Component 1 | 1.1.1 | Structure and function of the processor

Specification & learning objectives

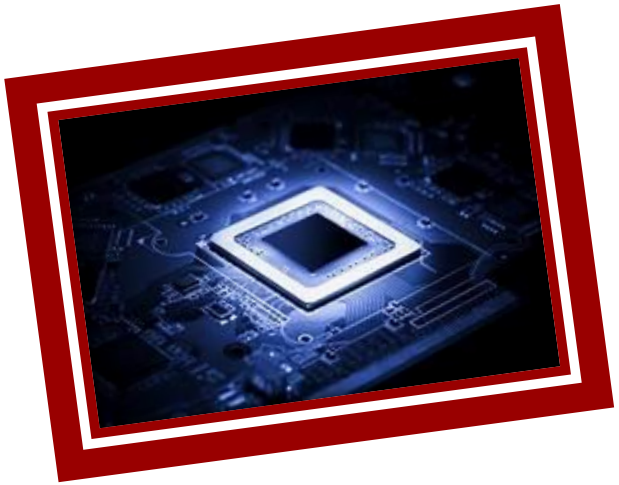
	A Level	Specification point description
	1.1.1a	The arithmetic logic unit; ALU, Control Unit and Registers (Program Counter; PC, Accumulator; ACC, Memory Address Register; MAR, Memory Data Register; MDR, Current Instruction Register; CIR). Busses: data, address and control: How this relates to assembly language program.
	1.1.1b	The fetch-decode-execute cycle, including its effect on registers.
	1.1.1c	The factors affecting the performance of CPU, clock speed, number of cores, cache.
	1.1.1d	The use of pipelining in a processor to improve efficiency.
	1.1.1e	Von Neumann, Harvard and contemporary processor architecture.

