Tishk International University
Science Faculty
IT Department

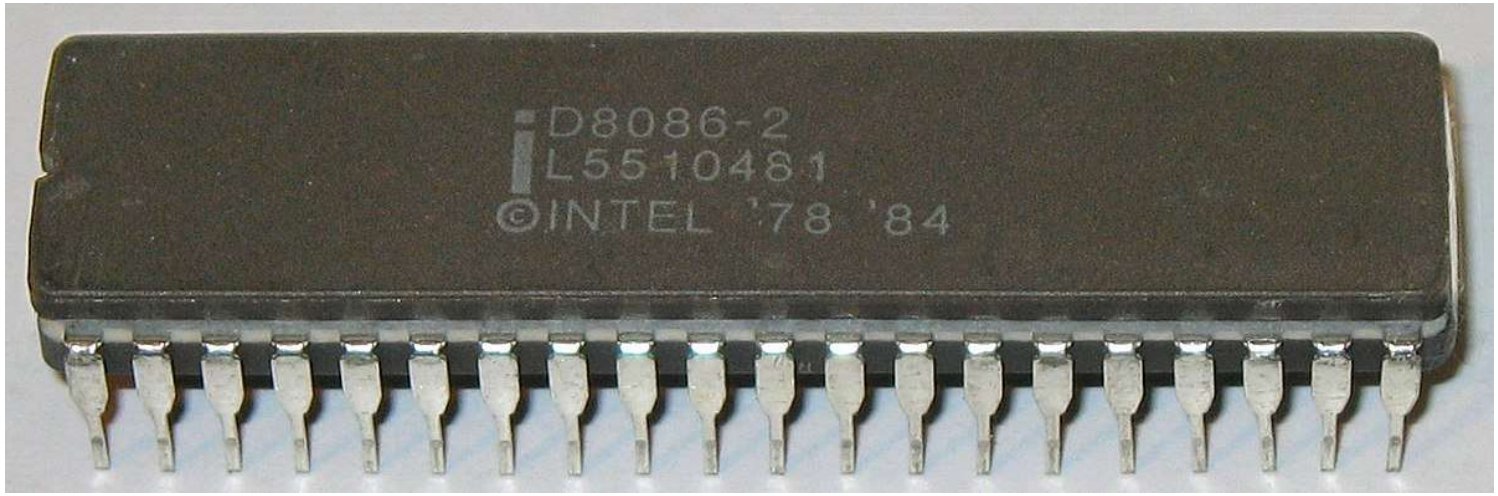
Computer Hardware

Lecture 02: CPU

2nd Grade – Spring Semester

Instructor: Alaa Ghazi

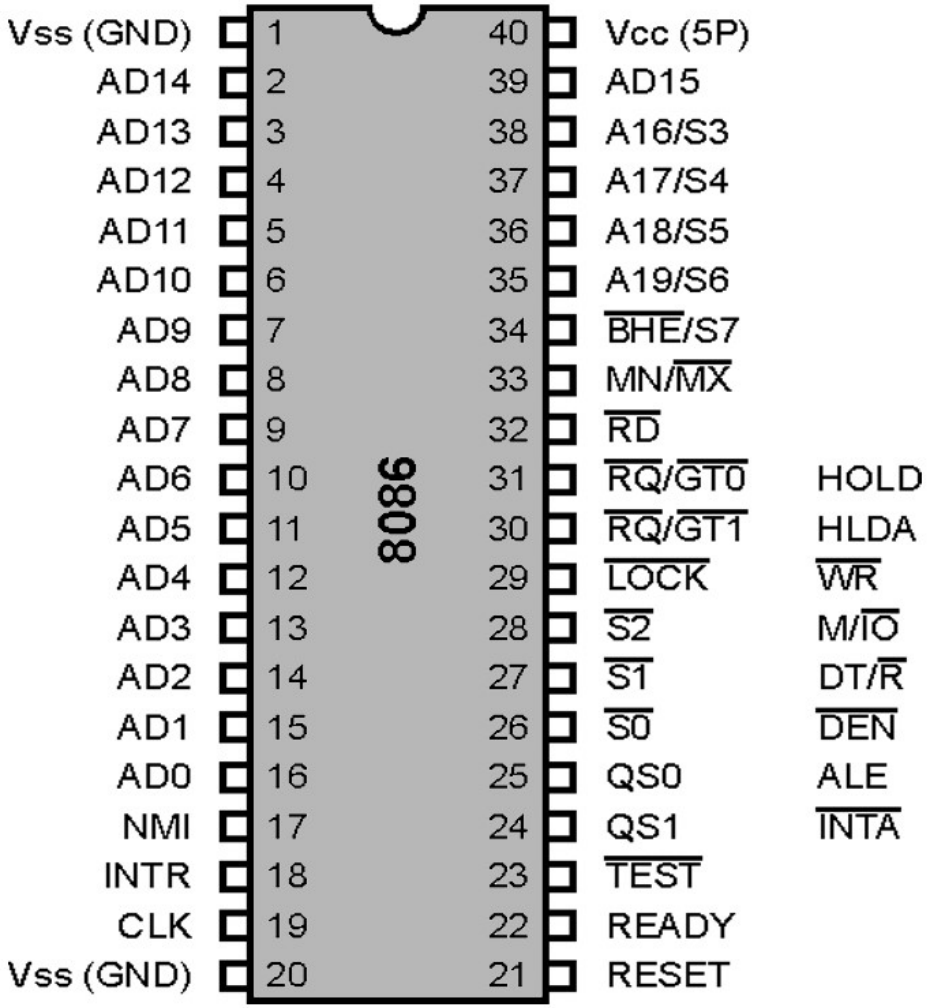
Lecture 02: CPU
PART1- Intel 8086



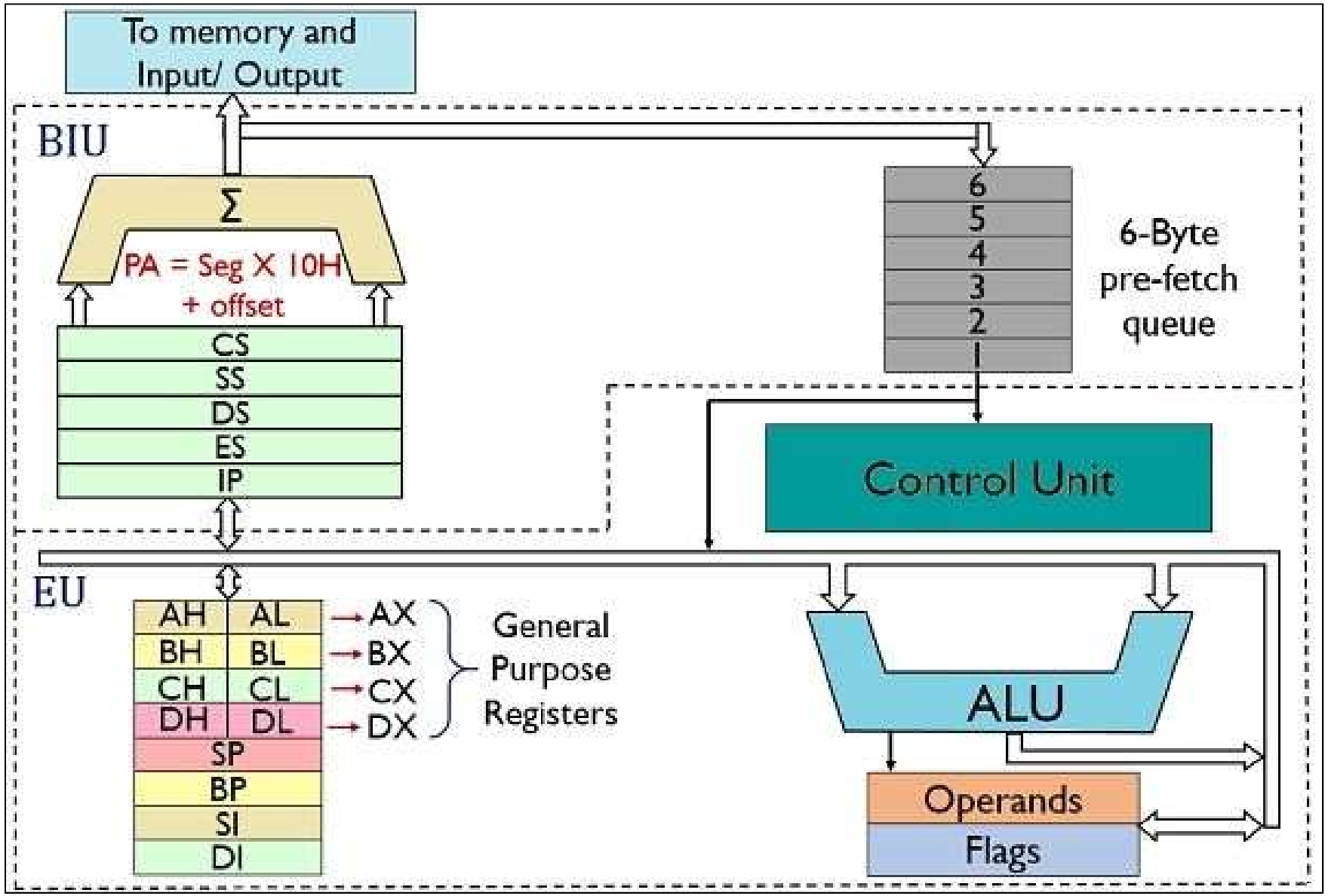
8086 Features

- IBM PC was based on Intel 8088 CPU.
- Intel has manufactured two 16-bit CPUs
  - 8086 has 16-bit registers with 16-bit external data bus.
  - 8088 has 16-bit registers with 8-bit external data bus.
- Each register is 16-bits so the word size is 16 bits.
- 8086/8088 has a 20 bit address bus can access up to 2^{20} memory locations (1 MB).
- Address ranges from 00000H to FFFFFH and Every byte has a separate address.

Intel 8086 Pinout (not required in the exam)



Block Diagram of Intel 8086 Architecture



Internal architecture of 8086

- **8086 has two blocks**
  - Bus Interface Unit (BIU)
  - Execution Unit (EU).

- Both units operate asynchronously to give the 8086 an overlapping instruction fetch and execution mechanism which is called as Pipelining. This results in efficient use of the system bus and system performance.

