CSE 3109
Microprocessors and Assembly Language

Microprocessors

Presented by

Mohiuddin Ahmed

Lecturer

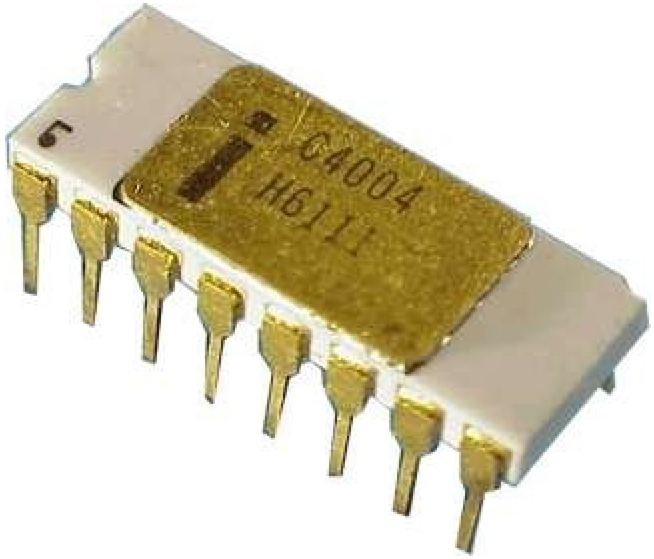
Department of CSE

RUET

EVOLUTION OF MICROPROCESSORS

4-BIT MICROPROCESSORS

INTEL 4004



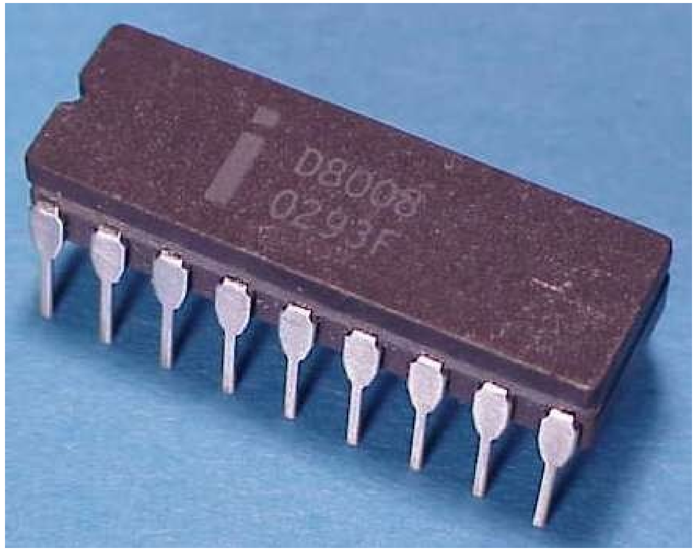
- Introduced in 1971.
- It was the first microprocessor by Intel.
- It was a 4-bit µP.
- Its clock speed was 740KHz.
- It had 2,300 transistors.
- It could execute around 60,000 instructions per second.

INTEL 4040

- Introduced in 1974.
- It was also 4-bit µP.

8-bit Microprocessors

INTEL 8008



- Introduced in 1972.
- It was first 8-bit μP.
- Its clock speed was 500 KHz.
- Could execute 50,000 instructions per second.

INTEL 8080

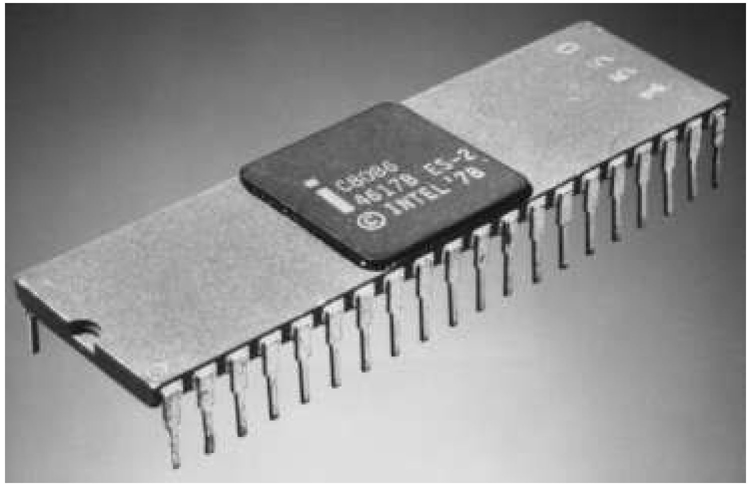
- Introduced in 1974.
- It was also 8-bit µP.
- Its clock speed was 2 MHz.
- It had 6,000 transistors.
- Was 10 times faster than 8008.
- Could execute 5,00,000 instructions per second.

