Tishk International University
Science Faculty
IT Department

Computer Hardware

Lecture 01: Introduction

2nd Grade – Fall Semester

Instructor: Alaa Ghazi

Course Name: COMPUTER HARDWARE
Code/Section: IT 232/A

Instructor: Mr. Alaa Ghazi

Qualification: M.Sc. in Computer Engineering

Email: **alaa.ghazi@tiu.edu.iq**

Room No.: 313

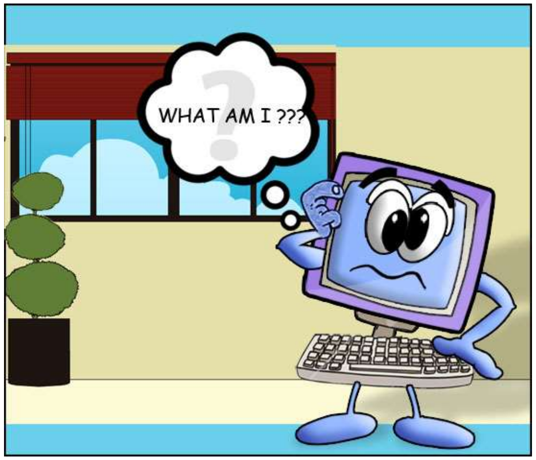
Reference: Mueller, Scott. Upgrading and Repairing PCs: Upgrading and Repairing_e22. Que Publishing, 2015.

COURSE CONTENT

#	**Topic**
1	Introduction
2	CPU
3	Standard Input Output Systems
4	Network Cards
5	Motherboard
6	Memory Organization
7	Video and Audio Systems
8	Storage Devices

Lecture 1 - Introduction

	Topic
Part 1	**Personal Computer Evolution**
Part 2	**Digital Signals and Circuits**
Part 3	**Von-Neumann Architecture**



Lecture 01: Introduction
Part 1 - Personal Computer Evolution

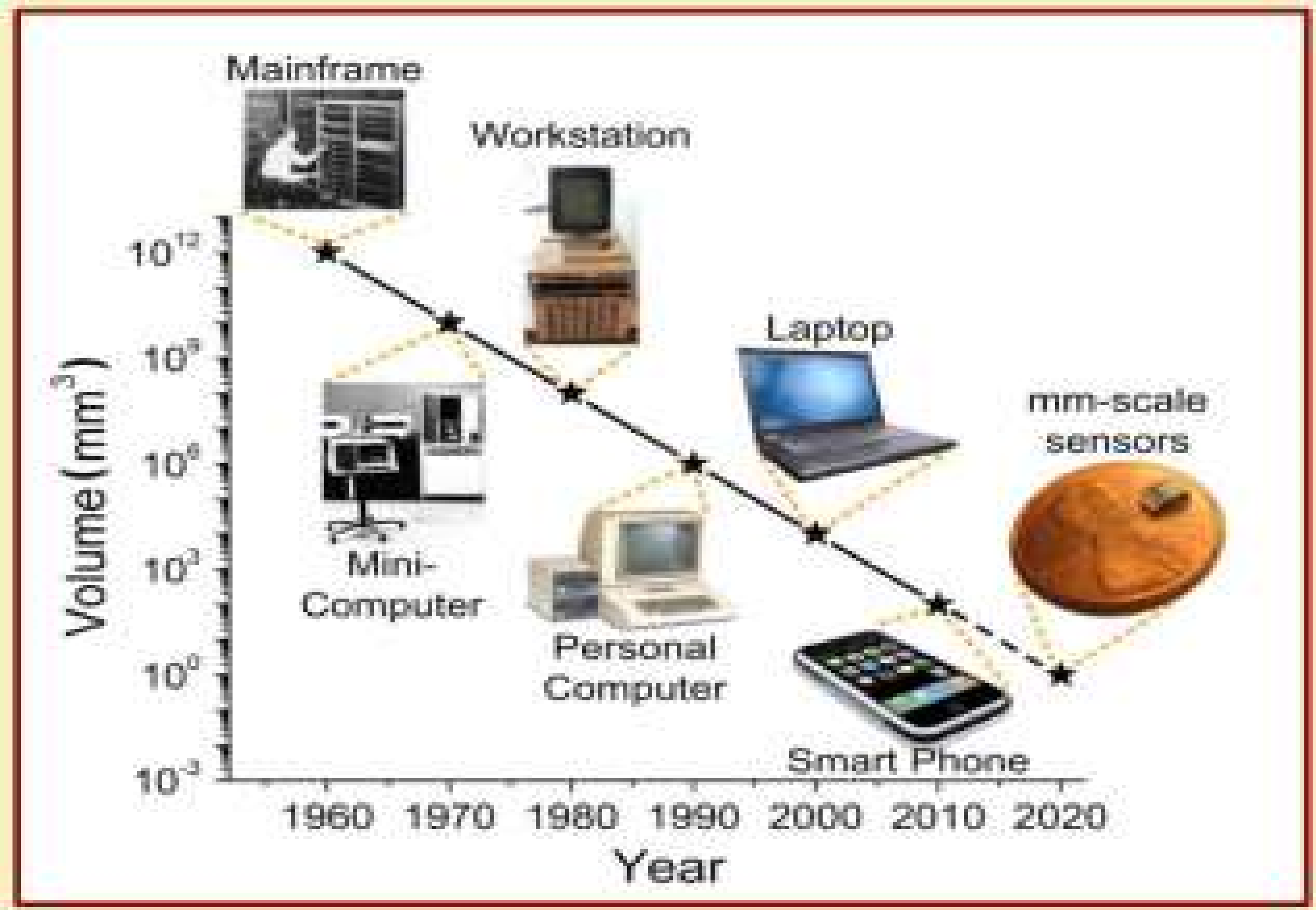
Computer Architecture Definition

- **Computer Architecture**: is how to integrate computer components to build a computer system to achieve a desired level of performance.

– Analogy: architect’s task during the planning of a building (overall layout, floorplan, etc).

- This lecture will focus on the IBM PC evolution, since it remains the most popular architecture and, most modern PCs could still run MS-DOS based programs from the 80’s and 90’s
- Understanding how the design of the PC has evolved will help when solving problems as it is necessary to have a deeper understanding of why things are done and what problems can happen to it.

Evolution of the Types of Computers (not required in the exam)



Mainframe: IBM System/360

- Very popular mainframe computer of the 60s and 70s.
- Introduced many advanced architectural concepts that appeared in microprocessors several decades later.

