Computer Structure

Learning Intentions

By the end of this lesson pupils at will be able to:

- ❑ Discuss the different models of processor
- ❑ Discuss the different types of memory
- ❑ Explain the use of cache memory
- ❑ Explain when static/dynamic memory may be used
- ❑ Discuss the function of the different buses in operations that may be carried out
- ❑ Discuss the different types of interfaces

Von Neumann Architecture

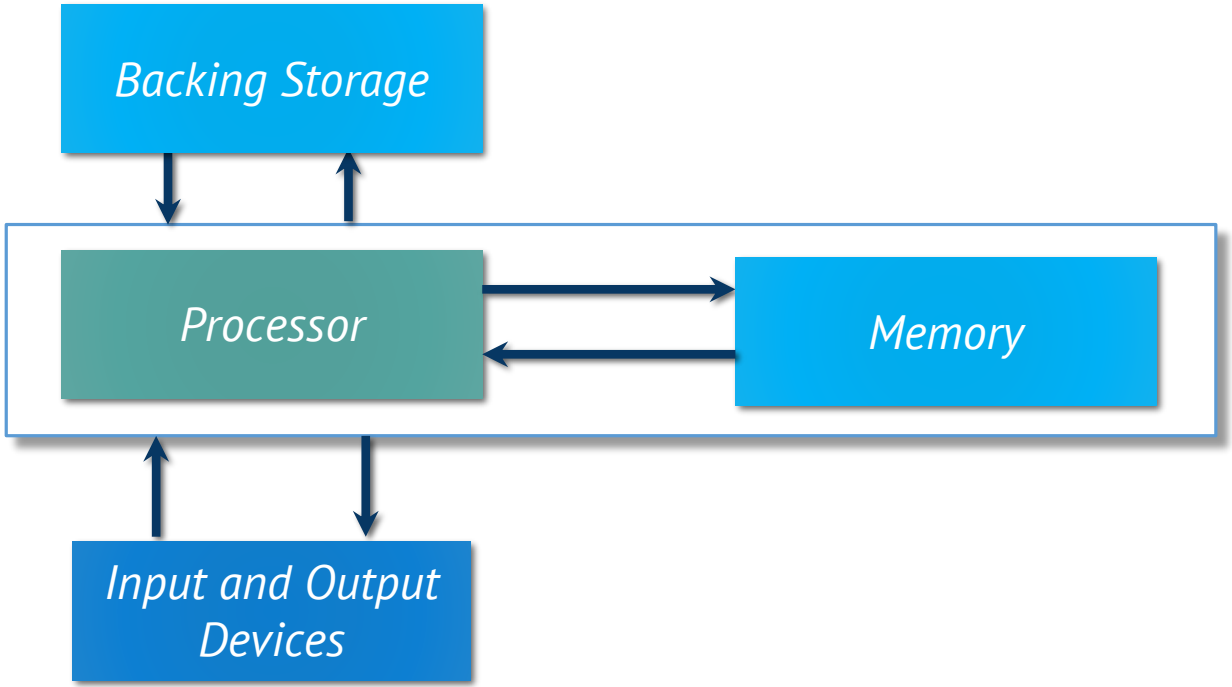
The **Von Neumann architecture** was originally described in a paper by John von Neumann in 1945

It described a computer that had 4 main parts:

- **CA – Central Arithmetic Logic Unit**
- **CC – Central Control Unit**
- **Memory**
- **Input and Output Channels**

First Draft of a Report on the EDVAC – Von Nuemann, June 30 1945, University of Pennsylvania

The Von Neumann model



Von Neumann Architecture

Input Devices
These are devices that enter data **INTO** the computer. i.e. Keyboard and mouse

Output Devices
These are devices that generate output **FROM** the computer i.e. printer.

Memory
There are two types of memory. RAM and ROM

Backing Storage
This is required to store **permanent** copies of our files.

Stored Program Concept

Von Nuemann proposed that instructions should be stored in binary within memory.

