Vietnam National University HCMC
INTERNATIONAL UNIVERSITY

Lecture 3: Hardware layer

IT064IU - INTRODUCTION TO COMPUTING

S1_2022-23

Contents

1. Gates and circuits
   - Basic gates such as NOT, AND, OR, XOR, NAND, NOR
   - How gates are implemented using transistors.
   - How to combine basic gates into circuits.
   - Describe the behavior of a gate or circuit using Boolean expressions, truth tables, and logic diagrams.
   - Some circuits: adder, multiplexer, S-R latch
2. Computing components

Definitions

Gate (logic gate): A device that performs a basic operation on electrical signals, accepting one or more input signals and producing a single output signal

Circuit: A combination of interacting gates designed to accomplish a specific logical function

To describe the behavior of gates and circuits:

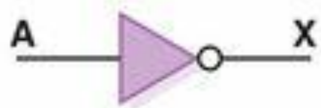
- **Boolean algebra** A mathematical notation for expressing two-valued logical functions
- **Logic diagram**: A graphical representation of a circuit; each type of gate has its own symbol
- **Truth table**: A table showing all possible input values and the associated output values

Gates – NOT, AND

Boolean Expression

$X = A'$

Logic Diagram Symbol



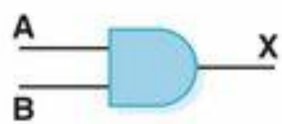
Truth Table

A	X
0	1
1	0

Boolean Expression

$X = A \cdot B$

Logic Diagram Symbol



Truth Table

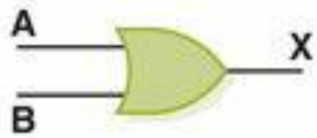
A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

Gates – OR, XOR

Boolean Expression

$X = A + B$

Logic Diagram Symbol



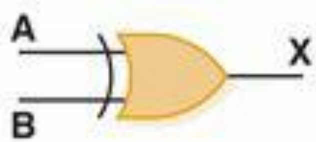
Truth Table

A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

Boolean Expression

$X = A \oplus B$

Logic Diagram Symbol



Truth Table

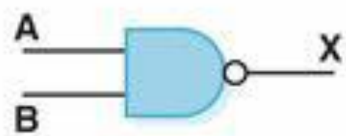
A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

