Київський національний університет імені Тараса Шевченка

Факультет радіофізики, електроніки і комп’ютерних систем

Звіт

З лабораторної роботи №2

Предмету «Комп’ютерні системи»

Студента 3-го курсу

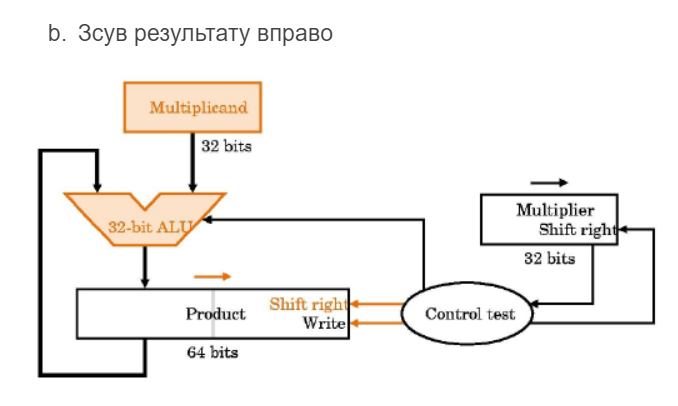
Спеціальності КІ МА

Єрмоленка Романа

Київ 2022

Множення

Варіант



Код

.686

.model flat, stdcall

include \masm32\include\msvcrt.inc

include \masm32\include\kernel32.inc

includelib \masm32\lib\msvcrt.lib

includelib \masm32\lib\kernel32.lib

.data

X word 00F00Dh  ; Result = X \* Y

Y word 00BABAh  ;

fmt  DB  "Result: %lu", 13, 10, 0

.code

\_start:

mov AX, X ; Put X in EAX

mov BX, Y ; Put Y in ECX

shl EBX, 16 ; Move Y to be in proper place for the algorythm

xor EDX, EDX ; Zero EDX

mov ECX, 16

\_mult:

xor ESI, ESI

test AX, 1 ; Check if bit of X is 1

jz mult\_nop

add EDX, EBX ; If 1 add Y

jnc mult\_nop

mov ESI, 80000000h

mult\_nop:

shr EAX, 1

shr EDX, 1

or EDX, ESI

loop \_mult

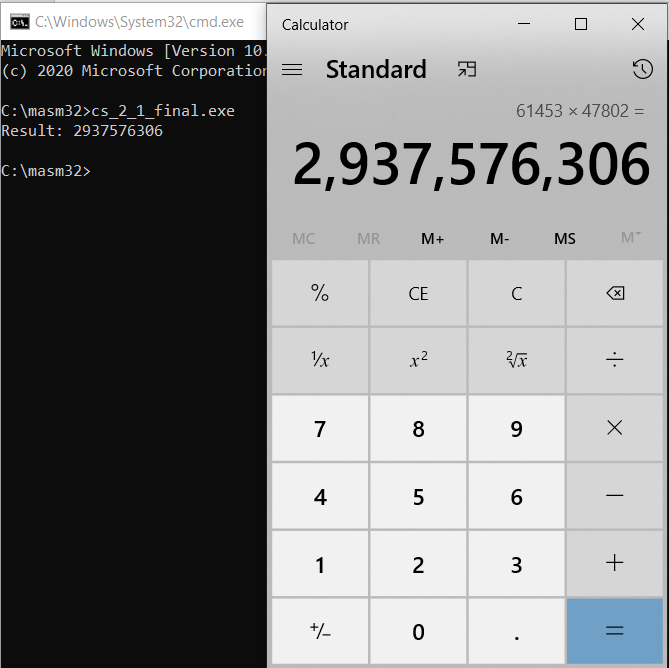
invoke crt\_printf, addr fmt, EDX

invoke ExitProcess, 0

ret

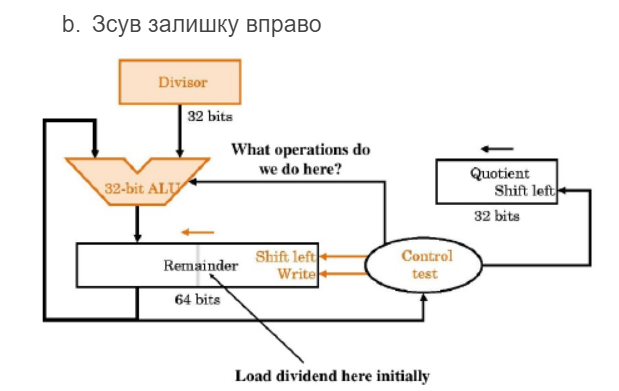
end \_start

Результат



Ділення

Варіант



Код

.686

.model flat, stdcall

include \masm32\include\msvcrt.inc

include \masm32\include\kernel32.inc

includelib \masm32\lib\msvcrt.lib

includelib \masm32\lib\kernel32.lib

.data

X WORD 1337  ; Result = y/x

Y WORD 228   ;

fmt\_rs  DB  "Result: %d", 13, 10, 0

fmt\_rm  DB  "Remainder: %d", 13, 10, 0

.code

\_start:

mov ECX, 16

movzx EAX, X ; Put X in EAX

mov DX, Y ; Put y in BD

shl EDX, 16 ; Shift EDX 16 to the left

xor EBX, EBX ; Zero EBX

\_div:

shl EAX, 1 ; Shift EAX and EDX left

shl EBX, 1 ;

