5 Computer Organisation and Architecture

5.25 Internal Computer Hardware

- **External components** include input/output and storage devices.
- **Internal components** are those within the CPU.
  - Processor
  - Main memory
  - Address/control/data bus
  - I/O controllers

The Processor

The processor **responds to and processes the instructions** that drive the computer. It contains

- The **control unit** coordinate s and controls all operations carried out by the computer. It operates by repeating the **fetch-decode-execute cycle**.
- ALU performs operations on data, such as **arithmetic operations** and **logical operations** and **shift operations**.
- Registers are **special memory cells** that operate at very high speeds. All arithmetic and logical operations take place within Registers.

Buses

Each bus is a **shared transmission medium**, only one device can transmit at a time..

- **Control bus**, a bidirectional bus to **transmit command** between components, and ensure the access to and use of the data and address buses by different components **does not lead to conflict**.

  The control bus is made of **control lines**, including
  - Memory read/write
  - Interrupt request
  - Bus request/grant
  - Clock
  - Reset
- **Data bus**, a bidirectional bus for **moving data and instructions** between components. The width of the data bus is a key factor in determining **overall system performance**.

- **Address bus** - specify an address to access a particular memory location.

  The memory is divided into **words** handled as a unit by the processor, each word in memory has its own address. The width of the address bus determines the **maximum possible memory capacity** of the system.

I/O Controller

An I/O controller is a device which **interfaces between and I/O device and the processor**. Each device has a **separate controller** which connects to the control bus.

The controller consists of

- An interface that allows connection of the controller **to the system bus**.
- A set of data, command, and status **registers**.
- An interface that enables connection of the controller **to the cable** connecting the device to the computer.

An **interface** is a standardised form of connection defining things such as signal, voltage levels, etc.

- The **von Neumann architecture** - a shared memory and bus is used for both data and instructions.
- The **stored program concept** - a program **must be in main memory** to be executed, and instructions are fetched from memory one at a time.
- The **Harvard architecture** - physically separate memories for instructions and data. It is used in **embedded systems** as instruction can use a read-only memory.

Harvard architecture is faster than von Neumann because data and instructions can be **fetched in parallel**.

5.26 The Processor

- The **ALU** perform arithmetic and logical operations on the data, as well as **shift operations** and **boolean logic operations**.
- The **control unit** controls and coordinate the activities of the CPU - directing the **flow of data** between the CPU and other devices.
  - Accepts the **next instruction**.
  - **Break down** its processing into several sequential steps.
  - Manages its **execution**.
  - **Stores** the resulting data back in memory or registers.

- The **system clock** generates a series of signals to **synchronise CPU operations**.
- **General-purpose registers** are very fast memory, all arithmetic, logical or shift operations take place in registers.

Dedicated registers include

- The **program counter** holds the address of the **next instruction** to be executed.
- The **current instruction register** holds the **current instruction** being executed.
- The **memory address register** holds the address of the memory location from which data is to be fetched or written.
- The **memory buffer register** is used to temporarily store the data read from or written to memory.
- The **status register** contains bits that are set or cleared depending on the result of an instruction.

The Fetch-Execute Cycle

This cycle is repeated over and over as each instruction of the program is executed.

1. **Fetch phase**

   The address of the next instruction is copied **from PC to MAR**, the address is **sent via address bus** to memory.

2. The instruction held at that address is returned along the data bus **to the MBR**. Simultaneously, the content of the **PC is incremented** so it holds the address of the next instruction.
3. The content of the MBR is **copied to the CIR**.
4. **Decode phase**

   The instruction held in the CIR is decoded - it is split into **opcode** and **operand**.

   - The opcode determine the **type of instruction** and what hardware to use to execute it.
   - **Additional data** is fetched if necessary and passed to the registers.

5. **Execute phase**

   The instruction is executed using the ALU if necessary, the results are stored in general purpose registers or memory.

Processor Performance

- **Number of cores**: each core is able to process a different instruction at the same time with **its own fetch-execute cycle**.

  However some software may not be able to take full advantage of multiple processors.

- **Cache** is a very small amount of expensive, very fast memory inside the CPU. An instruction fetched from main memory is copied into the cache if it is **needed again soon**.

  As cache fills up, unused data are **replaced with more recent ones**.

- **Clock speed**: all processor activities begin on a clock pulse. The greater the clock speed, the faster the instructions will be executed.

- **Word length** is the number of bits that the CPU can **process simultaneously**.

- The **width of data bus** determines how many bits can be transferred simultaneously.

  The **width of address bus** determines the **maximum memory address** that can be directly referenced.

Interrupts

An interrupt is a signal sent by a software program or a hardware device to the CPU.

- **Software interrupt** occurs when an application terminates or requests certain services from the operating system.

- **Hardware interrupt** occur when an I/O operation is complete or an error occurs.

When the CPU receives an interrupt signal

1. **Suspends execution** or running program.
2. Puts values of each register and PC onto the **system stack**.
3. An **interrupt service routine** is called to deal with the interrupt.
4. Once served, the original values of the registers are retrieved from the stack, and the fetch-execute cycle resumes.

5.27 The Processor Instruction Set

Each **different type of processor** has its own instruction set, comprising of all the instruction which are supported by its hardware.

- Data transfer e.g. load/store.