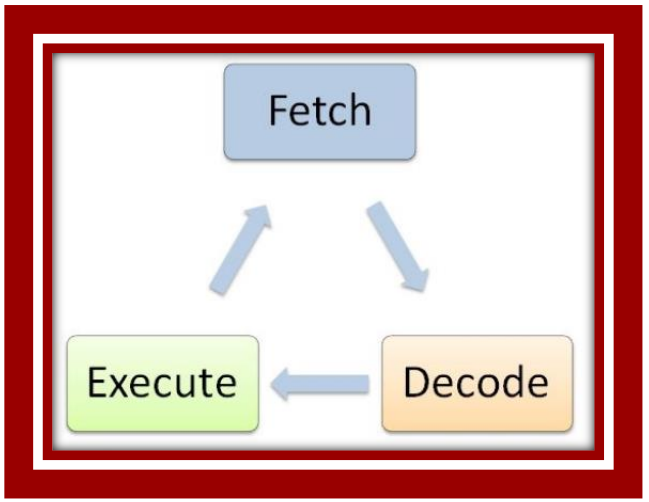
Resources

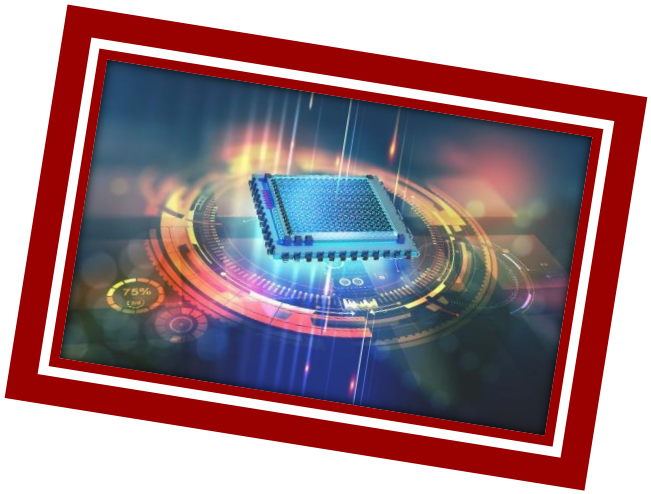
PG Online textbook page ref: 1-12

Hodder textbook page ref: 124-133

CraignDave videos for SLR 1





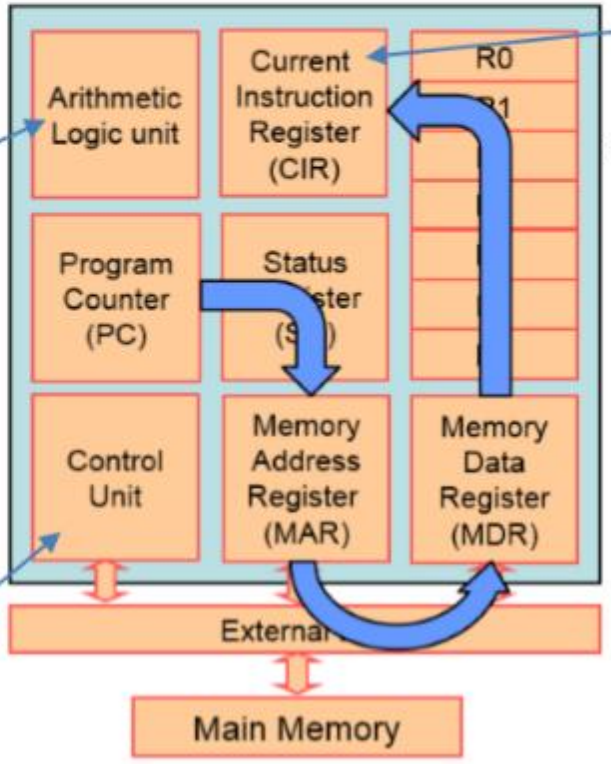


The 'Fetch' part of the cycle

1) The PC holds the address of the next instruction to be carried out.
2) This address is copied into the MAR.
3) The contents of the address in the MAR are copied into the MDR.
4) The contents of the MDR are copied into the CIR.
5) The contents of the PC are incremented.

Arithmetic Logic Unit: "The part of the CPU where data is processed and manipulated. This processing and manipulation normally consists of arithmetic operations or logical comparisons allowing a program to make decisions."

Control Unit: "The part of the CPU that manages the execution of instructions. The control unit fetches each instruction in sequence, and decodes and synchronises it before executing it by sending control signals to other parts of the computer."



Current Instruction Register: "A register in the control unit that stores the address of the next instruction currently being executed and decoded."

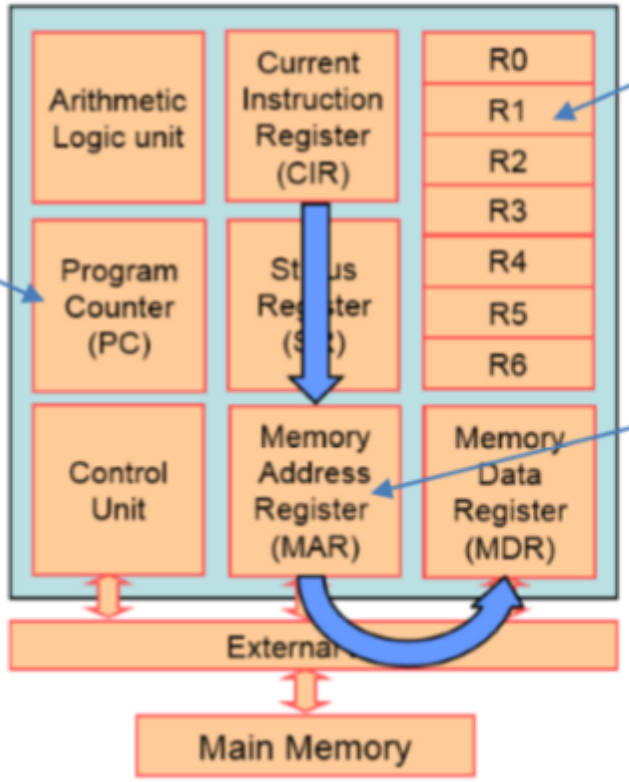
Register: "Tiny areas of extremely fast memory located in the CPU normally designed for a specific purpose, where data or control information is stored temporarily."

The 'Decode' part of the cycle

6) The contents of the CIR are then divided into the binary code standing for the operation to be carried out, and probably the address of the data that will be used by the program.

7) The control unit then interprets the operation code so that the processor knows what to do next.

Program Counter: "A register in the control unit which holds the address of the next instruction to be executed."



