Revision: Hardware

Processor Architecture

Define the terms hardware and software

Hardware - physical parts of a computer system
Software – instructions and programs to make the computer work

State what is meant by a input and output devices

Input - Hardware to put data into a computer

Output—hardware to get data from a computer, for example, screen, printer, actuator—produces physical movement in response to computer signals

Explain what is meant by a peripheral
External to the computer and connects to the computer, for example a printer, keyboard and mouse

Input devices

Manual	Automatic
Keyboards Touch-sensitive keyboards and concept keyboards Touch screens Graphical tablets Mouse Image capture Web Cams and microphones Voice recognition Scanners Digital Cameras Biometric devices – Fingerprint, facial, and retinal scanning	Automatic data input methods: Bar codes Laser scanners Camera based readers Radio Frequency Identification (RFID) **Data logging:** Heart rate sensor GPS (receiver) Accelerometer/gyroscope/motion sensor Thermometers, light and UV sensors, skin response sensors, magnetometers, gyrometer, ECG etc

Describe the function and purpose of the control unit memory unit and ALU

State two items that would be found in the **memory** unit.

- Parts of operating system in current use
- Parts of application software in current use
- Data files in current use

Describe what each of the following parts of a computer does: **control unit, memory unit, ALU.**

Control Unit
- Sends signals to **synchronise** / **control** the F-E cycle
- **Decodes** instructions (in CIR)

Memory Unit
- Stores OS
- Data currently in use
- Software currently in use/ boot program/operations/instructions

ALU
- Carries out arithmetic instructions/calculations
- Carries out logical instructions

Explain the need for, and use of, registers in the functioning of the processor.

Program Counter
- Register holds the address of the next instruction to be executed by the processor.

Memory Address Register

- holds the address of the location where data will be stored or retrieved from memory.
- The position/address in memory... of the location containing either ...the next piece of data to be read or ...the next instruction to be used.

Memory Data Register (MDR)
- The contents of the address specified in the MAR are copied to the MDR
- This may be an instruction/ operation... or data to be used (with an instruction)
- It may contain data to be copied to an address

Current Instruction Register (CIR)
- Holds the instruction while it is being decoded/executed
- The contents of MDR are copied into the CIR if it is an instruction
- Operation code as first part of instruction
- Remainder of instruction is address of data to be used in operation or ...the data to be used if immediate operand is used

Accumulator
- Holds the data currently being processed
- Results of processing are stored in the accumulator
- The results of arithmetic carried out in ALU
- All I/O goes through accumulator

Describe 4 steps in the FDE cycle

Fetch

1. The CPU reads the contents of the Program Counter to find the address of the next instruction to be fetched, decoded and executed.
2. As soon as it is read, the PC increments.
3. The contents from the PC are then copied into the MAR.
4. The address in the MAR is sent across the address bus to locate the contents in RAM
5. The control unit sends a read signal across the control bus to read from the memory address
6. The contents of this address are copied to the MDR via the data bus
7. The MDR now holds the instruction that must be executed.
8. The instruction in the MDR is then copied to the CIR, as we will often need to use the MDR again to complete the execution of an instruction.

Decode

- The control unit decodes the instruction (using the instruction set)
- The contents of the CIR are divided.
- Part of the instruction might be an operation (like ADD) and part of the instruction might be data,
- The ADD part is known as the **OPCODE** and the data part is known as the **OPERAND**.

Execute

The instruction can now be executed.

This could be:

- Load data from memory
- Write data to memory
- Do a calculation or logic operation using the ALU
- Change the address in the PC
- Halt the program

