure

- The ALE is connected to the Enable Pin (EN) of the external 8-bit latches
- When ALE is high and then Low, the addresses are latched into the output lines of the latch
- It holds the address until the next bus cycle
- After latching the address, the AD_0 – AD_{15} lines are free for data transfer and A_{16}-A_{19}/status lines are free for carrying status information.
- The first T-state of every bus cycle is used for address latching in 8086 and the remaining T states are used for reading or writing operation.
- The data bus is provided with a bidirectional buffer in order to drive the data to a longer distance in the bus
- The 8086 provides two control signals DT/R’ and DEN’ for controlling the data buffers
- The DT/R’ is used to decide the direction of data flow and DEN’ is used to enable the data buffer

Flag register of 8086 microprocessor

- The flag register is one of the special purpose register.
- The flag bits are changed to 0 or 1 depending upon the value of
- Result after arithmetic or logical operations.
- 8086 has 16-bit flag register, and there are 9 valid flag bits.
- The format of flag register is like below.

Bits	D_{15}	D_{14}	D_{13}	D_{12}	D_{11}	D_{10}	D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Flags					O	D	I	T	S	Z		AC		P		CY

Flag Bit	Function
S	After any operation if the MSB is 1, then it indicates that the number is negative. And this flag is set to 1
Z	If the total register is zero, then only the Z flag is set
AC	When some arithmetic operations generates carry after the lower half and sends it to upper half, the AC will be 1
P	This is even parity flag. When result has even number of 1, it will be set to 1, otherwise 0 for odd number of 1s
CY	This is carry bit. If some operations are generating carry after the operation this flag is set to 1
O	The overflow flag is set to 1 when the result of a signed operation is too large to fit.

Control Flags

- In 8086 there are 3 different flags which are used to enable or disable some basic operations of the microprocessor.
- These flags and their functions are listed below.

Flag Bit	Function
D	This is directional flag. This is used in string related operations. D = 1, then the string will be accessed from higher memory address to lower memory address, and if D = 0, it will do the reverse.
I	This is interrupt flag. If I = 1, then MPU will recognize the interrupts from peripherals. For I = 0, the interrupts will be ignored
T	This trap flag is used for on-chip debugging. When T = 1, it will work in a single step mode. After each instruction, one internal interrupt is generated. It helps to execute some program instruction by instruction.