Cambridge IGCSE

Computer Science

Section 3

Hardware

3.1 Computer Architecture (i)

GS – 11/2021

Objectives

- Understand the role of the central processing unit (CPU) in a computer
- Understand what is meant by a microprocessor
- Understand the purpose of the components in a CPU, in a computer that has a Von Neumann architecture system
- Describe the process of the fetch-decode-execute cycle including the role of each component in the process

Vocabulary

- **processor**
- **cpu**
- **components**
- **control unit**
- **arithmetic and logic unit**
- **register**
- **bus**
- **microprocessor**
- **fetch**
- **decode**
- **execute**
- **integrated circuit**

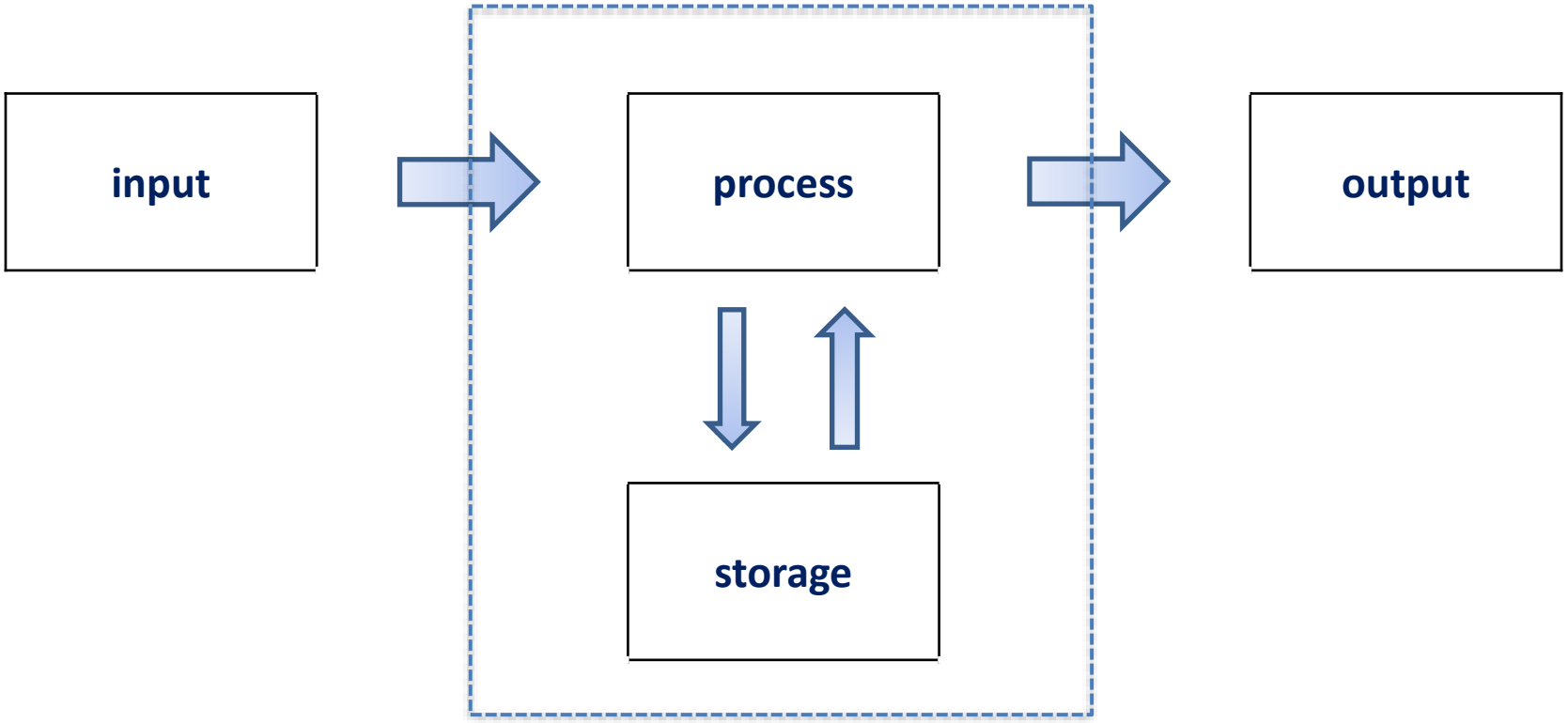
The main components of a computer

Watch:

