Experiment No. 12: 8-bit arithmetic operations using 8051

Date: 19-10-2020

NAME: Harshini S
REG.NO: 185001058

A.AIM:

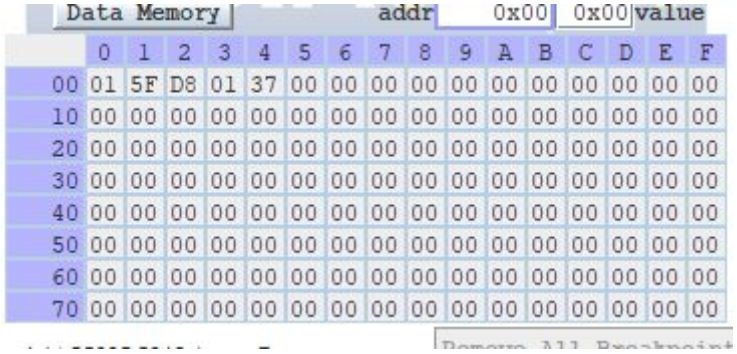
Program to add 2 8-bit numbers using 8051.

PROGRAM:

PROGRAM	COMMENTS

MOV R0, #00
MOV A, r1
ADD A, r2
JNC LABEL
INC R0
LABEL:
MOV R4, A
MOV 03, R0
HERE:
SJMP HERE
	Loading R0 with 00
	Move data from R1 to A
	A=A+R2
	Jump if no carry to Label
	Increment R0
	
	Load value of A to R4
	Move data from R0 to 03
	
	Continuous loop

SAMPLE INPUT/OUTPUT:



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	01	5F	D8	01	37	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 addition of 2 8-bit numbers has been performed.

B.AIM:

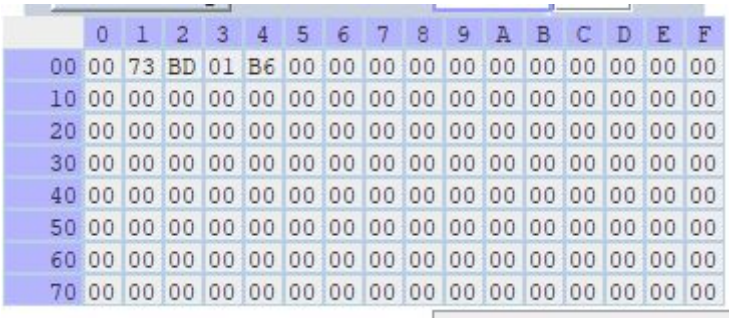
Program to subtract 2 8-bit numbers using 8051.

PROGRAM:

PROGRAM	**COMMENTS**

MOV R0, #00
MOV R3, #00
MOV A, R1
SUBB A, R2
JNC LABEL
INC R3
MOV B, R3
MOV R3, B
LABEL:
MOV R4, A
HERE:
SJMP HERE
	Loading R0 with 00
	Loading R3 with 00
	Move data from R1 to A
	A=A-R5 Subtract with borrow
	Jump if no carry to Label
	Increment R3
	Move data from R3 to B
	Load value of B to R3
	
	Move data from A to R4
	
	Continuous loop

SAMPLE INPUT/OUTPUT:



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	00	73	BD	01	B6	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 subtraction of 2 8-bit numbers has been performed.

C.AIM:

Program to multiply 2 8-bit numbers using 8051.

PROGRAM:

PROGRAM	**COMMENTS**

MOV R0, #00
MOV A, R1
MOV B, R2
MUL AB
MOV R3, B
MOV R4, A
HERE:
SJMP HERE
	Loading R0 with 00
	Move data from R1 to A
	Move data from R2 to B
	BA=AxB
	Load value of B to R3
	Load value of A to R4
	
	Continuous loop

SAMPLE INPUT/OUTPUT:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	00	17	12	01	9E	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 multiplication of 2 8-bit numbers has been performed.

