A-level
Computer Science

Computer architecture

Lesson Objectives

Students will learn about:

- Various components of a computer such as memory and processor.
- How are these components connected?
- How data is accessed from memory? How are contents copied to memory?
- How are programs executed?
- Interrupts

1.

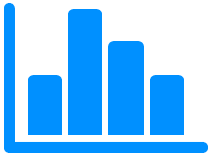
Content

Introduction

- Very early computers could not store programs and, hence, in 1945, John von Neumann developed an idea of a computer that could store programs, which is also known as von Neumann architecture.
- He suggested that data and programs could be stored in memory and that the data would be transferred between the memory and the processor during processing.

Central Processing Unit (CPU)

- CPU is responsible for the processing of the instructions given to a computer.
- It consists of arithmetic logic unit, control unit and cache memory.
- The hardware of CPU is made of billions of transistors that form logic gates.



4 000 000 000

A 4 GHz processor processes these many instructions per second.

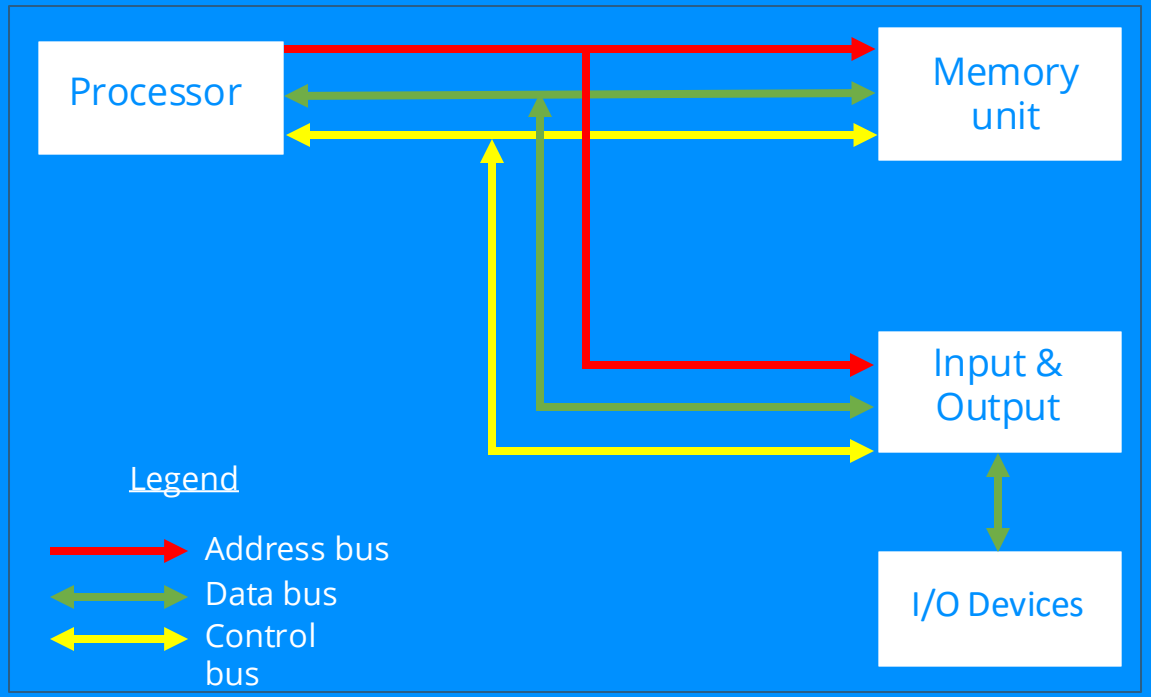
It would appear that we have reached the limits of what it is possible to achieve with computer technology, although one should be careful with such statements, as they tend to sound pretty silly in 5 years.

-John von Neumann

von Neumann architecture

- von Neumann architecture shows the components of a computer in detail.
- It shows the connection between the processor, memory and input-output devices. These connections are called buses.
- Three different kinds of buses are the address bus, control bus and data bus.

von Neumann architecture



Buses

Type of bus	Function	Direction of bus
Address bus	Transmits the address from the processor to memory or I/O controller	Unidirectional: From the processor to memory and input-output devices
Data bus	Sends data between the processor, memory and input-output devices	Bidirectional
Control bus	Signals sent by processor to control the memory and peripheral devices	Bidirectional: From the processor to memory and input-output devices

Address bus

- Memory is divided into several fixed segments called words.
- The words may be of 16, 32 or 64 bits depending on the processor type.
- Address bus is a unidirectional bus that transmits the address from the processor to memory and input-output devices.
- In a program, the addresses of operands are sent via the address bus.
- The results calculated by the processor are stored in a particular memory location, the address of which is transmitted via address bus.