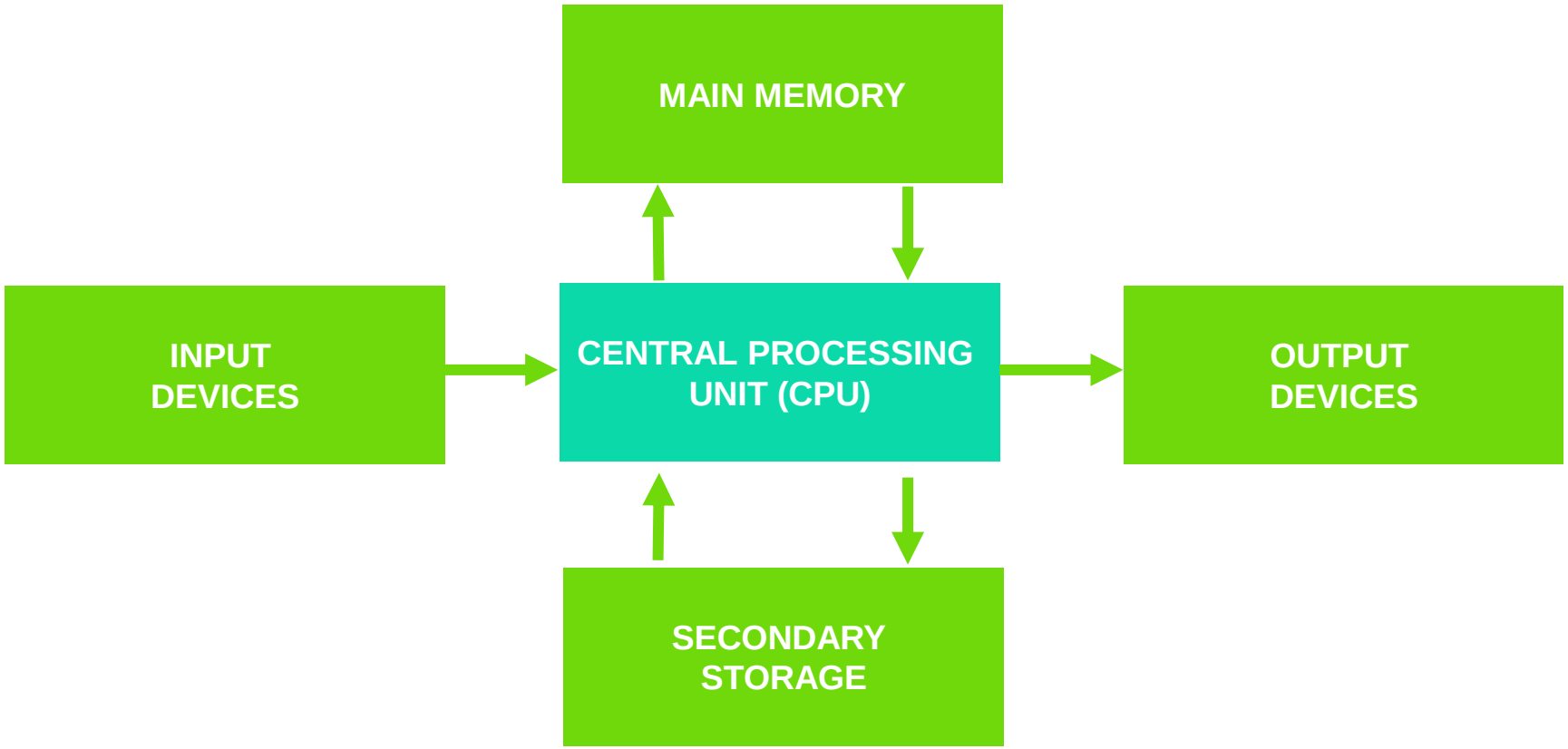
Video - What makes a computer

Video - Computer parts explained

A computer system

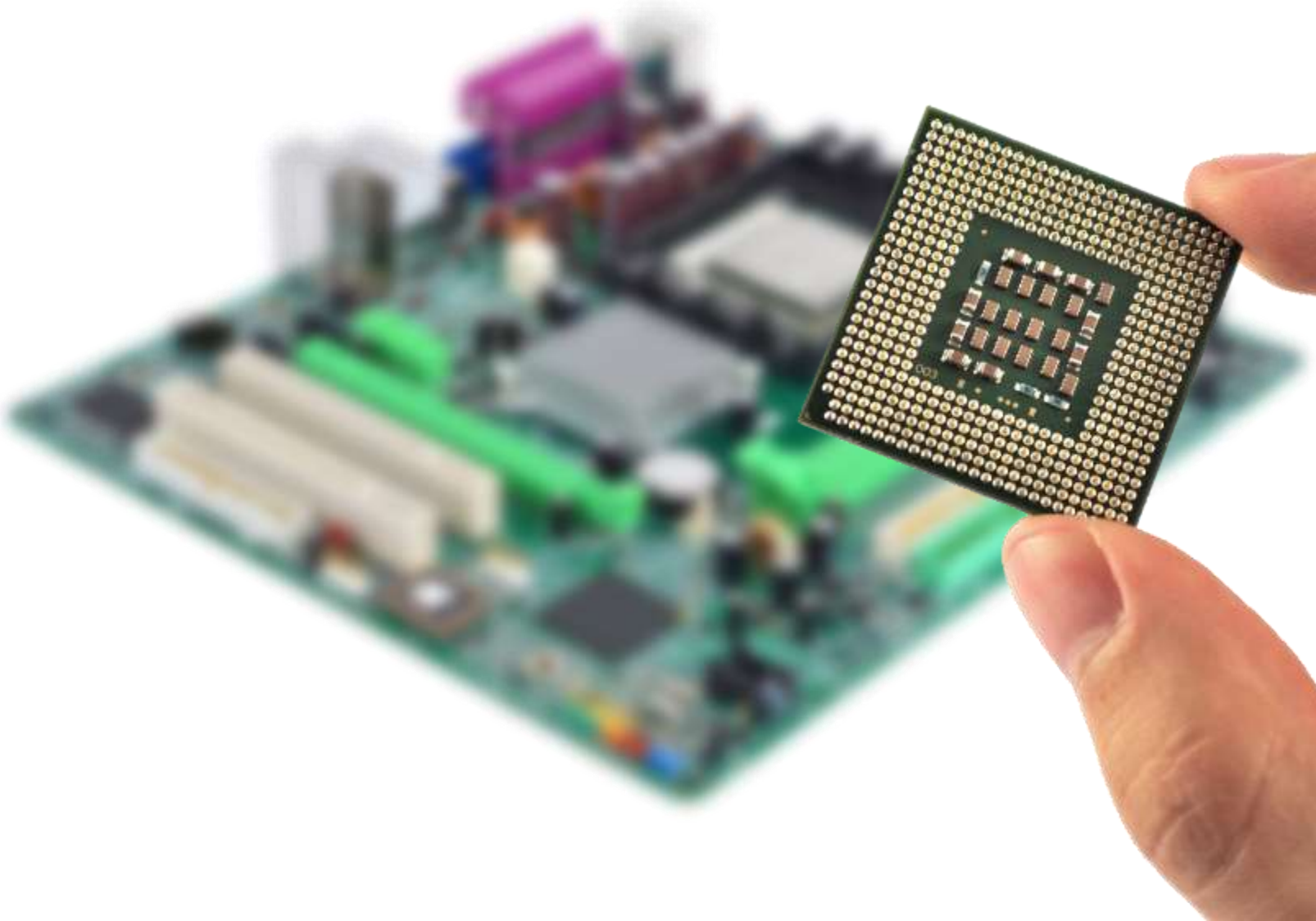


The main components of a computer



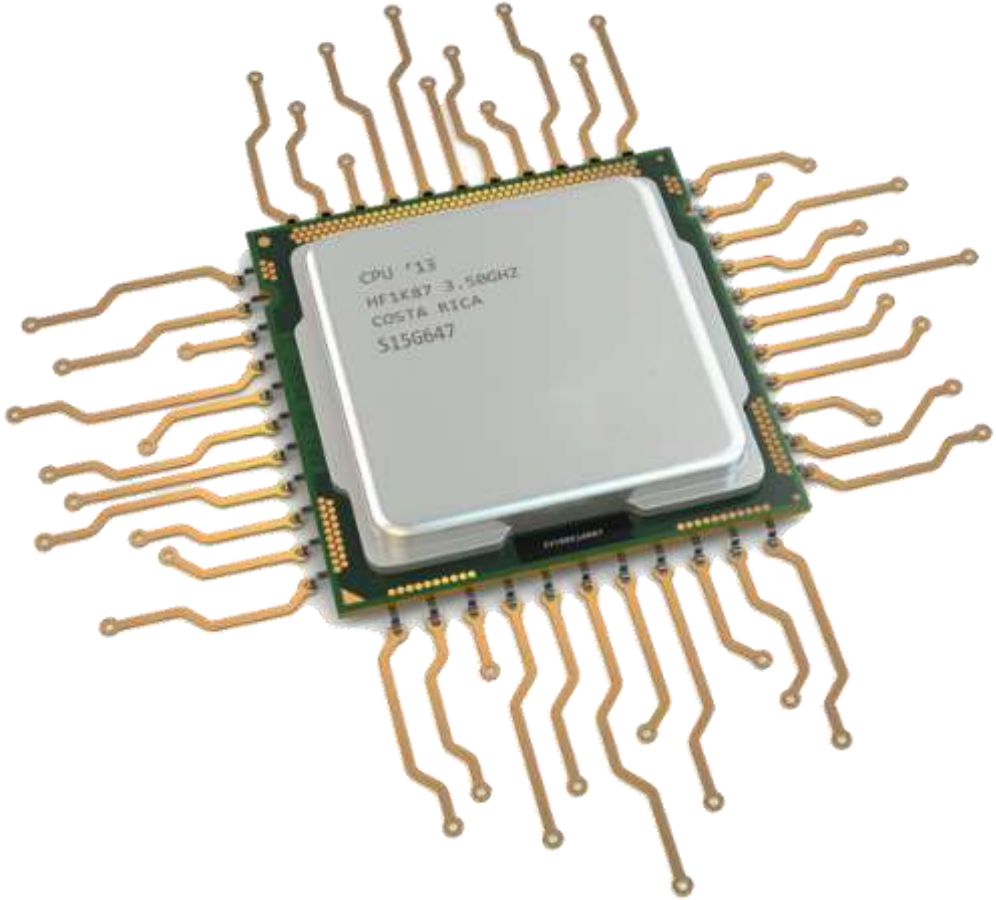
What is this?

The CPU



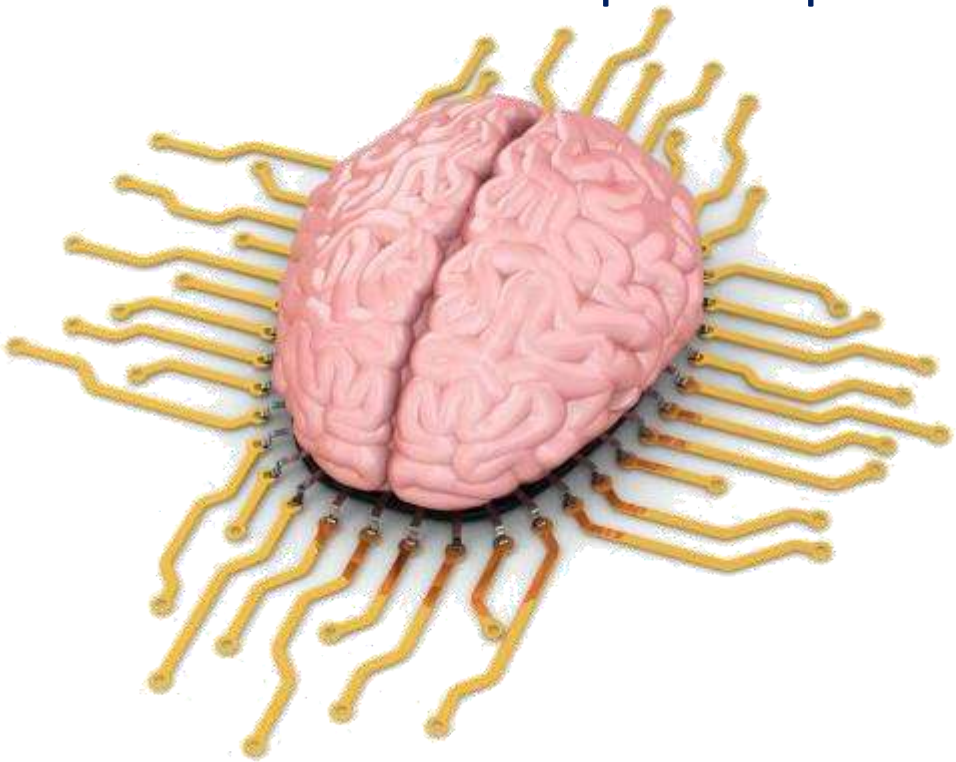
Central Processing Unit (CPU)

- The Central Processing Unit or CPU is arguably the most important component of a computer
  - What does it do?
  - If it was one organ in the human body what would it be?



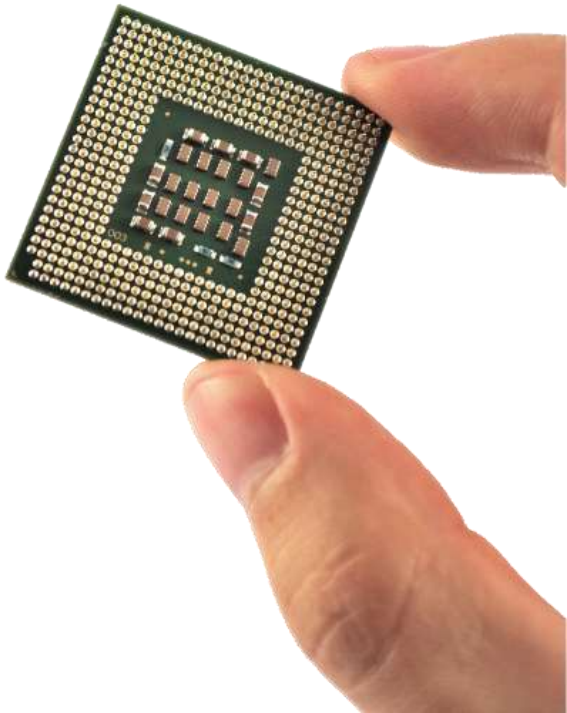
Central Processing Unit (CPU)

- You can think of the CPU is being like the brain in a human
- It is responsible for all of a computer's processing



Central Processing Unit (CPU)

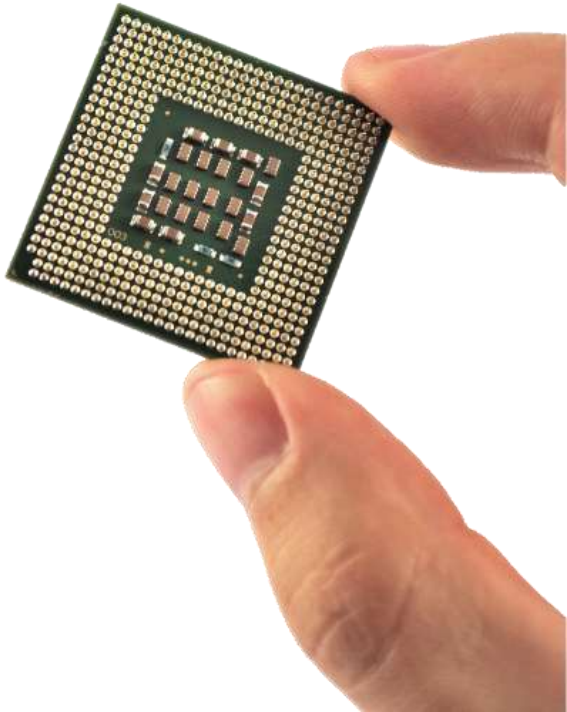
- Key to all modern computer systems (including tablets, smartphones etc.)
- Responsible for the **processing of all the instructions** and data in a computer application.
- A **4GHz** processor processes **4,000,000,000 instructions per second**!
- The CPU consists of:
  - control unit
  - arithmetic and logic unit
  - registers and buses



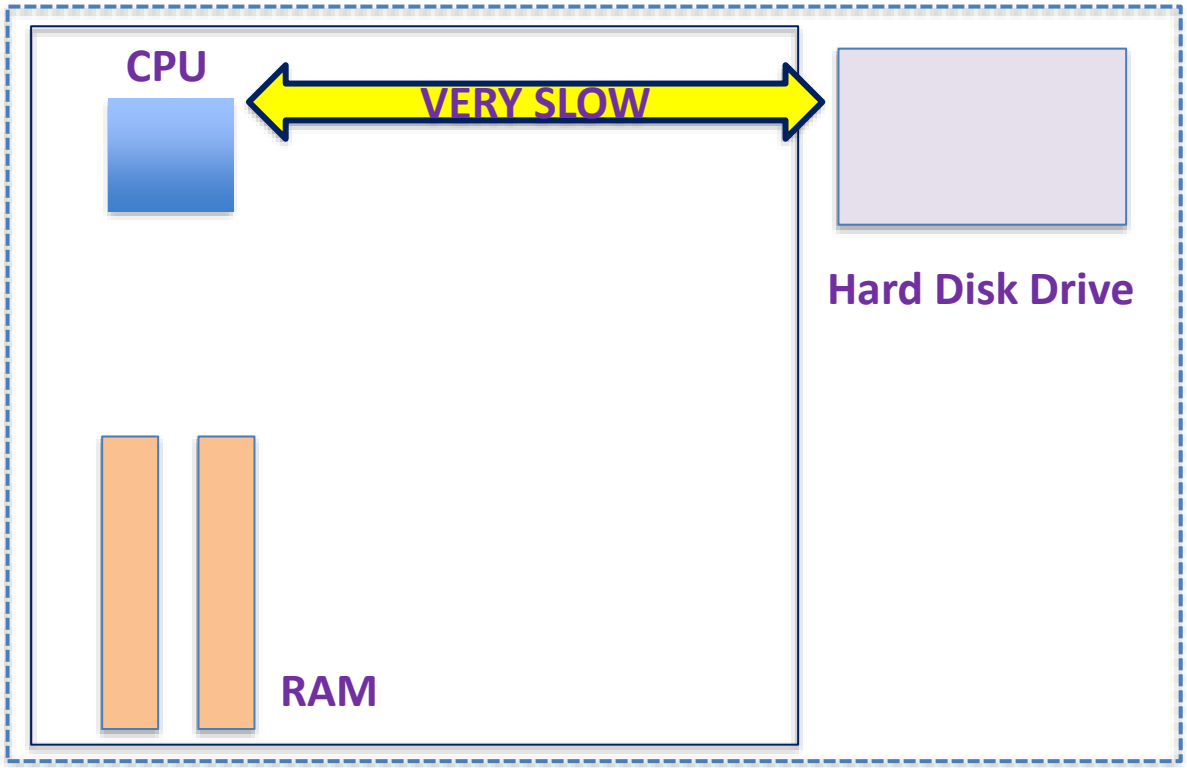
Central Processing Unit (CPU)

