Computer architecture

Hardware abstraction

&

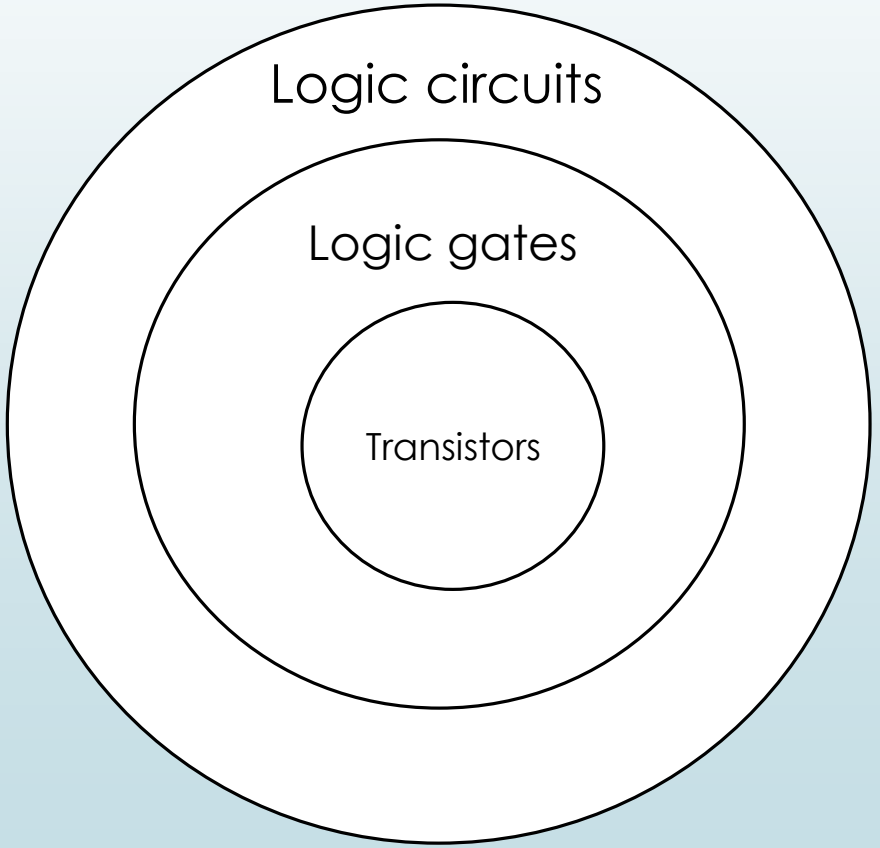
Software abstraction

Hardware abstraction

Transistor is the basic element of computer. It can turn a current on and off.

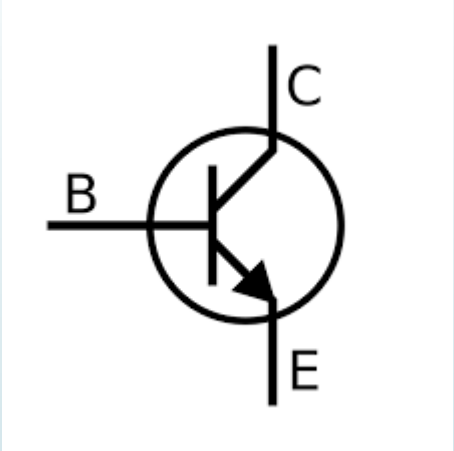
Logic gate (And, …) is built from transistors.

Logic circuit (Adder, …) is built from logic gates.



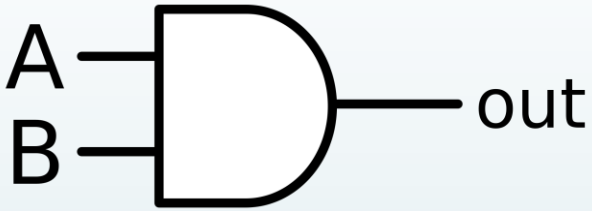
Transistor

A transistor contains a Base (B), Collector (C) and Emitter (E).



