The Four Primary Functions of the CPU

The CPU processes instructions it receives in the process of decoding data. In processing this data, the CPU performs four basic steps:

1. **Fetch:** Each instruction is stored in memory and has its own address. The processor takes this address number from the program counter, which is responsible for tracking which instructions the CPU should execute next.
2. **Decode:** All programs to be executed are translated to into Assembly instructions. Assembly code must be decoded into binary instructions, which are understandable to your CPU. This step is called decoding.
3. **Execute:** While executing instructions the CPU can do one of three things: Do calculations with its ALU, move data from one memory location to another, or jump to a different address.
4. **Store:** The CPU must give feedback after executing an instruction, and the output data is written to the memory.

Fundamental principle of cpu

- **Fetch instruction**: The CPU reads an instruction from memory.
- **Interpret instruction**: The instruction is decoded to determine what action is required.
- **Fetch data**: The execution of an instruction may require reading data from memory or an I/O module.
- **Process data**: The execution of an instruction may require performing some arithmetic or logical operation on data.
- **Write data**: The results of an execution may require writing data to memory or an I/O module.

Von Neumann architecture

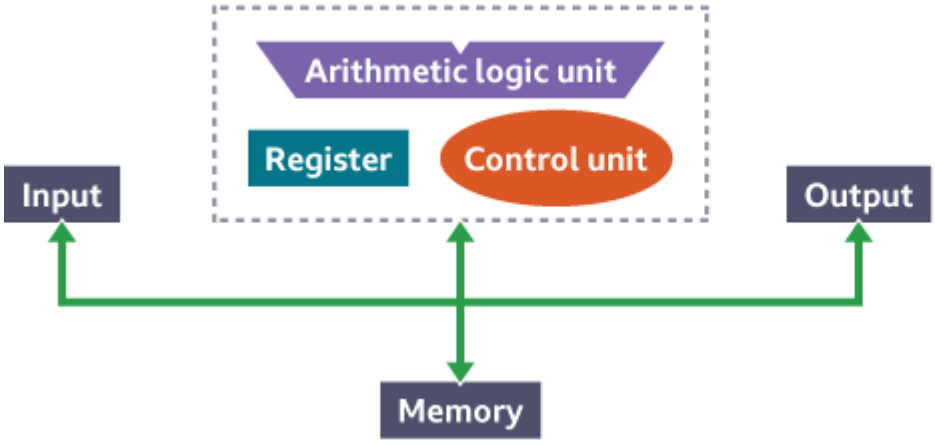
Von Neumann architecture is the design upon which many general purpose computers are based. The key elements of von Neumann architecture are:

- 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



Von neumann architecture

REGISTER

Registers are a type of computer memory used to quickly accept, store, and transfer data and instructions that are being used immediately by the CPU. The registers used by the CPU are often termed as Processor registers. A processor register may hold an instruction, a storage address, or any data (such as bit sequence or individual characters). The computer needs processor registers for manipulating data and a register for holding a memory address. The register holding the memory location is used to calculate the address of the next instruction after the execution of the current instruction is completed.

USE OF REGISTER

Fetching : this fetch instructions thay are keyed in by the user and locate the potentially stored data from the system.

Decoding : this interpretes the sets of instructions and process them according to their desired result by the user.

Execution : Many computer related tasks are processed by the registers, stored into the computer memory, and given out as per the commands keyed in by the user. Information is processed according to the needs of the user.