- A **microprocessor**

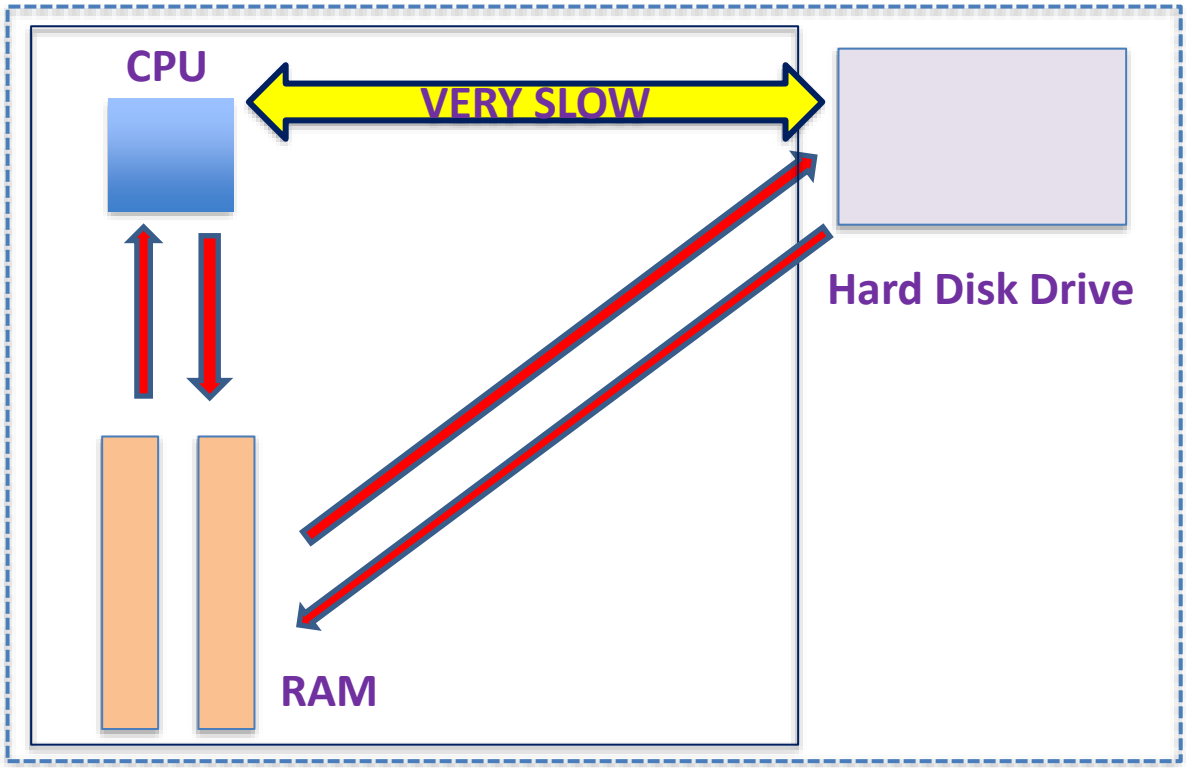
  is a type of integrated circuit on a single chip



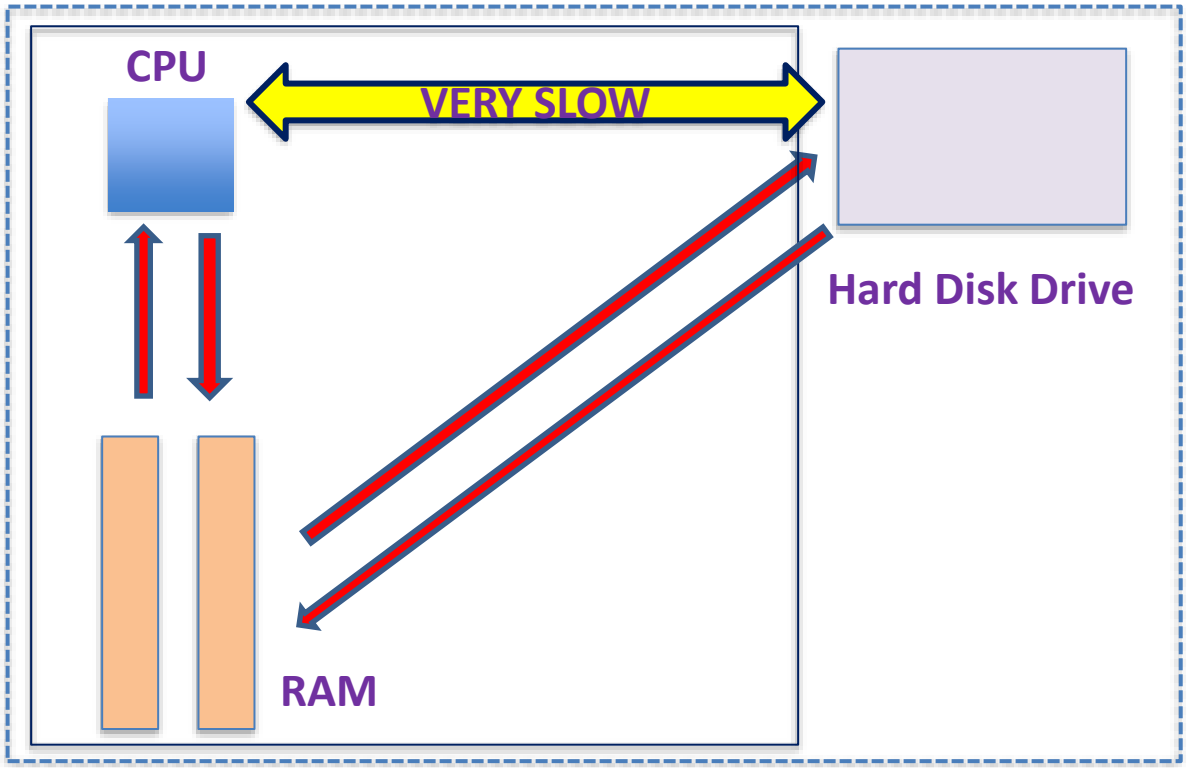
The main components of a computer



The main components of a computer



The main components of a computer



non-volatile

volatile

clipboard

The main components of a computer

Watch:

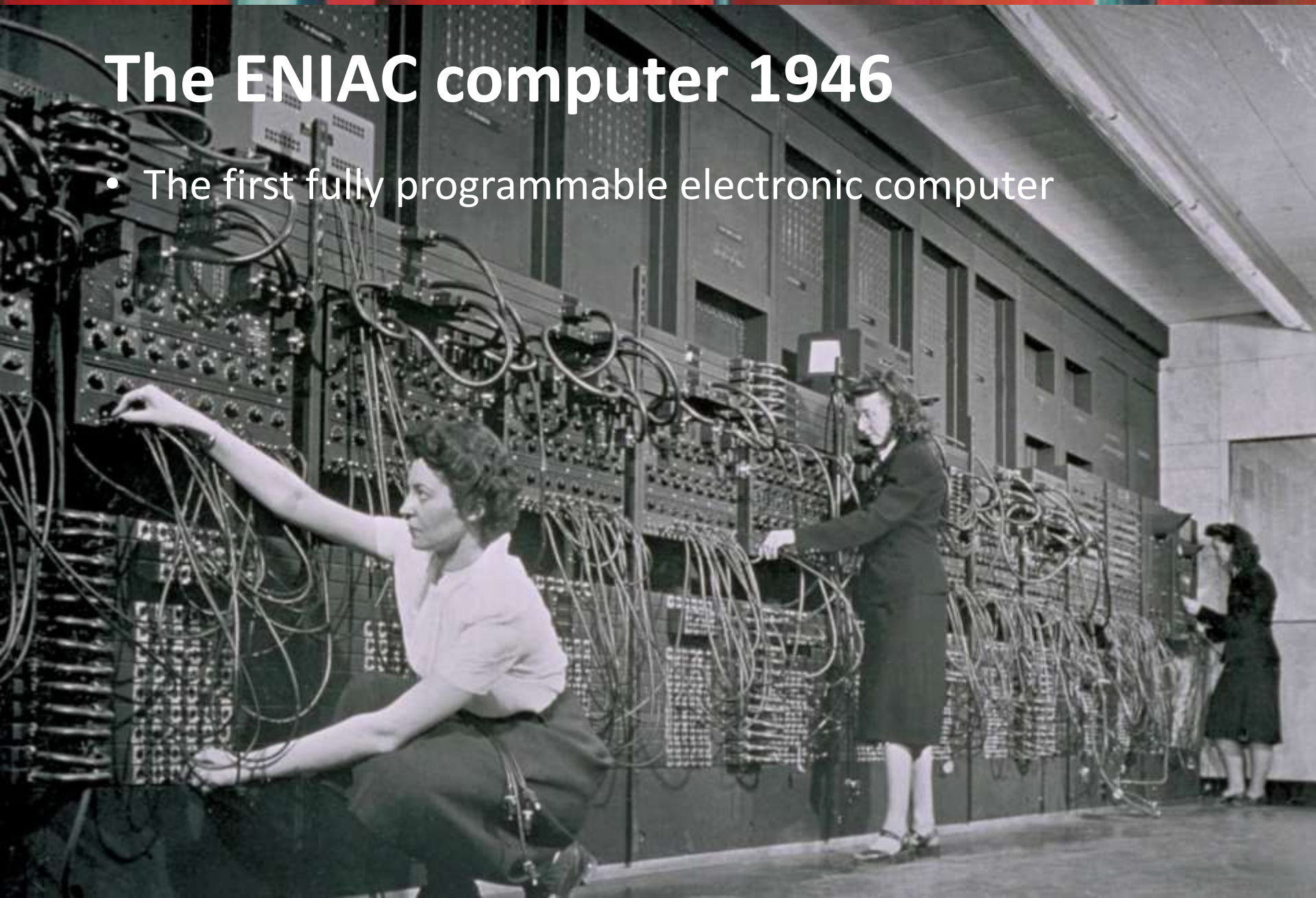
Video - Hardware and software

Stored program concept

- Before about 1943, early computers stored the data to be worked on in memory
- The program was not stored - instructions were input one at a time using switches, or read in from paper tape and executed one at a time
- In 1943-44, mathematician von Neumann and his colleagues had the idea of storing the program instructions as well as the data in the same store
- The **stored-program computer** was born!

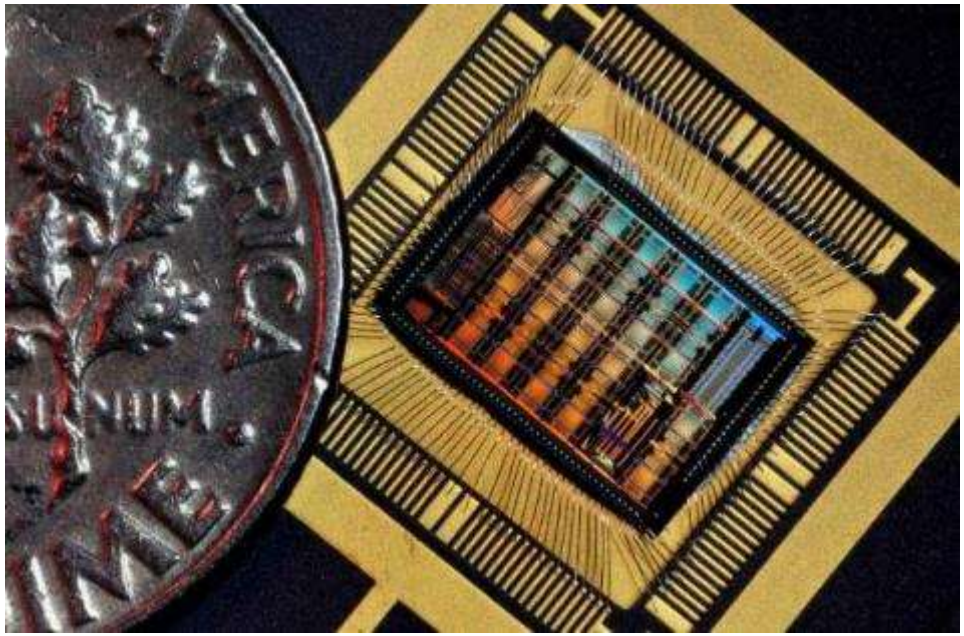
The ENIAC computer 1946

- The first fully programmable electronic computer



ENIAC

- The ENIAC computer was 8 feet high, 80 feet long and weighed 30 tons
  - In 1996, it was rebuilt on a 0.5cm^2 chip



Von Neumann architecture

- Computer **programs** and the **data** they are using are stored in the same memory.
  Data then moves between the memory and processor.