- Arithmetic operations e.g. add/subtract.
- Comparison operators compares two values.
- Logical operators e.g. and/or/not
- Branching - conditional and unconditional branching.
- Logical - bit shifts.
- Halt.

Machine Code Instruction

The number of bits allocated to the **opcode** and the **operand** will vary according to the architecture and word size of the particular processor type.

The **addressing mode** is represented by binary digits in the opcode.

- **Immediate addressing -** the operand is the **actual value** to be operated on.
- **Directed addressing -** the operand holds the **memory address** of the value to be operated on.

The # symbol signifies that the **immediate addressing mode** is being used.

Assembly Language Instructions

Assembly language use **mnemonics instead of binary codes**. Each assembly language instruction translates into **one machine code instruction**.

5.28 Assembly Language

Assembly languages uses mnemonics to represent **operation code and addresses**. The **assembler** translates an assembly language program into machine code for execution.

A **logical shift right** causes the least significant bit to be shifted into the **carry bit**, and a zero moves in to occupy the vacated space.

5.29 Input-output Devices

Barcodes

Barcodes are used for

- Tracking parcels.
- Sale of items in shops.
- Record the details of people attending events.

Linear barcode are 1D, 2D barcodes such as the **Quick Response** code can hold more information than the 1D barcode.

2D barcodes are used for

- **Ticketless entry** to concerts.
- To provide a **website URL**.

Barcode Readers

- **Pen-type reader** - a **light source** and a **photo diode** are placed next to each other in the tip of the pen.

  1. The tip of the pen is **dragged across all the bars** at even speed.
  2. The photo diode **measures the intensity** of the light reflected back from the light source.
  3. Dark bars absorb light and white spaces reflect light, so the voltage waveform generated by the photo diode can be used to measure the widths of the bars and spaces in the barcode.
  4. The signal is converted from analogue to digital.

  A simple encoding translates areas of light and dark as 1s and 0s, these can be used to create **ASCII character codes** for a string which could be a product code.

  Pen-type readers are suited for use with portable computers or very **low volume scanning applications**.

  - **Durable** and can be sealed against dust, dirt and other environmental hazards.
  - **Small size** and **low weight**.
  - Applications are limited because they **must come into direct contact** with the barcode to read it.

- **Laser scanners** use a laser beam as the light source. The laser reflects off a **moving mirror** which allows the barcode to be read in many different positions.

  - **Reliable** and economical for **low-volume** applications.
  - Used as **in-counter** units in supermarkets.

- **Carged-coupled device (CCD) readers** use an array of **hundreds of tiny light sensors** lined up in a row at the head of the reader.

  1. Each sensor measures the intensity of the light immediately in front of it.

2. A **voltage pattern** identical to the pattern in a barcode is generated in the reader by sequentially measuring the voltages across each sensor in the row.

- **Camera-based imaging scanner** uses a camera and **image processing** techniques to decode a 1D or 2D barcode.

  An imaging scanner can read barcode

  - On any surface - printed or onscreen.
  - Damaged or poorly printed.

  Image processing has to be carried out by the software as the barcode might be in **any rotation or distance** from the scanner.

  - Event ticketing - electronic tickets are scanned off a phone screen.
  - Using cell phone to scan a QR code which display information about a product.

Digital Cameras

Uses a **CCD** or **CMOS** sensor comprising of millions of tiny light sensors arranged in a grid.

1. When the **shutter opens**, light enters the camera and **projects an image** onto the sensor at the back of the lens.
2. Each sensor **measures the brightness** of each pixel - turns light into electricity, and stores the amount of charge as binary data.

The binary data is recorded onto the camera's **memory card**, so the image can be reproduced using suitable software.

3. To record colour, the sensors are placed under a **mosaic of red, green and blue filters** to separate out the different colour wavelengths.
4. The processor can then **approximate binary values** for the three RGB channels of each individual pixel based on the value of neighbouring pixels.

- **CCD sensor** produce **higher quality images** and are used in high end cameras, and are **more reliable**.
- **CMOS sensors** consume 100 times less power than a CCD sensor.

Radio Frequency Identification

RFID tags are used to track household products, cars, bank cards and animals.

- Can be read **without line of sight** and up to 300 metres away.
- Can **pass stored data** from the tag to the receiver and vice versa.

RFID chip consists of a **small microchip transponder** and an antenna to communicate with the **base unit**.

- **Active tags** are physically large as they **include a battery** to power the tag.

  It **actively transmits a signal** for a reader to pick up. And is used to track things to be read from further away, e.g. cars as they pass through a toll booth.

- **Passive tags** do not have a battery. They rely on **radio waves emitted from a reader** to provide electromagnetic power to the card using its **coiled antenna**.

  Once energised, the transponder inside the RFID tag can send its data to the reader nearby. E.g. tagging grocery items and smart cards.

Laser Printer

Laser printers offer **high-quality, high-speed printing**. It uses **powdered ink** called **toner**.

- Generates a bitmap image of the printed page.
- Use a laser unit and mirror to **draw a negative** onto a negatively charged drum, causing the affected area of the drum to lose their charge.
- The drum rotates pass a toner hopper to **attract charged toner particles** onto the areas which have not been lasered.
- The particles are **transferred onto a sheet** of paper then bonded onto it using pressure and heat.

Coloured laser printers contains four toner cartridges, and the paper must go through a similar process to the back-only printer four times - once for each colour.

- Quality is limited.
- Photorealistic prints is impossible (use inkjet printers instead).