IBM PC History

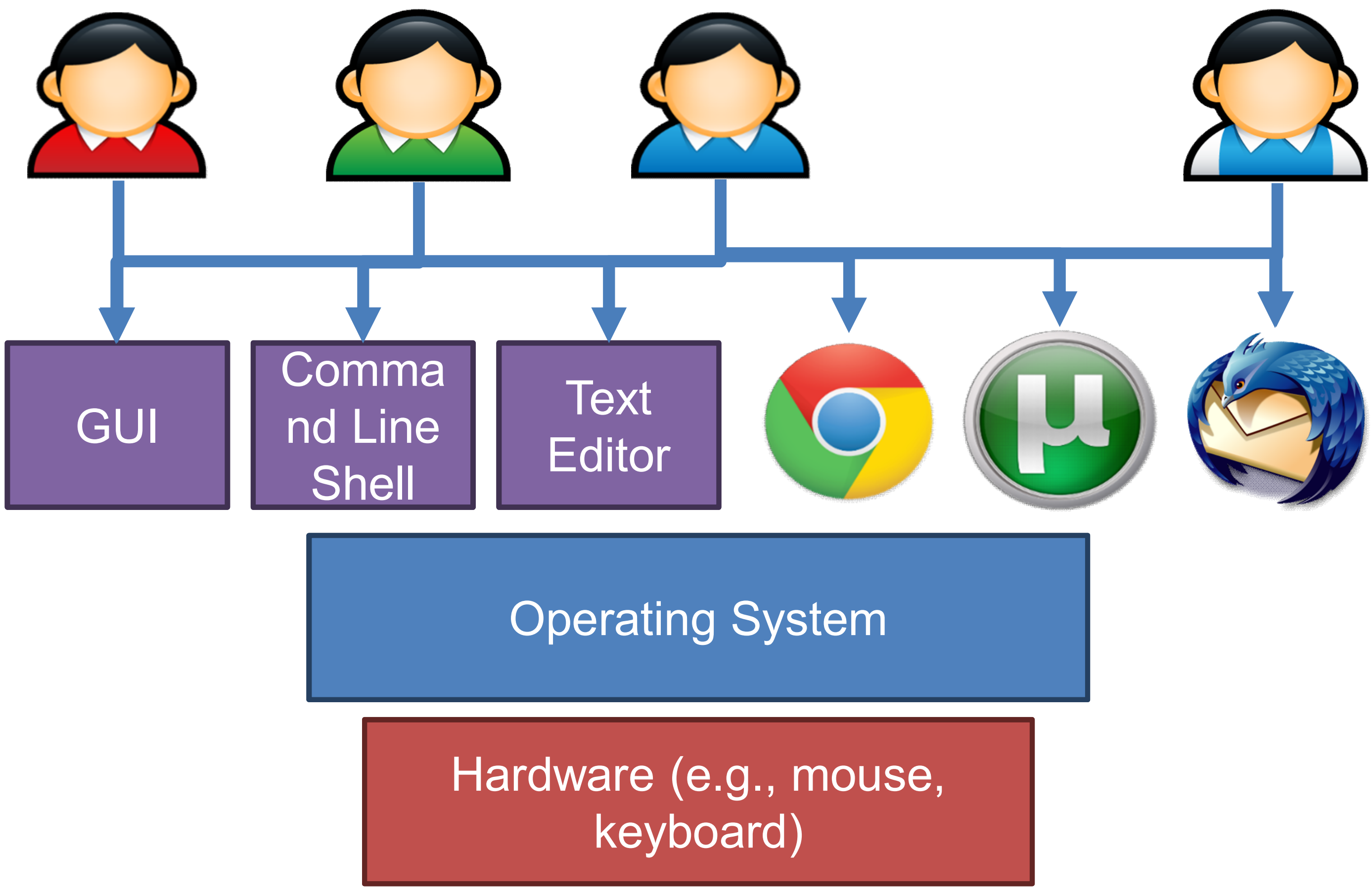
- 1981: IBM releases a Personal Computer (PC).
- It has the below features:
  - CPU (Central Processing Unit) is Intel 8088
  - 16 KB (Kilo Bytes) of memory is built-in in the basic model
  - Five hardware expansion slots
  - Basic Input/Output System (BIOS) firmware in ROM (Read Only Memory)
  - Its operating system is PC DOS 1.0 which was made by Microsoft.
  - It was based on **open architecture** and third-party peripherals:
    - 3rd party designers were asked to design extension cards.
    - 3rd party developers were asked to write more application software.

Due to above IBM PC hardware became the de-facto standard for that time.