INTEL 8085

- Introduced in 1976.
- It was also 8-bit µP.
- Its clock speed was 3 MHz.
- Its data bus is 8-bit and address bus is 16-bit.
- It had 6,500 transistors.
- Could execute 7,69,230 instructions per second.
- It could access 64 KB of memory.
- It had 246 instructions.

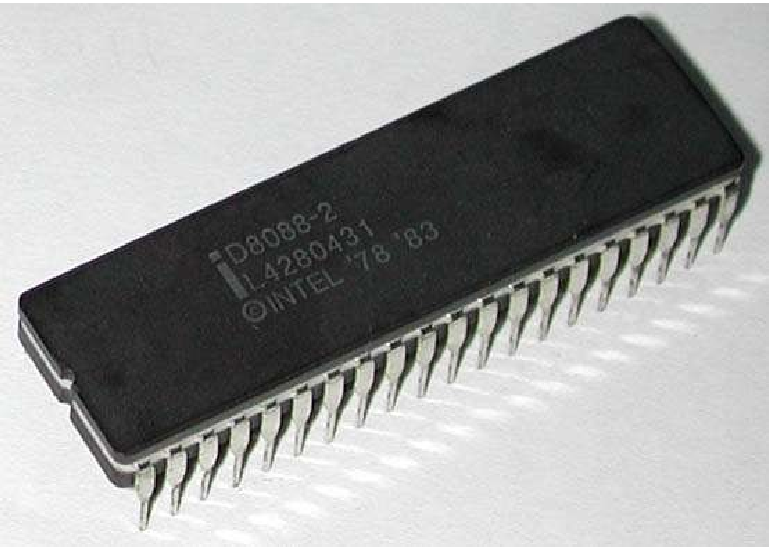
16-bit Microprocessors

INTEL 8086



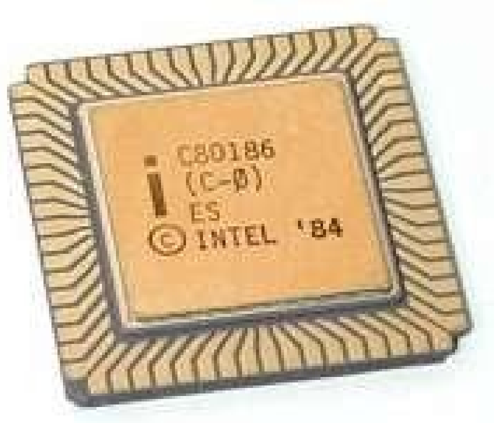
- Introduced in 1978.
- It was first 16-bit µP.
- Its clock speed is 4.77 MHz, 8 MHz and 10 MHz, depending on the version.
- Its data bus is 16-bit and address bus is 20-bit.
- It had 29,000 transistors.
- Could execute 2.5 million instructions per second.
- It could access 1 MB of memory.
- It had 22,000 instructions.
- It had ***Multiply*** and ***Divide*** instructions.

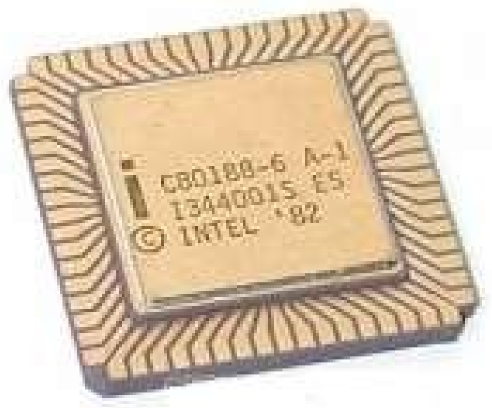
INTEL 8088



- Introduced in 1979.
- It was also 16-bit μP.
- It was created as a cheaper version of Intel's 8086.
- It was a 16-bit processor with an 8-bit external bus.