Accumulator: "A special register within the ALU. It is used to hold the data currently being processed by the central processor. Any data to be processed is stored temporarily in the accumulator, the results ending up back in the accumulator being stored in the memory unit."

Memory Address Register: "A register in the CPU that stores the address of the memory location currently in use. In the fetch phase, this would be the address of the instruction being loaded; in the execute phase, it would be the address of the data being used."

The 'Execute' part of the cycle

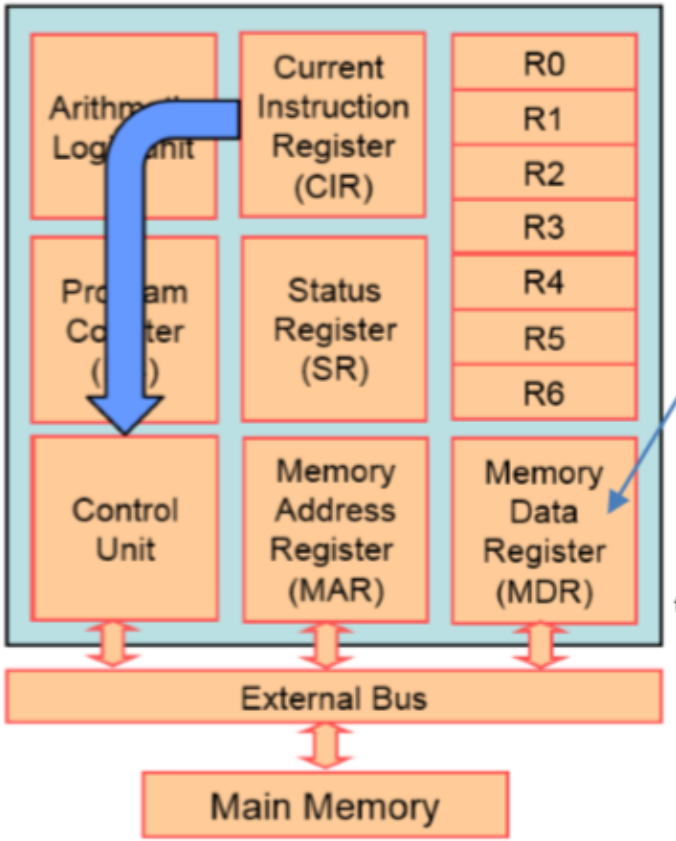
8) The address part of the instruction is copied from the CIR to the MAR.

9) The data found in the address in the MAR is copied to the MDR.

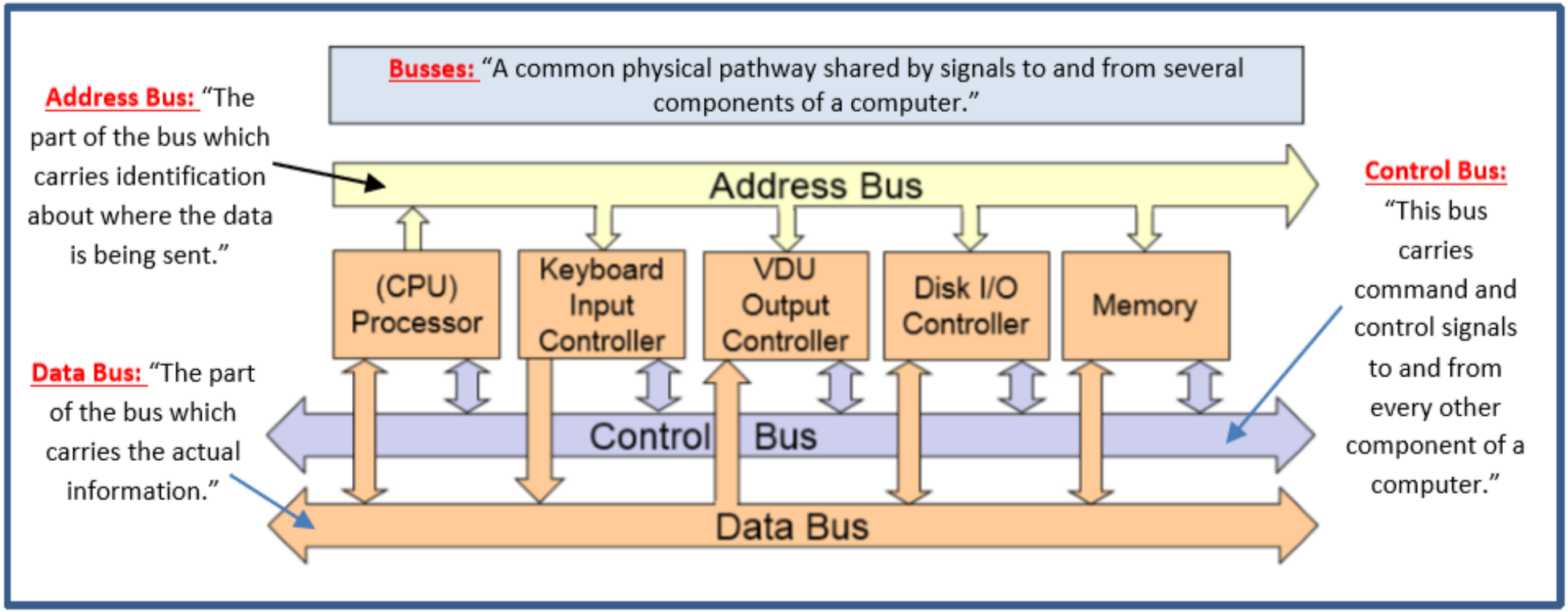
10) The data is used.