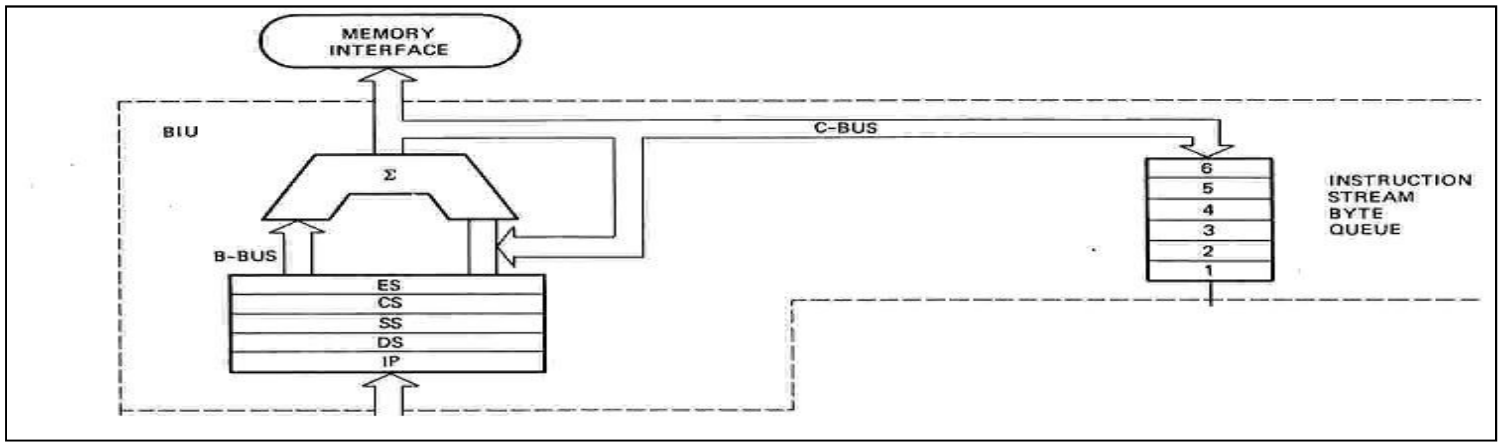
BUS INTERFACE UNIT (BIU)

BIU Functions are:

- instruction fetching, and storing it in instruction queue.
- reading and writing data into memory
- calculating the addresses of the memory locations

BIU Parts are:

- Instruction Queue
- The Segment Registers (CS, DS, ES, SS).
- The Instruction Pointer register (IP).
- The Address Summing block.



Segment registers

- 8086 has 4 segments registers, and all are 16 bit registers:

1. Code Segment register (CS),
2. Data Segment register (DS),
3. Extra Segment register (ES) and
4. Stack Segment register (SS).

- Each of the Segment registers store the upper 16 bit address of the starting address of the corresponding segments.
- Segment and Address register combinations are:
  - **CS:IP**
  - **SS:SP SS:BP**
  - **DS:BX DS:SI**
  - **ES:DI**

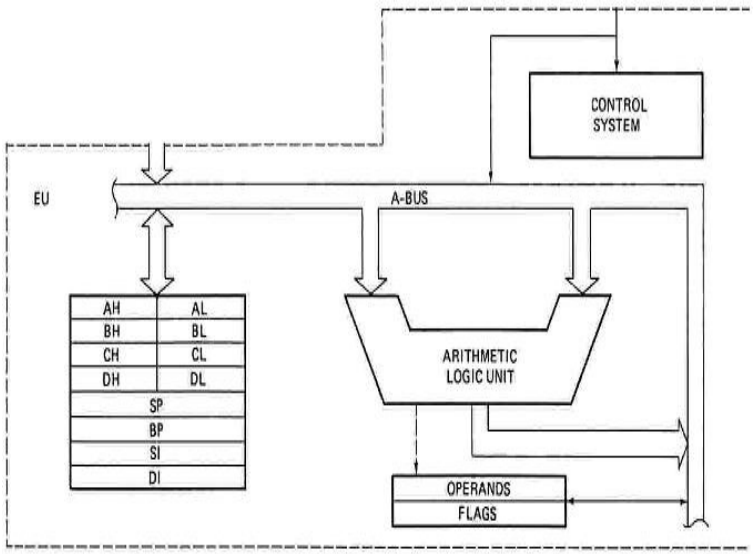
EXECUTION UNIT (EU)

EU Functions are:

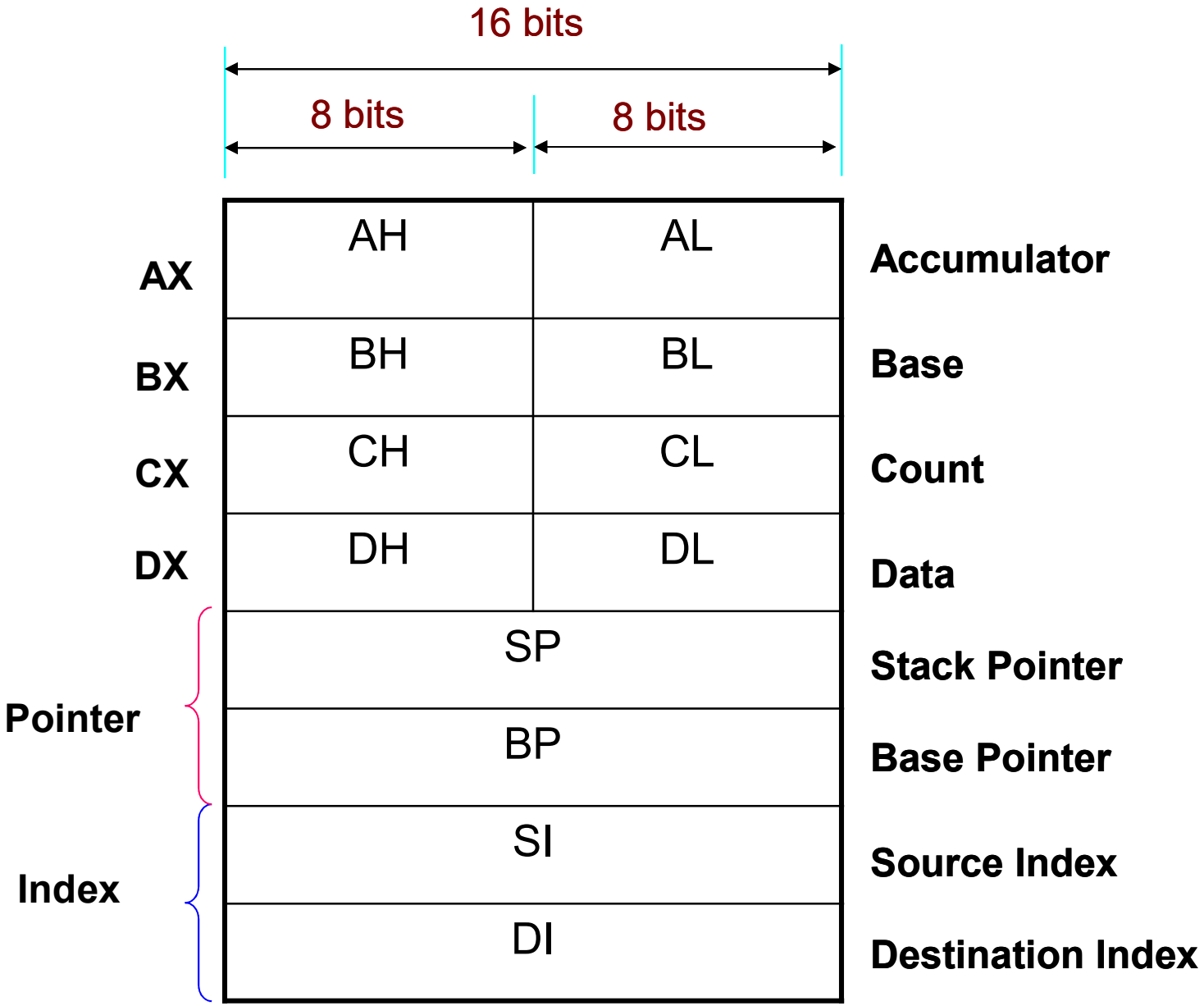
- Decodes instructions fetched by the BIU
- Generate control signals,
- Executes instructions.

EU Parts are:

- Control System
- Arithmetic and Logic Unit (ALU)
- General Purpose Registers
- Pointer And Index Registers



EXECUTION UNIT – Registers



EXECUTION UNIT – Registers

- General Purpose Registers:

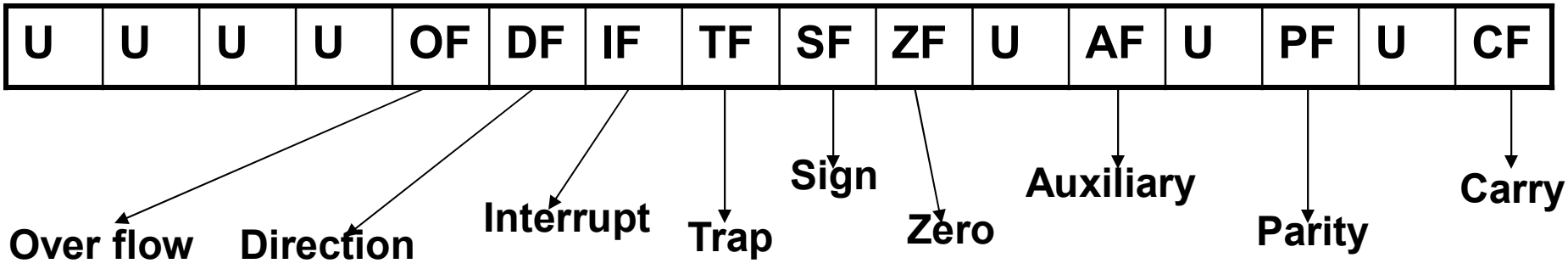
Register	Segment Register	Purpose
AX		Word multiply, word divide, word I /O
BX	DS	Store data address
CX		String operation, loops counter
DX		Word multiply, word divide, indirect I/O

- Pointer and Index Registers:

Register	Segment Register	Purpose
Source Index (SI)	DS	To access data stored in arrays
Destination Index (DI)	ES	To access data stored in arrays
Stack Pointer (SP)	SS	To access stack
Base Pointer (BP)	SS	To access stack

Execution Unit -Flag Register

- A flag is a flip flop which indicates some conditions produced by the execution of an instruction or controls certain operations of the EU, which helps to change program flow.
- For example using ZF to check if loop counter reachs end.
- EU contains 16 bit flag register
  - ❑ 6 status flags
  - ❑ 3 control Flags
  - ❑ 7 undefined flags.



U - Unused

Flag Register- Status Flags

Flag	Purpose
Carry (CF)	Holds the carry after addition or the borrow after subtraction
Parity (PF)	PF=0;odd parity, PF=1;even parity.
Auxiliary (AF)	Holds the carry (half – carry) after addition or borrow after subtraction between bit positions 3 and 4 of the result.
Zero (ZF)	Shows the result of the arithmetic or logic operation. Z=1; result is zero. Z=0; The result is non zero
Sign (SF)	Holds the sign of the result after an arithmetic/logic instruction execution. S=1; negative, S=0
Overflow (OF)	Set to indicate an overflow, when the result has exceeded the capacity of the Machine

Flag Register- Control Flags

Flag	Purpose
Trap (TF)	Enables the trapping through an on-chip debugging feature.
Interrupt (IF)	Controls the operation of the INTR (interrupt request) I=0; INTR pin disabled. I=1; INTR pin enabled.
Direction (DF)	It selects either the increment or decrement mode for DI and /or SI registers during the string instructions.

Segmented Memory

- The memory in an 8086 based system is organized as segmented memory.

