Exp No: 12 **Date:** 01/11/2020

Name: Swetha Saseendran

Reg No: 185001183

8-BIT ARITHMETIC OPERATIONS USING 8051

Aim:

To program and execute 8-bit arithmetic operations using 8051 microcontrollers using EDSim.

Programs:

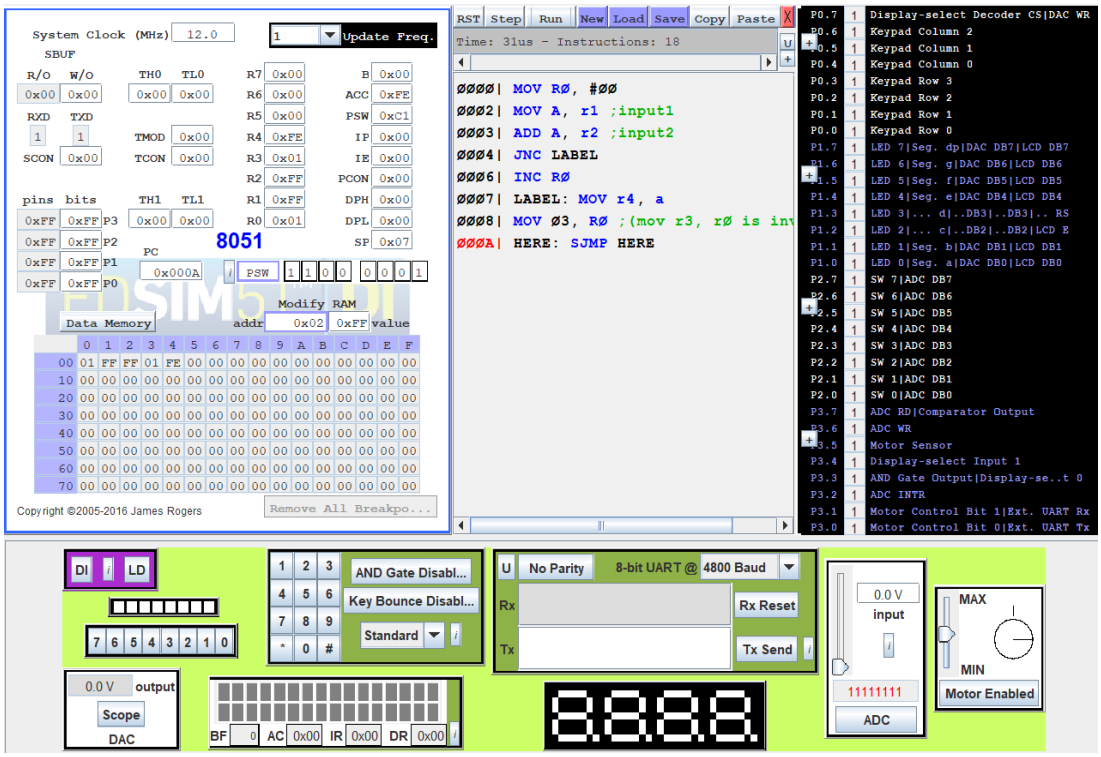
(i) 8-BIT ADDITION

Algorithm:

- Move input1 to A.
- Add the second input to A and store the result in A.
- Jump if no carry to label.
- Increment register 0.
- Label: mov the result to register 4 and carry to register 3.
- Here: short jump here.

Program	*Comment*
;Program to subtract 2 8 bit numbers using 8051 microcontroller	
MOV R0, #00	Move value 00 to R0.
MOV A, r1	Move input1 to A
ADD A, r2	Add A and 2nd input and store in A
JNC LABEL	Jump if no carry to label
INC R0	Increment R0
LABEL: *MOV r4, a*	Move result to R4
MOV 03, R0 ;(mov r3, r0 is invalid)	Move carry to R3
HERE: *SJMP HERE*	End

Snapshot of sample input and output:



(ii) 8-BIT SUBTRACTION

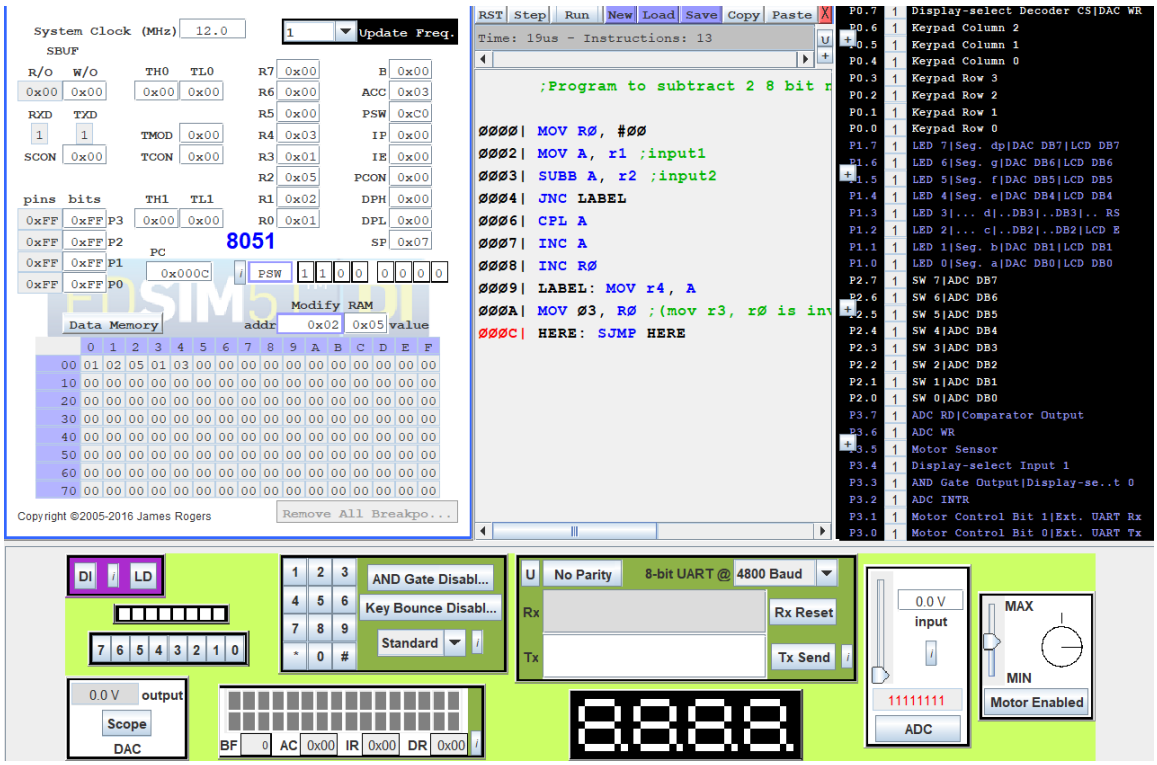
Algorithm:

- Move input1 to A.
- Add the second input to A and store the result in A.
- Jump if no carry to label.
- Increment register 0.
- Label: mov the result to register 4 and carry to register 3.
- Here: short jump here.

Program	*Comment*
;Program to subtract 2 8 bit numbers using 8051 microcontroller	
MOV R0, #00	Move value 00 to R0.
MOV A, r1	Move input1 to A
SUBB A, r2	Subtract A and 2nd input and store in A
JNC LABEL	

CPL A	Jump if no carry to label
INC A	complement A
INC R0	Increment A
LABEL: MOV r4, A	Increment R0
MOV 03, R0	Move result to R4
***HERE:** SJMP HERE*	Move carry to R3
	End

Snapshot of sample input and output:



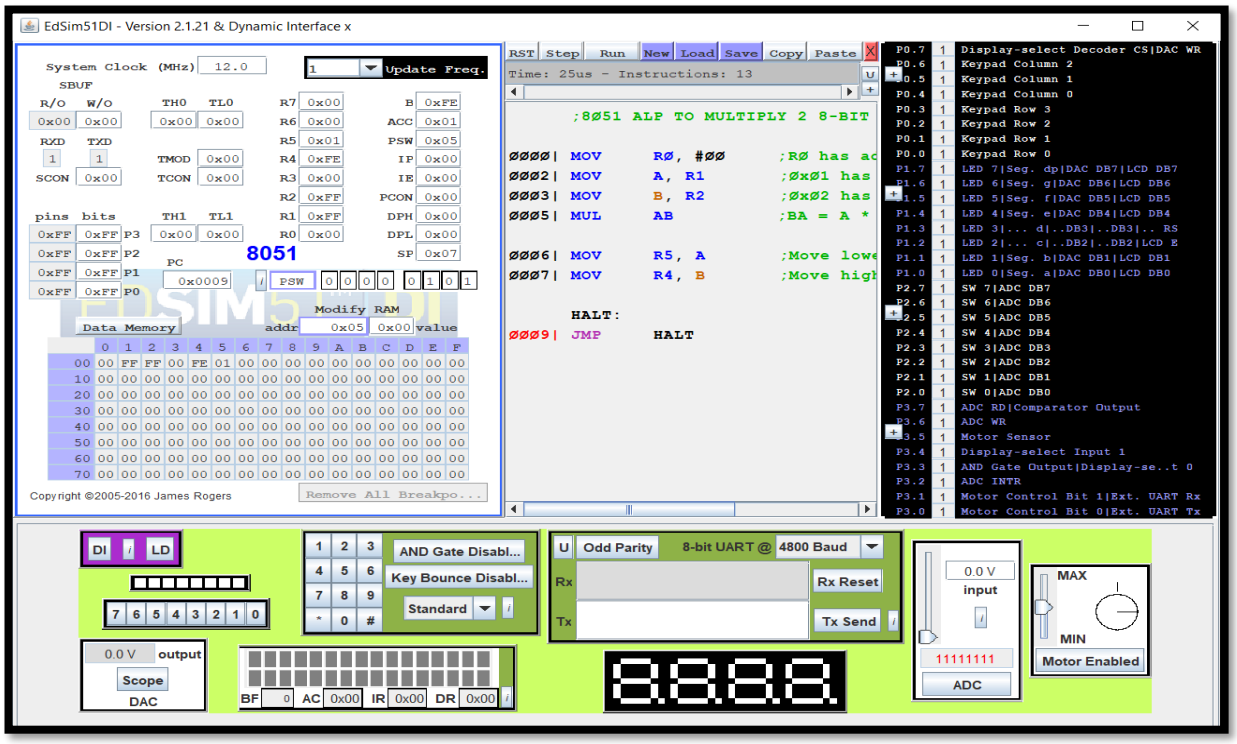
(iii) 8-BIT MULTIPLICATION

Algorithm:

- Initialize R0 with 00h
- Move the value in R1 to A.
- Move the value in R2 to B.
- Multiply A and B.
- Move B to R4 (MSB of product) and A to R5 (LSB of product)

PROGRAM	*COMMENTS*
MOV R0, #00	R0 has address of 0x00
MOV A, R1	0x01 has 1st 8-bit number
MOV B, R2	0x02 has 2nd 8-bit number
MUL AB	BA = A * B
MOV R5, A	Move lower byte to R5 from A
MOV R4, B	Move higher byte to R4 from B
HALT:	
SJMP HALT	Halt the program with a loop.

Snapshot of sample input and output:



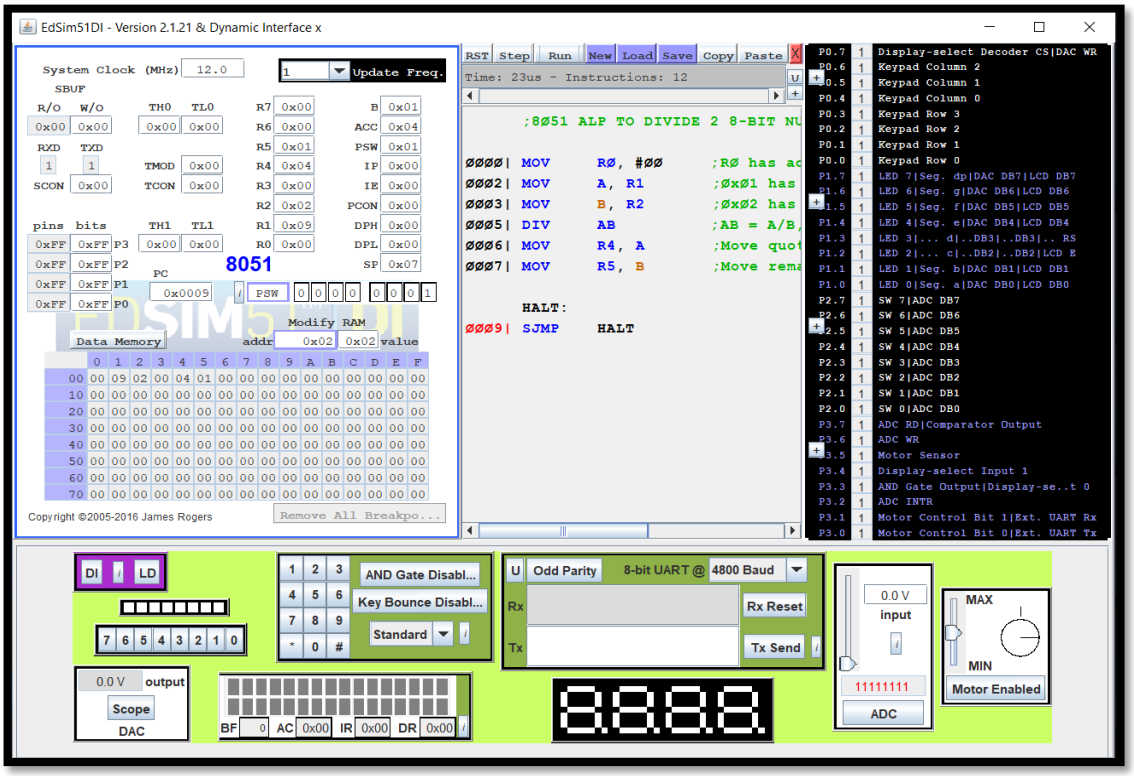
(iv) 8-BIT DIVISION

Algorithm:

- Initialize R0 with 00h.
- Move the value in R1 to A.
- Move the value in R2 to B.
- Divide A by B.
- Move A to R4 (Quotient) and B to R5 (Remainder)

PROGRAM	*COMMENTS*
MOV R0, #00	R0 has address of 0x00
MOV A, R1	0x01 has 1st 8-bit number
MOV B, R2	0x02 has 2nd 8-bit number
DIV AB	BA = A / B, A: Quotient, B: Remainder
MOV R5, A	Move quotient to R4 from A
MOV R4, B	Move remainder to R5 from B
HALT:	
SJMP HALT	Halt the program with a loop.

Snapshot of sample input and output



Result:

The assembly level programs were written to perform the above specified 8-bit arithmetic operations using an 8051 microcontroller and the outputs were verified.