Point 10 in reality is more complex than this because the action depends entirely on the type of instruction.

Can you think of an example of what might actually happen here?



Memory Data Register: "A register in the CPU that stores data being transferred to and from the immediate-access store. It acts as a buffer, allowing the central processor and memory unit to act independently without being affected by minor differences in operation. A data item will be copied to the MDR ready for use at the next clock pulse, when it can either be used by the central processor or be stored in main memory."

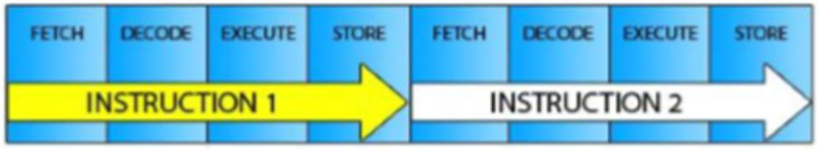

Busses: "A common physical pathway shared by signals to and from several components of a computer."
Address Bus: "The part of the bus which carries identification about where the data is being sent."
Address Bus
(CPU) Processor
Keyboard Input Controller
VDU Output Controller
Disk I/O Controller
Memory
Control Bus: "This bus carries command and control signals to and from every other component of a computer."
Data Bus: "The part of the bus which carries the actual information."
Control Bus
Data Bus

Component 1 | 1.1.1 | Structure and function of the processor

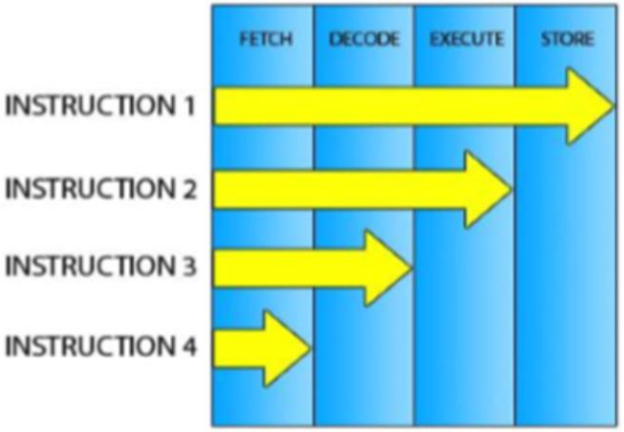
Address	**The location in memory of a particular instruction or piece of data.**
Accumulator	Holds the results of calculations made by the ALU.
Buses	Physical connections between computer components that carry data and instructions back and forth.
CIR	Holds the instruction currently being processed by the control unit .
MAR	Temporarily holds the address of the next instruction or data to be loaded from RAM.
MDR	Holds instructions on their way from memory, as well as data passing in both directions.
Program Counter	Holds the address of the next instruction to be processed.
Instruction set	A complete list of instructions that the CPU is capable of carrying out.
Register	A fast type of memory located within the CPU used to hold temporary data while a program is running.