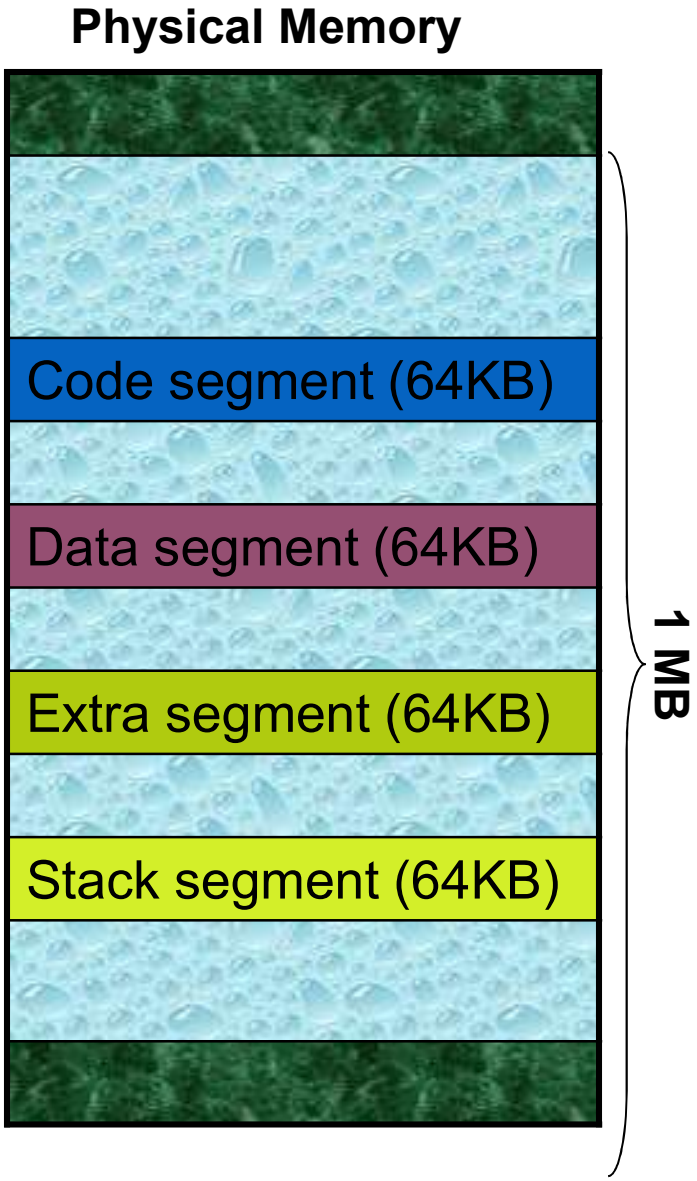
- The CPU 8086 is able to address 1Mbyte of memory.

- The Complete physically available memory may be divided into a number of logical segments.

The size of each segment is 64 KB

Advantages of Segmented memory Scheme

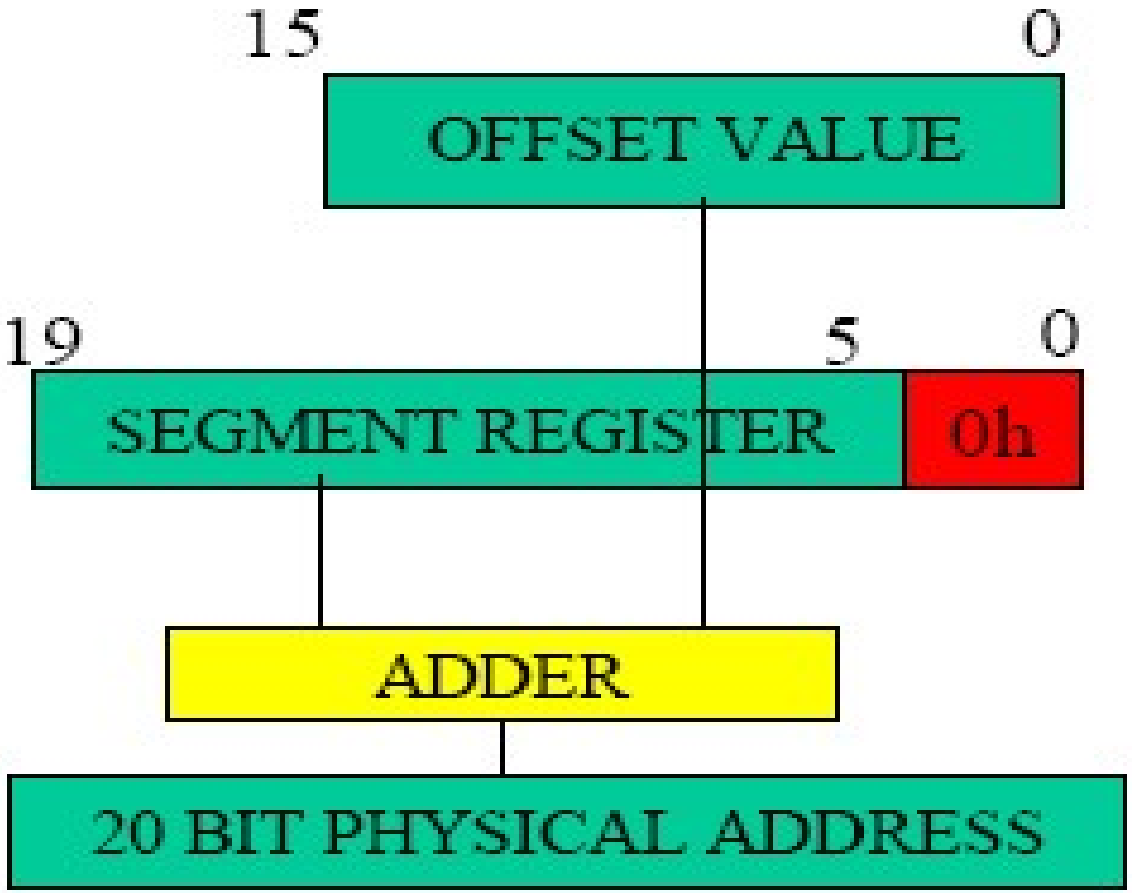
- Allows the placing of code, data and stack portions of the same program in different parts (segments) of the memory, for data and code protection.
- Permits a program and data to be put into different areas of memory each time program is executed.



