МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ АВТОМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

КАФЕДРА РАДИОЭЛЕКТРОННЫХ СРЕДСТВ

Отчет по дисциплине

«Цифровые устройства и микропроцессоры»

Лабораторная работа №3

«Использование математического сопроцессора»

Вариант №14

Выполнил: студент группы ИКТб– 31\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А. С. Шабардин

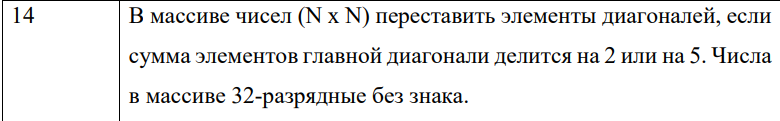
Проверил: доцент кафедры РЭС \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ М.А. Земцов

Киров 2022

Цель работы: изучение принципов выполнения арифметических команд с помощью математического сопроцессора FPU микропроцессоров с архитектурой x86.

1.Исходные данные представлены в таблице 1.

Таблица 1 – Исходные данные



2.Текст программы:

.686

.model flat,stdcall

.stack 100h

.data

Array dd 42, 51, 36, 185, 149, 12, 192, 241, 156, 11, 14, 98, 34, 14, 17, 4; задание массива

ArrayLength equ ($-Array); определение длины массива

NumberOfElementsInString dd ?; количество элементов в строке массива

MainDiagonalElementNumber dd ?; начальный адресс элемента главной диагонали

ChangeDiagonalElement dd ?; начальный адресс элемента неглавной диагонали массива

CurrentNumberOfElement dd ?; адресс текущего элемента в массиве

MainDiagonalSum dw ?;

CurrentNumber dd ?

Variable dd ?;

.code

ExitProcess PROTO STDCALL :DWORD

Start:

finit ;освобождает все регистры сопроцессора

mov NumberOfElementsInString, 4

mov MainDiagonalElementNumber, 0

mov ChangeDiagonalElement, 4

mov CurrentNumberOfElement, 1

mov MainDiagonalSum, 0

mov CurrentNumber, 0

xor eax,eax

xor ebx,ebx

xor edx,edx

xor ecx,ecx

mov ebx, 20

mov ecx, 4

AddToRegister:

add edx, ecx

sub edx, 1

mov eax, edx

mul ebx

fild Array[eax]

loop AddToRegister

mov ecx, 4

mov eax, 0

SumFinding:

fadd st(0), st(1)

fadd st(0), st(2)

fadd st(0), st(3)

fist MainDiagonalSum

DivisionCheck2:

test MainDiagonalSum, 2

jnz DivisionCheck5

jmp SwitchDiagonalElements

DivisionCheck5:

test MainDiagonalSum, 5

jnz ExitProgram

jmp SwitchDiagonalElements

SwitchDiagonalElements:

mov ebx, 20

mov eax, ecx

sub eax, 1

mul ebx; расчет номера элемента главной диагонали

mov edx, Array[eax]

mov ChangeDiagonalElement, edx

mov CurrentNumber, ecx

mov Variable, eax

mov eax, ecx

mov ebx, 12

mul ebx

mov ecx, eax

mov eax, Variable

mov Variable, ecx; расчет номера побочной диагонали

mov ecx, Array[ecx]

mov Array[eax], ecx

mov ecx, Variable

mov edx, ChangeDiagonalElement

mov Array[ecx], edx

mov ecx, CurrentNumber; меняем местами элементы диагонали

loop SwitchDiagonalElements

ExitProgram:

mov eax, Array[0]

mov ebx, Array[4]

mov ecx, Array[8]

mov edx, Array[12]

mov eax, Array[16]

mov ebx, Array[20]

mov ecx, Array[24]

mov edx, Array[28]

mov eax, Array[32]

mov ebx, Array[36]

mov ecx, Array[40]

mov edx, Array[44]

mov eax, Array[48]

mov ebx, Array[52]

mov ecx, Array[56]

mov edx, Array[60]

exit:

Invoke ExitProcess,1

End Start

3. Верификация программы

Зададим массив размером 4x4

42 51 36 185

149 12 192 241

156 11 14 98

34 14 17 4

В шестнадцатеричной системе счисления массив будет выглядеть:

2A 33 24 B9

95 C C0 F1

9C B E 62

22 E 11 4

Сумма элементов на главной диагонали равна:

42 + 12 + 14 + 4 = 72

72 делится на 2 => меняем элементы диагоналей местами

Полученный с помощью расчётов массив:

B9 33 24 2A

95 C0 C F1

9C E B 62

4 E 11 22

На рисунках 1-4 представлен результат выполнения программы.



Рисунок 1 – Первая строка массива



Рисунок 2 – Вторая строка массива



Рисунок 3 – Третья строка массива



Рисунок 4 – Четвертая строка массива

Полученный с помощью программы массив:

B9 33 24 2A

95 C0 C F1

9C E B 62

4 E 11 22

**Вывод:** в ходе лабораторной работы были изучены операции, которые можно выполнять с помощью математического сопроцессора. Программа прошла верификацию ручным расчетом, а это значит, что она выполнена правильно.