**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ**

**УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»**

**РУТ (МИИТ)**

Институт управления и цифровых технологий

Кафедра «Цифровые технологии управления транспортными процессами»

**ОТЧЁТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3**

|  |  |  |
| --- | --- | --- |
| **по дисциплине:** | | Ассемблер |
| **на тему:** | БИТОВЫЕ ОПЕРАЦИИ | |
| **цель работы:** | Изучение логических команд и команд сдвига; написание и откладка программы анализа битовой структуры данных. | |

|  |  |
| --- | --- |
| **Выполнил студент группы УИС-312** | **(Рыжов В. Р.)** |
| **Принял \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **(Преподаватель кафедры ЦТУТП Победоносцев И. Н.)** |

**Москва 2022 г.**

Оглавление

[**Требования к отчету** 3](#_Toc117530324)

[**Цель работы:** 4](#_Toc117530325)

[**1 Блок-схема программы** 6](#_Toc117530326)

[**2 ТЕКСТ ПРОГРАММЫ С КОМЕНТАРИЯМИ** 7](#_Toc117530327)

[**3 РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММЫ** 9](#_Toc117530328)

[**4 РЕЗУЛЬТАТЫ ПОШАГОВОГО ВЫПОЛНЕНИЯ ПРОГРАММЫ В ОТЛАДЧИКЕ** 12](#_Toc117530329)

[**5 ВОПРОСЫ** 14](#_Toc117530330)

[**ЗАКЛЮЧЕНИЕ** 16](#_Toc117530331)

# **Требования к отчету**

Отчет должен включать:

1. Продемонстрируйте работающую программу преподавателю с предъявлением результатов в командной строке.

2. Подготовьте отчет, включающий:

- текст варианта задания;

- ручной вариант расчета;

- укрупненную блок-схему алгоритма разработанной программы;

- текст разработанной программы с обязательными комментариями;

- результат работы программы в виде скриншота;

- результаты пошагового выполнения программы в отладчике в виде заполненной таблицы.

**Задачи:**

Написать и отладить программу, выполняющую следующие действия (значения R, i, j и k приведены на рисунке 0.1):

(A AND B) OR B => R, где R заданный регистр;   
2) Если i-й бит регистра R равен 0, то записать 1 в k-й бит регистра R, иначе 0;   
3) Инвертировать j-й бит регистра R;   
4) Поменять местами тетрады байта, записанного в регистре R;   
5) Если полученное значение меньше, чем 32, прибавить к R 32;   
6) Вывести на экран символ, код которого представлен в регистре R с помощью вызова функции MessageBox.


Рис 0.1 – значения R, i, j и k

# **1 Блок-схема программы**

На рисунке 1.1 представлена блок схема для программы lab3.

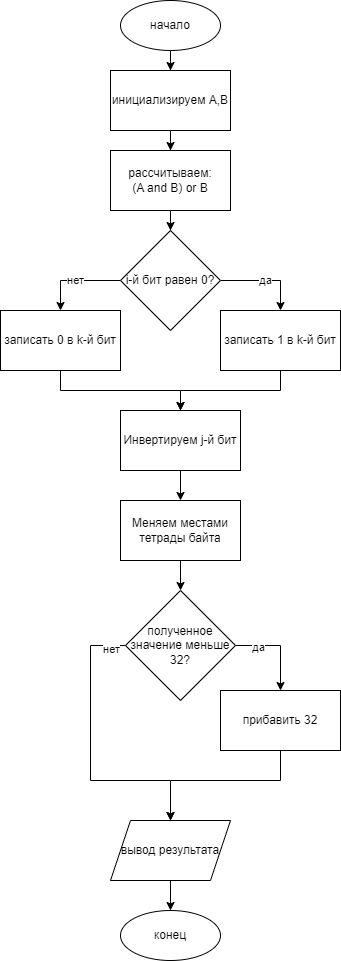


Рис 1.1 – Блок схема для приложения lab3.

**2 Ручной расчёт**

Ручной расчёт приведён на рисунке 2.1

Изображение выглядит как текст

Автоматически созданное описание  
Рис 2.1 – ручной расчёт

47 в таблице ASCII обозначается, как символ «/»

# **3 ТЕКСТ ПРОГРАММЫ С КОМЕНТАРИЯМИ**

;Лабораторная работа №3

;Выполнил ст. гр. УИС-312 Рыжов В.Р. 24.10.2022

;Вариант 20

;A= "R"

;B= "v"

;R=bh, i=4, j=7, k=2

.486

.model flat, stdcall

option casemap: none

.stack 100h

;=========================================

include c:\masm32\include\windows.inc

include c:\masm32\include\user32.inc

include c:\masm32\include\kernel32.inc

includelib c:\masm32\lib\user32.lib

includelib c:\masm32\lib\kernel32.lib

;=========================================

.data

tit db 'Рыжов В.Р. УИС-312',0

A db "R" ; первая буква фамилии

B db "v" ; первая буква имени

rezz db ?, 0

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

.code

main:

;=============Операция 1==================

mov bh,A ; Помещаем значение A в регистр bl

and bh,B ; Производим логическое умножение (А and B)

or bh,B ; Производим логическое сложение (A and B) or B

;=============Операция 2==================

test bh,00010000b ; Если i-й бит регистра R равен 0, то записать 1 в k-й бит регистра R

jz zero ; переход к метке zero

and bh,11111011b ; записываем 0 в 2-ой бит регистра bh

jmp invert ; переход к метке

zero: ; метка zero

or bh,00000100b ; записываем 1 в 2-ой бит регистра bl

;=============Операция 3==================

invert: ; метка invert

xor bh,10000000b ; инвертируем 7-ой бит регистра bh

;=============Операция 4==================

ror bh,4 ; сдвиг тетрадов

;=============Операция 5==================

cmp bh, 32 ; сравнение регистра bh с 32

jnb fin ; переход к метке, если не меньше 32

or bh, 00100000b ; прибавляем 32 к bh

;=============Вывод результатов==================

fin: mov rezz, bh ;Запись кода символа в rez

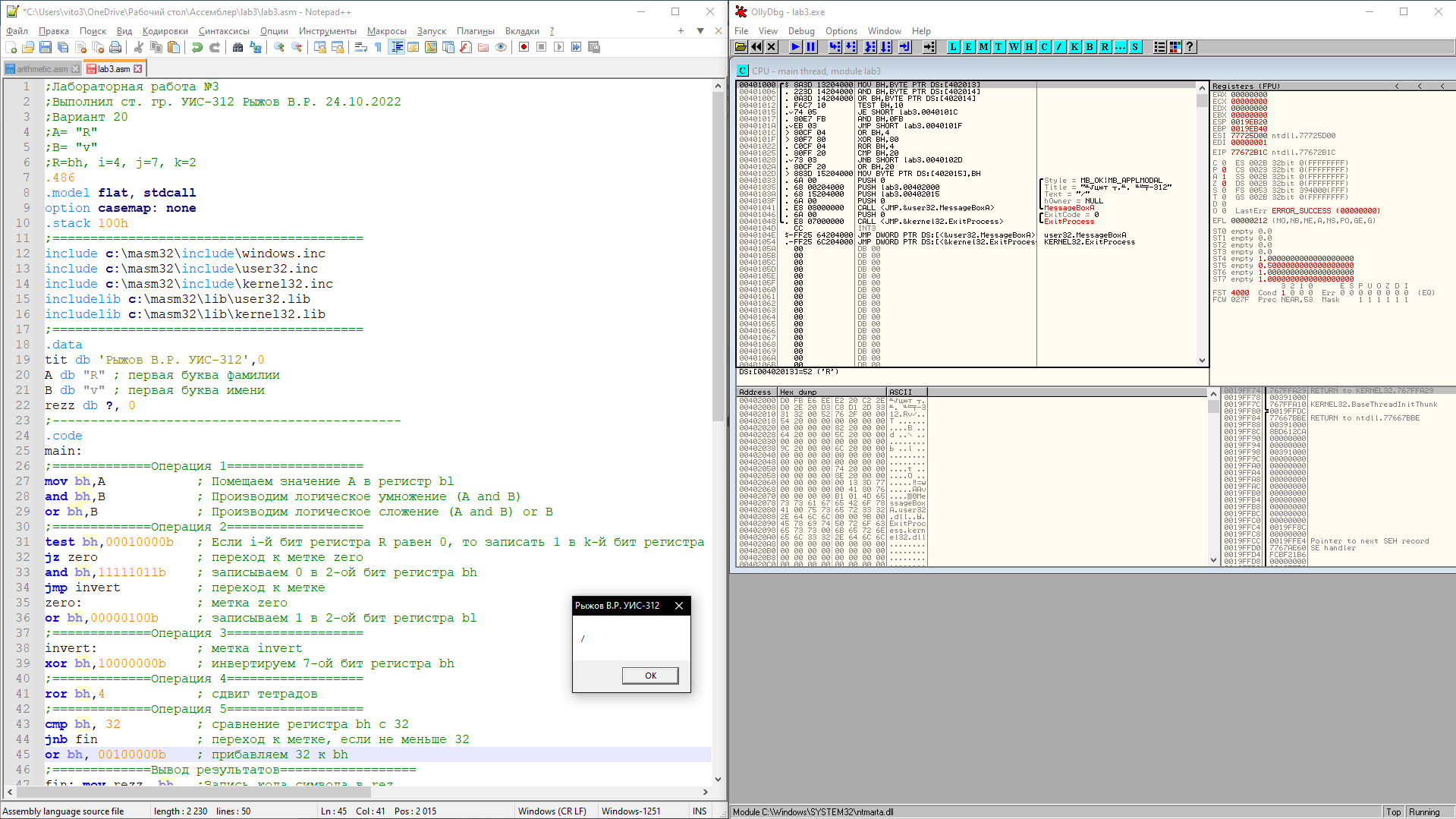
INVOKE MessageBox, 0, OFFSET rezz, OFFSET tit , 0

invoke ExitProcess, 0

end main

# **4 РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММЫ**

На рисунке 4.1 представлен результат работы программы lab3

  
Рис 4.1 – результат работы программы lab3

# **5 РЕЗУЛЬТАТЫ ПОШАГОВОГО ВЫПОЛНЕНИЯ ПРОГРАММЫ В ОТЛАДЧИКЕ**

В таблице 1.1 приведено пошаговое выполнение программы lab3  
~ - значение не меняется

