

PROCESSOR FUNDAMENTALS

- This is unit 1.4 of the syllabus, and chapter 4 of Sir Zain's notes.

- It contains the following subtopics:

1. CPU architecture → (1.4.1 and 1.4.2 of the 2021 syllabus)

✓ Details about the Von Neumann Architecture

✓ Details about register [Special vs. General Purpose Registers]

✓ Components of a CPU → their effect on performance of CPU

- Different ports

- Fetch Execute cycle + interrupts

2. Assembly Language → (1.4.3 and 1.4.4 of the 2021 syllabus)

- Relationship bw. machine code and assembly language

- Two Pass Assembly process

- Tracing assembly language programs

- Processor's Instruction Set: Instruction groups

↳ Modes of addressing

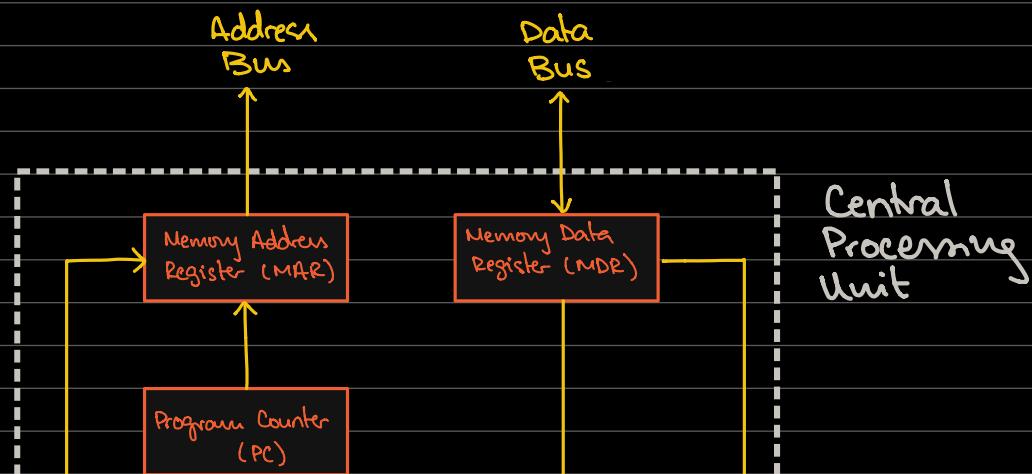
- Bitwise operations

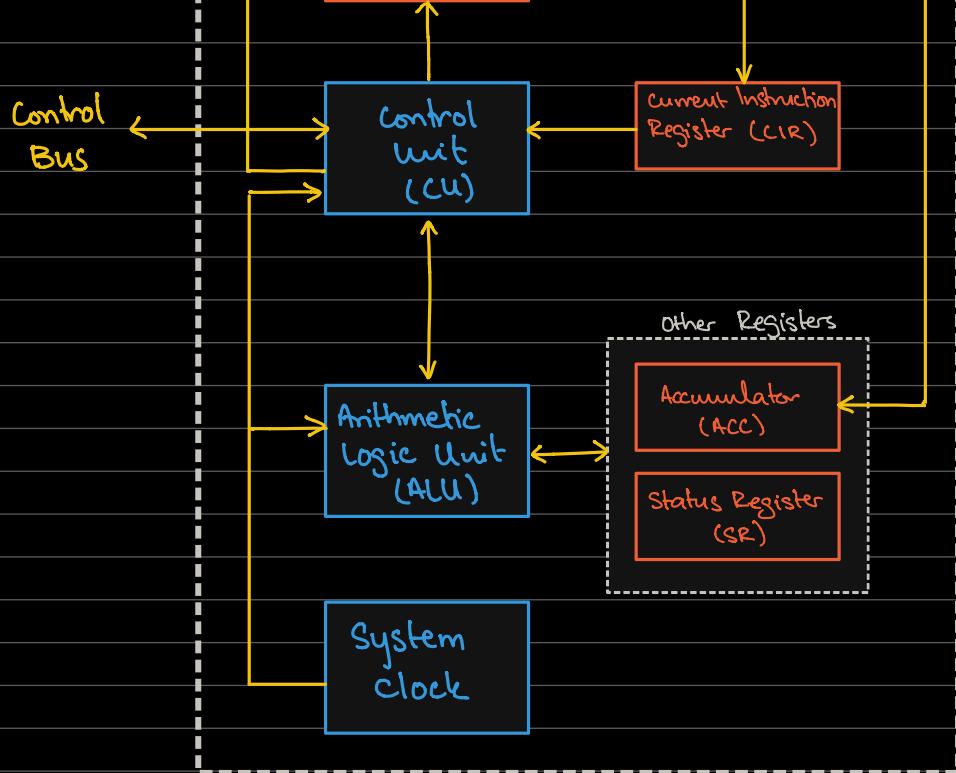
PROCESSOR FUNDAMENTALS: VON NEUMANN & THE CPU