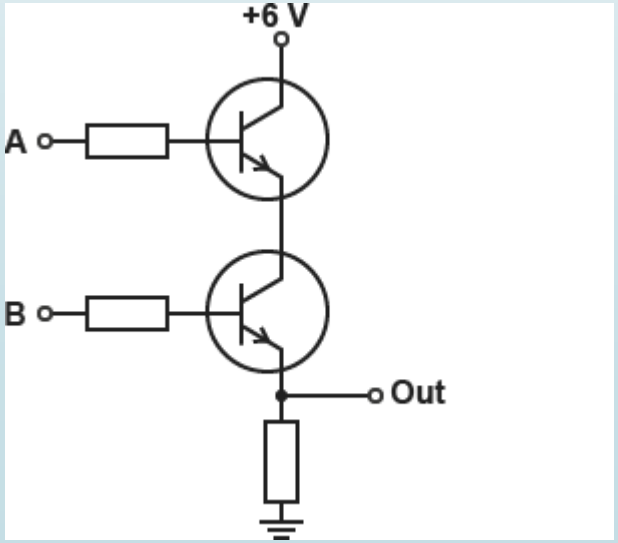
If a current flows on B it allows for a current to flow from C to E.

AND Gate

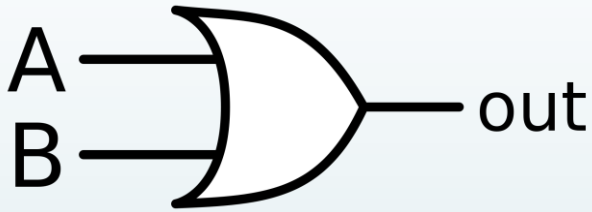


A	B	Out
0	0	0
0	1	0
1	0	0
1	1	1

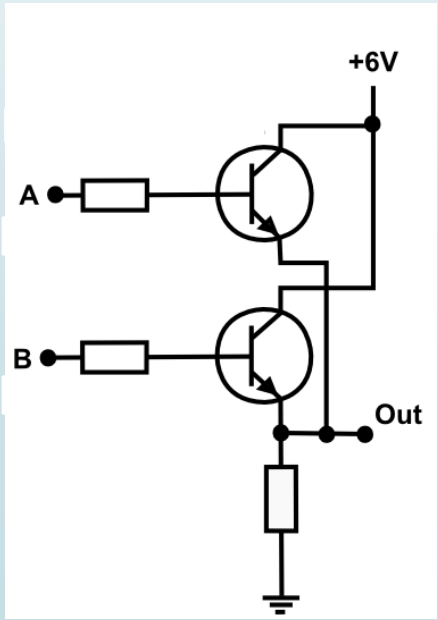
0: 0V
1: 6V



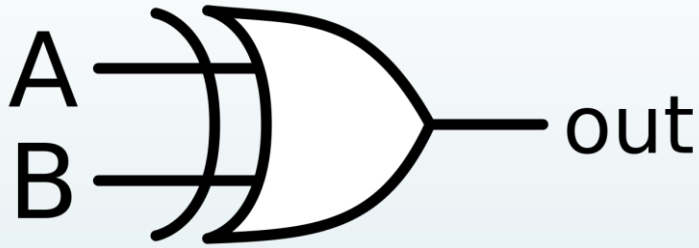
OR Gate



A	B	Out
0	0	0
0	1	1
1	0	1
1	1	1



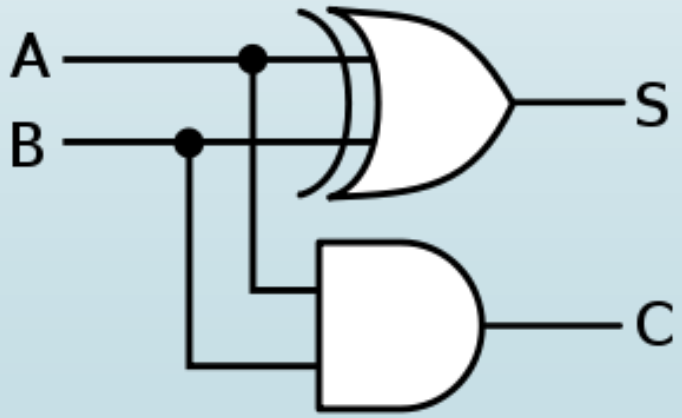
