**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №6

по дисциплине «Архитектура ЭВМ»

на тему: «Исследование арифметического сопроцессора»

Выполнил: студент гр. ИТП-21

Воробьев В. А.

Принял: преподаватель

Савельев В. А.

Гомель 2022

**Цель работы:** знакомство с работой арифметического сопроцессора, исследование инструкций сопроцессор, получение навыков работы с отладчиком.

**Ход работы**

**Задание:**

1. Согласно номеру студента в журнале группы выбрать вариант задания и написать программу вычисления заданного выражения на языке Ассемблера. Задание согласно варианта: вычислить 6 значений функции Yn = x \* 4 / (x + 5) (х изменяется от 3 с шагом 1,25). Код программы для вычисления заданного выражения представлен ниже:

.686

.model flat, stdcall

option casemap: none

; Библиотеки и подключаемые файлы проекта

;--------------------------------------------

include C:\masm32\include\windows.inc

include C:\masm32\include\kernel32.inc

include C:\masm32\include\user32.inc

include C:\masm32\include\fpu.inc

; содержит прототип функции FpuFLtoA

includelib C:\masm32\lib\kernel32.lib

includelib C:\masm32\lib\user32.lib

includelib C:\masm32\lib\fpu.lib

; Сегмент даных

;--------------------------------------------

.data ; инициализированные данные

MsgBoxTitle byte "Операции в сопроцессоре x87", 0

; заголовок окна

MsgBoxText db "Yn = 4 \* x / (x + 5) (х изменяется от 3 с шагом 1,25", 13, 13,

"y1="

res1 db 14 DUP(0), 10, 13 ; зарезервировать 14 байт для первого

; результата и поместить туда 0

db "y2="

res2 db 14 DUP(0), 10, 13

db "y3="

res3 db 14 DUP(0), 10, 13

db "y4="

res4 db 14 DUP(0), 10, 13

db "y5="

res5 db 14 DUP(0), 10, 13

db "y6="

res6 db 14 DUP(0), 10, 13

CrLf equ 0A0Dh

y1 dt 0.0 ; тип 80 бит без знака (TBYTE = dt)

y2 dt 0.0

y3 dt 0.0

y4 dt 0.0

y5 dt 0.0

y6 dt 0.0

val dd 0.0

zer dd 0.0

x dd 3.0 ; тип 32 бита без знака (DWORD = dd)

op1 dd 4

op2 dd 5

zero dd 0.0

step dd 1.25

.data? ; неинициализированные данные

.const ; константы

; Cегмент кода

.code

start: ; метка (точка входа в программу)

finit ; инициализация регистров FPU

; (CWR = 037Fh, SWR = 0h, TWR = FFFFh,

; DPR = 0h, IPR = 0h)

mov ecx, 6 ; счётчик X

fld x

m1: ; метка начала цикла

fld val

fmul zer

fadd op1

fmul x

fld op2

fadd x

fdivrp st(1), st(0)

fld x ; увелечение X на величину шага

fadd step

fstp x

loop m1

; переходим на m1

fstp y6

fstp y5 ; сохраняем стек в память

fstp y4

fstp y3

fstp y2

fstp y1

; преобразование результатов вычислений в массив символов

invoke FpuFLtoA, addr y1, 10, addr res1, SRC1\_REAL or SRC2\_DIMM

mov word ptr res1 + 14, CrLf

invoke FpuFLtoA, addr y2, 10, addr res2, SRC1\_REAL or SRC2\_DIMM

mov word ptr res2 + 14, CrLf

invoke FpuFLtoA, addr y3, 10, addr res3, SRC1\_REAL or SRC2\_DIMM

mov word ptr res3 + 14, CrLf

invoke FpuFLtoA, addr y4, 10, addr res4, SRC1\_REAL or SRC2\_DIMM

mov word ptr res4 + 14, CrLf

invoke FpuFLtoA, addr y5, 10, addr res5, SRC1\_REAL or SRC2\_DIMM

mov word ptr res5 + 14, CrLf

invoke FpuFLtoA, addr y6, 10, addr res6, SRC1\_REAL or SRC2\_DIMM

; вывод результатов вычислений

invoke MessageBox, NULL, addr MsgBoxText, addr MsgBoxTitle,

MB\_ICONINFORMATION

invoke ExitProcess, NULL ; функция завершения с параметром NULL

end start

2. Выполнить программу в пошаговом режиме в отладчике и заполнить таблицу, отображающую состояние регистров и флагов сопроцессора.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Шаг | Регистр | Стек | | | | | |
| ECX | St0 | St1 | St2 | St3 | St4 | St5 |
| 0 | 768E346B | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 6 | 3 | 0 | 0 | 0 | 0 | 0 |
| 2 | 6 | 3 | 3 | 0 | 0 | 0 | 0 |
| 3 | 6 | 2.1 | 3 | 0 | 0 | 0 | 0 |
| 4 | 6 | 3 | 2.1 | 3 | 0 | 0 | 0 |
| 5 | 6 | 1.47 | 3 | 0 | 0 | 0 | 0 |
| 6 | 6 | 3 | 1.47 | 3 | 0 | 0 | 0 |
| 7 | 6 | 4.25 | 1.47 | 3 | 0 | 0 | 0 |
| 8 | 6 | 2.97 | 1.47 | 3 | 0 | 0 | 0 |
| 9 | 6 | 4.25 | 2.97 | 1.47 | 3 | 0 | 0 |
| 10 | 6 | 1.47 | 1.47 | 3 | 0 | 0 | 0 |
| 11 | 6 | 4.25 | 1.42 | 1.42 | 1.42 | 3 | 0 |
| 12 | 6 | 5 | 1.42 | 1.42 | 3 | 0 | 0 |
| 13 | 6 | 5.5 | 1.42 | 1.42 | 3 | 0 | 0 |
| 14 | 6 | 3.85 | 1.42 | 1.42 | 3 | 0 | 0 |
| 15 | 6 | 5.5 | 3.85 | 1.42 | 1.42 | 3 | 0 |
| 16 | 6 | 6.75 | 1.42 | 1.42 | 1.42 | 3 | 0 |
| 17 | 6 | 4.72 | 1.42 | 1.42 | 1.42 | 3 | 0 |
| 18 | 6 | 6.75 | 4.72 | 1.42 | 1.42 | 1.42 | 0 |
| 19 | 6 | 8 | 1.42 | 1.42 | 1.42 | 1.42 | 0 |
| 20 | 6 | 5.6 | 1.42 | 1.42 | 1.42 | 1.42 | 3 |

Таблица 1 – Значения регистров

**Вывод**: в ходе работы была изучена работа арифметического сопроцессора, исследованы инструкции сопроцессора, получены навыки работы с отладчиком, также была написана программа, которая вычисляет арифметическое выражение.