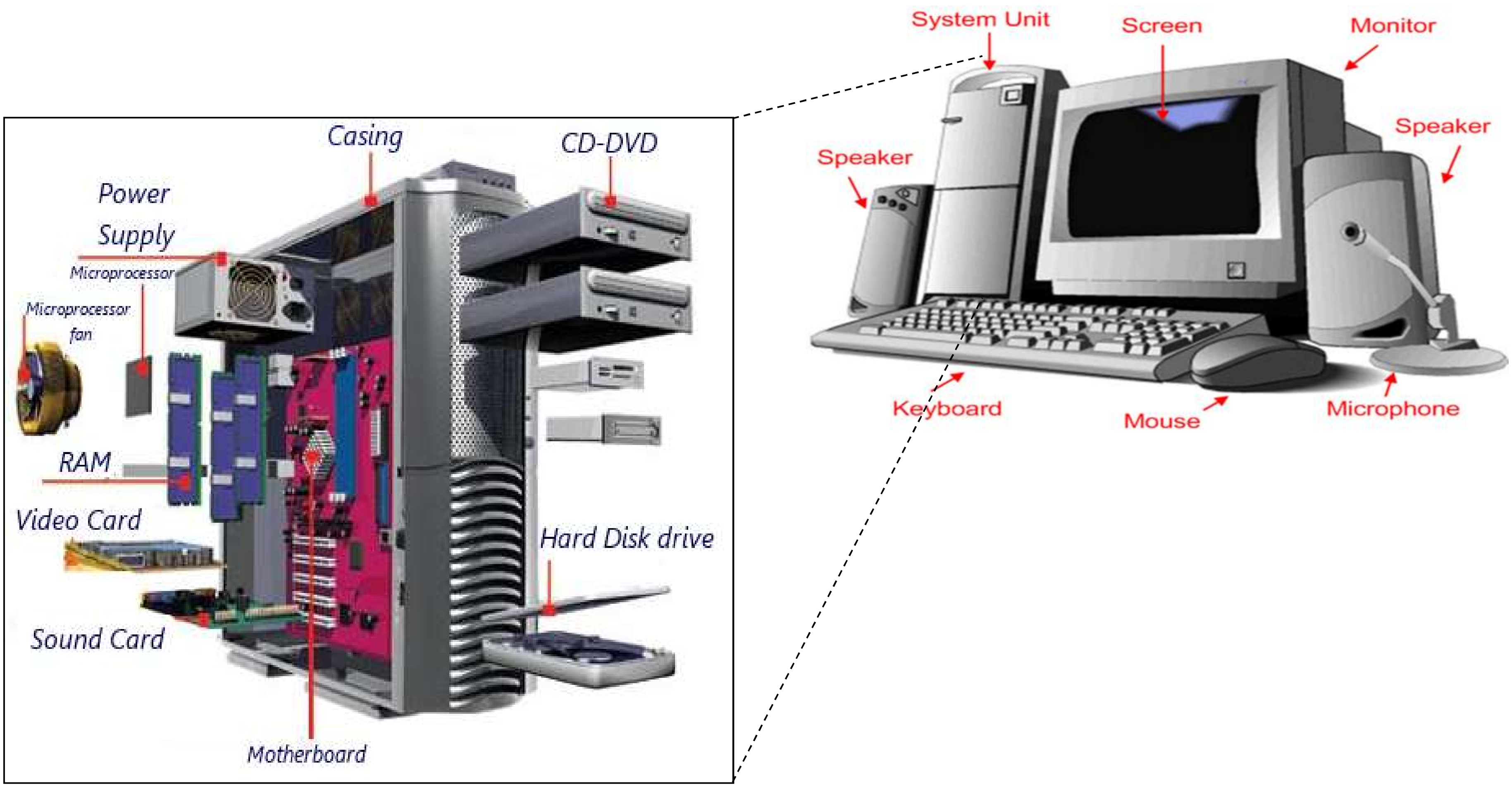
IBM-Compatibles PCs

- 1982: Compaq, HP and others released IBM-compatible PCs or IBM-Clones
  - IBM has obtained no patents on the PC design.
  - They used the same underlying CPUs and hardware chips
  - They reversed engineered and re-implemented BIOS
  - They relied on customized version of MS-DOS
- 1985: IBM clones dominated PC sales
  - Until recently, most desktop and laptop computers could still run MS-DOS based programs from the 80's and 90's.
- 2005: IBM sold its PC division to Lenovo.

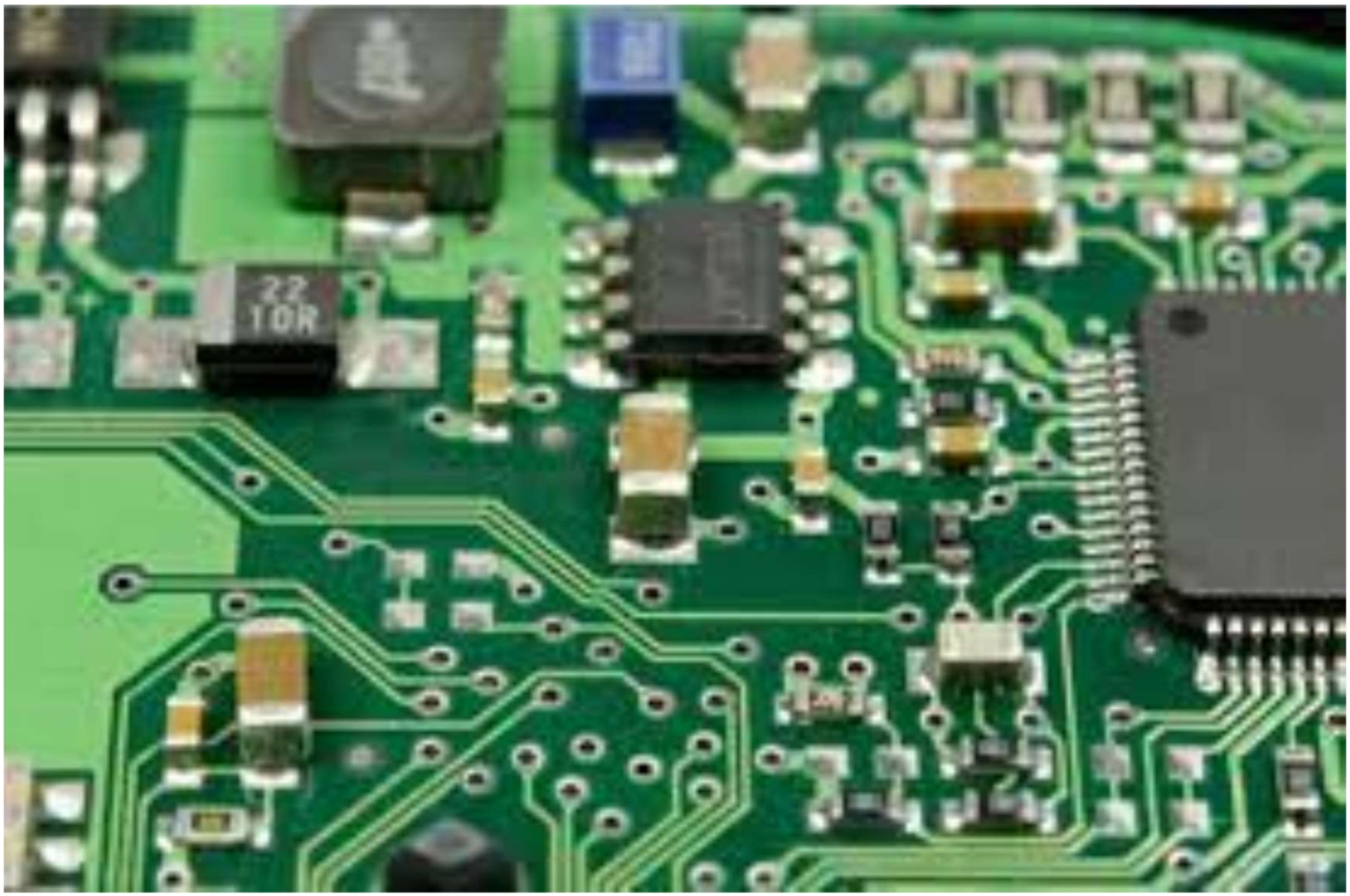
OS and Hardware Relationship (not required in the exam)



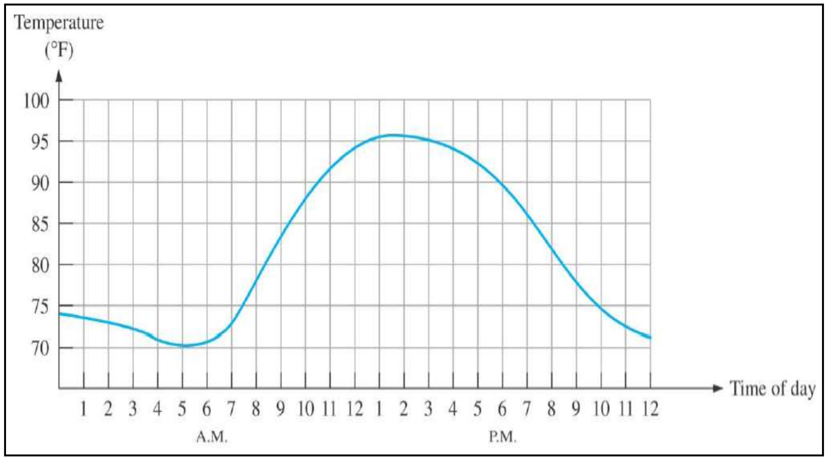
Typical Modern Computer Hardware Components