This is referred to as the **Stored Program Concept**

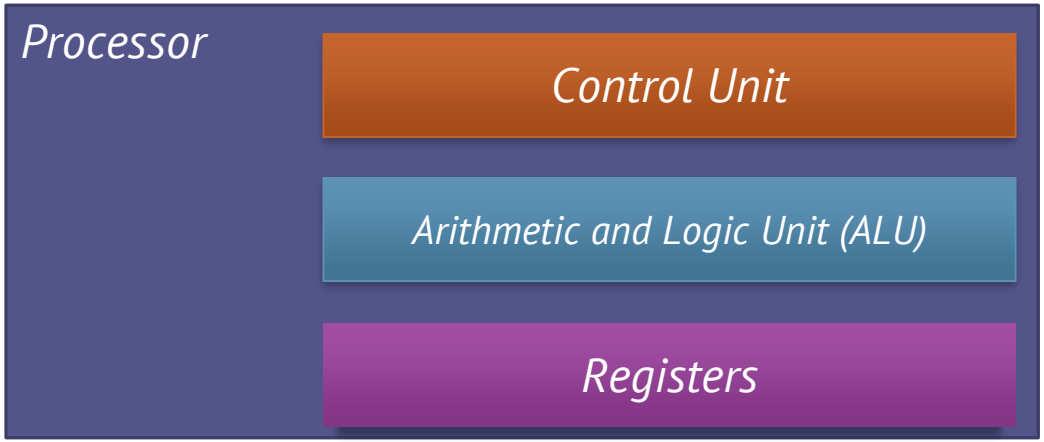
There are other architectures that treat data and instructions separately

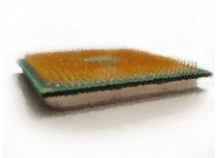
The Processor

The **processor** is the ‘brain’ of the computer.

It is the piece of equipment that actually performs the instructions required.

It can be shown in the following diagram:





Parts of the CPU

Control Unit

This controls the order of execution and the timing of instructions

Arithmetic and Logic Unit (ALU)

This performs the calculations for the processor. It also carries out the Logical decisions.

Registers

These are temporary stores of information, at best they will store a few bytes of information.

The Control Unit

The **control unit's** main purpose is in the execution of instructions

It ensures that instructions are fetched, decoded and executed in the correct order

It has a number of 'lines' that control various operations

The Control Unit

The control lines in the Control Unit have various functions:

Control Line	Function
Clock	Generates a constant pulse (measured in Mhz). One operation will be carried out per pulse
Reset	Causes the processor to halt execution of the current instruction. All registers cleared and machine reboots.
Interrupt	Notifies the processor that an external event has occurred such as I/O from a device. This type of interrupt can be ignored
Non Maskable Interrupt (NMI)	This is an interrupt that CANNOT be ignored such as a low power failure notification.

Arithmetic and Logic Unit

The **ALU's** primary role is arithmetic.

Such as addition, multiplication etc.

With regards to logic it will perform operations such as:

Logical AND/OR

Such as when used in programming or when used in processor instructions

One particular register used by the ALU is the **Accumulator**

This holds the results of calculations.

Intel calls this the EAX

Some more about Registers

Registers are small temporary stores of information.
They are on the processor and are very quick to access!

They can hold:

- ❑ Data
- ❑ Instructions
- ❑ Addresses

Modern processors have general purpose register sizes of up to 64 bits

Registers

Some particular registers are shown below

Register Name	Description
Instruction Pointer	Holds the address in memory of the next instruction to be executed
Return Instruction Pointer	Holds the address in memory of the instruction to be executed after a called procedure is finished
Memory Address Register (MAR)	This is used to hold a value representing the address in memory that the processor needs to access. Would be connected to the Address Bus.
Memory Data Register (MDR)	This is used to hold data read from or being written to memory. Connected to the Data Bus

What does a bus do?