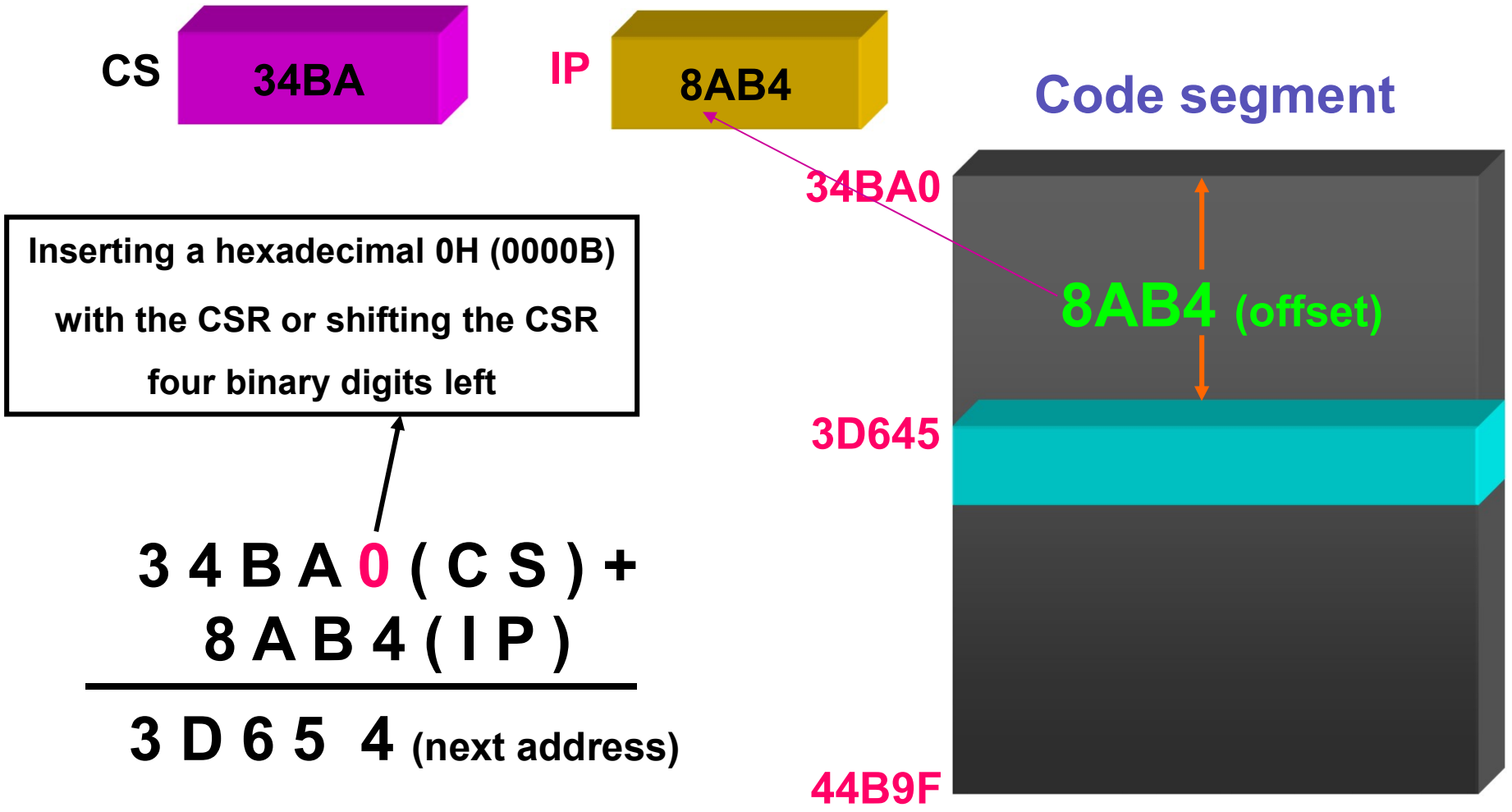
Instruction pointer & summing block

- The instruction pointer (IP) register contains a 16-bit offset address of instruction that is to be executed next.
- The IP always references the Code segment register (CS).
- The value contained in the instruction pointer is called as an offset because this value must be added to the base address of the code segment, which is available in the CS register to find the 20-bit physical address.
- **The value of the instruction pointer is incremented after executing every instruction.**
- To form a 20bit address of the next instruction, the 16 bit address of the IP is added (by the address summing block) to the address contained in the CS , which has been shifted four bits to the left.

Address Formation in 8086



Address Formation Example: (not required in the exam)



CPUs History

CPU	Year	Bus width	Freq.	Description
4004	1971	4	750 kHz	First Intel microprocessor.
8085	1976	8	3 MHz	Enhanced version of Intel 8-bit CPU.
8086/8088	1978	16	5-10 MHz	First generation of Intel 80x86 processors. Used in IBM PC
80386	1985	32	16-33 MHz	Third generation of 80x86 processors: 32 bit architecture, new processor modes.
Pentium	1993	32	60 - 66 MHz	Fifth generation of x86 processors: superscalar architecture, MMX. Intel shifts from numbers to names for its chips after the company learns it's impossible to trademark a number
Celeron	1998	32	366 - 400 MHz	Low-cost version of Pentium series.

