

8 Bit Arithmetic Operations using 8051

Expt No: 12

Name: Shivanirudh S G

Date : 23/10/2020

Reg No: 185001146

Aim:

To perform arithmetic operations on two 8 bit numbers using **8051 microcontroller**.

8 Bit Addition

Algorithm:

- Move hex value 00 to register 0.
- Move the value of register 1 to A.
- Add value in register 2 to A using ADD A, R2
- Using JNC instruction check for carry and if there is no carry, no need to increment R0.
- Else, increment R0 by 1.
- The result and carry stored in A and R0 should be moved to R4 and R3 respectively.

Program:

Program	Comments
mov r0, #00	Move hex value 00 to Register 0
mov a, r1	Move value in Register 1 to A
add a, r2	A = A + R2
jnc label	Jump if no carry to label LABEL
inc r0	Increment value in R0
label: mov r4, a	Move result in A to Register 4
mov 03, r0	Move carry in R0 to R3
here: sjmp here	Halt

Input and Output:

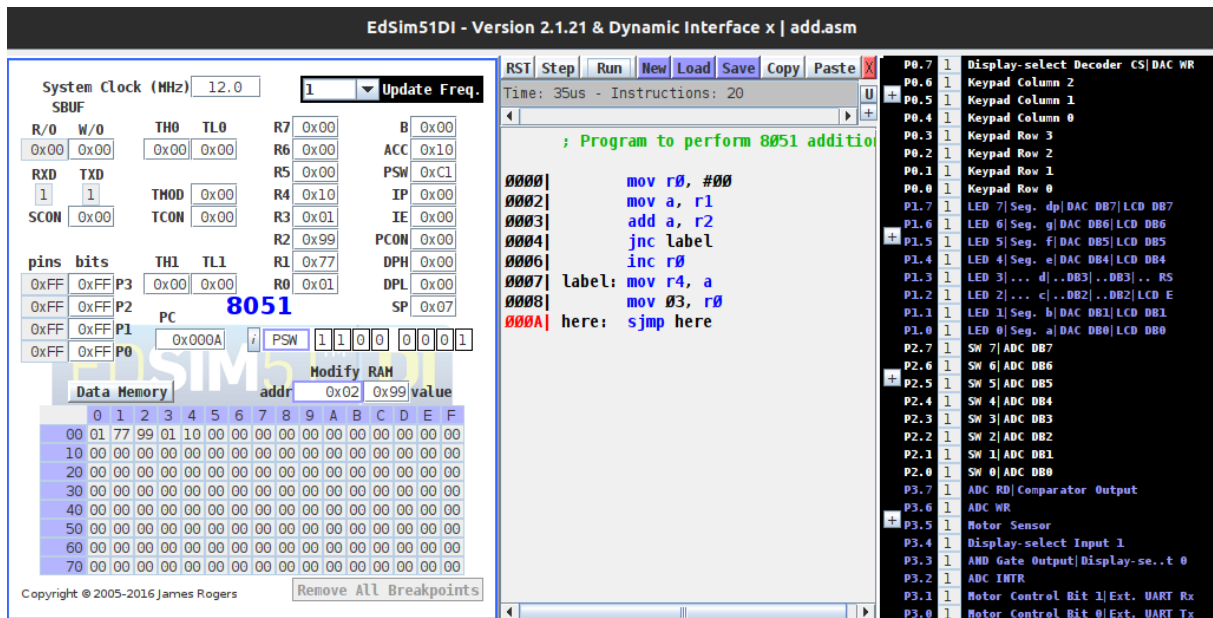


Figure 1: **Input:** r1: 77h, r2: 99h; **Output:** Result: 10h, Carry: 01h

8 Bit Subtraction

Algorithm:

- Move hex value 00 to register 0.
- Move the value of register 1 to A.
- Subtract value in register 2 from A using SUBB A, R2
- Using JNC instruction check for carry and if there is no carry, no need to increment R0.
- Else, increment R0 by 1, complement and increment A by 1.
- The result and sign stored in A and R0 should be moved to R4 and R3 respectively.

