Exp No: 1 **8-bit Arithmetic Operations**

Date: 19/8/2020

Name: Gokhulnath T
Register Number: 185001051

Aim: To write assembly program to do following 8-bit arithmetic operations:

a) 8-bit Addition
b) 8-bit subtraction
c) 8-bit multiplication
d) 8-bit division

Procedure:

1. Install all the required file for executing MASM programs.(Masm, edit, link, debug etc..).
2. Write the assembly program in any editor before mounting the folder to the MASM.
3. Mount the folder that contains the assembly program with any name such as “d”.
   - mount d e:\masm
4. Create the object file of the assembly program using masm.
   - masm 8BITADD.asm
5. Use the link to create the executable file of the object file created from the above step.
   - Link 8BITADD.obj
6. Run the executable file using debug.
   - debug 8BITADD.exe
7. By un-assembling the program you can check the code segment of the program
   - u 076b-0100
8. To check the data memory segment, you can use the memory option to view the data stored.
   - d 076a-0000
9. To enter your own values, you can use the enter option which will prompt for new values.
   - e 076a-0000
10. To execute the program, you can use go option
    - G
11. After successful execution and termination of the program, you can check the result by checking the data memory segment
    - d 076a-0000
12. The result can be viewed in the respective address mentioned in the program.

1 a) 8-bit addition

Algorithm:

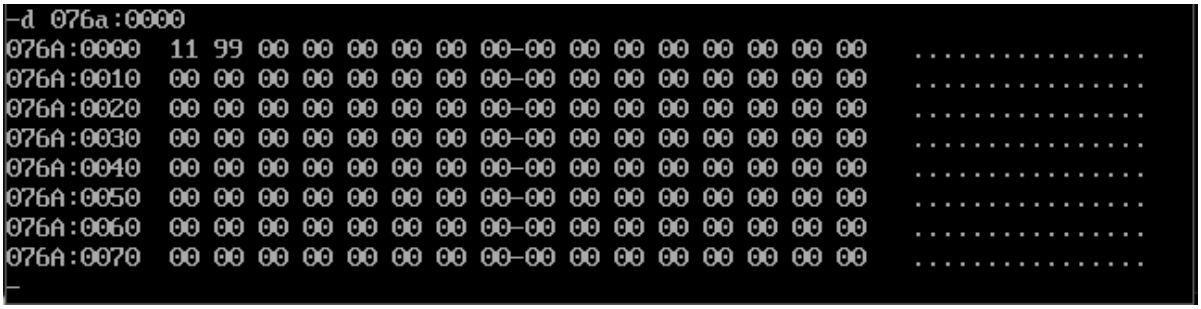
a) Load contents of memory location opr1 in register ah
b) Load contents of memory location opr2 in register bh
c) Assign 00h to ch register
d) Add ah and bh content, store the result in ah
e) If carry is generated continue else jump to step g
f) Increment ch register
g) Load content of ah register to memory location result
h) Load content of ch register to memory location carry
i) Load content 4ch termination code to ah register
j) Stops execution of the program

Program:

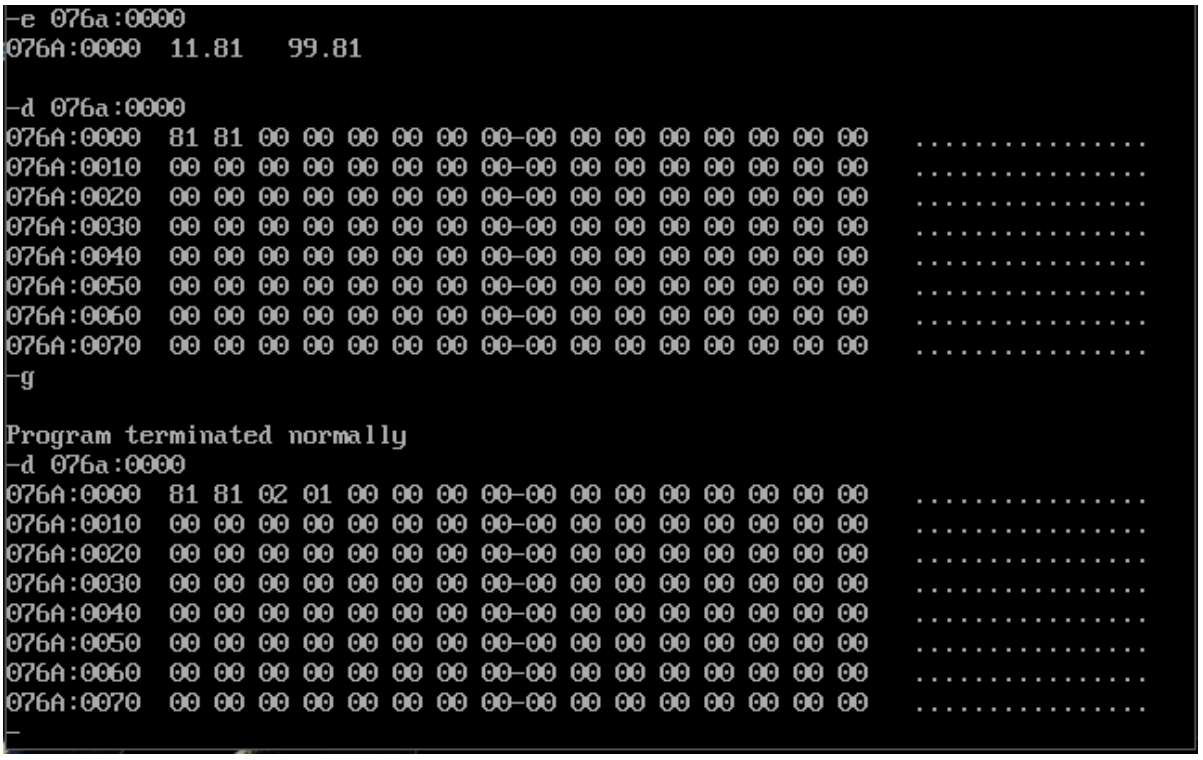
Program	**Comments**
start: mov ax,data	Transferring the data from memory location data to ax
mov ds,ax	Transferring the data from memory location ax to da
mov ah,opr1	Transferring the data from memory location opr1 to ah
mov bh,opr2	Transferring the data from memory location opr2 to bh
mov ch,00h	Transferring the data 00hto ch
add ah,bh	ah = ah + bh
jnc here	Jump if no carry to here
inc ch	Increment ch
here: mov result,ah	Transferring the data from memory location ah to result
mov carry,ch	Transferring the data from memory location ch to carry
mov ah,4ch	Transferring the termination code 4ch to ah
int 21h	Termination
code ends	Code ends

Snapshot of sample input and output:

- **Default value:**



- **Input and output:**



Output explanation: 81+81 = 02 with carry 1

Result:

Thus, the assembly program for 8-bit adder is written and executed.

1 b) 8-bit Subtraction

Algorithm:

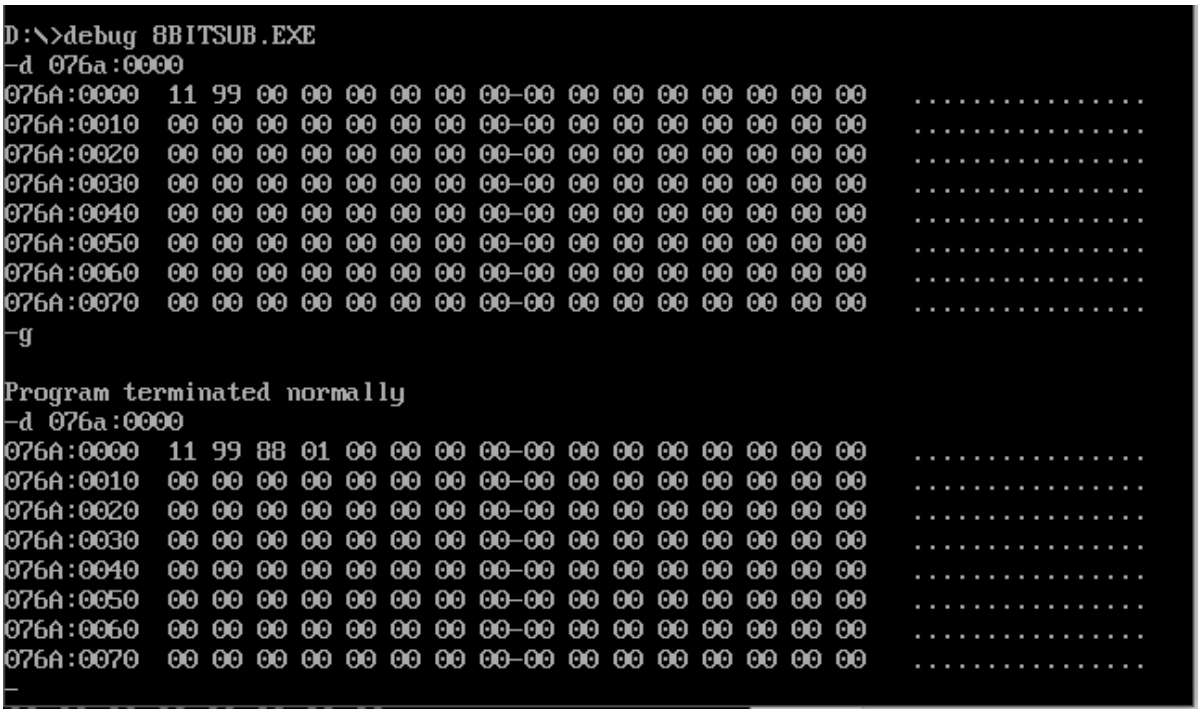
a) Load contents of memory location opr1 in register ah
b) Load contents of memory location opr2 in register bh
c) Assign 00h to ch register
d) sub ah and bh content, store the result in ah
e) If carry is generated continue else jump to step g
f) Increment ch register
g) 2's complement(ah).
h) Load content of ah register to memory location result
i) Load content of ch register to memory location carry
j) Load content 4ch termination code to ah register
k) Stops execution of the program

Program:

Program	**Comments**
start: mov ax,data	Transferring the data from memory location data to ax
mov ds,ax	Transferring the data from memory location ax to da
mov ah,opr1	Transferring the data from memory location opr1 to ah
mov bh,opr2	Transferring the data from memory location opr2 to bh
mov ch,00h	Transferring the data 00hto ch
sub ah,bh	ah = ah - bh
jnc here	Jump if no carry to here
inc ch	Increment ch
neg ah	2's complement(ah).
here: mov result,ah	Transferring the data from memory location ah to result
mov carry,ch	Transferring the data from memory location ch to carry
mov ah,4ch	Transferring the termination code 4ch to ah
int 21h	Termination
code ends	Code ends

Snapshot of sample input and output:

- **Input and output:**



Output explanation: 11+99 = -88 (01 carry indicates negative)

Result:

Thus, the assembly program for 8-bit subtractor is written and executed.

1 c) 8-bit Multiplication

Algorithm:

a) Load contents of memory location opr1 in register al
b) Load contents of memory location opr2 in register bl
c) mul al and bl content, store the result in ax
d) Load contents of memory location ax in result
e) Load content 4ch termination code to ah register
f) Stops execution of the program

Program:

Program	**Comments**
start: mov ax,data	Transferring the data from memory location data to ax
mov ds,ax	Transferring the data from memory location ax to da
mov al,opr1	Transferring the data from memory location opr1 to ah
mov bl,opr2	Transferring the data from memory location opr2 to bh
mul bl	Multiplication AX = AL x BL,
Mov result, ax	al->result; ah->result+1
mov ah,4ch	Transferring the termination code 4ch to ah
int 21h	Termination
code ends	Code ends

Snapshot of sample input and output:

- **Input and output:**

Output explanation: 11*99 = 0A29

Result:

Thus, the assembly program for 8-bit multiplier is written and executed.

