Unit 4: Computer Systems

The fetch-decode-execute cycle

1. The memory address held in the program counter is copied into the MAR.
2. The address in the program counter is then incremented - or increased - by one. The program counter now holds the address of the next instruction to be fetched.
3. The processor sends a signal containing the address of the instruction to be fetched along the address bus to the computer’s memory.
4. The instruction held in that memory address is sent along the data bus to the MDR.
5. The instruction held in the MDR is copied into the CIR.
6. The instruction held in the CIR is decoded and then executed. The results of processing are stored in the ACC.
7. The cycle then returns to step one.

Compilers

- Translate the whole code in one go into Machine Code.
- Optimise the code
- Used at the end of development when code is finished
- Create error reports and object code

Interpreters

- Dos not generate machine code directly.
- Calls machine code subroutines using their own code.
- Translate and execute source code
- Work line by line.
- Syntax is checked
- If code is correct it is executed
- If code is incorrect interpreting is stopped.

Assemblers

- Translate assembly language into machine code.
- Create one machine code instruction for each assembly instruction.

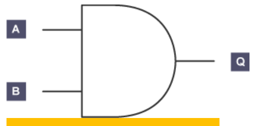
Hardware and Software

- **Hardware** is the physical components of a computer, such as the monitor, CPU, RAM or keyboard.
- **Software** is the programs and applications which run on the computer such as the operating system, games or a word processor.

Boolean Operators

AND - two conditions must be met for the statement to be true

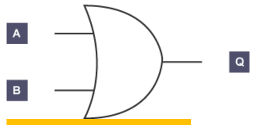
Written as AND or **.**



A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

OR - at least one condition must be met for the statement to be true

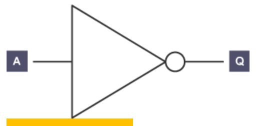
Written as OR or +



A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

NOT – inverts the result, e.g. NOT(A AND B) will only be false when both A and B are true

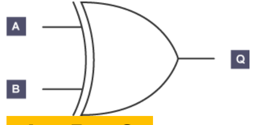
Written as NOT or ‾



A	Q
1	0
0	1

XOR – Also know as Exclusive OR. Works the same as an OR gate, but will output 1 only if one or the other and not both inputs are 1.

Written as XOR or ⊕



A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

Trace Tables

- A method of recording the values used within an algorithm at each stage of processing to help in troubleshooting
- Tests algorithms for logic errors which occur when the algorithm is executed.
- Simulates the steps of algorithm.
- Each stage is executed individually allowing inputs, outputs, variables, and processes to be checked for the correct value at each stage.
- A great way to spot errors

```
X = 3
Y = 1
while X > 0
         Y = Y + 1
         X = X - 1
print(Y)
```

Stage	X	Y	Output
1	3	1	
2		2	
3	2		
4		3	
5	1		
6		4	
7	0		
8			4

Embedded Systems

- a small computer which includes hardware and software, designed to control a specific device.
- Forms a part of a larger device such as a washing machine.
- Can perform only a limited number of tasks.

Advantages

- Cheaper to design and build.
- Require less power.
- Do not need much processing power.
- Less susceptible to viruses.

Disadvantages

- Much more limited in function.
- Adding functionality requires the system to be rewritten.
- Requires specialist skills to build and update.

High Level Programming Languages

- Much easier to learn, write and debug.
- Examples include Python, Java and C
- Code written in these languages must be translated to machine code before it can be executed.

Advantages

- Much more widely understood and used.
- Easier to learn, code in and understand.
- Much quicker to produce usable code.
- More support and learning resources are available.
- Easier to debug and find issues

Disadvantages

- Less flexible.
- Must be translated before being executed
- Very difficult to write and understand.
- Much more time consuming to produce code.

Low Level Programming Languages

Machine code

- Processors understand machine code can directly execute it.
- Each type of processor has its own specific machine code.
- Consists of 0s and 1s (binary) only.
- Very difficult to learn, write and debug.

Assembly Language

- Also known as Assembly Code
- Uses mnemonics (abbreviations)
- Easier for humans to understand and program but still difficult
- Must be translated into Machine Code for execution
- Has a 1:1 relationship with machine code
- Commonly used to program device drivers
- Often used in embedded systems

Advantages More tasks are possible using low level programming languages.

- More flexible.
- Programmers can instruct the processor directly to do whatever is needed,
- Less limited than high level languages.
- More efficient than high level languages

Disadvantages Very difficult to write and understand.