CPUs History

CPU	Year	Bus width	Freq.	Description
Itanium	2001	64	733 - 800 MHz	High-performance 64-bit microprocessor.
Xeon	1998	32, 64	0.4 - 4.80 GHz	High-performance version of Pentium 4 CPU.
Core 2	2006	64	1.86 GHz	64-bit microprocessor.
Atom	2008	32, 64	600 - 800 MHz	Ultra-low power microprocessor.

CPUs History

CPU	Year	Bus width	Freq.	Description
Core i7	2008	64	3.3 GHz	• The higher the number, the more powerful the CPU • More powerful CPUs have more cores and are clocked at a faster frequency. • They all use the same sockets and motherboards • more powerful CPUs tend to have better support for higher RAM sizes and frequencies as well as higher overclocking speeds
Core i5	2009	64	3.4 GHz	
Core i3	2010	64	2.5 GHz	
Core i9	2017	64		

Modern CPU Naming

U: Ultra-Low Power. The U rating is only for laptop processors. These draw less power and are better for the battery.

Example:- i7 7500u, i5 8250u, etc

Y: Low Power. Typically found on older generation laptops and mobile processors.

Example:- m3 8100Y, i5 8200Y, i7 8500Y etc.

T: Power Optimized for desktop processors.

Example:- i5 7500T, i5 7400T, i3 7300T, i3 7100T etc.

Q: Quad-Core. The Q rating is only for processors with four physical cores.

Example:- i7 7700HQ, i5 7300HQ etc.

H: High-Performance Graphics. The chipset has one of Intel's better graphics units

Example:- i7 7700HQ, i5 7300HQ, i7 8750H etc.

G: Includes Discrete Graphics. Typically found on laptops, this means there is a dedicated GPU with the processor.

Example:- i7 8705G, etc.

K: Unlocked. This means you can overclock the processor above its rating.

Example:- i7 7700k, i7 8700k, i9 9900k, etc

Lecture 02: CPU
PART2- Intel 80386

NEW 80386 Features

- First 32-bit microprocessor in the x86 family released in 1986.
- 32-bit ALU, 32-bit Registers, 32-bit Data Bus, 32-bit Address Bus
- Uses 32-bit address for memory and 16-bit address for I/O ports
- Maximum physical memory 4 Gb.
- Built-in Memory Management Unit to support Segmentation, Paging and Virtual Memory.
- Supports Virtual Memory upto 64 Tb.
- Maximum size of Segment 4 Gb.
- New "Debug" Registers provide Breakpoint facility to programmers.
- Three different speed models: 16/25/33 MHz

80386 Modes

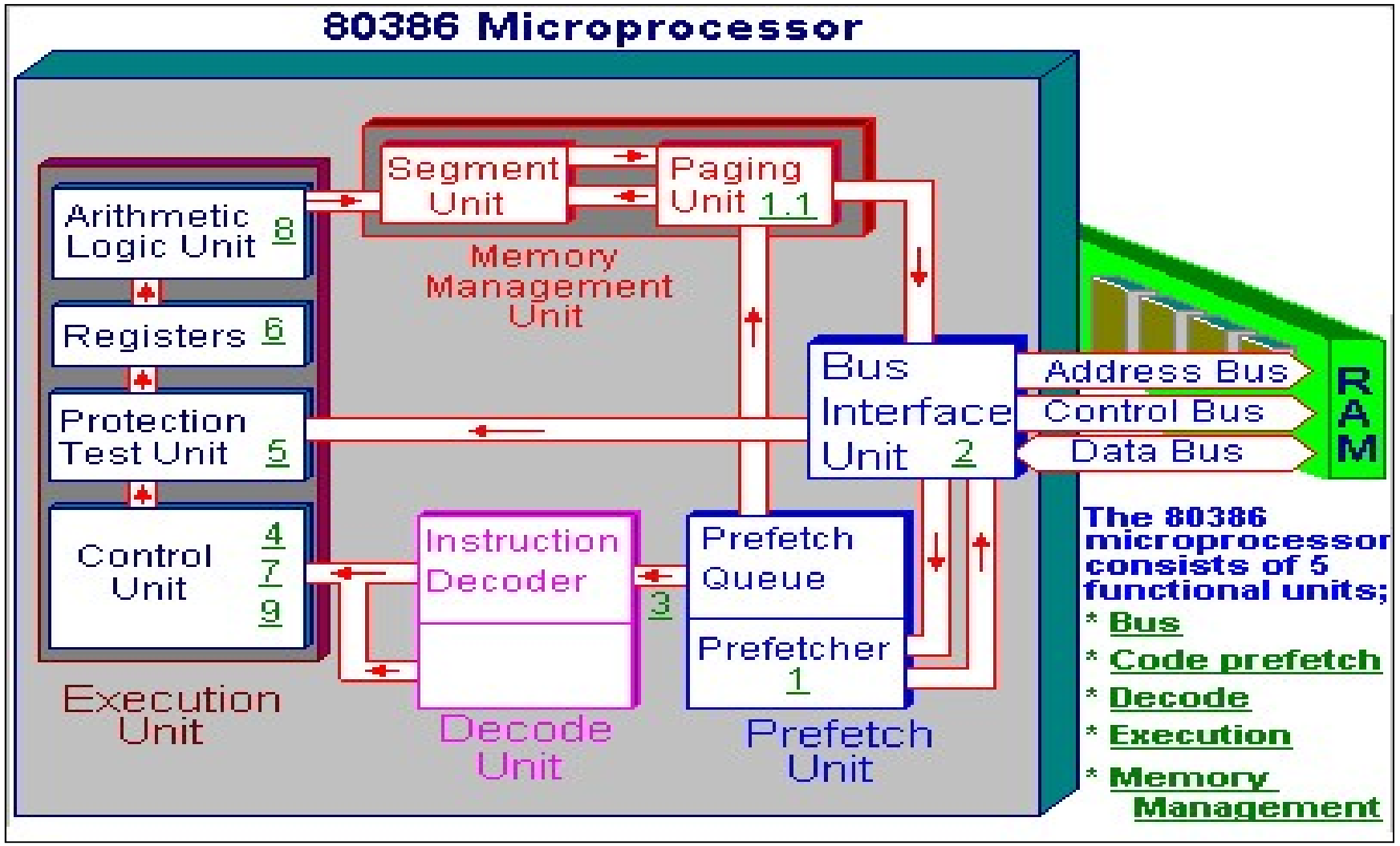
- Three Modes of Operation:
  - **Real Mode**: This is the Start-up mode. In this mode it behaves like a 8086, can access only 1 Mb memory, does not support multitasking/privilege levels/virtual memory/paging.
  - **Protected Mode**: In this mode, it can access all 4 Gb memory, supports multitasking/privilege levels/virtual memory/paging.