Gates – NAND, NOR

Boolean Expression

$X = (A \cdot B)'$

Logic Diagram Symbol



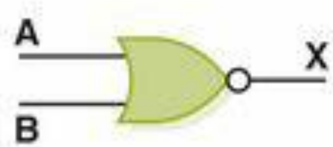
Truth Table

A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

Boolean Expression

$X = (A + B)'$

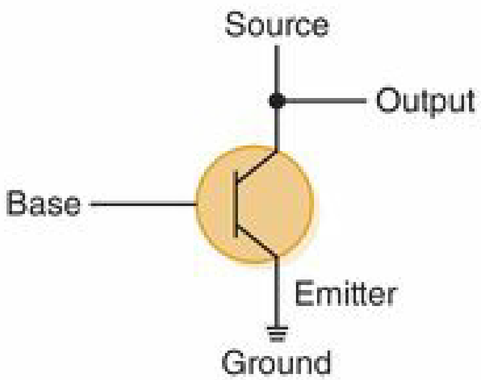
Logic Diagram Symbol

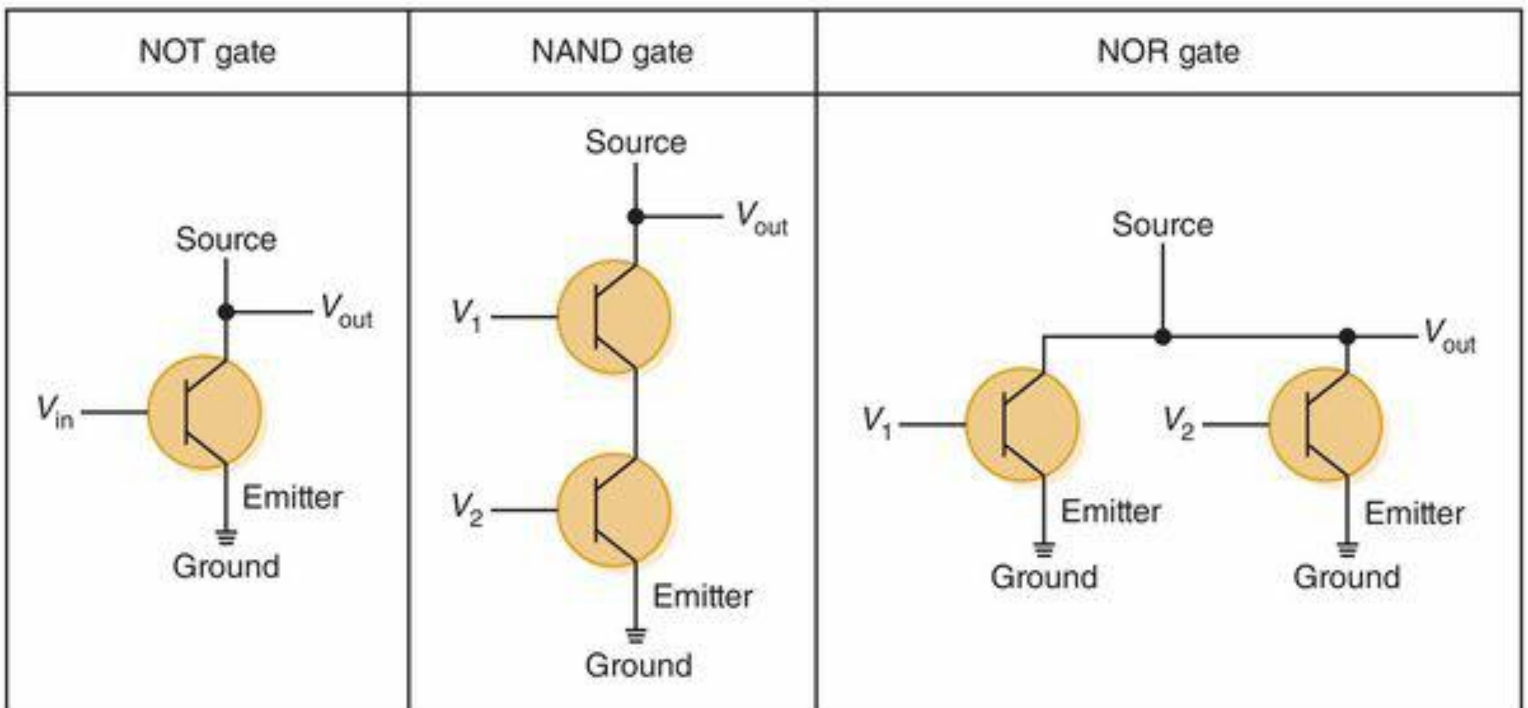


Truth Table

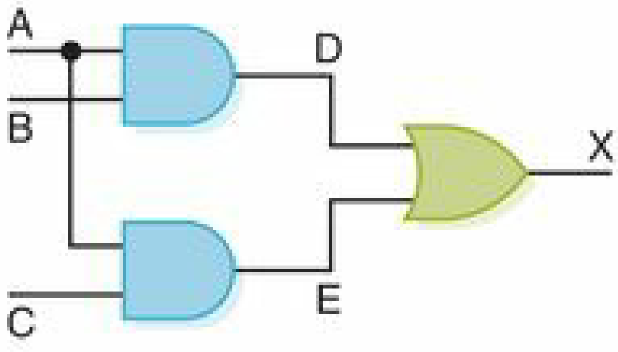
A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

Constructing Gates - transistors





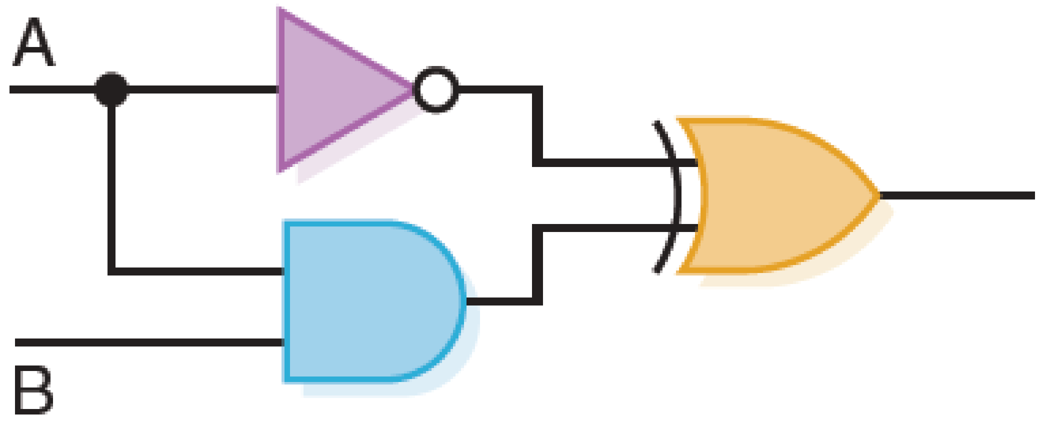
Circuit: AB + AC



A	B	C	D	E	X
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	1	1

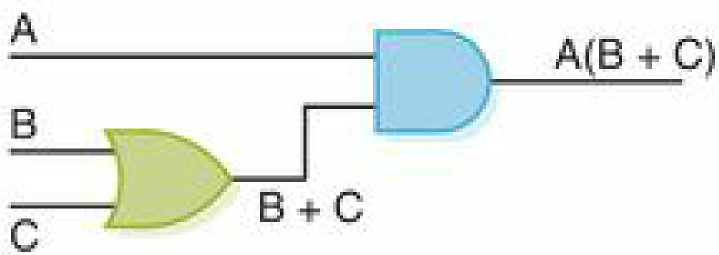
Write the Boolean expression and True table?