- Much more time consuming to produce code.
- Much harder to debug and fix problems.
- Machine code is specific to a single processor, so code which works on one processor will not work on another

Operating Systems

- Examples include Microsoft Windows, macOS, Linux, Android, iOS
- The Operating system provides a foundation for the user to interact with the computer and for other applications to run.
- It handles management of key hardware and functions including processors, memory, input/output (I/O) devices, applications and security.

<u>File management</u>

- Allows users to find and manage data stored by the computer.
- Data is stored in files, within folders, within drives.
- Uses a virtual file structure of the physical components.
- Assigns metadata to files including date created, date modified, last date accessed

<u>Process management</u>

- Allows users to run applications such as web browsers or word processors.
- Multiprogramming enables several programs to run at the same time.
- Each program is made up of program instructions. When these instructions are running, they are called a process.
- Allocates use of the main memory and the CPU between processes.
- A scheduler is used to time the different processes.

<u>Peripheral Management</u>

- Manages input and output between peripherals and a process.
- Data is transferred between input devices, the CPU and main memory, and output devices.
- Uses device drivers to communicate with devices.

<u>User Management</u>

- Individual users can be created and deleted.
- Allows more than one person to use a computer with their own files and settings.
- Access levels control user access to systems for security.
- A log is kept of files a user creates, accesses, edits and deletes

Cloud Storage

- Remote storage over The Internet.
- Files are stored on a server hosted by a cloud storage provider in a secure datacentre.
- The provider will store files on magnetic or solid state storage.
- Users do not know physically where their data is stored.
- Some providers offer the option to choose a specific country or continent to store data in.
- Files can be uploaded and downloaded as required from anywhere with internet access.

Advantages

- Provides the ability to access files from any location or device with Internet access.
- Files can easily be shared with other users.
- Cloud storage services often back up data for their users.
- Provides increased security as data is stored in high security data centres.
- Allows small devices such as phones or tablets to access huge amounts of data which it would not be practical to store directly on them.
- More storage can be easily added without having to buy and install more hardware.

Disadvantages

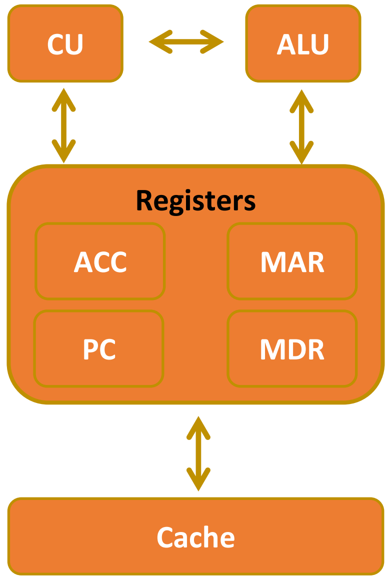
- Can be targeted by hackers.
- Data is stored by someone else, so the owner has less control.
- Data can only be accessed in locations with a working internet connection.

Von Neumann Architecture

- Data and instructions are stored in binary.
- Data and instructions are stored together in **RAM**.
- Instructions are fetched from **RAM** one at a time in order
- The **CPU** decodes and executes an instruction, before fetching the next instruction
- The cycle continues until no more instructions are available

A CPU using Von Neumann architecture have five special registers

- **Program counter** - holds the memory address of the next instruction to be fetched.
- **Memory address register (MAR)** - holds the address of the current instruction.
- **Memory data register (MDR)** - holds the content at the address held in the MAR.
- **Current instruction register (CIR)** - holds the instruction that is currently being decoded and executed
- **Accumulator (ACC)** - holds the results of processing



RAM

- Random Access Memory
- Volatile memory – contents is lost when the computer is turned off.
- Called random access because data can be directly written to or read from any location.
- Used to hold data and instructions that are in use.
- The more RAM a computer has, the more data it can hold simultaneously.

RAM

- Read Only Memory
- Non-volatile main memory – contents are not lost when the computer is turned off.
- Can be read from, but not written to.
- Ideal for storing instructions and data that are needed for the computer to run.
- Usually programmed by the computer's manufacturer and cannot be overwritten.
- The Basic Input Output System (BIOS) is an example of a program stored in ROM.

