Tiverton High School Year 11
J277 GCSE Computer Science
Knowledge Organiser / Recap
Part 1: The Von Neumann Architecture and CPU

Unit 1-1-1 Architecture of an Electronic Computer

A **computer** is an **electronic device** that follows a **stored program of instructions**.

The **program** of instructions tell it **how** to process data and how to make things happen e.g. activate outputs.

A **computer system** is a **collection** of parts that **work together** to perform a task. It is comprised of **hardware** and **software**.

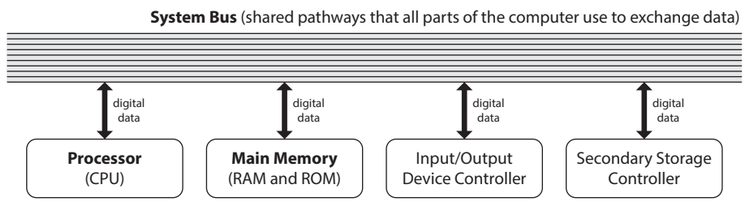
Modern electronic computers are organised in a particular way. This is called the **Von Neumann Architecture**.

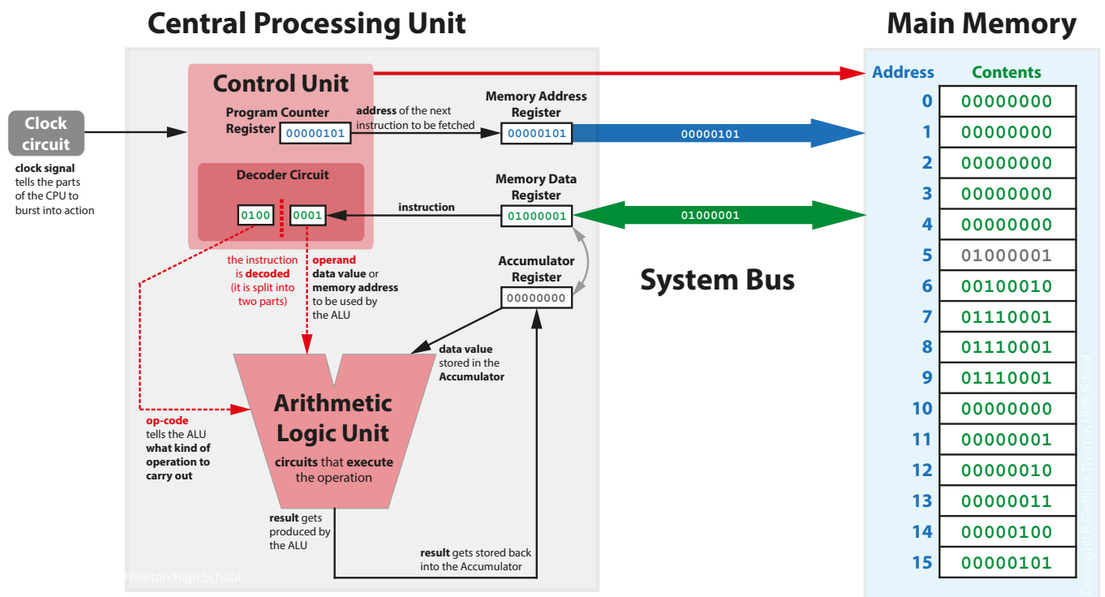
They must contain a **Central Processing Unit (CPU)**, a **main memory** and a **system bus**.

The **CPU** is needed to **execute** the **instructions** from the **program**.
The **Main Memory** (Primary Memory) is used to **store** the **program instructions** and **data values** that the CPU is using.
A **system bus** is a set of **pathways** that program instructions and data can travel along, between different parts of the computer.

Primary Storage (Main Memory) is memory inside the main computer. It is used to hold the program of instructions that the computer will carry out and the data that it will process. Most primary storage is **RAM (Random Access Memory)**.
RAM is **volatile** - it can only store things while the computer is **switched on**.

Secondary Storage Devices store data for **long term/when computer is switched off**.
Examples: **Hard-disk drive**, **solid-state drive**, **optical drive**, **USB Flash-drive**, **magnetic tape drive**.





Unit 1-1-1 Architecture of the Central Processing Unit

The main parts **inside** the CPU are the **Control Unit** and the **Arithmetic Logic Unit**.

Together, these parts carry out the **Fetch-Decode-Execute cycle** to run a computer program.

The **Control Unit** **supervises the fetching and decoding of each program instruction from main memory**.

The **Arithmetic Logic Unit** **carries out** (**executes**) **mathematical** (**arithmetic**) **and logical operations** (**comparisons**).

The CPU also contains **Registers**. A register is a **single high-speed memory location inside the CPU**.

The **Program Counter** is a special **register** inside the Control Unit that **holds the memory address of the next program instruction that needs to be fetched** from main memory into the CPU.

When transferring data values and program instructions between the CPU and main memory, the computer uses the **MAR** and **MDR** registers.

MAR (**Memory Address Register**) is used to determine **which memory address will be used in a transfer** between the CPU and main memory.

MDR (**Memory Data Register**) is used as a **buffer** (a holding place) for the actual **value** being transfered into/out of the CPU.

Once the Arithmetic Logic Unit has carried out an operation, any **result** that is produced will usually be stored in a special register called the **Accumulator register**.

Unit 1-1-2 Performance of the CPU

Clock speed - **Increasing the clock speed can potentially increase the number of instructions executed per second**
e.g. increasing clock speed from 2 MHz to 2.5 MHz.

Cache size - **Increasing the cache size will reduce the number of memory transfers between the CPU and RAM**, so it may speed up processing. With a large cache memory, **less time is spent transferring data between the CPU and RAM**.

Number of cores - **Increasing the number of cores may increase the number of instructions executed per second**
e.g. use a **quad-core** CPU instead of a **single-core** CPU.

Unit 1-1-3 Embedded Systems

An **embedded system** is a **dedicated single-purpose computer** that is **built into some other electronic device**.
The embedded computer **controls the operation of that device**.

Embedded systems usually use **cheap, simple electronic components**, so they are **ideal for mass-produced electronic devices and consumer goods.**

Examples of embedded systems include **burglar-alarms**, **central-heating controllers**, **microwave ovens**, **dishwashers**, **washing machines**, **digital TV receiver boxes** and **smart-TVs**, **robot vacuum cleaners**, **satellite-navigation systems**, **car-alarms** and **engine immobilisers**, **fuel management systems**, **traffic lights** and **pedestrian crossings**, **lifts/elevators**.

An embedded system usually contains these parts:
A **simple CPU** to execute program instructions. This processor may not be very fast or process many bits in one operation.
A **small amount of RAM** to store values that change while the control program is running (e.g. variables that are needed).
A **non-volatile ROM chip** or some **flash memory** to hold the low-level program that the system will run when you switch it on.
Some **input/output ports** that allow it to connect to other components, such as buttons, LED lights, timers etc.

Many of these parts can be built into one single, compact chip, called a "**System On A Chip**".
An embedded system **usually contains fewer parts/components than a general-purpose desktop computer**.
Because they contain fewer, simpler parts, they **usually cost less to produce** and **consume less power than a full-size computer**.