60. Show the behavior of the following circuit with a truth table:



Circuit equivalence

Circuit equivalence: The same output for each corresponding input–value combination for two circuits



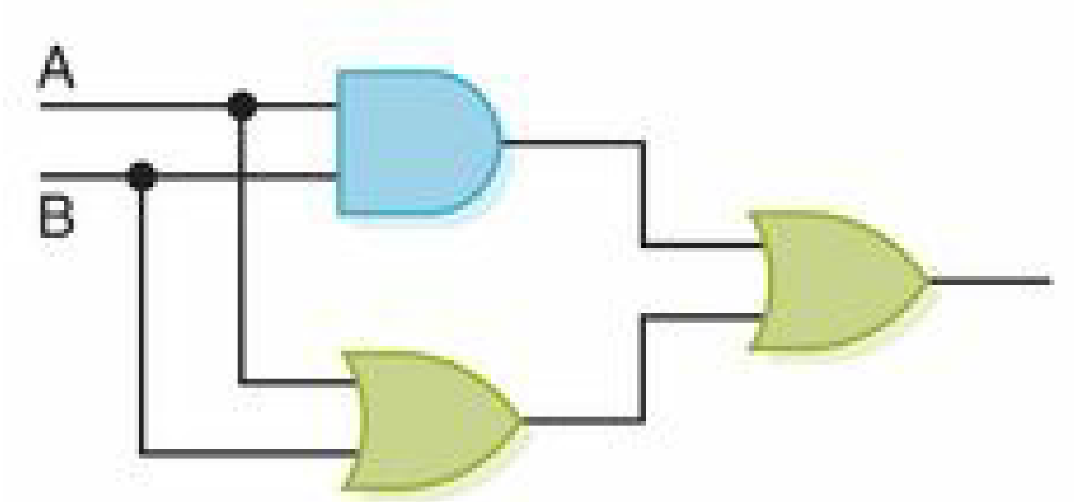
A	B	C	B + C	A(B + C)
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

Boolean algebra

PROPERTY	AND	OR
Commutative	AB = BA	A + B = B + A
Associative	(AB) C = A (BC)	(A + B) + C = A + (B + C)
Distributive	A (B + C) = (AB) + (AC)	A + (BC) = (A + B) (A + C)
Identity	A1 = A	A + 0 = A
Complement	A(A′) = 0	A + (A′) = 1
De Morgan’s law	(AB)′ = A′ OR B′	(A + B)′ = A′B′

Other example

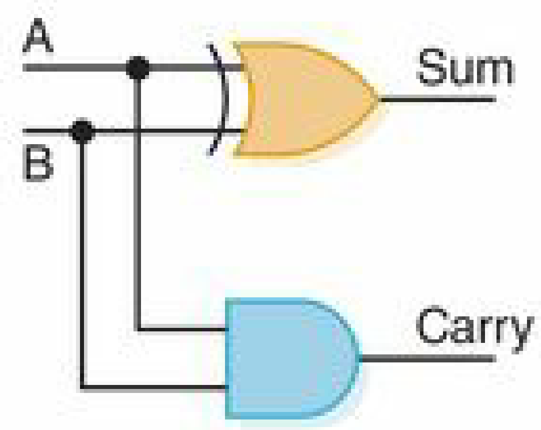
1. Draw a circuit diagram corresponding to the following Boolean expression: A’B + (B + C)’
2. Show the behavior of the following circuit with a truth table: AB + (A+B) = (AB+A) +B = A(B+1) + B = A1 + B = A+B



Half adder

Adder: An electronic circuit that performs an addition operation on binary values.

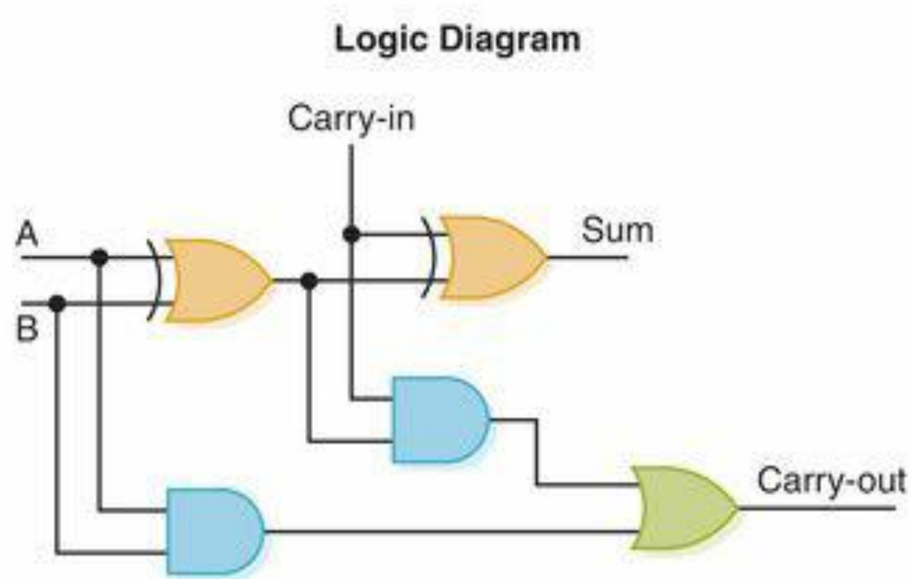
Half adder: A circuit that computes the sum of two bits and produces the appropriate carry bit.



