

## Practical - 10

**Problem Statement:** Write X86/64 ALP to perform multiplication of two 8-bit hexadecimal numbers. Use successive addition and add and shift method. (use of 64-bit registers is expected).

### Program:

```
%macro dispmsg 2
mov rax,1
mov rdi,1
mov rsi,%1
mov rdx,%2
syscall
%endmacro

%macro accept 2
mov rax,0
mov rdi,0
mov rsi,%1
mov rdx,%2
syscall
%endmacro

section .data
msg db 10,'Enter two digit Number::'
msg_len equ $-msg
res db 10,'Multiplication of elements is::'
res_len equ $-res
choice db 10,13,'Enter your Choice:'
db 10,13,'1.Successive Addition'
db 10,13,'2.Add and Shift method'
```

```
db 10,13,'3.Exit',10
choice_len equ $-choice
section .bss
num resb 03
num1 resb 01
result resb 04
cho resb 2
section .text
global _start
_start:
mov rax,00
mov rbx,00
mov rcx,00
mov rdx,00
mov byte[result],0
mov byte[num],0
mov byte[num1],0
dispmsg choice,choice_len
accept cho,2
cmp byte[cho],31h
je a
cmp byte[cho],32h
je b
jmp exit
a:call Succe_addition
jmp _start
```

b:call Add\_shift

jmp \_start

exit:

mov rax,60

mov rdi,0

syscall

convert:

mov rbx,00

mov rcx,00

mov rax,00

mov rcx,02

mov rsi,num

up1:

rol bl,04

mov al,[rsi]

cmp al,39h

jg p1

sub al,30h

jmp p2

p1: sub al,37h

p2:add bl,al

inc rsi

loop up1

ret

display:

mov rcx,4

```
mov rdi,result
dup1:
rol bx,4
mov al,bl
and al,0fh
cmp al,09h
jg p3
add al,30h
jmp p4
p3: add al,37h
p4:mov [rdi],al
inc rdi
loop dup1
dispmsg result,4
ret
Succe_addition:
dispmsg msg,msg_len
accept num,3
call convert
mov [num1],bl
dispmsg msg,msg_len
accept num,3
call convert
mov rcx,00
mov rax,00
mov rax,[num1]
```

```
repet:
add rcx,rcx
dec bl
jnz repet
mov [result],rcx
dispmsg res,res_len
mov rbx,[result]
call display
ret
Add_shift:
dispmsg msg,msg_len
accept num,3
call convert
mov [num1],bl
dispmsg msg,msg_len
accept num,3
call convert
mov [num],bl
mov rbx,00
mov rcx,00
mov rdx,00
mov rax,00
mov dl,08
mov al,[num1]
mov bl,[num]
p11:shr bx,01
```

```
jnc p
add cx,ax
p:shl ax,01
dec dl
jnz p11
mov [result],rcx
dispmsg res,res_len
mov rbx,[result]
call display
ret
```

**Output:**

```
atharva@atharva:~$ gedit lab10.asm
atharva@atharva:~$ nasm -f elf64 lab10.asm
atharva@atharva:~$ ld -o lab10 lab10.o
atharva@atharva:~$ ./lab10
```

Enter your Choice:

- 1.Successive Addition
- 2.Add and Shift method
- 3.Exit

1

Enter two digit Number::40

Enter two digit Number::12

Multiplication of elements is::0480

Enter your Choice:

- 1.Successive Addition
- 2.Add and Shift method

3.Exit

2

Enter two digit Number::10

Enter two digit Number::10

Multiplication of elements is::0100

Enter your Choice:

1.Successive Addition

2.Add and Shift method

3.Exit

3