SSN College of Engineering Department of Computer Science and Engineering

III year - UCS1512 - Microprocessors Lab8-bit arithmetic operations using 8051

Exp No: 12

Name: Rahul Ram M

Register Number: 185001121

Date: 20/10/2020

12a) Addition:

Aim:

To design 8051-program for 8-bit addition.

Algorithm:

- 1. Move 00H to R0.
- 2. Move the value in R1 to register A.
- 3. Add register A and the value in the R2. The result is stored in register A.
- 4. Jump to LABEL if there is no carry.
- 5. Else increment value in RO.
- 6. LABEL: Move the register A's value to R4 and move the value in R0 to R3.
- 7. HERE: Infinite loop to HERE using SJMP HERE.

Program:

MOV R0, #00

MOV A, r1 ;input1

ADD A, r2 ;input2

JNC LABEL

INC R0

LABEL: MOV r4, a

MOV 03, R0 ;(mov r3, r0 is invalid)

HERE: SJMP HERE

	Program	Comments
START:	MOV R0, #00	R0 <- 00H
	MOV A, r1	A <- R1
	ADD A, r2	A <- A + R2
	JNC LABEL	Jump to LABEL, if no carry.
	INC RO	R0++
LABEL:	MOV r4, a	R4 <- A
	MOV 03, R0	R3 <- R0
HERE:	SJMP HERE	Transfers execution to HERE.

Snapshot of sample output:

R1 = 6F, R2 = EE.



Result:

Thus the 8051-program for 8-bit addition is executed successfully.

12a) Subtraction:

Aim:

To design 8051-program for 8-bit subtraction.

Algorithm:

- 1. Move 00H to R0.
- 2. Clear the carry flag using CLR C.
- 3. Move the value in R1 to register A.
- 4. Sub register A with the value in the R2 using SUBB A, R2. The result is stored in register A.
- 5. Jump to LABEL if there is no carry.
- 6. Else increment value in R0 for carry. Take 2's compliment of register A by using the instructions CPL A and INC A.
- 7. LABEL: Move the register A's value to R4 and move the value in R0 to R3.
- 8. HERE: Infinite loop to HERE using SJMP HERE.

Program:

MOV RO, #00

CLR C

MOV A, r1 ;input1 SUBB A, r2 ;input2

JNC LABEL INC RO CPL A

INC A

LABEL: MOV r4, a

MOV 03, R0 ;(mov r3, r0 is invalid)

HERE: SJMP HERE

	Program	Comments
START:	MOV R0, #00	R0 <- 00H
	CLR C	Clear the carry flag.
	MOV A, r1	A <- R1
	ADD A, r2	A <- A + R2
	JNC LABEL	Jump to LABEL, if no carry.
	INC RO	R0++
	CPL A	Complient the value in Register A.
	INC A	Increment A to get 2's compliment.
LABEL:	MOV r4, a	R4 <- A
	MOV 03, R0	R3 <- R0
HERE:	SJMP HERE	Transfers execution to HERE.

Snapshot of sample output:

R1 = 6F, R2 = EE.

_												_				
	0	1	2	3	4	5	6	7	8	9	A	В	С	D	Ε	F
0.0	01	6F	EE	01	7F	00	00	00	00	00	00	00	00	00	00	00
10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Result:

Thus the 8051-program for 8-bit subtraction is executed successfully.

12a) Multiplication:

Aim:

To design 8051-program for 8-bit multiplication.

Algorithm:

- 1. Move the value in R0 to register A.
- 2. Move the value in R1 to register B.
- 3. Multiply A and B using MUL AB (BA <- A x B).
- 4. Move the register A's value to R3 and move the Register B's value to R2.
- 5. HERE: Infinite loop to HERE using SJMP HERE.

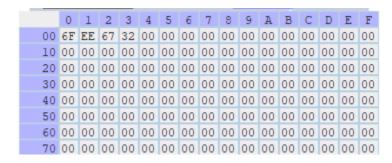
Program:

	MOV A, r0	;input1
	MOV B, r1	;input2
	MULAB	;BA = A x B
	MOV r2, B	
	MOV r3, A	
HERE:	SJMP HERE	
1		

	Program	Comments
START:	MOV A, R0	A <- R0
	MOV B, R1	B <- R1
	MUL AB	BA = A x B
	MOV R2, B	R2 <- B
	MOV R3, A	R3 <- A
HERE:	SJMP HERE	Transfers execution to HERE.

Snapshot of sample output:

R0 = 6F, R1 = EE.



Result:

Thus the 8051-program for 8-bit multiplication is executed successfully.

12a) Division:

Aim:

To design 8051-program for 8-bit division.

Algorithm:

- 1. Move the value in R0 to register A.
- 2. Move the value in R1 to register B.

- 3. Divide A by B using DIV AB with Quotient in A and Remainder in B.
- 4. Move the register A's value to R3 and move the Register B's value to R2.
- 5. HERE: Infinite loop to HERE using SJMP HERE.

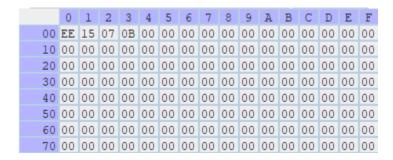
Program:

MOV A, RO ; input1
MOV B, R1 ; input2
DIV AB ; A / B; Quotient in A, Remainder in B
MOV R3, A
MOV R2, B
HERE: SJMP HERE

	Program	Comments
START:	MOV A, RO	A <- R0
	MOV B, R1	B <- R1
	DIV AB	A / B; Quotient in A, Remainder in B
	MOV R3, A	R3 <- A
	MOV R2, B	R2 <- B
HERE:	SJMP HERE	Transfers execution to HERE.

Snapshot of sample output:

R1 = EE, R2 = 15.



Result:

Thus the 8051-program for 8-bit division is executed successfully.