**Министерство образования и науки**

**Российской Федерации**

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Факультет программной инженерии и компьютерной техники

Дисциплина: Основы профессиональной деятельности

**Лабораторная работа №6**

Вариант 63302

Выполнил студент группы Р3133 Анисимов Максим Дмитриевич

Проверил Блохина Елена Николаевна

Санкт-Петербург

2023 г

Оглавление

[Текст задания 3](#_Toc136385826)

[Назначение программы 3](#_Toc136385827)

[Текст исходной программы на языке Ассемблера 3](#_Toc136385828)

[Область допустимых значений 6](#_Toc136385829)

[Расположение данных в памяти 6](#_Toc136385830)

[Область представления 6](#_Toc136385831)

[Методика проверки 6](#_Toc136385832)

# Текст задания

По выданному преподавателем варианту разработать и исследовать работу комплекса программ обмена данными в режиме прерывания программы. Основная программа должна изменять содержимое заданной ячейки памяти (Х), которое должно быть представлено как знаковое число. Область допустимых значений изменения Х должна быть ограничена заданной функцией F(X) и конструктивными особенностями регистра данных ВУ (8-ми битное знаковое представление). Программа обработки прерывания должна выводить на ВУ модифицированное значение Х в соответствии с вариантом задания, а также игнорировать все необрабатываемые прерывания.

Введите номер варианта 

1. Основная программа должна увеличивать на 2 содержимое X (ячейки памяти с адресом 03316) в цикле.
2. Обработчик прерывания должен по нажатию кнопки готовности ВУ-3 осуществлять вывод результата вычисления функции F(X)=-7X на данное ВУ, a по нажатию кнопки готовности ВУ-2 записать содержимое РД данного ВУ в Х
3. Если Х оказывается вне ОДЗ при выполнении любой операции по его изменению, то необходимо в Х записать минимальное по ОДЗ число.

# Назначение программы

Основная программа должна увеличивать на 2 содержимое X (ячейки памяти с адресом 03316) в цикле.

Обработчик прерывания должен по нажатию кнопки готовности ВУ-3 осуществлять вывод результата вычисления функции F(X)=-7X на данное ВУ, a по нажатию кнопки готовности ВУ-2 записать содержимое РД данного ВУ в Х

Если Х оказывается вне ОДЗ при выполнении любой операции по его изменению, то необходимо в Х записать минимальное по ОДЗ число.

# Текст исходной программы на языке Ассемблера

# ORG 0x0

# V0: WORD $DEFAULT, 0x180

# V1: WORD $DEFAULT, 0x180

# V2: WORD $INT2, 0x180

# V3: WORD $INT3, 0x180

# V4: WORD $DEFAULT, 0x180

# V5: WORD $DEFAULT, 0x180

# V6: WORD $DEFAULT, 0x180

# V7: WORD $DEFAULT, 0x180

# DEFAULT: IRET

# ORG 0x033

# X: WORD ?

# MIN: WORD 0xFFEE

# MAX: WORD 0x0012

# START: DI

# CLA

# OUT 0x1

# OUT 0x3

# OUT 0xB

# OUT 0xE

# OUT 0x12

# OUT 0x16

# OUT 0x1A

# OUT 0x1E

# LD #0xB

# OUT 7

# LD #0xA

# OUT 5

# PROG: DI

# LD X

# ADD #0x2

# CMP MIN

# BMI MINX

# CMP MAX

# BPL MINX

# ST X

EI

# JUMP PROG

# INT3: LD X

# HLT

# NEG

# ASL

# ASL

# ASL

# ADD X

# OUT 6

# HLT

# IRET

# INT2: CLA

# IN 4

# HLT

# CMP MIN

# BMI MINX

# CMP MAX

# BPL MINX

# ST X

# HLT

# IRET

# MINX: LD MIN

# ST X

# HLT

# JUMP PROG

# Область допустимых значений

Так как в регистре данных находится 8-битное знаковое число, функция F(X) находится в промежутке , тогда

-128

-18

X

# Расположение данных в памяти

Вектор прерываний: 0x000 – 0x00F

Переменные: 0x033 – 0x035

Программа: 0x036 – 0x061

# Область представления

X, min, max – знаковое 16-ричное целое число

DR КВУ – 8-ми разрядное целое знаковое число

# Методика проверки

**Проверка основной программы:**

1. Заменить NOP на HLT
2. Записать в переменную X максимальное число из ОДЗ
3. Запустить БЭВМ в режиме ‘работа’
4. Дождаться остановки программы
5. Убедиться, что при выполнении программы в X записывается минимальное по ОДЗ число

**Проверка прерываний:**

1. Загрузить программу в ассемблер БЭВМ
2. Заменить NOP на HLT
3. Нажать кнопку “компилировать”
4. Запустить БЭВМ в режиме ‘работа’
5. Нажать кнопку ‘готов’ на ВУ – 3
6. Дождаться остановки программы
7. Записать значение X в ячейке памяти 0x033
8. Рассчитать ожидаемое значение
9. Сравнить значение регистра данных ВУ – 3 с ожидаемым значением
10. Загрузить программу в ассемблер БЭВМ
11. Заменить NOP на HLT
12. Запустить БЭВМ в режиме ‘работа’
13. Ввести на ВУ – 2 произвольное число
14. Нажать кнопку «Готов»
15. Дождаться остановки программы

7. Сравнить значение X в ячейке памяти 0x033 с введённым значением

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Прерывание ВУ-3 | | | Прерывание ВУ-2 | | |
| AC (0…7) | Ожидаемое значение  -7X | DR | DR  КВУ - 2 | Ожидаемое значение  X | Результат |
| 2 | -14 | -14 | 7F | 7F | 7F |
| 3 | -21 | -21 | CC | CC | CC |

0010 0011

ORG 0x0

V0: WORD $DEFAULT, 0x180

V1: WORD $DEFAULT, 0x180

V2: WORD $INT2, 0x180

V3: WORD $INT3, 0x180

V4: WORD $DEFAULT, 0x180

V5: WORD $DEFAULT, 0x180

V6: WORD $DEFAULT, 0x180

V7: WORD $DEFAULT, 0x180

DEFAULT: IRET

ORG 0x033

X: WORD ?

MIN: WORD 0xFFEE

MAX: WORD 0x0012

ORG 0x033

START: DI

CLA

OUT 0x1

OUT 0x3

OUT 0xB

OUT 0xE

OUT 0x12

OUT 0x16

OUT 0x1A

OUT 0x1E

LD #0xB

OUT 7

LD #0xA

OUT 5

JUMP $PROG

PROG: EI

LD X

CYCLE:

ADD #0x2

CMP MIN

BMI MINX

CMP MAX

BPL MINX

ST X

JUMP $CYCLE

INT3: LD X

HLT

NEG

ASL

ASL

ASL

ADD X

OUT 6

HLT

IRET

INT2: CLA

IN 4

HLT

ST X

HLT

IRET

MINX: LD MIN

ST X

HLT

IRET