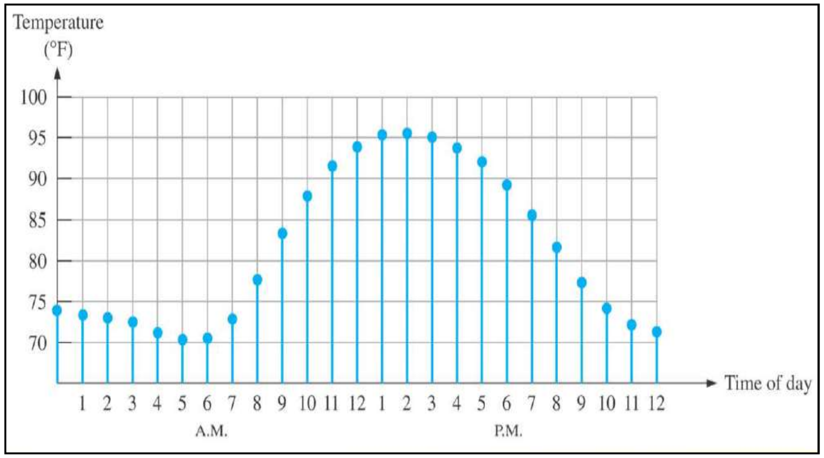


Lecture 01: Introduction
Part 2 – Digital Signals and Circuits



Digital and Analog Quantities

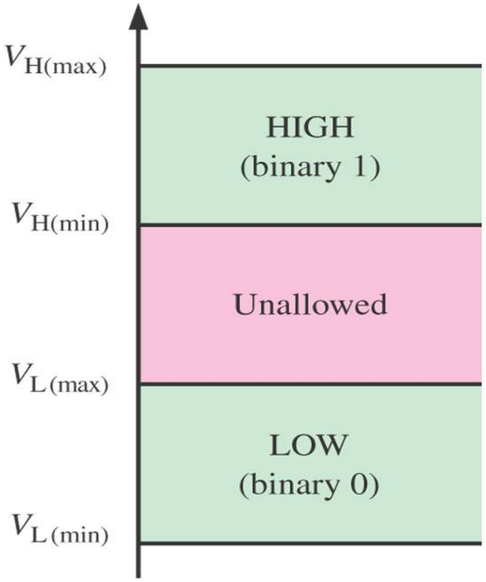




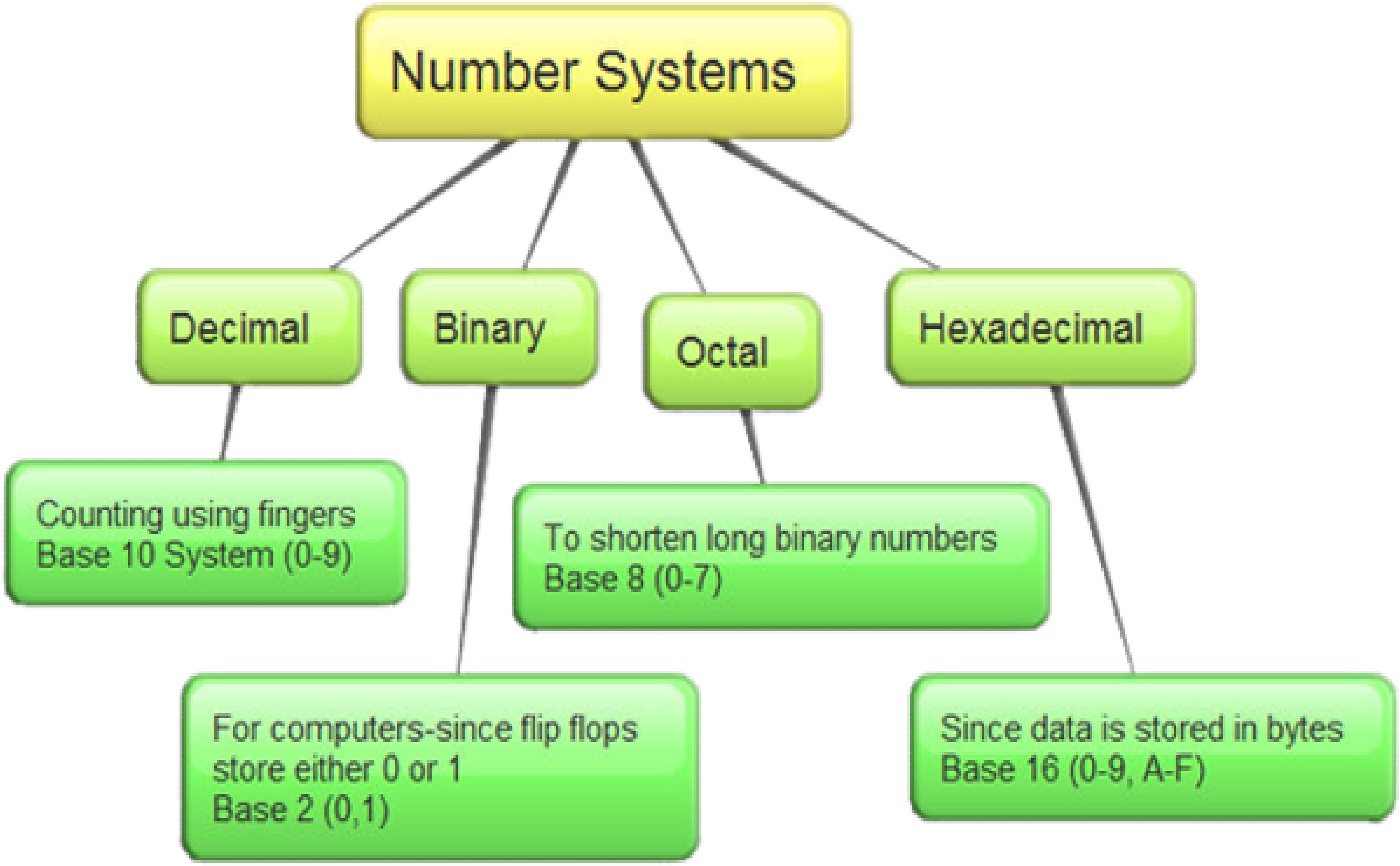
Analog quantities have continuous values

Digital quantities have discrete sets of values

- **Sampling** Converts Analog quantity to Digital Quantity.
- Each discrete number will be represented in Digital Systems (like computer, Smartphone,..etc) as a binary number with set of bits.
- Each bit can have two values **0** or **1**
- **The Binary values 0 and 1 are represented by voltage levels.**
- They can also be called LOW and HIGH, where **LOW = 0** and **HIGH = 1**



Number Systems
(not required in the exam)



Binary Numbers

- The binary digit is called bit which can be either 0 or 1
- 1 Byte = 8 bits
- Right most bit is least significant bit LSB
- Left most bit is most significant bit MSB
- Each position is weighted by a factor of 2.
- Weight of each bit depends on position

1 Byte = 8 bits

Binary Number Example	1	1	0	1	0	0	1	1
Weights	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Decimal Equivalent	128	64	32	16	8	4	2	1

MSB LSB

Counting in Binary

Decimal	Binary			
	2^3	2^2	2^1	2^0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Hexadecimal Numbers

- The Hexadecimal (simply Hex) is a positional number system that uses a base of sixteen.
- Software developers and system designers widely use hexadecimal numbers because they provide a human-friendly representation of binary-coded values.
- Each hexadecimal digit represents four bits.
- For example, an 8-bit byte can have values ranging from 00000000 to 11111111 in binary form (0 to 255 decimal), which can be conveniently represented as 00 to FF in hexadecimal.