XOR Gate



A	B	Out
0	0	0
0	1	1
1	0	1
1	1	0

Half adder circuit

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



Software abstraction

Passage from a high-level language to a lower level language understood by the machine via several intermediate languages.

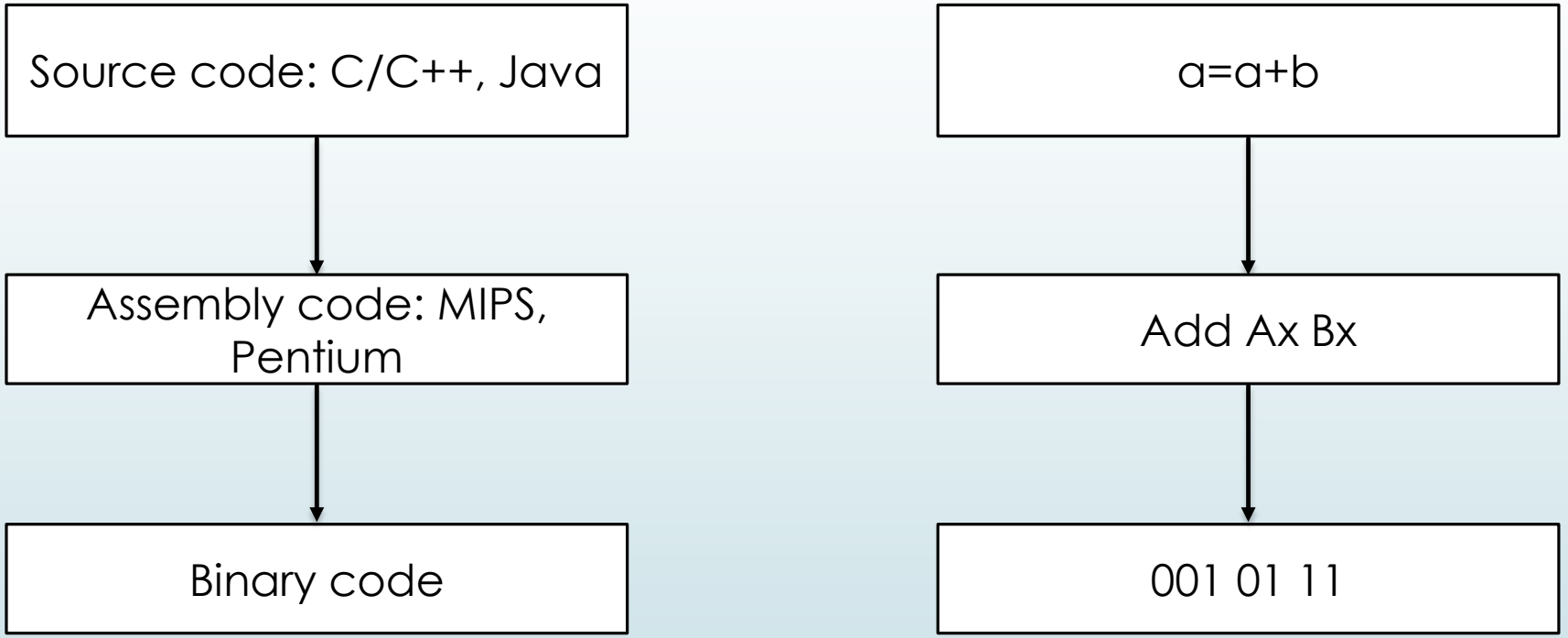
Human language X: read a video

Programmer language Y: load(video)