Key question: How can the speed of a processor be increased further?

MACHINE CYCLE (without pipeline):



MACHINE CYCLE (with pipeline):

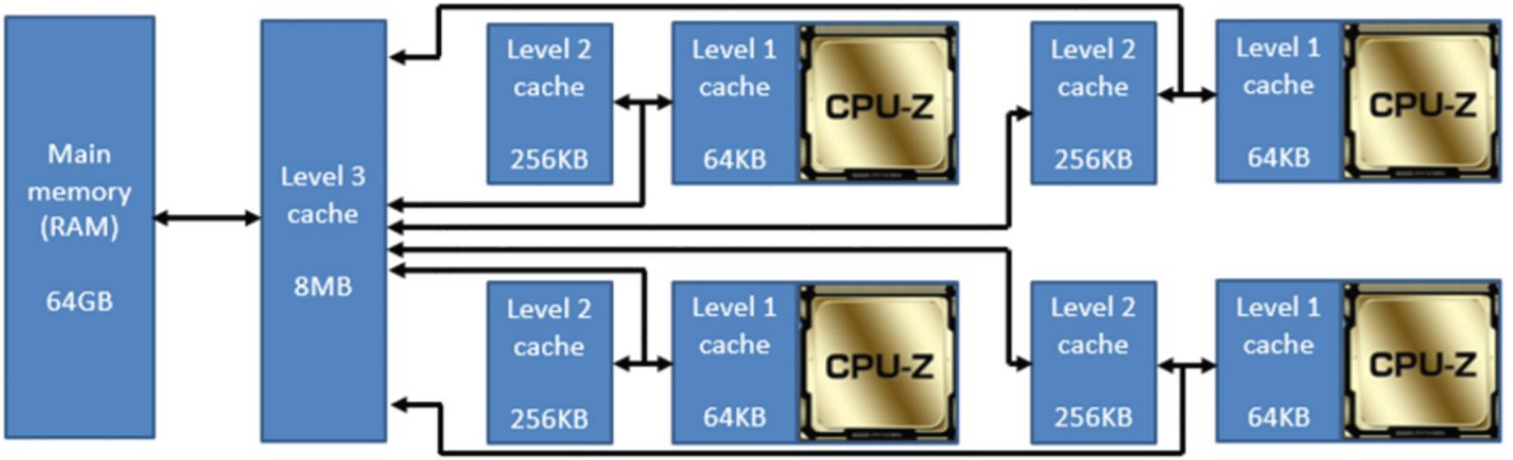


Pipelining: The concurrent decoding of two or more machine instructions. While part of one instruction (for example, an address field) is being decoded, another part of a second instructions (for example, an operation code) may also be decoded."

Key question: How can the speed of a processor be increased further?

Cache: "A part of the main store between the central processor and the rest of the memory. It has extremely fast access, so sections of a program and its associated data are copied there to take advantage of its short fetch cycle."

A quad-core i7 Haswell processor with dedicated Level 1 and Level 2 cache for each core and a shared Level 3 cache across all cores.



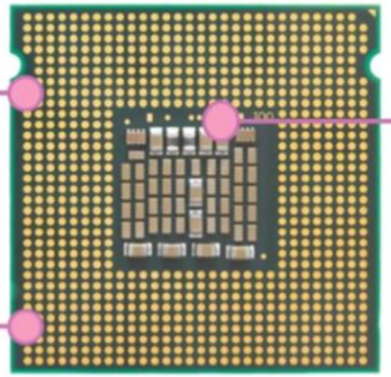
Level 1 Cache: Often located directly on the CPU itself, very low capacity, very expensive, typically runs at the same speed as the CPU.

Level 2 Cache: Often part of the CPU module, still very expensive, larger capacity, but still less than 1MB, runs at CPU or close to CPU speed

Level 3 Cache: Often located further away on motherboard, less expensive, larger capacity, now in MBs, typically shared between on cores.

Key question: How can the speed of a processor be increased further?

Cores: "A part of a multi-core processor. A multi-core processor is a single component with two or more independent actual CPUs, which are the units responsibly for the fetch-decode-execute cycle."