Data bus

- A bidirectional bus consisting of 8, 16, 32 and 64 parallel lines.
- This bus is used to transmit data and instructions between the processor, memory and input-output devices.

Control bus

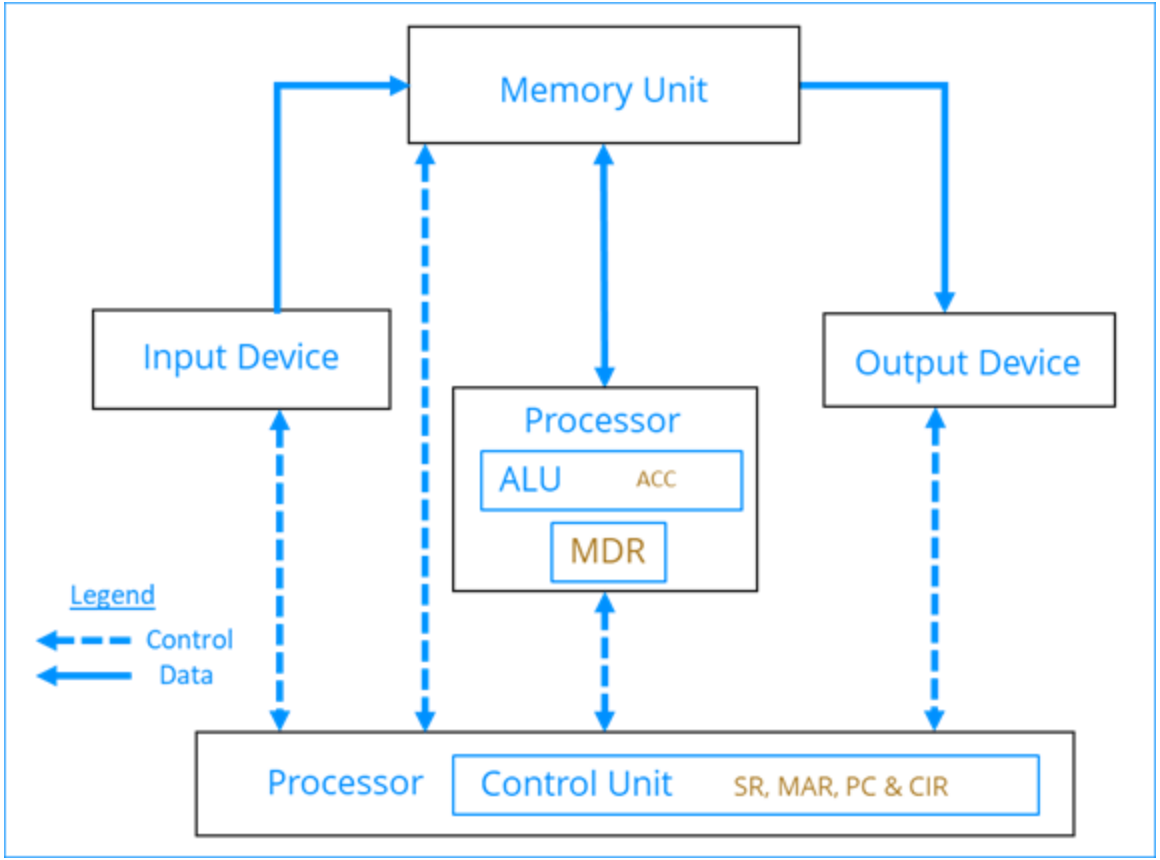
- A bidirectional bus that transmits timing, status signals and commands between the processor, memory and I/O devices.
- The different control signals are:
  - Clock: Synchronises the operations on a computer.
  - Memory Read: The contents in the specified address is copied to the data bus.
  - Memory Write: The contents in the data bus is copied to the specified address.

Control bus

- The different control signals are:
  - Bus Request: A device requests to use the data bus so that it can perform the Read/Write operation.
  - Bus Grant: The signal from the processor indicating that the device is granted access to use the data bus.
  - Interrupt Request: A signal sent from a device or software to the processor which temporarily stops the current process and initiates an interrupt service routine. An interrupt request is a signal sent by the device requesting access to the processor.

Detailed von Neumann architecture

MAR, MDR, ACC, PC, SR and CIR shown are registers. Addresses are stored in the memory unit.



Registers

- Program Counter: Holds the address of the next instruction to be executed.
- Current Instruction Register: Holds the current instruction being executed.
- Memory Address Register: Holds the address of the register from which data is read, or to which data is to be written.
- Memory Data Register: Temporarily stores the data being read from or written to the memory.
- Status Register: Holds the bits that are set or cleared based on the result of an instruction. For example, overflow in case of addition, carry in case of addition, etc.