Binary language Z: 011…010

Conversion: X → … → Y → … → Z

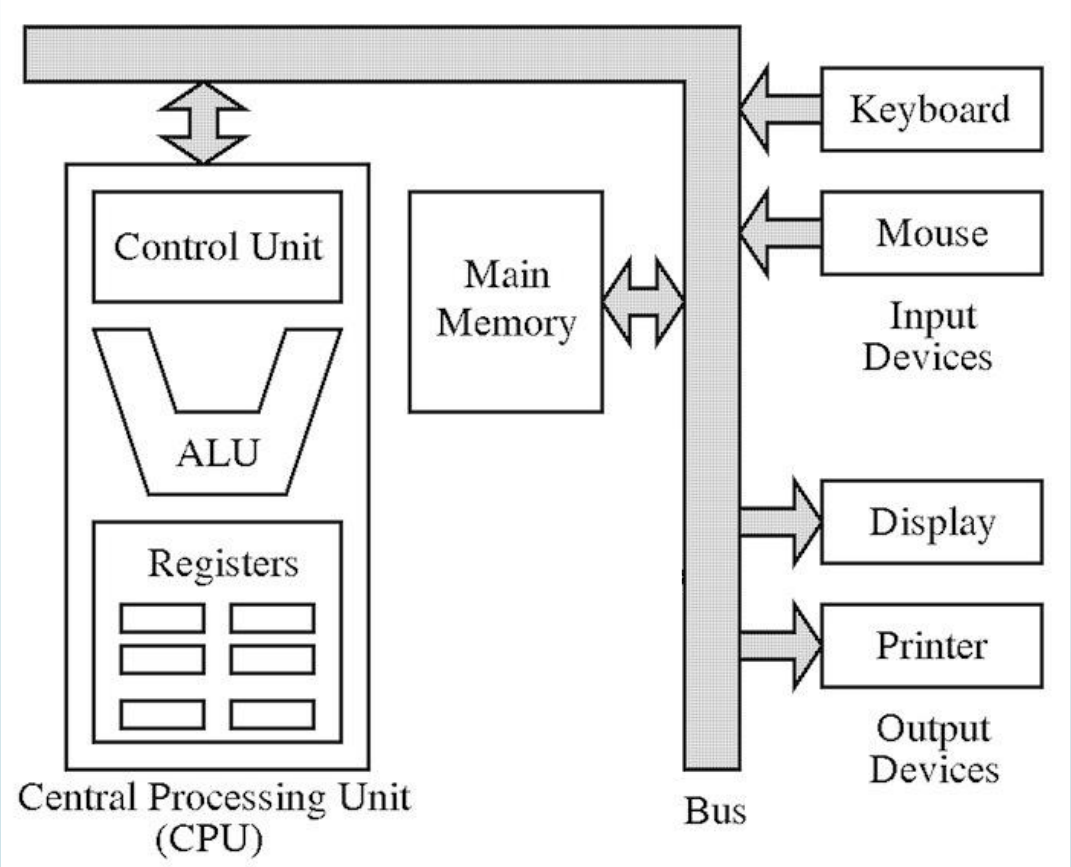
Layers architecture



Ax and Bx are two registers (small memories)

Sub, Add, Mul, … are mnemonics (functions)

Von Neuman machine



CPU: processor

Arithmetic and Logic Unit (ALU): permforms computation (+,-,*,/,…)

Register: a small memory that usually hols 32 or 64 bits (32-bit or 64-bit CPU).