Clock speed

Cycles per second measured in hertz

Number of cores

The number of duplicate CPUs on a single chip

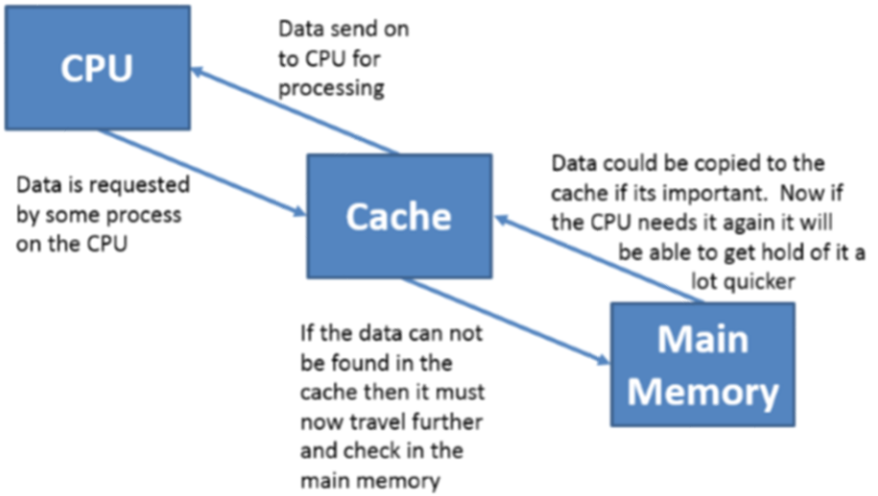
Cache size

Superfast working memory

Clock Speed: "Measured in Hertz, the clock speed is the frequency at which the internal clock generates pulses. The higher the clock rate, the faster the computer may work. The "clock" is the electronic unit that synchronises related components by generating pulses at a constant rate."

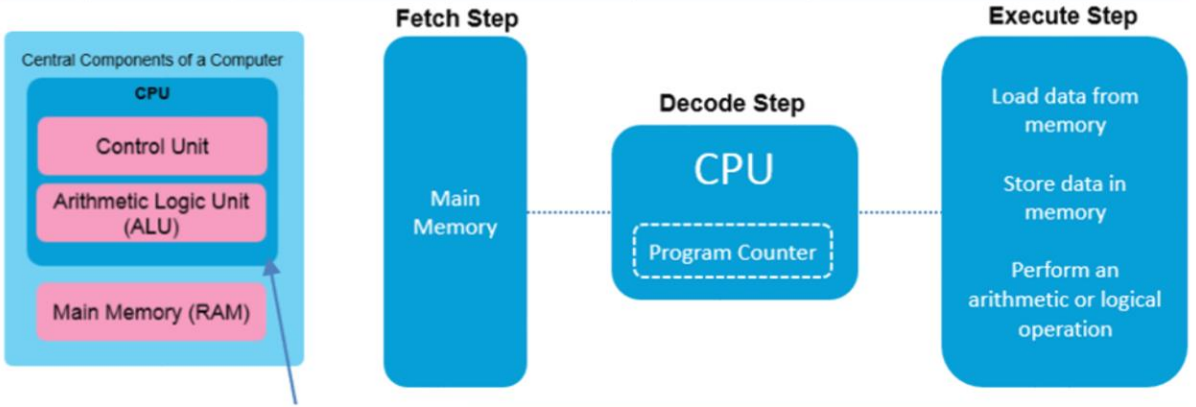
PROCESSOR CLOCK SPEEDS

One cycle per second = 1 Hertz (Hz) = 1 instruction carried out each second
1 Kilohertz (KHz) = 1024 cycles per second
1 Megahertz (MHz) = 1,048,576 cycles per second
1 Gigahertz (GHz) = 1,073,741,824 cycles per second (Approximately 1 billion!)