Conversion Table Decimal - Binary - Hexadecimal		
Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Clock Cycles

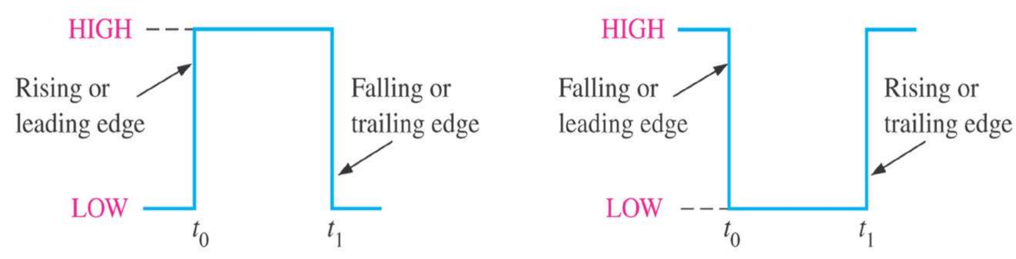
The **clock signal** is a 1-bit signal that oscillates between a ”1” and a ”0” with a certain frequency.

When the clock transitions from a ”0” to a ”1” it is called the **positive edge**, and when the clock transitions from a ”1” to a ”0” it is called the **negative edge**.

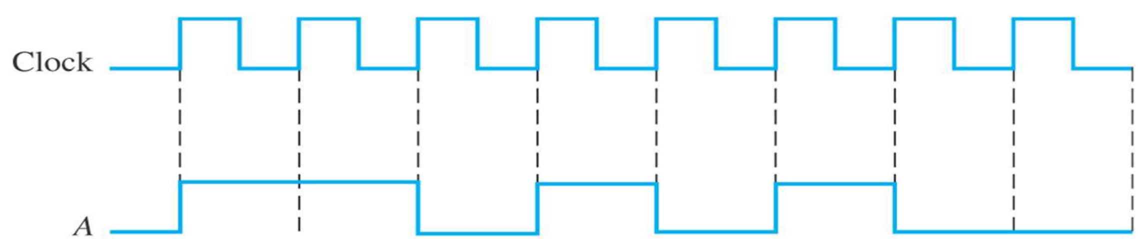
The time it takes to go from one positive edge to the next positive edge is known as the **clock cycle**.

The number of clock cycles that can fit in 1 second is called the **clock frequency** and it is measured in units of cycles per second.

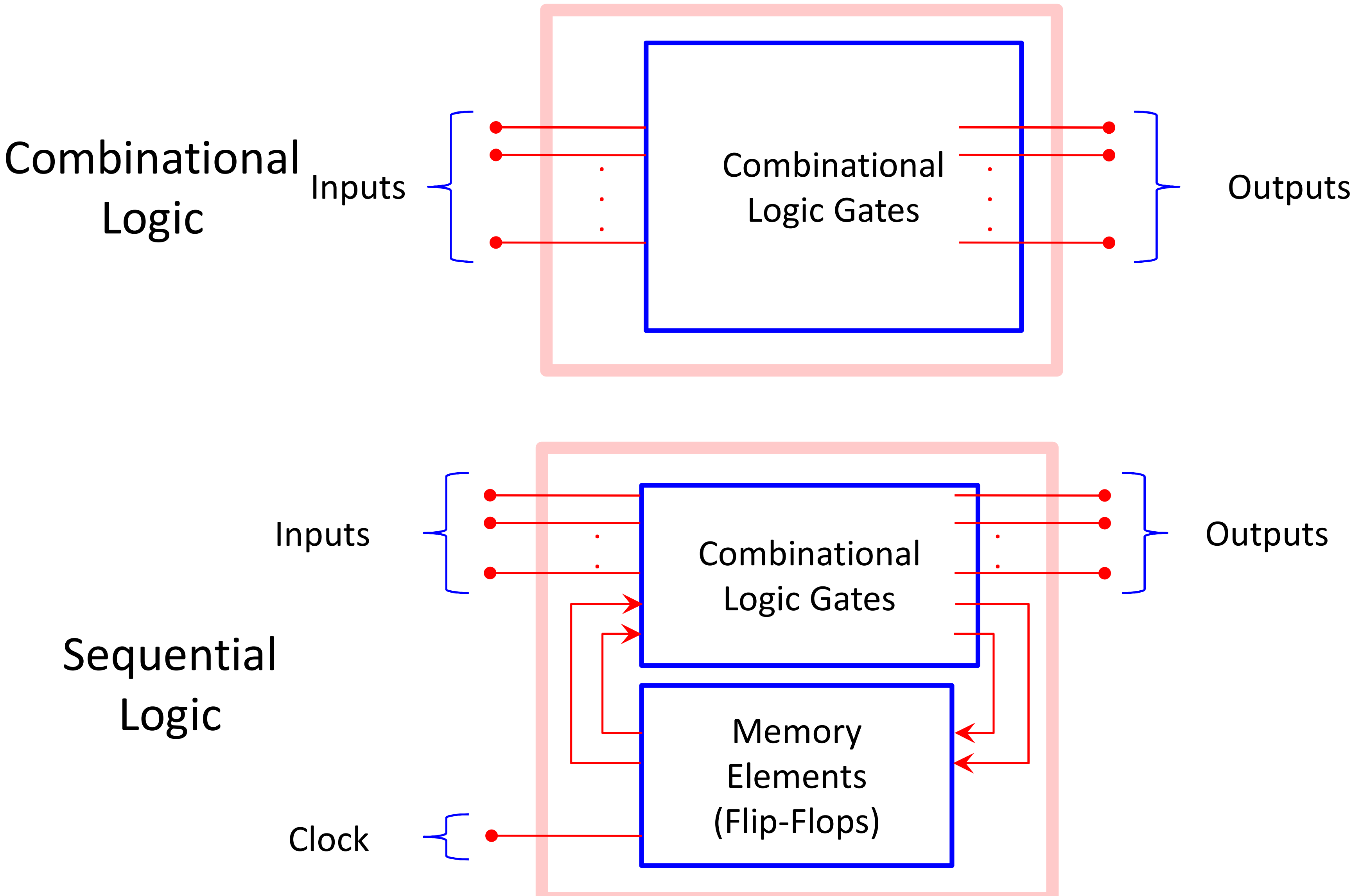
$$\text{Clock Frequency} = \frac{1}{\text{Clock Period}}$$



