**МИНОБРНАУКИ РОССИИ**

**Федеральное государственное автономное образовательное  
 учреждение высшего образования   
«Южный федеральный университет»**

**Институт высоких технологий и пьезотехники**

**Кафедра прикладной информатики и инноватики**

**Направление подготовки:   
09.03.03 "Прикладная информатика"**

**Отчет по Лабораторной работе №2**

**«Простейшая программа на языке ассемблера»**

**По дисциплине «Основы функционирования вычислительной техники»**

**Выполнил:**

студент 2 курса 6 группы

Куракин Н. А.

\_\_\_\_\_\_\_\_\_\_\_\_

*подпись*

**Проверил:**

Преподователь Толмачев С.А.

Ростов-на-Дону

2024

# **ЛАБОРАТОРНАЯ РАБОТА №2**

**Тема:** «Запись программ на языке ассемблера».

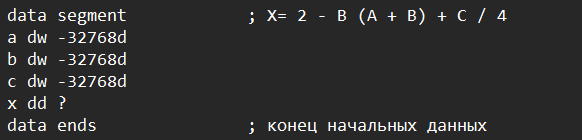
**Цель работы:** изучить конструкции программ на языке ассемблера, принципов действия и отладки программ на языке ассемблера.

**Ход работы:**

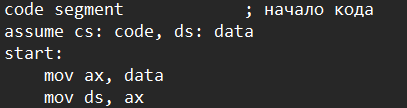
Ознакомился с теоретической частью и приступил к выполнению задания. Необходимо было разработать программу, реализующую формулу 9:

X = 2 - B(A+B) + C/4

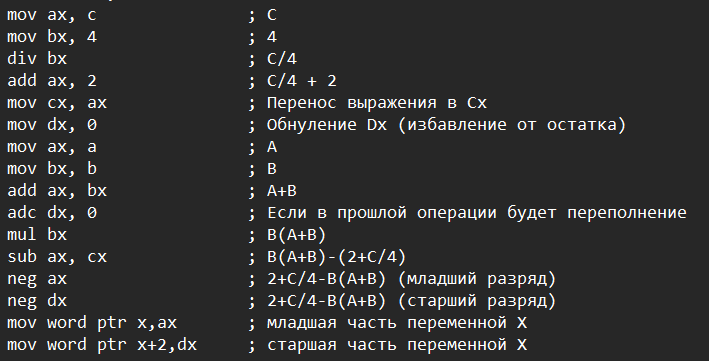
Внёс в сегмент данных исходные данные a, b, c в формате слова и x в формате двойное слово:

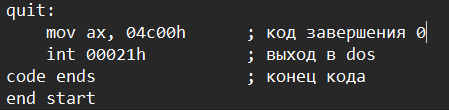


После в начале кода загрузил сегмент данных:



Далее приступил к реализации самой программы. Т.к. от перемены слагаемых сумма не меняется, решил сначала вычислить С/4 + 2, после вычислить B(A+B) (не забыл про переполнение, занес его в dx). Из-за величины B(A+B) решил от него отнять C/4 + 2, а затем, используя команду neg инвертировать полученный результат в обоих регистрах (dx как старший, ax как младший). Итоговые значения записал в переменную X с помощью двух подобных команд. Mov word ptr x, ax. В данной команде word – указание, какую память будет занимать вставляемое число. X+2 – старший регистр.



Далее завершил программу:  


В результате полный код выглядит следующим образом:

data ends ; конец начальных данных

code segment ; начало кода

assume cs: code, ds: data

start:

mov ax, data

mov ds, ax

mov ax, c ; C

mov bx, 4 ; 4

div bx ; C/4

add ax, 2 ; C/4 + 2

mov cx, ax ; Перенос выражения в Cx

mov dx, 0 ; Обнуление Dx (избавление от остатка)

mov ax, a ; A

mov bx, b ; B

add ax, bx ; A+B

adc dx, 0 ; Если в прошлой операции будет переполнение

mul bx ; B(A+B)

sub ax, cx ; B(A+B)-(2+C/4)

neg ax ; 2+C/4-B(A+B) (младший разряд)

neg dx ; 2+C/4-B(A+B) (старший разряд)

mov word ptr x,ax ; младшая часть переменной Х

mov word ptr x+2,dx ; старшая часть переменной Х

quit:

mov ax, 04c00h ; код завершения 0

int 00021h ; выход в dos

code ends ; конец кода

end start

После выполнил ручную проверку на калькуляторе. Результаты совпали. Сгенерировал ещё три тройки исходных данных для тестирования: 1(A: 7FFFh, B: 7FFFh, C: 7FFFh), 2(A: -32768d, B: -32768d, C: -32768d), 3(A: 20202d, B: 2020d, C: 202d). Все результаты оказались правильными.