INTEL 80186 & 80188





- Introduced in 1982.
- They were 16-bit µPs.
- Clock speed was 6 MHz.

INTEL 80286

- Introduced in 1982.
- It was 16-bit μP.
- Its clock speed was 8 MHz.

32-BIT MICROPROCESSORS

INTEL 80386

- Introduced in 1986.
- It was first 32-bit µP.
- Its data bus is 32-bit and address bus is 32-bit.
- It could address 4 GB of memory.

INTEL 80486

- Introduced in 1989.
- It was also 32-bit µP.
- It had 1.2 million transistors.
- Its clock speed varied from 16 MHz to 100 MHz depending upon the various versions.

INTEL PENTIUM

- Introduced in 1993.
- It was also 32-bit μP.
- It was originally named 80586.
- Its clock speed was 66 MHz.

INTEL PENTIUM PRO

- Introduced in 1995.
- It was also 32-bit µP.

INTEL PENTIUM II

- Introduced in 1997.
- It was also 32-bit µP.

INTEL PENTIUM II XEON

- Introduced in 1998.
- It was also 32-bit µP.

INTEL PENTIUM III

- Introduced in 1999.
- It was also 32-bit µP.

INTEL PENTIUM IV

- Introduced in 2000.
- It was also 32-bit μP.

INTEL DUAL CORE

- Introduced in 2006.
- It is 32-bit or 64-bit µP.
- It has two cores.
- Both the cores have there own internal bus and L1 cache, but share the external bus and L2 cache

64-BIT MICROPROCESSORS

INTEL CORE 2 DUO

- Introduced in 2006.
- It is a 64-bit µP.

INTEL CORE i3

- Introduced in 2008.
- It is a 64-bit µP.