A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Full adder

Full adder: *A* circuit that computes the sum of two bits, taking an input carry bit into account.



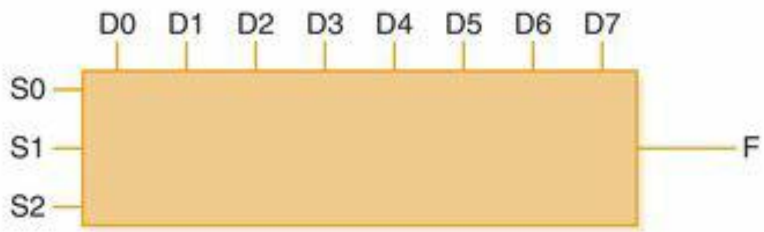
Truth Table

A	B	Carry-in	Sum	Carry-out
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

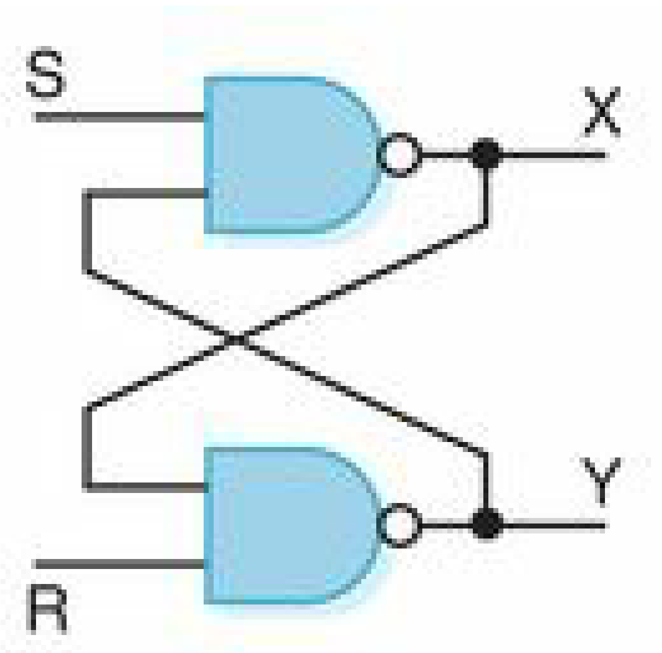
Multiplexer

Multiplexer: A circuit that uses a few input control signals to determine which of several input data lines is routed to its output

S0	S1	S2	F
0	0	0	D0
0	0	1	D1
0	1	0	D2
0	1	1	D3
1	0	0	D4
1	0	1	D5
1	1	0	D6
1	1	1	D7



Circuits as Memory - S-R latch



- Normally, S=R=1 → S-R latch remains the state X=0 or X=1
- S = 0 for a moment → X=1 (Set)
- R = 0 for a moment → X=0 (Reset)

Integrated Circuits

Integrated circuit (chip): A piece of silicon on which multiple gates have been embedded.

Abbreviation	Name	Number of Gates
SSI	Small-scale integration	1 to 10
MSI	Medium-scale integration	10 to 100
LSI	Large-scale integration	100 to 100,000
VLSI	Very-large-scale integration	more than 100,000

Contents

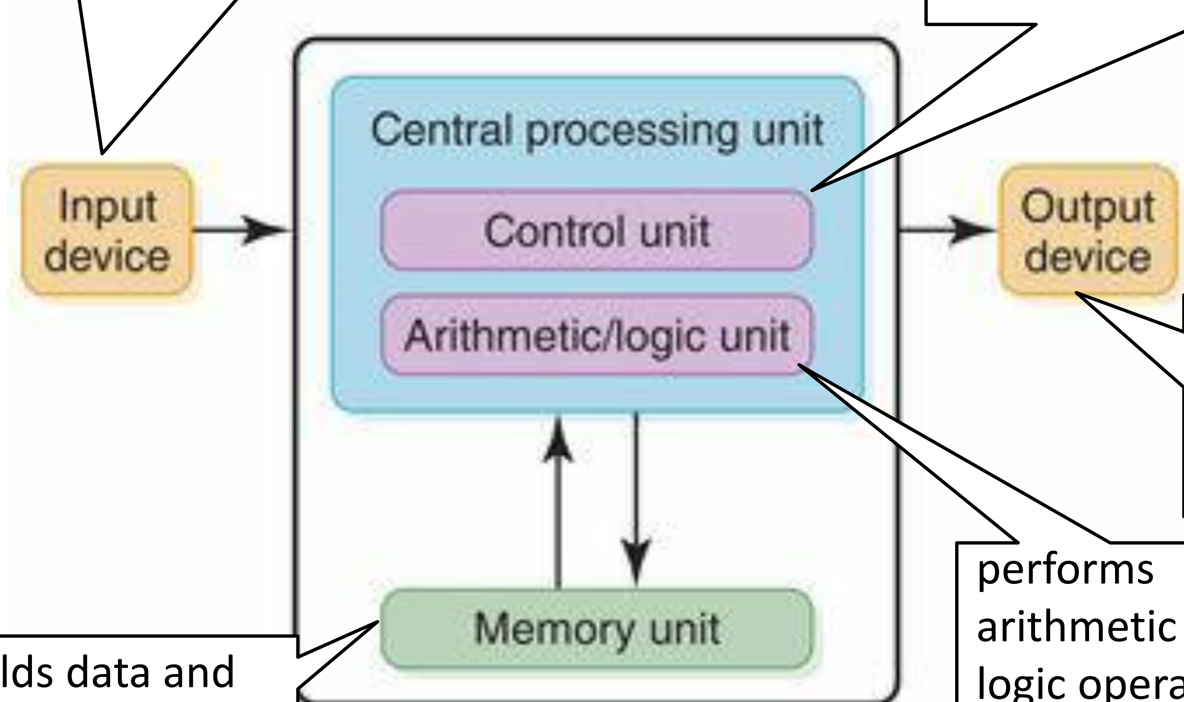
1. Gates and circuits
2. Computing components.
   - Von Neumann machine architecture
   - Components and their function in a von Neumann machine
   - Fetch–decode–execute cycle of the von Neumann machine
   - Computer memory
   - Embedded systems