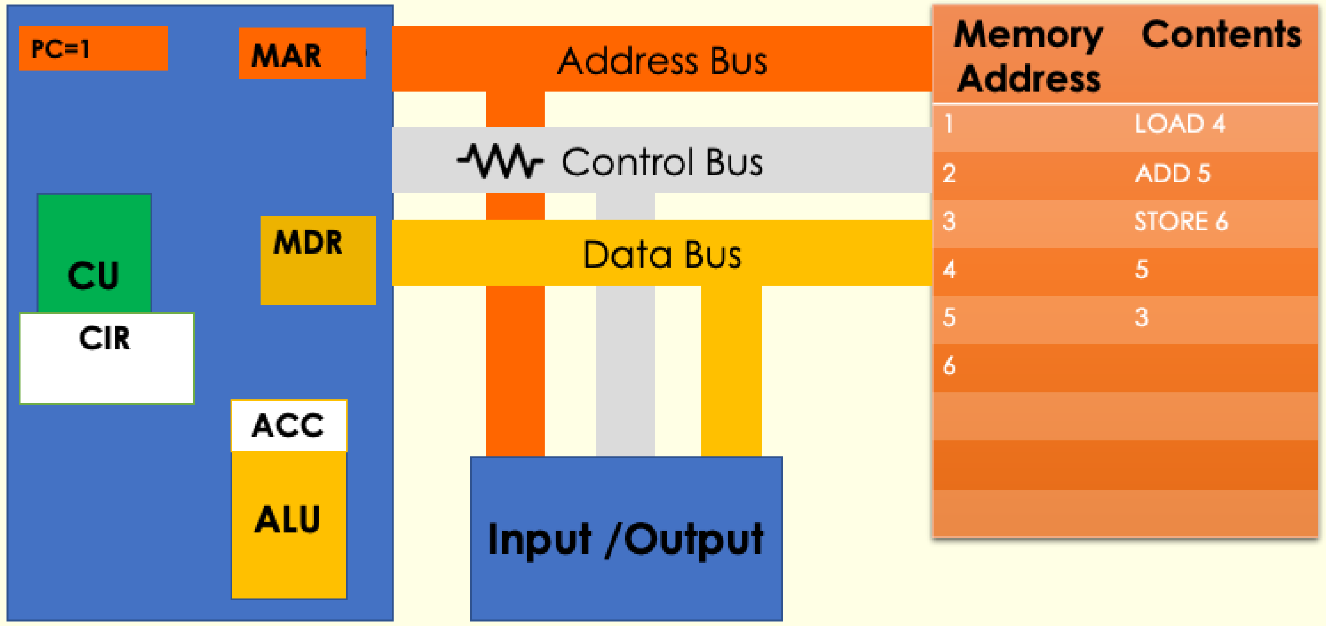
Key words: Contents, Copied SEND/TRANSFERRED

Explain the need for, and describe the use of, buses to convey information

- A bus is a parallel group of wires ...able to transmit groups of bits together ...from one location/register to another in the processor
- Control bus... transmits control signals from the control unit to the rest of the
- Address bus... carries the location address to where the data is going
- Data bus... carries the data from one register to another

What is a register?

- Temporary storage/memory location…inside the CPU
- Used for a single specific purpose
- Faster access speed than RAM / secondary storage



RAM, ROM and Virtual Memory

Describe the differences between types of primary memory and explain their uses

Random Access Memory (RAM)	Read Only Memory (ROM)
• Volatile, i.e. it loses its data when power is re-moved. • Read/Write **Type of Software:** • Applications software/Operating System/User files Reason: • Allows changes to be made to saved contents/files in current use/fast access to data	• Non-volatile i.e. it does not lose its data when power is switched off. • Read only **Type of Software:** Bootstrap program Reason: Program is required immediately when the power is switched on, therefore must still be there

An embedded processor has control systems

Explain why the control software is stored on ROM and explain why it will be necessary to have some RAM.

- The control software will not need to be changed
- Cannot be changed
- Will not need loading/installing
- Immediately available when switched on

ROM and the start up process

- Contains the computer start up instructions
- Loads **settings/configuration** (CMOS/NVRAM)
- Initialises/checks hardware/peripheral devices are available/work // carry out a POST check...
- ...and **reports** errors
- Determines the drive on which the OS is stored
- Loads the bootstrap/operating system (into main memory)

What is Virtual Memory?

- If the amount of RAM available is insufficient…
- …Used to store instructions/data
- …Using secondary storage.