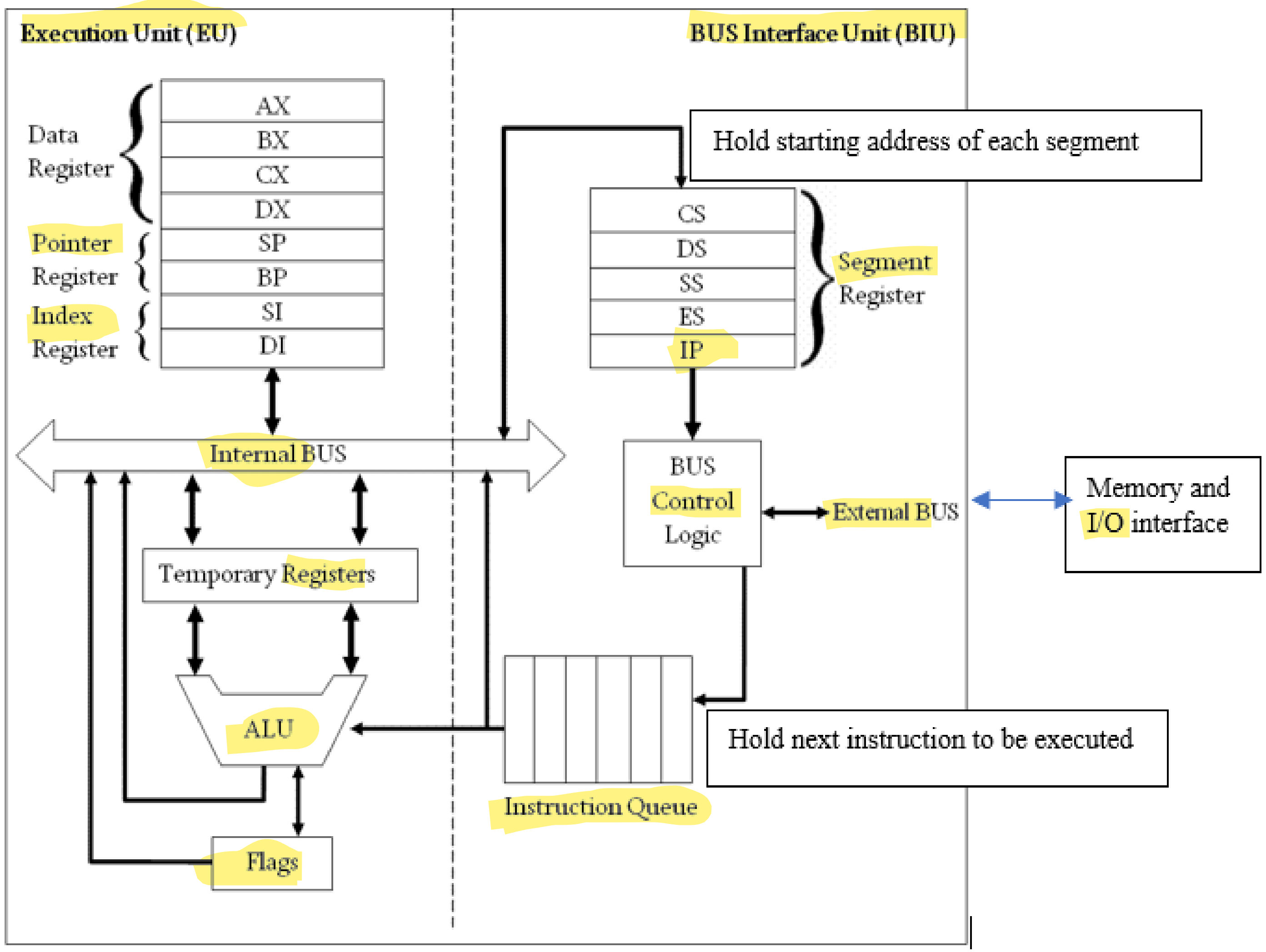
INTEL CORE i5

- Introduced in 2009.
- It is a 64-bit µP.

INTEL CORE i7

- Introduced in 2010.
- It is a 64-bit µP.

Internal Architecture of 8086 Microprocessor



Internal Architecture of 8086 Microprocessor

- It has mainly two functional parts. Dividing the works between two units speeds up the processing.
- **Bus Interface Unit:** Main function of BIU is divided into three sections.

1) Calculating the 20-bit physical address.
2) Provide the interface with memory and I/O devices.
3) Provide pipelining by pre-fetching instructions and stored in the instruction queue.

- The BIU sends out addresses, fetches instructions from memory, reads data from ports and memory, and writes data to ports and memory.
- BIU handles all transfers of data and addresses on the buses for the execution unit.

Internal Architecture of 8086 Microprocessor

❖ **Execution Unit:** The execution unit of 8086 tells the BIU where to fetch instructions or data, decodes instructions, and executes instructions

1) Process the instructions or data provided by the BIU.
2) Stores the result in the internal registers.
3) Update the offset address of the registers and status flags.

Types of Registers in the 8086 Microprocessor

- Data Registers – **AX, BX, CX, DX**
- Segment Registers – **CS, DS, SS, ES**
- Pointers and Index Registers – **IP, BP, SP, SI, DI**
- Flags or Status Register

Data Registers

- AX (Accumulator)
  It is used in arithmetic, logic and data transfer instructions in 8086 microprocessors. In multiplication and division, one of the numbers involved must be in AX or AL.

- BX (Base Register)
  BX register is an address register. It usually contains a data pointer used for based, based indexed or register indirect addressing.