- The stored program concept :
  program instructions and data are stored in main memory.
  Instructions are fetched and executed one after another.

Von Neumann architecture

- Computer **programs** and the **data** they are using are stored in the same memory
  - Data then moves between the memory and processor

Memory address	Memory content
1	`Program`
2	`Program`
3	`Program`
4	
5	`Data`
6	`Data`
7	`Data`
8	`Data`

Von Neumann model for a computer system

- The stored program concept :
  program instructions and data are stored in main memory. Instructions are fetched and executed one after another.

Von Neumann model for a computer system

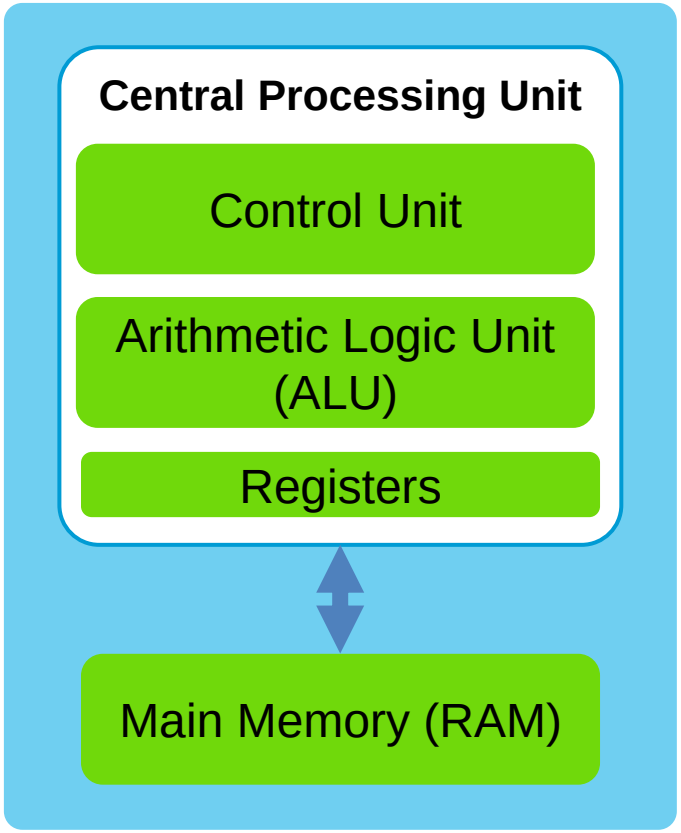
Watch:

Video – How a CPU works

(first part - data in memory and inside the CPU)

Main components of the CPU

- The CPU has two major components:
  - **The Control Unit**
  - **The ALU**
- In addition, it contains a few special, and very fast memory locations called **registers**

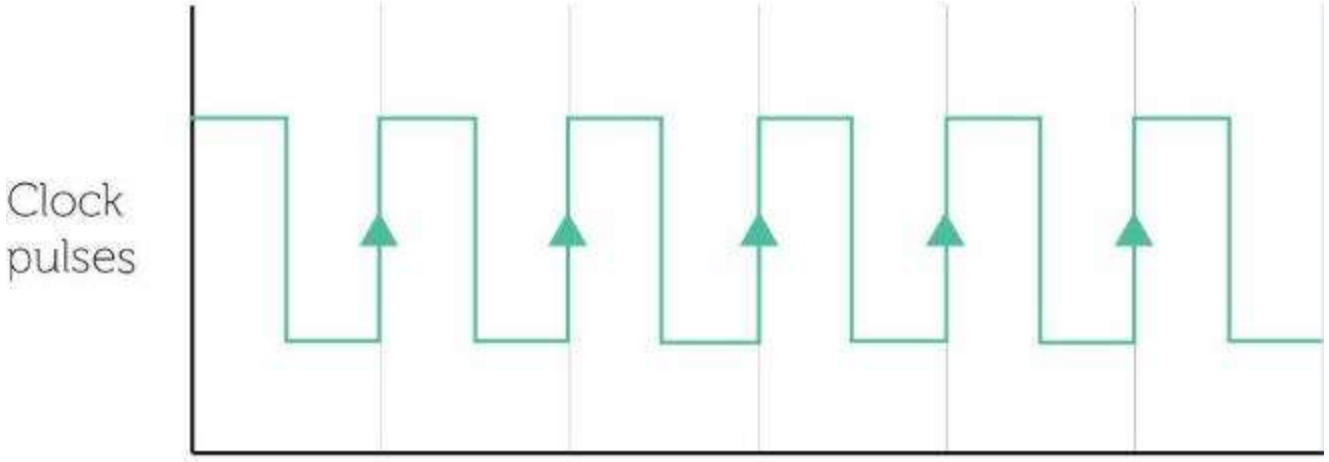


The Control Unit

- Like the conductor of an orchestra, the **Control Unit** co-ordinates, controls and regulates the operation of the computer

The Control Unit

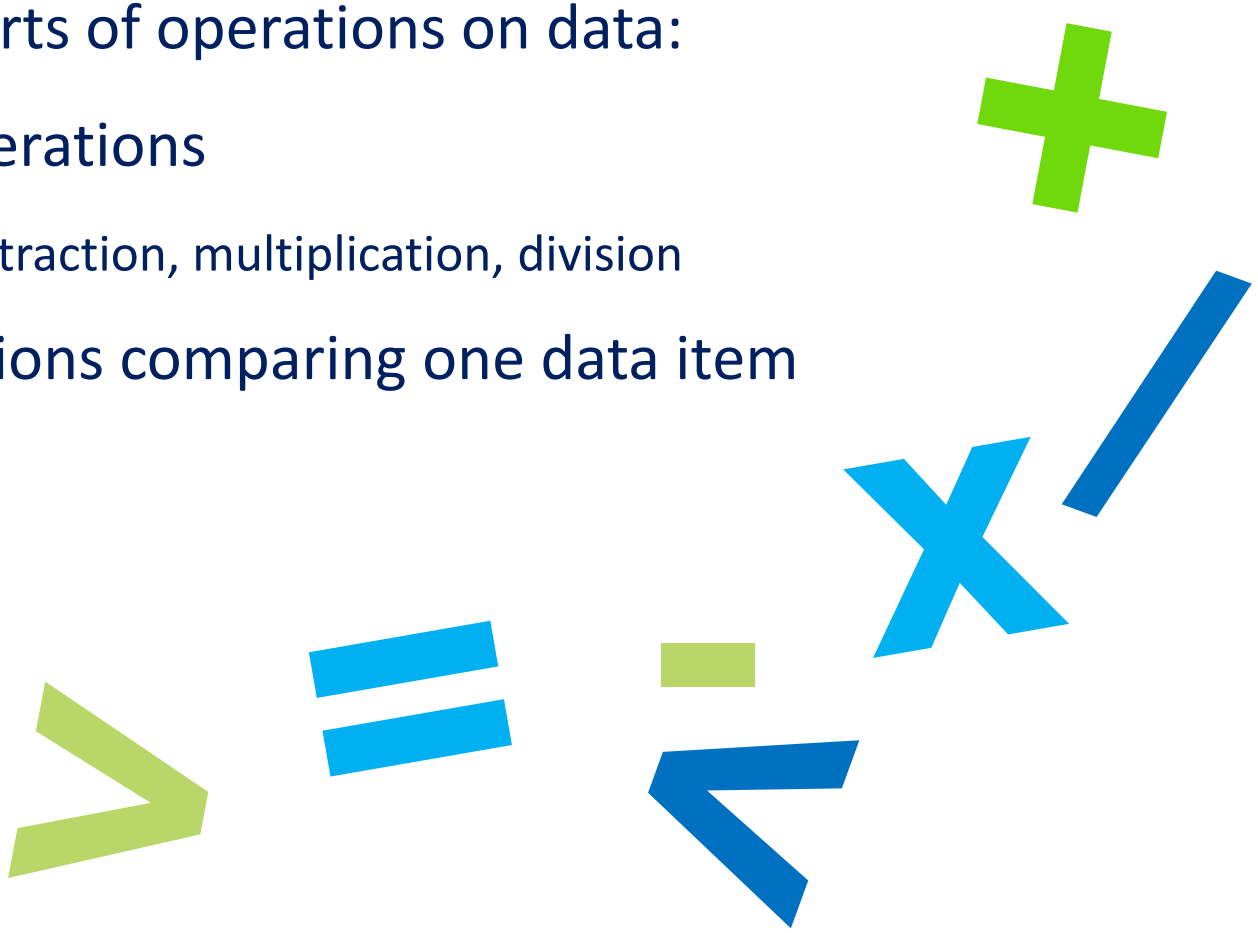
- It **fetches the instructions and data from memory**
- Provides **clock and control signals**



Arithmetic Logic Unit (ALU)

Performs two sorts of operations on data:

- Arithmetic operations
  - Addition, subtraction, multiplication, division
- Logical operations comparing one data item to another:
  - Is A > B?
  - Is X = Y?



Registers in the CPU

Very fast memory locations in the CPU, used to store data temporarily when executing instructions.

Simple component of CPU that stores data temporarily, just like RAM, but inside CPU, so is faster to access than RAM.

Stores data temporarily while instruction is being processed.

Registers in the CPU

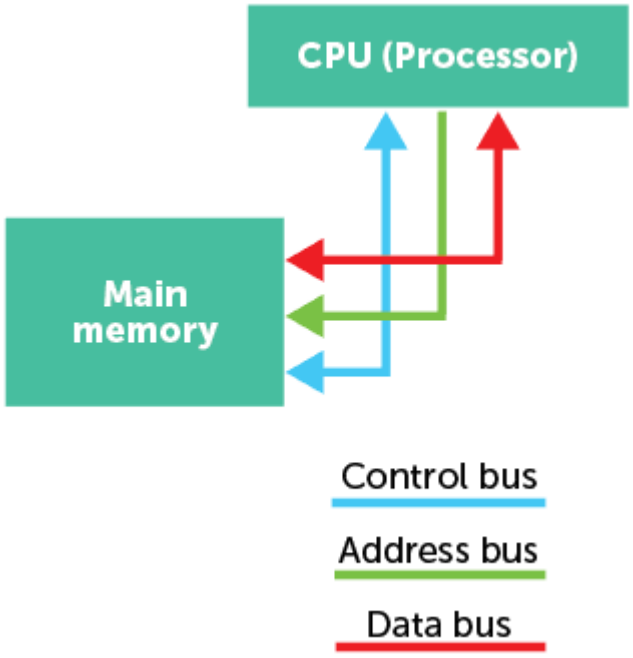
- **Program Counter (PC)**
  holds the address of the next instruction to be executed
- **Memory Address Register (MAR)**
  holds the memory address of the current instruction, and then the data that it uses, so that these can be fetched from memory
- **Memory Data Register (MDR)**
  holds the actual instruction, and then the data that has been fetched from memory
- **Accumulator/ General purpose registers**
  hold(s) data or the result of an instruction before it is transferred to memory
- **Current Instruction Register (CIR)**
  holds the instruction that is being processed

