Von Neumann Architecture:

Main Memory — System Buses — Microprocessor

```
30 | LDD 34
31 | ADD 35
32 | STO 36
33 | END
34 | 3
35 | 6
36 | 9
```

Instructions (Program)
Data.

Control Bus
Address Bus
Data Bus

Control Unit (CU)
PC 31 30
CIR LDD34
MAR 30
MDR LDD 34

Arithmetic & Logic Unit (ALU)
Accumulator.

Immediate Access Storage (IAS)

Addresses   Content

ACC:
```
1+2+3+4=5
1 | 3 | 6 | 10 | 15
```
Interim data → Final output
Values produced during the execution of current program.
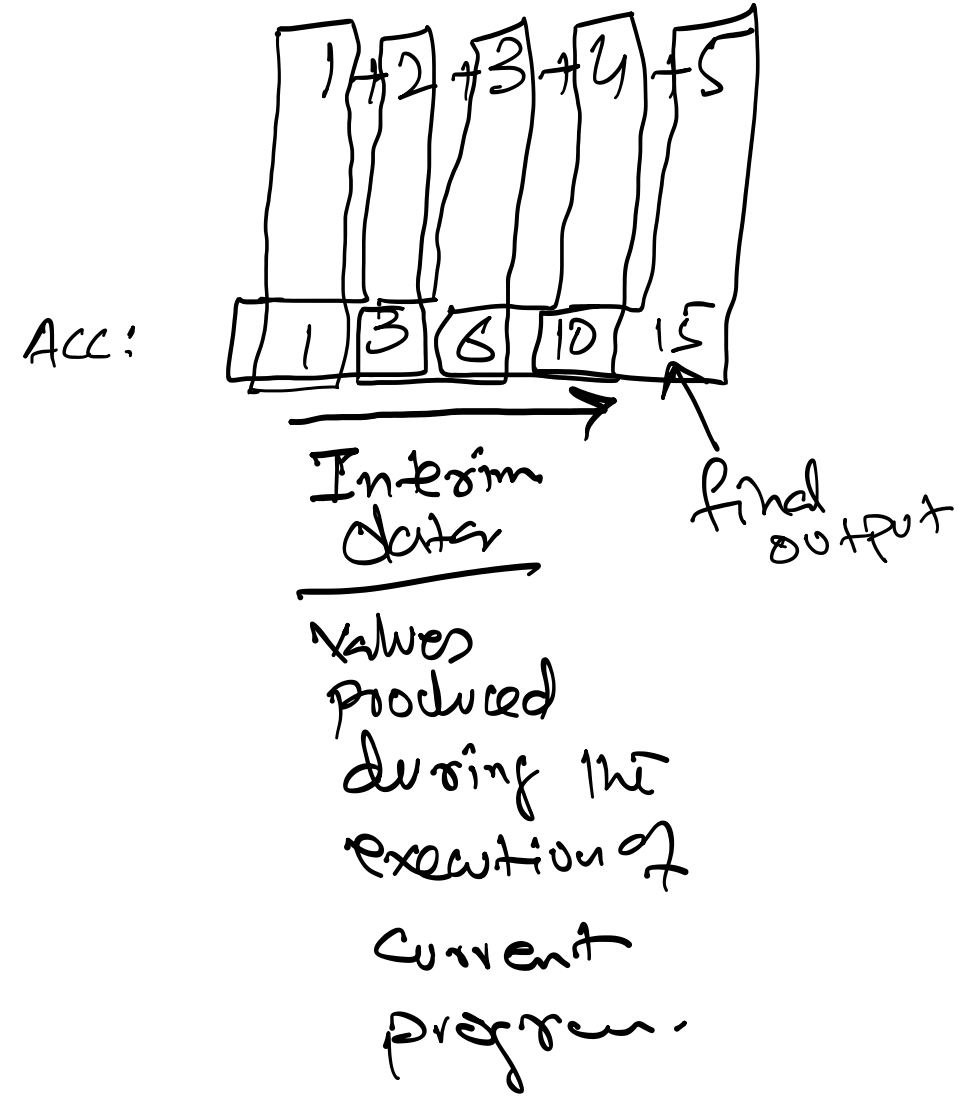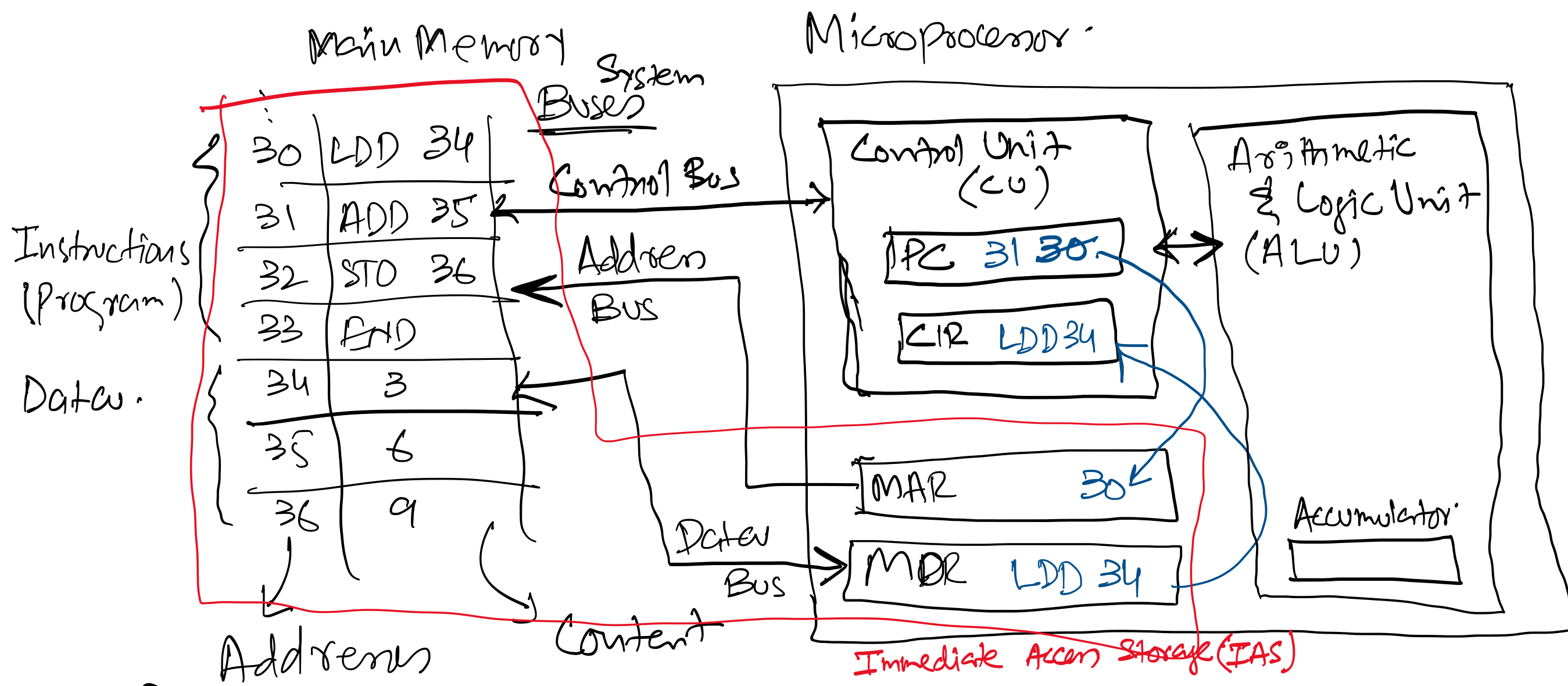
Registers: Smallest & fastest memories with a purpose inside the processor.

