**The purpose of the CPU**

The central processing unit (CPU) is the most important hardware component in a computer. It has two main functions:

* to process data and instructions
* to control the rest of the computer system

All programs and data processing are run in the CPU and all hardware components are, to some extent, controlled by it.

All general purpose computers follow the same basic model. This diagram illustrates the flow of data within a computer.

The CPU receives instructions and data from an input or memory. The instructions and data are processed by the CPU and the results are either sent to an output or transferred to secondary storage. Data is held in primary storage while it is being processed.

Input is received from an input device such as a keyboard, mouse, camera or scanner. Output is sent to an output device such as a monitor, printer or speaker.

# Common CPU components

The central processing unit (CPU) consists of six main components:

* control unit (CU)
* arithmetic logic unit (ALU)
* registers
* cache
* buses
* clock

All the components work together to allow processing and system control.

## Control unit (CU)

The CU provides several functions:

* it fetches, decodes and executes instructions
* it issues control signals that control hardware components within the CPU
* it transfers data and instructions around the system

## Arithmetic logic unit (ALU)

The ALU has two main functions:

* it performs arithmetic and logical operations (decisions).
* it acts as a gateway between primary storage and secondary storage - data transferred between them passes through the ALU.

**The ALU performs arithmetic and logical operations.**

## Registers

Registers are small amounts of high-speed memory contained within the CPU. They are used by the processor to store small amounts of data that are needed during processing, such as:

* the address of the next instruction to be executed
* the current instruction being decoded
* the results of calculations

Different processors have different numbers of registers for different purposes. Most have some, or all, of the following:

* program counter (PC)
* memory address register (MAR)
* memory data register (MDR)
* current instruction register (CIR)
* accumulator (ACC)

## Cache

Cache is a small amount of high-speed random access memory (RAM) built directly within the processor. It is used to temporarily hold data and instructions that the processor is likely to reuse. This allows for faster processing, as the processor does not have to wait for the data and instructions to be fetched from the RAM.

## Clock

The CPU contains a clock which, along with the CU, is used to coordinate all of the computer's components. The clock sends out a regular electrical pulse which synchronises (keeps in time) all the components.

The frequency of the pulses is known as clock speed. Clock speed is measured in hertz (Hz). The greater the speed, the more instructions can be performed in any given moment of time.

In the 1980s, processors commonly ran at a rate of between 3 megahertz (MHz) and 5 MHz, which is 3 million to 5 million pulses or cycles per second. Today, processors commonly run at a rate of between 3 gigahertz (GHz) and 5 GHz, which is 3 billion to 5 billion pulses or cycles per second.

## Buses

A bus is a high-speed internal connection. Buses are used to send control signals and data between the processor and other components.

Three types of bus are used.

* Address bus - carries memory addresses from the processor to other components such as primary storage and input/output devices. The address bus is unidirectional.
* Data bus - carries the data between the processor and other components. The data bus is bidirectional.
* Control bus - carries control signals from the processor to other components. The control bus also carries the clock's pulses. The control bus is unidirectional.

# Von Neumann architecture

Von Neumann architecture is the design upon which many general purpose computers are based. The key elements of von Neumann architecture are:

* data and instructions are both stored as binary digits
* data and instructions are both stored in primary storage
* instructions are fetched from memory one at a time and in order (serially)
* the processor decodes and executes an instruction, before cycling around to fetch the next instruction
* the cycle continues until no more instructions are available

A processor based on von Neumann architecture has five special registers which it uses for processing:

* the program counter (PC) holds the memory address of the next instruction to be fetched from primary storage
* the memory address register (MAR) holds the address of the current instruction that is to be fetched from memory, or the address in memory to which data is to be transferred
* the memory data register (MDR) holds the contents found at the address held in the MAR, or data which is to be transferred to primary storage
* the current instruction register (CIR) holds the instruction that is currently being decoded and executed
* the accumulator (ACC) is a special purpose register and is used by the arithmetic logic unit (ALU) to hold the data being processed and the results of calculations

## Harvard architecture

In early computer systems, machine instructions were stored on punch cards and data could be stored on another media such as magnetic tape. This kept the instructions and data entirely separate from one another, known as the Harvard architecture. In modern computer systems this can be achieved by using a central processing unit with two separate memory units, one to store machine instructions and another to store data, which are connected by different buses.

**The fetch-decode-execute cycle**

The fetch-decode-execute cycle is a key feature of the von Neumann architecture and consists of seven stages:

1. The memory address held in the program counter (PC) is copied into the memory address register (MAR).
2. The address in the program counter is incremented (increased) by one. The program counter now holds the address of the next instruction to be fetched.
3. The processor sends a signal along the address bus to the memory address held in the MAR.
4. The instruction or data held in that memory address is sent along the data bus to the memory data register (MDR).
5. The instruction or data held in the MDR is copied into the current instruction register (CIR).
6. The instruction or data held in the CIR is decoded and then executed. Results of processing are stored in the accumulator (ACC).
7. The cycle returns to step one.

# Factors affecting CPU performance

Even though today's processors are tremendously fast, their performance can be affected by several factors, such as:

* clock speed
* cache size
* number of cores

**Processor performance can be affected by clock speed, cache size and the number of cores the processor has.**

## Clock speed

Clock speed is the number of pulses the central processing unit’s (CPU) clock generates per second. It is measured in hertz (Hz).

The faster the clock speed, the faster the computer is able to run fetch-decode-execute cycles. This means that it can process more instructions in the same amount of time.

CPU clocks can sometimes be sped up by the user. This process is known as overclocking. When the number of pulses per second is increased, more fetch-decode-execute cycles can be performed and more instructions can be processed in a given time. This increases performance, but also requires more power, which results in a greater need for heat dissipation and can strain the life of the battery. Overclocking causes the CPU to work harder and produce more heat, which can lead to long-term damage to the hardware.

## Cache

Cache is a small amount of high-speed random access memory (RAM) built directly within the processor. It is used to temporarily hold data and instructions that the processor is likely to reuse. This allows for faster processing as the processor does not have to wait for the data and instructions to be fetched from RAM.

The larger the cache size, the less time a processor has to wait for instructions to be fetched. This improves performance.

## Number of cores

A processing unit within a CPU is known as a core. Each core is capable of fetching, decoding and executing its own instructions.

A CPU with more cores can process more instructions in a given time. However, sometimes one core will have to wait for the output from another before carrying out the next phase of the instruction, which can cause a delay.

Multiple cores increase the processor cost. Many modern CPUs are dual-core (two) or quad-core (four) processors. This provides vastly superior processing power.

## RISC and CISC

There are two main types of processor:

* reduced instruction set computer (RISC)
* complex instruction set computer (CISC)

RISC processors can process a limited number of relatively simple instructions by breaking each one down into even simpler instructions that can be carried out quickly. Processing simpler instructions requires less circuitry, which consumes less power, so less heat is generated. This makes them ideal for use in smartphones. RISC processors may be used in an embedded system where it is not necessary to process complex instructions and where the instructions are relatively simple and repetitive. They use less power so do not have a need for dedicated cooling systems.

CISC processors are physically larger and can process more complex instructions. They can understand and carry out complex tasks with only a few instructions. Processing complex instructions requires more complex circuitry which needs more power and therefore more heat is generated. Most modern computers use CISC processors.