Von Neumann architecture

A system that can store, retrieve and process data

moves data from the outside world into the computer

acts as the stage manager to ensure that all the other components act in concert



moves results from inside the computer to the outside world

performs arithmetic and logic operations on data

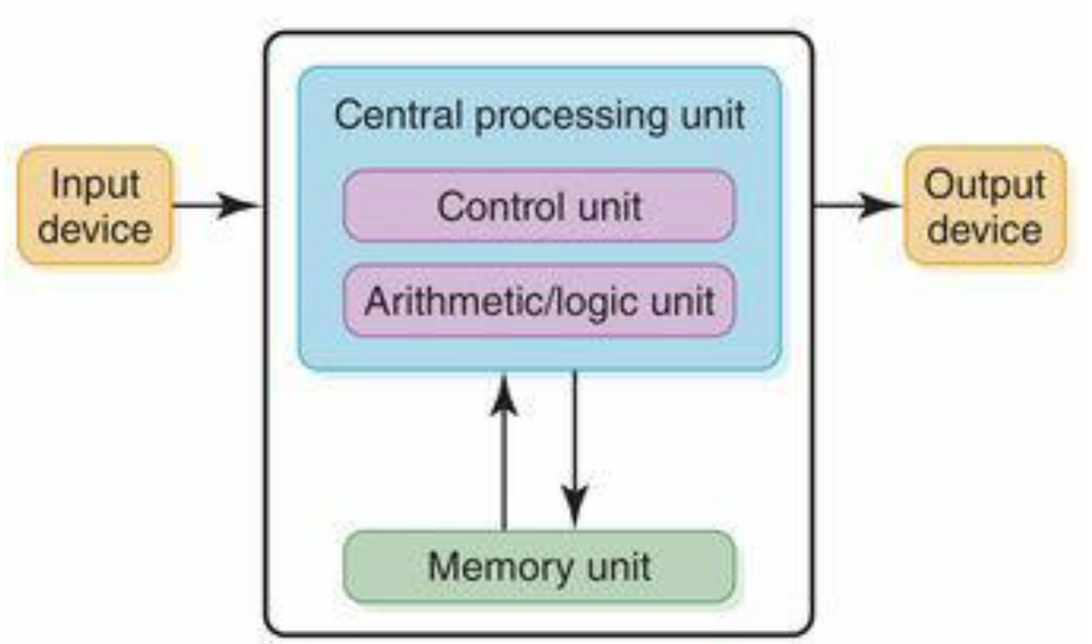
holds data and instructions

Memory

Bit, byte, word

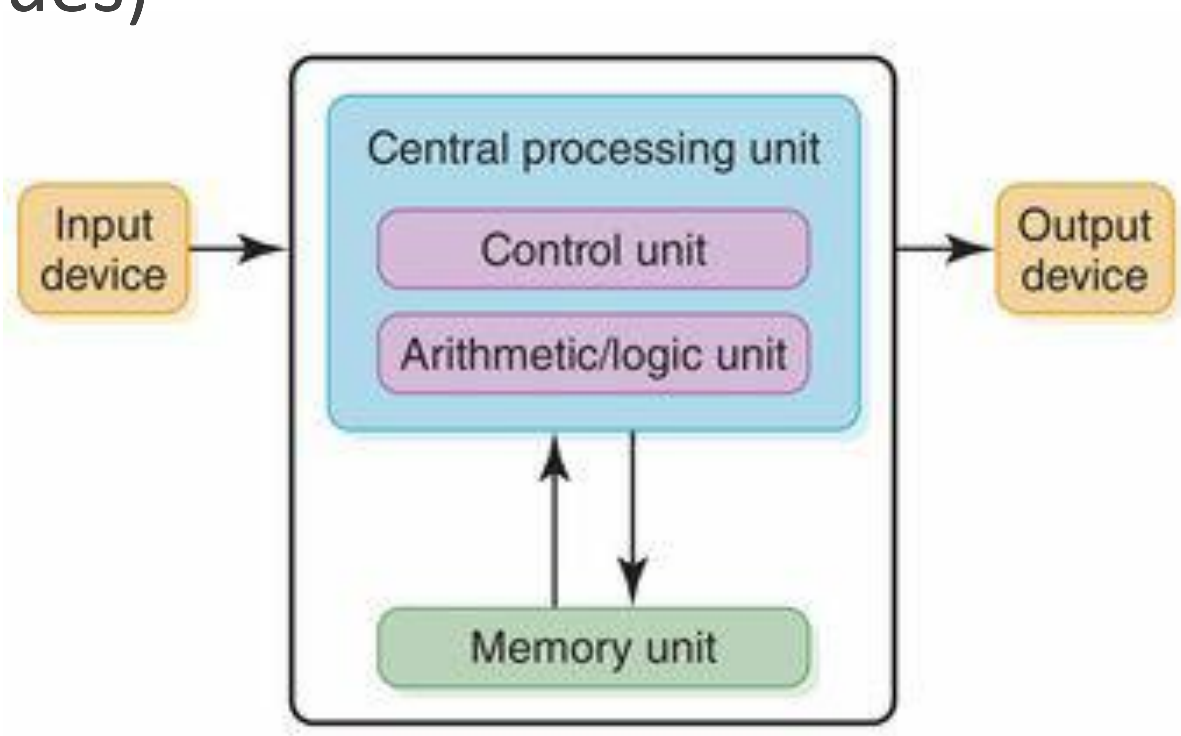