Таблица 1.1 – Пошаговая откладка

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  пп | Команда | EBX | ESP | EIP | Флаги | | |
| PF | ZF | OF |
| 1 | 8A3D | 003C5000 | 0019FF74 | 00401000 | 1 | 1 | 0 |
| 2 | 223D | 003C5200 | ~ | 00401006 | ~ | ~ | ~ |
| 3 | 0A3D | ~ | ~ | 0040100C | 0 | 0 | ~ |
| 4 | F6C7 | 003C7600 | ~ | 00401012 | ~ | ~ | ~ |
| 5 | 74 05 | ~ | ~ | 00401015 | ~ | ~ | ~ |
| 6 | 80E7 FB | ~ | ~ | 00401017 | ~ | ~ | ~ |
| 7 | EB 03 | 003C7200 | ~ | 0040101A | 1 | ~ | ~ |
| 8 | 80F7 80 | ~ | ~ | 0040101F | ~ | ~ | ~ |
| 9 | C0CF 04 | 0022F200 | ~ | 00401022 | 0 | ~ | ~ |
| 10 | 80FF 20 | 00222F00 | ~ | 00401025 | ~ | ~ | ~ |
| 11 | 73 03 | ~ | ~ | 00401028 | 1 | ~ | ~ |
| 12 | 883D | ~ | ~ | 0040102D | ~ | ~ | ~ |
| 13 | 6A 00 | ~ | ~ | 00401033 | ~ | ~ | ~ |
| 14 | 68 00204000 | ~ | 0019FF70 | 00401035 | ~ | ~ | ~ |
| 15 | 68 15204000 | ~ | 0019FF6C | 0040103A | ~ | ~ | ~ |
| 16 | 6A 00 | ~ | 0019FF68 | 0040103F | ~ | ~ | ~ |
| 17 | E8 08000000 | ~ | 0019FF64 | 00401041 | ~ | ~ | ~ |
| 18 | 6A 00 | 00222F00 | 0019FF74 | 00401046 | 1 | 1 | 0 |

# **ЗАКЛЮЧЕНИЕ**

В ходе данной работы я познакомился c логических командами и команд сдвига кода ассемблера. А также лучше понял алгоритмами работы программ, написанных для ассемблера.

# **Приложение**

1. В чем отличие логических команд от арифметических?

* Логические и арифметические операции отличаются тем, что в логических операциях вычисления производятся поразрядно (между собой взаимодействуют только одноименные разряды и переносов между разрядами нет). При выполнении арифметических операций в случае необходимости происходят переносы между соседними разрядами (от младшего разряда к старшему)

1. В каких случаях следует пользоваться командой логического

умножения AND, а в каких - командой TEST?

* логическое умножение (AND, TEST): если оба из сравниваемых битов равны 1, то результат равен 1; во всех остальных случаях результат - 0. Команда TEST (в отличие от AND) не сохраняет результат выполненной операции;

1. Как можно с помощью логических команд производить очистку

(обнуление) регистра?

* Установка нулевого значения заданного бита числа, записанного в регистре или переменной, производится путем логическое умножения данного числа на двоичное число, которое содержит единицы во всех разрядах, кроме заданного. Заданный разряд содержит 0.

1. Объясните, как и в каких случаях с помощью команд логического

сдвига можно выполнять операции умножения и деления?

* Сдвиг операнда на один бит влево эквивалентен умножению его на 2, а сдвиг на один бит вправо соответствует его делению на 2, то команды сдвига можно использовать для быстрого умножения и деления чисел.

1. Каковы особенности выполнения операций логического, арифметического и циклического сдвига?

* При логическом сдвиге происходит смещение в разрядной сетке всех цифр числового кода, включая и разряд знака. При этом в освободившимся при сдвиге разряда сетки устанавливаются нули, а не уместившиеся в сетке при сдвиге теряются.
* Циклический сдвиг отличается от логического тем, что при смещении всей числовой последовательности цифры, выходящие за пределы разрядной сетки, снова вводятся в освобождающиеся разрядные позиции.
* При арифметическом сдвиге производится смещение всей числовой последовательности (вправо или влево) без изменения позиции знака числа. Для положительного и отрицательного чисел, представленных в прямом коде при сдвиге влево или вправо освободившиеся разряды сетки, заполняются нулями. Для отрицательных чисел, представленных в обратном коде, освобождающиеся разряды сетки при сдвиге числа вправо или влево заполняются единицами.

1. Предложите способы решения следующих задач с помощью логических команд и команд сдвига:

* проверка значений двух заданных битов; (TEST)
* установка (сброс) заданных битов (двух и более); (AND)
* поменять местами значения двух заданных битов числа; (SHR)
* поменять местами значения байтов AH и AL в регистре AX c помощью циклического сдвига. (ROR)