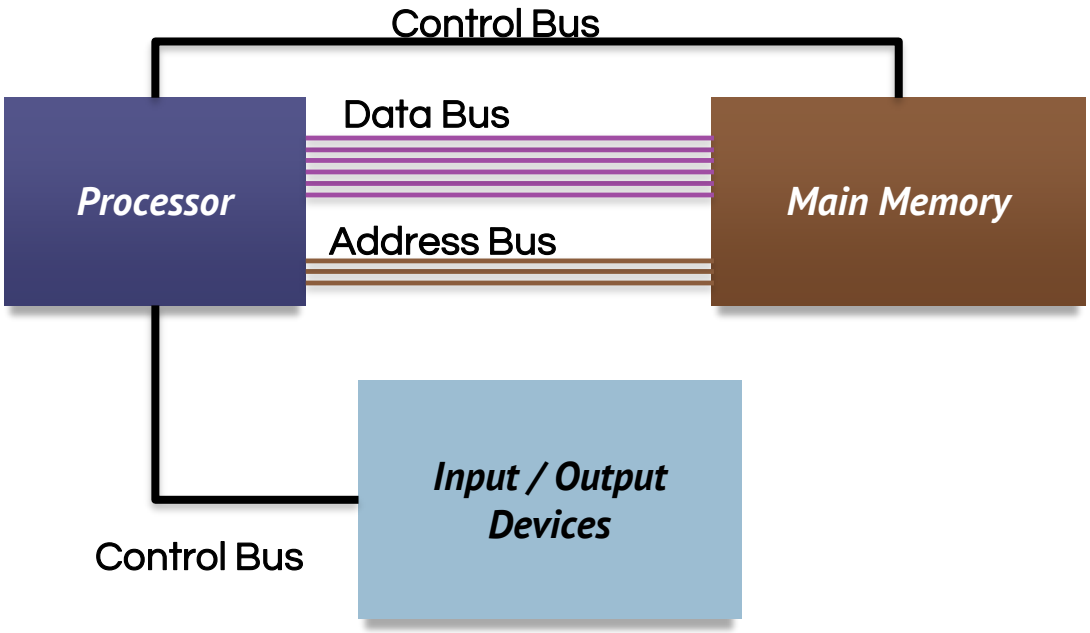


Image recreated from http://homepages.ius.edu/RWISEMAN/C335/HTML/ch1_1.jpg

Functions of the buses

Data Bus

The data bus is used to carry data to and from the CPU and Memory.
It will hold a binary word – **BI DIRECTIONAL**

Address Bus

This carries the address of the memory location that is being used
It is **UNI DIRECTIONAL**

Control Bus/Unit

This doesn't carry data but has a number of status signals it can send

Data Buses

The data bus is used to carry data to and from the CPU.

It will hold a binary word.

This is the biggest binary number that can be processed by the processor at one time. At present 32 or 64 bits

It is Bi-Directional

Means it can carry data to and from the CPU

32 bit and 64 bit computing?

You often hear a processor being described as 32 bit or 64 bit.

This is usually referring to the '**word length**' of the computer

This is the largest binary number that the processor can manipulate in one operation.

The effect of Bus changes

Changing the **width** of the **data bus** means that we can **move more or less data in a single operation**

- Hopefully leading to an increase in performance due to less operations required to move

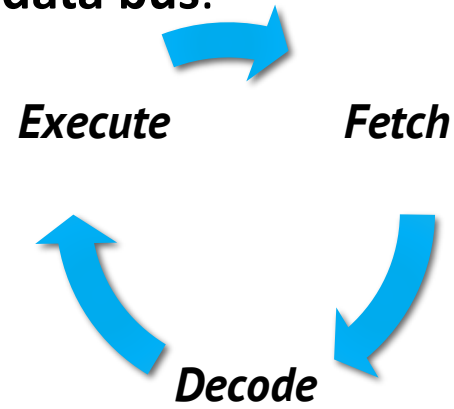
Changing the **width** of the **address bus** means that we can **address more memory**

- Not necessarily leading to an increase in performance in all cases

The Fetch Execute cycle

There is a set series of steps that takes place for a processor to **execute** an instruction

1. The **memory address** of the next instruction is placed on the **address bus**.
2. **Read line** is activated
3. The instruction is **transferred to the processor on the data bus**.
4. Instruction is **decoded and executed**



Memory Write Operation

When the processor is writing data to memory it will use the following steps

1. Set up the **MAR** with the **address to be written to**
2. Set up the **MDR** with the **data to be written**
3. **Address bus** is set up with the **address to be written to** (from the MAR)
4. Copy **data** from MDR to **Data bus**
5. **Write (control) line** is activated
6. **Data on data bus is placed in memory location** specified by address bus

Memory Read Operation

When the processor is reading data from memory it will use the following steps

1. Place the **address to be read from** in the **MAR**
2. The **Address bus is set up with the address** from the MAR
3. The **Read (control) line** is activated
4. The **Data bus** is **set up with data to be read**
5. The data on data bus is placed in **Memory Data Register**
6. **Decode and then execute**

READ

Summary

The control unit is responsible for timing and execution it has various control lines:

Clock, Reset, Interrupt, NMI

The ALU unit will perform arithmetical and logical decisions
There are numerous registers, some are:

Memory Address Register, Memory Data Register