Memory Unit

- A memory unit consists of a number of partitions.
- Each partition consists of data and an address.
- The address uniquely identifies every location in memory and the contents are in binary form.
- An example of data and addresses stored in a memory unit is shown.

Address	Contents
11010000	10010001
11010001	11110101
11010010	11100011
11010011	10101010
.	.
.	.
.	.
11011110	11011011
11011111	11001100

READ operation

- To read the contents from any location, the registers MAR (Memory Address Register) and MDR (Memory Data Register) are used. To read the contents from the address 11010000, the address is copied to MAR.

Register	Contents
MAR	1 1 0 1 0 0 0 0

- A read signal is sent by the processor to the memory. The contents are then copied to the MDR (Memory Data Register).

Register	Contents
MDR	1 0 0 1 0 0 0 1

WRITE operation

- To write data into a particular address, the data is first written to the MDR. Let us consider writing the data 10111011 to the address 11101110.

Register	Contents							
MDR	1	0	1	1	1	0	1	1

- The address is written to the MAR register.

Register	Contents							
MAR	1	1	0	1	1	1	1	0

- A write signal is sent by the processor to the memory unit through the control bus and the location is written with the data in MDR.

Processor

- The processor contains ALU (Arithmetic and Logic Unit) and Control Unit (CU).

Processor: ALU

- The ALU is responsible for arithmetic functions such as addition, subtraction, multiplication, etc. and logic operations such as AND, OR, NOT, etc.
- The Accumulator (ACC) is the register that stores the result of arithmetic and logical operations performed by a processor.

Processor: CU

- The control unit (CU) is responsible for controlling the memory, processor and input-output devices.
- It contains the CIR (Current Instruction Register) and PC (Program Counter).
- The CIR contains the current instruction carried out by the processor.
- The PC contains the location of the instruction that is to be executed next.
- The control unit reads the instructions from the memory, decodes it and sends control signals to the memory and input-output devices.

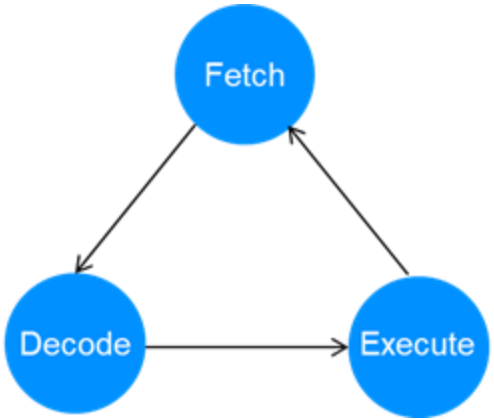
Input-Output Devices

- An input device like a keyboard or mouse converts the signals sent by humans into a form that can be understood by the computer.
- Output devices such as printers, monitors, etc. convert the information from the computer into a form that is understandable by humans.

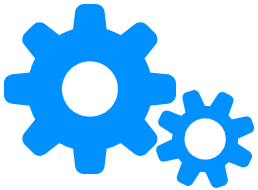
I/O controller

- I/O controller is a device that interfaces an input or output device with the processor.
- Each device has its own controller which is connected to a control bus. I/O controller is responsible for receiving the requests from the processor and sending control signals to the device specified for that operation.
- It consists of:
  - An interface to connect it to the system or I/O bus
  - a set of data, command and status registers
  - an interface that connects it to the cable connecting the device and the processor.

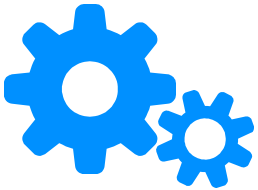
FETCH – DECODE – EXECUTE CYCLE



PC	The program counter (PC) contains the allocation of the instruction which has to be fetched
MAR	Using the address bus, the contents of PC is copied to MAR
MDR	The instruction at that particular location is copied to MDR temporarily
CIR	The contents of the MDR is copied to the current instruction register (CIR)
+1	The value in the PC is then incremented by 1 so that next instruction can be fetched
Execution	The instruction is decoded and then executed by sending out control signals to the memory and input-output devices

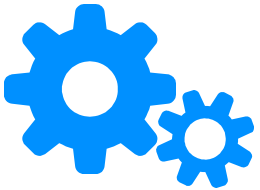


The fetch-decode-execute cycle



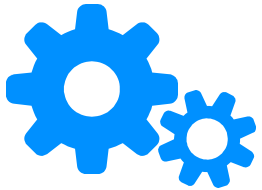
- To carry out the instructions, the processor fetches the data and instructions from the memory and stores it in suitable registers.
- These instructions are decoded and then executed.

The fetch-decode-execute cycle: Fetch



- The program counter contains the location of the instruction that is to be executed next. This address is copied to MAR.
- The instruction is fetched from the memory and copied to MDR first.
- Then, the contents of MDR is copied to the CIR (Current Instruction Register).
- The value in a program counter is incremented by 1 and, hence, the instruction in the next memory location is processed.