Main features of the Von Neumann Model of the computer:

- Within the computer, there is a processor (CPU / Central Processing Unit)
- The processor has direct access to a memory (RAM / ROM)
- The computer "memory" stores programs as well as the data required for those programs.
- The stored program is made up of multiple instructions that are executed sequentially.

Architecture of a simple CPU





The Components of the CPU and their roles:

1. ALU (Arithmetic Logic Unit).

- As its name implies, the ALU is responsible for carrying out any arithmetic ($+$, $-$, \times , \div) or logical ($<$, $>$, $=$) operations that may be required by the current instruction.
- A computer may have more than one ALU, for example, one to perform fixed point operations and the other for floating point operations.
- When the ALU's carrying out calculations, it temporarily stores data in the Accumulator (ACC).

2. CU (Control Unit)

- The control unit is responsible for controlling the flow of data throughout the processor as well as making sure that the instruction is handled and executed correctly.
 - ↳ "The CU ensures synchronization of data flow and program instructions throughout the computer."

3. System Clock

- The system clock works in tandem with the Control Unit to ensure that the different components of the processor are operating in a synchronised manner
 - ↓
- The system clock produces timing signals on the control bus in order to ensure perfect synchronicity.

- The system clock has a certain frequency / clock cycle, that can be defined as the minimum period of time that separates successive activities within the processor.

4. Immediate Access Store

- A temporary storage that contains the data and programs that the CPU needs to access.
- The CPU fetches this data and these programs from the secondary storage / backing store and stores them in the IAS because read/write operations to and from the IAS are faster than those from the backing store.
- IAS may also be referred to as primary (RAM) memory.

Registers within the CPU :

- The concept of registers, ie. temporary data stores, is a key one in the Von Neumann Model.
- Registers are formally defined as storage components that, due to their close proximity to the ALU, are ideal for accessing in minimal time.
- Each register has a limited memory, typically 16, 32, or 64 bits
- Registers can be either general purpose or special purpose
 - An example of a general purpose register is the ACC (Accumulator), which is the only one we'll be concerned about through our syllabus
 - Examples of special purpose registers include:
 - 1. Index Register → discussed later during Assembly Lang.
 - 2. Status Register
 - ↳ Used by the ALU when an instruction requires some sort of arithmetic or logical processing
 - A status register has several bits, with each bit signifying a different "state" of data
 - For the 4 most common bits/flags and their usages, refer to Sir Zain's notes
 - 3. Current Instruction Register (CIR)
 - Stores the current instruction while it is being decoded and executed

4. Memory Address Register (MAR)

- Stores the address of a memory location which is about to have a value read from or written to

