**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение

высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Отчет**

по лабораторной работе № 6

по дисциплине «**Основы профессиональной деятельности**»

Вариант № 98465

Автор: Дениченко Александр Олегович

Факультет: ПИиКТ

Группа: P3112

Преподаватель: Осипов Святослав Владимирович



Санкт-Петербург, 2023

**Цель** **работы:**

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.

**Задание:**

1. Основная программа должна инкрементировать содержимое X (ячейки памяти с адресом 01F16) в цикле.
2. Обработчик прерывания должен по нажатию кнопки готовности ВУ-3 осуществлять вывод результата вычисления функции F(X)=-3X на данное ВУ, a по нажатию кнопки готовности ВУ-2 выполнить операцию побитового маскирования, оставив 4-х младших разряда содержимого РД данного ВУ и Х, результат записать в Х
3. Если Х оказывается вне ОДЗ при выполнении любой операции по его изменению, то необходимо в Х записать минимальное по ОДЗ число.

**Ход работы:**

**Назначение программы**

Программа циклически увеличивает значение ячейки памяти на 1 и обрабатывает прерывания.

**Расположение в памяти БЭВМ данных программы**

Вектор прерываний: 0x000 – 0x00F

Переменные: 0x018 – 0x01A

Программа: 0x01B – 0x04D

**Область представления**

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

**Область допустимых значений**

**Код**

ORG 0x000

V0: WORD $DEFAULT, 0x180 ;Вектор прерываний #0

V1: WORD $DEFAULT, 0x180 ;Вектор прерываний #1

V2: WORD $INT2, 0x180 ;Вектор прерываний #2 -- 0010

V3: WORD $INT3, 0x180 ;Вектор прерываний #3 -- 0011

V4: WORD $DEFAULT, 0x180 ;Вектор прерываний #4

V5: WORD $DEFAULT, 0x180 ;Вектор прерываний #5

V6: WORD $DEFAULT, 0x180 ;Вектор прерываний #6

V7: WORD $DEFAULT, 0x180 ;Вектор прерываний #7

DEFAULT: IRET ;Возврат из прерываний

ORG 0x01F

X: WORD 0x0000 ;Переменная X

TMP: WORD 0x0000 ;Временная переменная

ADDRX: WORD $X ;Адрес X

MIN: WORD 0xFFD6 ;Нижняя граница значений X

MAX: WORD 0x002A ;Верхняя граница значений X

START: DI ;загрузка начальных векторов прерываний

LD #0xA ;разрешить прерывания и вектор 2

OUT 5 ;(1000|0010=1010) в MR КВУ-2

LD #0xB ;разрешить прерывания и вектор 3

OUT 7 ;(1000|0011=1011) в MR КВУ-3

MAIN:

EI

LD X

ST TMP

INC

CALL $CHECK\_BORDERS

PUSH

LD TMP

PUSH

LD ADDRX

PUSH

CALL $CAS

JUMP MAIN

INT2: ;обработка прерывания на ВУ-2

PUSH

LD X

NOP

IN 4

AND #0x000F

AND X

PUSH

LD TMP

PUSH

LD ADDRX

PUSH

CALL $CAS

NOP

POP

IRET

INT3: ;обработка прерывания на ВУ-3

PUSH

LD X

NOP

ASL

ADD X

NEG

OUT 6

POP

IRET

CHECK\_BORDERS:

CHECK\_MIN:

CMP MIN

BPL CHECK\_MAX

JUMP LD\_MIN

CHECK\_MAX:

CMP MAX

BMI RETURN

LD\_MIN:

LD MIN

RETURN: RET

; CAS(ADDRX; EXPECT; NEW)

DEREF: WORD 0x0000

CAS:

PUSHF; Кладёт флаг прерываний

DI

LD &2

ST DEREF

LD (DEREF)

CMP &3

BNE FAIL

SUCCES:

LD &4

ST (DEREF)

LD #0x1

JUMP EXIT

FAIL:

CLA

EXIT:

POPF

SWAP

ST &3

SWAP

SWAP

POP

SWAP

POP

SWAP

POP

NOP

RET

**Методика проверки:**

1. Поставить HLT вместо NOP
2. Включить КВУ-2 И КВУ-3
3. После нажатия кнопки на КВУ-3 взглянуть на регистр аккумулятора ()
4. Умножить на -3 (принадлежит кольцу целых чисел) число из аккумулятора
5. После следующих действий бэвм посмотреть на аккумулятор, там будет ответ
6. Ввести число в КВУ-2 и поставить готовность
7. Проверить CAS там будет 1, если число записалось и 0 если нет
8. После остановки БЭВМ посмотреть на регистр аккумулятора, там будет лежать текущий Х
9. Нажать продолжение и дождаться остановки, в регистре аккумулятора будет лежать результат побитового маскирования числа и полученного числа
10. Проверить границы можно, дождавшись переполнения, должен произойти сброс значения Х до минимального по ОДЗ

**Трассировка**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Проверка | | | | | | |
| AC(0…7) | -3x | DR | AC(0…7) | DR Кву-2 | AC(DR & 0xF & X) | ans |
| 14 | C4 | C4 | EF | FF | F | F |
| F0 | 30 | 30 | E | F3 | 2 | 2 |
| 4 | F4 | F4 | F4 | 23 | 0 | 0 |
| B | DF | DF | 4 | 18 | 0 | 0 |
|  |  |  |  |  |  |  |

**Выводы:**

В процессе выполнения лабораторной работы был получен опыт работы с подпрограммами и стеком.

**Список литературы:**

1. В.В. Кириллов АРХИТЕКТУРА БАЗОВОЙ ЭВМ Учебное пособие / В.В. Кириллов — 1. — Санкт-Петербург: САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ, 2010 — 142 c.

FAA

CAS