Unit 4: Computer Systems

CPU – Central Processing Unit

- Processes data and instructions
- Controls the rest of the computer system

Different things affect CPU Performance:

- Clock Speed – The faster the clock speed the quicker the CPU can execute instructions and so the better the performance.
- Number of Processor Cores – The more cores a CPU has, the more executions it can execute at the same time and so the better the performance.
- Cache Size – The greater the cache available to the CPU, the more results it can store. This means less instructions need to be repeated and so improves performance.

Control Unit (CU)

- Fetches, decodes, and manages the execution of instructions
- Issues control signals to control hardware
- Moves data around the system

Arithmetic Logic Unit (ALU)

- Performs arithmetic and logical operations.
- Where calculations are done and where decisions are made.

Registers

- Small amounts of high speed memory in the CPU.
- Used to store small amounts of data that are needed during processing.

Cache

- A small amount of high speed memory In the CPU.
- Used to temporarily hold data the CPU will reuse.
- Allows for faster processing since as the CPU need not wait for data to be fetched from RAM.

Clock

- Used to coordinate all the computer's components.
- Sends out a regular electrical pulse to do this.
- The frequency of the pulses = clock speed, measured in hertz.
- Higher clock speed = greater number of instructions which can be performed at a time.

Buses

- High speed internal connections.
- Used to send control signals and data between the processor and other components.
- Address bus - carries memory addresses from the CPU to other components.
- Data bus - carries data between the CPU and other components.
- Control bus - carries control signals from the CPU to other components.

Utility Software

Data Compression

- Reduces the size of a file using algorithms.
- Smaller files are easier to transmit.
- Allows more files to be stored in the same space.
- Lossless - no data is lost, and the original can be recreated.
- Lossy - some data is lost, and the original file cannot be recreated.

Defragmentation

- Files on a disk are broken down into a series of segments.
- When files are deleted, the segments where they were stored are made available for new files.
- The new file may need more segments than the old, and so the segments allocated to it are not together on the disk. This is known as fragmentation.
- A fragmented disk takes longer to read from and write to, making the computer slower.
- Defragmentation software rearranges the segments so that they are stored next to each other.
- This decrease read/write time and improves performance.

File Repair

- Files can become corrupt due to crashes, damaged storage or a virus.
- Corrupt files can sometimes be repaired.
- Can detect and recover physical errors on the disk.
- Can scans the disk surface for damage and mark sections as unavailable.

Backups

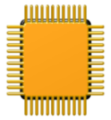
- Data can be lost accidentally or deliberately.
- A copy of data is known as a backup.
- These allow damaged or deleted data to be restored.
- Full backups include every file. This requires a lot of storage and time.
- Incremental backups include new and changed files since the last backup.

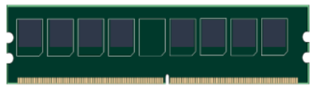
Anti-Malware

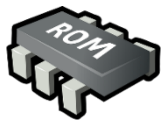
- Protects against viruses, spyware, and other unwanted software.
- Scans the system to identifies potential viruses.
- Will attempt to delete or fix potential threats once they have been identified.
- Runs either when activated or automatically at a specified date and time.

Systems Software

- Systems software manages the computer system including:
  - Controlling hardware, including peripherals
  - Allowing other programs to run
  - Providing an interface for the user to interact with the computer
  - Maintaining the computer system
- Operating Systems and Utility Software are the two main types of system software.







Secondary Storage

- Used to store programs and data for longer term when the computer is switched off
- Non-volatile – data is retained with the computer is switched off.
- Stores the operating system, software and other files needed for the computer to operate
- Stores the applications used on the computer.

Magnetic devices

- Use magnetic fields to magnetise individual sections of a spinning disk.
- Each section represents one bit.
- A read/write head moves across its surface.
- Fairly cheap, high in capacity and durable.
- Susceptible to damage if dropped.
- Vulnerable to magnetic fields.

Optical Devices

- Use a laser to scan the surface of a spinning disc.
- The disc surface is divided into tracks, with each track containing flats and hollows.
- The flat areas are known as lands and the hollows as pits.
- Lands reflects the laser light back; pits scatter the beam.
- ROM (Read Only Media) cannot be overwritten. Used for music, films, software and games.
- Read (R) media is blank, can only be written to once, but read many times.
- Read/write (RW) media can be written to more than once.

Solid State Devices

- Use flash memory to store data indefinitely.
- Have faster access times than other devices
- Because they have no moving parts, are more durable.
- More expensive so tend to be smaller in capacity.
- Require little power, so used where battery life is a consideration.
- Portable due to their small size and durability.