**Andrew Tran**

**Lab Experience Eleven**

**Objectives:**

1. Understanding the shift instructions.
2. Understanding multiplication and division

**What you are to do:**

1. Show all work necessary to do the problem. Show the original contents of the register before the code executes and also show the contents of the register after the code executes. Also show the contents of any of the affected flags. For parts a and b only, show the contents of the register in binary and hex. For all other parts of the exercise use hex.
   1. mov al, 0D4h

shr al, 1 **; al, 0110 1010b or 6Ah**

mov al, 0D4h

sar al, 1 **; al, 1110 1010b or EAh**

mov al, 0D4h

sar al, 4 **; al. 1111 1101b or FDh**

mov al, 0D4h

rol al, 1 **; al, 1010 1001b or A9h**

* 1. mov al, 0D4h

ror al, 3 **;al, 1001 1010b or 9Ah**

mov al, 0D4h

rol al, 7 **;al, 0110 1010b or 6Ah**

stc **;carry flag 1**

mov al, 0D4h

rcl al, 1 **;al, 1010 1001b, or A9h**

stc **;carry flag 1**

mov al, 0D4h

rcr al, 3 **;al, 1001 1010b, or 9Ah**

* 1. mov dx, 0

mov ax, 222h

mov cx, 100h

mul cx **;ax, 2200h | dx, 2h**

* 1. mov ax, 63h

mov bl, 10h

div bl **;ax, 306h | al, 06h | ah, 03h**

1. Write a sequence of shift instructions that cause AX to be sign extended into EAX. Do not use the cwd instruction.

Mov ax, FFFFh

Shl eax, 16

Sar eax, 16

1. Suppose the instruction set of the CPU contains no rotate instructions. Write the assembler commands to replicate the following instruction:

ror al, 1

jnc haha

add al, 80h

haha:

1. Write a logical shift instruction that multiplies the contents of EAX by 16.

Shl eax, 4

1. Write a logical shift instruction that divides EBX by 4.

Shr ebx, 2

1. Write a single rotate instruction that exchanges the high and low halves of the DL register.

Rol dl, 4

1. Write a single SHLD instruction that shifts the highest bit of the AX register into the lowest bit position of DX and shifts DX one bit to the left.

Shld dx, ax, 1

1. Write instructions to divide -276 by 10 and store the result in a 16 bit variable val1.

Mov ax, -276

Cwd

Mov bx, 10

Idiv bx

Mov val1, ax

1. Implement the following C++ expression in assembly language using 32 bit unsigned operands.

val1 = (val2 / val3) \* (val1 + val2);

move eax, val2

mov ebx, val3

div ebx

mov ebx,val1

add ebx, val2

mul ebx

mov val1, eax

1. Write a procedure that displays an unsigned 8 bit binary value in decimal format. Pass the binary value in the register al. The input range is limited to 0 through 99, decimal. The only procedure you call from the book’s link library is WriteChar. The procedure should contain approximately 8 instructions. Below is an example call:

mov al, 65 ; range limit is 0 to 99

call showDecimal18

showDecimal18 PROC

or ax, 3030h

mov al,ah

call WriteInt

ret

showDecimal18 ENDP

1. Do problem one on page 284.
   1. Copy and paste the program into your word document along with a screen shot of the execution of your program.

INCLUDE Irvine32.inc

DECIMAL\_OFFSET = 5

.data

array BYTE "100123456789765"

data BYTE(SIZEOF array + 1) DUP(? )

.code

main PROC

mov edx, OFFSET array; pass in offset in edx

mov ecx, 15; pass in number length in ecx

mov ebx, DECIMAL\_OFFSET; pass in decimal offset in ebx

mov edx, OFFSET array

call WriteString

call crlf

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

call WriteScaled

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

mov edx, OFFSET data

call WriteString

call crlf

main ENDP

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

WriteScaled PROC

sub ecx, ebx

mov esi,0

L1:

mov ah, array[esi]; loop in array into data

mov data[esi], ah

inc esi

loop L1

mov data[esi], '.'; add in period

inc esi

mov ecx,5 set to decimal offset

L2:

mov ah, array[esi - 1]; put in the rest of array

mov data[esi], ah

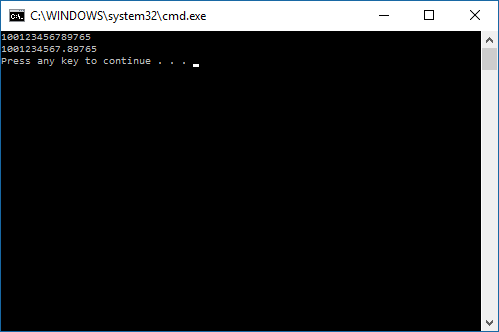
inc esi

loop L2

ret

WriteScaled ENDP

END main



**What to hand in:**

1. A print out of the assembler program in problem 11.

1. A printout of the word document containing the answers to the problems assigned.
2. Compress the assembler file into a single file using your name and the lab number as the filename.
3. Place the compressed file into the D2L Assignment folder titled Lab Experience Eleven.

**Due Date: One week after your lab session.**