(a) Positive–going pulse (b) Negative–going pulse

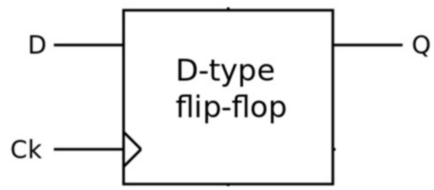


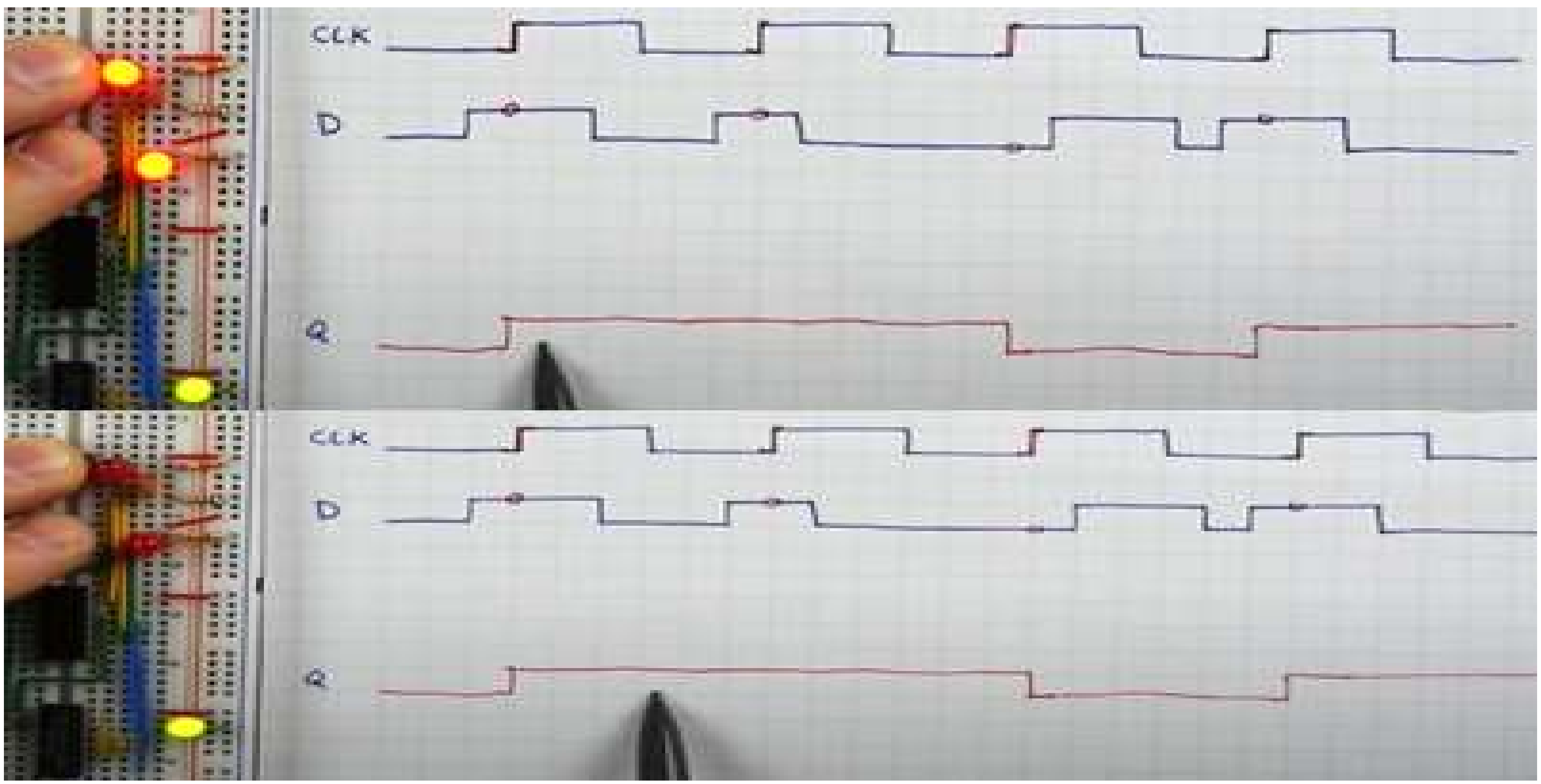
Combinational & Sequential Logic



Flip-Flop

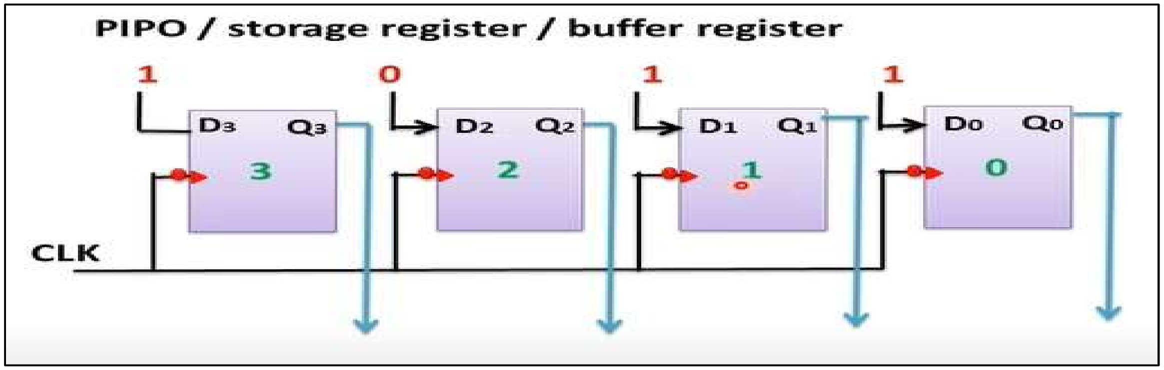
- A **Flip-Flop** is a memory to store 1 bit, in which only the clock edge determines when a new bit is entered.
- In D-type Flip-Flop the output Q assumes the state of the D input on the triggering edge of a clock pulse and keep memorizing this value till next clock edge.