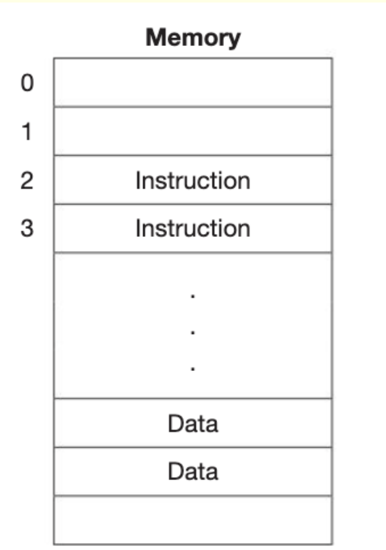
Drawbacks

Slower access speeds (as it has to move sections of the program from the secondary storage to RAM.)

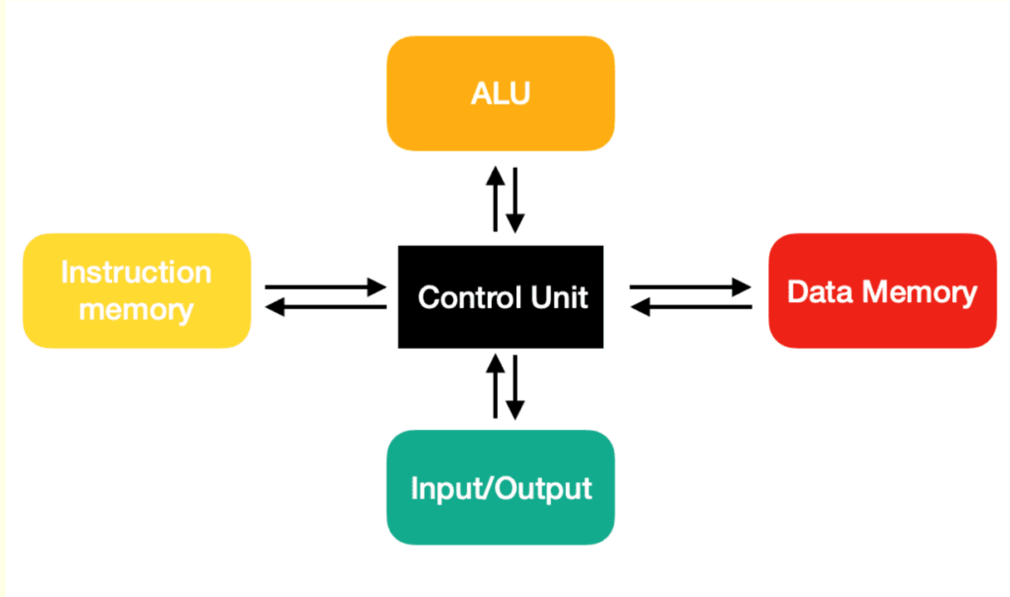
Disk thrashing - happens when a computer's **RAM (main memory)** is too full, and the system has to keep **swapping data between RAM and the hard drive (virtual memory)** too often.

Features of Von Neumann Architecture compared with Harvard architecture

Von Neumann Architecture	**Harvard architecture**
Instructions and Data stored in same area of memory	**Harvard** stores data and instructions in separate memory units
Von Neumann fetches data and instructions sequentially/follows FDE cycle	**Harvard** can fetch data and instructions at the same time (fetching the next instruction while reading/writing data.)
Von Neumann uses a single bus for both data and instructions	Separate control buses
Single control bus	**Harvard** uses different buses for data and instructions
Slower due to the **Von Neumann bottleneck** (using the same bus)	Faster because data and instructions are accessed **in parallel**.
Used in **general-purpose computers** (e.g. PCs, laptops).	Common in **embedded systems** and **microcontrollers.** Digital Signal Processing (DSP) systems that require fast access to data and instructions
Canbe optimised by using pipelining	Can use **different word lengths** for data and instructions, (optimising memory use)
Same word length for both data and instructions	



The stored program concept



Must not use: "instructions and data are stored in the same memory location"

Examiner Tips and Tricks

You will not be asked about specific aspects of "contemporary processor architecture" apart from those on this page. You may be asked to show an awareness of how contemporary processors differ from a pure Von Neumann architecture in more open questions

Features of contemporary processors

- Two separate areas of memory…one for instructions & one for data./instructions and data can be accessed **concurrently.**
- Different (sets of) buses…one for instructions & one for data./instructions and data can be accessed concurrently.
- **Pipelining**…whilst an instruction is being executed the next can be decoded and the subsequent one fetched.
- **Use of Cache**…A small amount of high performance memory is (next to the CPU) / which stores frequently used data/instructions
- **Virtual cores/Hyper-threading** …Treating a physical core as two virtual cores.
- **Multiple Cores**…Each core acts as a separate processing unit.
- **Onboard Graphics**…Built in circuitry for graphics processing.

