

Федеральное государственное автономное образовательное

учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Направление подготовки: 09.03.04 – Системное и прикладное программное обеспечение

Дисциплина «Основы профессиональной деятельности»

**Отчёт по лабораторной работе №6**

**Асинхронный обмен данными с ВУ**

Вариант №15666

Выполнил

Чимирев Игорь Олегович

P3115

Проверил

Блохина Елена Николаевна

Санкт – Петербург, 2025

Содержание

[Задания и цели работы 3](#__RefHeading___Toc3867_2985257696)

[Задание 3](#__RefHeading___Toc10225_2376174330)

[Цель работы 3](#__RefHeading___Toc10227_2376174330)

[Ход работы 4](#__RefHeading___Toc10229_2376174330)

[Программа 4](#__RefHeading___Toc10231_2376174330)

[Ассемблер 4](#__RefHeading___Toc5943_691564783)

[Назначение программы и комплекса подпрограмм 6](#__RefHeading___Toc10235_2376174330)

[Область представления 6](#__RefHeading___Toc10239_2376174330)

[Область допустимых значений 6](#__RefHeading___Toc10241_2376174330)

[Расположение в памяти ЭВМ программы, подпрограмм и программных комплексов, исходных данных и результатов 7](#__RefHeading___Toc10243_2376174330)

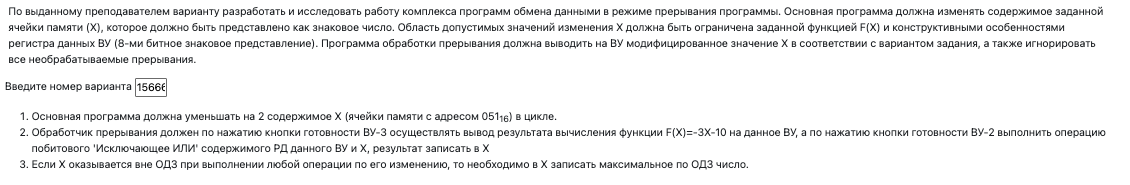
[Подпрограммы 7](#__RefHeading___Toc6899_691564783 ÐÐ¾Ð¿Ð¸Ñ 1)

[Адреса первой и последней исполняемой команды 7](#__RefHeading___Toc7976_742792518)

[Методика проверки 7](#__RefHeading___Toc7978_742792518)

[Заключение 8](#__RefHeading___Toc10247_2376174330)

# Задание



## Цель работы

Изучение организации системы процесса прерывания программы и исследования порядка функционирования ЭВМ при обмене данными в режиме прерывания программы.

# Ход работы

## Программа

### Ассемблер

|  |
| --- |
| ORG 0x0  V0: WORD $DEFAULT, 0x180; вектор прерывания #7  V1: WORD $DEFAULT, 0x180; вектор прерывания #7  V2: WORD $INT2, 0x180; вектор прерывания #7  V3: WORD $INT3, 0x180; вектор прерывания #7  V4: WORD $DEFAULT, 0x180; вектор прерывания #7  V5: WORD $DEFAULT, 0x180; вектор прерывания #7  V6: WORD $DEFAULT, 0x180; вектор прерывания #7  V7: WORD $DEFAULT, 0x180 ; вектор прерывания #7  ORG 0x080  START:  CLA  OUT 0x1 ; Сброс устройств (порты 1, 3, 11, 14, 18, 22, 26, 30)  OUT 0x3  OUT 0xB  OUT 0xE  OUT 0x12  OUT 0x16  OUT 0x1A  OUT 0x1E  LD #0xA  OUT 5  LD #0xB  OUT 7  EI  MAIN\_LOOP:  DI  LD X  SUB #2  CALL CLAMP\_VALUE  ST X  EI  JUMP MAIN\_LOOP  CLAMP\_VALUE:  **CMP** $MIN\_VALUE  BLT LD\_MIN  **CMP** $MAX\_VALUE  BEQ VALUE\_OK  BGE LD\_MIN  JUMP VALUE\_OK  LD\_MIN:  LD $MAX\_VALUE  VALUE\_OK:  RET  XOR\_X:  PUSH ;~x  PUSH ;~y  LD X  NOT  ST &1  LD &3  NOT  ST &0  LD X  AND &0  ST &0  LD &1  AND &3  ST &1  LD &0  OR &1  ST &3  POP  POP  RET  INT2:  PUSH    HLT  IN 4 ; Чтение значения из порта 4  SXTB ; Знаковое расширение байта  PUSH  CALL XOR\_X ; Вызов XOR\_X (результат в аккумуляторе)  POP  HLT  CALL CLAMP\_VALUE ; Ограничение значения  ST X ; Сохранение в X  HLT  POP  IRET  INT3:  PUSH  LD X ; Загрузка X  HLT  ADD X  ADD X  NEG  SUB #10 ; -3X - 10  ;CALL CLAMP\_VALUE  OUT 6 ; Вывод результата в порт 6  NOP  POP  IRET ; Выход из прерывания  DEFAULT:  IRET ; обработка прерывания по умолчанию  MIN\_VALUE: WORD -45  MAX\_VALUE: WORD 39  ORG 0x051  X: WORD 90 |

