

8-BIT DIVISION

EXP NO: 4

AIM:

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

ALGORITHM:

- 1) Start
the program by loading a register pair with the address of memory location.
- 2) Move
the data to a register.
- 3) Get
the second data and load it into the accumulator.
- 4) Subtract
the two register contents.
- 5) Increment
the value of the carry.
- 6) Check
whether the repeated subtraction is over.

7) Store
the value of quotient and the reminder in the memory location.

8) Halt.

PROGRAM:

LDA 8501

MOV B, A

LDA 8500

MVI C,00

LOOP: CMP B

JC LOOP1

SUB B

INR C

JMP LOOP

LOOP1: STA 8502

MOV A, C

STA 8503

RST 1

INPUT

Address (Hex)	Address	Data
2134	8500	6
2135	8501	42

OUTPUT

GNUSim8085 - 8085 Microprocessor Simulator

File Reset Assembler Debug Help

Registers: A 00, BC 2A 00, DE 00 00, HL 00 00, PSW 00 00, PC 42 1E, SP FF FF, Int-Reg 00. Flag: S 1, Z 0, AC 0, P 0, C 1.

Decimal - Hex Conversion: Decimal 0, Hex 0. Buttons: To Hex, To Dec.

I/O Ports: 0, -, +, 00. Button: Update Port Value.

Memory: 0, -, +, 00. Button: Update Memory.

Load me at: []

```
1 ;<Program title>
2 jmp start
3
4 ;data
5
6 ;code
7 start: nop
8
9 LDA 8500
10 MOV B, A
11 MVI C, 00
12 LOOP: CMP B
13 JC LOOP1
14 SUB B
15 TNR C
16 JMP LOOP
17 LOOP1: STA 8502
18 MOV A, C
19 STA 8503
20 RST 1
21 hlt
```

Start: 8500

Address (Hex)	Address	Data
2134	8500	6
2135	8501	42
2136	8502	6
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
2140	8512	0
2141	8513	0

Line No Assembler Message

0 Program assembled successfully

Simulator: Idle

30°C Mostly cloudy

Search

ENG IN

10:05 AM 16-10-2023

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