МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

#### 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 №1

«СИСТЕМА КОМАНД МИКРОПРОЦЕССОРА X86»

Вариант №12

|  |  |  |
| --- | --- | --- |
| Выполнил: студент гр. ИНБс-3301-01-00 | \_\_\_\_\_\_\_\_\_\_\_\_ | П.С. Кривошеин |
| Проверил: доцент кафедры РЭС | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ | М. А. Земцов |

#### Киров 2025

**Цель работы:** изучение системы команд и способов адресации микропроцессоров с архитектурой x86.

**Ход работы:**

1. Исходные данные представлены в таблице 1.

Таблица 1 – Исходные данные



1. Расчет выражения

Вычислить M = (Z’ xor(X + Y))and(X - Y) , где Z’ – получено в результате инверсии всех бит Z.

Расчёт выражений:

Z = 76 = 4C



Z’ (Инверсия битов) = 179 = B3



X = 11 = B



X + Y = 11 + 5 = 10 (16-ричная система)



Z’ xor (X + Y) = 179 ⊕ 16 = 163 = А3



X – Y = 11 – 5 = 6



(Z’ xor (X + Y)) and (X - Y) = 163 ∧ 6 = 2



***Вывод:*** в ходе лабораторной работы была изучена система команд и способов адресации микропроцессоров с архитектурой х86, получены навыки программирования на языке assembler.

**3. Код программы:**

.686 ; Использование программой процессора уровня 686

.model flat,stdcall ; Модель памяти flat и соглашение вызова stdcall (аргументы ;передаются через стек, очистка стека)

.stack 100h ; Резервирует 256 байт для стека программы

.data ; Начало, где объявляются переменные

x db 11

y db 5

z db 76

**.code** ; Начало кода, где исполняется код (инструкции)

ExitProcess PROTO STDCALL :DWORD ; Прототип функции из Windows API, которая завершает процесс.

Start: // Метка точки входа в программу

; Зануляем регистры, чтобы не запутаться в расчетах (дополнительно)

xor eax,eax

xor ebx,ebx

xor ecx,ecx

; Инверсия переменной z

mov al, [z]

not al

; Сложение X + Y

add bl, [x]

add bl, [y]

; Исключающее ИЛИ

xor al, bl

; Вычитание X - Y

sub bl, [y]

sub bl, [y]

; Конъюнкция двух переменных

and al, bl

exit: ; Конец программы

Invoke ExitProcess,1 ; Завершение программы, код возврата процесса

End Start ; Конец исходного кода