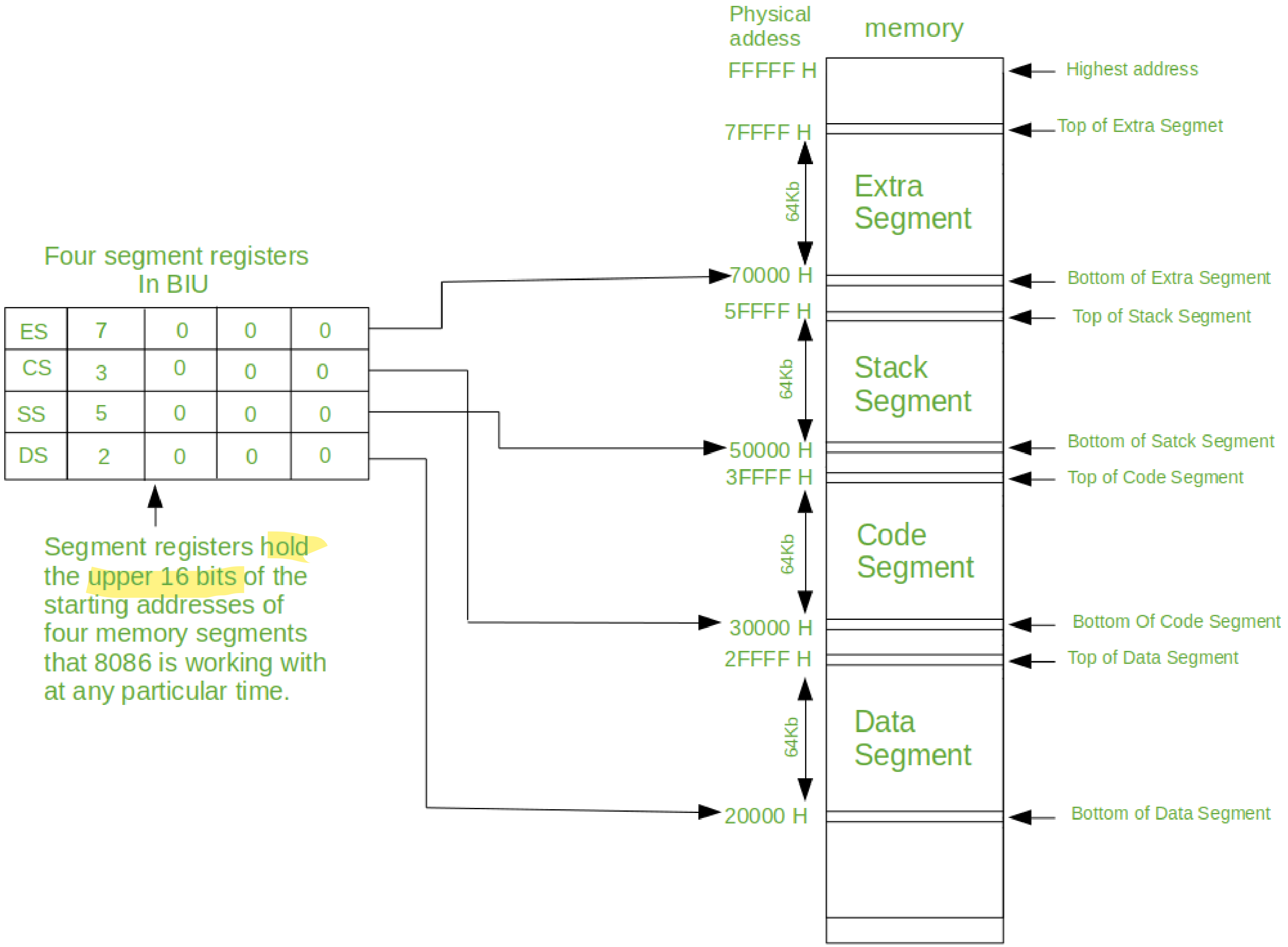
- CX (Count register)
  This serves as a loop counter. Program loop constructions are facilitated by it. Count register can also be used as a counter in string manipulation and shift/rotate instruction.

- DX (Data Register)
  DX is data register. The two parts are DH and DL. This register can be used in Multiplication, Input/output addressing etc.

15 7 0

AX	AH	AL
BX	BH	BL
CX	CH	CL
DX	DH	DL

Segment Registers



Four segment registers In BIU

ES	7	0	0	0
CS	3	0	0	0
SS	5	0	0	0
DS	2	0	0	0

Segment registers hold the upper 16 bits of the starting addresses of four memory segments that 8086 is working with at any particular time.

