

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:

Start	8500	
Address (Hex)	Address	Data
2134	8500	3
2135	8501	4
2136	8502	7
2137	8503	0

OUTPUT:

The screenshot displays the GNUSim8085 - 8085 Microprocessor Simulator interface. The main window shows the assembly code being executed, with line numbers 1 through 32. The code includes instructions like `<Program title>`, `jmp start`, `data`, `code`, `start: nop`, `LDA 8500`, `MOV B, A`, `LDA 8501`, `ADD B`, `STA 8502`, `RST 1`, and `hlt`. The left panel shows the registers (A, BC, DE, HL, PSW, PC, SP, Int-Reg) and flags (S, Z, AC, P, C). The right panel shows the memory dump, which matches the table provided in the first image. The bottom status bar indicates the simulator is idle.

Registers:

Register	Value
A	07
BC	03 00
DE	00 00
HL	00 00
PSW	00 00
PC	42 10
SP	FF FF
Int-Reg	00

Flags:

Flag	Value
S	0
Z	0
AC	0
P	0
C	0

Memory Dump:

Address (Hex)	Address	Data
2134	8500	3
2135	8501	4
2136	8502	7
2137	8503	0
2138	8504	0
2139	8505	0
213A	8506	0
213B	8507	0
213C	8508	0
213D	8509	0
213E	8510	0
213F	8511	0

Assembler Message:

```

Line No  Assembler Message
0         Program assembled successfully
  
```

Simulator: Idle

RESULT:

Thus the program was executed successfully using 8085 processor simulator.