

**Федеральное государственное автономное образовательное учреждение высшего
образования «Национальный исследовательский университет ИТМО»**

Факультет программной инженерии и компьютерной техники

Лабораторная работа №6

По Основам Профессиональной Деятельности

Вариант №14013

Выполнил:

Ступин Тимур Русланович

Группа № Р3108

Проверил:

Вербовой Александр Александрович

Санкт-Петербург 2024

Содержание

Задание.....	3
Текст программы	3
Область допустимых значений.....	3
Расположение данных в памяти	5
Область представления	6
Методика проверки программы	6
Вывод.....	8

Задание

По выданному преподавателем варианту разработать и исследовать работу комплекса программ обмена данными в режиме прерывания программы. Основная программа должна изменять содержимое заданной ячейки памяти (X), которое должно быть представлено как знаковое число. Область допустимых значений изменения X 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 X в соответствии с вариантом задания, а также игнорировать все необрабатываемые прерывания.

1. Основная программа должна уменьшать на 2 содержимое X (ячейки памяти с адресом 013_{16}) в цикле.
2. Обработчик прерывания должен по нажатию кнопки готовности ВУ-3 осуществлять вывод результата вычисления функции $F(X) = -2X - 6$ на данное ВУ, а по нажатию кнопки готовности ВУ-2 вычесть X из содержимого РД данного ВУ, результат записать в X .
3. Если X оказывается вне ОДЗ при выполнении любой операции по его изменению, то необходимо в X записать максимальное по ОДЗ число.

Текст программы

```
ORG 0x0          ; Инициализация векторов прерывания
V0: WORD $DEFAULT, 0x180    ; Вектор прерывания #0
V1: WORD $DEFAULT, 0x180    ; Вектор прерывания #1
V2: WORD $INT2,      0x180    ; Вектор прерывания #2
V3: WORD $INT3,      0x180    ; Вектор прерывания #3
V4: WORD $DEFAULT, 0x180    ; Вектор прерывания #4
V5: WORD $DEFAULT, 0x180    ; Вектор прерывания #5
V6: WORD $DEFAULT, 0x180    ; Вектор прерывания #6
V7: WORD $DEFAULT, 0x180    ; Вектор прерывания #7

ORG 0x13
X: WORD ?        ; Переменная

MIN_V: WORD 0xFFBE    ; Минимальное значение X {-66}
MAX_V: WORD 0x003D    ; Максимальное значение X {61}

; Обработка прерываний по умолчанию
; Сбрасывает готовность со всех неиспользуемых ВУ и выходит из
прерывания
DEFAULT:
    IRET
```

```

        ORG 0x020      ; Загрузка начальных векторов прерываний
START:
        DI      ; Запрет прерываний на период загрузки
        CLA    ; Заносим 0 в АС
        OUT 0x1 ; MR КВУ-0 на вектор 0
        OUT 0x3 ; MR КВУ-1 на вектор 0
        OUT 0xB ; MR КВУ-4 на вектор 0
        OUT 0xE ; MR КВУ-5 на вектор 0
        OUT 0x12 ; MR КВУ-6 на вектор 0
        OUT 0x16 ; MR КВУ-7 на вектор 0
        OUT 0x1A ; MR КВУ-8 на вектор 0
        OUT 0x1E ; MR КВУ-9 на вектор 0
        LD #0xA ; Разрешить прерывания на вектор 2
        OUT 0x5 ; (1000 | 0010 = 1010) в MR КВУ-2
        LD #0xB ; Разрешить прерывания на вектор 3
        OUT 0x7 ; (1000 | 0011 = 1011) в MR КВУ-3
        EI      ; Разрешение прерываний
; Основная программа
PROG:
        DI      ; Запрет прерываний для безопасного доступа к памяти
        LD X   ; Загрузка переменной
        EI      ; Разрешение прерываний
        SUB #0x2 ; Уменьшение АС
        CALL CHECK ; Проверка принадлежности ОДЗ
        DI      ; Запрет прерываний для безопасного доступа к памяти
        ST X   ; Сохранение переменной
        EI      ; Разрешение прерываний
        JUMP PROG ; Циклический переход

; Обработка прерывания ВУ-2
INT2:
        CALL CHECK ; Проверка принадлежности ОДЗ
        PUSH ; Сохранили АС в стек
        NOP ; Отладочная точка останова
        CLA ; Очистка АС
        IN 0x4 ; Прочитали данные с ВУ-2
        SXTB ; Расширение знака для корректной работы
        NOP ; Отладочная точка останова
        SUB (SP+0) ; Вычли X
        NOP ; Отладочная точка останова
        CALL CHECK ; Проверка принадлежности ОДЗ
        ST (SP+0) ; Сохранили результат в X
        NOP ; Отладочная точка останова
        POP ; Вернули АС
        IRET ; Возврат из прерывания

; Обработка прерывания ВУ-3
INT3:

```

```

CALL CHECK      ; Проверка принадлежности ОДЗ
PUSH          ; Сохранили АС в стек, чтобы не испортить
NOP          ; Отладочная точка останова
ASL
NEG
SUB #0x6
OUT 0x6    ; Вывод результата на ВУ-3
NOP          ; Отладочная точка останова
POP          ; Возврат значения АС
IRET          ; Возврат из прерывания

; Проверка принадлежности ОДЗ
; Проверяет значения из АС
; Если значение не принадлежит ОДЗ то оно приводиться к нему
CHECK:
    CMP $MIN_V    ; Проверка на минимум
    BGE CHECK_MAX ; x >= min_v, переход на проверку максимума
    LD $MAX_V     ; Сохранение максимума
    JUMP RETURN    ; Выход из функции
CHECK_MAX:
    CMP $MAX_V    ; Сравнение с максимумом
    BLT RETURN    ; Возврат если x < max_v
    LD $MAX_V     ; Сохранение максимального значения
RETURN:   RET