- **Code Segment (CS) Register:**
  CS holds the base address for the Code Segment. All programs are stored in the Code Segment and accessed via the IP.
- **Data Segment (DS) Register:**
  DS holds the base address for the Data Segment.
- **Stack Segment (SS) Registers:**
  The SS is used to store the information about the memory segment. The operations of the SS are mainly Push and Pop.
- **Extra Segment (ES) Register:**
  By default, the control of the compiler remains in the DS where the user can add and modify the instructions. If there is less space in that segment, then ES is used. ES is also used for copying purpose.

Pointers and Index Registers

- IP (Instruction Pointer)

  It is a 16-bit register. It holds offset of the next instructions in the Code Segment.

- SP (Stack Pointer)

  This is stack pointer register pointing to program stack. It is used in conjunction with SS for accessing the stack segment. It is of 16 bits. It points to the topmost item of the stack.

- BP (Base Pointer)

  This is base pointer register pointing to data in stack segment. Unlike SP, we can use BP to access data in the other segments. It is of 16 bits. It is primarily used in accessing parameters passed by the stack. Its offset address is relative to the stack segment.

- SI (Source Index)

  This is source index register which is used to point to memory locations in the data segment addressed by DS. Thus, when we increment the contents of SI, we can easily access consecutive memory locations. It is of 16 bits. Its offset is relative to the data segment.

- DI (Destination Index)

  This is destination index register performs the same function as SI. There is a class of instructions called string operations, that use DI to access the memory locations addressed by ES

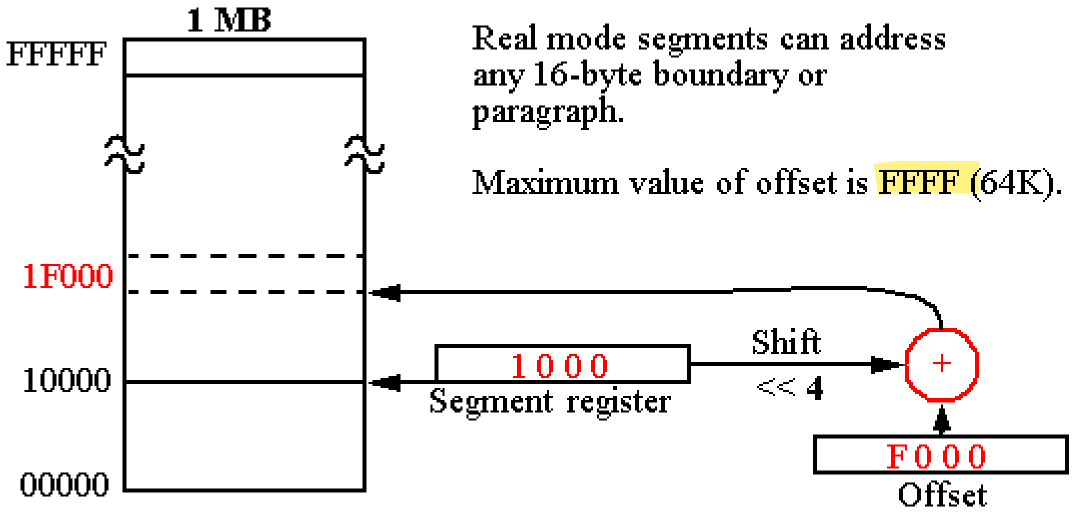
Segment and Offset Address

Segment Address:

- The content of a segment register is called as segment address.

Offset Address:

- The offset address is a location within a 64K-byte segment range.
- It is also called the displacement from the base address



Real mode segments can address any 16-byte boundary or paragraph.

Maximum value of offset is FFFF (64K).