Special Purpose {
Program Counter (PC): It holds the address of next inst.
Memory Add. Register (MAR): It holds the add. of current inst.
Memory Data Register (MDR): It holds the current inst.
Current Inst. Register (CIR): It decodes & executes current inst
}

General Purpose {
Accumulator: It is general purpose register that holds the data produced during program execution.
}

PC → MAR   MDR → CIR → Decodes / Executes
30   Fetch
31
30   30   LDD34   LDD34

Add. Bus   Data Bus.
Main Memory
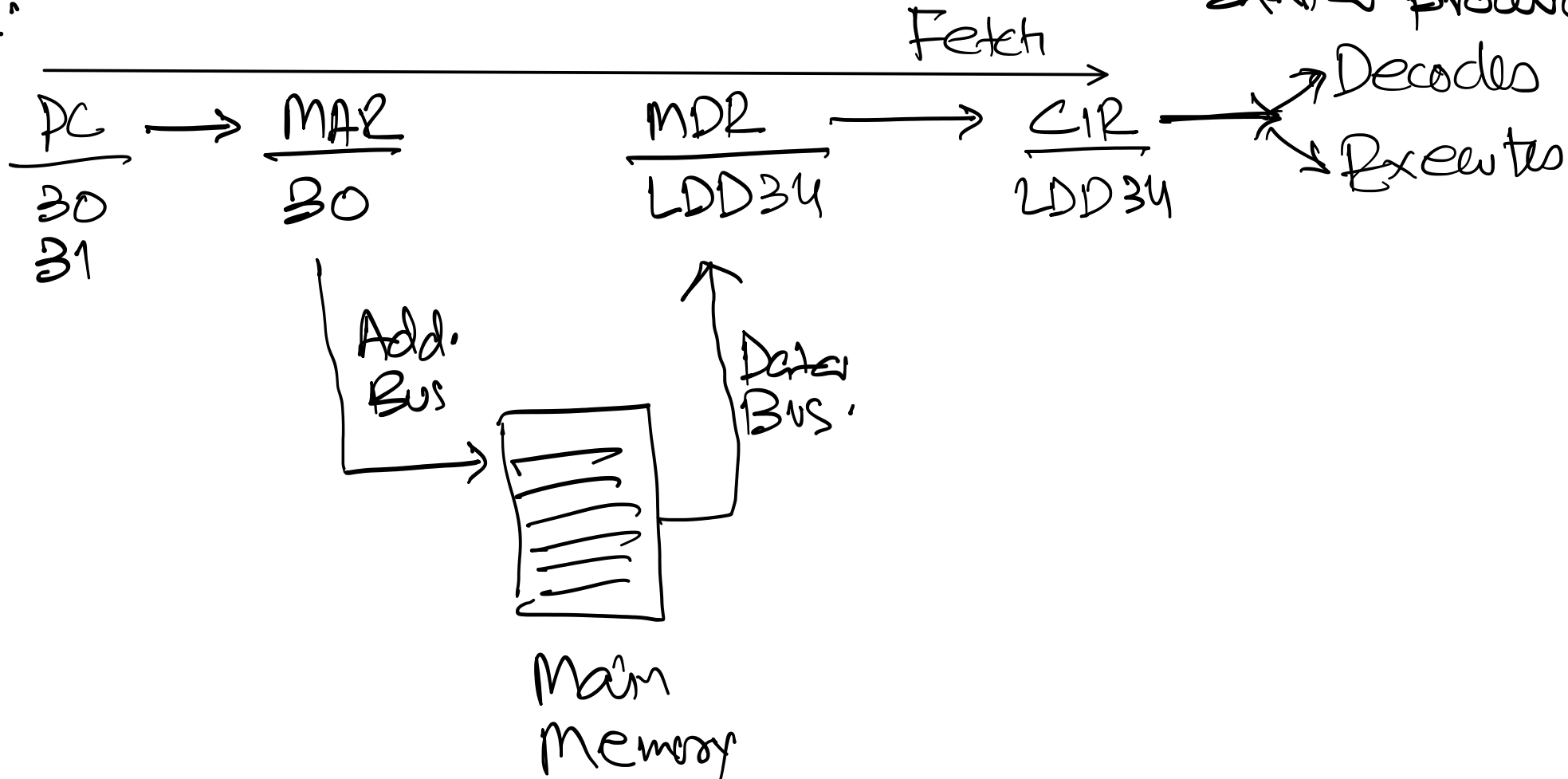
<u>CU</u> controls the flow of instructions execution by syncing all registers for fetch decode execute cycle.

<u>ALU</u> is suppose to be the hand of microprocessor. It completes all operations, like logical & arithmetic ones. Logical are AND, OR, NOT etc. Arithmetic are addition, subtraction, multiplication etc.

Von Neumann Architecture:
  Computer Architecture:
    Architecture:
      Idea of stored program:

- Both instructions & data in binary form are held in main memory.
- It is a single processor; made up of CU, ALU & memory unit.
- It uses input, output & storage devices.
- It is a serial machine.

Fetch-Decode-Execute Cycle:

Fetch {
1. Add. of the next inst. from PC goes to MAR.
2. PC increments it by 1.
3. Instruction in memory at the address mentioned in MAR arrives in MDR.
}

Decode & Execute {
4. From MDR, current inst. goes to CIR.
5. If the inst. has an address part then that address is copied to MAR.
6. Inst. is decoded and executed.
}