1.1 – Systems architecture

1.1.1 Architecture of the CPU

The purpose of a CPU is to fetch, decode and execute instructions, it does this by:

Fetch

- The address of the next instruction is copied from the program counter and placed in the MAR.
- The MAR now contains a memory address. the data that is stored at that address and copies it to the MDR.
- The program counter is incremented to point to the next instruction to be processed in the program, ready for the next cycle.

Decode

- The MDR now contains either data fetched from memory or an instruction. The control unit decodes the instruction to see what to do.

Execute

- The decoded instruction is executed. This might mean performing a calculation using the ALU.

Common CPU components and their function: CPU is made up from a number of components

ALU - Arithmetic Logic Unit

- add, subtract, repeated addition (multiply), repeated subtraction (division)
- AND, OR, XOR, NOT operations
- binary shift operations

Control Unit

- co-ordinate the activity of the CPU
- sending out control signals to control/synchronise how data moves around the parts of the CPU and memory
- decoding instructions

Cache Memory

- Most frequently used instructions are copied here to reduce the time needed to fetch instructions from the RAM

Registers

- Memory locations within the CPU that hold specific pieces of data temporarily and can be accessed very quickly

CPU architecture describes how the different components in the CPU are laid out and communicate with each other. The Von Neumann architecture describes a computer in which the data and instructions are stored in the same area of memory.

Four important registers in a CPU with Von Neumann architecture:

Memory address register (MAR)

- Stores the address of the data to be fetched from, or the address where the data is to be stored

Memory data register (MDR)

- Stores the data itself which has been read from main memory or is about to be written to main memory

Program Counter

- Stores the address of the next instruction to be fetched from memory. (Which sends the value to the MAR) This counter increments by 1 in each FDE cycle

Accumulator

- Stores the results of calculations

1.1.2 CPU performance

Clock Speed

- The CPU is constantly fetching and executing instructions and the speed at which it does this is determined by an electronic clock
- Clock speeds are measured in hertz (Hz), and means 'number of times per second'
- CPU's usually work at speeds of up to 4GHz (or 4 billion instructions per second)
- Each 'tick' of the clock represents one step in the FDE cycle. The faster the clock speed, the more instructions that can be executed every second meaning the program takes less time to run.

Cache Size

- Cache memory is located between the main memory and the CPU. It is used to hold data that needs to be accessed very quickly.
- Accessing cache memory is much faster than accessing main memory (RAM)
- The more data that can be stored in cache memory rather than main memory, the faster and more efficient the process.
- Cache memory is very expensive.

Number of processor cores

- Each core can fetch and execute instructions independently so a multiple core processor can handle several instructions at the same time.
- While these multiple cores can work on separate programs or parts of a program at the same time, this is only possible if the program has been written to take advantage of multiple cores.

1.1.3 Embedded systems

- An embedded system is a computer system that has been designed for a dedicated function.
- Embedded systems are often manufactured as a single chip
- The dedicated hardware and software make embedded systems more robust and reliable than general-purpose computers.
- Designed and engineered to perform a limited set of tasks to reduce size and improve performance
- Low power consumption to operate from a small power source.
- Small in size and low cost
- Examples include, washing machines, microwaves, home security systems, telephones, televisions, car engine management systems.