

Assignment #03 – COAL – AI -3-1

Saghir Ali – 67005

Simple Calculator in EMU 8086:

This is a simple and efficient calculator program developed by Saghir Ali using x86 assembly language.

It provides basic arithmetic operations in a clean, menu-driven interface that is easy to use and understand.

Working:

The calculator starts by displaying Saghir Ali as name followed by a menu of available operations.

Users select their desired operation by entering corresponding numbers (1-4). The program then prompts for two single-digit inputs, performs the selected arithmetic operation, and instantly displays the calculated result.

The operations are

- 1. Addition**
- 2. Subtraction**
- 3. Multiplication**
- 4. Division**

CODE:

```
emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator converter options help about

001 .model small
002 .stack 100h
003
004 .data
005     msg0 db 'Saghir Ali$'
006     msg1 db 10,13,10,13,'1:Addition$'
007     msg2 db 10,13,'2:Subtraction$'
008     msg3 db 10,13,'3:Multiplication$'
009     msg4 db 10,13,'4:Division$'
010     msg5 db 10,13,'Choose Any One:$'
011     msg6 db 10,13,10,13,'Enter 1st Number:$'
012     msg7 db 10,13,'Enter 2nd Number:$'
013     msg8 db 10,13,10,13,'The Result is:$'
014
015     num1 db ?
016     num2 db ?
017     result db ?
018
019 .code
020 main proc
021     mov ax, @data
022     mov ds, ax
023     lea dx, msg0
024     mov ah, 9
025     int 21h
026
027     lea dx, msg1
028     mov ah, 9
029     int 21h
030
031     lea dx, msg2
032     mov ah, 9
033     int 21h
034
035     lea dx, msg3
036     mov ah, 9
037     int 21h
038
039     lea dx, msg4
040     mov ah, 9
041     int 21h
042
043     lea dx, msg5
044     mov ah, 9
045     int 21h
046
```

```
emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator converter options help about

091     mov al, num1
092     sub al, num2
093     mov result, al
094     jmp display_result
095
096 multiplication:
097     mov al, num1
098     mul num2
099     mov result, al
100     jmp display_result
101
102 division:
103     mov al, num1
104     mov ah, 0
105     div num2
106     mov result, al
107     jmp display_result
108
109 display_result:
110     lea dx, msg8
111     mov ah, 9
112     int 21h
113
114     mov dl, result
115     add dl, 48
116     mov ah, 2
117     int 21h
118
119 exit_program:
120     mov ah, 4ch
121     int 21h
122
123 main endp
124 end main
```

Working Operations:

1. Addition

The screenshot shows the emu8086 interface. The assembly code in the editor window is:

```
.model smz
.stack 100
.data
    msg0 db '1:Addition',0
    msg1 db '2:Subtraction',0
    msg2 db '3:Multiplication',0
    msg3 db '4:Division',0
    msg4 db 'Choose Any One:',0
    msg5 db 'Enter 1st Number:',0
    msg6 db 'Enter 2nd Number:',0
    msg7 db 'The Result is:',0
    msg8 db ' ',0
    num1 db 0
    num2 db 0
    result db 0
.code
main proc
    mov ah, 0
    mov dx, msg4
    mov ah, 9
    int 21h
    mov dl, result
    add dl, 48
    mov ah, 2
    int 21h
    exit_program:
    mov ah, 4ch
    int 21h
endp
```

The output window shows the program running and printing "The Result is:8". The assembly window shows the instruction at address F400:0204 is INT 021h.

2. Subtraction

The screenshot shows the emu8086 interface. The assembly code in the editor window is identical to the addition code, except for the message strings and the result value:

```
.model smz
.stack 100
.data
    msg0 db '1:Addition',0
    msg1 db '2:Subtraction',0
    msg2 db '3:Multiplication',0
    msg3 db '4:Division',0
    msg4 db 'Choose Any One:',0
    msg5 db 'Enter 1st Number:',0
    msg6 db 'Enter 2nd Number:',0
    msg7 db 'The Result is:',0
    msg8 db ' ',0
    num1 db 0
    num2 db 0
    result db 0
.code
main proc
    mov ah, 0
    mov dx, msg4
    mov ah, 9
    int 21h
    mov dl, result
    add dl, 48
    mov ah, 2
    int 21h
    exit_program:
    mov ah, 4ch
    int 21h
endp
```

The output window shows the program running and printing "The Result is:5". The assembly window shows the instruction at address F400:0204 is INT 021h.

3. Multiplication:

The screenshot shows the emu8086 interface with the assembly code for multiplication. The code includes a menu bar with file, edit, bookmarks, assembler, emulator, math, ascii codes, help, and a toolbar with new, open, exam, and emulator screen (80x25 chars). The assembly code is as follows:

```
.model smz Saghir Ali
.stack 10h
.data
    msg0 db 1;Addition
    msg1 db 2;Subtraction
    msg2 db 3;Multiplication
    msg3 db 4;Division
    msg4 db "Choose Any One:3"
    msg5 db "Enter 1st Number:3"
    msg6 db "Enter 2nd Number:2"
    msg7 db "The Result is:6"
    num1 db ?
    num2 db ?
    result db ?

.code
main proc
    mov ax, 0
    mov dx, msg2
    mov ah, 9
    int 21h
    lea dx, msg8
    mov ah, 9
    int 21h
    mov dl, result
    add dl, 48
    mov ah, 2
    int 21h
    mov ah, 4ch
    int 21h
end proc
```

The assembly code is displayed in the left pane, and the CPU register dump (IP, SS, SP, BP, SI, DI, DS, ES) and memory dump (F400:0204 to F400:0204) are shown in the right pane. The memory dump shows the INT 021h instruction at address F400:0204.

4. Division:

The screenshot shows the emu8086 interface with the assembly code for division. The code is identical to the multiplication code, except for the selection of division in the menu. The assembly code is as follows:

```
.model smz Saghir Ali
.stack 10h
.data
    msg0 db 1;Addition
    msg1 db 2;Subtraction
    msg2 db 3;Multiplication
    msg3 db 4;Division
    msg4 db "Choose Any One:4"
    msg5 db "Enter 1st Number:8"
    msg6 db "Enter 2nd Number:2"
    msg7 db "The Result is:4"
    num1 db ?
    num2 db ?
    result db ?

.code
main proc
    mov ax, 0
    mov dx, msg3
    mov ah, 9
    int 21h
    lea dx, msg8
    mov ah, 9
    int 21h
    mov dl, result
    add dl, 48
    mov ah, 2
    int 21h
    mov ah, 4ch
    int 21h
end proc
```

The assembly code is displayed in the left pane, and the CPU register dump (IP, SS, SP, BP, SI, DI, DS, ES) and memory dump (F400:0204 to F400:0204) are shown in the right pane. The memory dump shows the INT 021h instruction at address F400:0204.