Program:

Program	Comments
mov r0, #00	Move hex value 00 to Register 0
mov a, r1	Move value in Register 1 to A
subb a, r2	A = A - R2
jnc label	Jump if no carry to label LABEL
inc r0	Increment value in R0
cpl a	Complement value in A
inc a	2's complement of A
label: mov r4, a	Move result in A to Register 4
mov 03, r0	Move carry in R0 to R3
here: sjmp here	Halt

Input and Output:

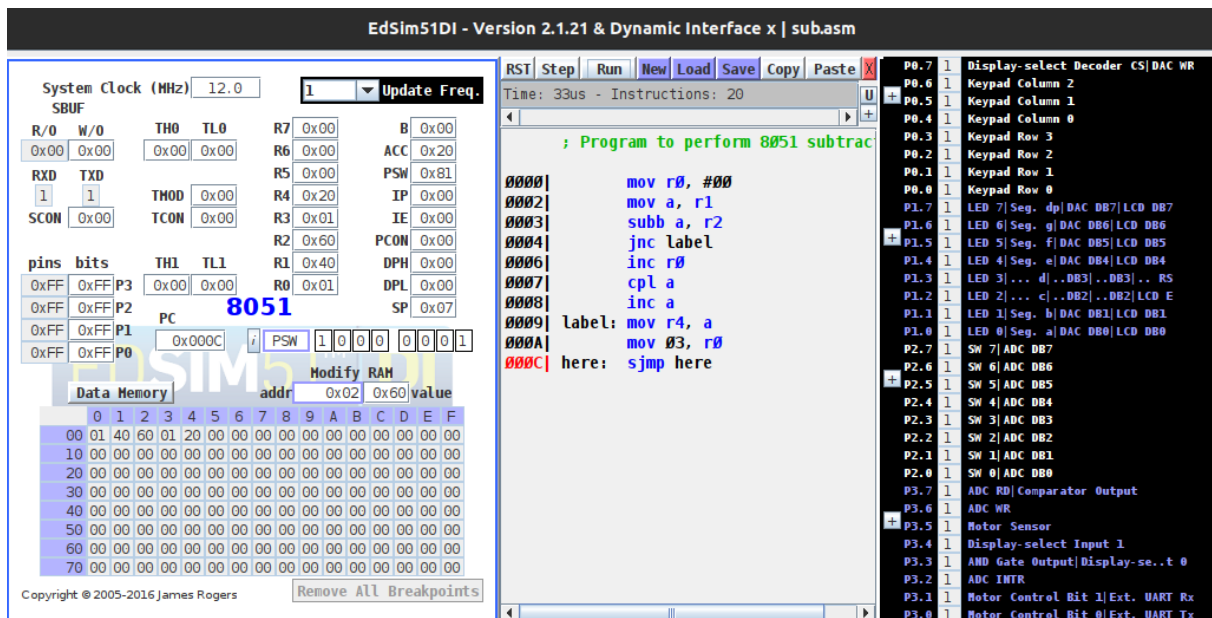


Figure 2: **Input:** r1: 40h, r2: 60h; **Output:** Result: 20h, Sign: 01h

8 Bit Multiplication

Algorithm:

- Move hex value 00 to register 0.
- Move the value of registers 1 and 2 to A and B respectively.
- Multiply using MUL A, B
- The higher and lower bits stored in A and B should be moved to R3 and R4 respectively.

Program:

Program	Comments
mov r0, #00	Move hex value 00 to Register 0
mov a, r1	Move value in Register 1 to A
mov b, r2	Move value in Register 2 to B
mul ab	A = A * B
mov r3, b	Move higher bits in B to Register 3
mov r4, a	Move lower bits in A to Register 4
here: sjmp here	Halt

Input and Output:

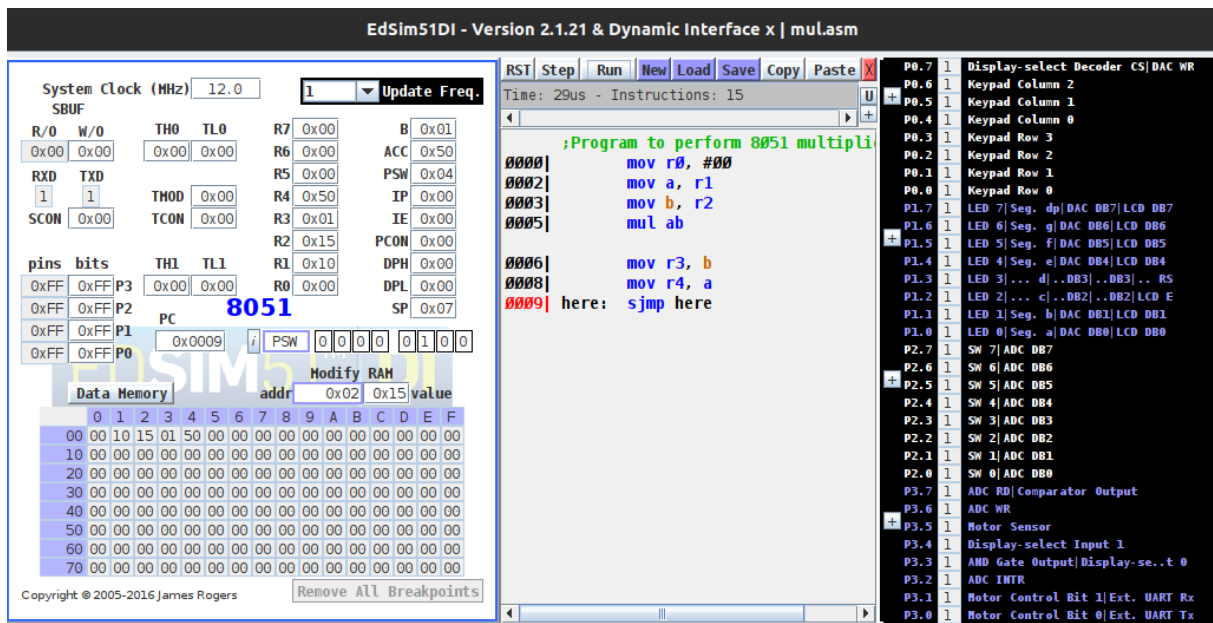


Figure 3: **Input:** r1: 10h, r2: 15h; **Output:** Lower: 50h, Higher: 01h

8 Bit Division

Algorithm:

- Move hex value 00 to register 0.
- Move the value of registers 1 and 2 to A and B respectively.
- Divide using DIV A, B
- The quotient and remainder bits stored in A and B should be moved to R3 and R4 respectively.

Program:

Program	Comments
mov r0, #00	Move hex value 00 to Register 0
mov a, r1	Move value in Register 1 to A
mov b, r2	Move value in Register 2 to B
div ab	$A = A / B$; $B = A \% B$
mov r3, a	Move quotient in A to Register 3
mov r4, b	Move remiander in B to Register 4
here: sjmp here	Halt

Input and Output:

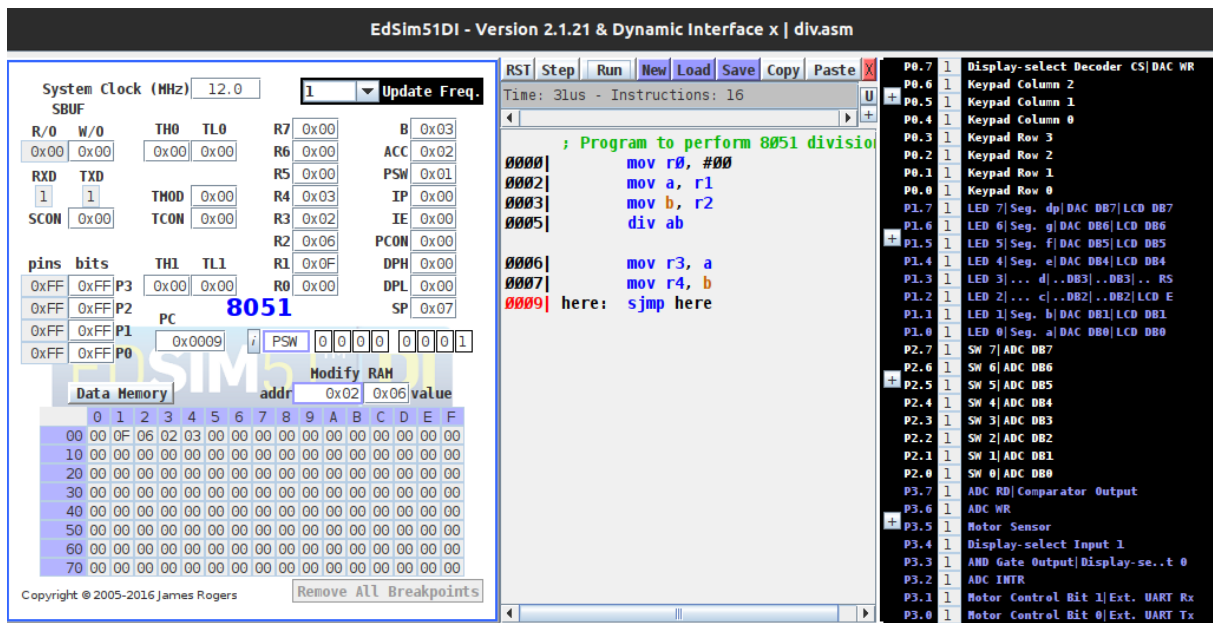


Figure 4: **Input:** r1: 0Fh, r2: 06h; **Output:** Quotient: 02h, Remiander: 03h

Result:

The 8051 programs were written to perform 8-bit arithmetic operations, and the results observed.