### Назначение программы и комплекса подпрограмм

Программа уменьшает значение ячейки 0x051 (X) на -2 в цикле. Выполнение может быть прервано ВУ-2 или ВУ-3 по нажатию «Готов».

Подпрограмма START — изначальная инициализация (запрет прерываний, загрузка векторов в MR)

Подпрограмма MAIN\_LOOP — основной цикл, где происходит уменьшение значения

CLAMP\_VALUE — сохранение значения в ячейку памяти в соответствии с ОДЗ.

XOR\_X -- функция, которая возвращает результат побитового исключающего ИЛИ с X и значением DR у ВУ-2

INT2 — обработчик прерывания ВУ-2. Вызывает функцию XOR\_X

INT3 — обработчик прерывания ВУ-3. Выводит результат вычисления функции F(x) = -3X — 10 (вызывает CLAMP\_VALUE)

#### Область представления

Переменная X — знаковое число [-2^15; 2^15 — 1]

Константа MIN\_VALUE — минимальное значение для X по ОДЗ — представлено знаковым числом [-2^15; 2^15 — 1]

Константа MAX\_VALUE — максимальное значение для X по ОДЗ — представлено знаковым числом [-2^15; 2^15 - 1]

Расширение знака, проверка ОДЗ неверная

#### Область допустимых значений

Функция F(X) принимает значения [-2^7, 2^7-1], так как выводит их на 8-ми разрядный регистр ВУ. Соответственно:

-2^7 ≤ -3X — 10 ≤ 2^7 - 1

-2^7 + 10 ≤ -3X ≤ 2^7 + 9

-118 ≤ -3X ≤ 137; так как X — целое число, то -45 ≤ X ≤ 39

### Расположение в памяти ЭВМ программы, подпрограмм и программных комплексов, исходных данных и результатов

Векторы прерываний: [0x000; 0x00F]

Программа и её комплекс подпрограмм: [0x080; 0x0C4]

Данные программы: 0x051 и [0x0C5; 0x0C6]

#### Подпрограммы

Подпрограмма START: [0x080; 0x08D]

Подпрограмма MAIN\_LOOP: [0x08E; 0x094]

Подпрограмма CLAMP\_VALUE: [0x095; 0x098]

Подпрограмма XOR\_X: [0x099; 0x0AC]

Подпрограмма INT2: [0xAD; 0x0B7]

Подпрограмма INT3: [0x0B8; 0x0C3]

#### Адреса первой и последней исполняемой команды

Первая исполняемая команда находится по адресу 0x080, последняя -- 0x094

### Методика проверки

1. Загрузить код в БЭВМ
2. Заменить инструкции NOP в обработчиках прерываний INT2 и INT3 на HLT
3. Запустить программу в режиме работы, нажать кнопку готовности на **ВУ-2/ВУ-3**.
4. Дождаться остановки программы, зафиксировать значение **X**.

5. **Проверка вычислений в обработчике**

\*Возобновить выполнение программы.

\*Дождаться остановки после выполнения основного вычисления в обработчике:

1) Для **ВУ-2**: операция XOR с регистром данных устройства.

2) Для **ВУ-3**: вычисление функции F(X) = -3X - 10.

6. Сравнить результат,выданный программой с ожидаемым.

7. **Проверка избирательности прерываний**

Нажать кнопки **«Готовность»** на остальных ВУ (1, 4, 5, 6, 7,8,9).

Убедиться, что программа **не останавливается** на точках останова (обработчики для этих устройств не активируются).

8. Дождаться пока X дойдет до границы ОДЗ, проверить чтобы его значение было установлено в минимальное по ОДЗ.

# Вывод

В ходе лабораторной работы были изучены принципы работы с внешними устройствами, на примере ВУ-3 и ВУ-2 была написана программа для асинхронного взаимодействия, также изучил цикл прерывания и циклы исполнения новых команд.