Physical and Logical Address

Physical Address:

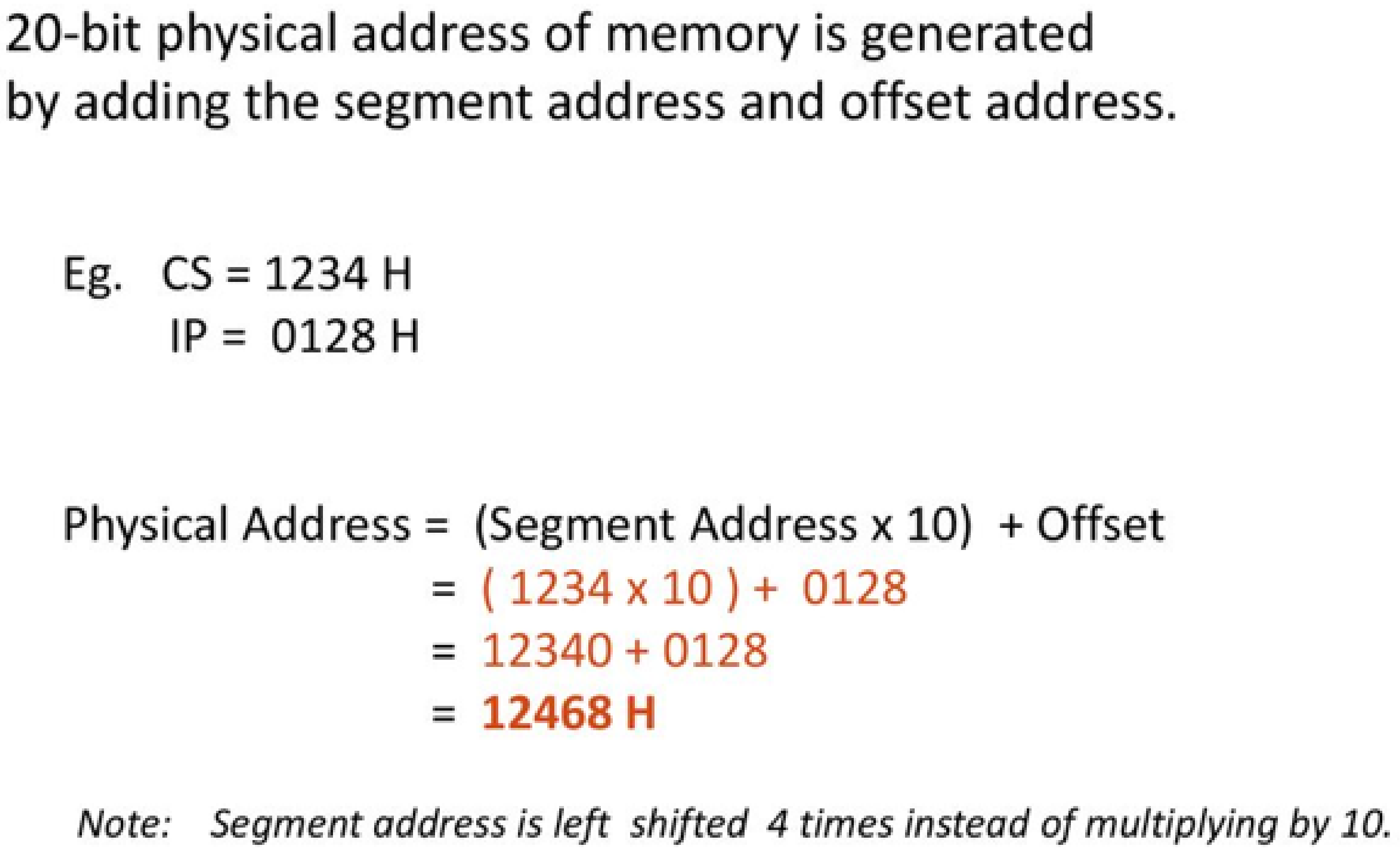
- The physical address is the 20-bit address that is actually put on the address pins of the 8086 microprocessor and decoded by the memory interfacing circuitry.

Logical Address:

- The logical address consists of a segment value and an offset address.
- Represented by Segment:Offset

Physical Address Calculation

Physical Address = Segment ×10h+Offset



20-bit physical address of memory is generated by adding the segment address and offset address.

