Федеральное государственное автономное образовательное учреждение

высшего образования «Национальный исследовательский университет ИТМО»

*Факультет программной инженерии и компьютерной техники*

**Лабораторная работа №6**

**По дисциплине**

**“Основы профессиональной деятельности”**

Вариант: 14016

Выполнила:

Армут Алина Юсуфовна

Группа: Р3114

Преподаватель:

Блохина Елена Николаевна

Санкт-Петербург, 2023г

## Задание

Изображение выглядит как текст, документ, Шрифт, снимок экрана

Автоматически созданное описание

## Ход работы

### Описание программы:

Программа циклично декрементирует значение ячейки X, обрабатывает прерывания с ВУ-1 и ВУ-2, вызывает подпрограмму.

Подпрограмма проверяет полученный аргумент X на -21 ≤ X ≤ 21, и если аргумент удовлетворяет данному условию, то возвращает его, иначе возвращает 21.

Прерывание ВУ-1 осуществляет вывод на это ВУ значения функции F(x) = -6x+1 для текущего значения X.

Прерывание ВУ-2 выполняет операцию, побитового маскирования, оставляя 5-х младших разряда содержимого РД данного ВУ и Х

## **Описание и назначение исходных данных, область представления и область допустимых значений исходных данных и результата:**

0x047 – ячейка X

0x048 - константа, хранит MIN значение по ОДЗ

0x049 – константа, хранит MAX значение по ОДЗ

##### Область представления:

X, MIN, MAX – знаковое 16-ричное целое число

##### Область допустимых значений:

**-21 = 1111.1111.1110.1011=** 0xFFEB

**21 = 0000.0000.0001.0101=** 0x0015

##### Расположение в памяти БЭВМ программы, исходных данных и результатов:

Вектор прерываний: 0x00 – 0x03

Переменные: 0x047-0x050

Программа: 0x5С-0x60

Обработка прерываний для ВУ-1:0x61-0x65

Обработка прерываний для ВУ-2:0x66-0x6E

Подпрограмма для проверки значений: 0x77-07F

##### Адрес первой и последней выполняемой команды программы

Адрес первой выполняемой программы:

Адрес последней выполняемой программы: нет, так как программа циклична

### Текст исходной программы на языке ассемблере БЭВМ

ORG 0x000

V1: WORD $INT1, 0x180

V2: WORD $INT2, 0x180

ORG 0x047

X: WORD ?

INTER: WORD ?

R: WORD ?

MIN: WORD 0xFFEB

MAX: WORD 0x0015

START: DI

CLA

OUT 0x1

OUT 0x7

OUT 0xB

OUT 0xE

OUT 0x11

OUT 0x15

OUT 0x19

OUT 0x1D

LD #0x8

OUT 0x3

LD #0x9

OUT 0x5

LD MAX

ST X

EI

MAIN:

LD X

DEC

ST R

CALL CHECK

JUMP MAIN

INT1:

LD X

HLT

CALL F

LD INTER

OUT 0x2

IRET

INT2:

LD X

HLT

IN 0x4

AND X

AND #0x001F

HLT

ST R

CALL CHECK

HLT

IRET

F: LD X

ASL

ADD X

ASL

NEG

INC

ST INTER

RET

CHECK:

LD R

CMP MIN

BMI LD\_MAX

CMP MAX

BPL LD\_MAX

ST X

RET

LD\_MAX:

LD MAX

ST X

RET

## Методика проверки:

Проверка обработки прерываний:

1. Загрузить текст программы в БЭВМ.

2. Заменить NOP на HLT.

3. Запустить программу в режиме РАБОТА.

4. Установить «Готовность ВУ-1».

5. Дождаться останова.

6. Записать текущее значение X из памяти БЭВМ:

1. Запомнить текущее состояние счетчика команд.

2. Ввести в клавишный регистр значение 0x047

3. Нажать «Ввод адреса».

4. Нажать «Чтение».

5. Записать значение регистра данных.

6. Вернуть счетчик команд в исходное состояние.

7. Записать результат обработки прерывания – содержимое DR контроллера ВУ-1

8. Рассчитать ожидаемое значение обработки прерывания

9. Нажать «Продолжение».

10. Ввести в ВУ-2 произвольное число, записать его

11. Установить «Готовность ВУ-2».

12. Дождаться останова.

13. Записать текущее значение X из памяти БЭВМ (аналогично п.6).

14. Нажать «Продолжение».

15. Записать текущее значение X из памяти БЭВМ (аналогично п.6).

16. Рассчитать ожидаемое значение переменной X после обработки прерывания

(если значение X выходит за пределы ОДЗ, тогда в X будет записано максимальное по

ОДЗ значение)

Проверка основной программы:

1. Загрузить текст программы в БЭВМ.

2. Записать в переменную X максимальное по ОДЗ значение (21)

3. Запустить программу в режиме останова.

4. Пройти нужное количество шагов программы, убедиться, что при уменьшении

X на 1, до того момента, когда он равен 21, происходит сброс значения в максимальное ОДЗ.

## Вывод

В ходе выполнения работы я ознакомилась с устройством обмена по прерываниям, изучила процесс прерывания. Также закрепила знания в написании программ на ассемблере БЭВМ.

## Список литературы

1. Тейнсли Д. Linux и UNIX программирование в shell. Руководство разработчика. BHV, «Ирина», Киев, 2001. – Режим доступа: <https://rtfm.co.ua/uploads/books/Linux_i_UNIX_proghrammirovaniie_v_shell_-_Devid_Tieinsli.pdf>
2. Операционная система Linux: Курс лекций. Учебное пособие / Г. В .Курячий, К. А. Маслинский — М. : ALT Linux; Издательство ДМК Пресс, 2010. — 348 с. : ил. ; 2-е изд., исправленное.— (Библиотека ALT Linux). Режим доступа:

<https://docs.altlinux.org/books/altlibrary-linuxintro2.pdf>

ORG 0x000

V1: WORD $INT1, 0x180

V2: WORD $INT2, 0x180

ORG 0x047

X: WORD ?

INTER: WORD ?

R: WORD ?

MIN: WORD 0xFFEB

MAX: WORD 0x0015

START: DI

CLA

OUT 0x1

OUT 0x7

OUT 0xB

OUT 0xE

OUT 0x11

OUT 0x15

OUT 0x19

OUT 0x1D

LD #0x8

OUT 0x3

LD #0x9

OUT 0x5

LD MAX

ST X

EI

MAIN:

LD X

DEC

ST R

CALL CHECK

JUMP MAIN

INT1:

HLT

CALL F

LD INTER

OUT 0x2

IRET

INT2:

HLT

IN 0x4

ST INTER

LD X

AND #0x001F

ST R

CALL CHECK

HLT

IRET

F: LD X

ASL

ADD X

ASL

NEG

INC

ST INTER

RET

CHECK:

LD R

CMP MIN

BMI LD\_MAX

CMP MAX

BPL LD\_MAX

ST X

RET

LD\_MAX:

LD MAX

ST X

RET