cmp EAX, EDX

jb less

sub EAX, EDX ; sub if EAX > EDX

or EBX, 1 ; and set lsb of EDX

less:

loop \_div

shr EAX, 16

push EAX

invoke crt\_printf, addr fmt\_rs, EBX

pop EAX

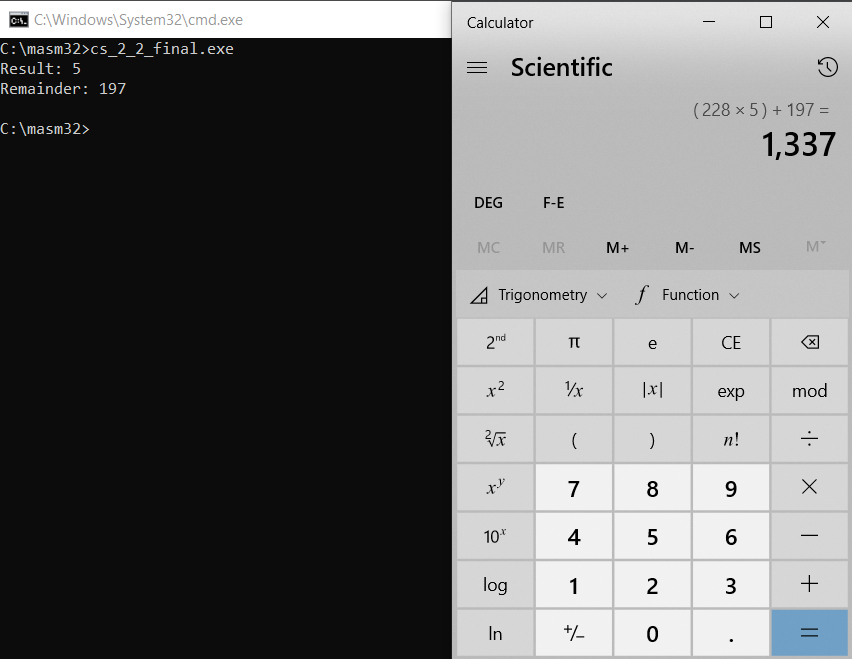
invoke crt\_printf, addr fmt\_rm, EAX

invoke ExitProcess, 0

ret

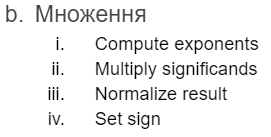
end \_start

Результат



Робота з IEEE 754 Floating Point

Варіант



Код

.686

.model flat, stdcall

include \masm32\include\msvcrt.inc

include \masm32\include\kernel32.inc

includelib \masm32\lib\msvcrt.lib

includelib \masm32\lib\kernel32.lib

.data

X dword 43640000h ; Result = X \* Y

Y dword 44a72000h ;

fmt  DB  "Result: %lx", 13, 10, 0

.code

\_start:

; Initialize all required registers

mov EAX, X   ; First float

mov EDX, Y   ; Second float

mov EBX, EAX ; Copy of X for calculations

mov ESI, EDX ; Copy of Y for calculations

mov EDI, EDX ; Copy of Y for calculations

; Calculate sign

xor EDI, EBX

and EDI, 80000000h

; Add Exponents

    ; Remove signs

shl EBX, 1

shl ESI, 1

    ; Cut exponents

shr EBX, 24        ; Get exponent of X

shr ESI, 24        ; Get exponent of Y

    ; Addition

add ESI, EBX      ; Add exponents together

sub ESI, 127       ; Substract static shift

; Multiply Mantisas

and EAX , 7FFFFFh ; Get X Mantissa

or EAX, 800000h   ;

and EDX , 7FFFFFh ; Get Y Mantissa

or EDX, 800000h   ;

mul EDX ; Multiply

shld EDX, EAX, 9 ; Move result to proper place

; Normalize Exponents

test EDX, 1000000h ; Check if we have to normalize

jz normalize\_end

shr EDX, 1 ; Move result to proper place

inc ESI ; And normalize Exponent

normalize\_end:

and EDX, 007FFFFFh ; Cut 1

; Finishing up Exponent addition

and ESI, 000000FFh ; Remove overflows if occured

shl ESI, 23        ; Move exponent back to its proper place

or EDX, EDI ; Set Sign

or EDX, ESI ; Set Exponent

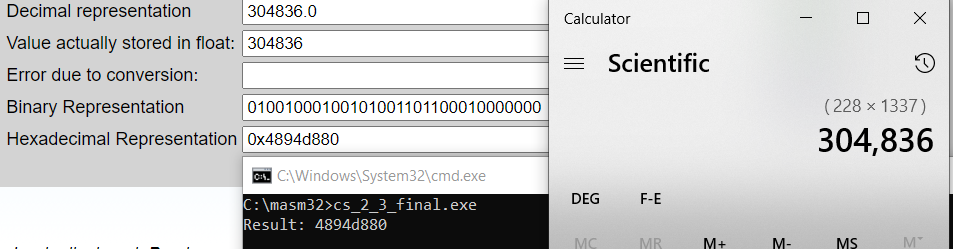
invoke crt\_printf, addr fmt, EDX

invoke ExitProcess, 0

ret

end \_start

Результат



Висновок

Протягом лабораторної роботу було реалізовно алгоритми множення та ділення, що імітують алгоритм роботи АЛУ. Також була здійснена робота з IEEE 754 Floating Point, виконана операція множення між числами представленими у цьому форматі.