1 The processor

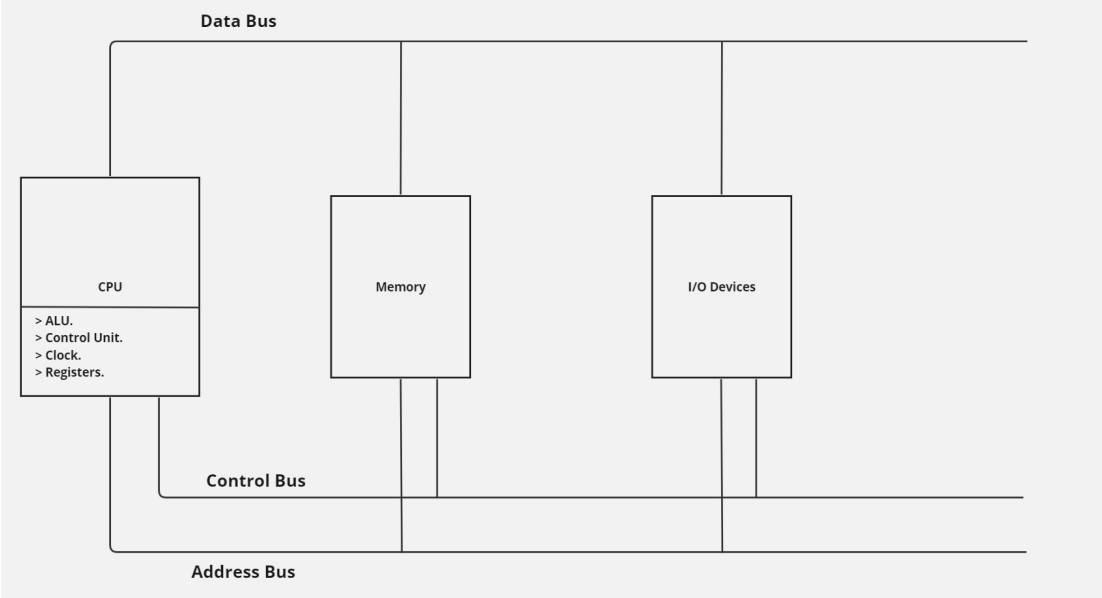

Figure 1: A simplified view of a processor with one CPU, one memory and certain number of I/O devices

CPU

Components of CPU:

- ALU.
- Control Unit.
- Clock.
- Registers.

CPU communication with the memory and the I/O devices

Buses:

- Data bus → Handles the transfer of instructions and data between the CPU, memory and I/O devices.
- Address bus → Helps to hold the address of instructions and the data that are being transferred between the CPU and memory and any of the other different devices.
- Control bus → Synchronizes all the actions between all of the devices that are attached to the bus. It helps us understand where we are reading and writing and what we are interacting with at the given time.

2 Components of a CPU

2.1 ALU

- **A**rithmetic **L**ogic **U**nit.
- Carries out logic and arithmetic.
- Performs operations like add, subtract, multiply, divide and logical operations like AND, OR, NOT, etc.

2.2 Memory registers

- A type of computer memory close to the CPU.
- Fastest way to access data.

2.3 CPU clock

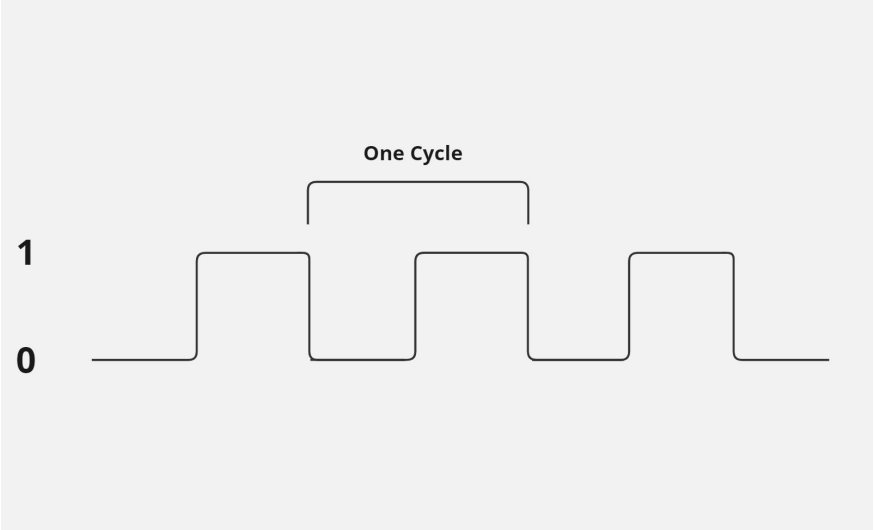

Figure 2: A cycle

The time between each drop from 1 to 0 gives us a single cycle of the clock. **Analogy:** Think of a cycle as a heart beat.

- Cycles between being on(1) and being off(0).
- Ticks at constant rate.
- Operations between CPU and bus are synchronized by an internal clock.
- Basic unit for instruction is a machine/clock cycle.
- Measured in oscillations per second($1\ GHz\ =\ 1\ billion\ times\ per\ second$).

2.4 Control unit

- Uses a binary decoder to convert coded instructions into timing and control signals.
- Direct operations to other units(memory, ALU, I/O).

3 Instruction Execution Cycle

- CPU completes a predefiend set of steps to execute an instruction.
- Steps:
  i. Fetch an instruction from the instruction queue.
  ii. Decode the instruction and check for operands.
  iii. If operands are involved, fetch the operands from memory/registers.
  iv. Execute the instruction and update status flags.
  v. Store the result if required.
- This is called **Fetch Decode Execute** procedure.

4 Reading from memory

- Memory access is slower than register access.
- Following steps are required:
  i. Place the address of the value you want to read on the address bus.
  ii. Change the processor's RD pin(called **assert**).
  iii. Wait one clock cycle for memory to respond.
  iv. Copy the data from the data bus to the destination.
- Each step takes approximately one clock cycle.
- Register access usually takes only one clock cycle.

5 Caching

- To reduce the read/write time for memory, caches are used.
- In x86:
  - Level-1 cache is stored on the CPU.
  - Level-2 cache is stored outside and accessed by high-speed data bus.
- Constructed using static RAM, which doesn't need to be refreshed constantly.