Computer Architecture

**+** A: special register called **accumulator.** It’s the main register used in arithmetic, logic, and data transfer operations. The accumulator = the "workbench" of the CPU.

**+** B, C, D, E, H, L: general register. Used to:

* Store temporary values.
* Work with the accumulator.
* Pair them up to form **16-bit registers**.

**🔹 Register Pairs (16-bit use)**

Because each register is only **8-bit**, some instructions need **16-bit addresses or data**. The 8085 allows registers to work in pairs:

* **BC pair** (B = high, C = low)
* **DE pair** (D = high, E = low)
* **HL pair** (H = high, L = low) → most important and often used as a memory pointer.

**+** MOV: copies content of the src register/memory into the dest register/memory

Syntax:

MOV destination, source

Example:

* MOV B, C ; Copy contents of register C into register B
* MOV A, M ; Copy contents of memory (pointed by HL) into A

**+** MVI: loads a register or memory with an immediate value (a constant) meaning the value is written directly in the instruction

Syntax:

MVI register, data

Example:

* MVI A, 32H ; Load hexadecimal 32 into register A
* MVI B, 05H ; Load decimal 5 into register B
* MVI M, 09H ; Load 09H into the memory location pointed by HL

(the H is to show that the numbers are written in hexadecimal)

**+** LDA: loads data from a specific memory address directly into accumulator (register A) basically it fetches the valued store at that memory address.

Syntax:

LDA 16-bit-address

(16-bit-address = the exact memory location in RAM/ROM (because 8085 has a 16-bit address bus, so addresses go from 0000H to FFFFH).

Example:

* LDA 2034H

This means:

1. Go to memory location 2034H.
2. Take the byte stored there (the content).
3. Load that value into the accumulator (A).

**+** LDAX: Copy the contents of a memory location (pointed to by a register pair) into the accumulator (A).

Syntax:

LDAX rp

(rp can be **BC** or **DE** (only these two, not HL).

Example:

Memory[3000H] = 55H

BC = 3000H

* LDAX B ; A = memory[BC] = memory[3000H]

This means:

1. Take the 16-bit address stored in BC or DE and go to that memory address.
2. Load the byte stored there into A (the accumulator).

**+** LXI: loads a 16-bit value directly into a register pair or stack pointer (SP).

Syntax:

LXI rp, 16-bit-data

Example:

* LXI H, 2500H

rg = HL pair = 2500H. H = 25H (high byte) L = 00H (low byte)

**+** LHLD: loads two memory bytes into HL (low byte first then high byte)

Syntax:

LHLD addr

Example:

* LHLD 3000H