GCST

1.1 System Architecture Cheatsheet

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.

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

Fetch
- The instruction is moved from memory to the CPU

Decode
- The instruction is understood by the CPU

Execute
- The instruction is carried out

<u>Factors affecting CPU Performance</u>
Clock Speed - The faster the clock speed, the more instructions that can be executed every second meaning the program takes less time to run.

Cache Size - The more data that can be stored in cache memory rather than main memory, the faster and more efficient the process.

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.

<u>Embedded Systems</u>
An embedded system is a computer system that has been designed for a dedicated function.
Designed and engineered to perform a limited set of tasks to reduce size and improve performance. Low power consumption and small in size and low cost.
Examples include, washing machines