

8-BIT ADDITION

EXP NO: 1

AIM:

To write an assembly language program to implement 8-bit addition using 8085 processor.

ALGORITHM:

- 1) Start the program by loading the first data into the accumulator.
- 2) Move the data to a register.
- 3) Get the second data and load it into the accumulator.
- 4) Add the two register contents.
- 5) Check for carry.
- 6) Store the value of sum and carry in the memory location.
- 7) Halt.

PROGRAM:

LDA 8500

MOV B, A

LDA 8501

ADD B

STA 8502

RST 1

INPUT\OUTPUT:

The screenshot displays the GNUSim8085 - 8085 Microprocessor Simulator interface. The main window is divided into several sections:

- Registers:** Shows the state of the 8085 registers. The Accumulator (A) contains 50, B contains 28, C contains 00, D contains 00, E contains 00, F contains 00, H contains 00, L contains 00, P contains 00, S contains 00, and SP contains FF. The PC (Program Counter) is 42, and the PSW (Program Status Word) is 0C.
- Flags:** Shows the status of the flags. The Sign flag (S) is 0, Zero flag (Z) is 0, Carry flag (C) is 0, and the Parity flag (P) is 1.
- Assembly Program:** The program is loaded and ready to be executed. The instructions are: 1. LDA 8500, 2. MOV B, A, 3. LDA 8501, 4. ADD B, 5. STA 8502, 6. HLT.
- Memory:** A table showing the memory dump. The address 8500 contains 40, 8501 contains 40, 8502 contains 80, and 8503 contains 0. The rest of the memory locations shown are empty (0).
- Assembler Message:** A message box at the bottom right indicates "Program assembled successfully".

RESULT: Thus the program was executed successfully using 8085 processor simulator.