

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**  
**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**  
**Кафедра вычислительных технологий**

**ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ №2**  
**по дисциплине**  
**«Системы реального времени»**

Работу выполнил студент группы 45/2 \_\_\_\_\_ Т. Э. Айрапетов

Отчет \_\_\_\_\_ принял  
доц. каф. ИТ \_\_\_\_\_ А. Н. Полетайкин

Краснодар  
2024

## Вариант 1

### Задание.

1. Составить программу для расчета заданного арифметического выражения (табл. 2.1). Длину и значение переменных А, В, С выбрать самостоятельно. Константы, заданные в выражении, использовать в кодовом сегменте. Программа должна корректно работать при любых допустимых значениях переменных.
2. Описать команды умножения и деления, используемые в программе на предмет длины операндов, участвующих в операции. Охарактеризовать длину результата и место его хранения.
3. Получить загрузочный модуль и протестировать выполнение программы в отладчике.
4. На основе составленной программы выполнить следующие действия:
  - а. загрузить в аккумулятор маскирующее слово, позволяющее определить заданную характеристику содержимого регистра DX;
  - б. выполнить заданную проверку и ее результат сохранить в переменной RESULT, объявленной в сегменте данных.
5. Перекомпилировать загрузочный модуль и протестировать выполнение программы в отладчике.
6. Произвести расчет времени выполнения программы.
7. Сделать выводы.

### Выполнение.

Код программы для подсчета выражения  $A/(976+B)-80*(C-15)$ :

```

.code
LStart:
    mov ax, 976
    add ax, vB ; B+976

    cmp ax, 0
    je div_by_zero

    xor dx, dx
    mov bx, ax
    mov ax, vA
    div bx ; A/(B+976)
    push ax

    mov cx, vC
    sub cx, 15 ; C-15

    mov ax, 80
    mul cx ; 80*(C-15)

    mov bx, ax
    pop ax
    sub ax, bx ; ИТОГ

    outint ax

    jmp LExit
div_by_zero:
    outstr offset er_msg

```

Рисунок 1 - Код для подсчета значения выражения

- Операция DIV осуществляет деление числа, находящегося в AX на передаваемый операнд. Длины операндов - 16 бит. Результат деления сохраняется в AX, остаток в DX.
- Операция MUL осуществляет умножение числа, находящегося в AX на передаваемый операнд. Длины операндов - 16 бит. При переполнении 16 бит, старшие биты результата сохраняются в DX, а младшие в AX.

lab2.<ModuleEntryPoint>+4D									
Address	Hex dump								ASCII
00403000	00	08	00	34	FC	02	00	12	.4ъ.
00403008	04	E4	E5	EB	E5	ED	E8	E5	деление
00403010	20	ED	E0	20	30	0D	0A	00	на 0...
00403018	0D	0A	00	00	50	72	65	73	....Pres
00403020	73	20	61	6E	79	20	6B	65	s any ke
00403028	79	20	74	6F	20	65	78	69	y to exi
00403030	74	2E	00	00	50	72	65	73	t...Pres
00403038	73	20	61	6E	79	20	6B	65	s any ke
00403040	79	20	74	6F	20	65	78	69	y to exi
00403048	74	2E	00	00	0D	0A	00	00	t.....
00403050	00	00	00	00	00	00	00	00	.....
00403058	00	00	00	00	00	00	00	00	.....
00403060	00	00	00	00	00	00	00	00	.....
00403068	00	00	00	00	00	00	00	00	.....
00403070	00	00	00	00	00	00	00	00	.....
00403078	00	00	00	00	00	00	00	00	.....
00403080	00	00	00	00	00	00	00	00	.....

Рисунок 2 - Выполнение программы в OllyDBG

```

mov ax, 1
test dx, ax
je is_not_zero
mov result, 0
jmp is_zero

is_not_zero:
mov result, 1
is_zero:

newline
outint dx
newline
outint result

```

Рисунок 3 - Код проверки 0 разряда DX на равенство 0

lab2.<ModuleEntryPoint>+0F5

Address	Hex dump	ASCII
00403000	00 08 00 34 FC 02 00 12	. .4ь .
00403008	04 00 00 E4 E5 EB E5 ED	. .делен
00403010	E8 E5 20 ED E0 20 30 0D	ие на 0.
00403018	0A 00 00 00 0D 0A 00 00	.....
00403020	0D 0A 00 00 0D 0A 00 00	.....
00403028	0D 0A 00 00 0D 0A 00 00	.....
00403030	50 72 65 73 73 20 61 6E	Press an
00403038	79 20 6B 65 79 20 74 6F	y key to
00403040	20 65 78 69 74 2E 00 00	exit

Рисунок 4 - Значение RESULT после выполнения кода

Подсчет времени исполнения:

- mov регистр, операнд  $3 \cdot 4 = 12$
- mov регистр, регистр  $2 \cdot 2 = 4$
- mov регистр, память  $2 \cdot (12 + 6) = 36$
- mov память, регистр  $13 + 6 = 19$
- mov память, операнд  $14 + 6 = 20$
- add регистр, память  $13 + 6 = 18$
- cmp регистр, операнд = 4
- je = 16 или 4
- jmp =  $2 \cdot 15 = 30$
- xor dx, dx  $2 \cdot 3 = 6$
- sub регистр, операнд = 4
- mul регистр = 118
- div регистр = 144
- push регистр = 15
- pop регистр = 15

Итог  $461/3.1 = 148.71$  нс.

### Вывод.

Были изучены команды арифметических и логических операций, приобретены практические навыки при их использовании в различных вычислительных задачах.