1 d) 8-bit Division

Algorithm:

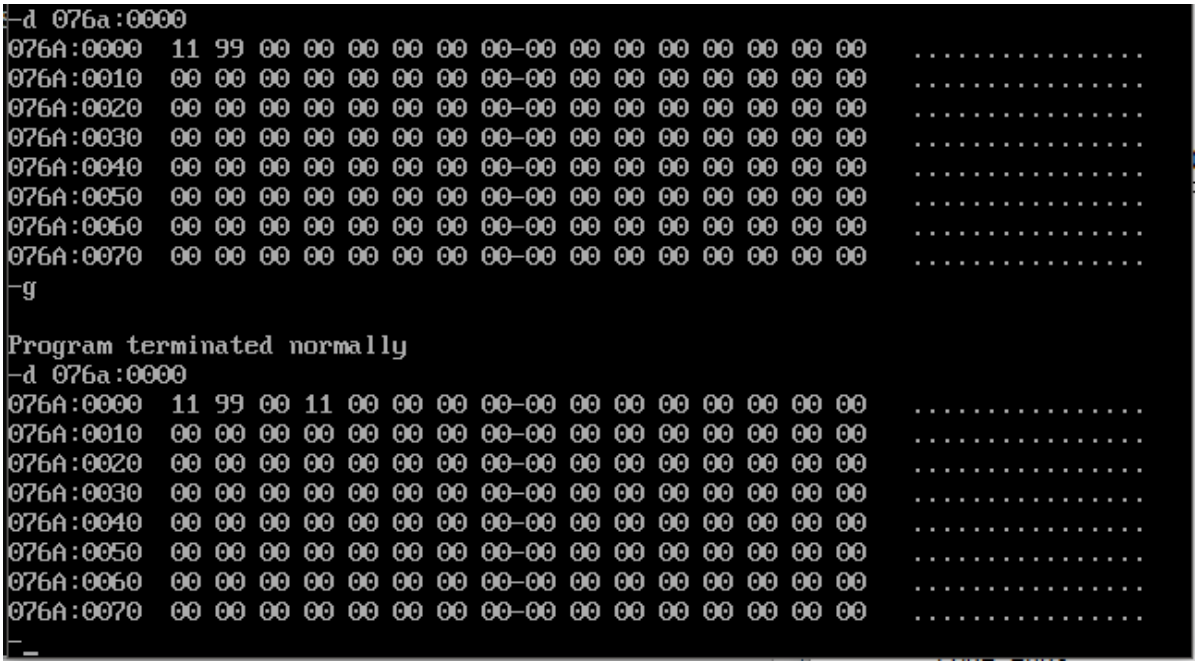
g) Load contents of memory location opr1 in register al
h) Load contents of memory location 00 in register ah
i) div al and bl content, store the result in ax
j) Load contents of memory location al in quotient
k) Load contents of memory location ah in remainder
l) Load content 4ch termination code to ah register
m) Stops execution of the program

Program:

Program	**Comments**
start: mov ax,data	Transferring the data from memory location data to ax
mov ds,ax	Transferring the data from memory location ax to da
mov al,opr1	Transferring the data from memory location opr1 to ah
mov ah,0h	Transferring the data 00 to ah
div opr2	Division AX / BL
mov quotient,al	Transferring the data from memory location al to quotient
mov remainder,ah	Transferring the data from memory location ah to remainder
mov ah,4ch	Transferring the termination code 4ch to ah
int 21h	Termination
code ends	Code ends

Snapshot of sample input and output:

- **Input and output:**



Output explanation: 11/99 = 0 ; remainder 11

Result:

Thus, the assembly program for 8-bit divider is written and executed.