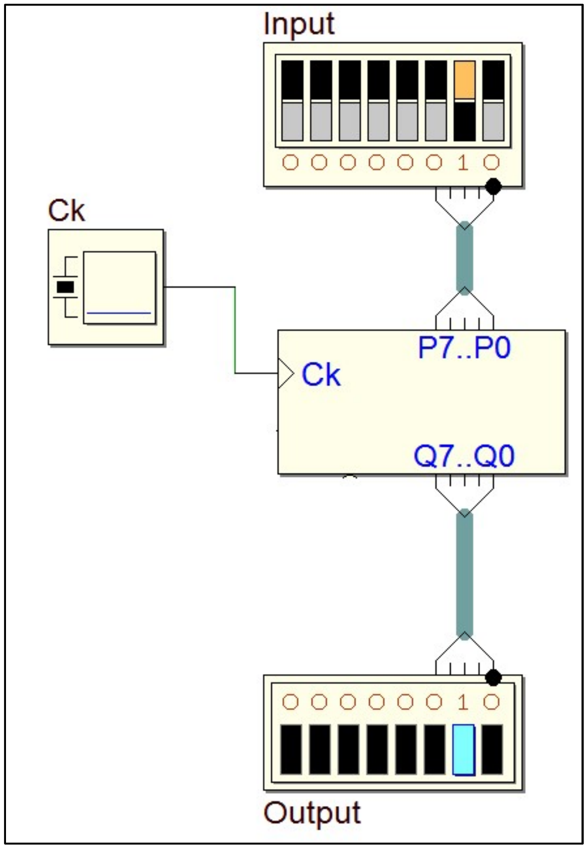
Register

- A register has a group of *n* flip-flops so it is capable of storing *n* bits of information.
- The maximum value that a register can represent is 2^n-1
- Example: A 4-bit register. A new 4-bit data is loaded every clock cycle. The maximum value for this register is 2^4-1 =15
- Register Functions:
  - ❖ Read data from register
  - ❖ Write new data into register
  - ❖ Shift the data within register

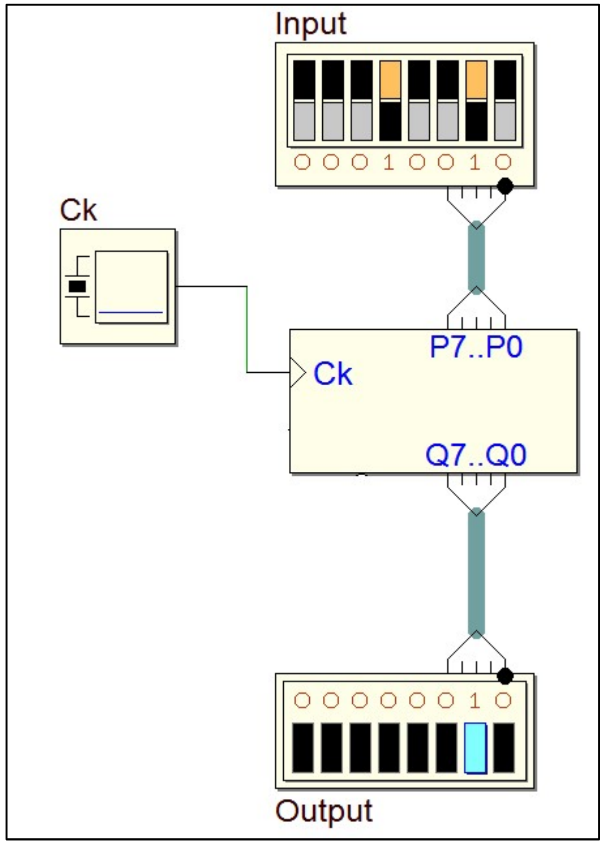


Register States & Clock Signal

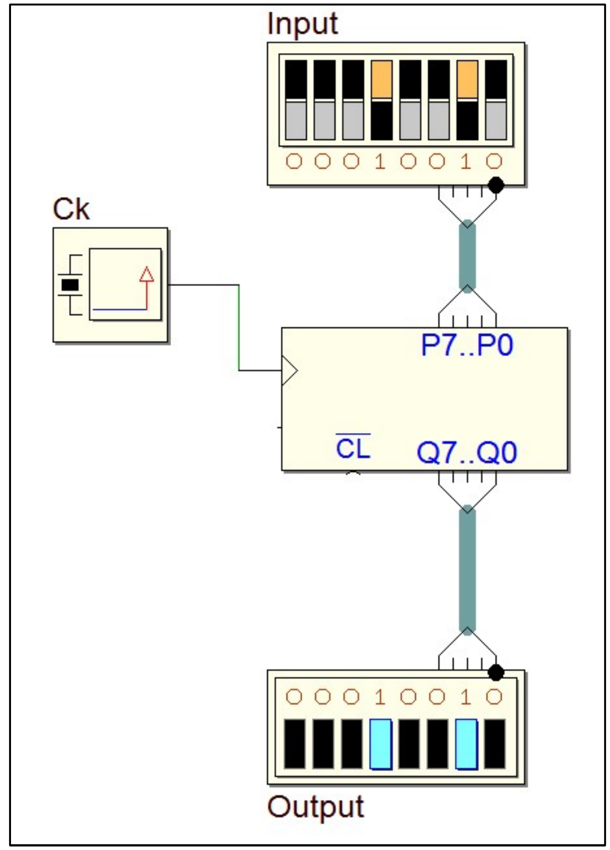
The output of the register will memorize the last state even when input changes until clock edge present



State 1



State 2



State 3

Lecture 01: Introduction
Part 3– Von-Neumann Architecture

Modern computers, based on the Intel x86 family of processors are based on Von-Neumann Architecture with below features:

- All instructions and data are stored in memory.
- An instruction and the required data are fetched (brought) from memory to the CPU for execution.
- Input and Output devices interface the CPU with the outside world.

CPU: Central Processing Unit

❖At the heart of all personal computers and smart phones sits a microprocessor.

❖In the world of computers, the terms ***microprocessor*** (**μP**)*,* ***processor,*** and ***CPU*** have the same meaning.

❖Microprocessors also control the logic of almost all digital devices, from clocks to fuel-injection systems for automobiles.

❖Three basic characteristics differentiate microprocessors:

i) **Instruction set**: The set of instructions that the CPU can execute.

ii) **Bus width**: The number of bits processed in a single instruction.

iii) **Clock speed** (in MHz): determines how many instructions per second the processor can execute.