Computer memory is addressability. Each cell has an addressed location. (cell can be bit, byte, or word depending on computing systems).

Address	Contents
00000000	11100011
00000001	10101001
. . .	. . .
11111100	00000000
11111101	11111111
11111110	10101010
11111111	00110011



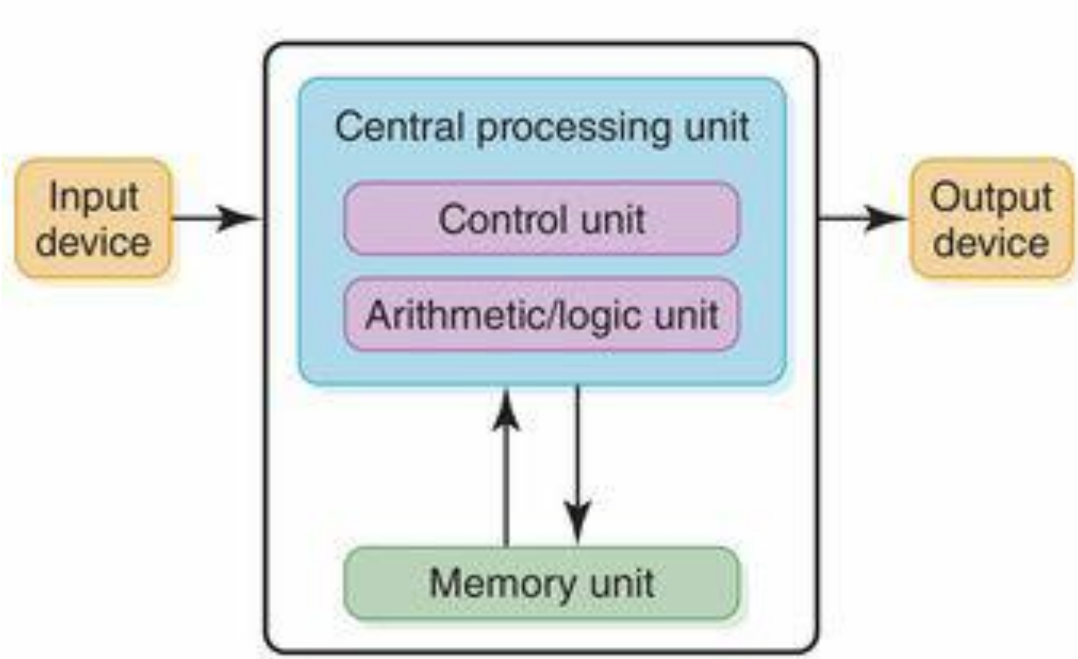
Arithmetic/logic unit (ALU)

Arithmetic/logic unit (ALU): The computer component that performs arithmetic operations (addition, subtraction, multiplication, and division) and logical operations (comparison of two values)