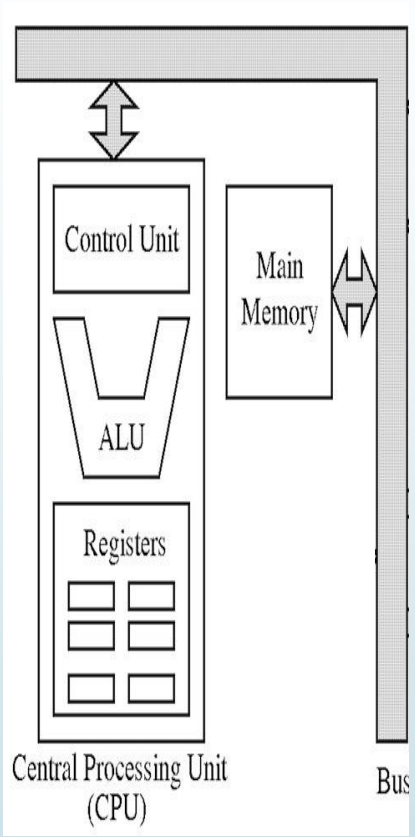
Von Neuman machine

Three types of bus exist:

- Control bus (unidirectional)

- Address bus (unidirectional)

- Data bus (bidirectional)



The control unit sends orders (read/write) via the control bus

The control unit sends addresses via the address bus

CPU and Memory exchange data (information) via the data bus

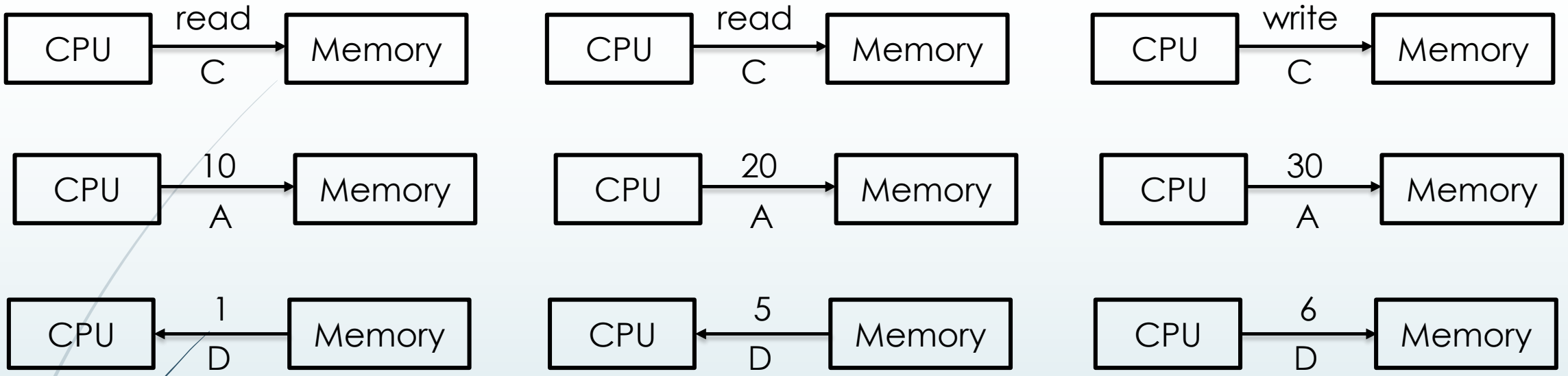
Exercise

Consider the following machine:

- 2 control signals
- A RAM of 64 KB divided into cells of 2 bytes each

Q1: Determine the width of control and address buses.

Q2: Expalin how the machine executes z=x+y using the control, address and data buses, where x, y and z are located in adresses 10, 20, 30 respectively and have values 1, 5, 2 respectively.


CPU
read
C
Memory
CPU
read
C
Memory
CPU
write
C
Memory
CPU
10
A
Memory
CPU
20
A
Memory
CPU
30
A
Memory
CPU
1
D
Memory
CPU
5
D
Memory
CPU
6
D
Memory