  To enable this mode, PE flag is set to 1, else system is in real mode
  - **Virtual 86 Mode**: This is a submode under Protected mode. In this mode, it provides virtual 1 Mb DOS Environment for DOS programs to work under multitask OS.

80386 Architecture

- It has Five functional units:
  1. Bus Interface unit
  2. Prefetch unit
  3. Decode unit
  4. Memory Management Unit:
     - Segmentation unit
     - Paging unit
  5. Execution unit

80386 Architecture Diagram
(not required in the exam)

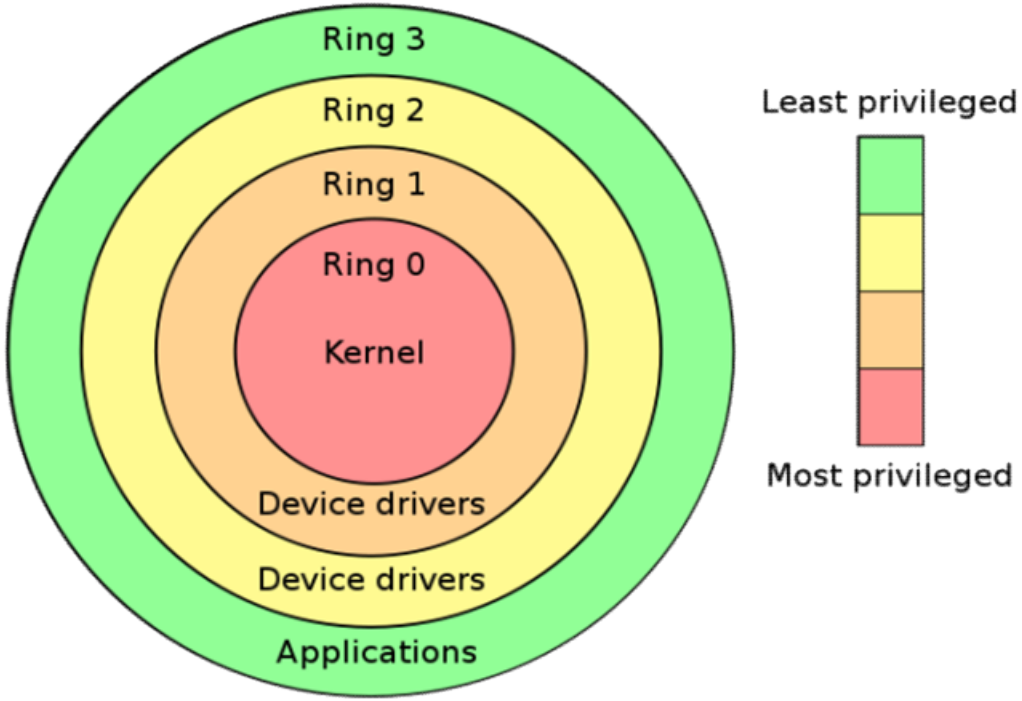


80386 Instruction Set Extensions

- 80386 extends the 8086/80186/80286 Instruction Set in two ways
  - Enhancing the scope of existing 16-bit instructions to 32-bit operands
  - New Instructions, not found in earlier members of x86 family

80386 Privilege levels

- Privilege levels determine what rights the code being executed has when issuing instructions, or accessing I/O ports and memory addresses. There are 4 privilege levels for 80386 processor architecture, and these are called rings. The four rings (0 to 3) are designed to protect the hardware resources, keeping user applications at level 3 (the least privilege) and the operating system kernel at ring 0 (the most privileged).



Kernel Mode vs User Mode

- 1.Kernel Mode: the executing code has complete and unrestricted access to the underlying hardware. It can execute any CPU instruction and reference any memory address. Kernel mode is generally reserved for the lowest-level, most trusted functions of the operating system. Crashes in kernel mode are requires to restart the entire PC.
- 2. User Mode: the executing code has no ability to directly access hardware or reference memory. Code running in user mode must delegate to Operating System to access hardware or memory. Due to the protection afforded by this sort of isolation, crashes in user mode are always recoverable. Most of the code running on your computer will execute in user mode.