Buses

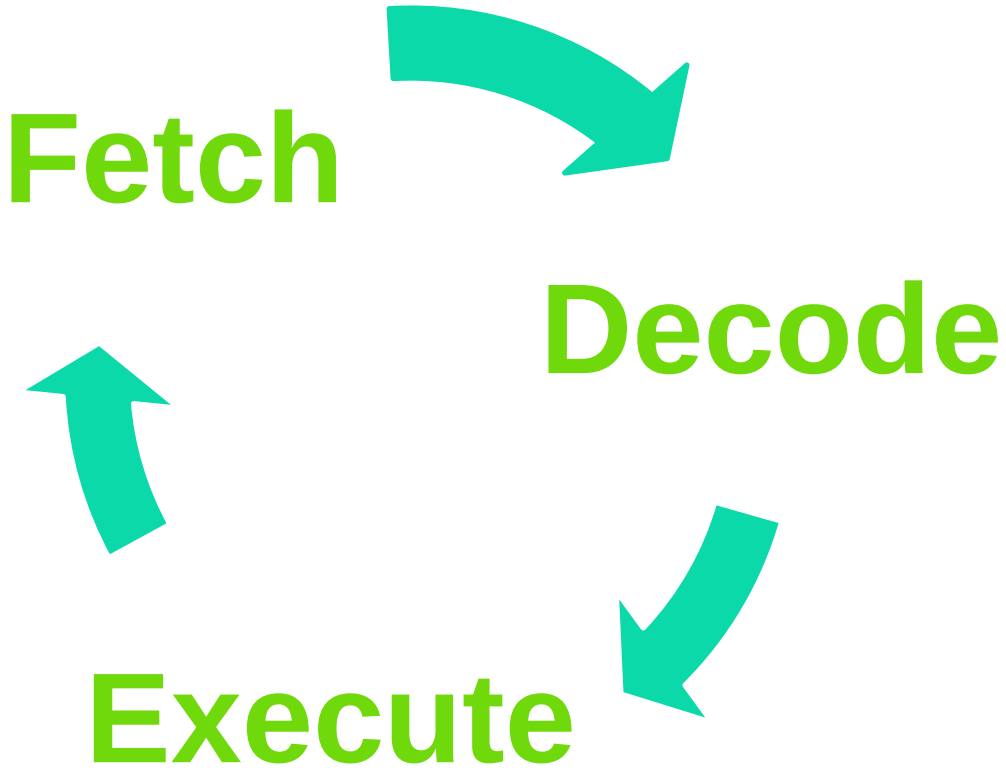
- **Control** bus:
  - Carries control signals, controlling the operation of the computer
- **Data** bus:
  - Transports the data around the computer
- **Address** bus:
  - The CPU uses this bus to send the address of where to get data from and where to store it
- These three buses make up the **system bus**

Buses

- A bus is a set of parallel wires connecting two or more components of the computer
  - When the CPU wishes to access a particular main memory location, it sends this address to memory on the address bus
  - The data in that location is then returned to the CPU on the data bus
  - Control signals are set along the control bus

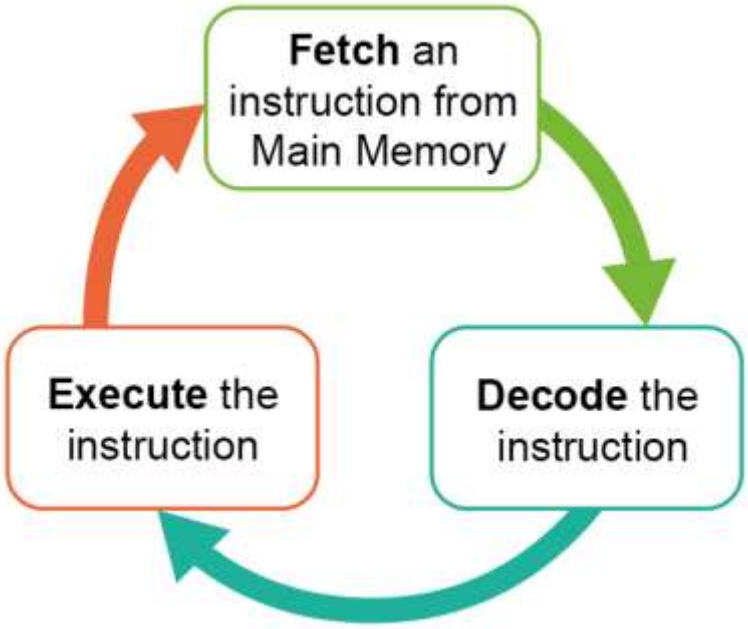


The Fetch - Decode - Execute cycle



The Fetch - Decode - Execute cycle

- The CPU operates by repeating three operations:
  - **FETCH** - get the next instruction and data from memory
  - **DECODE** - understand the instruction
  - **EXECUTE** - carry out the instruction
  - Repeat …



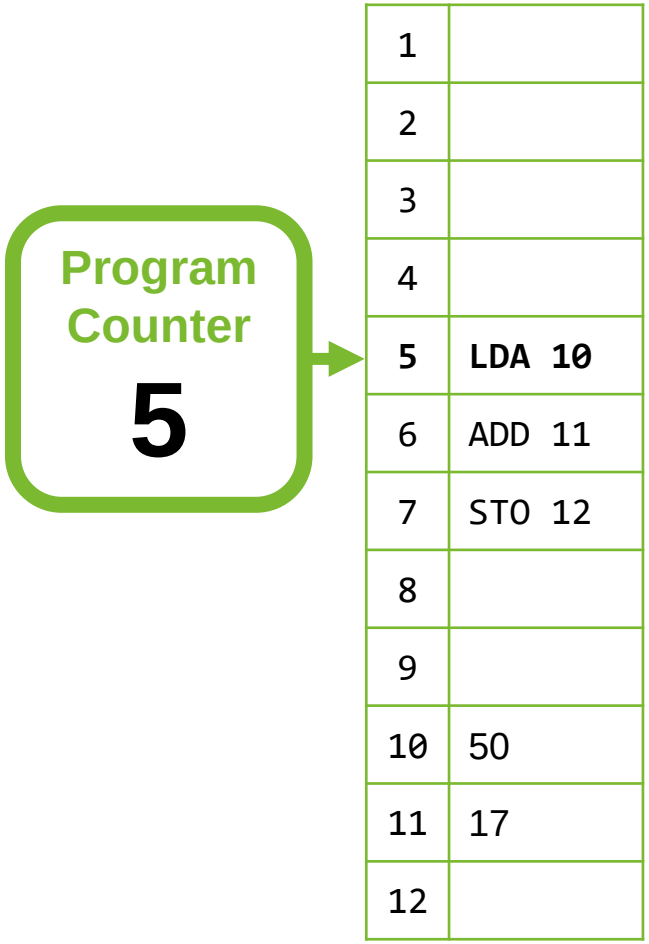
The Fetch - Decode - Execute cycle

Watch:

Video – Fetch-Execute cycle - what your computer is doing

Program Counter

- The Program Counter holds the address of the next instruction to be executed
  - It is located in the Control Unit
- The Program Counter is incremented as soon as that instruction has been fetched



Accumulator

- The accumulator is where arithmetic and logic results are temporarily stored, much like the **M+** function on a calculator
- Different processors have between 1 and 32 general purpose registers
- For simplicity we will refer to the **accumulator** instead of “a general purpose register”

MAR and MDR

- In the FETCH stage of the F-E cycle, the address of the **instruction** to be executed is copied from the PC to the Memory Address Register (MAR)
- The **instruction** at that address is fetched and copied to the Memory Data Register (MDR)
- The Control Unit decodes the instruction and decides if data needs to be fetched
- If so, the MAR is then used to hold the address of the **data** to be used in the instruction
- The **data** is fetched and copied to the MDR

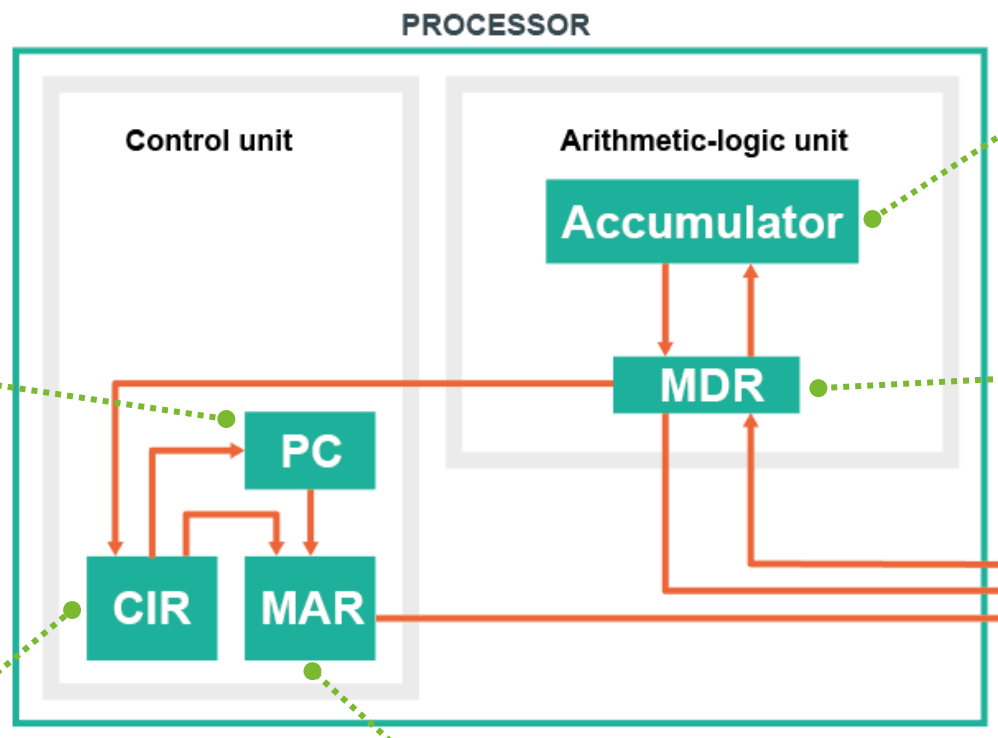
MAR and MDR

MAR (Memory Address Register)

MDR (Memory Data Register)

The two work together; The MAR knows where to look for data in RAM, the MDR keeps hold of that data until it's ready to be used by the CPU

Processor registers



Accumulator (or general purpose register): Temporarily stores arithmetic and logic results

PC: Points to the next instruction

MDR: Stores the actual instruction or data

CIR (Current Instruction Register): Holds the current instruction to be executed

MAR: Stores the address of the instruction/data

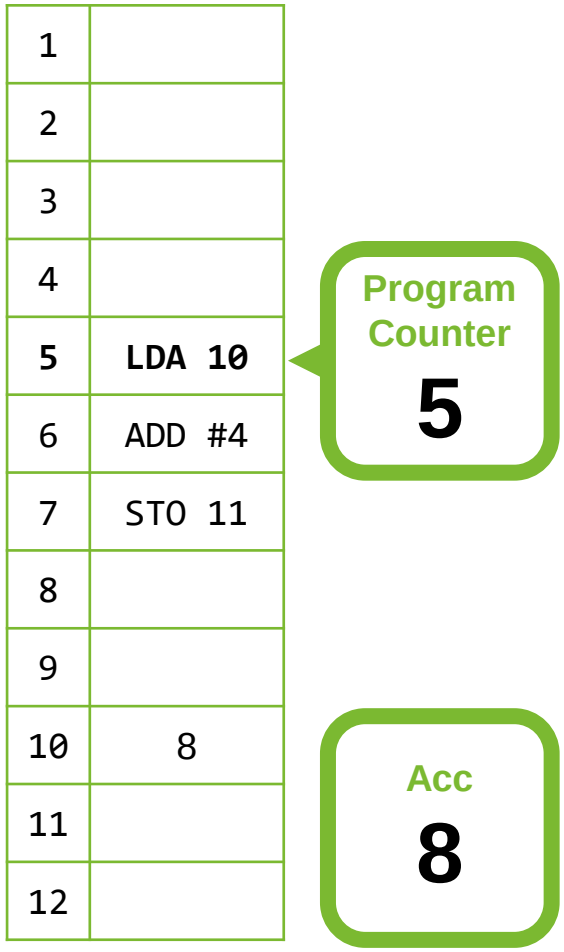
Example

1	
2	
3	
4	
5	**LDA 10**
6	ADD #4
7	STO 11
8	
9	
10	8
11	
12	

Program Counter
5

Acc

Example - Step 1 (Add 8 + 4)