Modern processors are Hybrids that combines features of both **Von Neumann** and **Harvard** designs. The **main memory (RAM)** stores both data and instructions together (like Von Neumann). - Inside the CPU, there are **separate caches** for data and instructions (like Harvard). This allows **simultaneous access** to data and instructions for higher performance

Contemporary Processors - CISC vs RISC

Example:
Multiply value in memory location "X" by value in memory location "Y"; store result back into location "X". Registers "A" and "B" are available.

CISC Assembly:	**RISC Assembly:**
IMUL X, Y	LOAD A, X LOAD B, Y PROD A, B STORE X, A

RISC	**CISC**
Smaller instruction set	Larger instruction set
Requires less complex hardware/ requires little cooling, minimising manufacture cost/ less transistors	Requires more complex hardware/ requires cooling, more expensive to manufacture/ more transistors
One clock cycle to execute an instruction	Multiple clock cycles to execute an instruction
Tends to use less energy	Tends to use more energy
Uses more RAM	Uses less RAM
Easier to pipeline	Difficult to pipeline
But compiler has to do more work to translate the code into machine code	Compiler has to do less work to translate the code into machine code
Fewer addressing modes (drawback)	More addressing modes
Applications requiring **high-speed processing** and **efficiency**, such as embedded systems and mobile devices, due to its **faster, simpler** instructions.	Preferred in general-purpose computers where **ease of programming** are important, as complex instructions can make the software simpler.

CPU Performance

What effects CPU performance and why?

Cores

Number of cores has an impact…Each core is a processing unit…Giving the potential for multiple instructions to be run simultaneously.
Depending on the situation 4 cores running at 100MHz may perform better than 1 core running at 300MHz.

Cache

The amount of cache (and levels) will benefit performance… Cache helps reduce the bottleneck caused by RAM being slow.
No matter how fast the clock speed, the access time to RAM will always be a limiting factor.

Contemporary processors

Contemporary processors have performance enhancing features such as pipelining and out of order execution.
Harvard architecture processors benefit from having separate data and instruction memories.

In conclusion one cannot judge performance solely on clock speed as...

A processor without cache may be outperformed by a processor with a slower clock speed but access to cache.

Processors will have other performance enhancements such as pipelining.

Pipelining is a technique used by some processors to improve performance.
Without pipelining, the steps in the Fetch-Execute cycle take place one after the other.

- Pipelining allows the next instruction to be fetched **whilst** the previous one is being decoded/executed
- It allows the overlapping of different parts of the FDE
- It increases **throughput** // increases the number of instructions processed in a set period of time
- It prevents the CPU having to wait and prevents idle components
- Pipelining is now common in microprocessors used in personal computers. Intel's Pentium chip uses
- pipelining to execute as many as six instructions simultaneously.

	Clock cycle 1	Clock cycle 2	Clock cycle 3	Clock cycle 4	Clock cycle 5
1st instruction	Fetch	Decode	Execute		
2nd instruction		Fetch	Decode	Execute	
3rd instruction			Fetch	Decode	Execute

The performance of a computer system can be improved by adding more RAM.

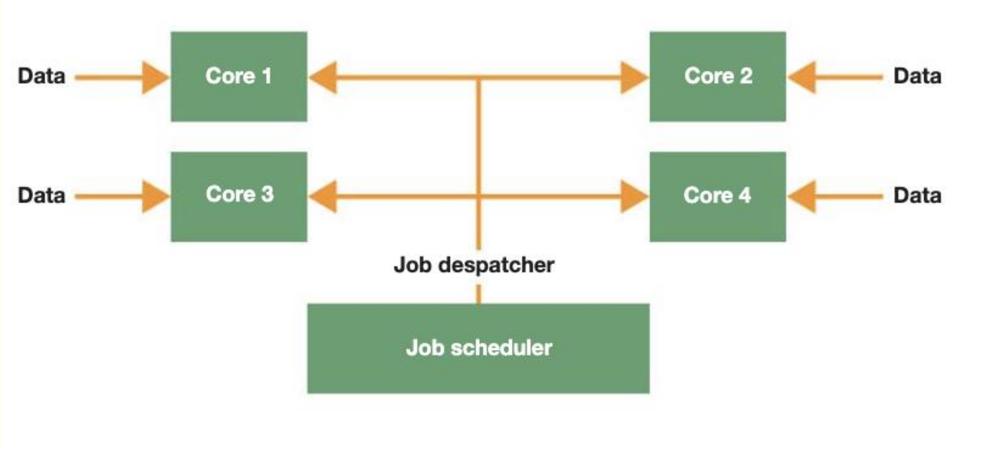
Explain why adding more RAM will improve the performance of a computer system.