Input/Output Units

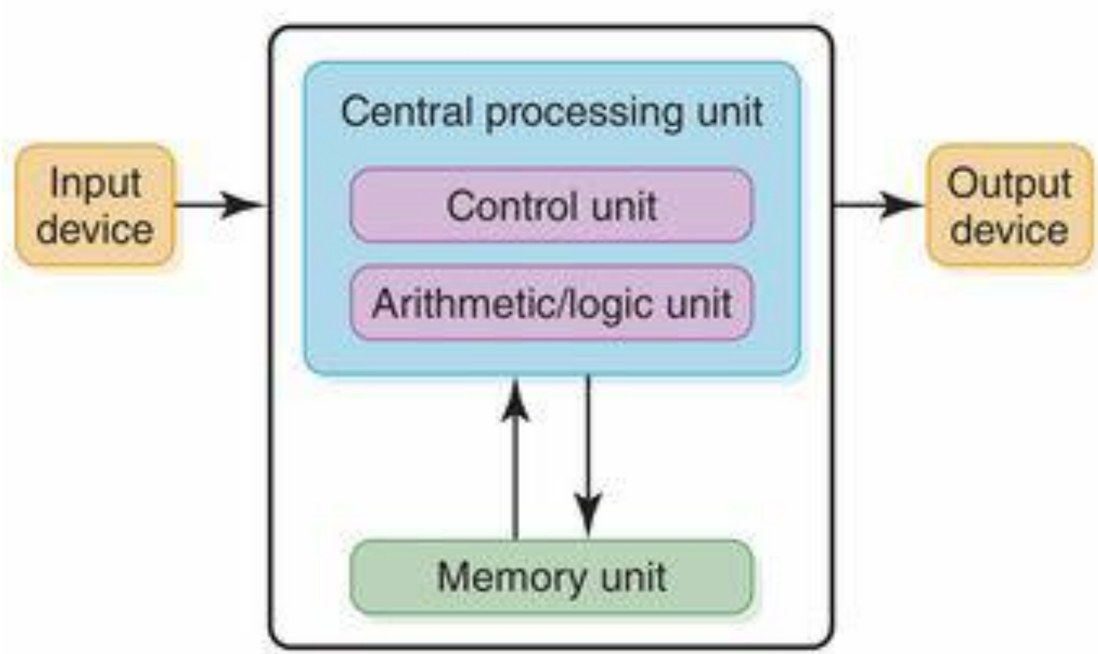
- **Input unit**: A device that accepts data to be stored in memory
- **Output unit**: A device that prints or displays data stored in memory or sends an information stored in memory to another device



Central Processing Unit

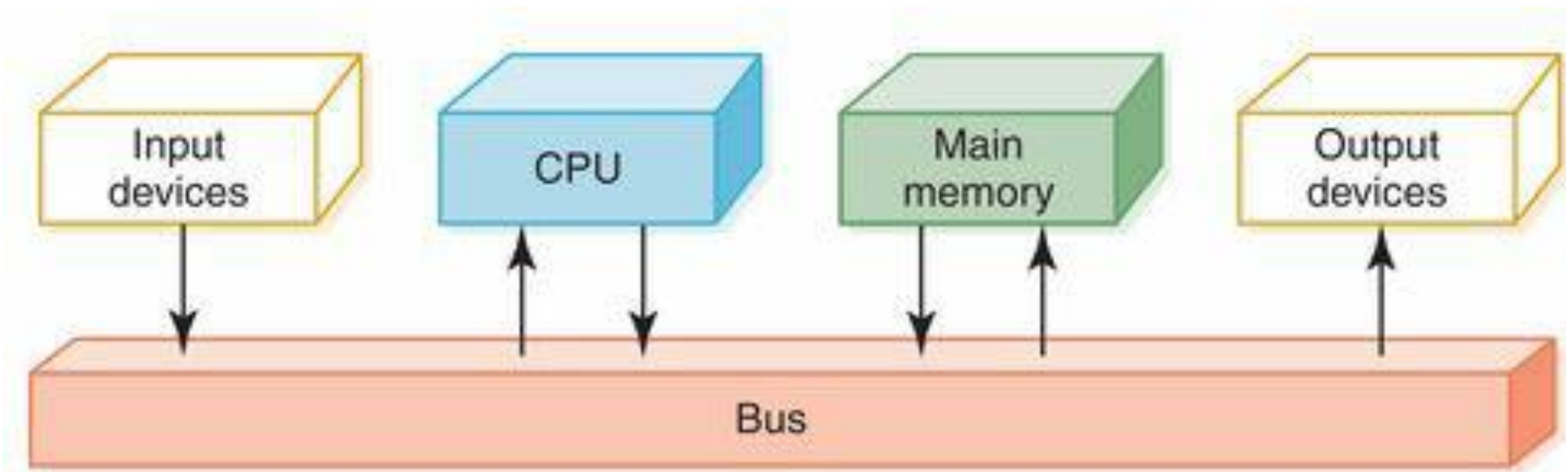
Control Unit

- Instruction register (IR) The register that contains the instruction currently being executed
- Program counter (PC) The register that contains the address of the next instruction to be executed



Bus

Bus width The number of bits that can be transferred in parallel over the bus



Group work

Understanding the key components of a modern computer.