- PC points to next instruction in location 5
- Address 5 is passed to MAR and PC is incremented. It now holds 6
- Instruction at Address 5 is copied from memory to MDR, then to Current Instruction Register (CIR)
- Address 10 passed from CIR to MAR
- The value in location 10, i.e. 8, is passed from main memory to the MDR
- 8 is loaded into the accumulator

Example - Step 2

1	
2	
3	
4	
5	LDA 10
6	**ADD #4**
7	STO 11
8	
9	
10	8
11	
12	

Program Counter
6

Acc
12

- PC now points to next instruction in location 6
- Instruction is passed to the MAR and the PC is incremented
- Instruction at Address 6 passed to MDR, then to Current Instruction Register (CIR)
- No more data from memory is needed, so instruction is decoded
- 4 is added to 8 in the ALU and the result is stored in the accumulator

Example - Step 3

1	
2	
3	
4	
5	LDA 10
6	ADD #4
7	**STO 11**
8	
9	
10	8
11	12
12	

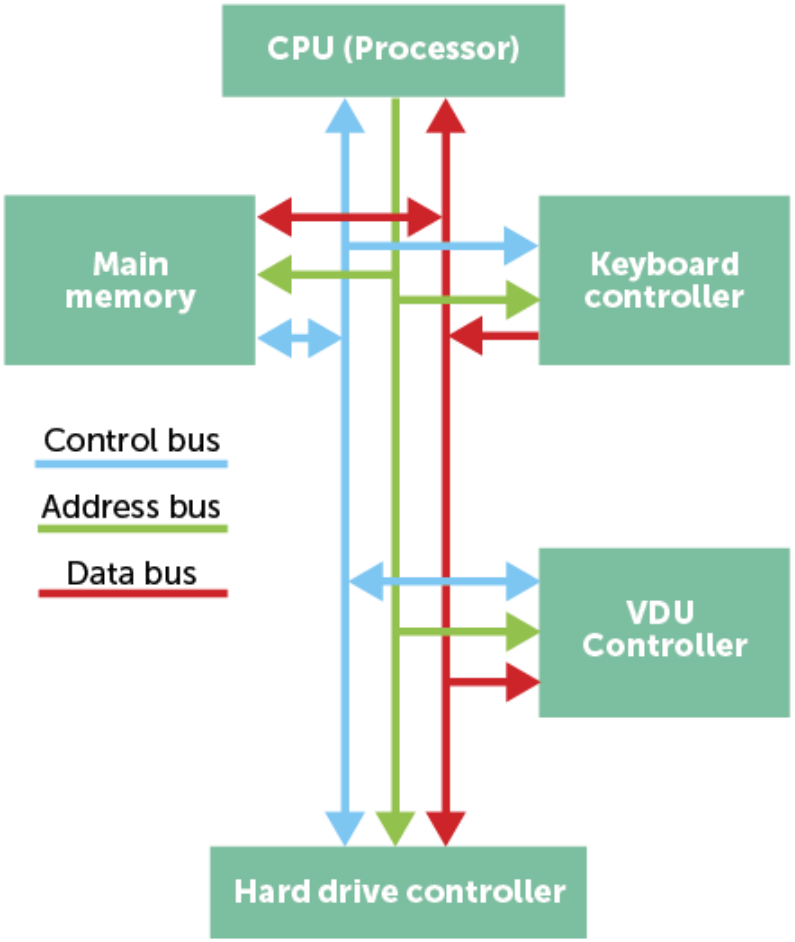
Program Counter
7

Acc
12

- PC points to the next instruction in location 7
- Instruction is passed to the MAR and the PC is incremented
- Instruction at Address 7 passed to MDR, then to Current Instruction Register (CIR)
- No more data from memory is needed, so instruction is decoded
- Total value 12 transferred from the accumulator into memory location 11

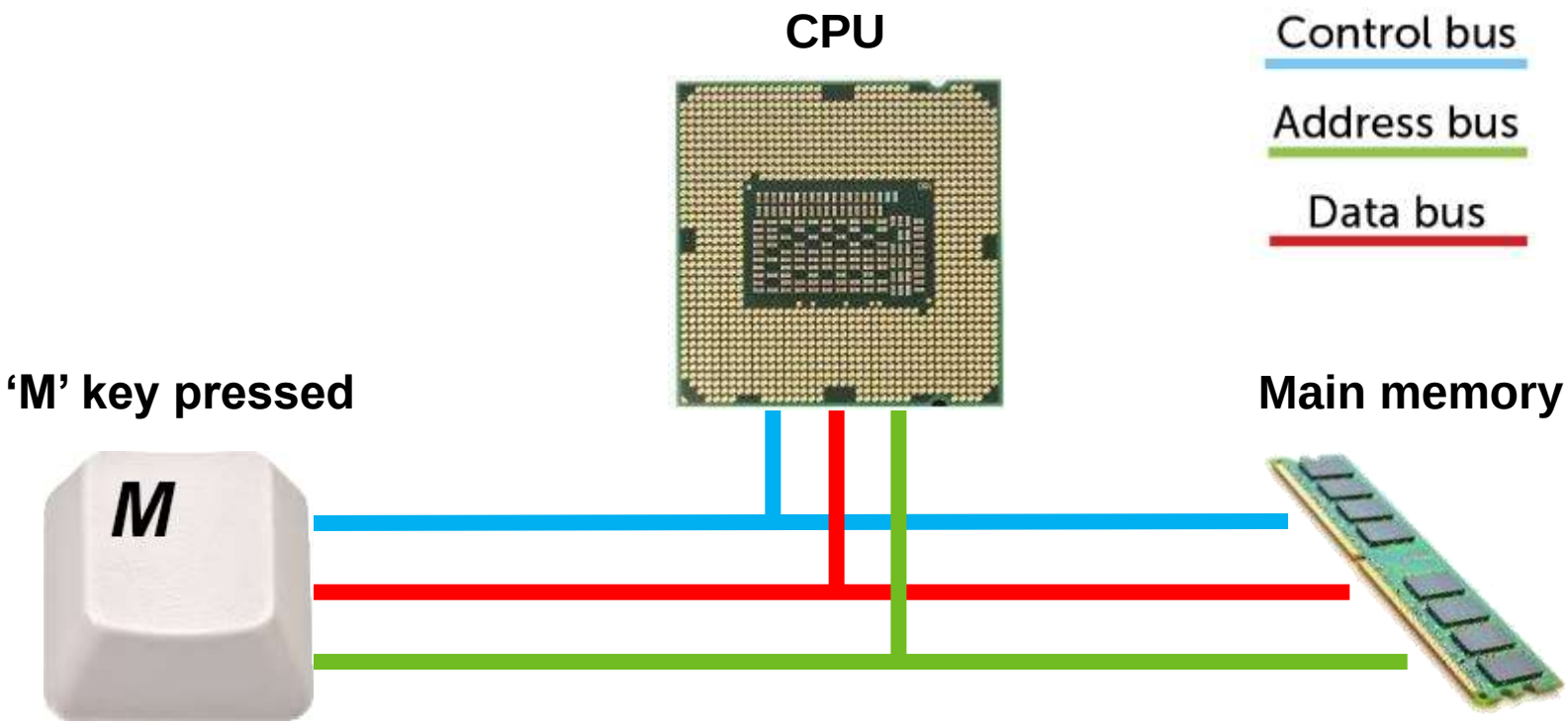
External buses

- The CPU is also connected to the external Input-output device controllers by means of buses
  - Only one device can transmit along a bus at any one time
  - The address bus is one way only, from the CPU
  - What about the data bus?



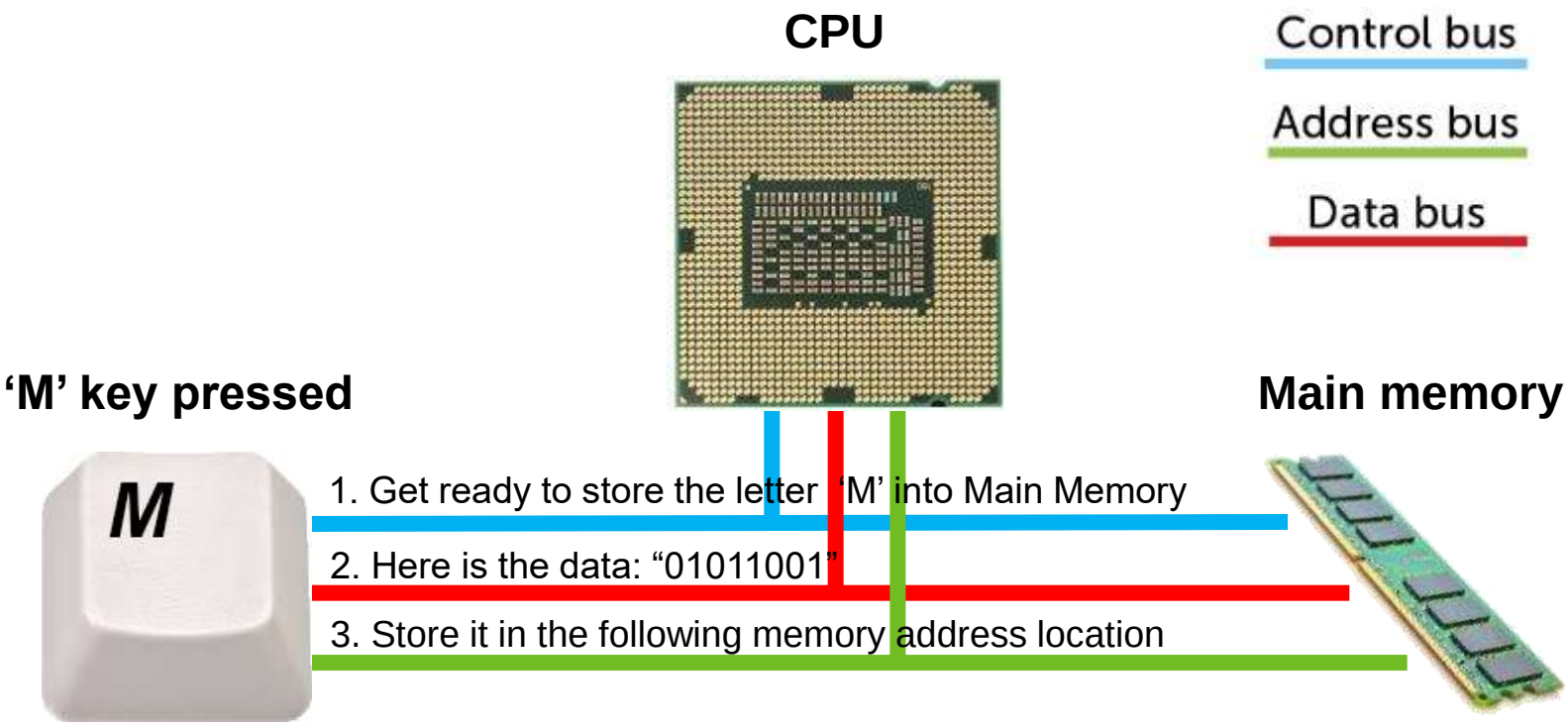
Buses - Summary

- Explain this diagram:



Buses - Summary

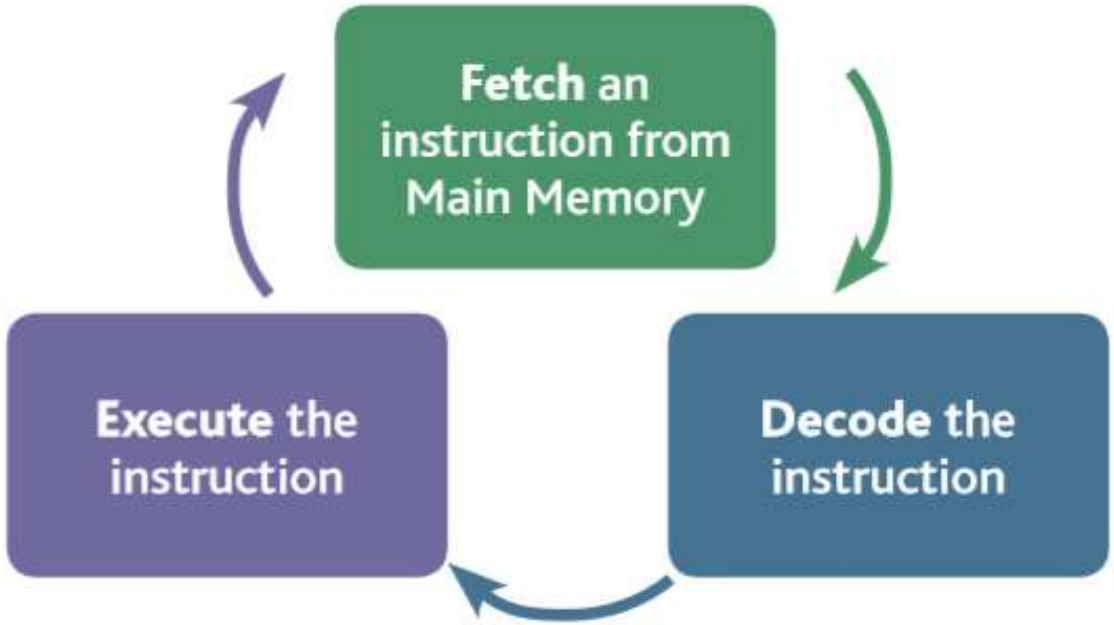
- Explain this diagram:



The Fetch - Decode - Execute cycle

Watch:

Video – Fetch decode execute cycle



Central Processing Unit

Control Unit

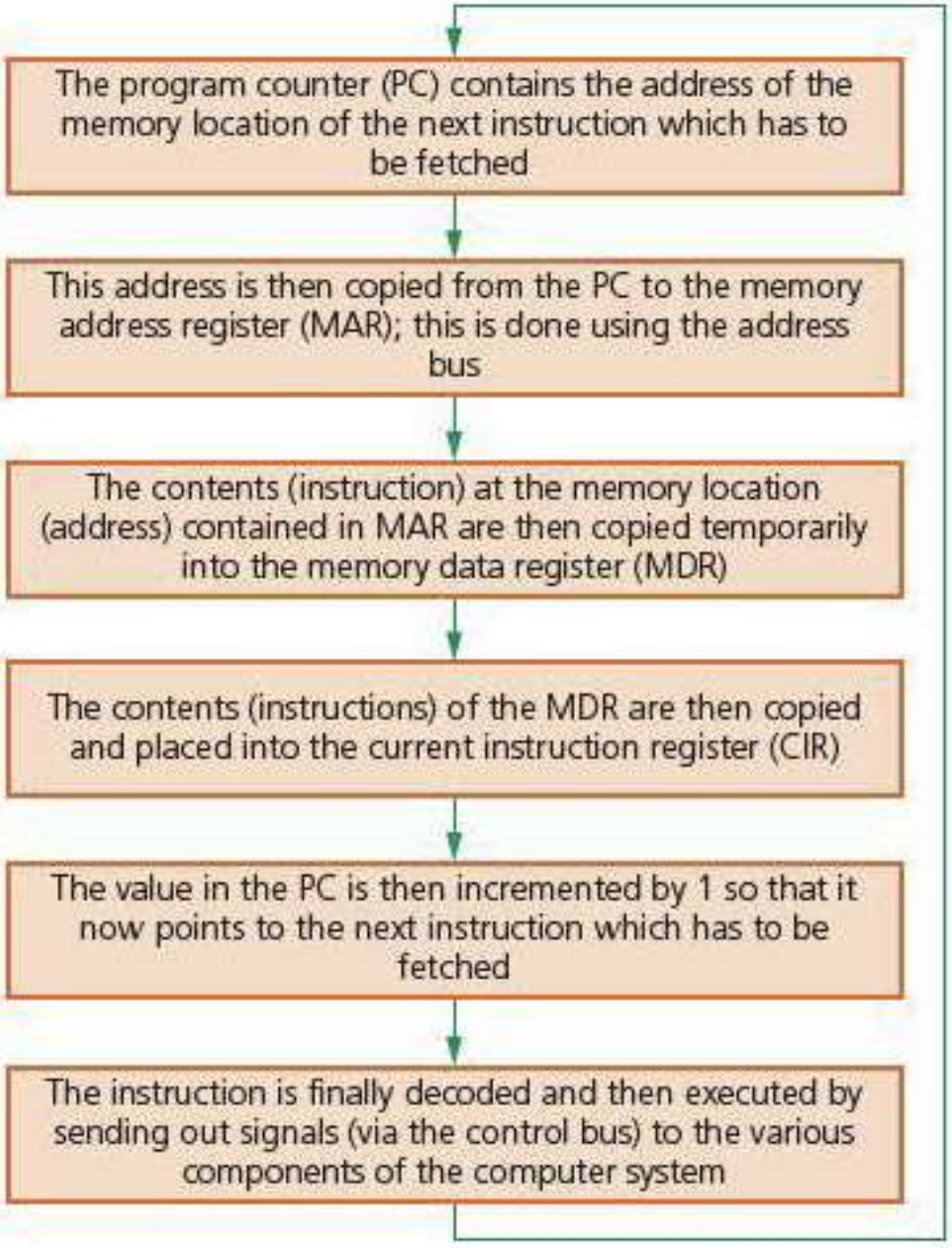
Arithmetic / Logic Unit

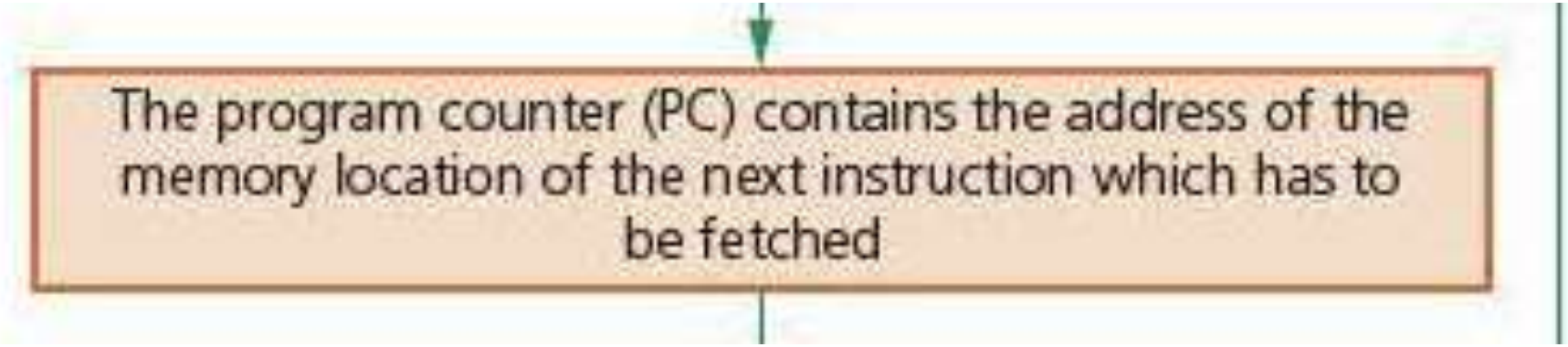
Registers

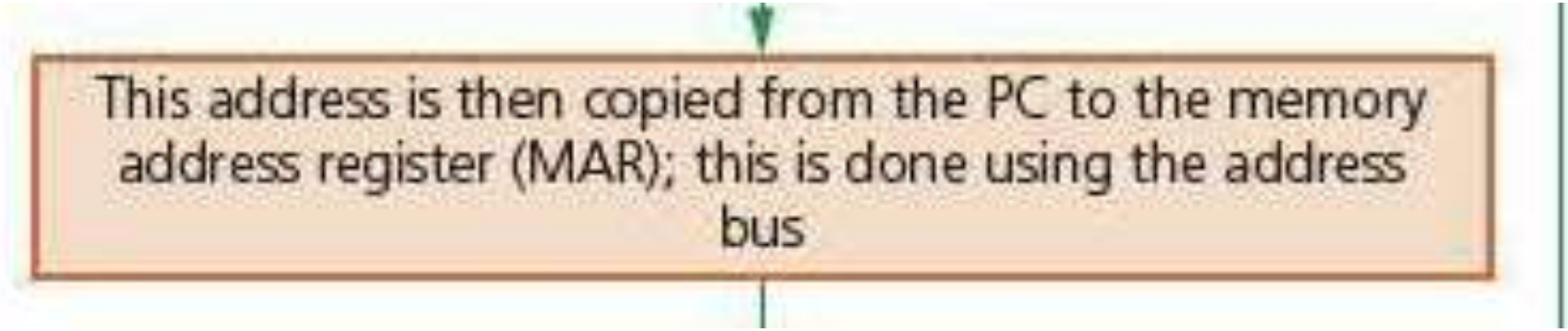
Program Counter	
Memory Address Register	
Memory Data Register	
Current Instruction Register	
Accumulator	

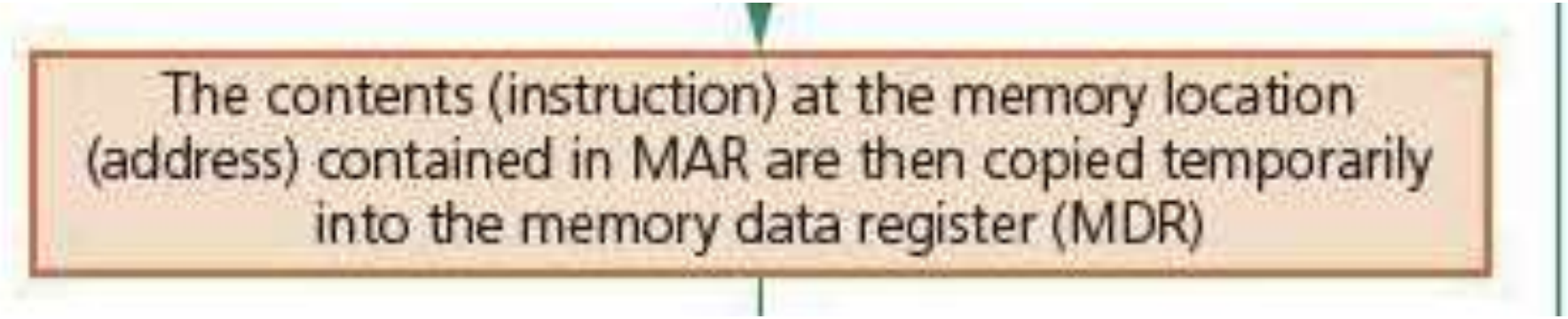
Memory

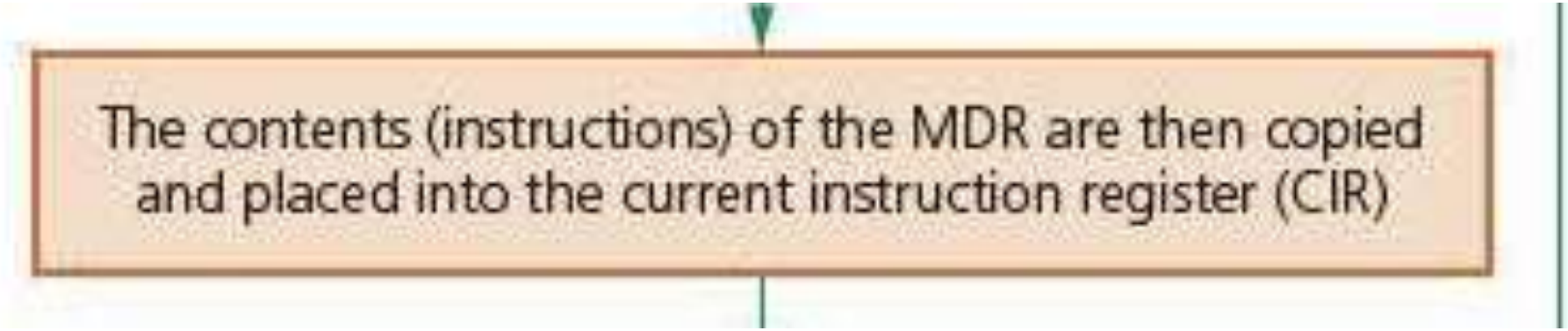
Address	Contents
0	LOAD 4
1	ADD 5
2	STORE 6
3	
4	300
5	600
6	