The Bus Concept

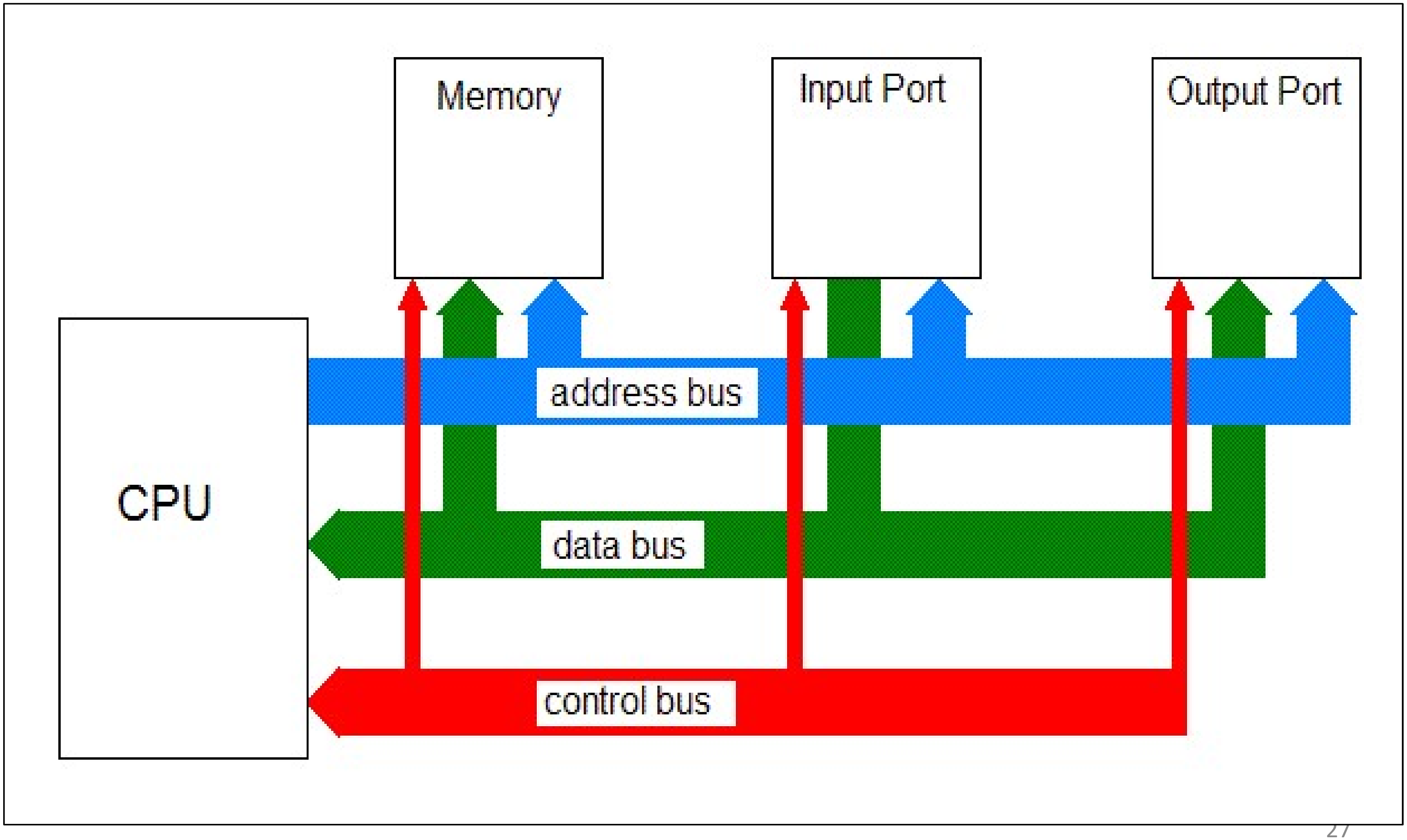
Bus: is a collection of wires through which address, data, or control signals are transmitted

- The speed of a bus is measured in megahertz (MHz)
- The size of a bus (its **width**) is how many bits it can transfer at a time
  E.g. a 64 bit computer has buses with 64-bit widths

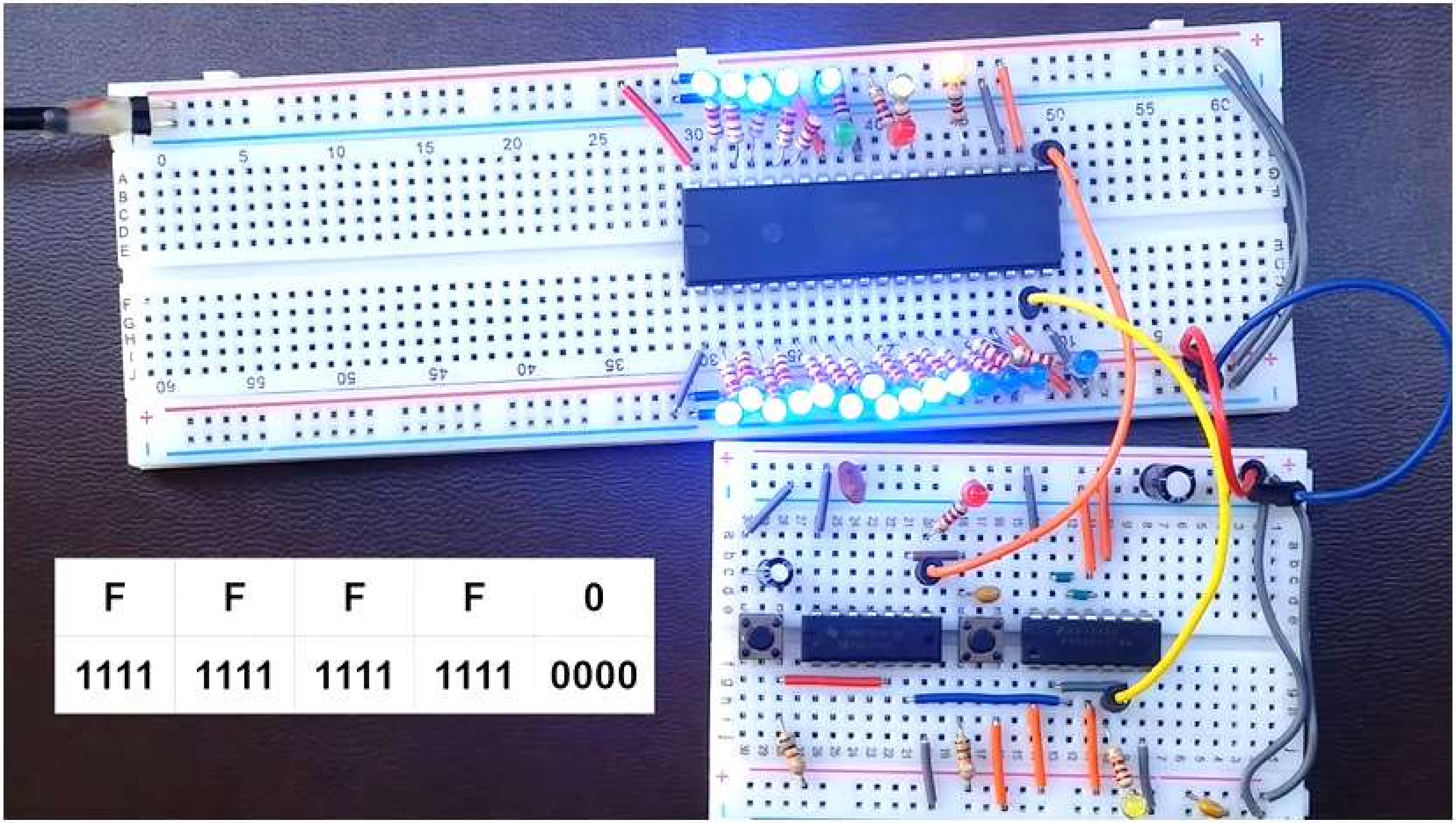
Bus Types:

- An address bus sends a memory address along the bus from the CPU *to the memory*. To fetch/write data, the CPU needs to tell the RAM the address
- A **data bus** sends the actual data *to and from the memory.*
- A control bus carries commands from the CPU and status messages from other hardware devices.

Von-Neumann Computer Architecture Block Diagram



CPU Buses Practical Demo



CPU Buses Practical Demo