- Allows more active/running/temporary data in RAM
- It reduces the need to use virtual memory
- RAM is faster to access than VM/secondary storage...
- ...because data in VM/SS has to be swapped with data in RAM first
- Use of RAM rather than VM reduces the risk of disk thrashing
- Faster bootup/ shutdown time // reduces load/access time

CPU Performance

Describe parallel processing.	**Graphics processing unit (GPU)**

- **More than one processor** controlled by a complex operating system
- Working together to perform a single job which is split into tasks
- Each task may be performed by any processor



Advantages and disadvantages of a parallel processor compared with a single processor system.

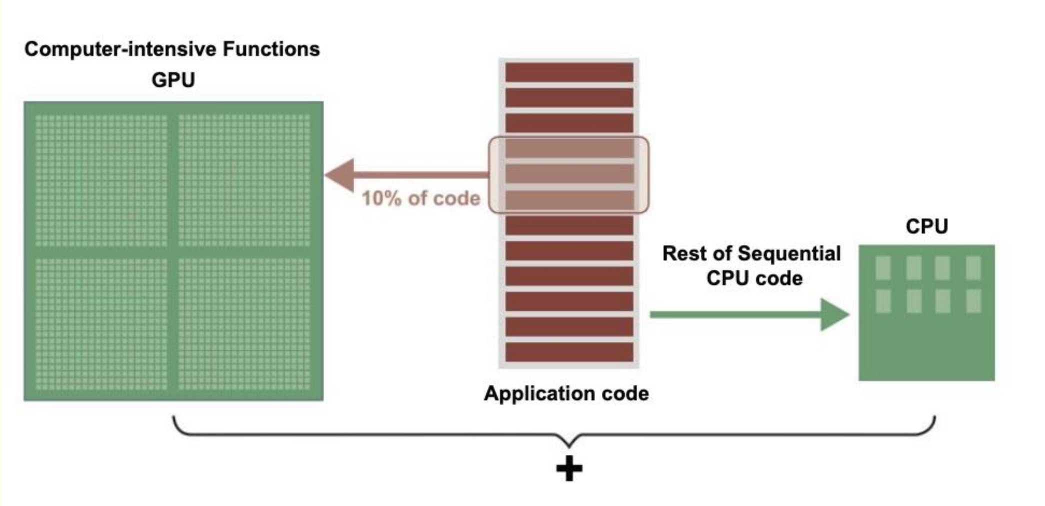
Advantages
- Increased speed/multiple instructions processed at once
- Complex tasks performed efficiently
- Allows faster processing
- More than one instruction (of a program) is processed at the same time
- Different processors can handle different tasks/parts of same job

Disadvantages
- Not suitable for some programs
- Programs written specially/may need to be rewritten
- Operating system is more complex
- ...to ensure synchronisation
- Program has to be written in a suitable format
- Program is more difficult to test/write/debug

In some computer systems, a **co-processor** may be used. Explain the term co-processor

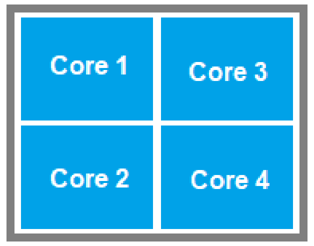
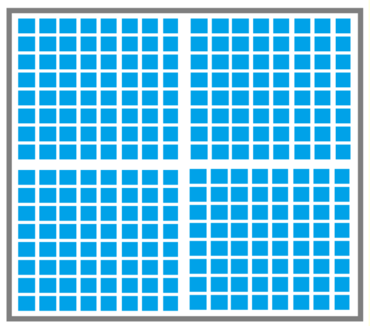
- An additional processor .used for a specific task
- Improves processing speed by executing concurrently
- Eg. maths co-processor/floating point accelerator