The fetch-decode-execute cycle: Decode and execute



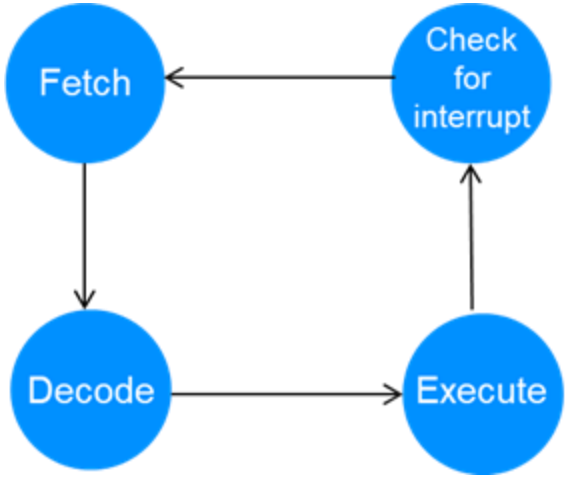
- **Decode:** The instructions are decoded so that it can be executed.
- **Execute:**
  - The processor sends appropriate control signals to the memory unit and input-output devices in the computer system according to the decoded instruction.

What is an interrupt?

- An interrupt is a signal sent from a device or software to the processor.
- The processor will temporarily stop its current process and will service the interrupt signal.
- For example: When paper is jammed in a printer, the CPU prompts the user to check the status.

Interrupts

- Interrupts allow the computer to carry out many tasks at the same time.
- When the interrupt is serviced, the status of the current job is saved. The contents of registers PC and CIR are saved on to the system stack.
- Once the interrupt is serviced using the interrupt service routine, the current job is serviced according to its status when it was saved before the interrupt service.



Vectored interrupt mechanism

- Each interrupt is associated with a vector, which points to the code associated with that interrupt.
- When an interrupt occurs, the current values of the registers are saved to a stack and the processor identifies the type of interrupt.
- Then, the processor points to the vector and processes the interrupt service routine. This technique is called vectored interrupt mechanism.

Interrupts

- Sometimes, when an interrupt is being serviced, another interrupt may occur.
- The processor may save the status of the current interrupt processing and proceed to service the new interrupt.
- Another methodology that can be used is to use priorities. Based on the priority of the interrupts, it can be decided whether the current interrupt service should be paused or not.
- For example, an interrupt that notifies the user that the battery of a laptop is running low must be given the highest priority.

Let’s review some concepts

Buses

The connection between the processor, memory and input-output devices.

Register

Registers are high-speed data storage areas in the computer.

CIR (Current Instruction register)

The CIR contains the current instruction carried out by the processor.

PC (Program Counter)

The PC contains the location of the instruction that is to be executed next.

Accumulator

Accumulator (ACC) is the register that stores the result of arithmetic and logical operations performed by the processor.

Processor

The processor contains ALU (Arithmetic and Logic unit) and Control Unit (CU).

2.

Activities

Activity-1 (Internet research)

Duration: 15 minutes

1. 8085 is an Intel microprocessor. Use the internet and analyse the architecture of 8085 microprocessor to answer the following questions.

A. What registers are used to carry memory locations while processing instructions? Also, state its function.

B. What is the function of a flag register? What does each bit in this register denote? State the function of each flag bit.

Activity-2

Duration: 10 minutes

1. A set of instructions in assembly language is given below with its memory location.

A. What is the function of this code?

B. Consider any instruction and in the space below describe its fetch-decode and execute cycle. Use appropriate registers in your explanation.

Address	Opcode	Instruction
2000	A1 22	MOV AX, 22h
2002	33 15	SUB AX, 15h
2004	F4	HLT

Activity-3

Duration: 15 minutes

1. Data stored in a part of the memory unit is given in the table.

A. In the box, explain how the content at the memory location 9D can be read?

B. In the box, explain how the data 76 can be written to memory location 99?

Address	Contents
98	78
99	65
9A	81
9B	2F
9C	3E
9D	19
9E	D4

3.

End of topic questions

End of topic questions

1. What are the three types of bus in computer architecture? Specify the function of each bus.
2. What are the different control signals transmitted via control bus? State the function of each.
3. How are the memory locations addressed?
4. What is the function of ALU?
5. How does the control unit control the memory and peripheral devices?

End of topic questions

6. List the different registers.
   a) Which of these are used to hold data and instructions?
   b) Which of these are used to hold addresses?
7. Explain what happens in the fetch phase of the fetch-decode-execute cycle.
8. What is vectored interrupt mechanism?