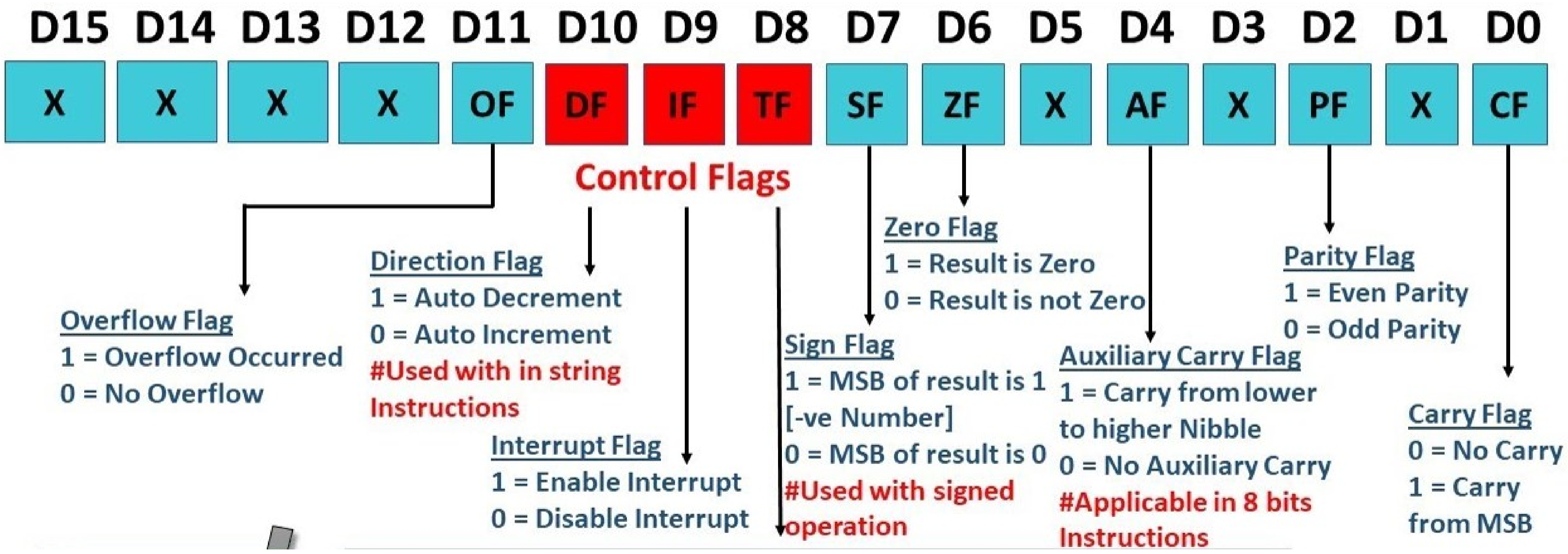
Eg. CS = 1234 H
IP = 0128 H

Physical Address = (Segment Address x 10) + Offset
= (1234 x 10) + 0128
= 12340 + 0128
= **12468 H**

Note: Segment address is left shifted 4 times instead of multiplying by 10.

Flags Register

The Flag register is a Special Purpose Register. Depending upon the value of result after any arithmetic and logical operation the flag bits become set (1) or reset (0).



D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
X	X	X	X	OF	DF	IF	TF	SF	ZF	X	AF	X	PF	X	CF

Control Flags

Overflow Flag
1 = Overflow Occurred
0 = No Overflow

Direction Flag
1 = Auto Decrement
0 = Auto Increment
#Used with in string Instructions

Interrupt Flag
1 = Enable Interrupt
0 = Disable Interrupt

Sign Flag
1 = MSB of result is 1 [-ve Number]
0 = MSB of result is 0
#Used with signed operation

Zero Flag
1 = Result is Zero
0 = Result is not Zero

Auxiliary Carry Flag
1 = Carry from lower to higher Nibble
0 = No Auxiliary Carry
#Applicable in 8 bits Instructions

Parity Flag
1 = Even Parity
0 = Odd Parity

Carry Flag
0 = No Carry
1 = Carry from MSB

Thank You