

## 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 remainder 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:

Data
 Stack
 KeyPad
 **Memory**
 I/O Ports

Start

Address (Hex)	Address	Data
2134	8500	10
2135	8501	3

OUTPUT:

File Reset Assembler Debug Help

Registers
 

A	03	S	1
BC	03 03	Z	0
DE	00 00	AC	0
HL	00 00	P	0
PSW	00 00	C	1
PC	42 1E		
SP	FF FF		
Int-Reg	00		

Flag
 

S	1
Z	0
AC	0
P	0
C	1

Decimal - Hex Conversion
 

Decimal	Hex
0	0

I/O Ports
 

0	-	+	00
---	---	---	----

Memory
 

0	-	+	00
---	---	---	----

Load me at

```

1  ;<Program title>
2
3
4  jmp start
5
6  ;data
7
8
9  ;code
10 start: nop
11
12 LDA 8500
13 MOV B, A
14 LDA 8500
15 MVI C, 00
16
17 LOOP: CMP B
18 JC LOOP1
19 SUB B
20 INR C
21 JMP LOOP
22
23 LOOP1: STA 8502
24 MOV A, C
25 STA 8503
26 RST 1
27 hlt

```

Start

Address (Hex)	Address	Data
2134	8500	10
2135	8501	3
2136	8502	1
2137	8503	3
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
2142	8514	0

Line No | Assembler Message
 

0	Program assembled successfully
---	--------------------------------

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