D.AIM:

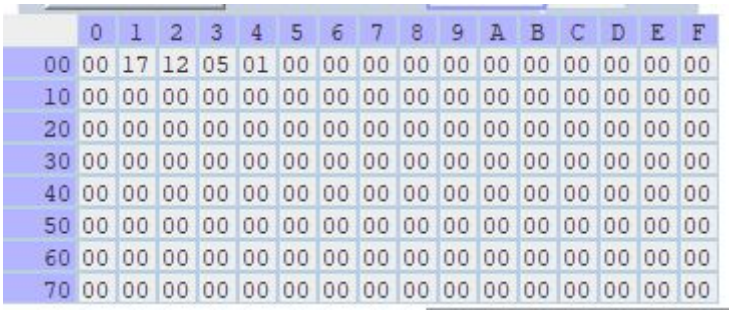
Program to divide 2 8-bit numbers using 8051.

PROGRAM:

PROGRAM	**COMMENTS**

MOV R0, #00
MOV A, R1
MOV B, R2
DIV AB
MOV R3, B
MOV R4, A
HERE:
SJMP HERE
	Loading R0 with 00
	Move data from R1 to A
	Move data from R2 to B
	A=A/B
	Load value of B to R3
	Load value of A to R4
	
	Continuous loop

SAMPLE INPUT/OUTPUT:



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	00	17	12	05	01	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 division of 2 8-bit numbers has been performed.

Experiment No. 13: Cube of a number using 8051

AIM:

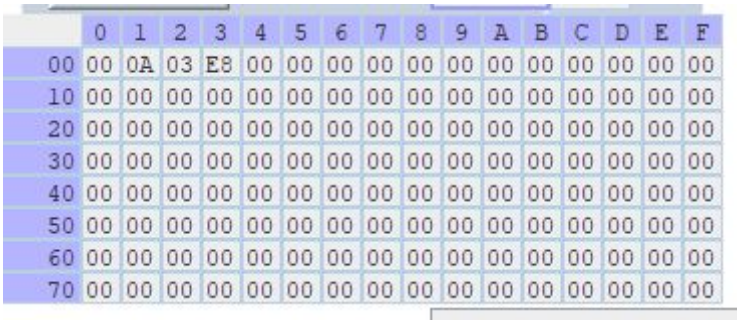
Program to find the cube of a number (0 - F) using 8051.

PROGRAM:

PROGRAM	COMMENTS

MOV R0, #00
MOV A, R1
MOV B, R1
MUL AB
MOV B, R1
MUL AB
MOV R2,B
MOV R3,A
HERE:
SJMP HERE
	Loading R0 with 00
	Load value of R1 to A
	Load value of R1 to B
	BA=AxB
	Load value of R1 to B
	BA=BxA
	Load value of B to R2
	Load value of A to R3
	
	Continuous loop

SAMPLE INPUT/OUTPUT:



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	00	0A	03	E8	00	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 cube of the number has been found.

Experiment No. 14: BCD to ASCII conversion using 8051

AIM:

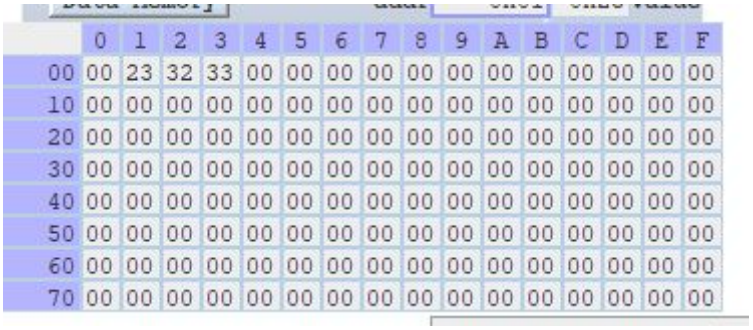
Program to perform BCD to ASCII conversion using 8051.

PROGRAM:

PROGRAM	COMMENTS

MOV A, R1
ANL A, #0FH
ADD A, #30H
MOV R3, A
MOV A, R1
SWAP A
ANL A, #0FH
ADD A, #30H
MOV R2, A
HERE:
SJMP HERE
	Move data from R1 to A
	Bitwise logical AND with 0F
	A=A+30H
	Move data from A to R3
	Move data from R1 to A
	Exchange lower-order and higher-order digits
	Bitwise logical AND with 0F
	A=A+30H
	Move data from A to R2
	
	Continuous loop

SAMPLE INPUT/OUTPUT:



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	00	23	32	33	00	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 ASCII equivalent of each BCD digit has been found.