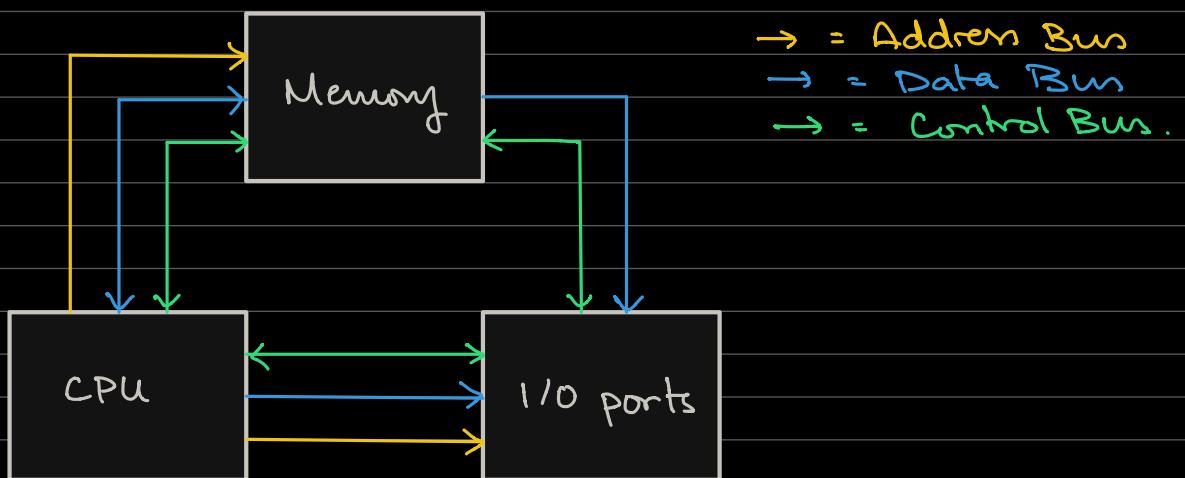
5. Memory Data Register

- Stores data that has just been read from memory or is about to be written to memory

6. Program Counter (PC)

- Stores the address of where the next instruction is to be read from

System Buses



- Buses in the Von Neumann model are used as parallel transmission components and consist of multiple wires / channels
 - ↳ Each wire / channel transmits one bit of data
- Three main buses: address bus, data bus, control bus.

Control Bus

- The control bus is a bi-directional bus that connects the CPU, memory, and the I/O ports to each other.
- This bus is used for carrying control signals / timing signals from the CPU / System clock to the other components, thereby ensuring the correct flow of data as well as the synchronous operation of the several components.
- It is typically only 8 bits / channels wide since it carries only control signals.

Address Bus

- The address bus is unidirectional between the CPU and the memory and between the CPU and the I/O ports

- This bus is responsible for carrying addresser throughout the system / to different components when required
- It is particularly important that this bus is unidirectional from the CPU to the memory, as addresser being carried back to the CPU would be undesirable.
- The width of the address bus is also of particular importance, as the wider the address bus, the more addresser that are directly accessible by the CPU.

↳ For example, an address bus that's 3 channels wide can access a maximum of 2^3 (8) memory addresser whereas...

an address bus that's 4 channels wide can access a maximum of 2^4 (16) memory addresser.

Data Bus

- The data bus is bidirectional between all three components.
- It is responsible for transmitting data to and from different components in the CPU
- The width of the data bus is also important
 - ↳ The wider the address bus, the larger the word length that can be transmitted at once.

[A word is a group of bits which can be regarded as a single unit, 32-bit and 64-bit systems are most common]
- A larger word size / A wider data bus can improve the computer's overall performance

Components and their impacts on computer performance :

System Clock

- The clock speed of a system determines the minimum time interval between processing and execution of consecutive instructions.
- By increasing the clock speed and the clock cycles / frequency, we are able to process and execute more instructions within the same time period → this is known as "over-clocking".

↳ As a result, the processing speed of the system increases, however, we cannot conclude that the overall performance of the system improves solely based on this.

Address / Data Bus

- As mentioned previously, the widths of the address bus and the data bus are also important in deciding a computer's

performance, as they control how many addresses are directly accessible, and how many bits of data can be transmitted at once.

Cache memory

- Cache memory is similar to RAM in that it stores the currently required data and instructions
- When a processor reads the memory, it first checks the cache memory and then the RAM.
- The benefit of using cache memory is that since it uses SRAM as compared to the DRAM that's used by RAM memory, it has a faster access time and can thus improve computer performance.

Number of cores

- Another way to improve computer performance is to increase the number of cores
 - ↳ A single core is made up of a CU, an ALU, and all the standard registers
- However, the increase in performance is not proportional to the increase in performance.
 - ↳ For example, doubling the number of cores from 2 to 4 does not mean that the performance will also double.
- This is because we need to consider the different cores now having to communicate with each other as well (which takes time)
 - ↳ In a dual core system, each core only needs to communicate with one other core, whereas in a quad core system, each core needs to communicate with three other cores, which takes up processing power and time.