```

Область допустимых значений

- $-128 \leq -2X - 6 \leq 127$
- $-122 \leq -2X \leq 133$
- $-66 \leq X \leq 61$
- $-66 = 1111\ 1111\ 1011\ 1110 = FFBE$
- $61 = 0000\ 0000\ 0011\ 1101 = 003D$
- Значит $X \in [-66; 61]$

Расположение данных в памяти

- Вектора прерываний: 0x000 – 0x00F
- Переменные: 0x013 – 0x015
- Основная программа: 0x02F – 0x037
- Подпрограмма обработки прерываний с ВУ-2: 0x038 – 0x045
- Подпрограмма обработки прерываний с ВУ-3: 0x046 – 0x04F
- Подпрограмма обработки для проверки ОДЗ: 0x050 – 0x057

Область представления

- X, MIN_V, MAX_V – знаковое 16-разрядное целое число
- DR (регистр данных) ВУ-2 и ВУ-3 – знаковое 8-разрядное целое число

Методика проверки программы

Проверка обработки прерываний:

1. Загрузить комплекс программ в память БЭВМ
2. Во всех точках останова заменить NOP на HLT
3. Запустить БЭВМ в режиме РАБОТА
4. Ввести в регистр данных ВУ-2 тестовые данные
5. Установить готовность ВУ-2
6. Дождаться остановки программы
7. Запомнить текущее значени АС (считаем что это X)
8. Нажать кнопку ПРОДОЛЖЕНИЕ
9. Дождаться остановки программы
10. Сверить значение в аккумуляторе со значением введенным на ВУ-2
11. Нажать кнопку ПРОДОЛЖЕНИЕ
12. Дождаться остановки программы
13. Записать значение из аккумулятора, сравнив его с регистром данных ВУ-2
14. Нажать кнопку ПРОДОЛЖЕНИЕ
15. Дождаться остановки программы
16. Рассчитать ожидаемы результат вычисления выражения (DR ВУ-2) – X и сравнить его с полученным в аккумуляторе
17. Нажать кнопку ПРОДОЛЖЕНИЕ
18. Дождаться остановки программы
19. Проверить корректность приведения вычисленного значения в ОДЗ
20. Нажать кнопку ПРОДОЛЖЕНИЕ
21. Установить готовность ВУ-3
22. Дождаться остановки программы
23. Записать значение переменной X из аккумулятора
24. Нажать кнопку ПРОДОЛЖЕНИЕ
25. Дождаться остановки программы

26. Рассчитать ожидаемый результат вычисления функции $F(X)$ и сравнить его с полученным в аккумуляторе и на ВУ-3

Проверка основной программы

1. Загрузить комплекс программы в память БЭВМ
2. Запустить БЭВМ в режиме РАБОТА
3. Дождаться достижения крайних значений аккумулятора для проверки корректности приведения к ОДЗ

Прерывание на ВУ-2				
<i>DR BY-2</i>	<i>X до</i>	<i>X без ОДЗ</i>	<i>X после</i>	<i>X ожидаемое</i>
$12_{16}(18)$	$FFF8_{16}(-8)$	$001A_{16}(26)$	$001A_{16}(26)$	$001A_{16}(26)$
$FF_{16}(-1)$	$FFF0_{16}(-16)$	$000F_{16}(15)$	$000F_{16}(15)$	$000F_{16}(15)$
$7F_{16}(127)$	$0007_{16}(7)$	$0078_{16}(120)$	$003D_{16}(61)$	$003D_{16}(61)$

Прерывание на ВУ-3			
<i>X</i>	<i>F(X) ожидаемое</i>	<i>F(X) полученное</i>	<i>DR BY-3</i>
$002B_{16}(43)$	$FFA4_{16}(-92)$	$FFA4_{16}(-92)$	$A4_{16}(-92)$
$001F_{16}(31)$	$FFBC_{16}(-68)$	$FFBC_{16}(-68)$	$BC_{16}(-68)$
$FFED_{16}(-19)$	$0020_{16}(32)$	$0020_{16}(32)$	$20_{16}(32)$

Основная программа		
<i>AC до</i>	<i>AC ожидаемое</i>	<i>AC после</i>
$FFC0_{16}(-64)$	$FFBE_{16}(-66)$	$FFBE_{16}(-66)$
$FFBE_{16}(-66)$	$003D_{16}(61)$	$003D_{16}(61)$
$003D_{16}(61)$	$003B_{16}(59)$	$003B_{16}(59)$

Дополнительное задание

Программа в цикле делает +1 до 32 и обнуляется.

Прерывание 1: вывести значение переменой 1, переключить программу на увеличение этого значения по таким таким же правилам

Прерывание 2: вывести значение переменной 2, переключить программу на увеличение этого значения по таким таким же правилам

[Ссылка на код](#)

Вывод

В ходе работы я изучил организацию процесса прерывания программы и исследовал порядок функционирования БЭВМ при обмене данными с внешними устройствами в режиме прерывания программы. Я разработал программу для обмена с устройствами ВУ-2 и ВУ-3 по прерываниям, а также протестировал её по составленной методике проверки работоспособности.