- **Graphics Card (GPU):**
  - Specialised for handling graphics renderingand image processing.
  - Acts as a **co-processor**, working alongside the CPU.
  - Hundreds or thousands of simpler cores designed for parallel tasks
- **Purpose in Gaming:**
  - **Creates 3-D images**: Builds wireframes, rasterizes (fills in pixels), and adds lighting, texture, and colour.
  - Handles this process **around 60 times per second** for smooth gameplay.
- **Why It's Needed:**
  - **1 million+ pixels** per screen image at common resolution.
  - Graphics processing is **too demanding for the CPU alone**.
  - Offloads intense tasks from CPU to **improve overall performance**.
- **Benefits:**
  - Faster image rendering.
  - Smoother gameplay.
  - Better visual quality and responsiveness.



Example Convert a colour image to grayscale
1. **Split the image** - The GPU divides the image into thousands of small blocks or pixels so each one can be worked on separately.
2. **Send tasks to many cores** -Each GPU core (tiny processor) gets a small piece of the image — like one pixel or a group of pixels — to process.
3. **Process pixels in parallel** -All cores run the same instructions (like adjusting color, brightness, or applying a filter) at the same time on different pixels.
4. **Combine the results** -Once all cores finish, the GPU combines all processed pixels back into a single, complete image.
5. **Display or save the image** -The final image is sent to the screen or stored in memory — ready for viewing or further processing.

CPU Vs GPU

Feature	**CPU**	**GPU**
Primary Role	General-purpose computing and system control	Specialised for graphics rendering and parallel processing
Architecture	Fewer, more powerful cores designed for sequential tasks  (Fewer strong cores)	Hundreds or thousands of simpler cores designed for parallel task.  (Thousands weaker cores)
Best For	Running operating systems, applications, and managing system operations A CPU is better at tasks that require fast switching between different instructions. Quickly process tasks that require interactivity.	Gaming, video editing, 3D rendering, and machine learning They are highly efficient at running the same instruction on many different data points at the same time. Breaks jobs into separate tasks to process simultaneously

Magnetic Storage	**Optical Storage**	**Solid State Storage**
Types: HDD (internal and external), Magnetic Tapes **How Data is Stored and read** Data is stored as magnetized regions on a spinning disk (platter). A read/write head changes the magnetic direction to represent binary data (0s and 1s). The data is read using the read/write head that moves across the spinning disk and detects the magnetic changes, converting them into binary data. **Advantages** - High storage capacity at a lower cost per GB. - Can be rewritten multiple times. - Suitable for large-scale data storage (e.g., servers, backups). **Disadvantages** - Slower than SSDs due to moving parts. - Fragile—moving parts can wear out or fail. - Can be noisy and generate heat.	Types: CDs, DVDs **How Data is Stored and read** Data is stored as tiny indentations (pits) and flat areas (lands) on a reflective disc. A laser burns the pattern onto the disc surface. A laser reads the data stored on the disc. **Advantages** - Small and light so very portable - Inexpensive for distributing media (movies, music, software). - Good for long-term archiving if stored properly. **Disadvantages** - Slower read/write speeds compared to HDDs and SSDs. - Limited storage capacity per disc (e.g., 4.7GB for a DVD). - Easily scratched or damaged, affecting data readability.	Types: SSDs, USB Flash Drives, Memory Cards **How Data is Stored and read** Solid-state storage stores data using NAND flash memory chips. Each memory cell holds an electric charge, representing binary data (0s and 1s). To read data, the storage controller sends an electric signal to a memory cell. Charge Detection: • If electrons are present, it is read as 0. • If electrons are absent, it is read as 1. **Advantages** - Very fast read/write speeds. - No moving parts—more durable and reliable. - Energy-efficient and silent. - Physically small and compact - Light and portable **Disadvantages** - Smaller capacity than hard disk drives as more expensive per GB - Limited lifespan