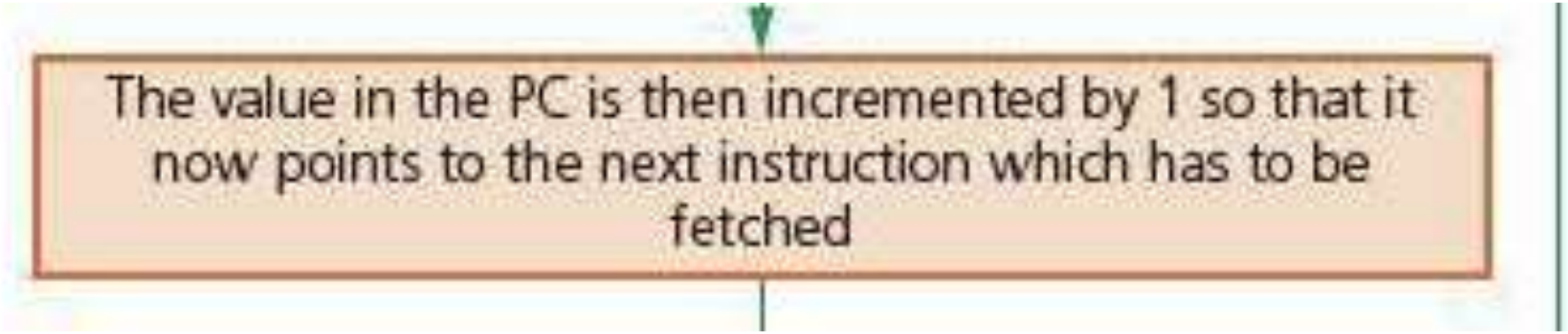

The program counter (PC) contains the address of the memory location of the next instruction which has to be fetched
This address is then copied from the PC to the memory address register (MAR); this is done using the address bus
The contents (instruction) at the memory location (address) contained in MAR are then copied temporarily into the memory data register (MDR)
The contents (instructions) of the MDR are then copied and placed into the current instruction register (CIR)
The value in the PC is then incremented by 1 so that it now points to the next instruction which has to be fetched
The instruction is finally decoded and then executed by sending out signals (via the control bus) to the various components of the computer system

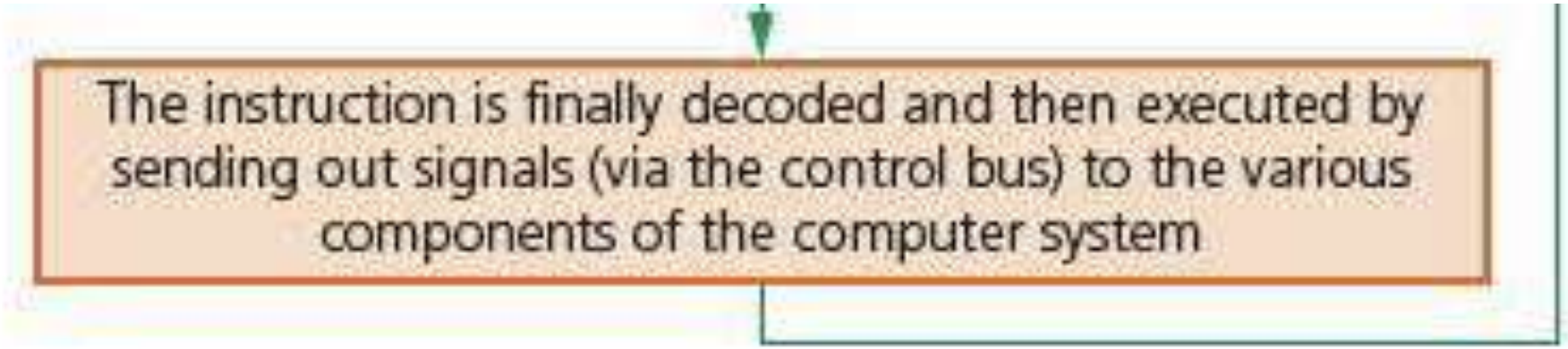

The program counter (PC) contains the address of the memory location of the next instruction which has to be fetched

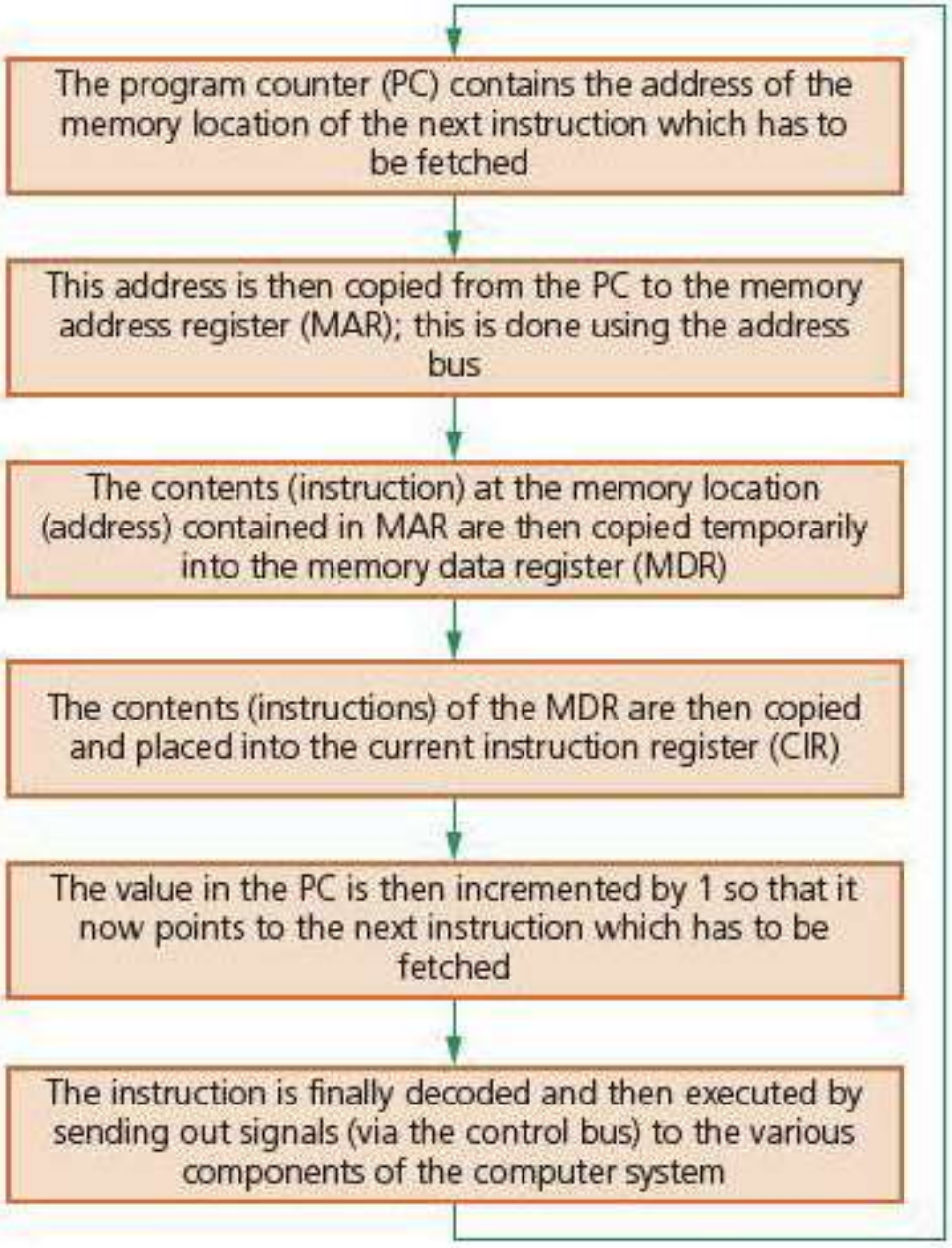

This address is then copied from the PC to the memory address register (MAR); this is done using the address bus


The contents (instruction) at the memory location (address) contained in MAR are then copied temporarily into the memory data register (MDR)


The contents (instructions) of the MDR are then copied and placed into the current instruction register (CIR)


The value in the PC is then incremented by 1 so that it now points to the next instruction which has to be fetched


The instruction is finally decoded and then executed by sending out signals (via the control bus) to the various components of the computer system


The program counter (PC) contains the address of the memory location of the next instruction which has to be fetched
This address is then copied from the PC to the memory address register (MAR); this is done using the address bus
The contents (instruction) at the memory location (address) contained in MAR are then copied temporarily into the memory data register (MDR)
The contents (instructions) of the MDR are then copied and placed into the current instruction register (CIR)
The value in the PC is then incremented by 1 so that it now points to the next instruction which has to be fetched
The instruction is finally decoded and then executed by sending out signals (via the control bus) to the various components of the computer system

Activity

- **Task 1, 2 and 3** on the worksheet

Summary

- **Central Processing Unit (CPU)**:
  - Controls all tasks inside a computer. Made up of the Control Unit, the Arithmetic and Logic Unit, Registers and Buses
  - **Control Unit**
    Co-ordinates and controls the operation of the computer. It fetches instructions and data from memory and provides a clock / timing signal.
  - **Arithmetic and Logic Unit**
    Performs arithmetic and logical operations on the data

Plenary

- **Control** bus:
  - Carries control signals, controlling the operation of the computer
- **Data** bus:
  - Transports the data around the computer
- **Address** bus:
  - The CPU uses this bus to send the address of where to get data from and where to store it
- These three buses make up the **system bus**

Plenary

- In a **von Neumann** machine, both instructions and data are stored in main memory
- The CPU operates by repeating three operations:
  - **Fetch, decode, execute**
- Fast memory locations called **registers** are used in these operations:
  - Program Counter (PC),
  - Memory Address Register (MAR)
  - Memory Data Register (MDR)
  - Current Instruction register (CIR)
  - Accumulator

Plenary

- **Register**
  Simple component of CPU that stores data temporarily, just like RAM, but inside CPU, so is faster to access than RAM.

  Stores data temporarily while instruction is being processed.

Homework

- Complete Homework sheet - Computer Architecture

Past paper question examples ...

1 The Von Neumann model for a computer system uses components, such as registers and buses, in the fetch-execute cycle.

(a) Draw a line to connect each component to its correct description.

Component	Description
Control Bus	Increments to point to the address of the next instruction to be fetched
Program Counter (PC)	Holds the result of a calculation. It is located within the Arithmetic Logic Unit (ALU)
Memory Data Register (MDR)	Carries signals to synchronise the fetch-execute cycle
Accumulator (ACC)	Temporary storage between the Central Processing Unit (CPU) and primary memory

[3]

(b) State **two** buses, other than the control bus, used in the Von Neumann model for a computer system.

1 .....

2 .....

[2]

Past paper question examples ...

(b) A computer uses the Von Neumann model and the stored program concept.

(i) Explain what is meant by the stored program concept.

.....

.....

.....

..... [2]

Past paper question examples ...

(ii) The Von Neumann model has several components that are used in the fetch-execute cycle.

One component is the Arithmetic Logic Unit (ALU).

Describe the role of the ALU.

.....

.....

.....

.....

.....

.....

.....

..... [4]

Past paper question examples ...

8 Explain how an instruction is fetched in a Von Neumann model computer.

.....

.....

.....

.....

.....

.....

.....

.....

..... [6]