Your friend which is an IT student want to buy a laptop, please recommend the configuration according to his/her budget:

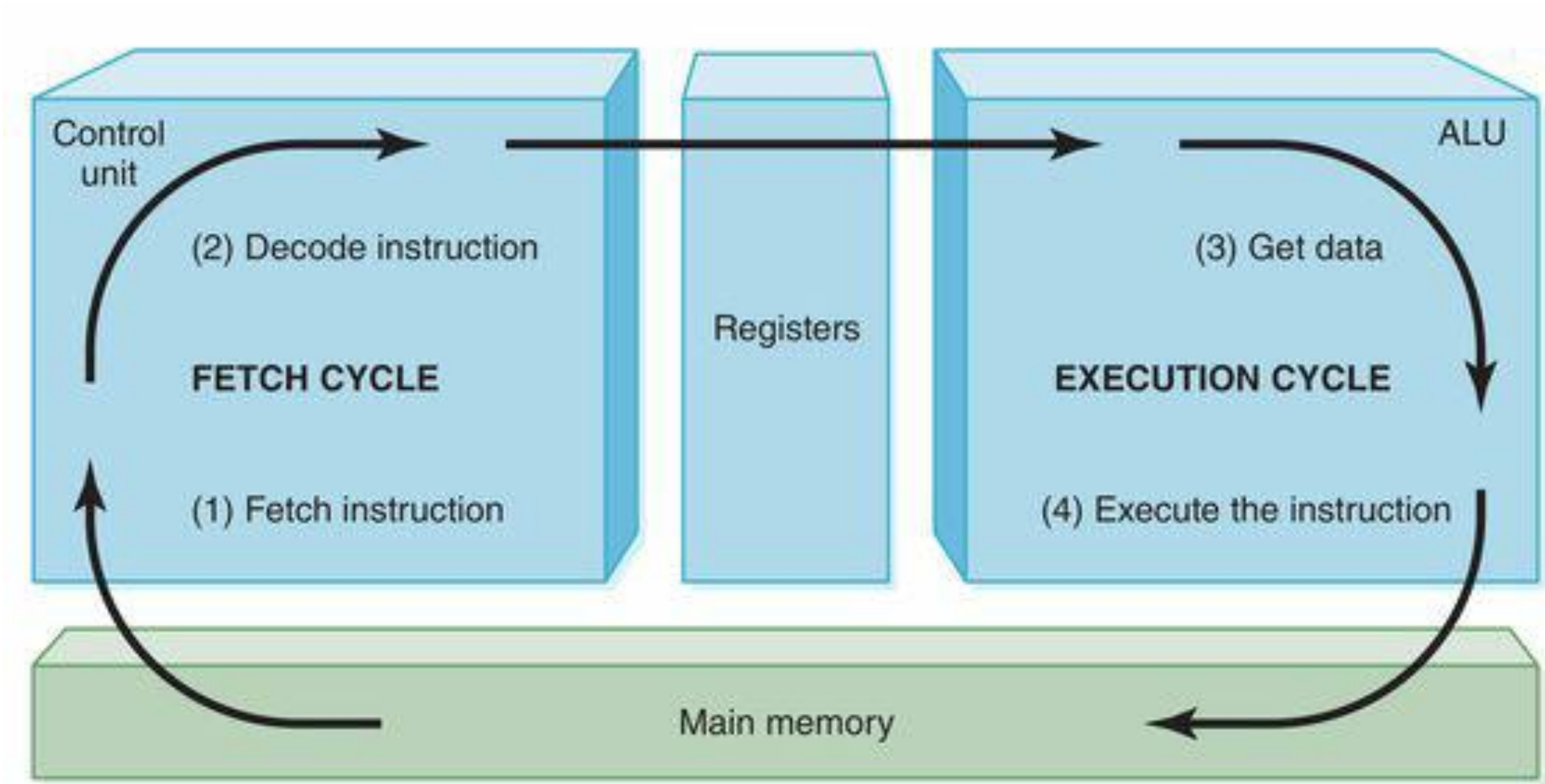
(1) <20 mil. VND;

(2) >=20 and <=40 mil. VND

(3) >=40 mil. VND

Please explain your choice.

Fetch–Execute Cycle



Fetch–Execute Cycle

Both instruction and data are addressable and stored in memory:

- Fetch the next instruction.
- Decode the instruction.
- Get data if needed.
- Execute the instruction.

RAM and ROM

RAM (random-access memory):

- Each cell (e.g., each byte) can be directly accessed
- Volatile memory

ROM (read-only memory):

- The contents in locations in ROM cannot be changed, cannot be altered by a stored operation.
- Store the instructions that the computer needs to start itself

Secondary Storage Devices

Devices that can store large quantities of data. They are also known as *mass storage devices*.

- Magnetic Disks (HDD – hard disk drive)
- CDs and DVDs
- Flash Drives
- Solid State Drive (SSD)

Units

Match the power of 10 to its name or use: A. 10^{-12}; B. 10^{-9}; C. 10^{-6}; D. 10^{-3}; E. 10^{3}; F. 10^{6}; G. 10^{9}; H. 10^{12}; I. 10^{15}.

1. Nano; 2. Pico;

3. Micro; 4. Milli;

5. Tera; 6. Giga;

7. Kilo; 8. Mega;

9. Often used to describe processor speed

10. Often used to describe size of memory

11. Used in relation to Internet speeds

12. Latin for “thousandth”

Embedded Systems

Computers that are designed to perform a narrow range of functions or as part of a larger system.

An embedded system usually includes a single microprocessor chip with the programs stored in ROM.

Thank you ☺