Лабораторная работа №10.
Основы работы с FPU

Макаров П. А.

1 Краткая теория

1. Raymond Filiatreault. SIMPLY FPU. Tutorial;
2. x86 Assembly/Floating Point — Wikibooks;
3. Online Binary-Decimal Converter;
4. В. Л. Яшкардин. IEEE 754 — стандарт двоичной арифметики;
5. IEEE 754-2008 — Википедия;

2 Задания для самостоятельного решения

1. Напишите программу, определяющую наличие математического сопроцессора. Используйте информацию, доступную в CMOS-памяти.
2. Исследуйте работу следующей программы в отладчике.

```
; simple FPU example
; y = sqrt(x)
; z = rint(y)
; x, y - is doubles
; z - is int

org 0x100

start:
finit
fld qword [x]
fsqrt
```

```
fst qword [y]
fistp dword [z]

quit:
mov ax, 0x4C00
int 0x21

x dq 6.25 ; try differ values: 6.26, 2.0, -1.0, ...
y dq 0.0
z dd 0
```

3. Напишите, транслируйте и изучите принцип работы следующей программы.

```
org 0x100

jmp start

N     dw 640
xMax  dw 639
yMax  dw 199
x     dw 0
y     dw 0
xx    dq -3.1415
xxStp dq 0.0
yyLen dq 2.0

start:   finit
findStp: fld qword [xx]
         fabs
         fadd st0, st0
         fidiv word [N]
         fstp qword [xxStp]

initScr: mov ah, 0
         mov al, 0xE
         int 0x10

         mov cx, [N]
calcY:   fld qword [xx]
         fsin
         fld1
```

```
          faddp
          fdiv qword [yyLen]
          fild word [yMax]
          fmulp
          fild word [yMax]
          fxch
          fsubp
          fistp word [y]
setDot:   push cx
          mov ah, 0xC
          mov bh, 0
          mov al, 5
          mov cx, [x]
          mov dx, [y]
          int 0x10
          pop cx
calcX:    mov ax, [x]
          inc ax
          mov word [x], ax
          fld qword [xx]
          fld qword [xxStp]
          fadd
          fstp qword [xx]
          loop calcY

waitKey:  mov ah, 0
          int 0x16

restScr:  mov ah, 0
          mov al, 3
          int 0x10

quit:     mov ax, 0x4c00
          int 0x21
```

4. **Вычисление площади треугольника по формуле Герона.** Даны три стороны треугольника a, b и c, хранящиеся в памяти как 64-битные числа с плавающей точкой (`double`). Необходимо написать фрагмент программы на ассемблере x87 FPU, который вычисляет площадь треугольника по формуле Герона:

$$S = \sqrt{p(p-a)(p-b)(p-c)}, \quad \text{где } p = \frac{a+b+c}{2}$$

- Опишите, как будет организована работа со стеком FPU для решения этой задачи. Сколько элементов стека потребуется?
- Напишите код, предполагая, что адреса переменных a, b, c и S известны. Используйте инструкции x87 FPU в синтаксисе NASM.

5. В регистрах `ST(0)` и `ST(1)` x87 FPU находятся два числа с плавающей точкой (`double`). Необходимо реализовать логику:

   - Если `ST(0) > ST(1)`, то установить регистр `AX` в 1.
   - Если `ST(0) == ST(1)`, то установить `AX` в 0.
   - Если `ST(0) < ST(1)`, то установить `AX` в -1.

   Объясните, почему для сравнения чисел с плавающей точкой нельзя использовать инструкции центрального процессора? Какие инструкция FPU для этого предназначены и как они влияют на флаги CPU? Напишите фрагмент кода, реализующий требуемую логику, используя инструкции `FCOM` и `FSTSW`.