

Write a program in 8085 Assembly language to divide a 16-bit number by an 8-bit number.

Address	Assembly Code	Hex Code	Comments
0000	LHLD 2050	2A 50 20	Load hl pair register from memory location 2050h
0003	LDA 2052H	3A 52 20	Load into Accumulator from memory location 2052h
0006	MOV B, A	47	Move data (divisor) from accumulator to B register
0007	MVI C, 08H	0E 08	Move 8 to C
0009	BACK: DAD H	29	HL <- HL + HL
000A	MOV A, H	7C	Move data (LB of dividend) in H register to accumulator
000B	SUB B	90	Subtract B register value from accumulator
000C	JC NEXT	DA 11 08	Jump if carry to Label NEXT
000F	MOV H, A	67	Move data in accumulator to H register
0010	INR L	2C	Increment L register by 1
0011	NEXT: DCR C	0D	Decrease register C by 1
0012	JNZ BACK	C2 09 08	Jump to Label BACK if C not zero
0015	SHLD 2053H	22 53 20	Store HL register pair in memory location 2053h
0018	HLT	76	Halt

#### **Procedure:**

Step – 1: Writing program in memory.

1. Press Reset
2. Press SET/MEM
3. Type in Address 0000
4. Press Enter
5. Type 1<sup>st</sup> Hex Code (Here 21)
6. Press Enter
7. Follow step 5 and 6 to type in all Hex Code

Step – 2: Assigning Value to the Address Location

1. Press Reset
2. Press SET/MEM

3. Type in Address of 1<sup>st</sup> Location (Here 2050)
4. Press Enter
5. Enter value of Dividend (LB)
6. Press Enter
7. Enter value of Dividend (UB)
8. Press Enter
9. Enter value of Divisor

### Step – 3: Executing the program

1. Press Reset to clear Buffer
2. Press Go
3. Enter Starting Address of Program (Here 0000)
4. Press Execute

### Step – 4: Checking output

1. Press reset and Clear the Buffer
2. Press Go
3. Enter Result Location (Here 2053 for quotient and 2054 for Remainder)
4. You will find the product of the numbers

### Output:

8085 Simulator

File Edit Tools Settings Simulation Subroutine View Load Sample Program Help

Editor Assembler

Assembler

* Address	Label	Mnemonics	Hexcode	Bytes	M-Cycles	T-States
✓ 0009	BACK	DAD H	29	1	3	10
✓ 000A		MOV A,H	7C	1	1	4
✓ 000B		SUB B	90	1	1	4
✓ 000C		JC 0011	DA	3	3	10
000D			11			
000E			00			
✓ 000F		MOV H,A	67	1	1	4
✓ 0010		INR L	2C	1	1	4
✓ 0011	NEXT	DCR C	0D	1	1	4
✓ 0012		JNZ 0009	C2	3	3	10
0013			09			
0014			00			
✓ 0015		SHLD 2053	22	3	5	16
0016			53			
0017			20			
✓ 0018		HLT	76	1	2	5

Simulate

Start From → 0000

Run all At a Time Step By Step

Registers Memory Devices

Memory Editor

Memory Range: 0000 ---- FFFF

Memory Address	Value
000C	DA
000D	11
000F	67
0010	2C
0011	0D
0012	C2
0013	09
0015	22
0016	53
0017	20
0018	76
2050	1E
2051	60
2052	06
2053	FF
2054	24

☐ Show entire memory content  
☒ Show only loaded memory location  
☐ Store directly to specified memory location

Created by : Jubin Mitra