Component 1 | 1.1.1 | Structure and function of the processor

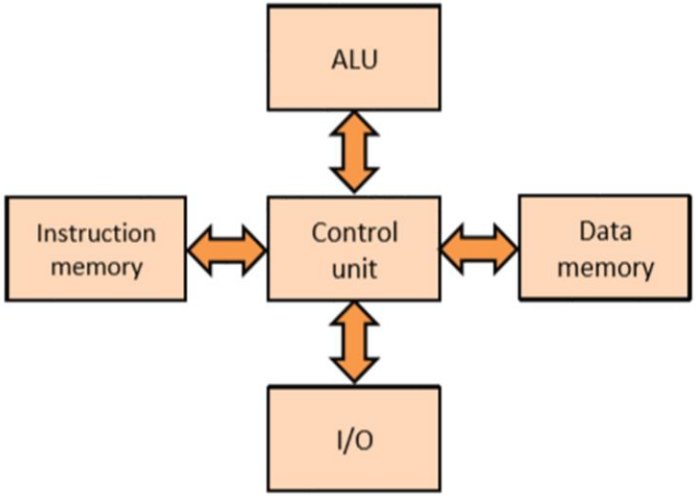
Von Neumann Architecture: "Traditional computer architecture that forms the basis of most digital computer systems. A single control unit manages program control flow following a linear sequence of "fetch-decode-execute"



Central Processing Unit: "The main part of the computer, consisting of the registers, ALU and control unit."

F-D-E: "The complete process of retrieving an instruction from store, decoding it and carrying it out. Also known as the instruction cycle."

Harvard Architecture: "A computer architecture with physically separate storage and signal pathways for instructions and data. These early machines had data torage entirely contained within the central processing unit, and provided no access to the instruction storage as data."



Typical exam questions

1. Describe the stages of the fetch-decode-execute cycle. **[4]**

2. Increasing the number of cores affects the performance of the CPU. Explain why this increase in performance is not linear. **[2]**

3. State the purpose of the address bus. **[1]**

4. Name one register other than the PC found in the CPU and describe its purpose. **[2]**

5. Explain why the Little Man Computing (LMC) instructions BRA / BRP would cause the contents of the PC to change. **[2]**

Component 1 | 1.1.1 | Structure and function of the processor

Target: Overall grade:

Minimum expectations & learning outcomes

☐	Terms 1-21 from your A Level Key Terminology should be included and formatted.
☐	You must include an illustration that depicts the stages of fetch, decode and execute including the connection between the registers, direction and connection of the three buses.
☐	You must show how an assembly language program using the commands ADD, STORE, LOAD, BRANCH affects the contents of the registers.
☐	You must describe how several factors affect the performance of the CPU.
☐	You must explain how pipelining improves the efficiency of a CPU.
☐	You must illustrate the difference between Von Neumann and Harvard processor architectures.
☐	Answer the exam questions.

Feedback

Breadth	Depth	Presentation	Understanding
☐ All	☐ Analysed	☐ Excellent	☐ Excellent
☐ Most	☐ Explained	☐ Good	☐ Good
☐ Some	☐ Described	☐ Fair	☐ Fair
☐ Few	☐ Identified	☐ Poor	☐ Poor

Comment & action required

Component 1 | 1.1.1 | Structure and function of the processor

Reflection & Revision checklist

Confidence	Clarification
☹ 😐 ☺	Candidates need to have an understanding of the purpose and function of the core components of a processor.
☹ 😐 ☺	Candidates need to understand the role and components of the ALU.
☹ 😐 ☺	Candidates need to understand the purpose and function or registers within the processor, including the PC, accumulator, MAR, MDR and CIR.
☹ 😐 ☺	Candidates need to understand the purpose, function and role of the data, address and control buses in the processor.
☹ 😐 ☺	Candidates need to understand how assembly language makes use of registers, and how data and addressed are transferred between registers.
☹ 😐 ☺	Candidates need to understand the purpose and stages within the FDE cycle.
☹ 😐 ☺	Candidates need to understand how and when the registers are used within this cycle, and how and where data and addresses are transmitted to/from in each part of this cycle.
☹ 😐 ☺	Candidates need to understand how the performance of the CPU can be affected by many factors.
☹ 😐 ☺	Candidates need to understand how and why the performance is affected by the clock speed, the number of cores and the size and speed of the cache.
☹ 😐 ☺	Candidates need to have an understanding of the Von Neumann and Harvard architectures. They should be aware of the different approaches the architectures take to storing instructions and data in memory and the benefits of each approach.
☹ 😐 ☺	Candidates will not be asked about specific aspects of “contemporary processor architecture” unless explicitly named in the specification. They may, however, be asked to show an awareness of how contemporary processors differ from a pure Von Neumann architecture in more open questions.