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

Факультет программной инженерии и компьютерных технологий

Основы профессиональной деятельности

Лабораторная работа №6

Вариант 1003

Выполнила:

Павличенко Софья Алексеевна, Р3115

Проверила:

Ершова Анна Ильинична

Санкт-Петербург 2024г.

Оглавление

[Цель 3](#_Toc164286865)

[Задание 4](#_Toc164286866)

[Текст исходной программы 5](#_Toc164286867)

[Описание программы 6](#_Toc164286868)

[Область определения 6](#_Toc164286869)

[ОДЗ 6](#_Toc164286870)

[Таблица трассировки 7](#_Toc164286871)

[Дополнительное задание 11](#_Toc164286872)

[Вывод 15](#_Toc164286873)

# Цель

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

# 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 (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 декрементировать содержимое X (ячейки памяти с адресом 01C16) в цикле.
2. Обработчик прерывания должен по нажатию кнопки готовности ВУ-1 осуществлять вывод результата вычисления функции F(X)=-3X-2 на данное ВУ, a по нажатию кнопки готовности ВУ-2 выполнить операцию побитового 'ИЛИ' 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 максимальное по ОДЗ число.

# Текст исходной программы

|  |  |
| --- | --- |
| ORG 0x0 |  |
| V0: WORD $DEFAULT, 0x180 | ; инициализация векторов прерывания |
| V1: WORD $INT1, 0x180 |
| V2: WORD $INT2, 0x180 |
| V3: WORD $DEFAULT, 0x180 |
| V4: WORD $DEFAULT, 0x180 |
| V5: WORD $DEFAULT, 0x180 |
| V6: WORD $DEFAULT, 0x180 |
| V7: WORD $DEFAULT, 0x180 |
|  |  |
| DEFAULT: IRET | ; возврат |
|  |  |
| ORG 0x1C |  |
| X: WORD 0x002A | ; по умолчанию X равен максимуму |
| MAX: WORD 0x002A | ; максимальное по ОДЗ значение X |
| MIN: WORD 0xFFFF | ; минимальное по ОДЗ значение X |
|  |  |
| ORG 0x20 |  |
| START: DI | ; запрещаем прерывания |
| CLA | ; обнуляем аккумулятор |
| OUT 1 | ; MR КВУ-0 на вектор 0 |
| OUT 7 | ; MR КВУ-3 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 #9 | ; разрешаем прерывания и MR КВУ-1 на вектор 1 |
| OUT 3 |
| LD #0xA | ; разрешаем прерывания и MR КВУ-2 на вектор 2 |
| OUT 5 |
| JUMP PROG |  |
|  |  |
| ORG 0x30 |  |
| PROG: EI | ; разрешаем прерывания |
| INCLP: DI | ; запрещаем прерывания для обеспечения целостности операции |
| LD X | ; декрементируем X |
| DEC |  |
| ST X |  |
| CALL CHECK | ; проверяем X на нахождение в ОДЗ |
| EI | ; обратно разрешаем прерывания |
| JUMP INCLP |  |
|  |  |
| CHECK: CMP MIN | ; сравниваем X с минимумом |
| BGE EXIT | ; не выходит за границу – возвращаемся в основную программу |
| NOP | ; исследуем значение X в момент его выхода за пределы ОДЗ |
| LD MAX | ; загружаем максимум в X |
| ST X |
| NOP | ; исследуем правильность записи в X максимального по ОДЗ значения |
| EXIT: RET |  |
|  |  |
|  |  |
| ORG 0x40 |  |
| INT1: | ; обработчик прерывания по вектору 1 |
| LD X | ; загрузили X в аккумулятор |
| NOP | ; исследуем значение X в момент возникновения прерывания |
| ASL | ; AC = 2X |
| ADD X | ; AC = 3X |
| NEG | ; AC = -3X |
| SUB #0x02 | ; AC = -3X - 2 |
| OUT 2 | ; записываем AC в DR ВУ1 |
| NOP | ; исследуем правильность подсчёта функции и вывода результата на ВУ-1 |
| IRET | ; возврат из обработчика |
|  |  |
| ORG 0x50 |  |
| INT2: | ; обработчик прерывания по вектору 2 |
| LD X | ; загрузили X в аккумулятор |
| NOP | ; исследуем значение X в момент возникновения прерывания |
| IN 4 | ; записываем DR ВУ-2 в AC |
| OR X | ; побитовое ИЛИ с X |
| ST X | ; сохраняем X |
| NOP | ; исследуем правильность подсчёта |
| IRET | ; возврат из обработчика |

# Описание программы

## Область определения

X – 8-разрядное беззнаковое число

## ОДЗ

# 

# Методика проверки

## Проверка обработки прерываний:

1. Загрузить комплекс программ в память БЭВМ.
2. Изменить все значения точки останова NOP на HLT.
3. Запустить программу в режиме “Работа”.

### Обработка прерываний ВУ-1:

1. Установить “Готовность ВУ-1”.
2. Дождаться останова.
3. Записать текущее значение X, которое находится в аккумуляторе.
4. Продолжить выполнение программы, дождаться останова.
5. Записать результат обработки прерывания - содержимое DR ВУ-1.
6. Рассчитать и записать ожидаемое значение результата обработки прерывания.

### Обработка прерываний ВУ-2:

1. Заполнить содержимое DR ВУ-2 и записать его.
2. Установить “Готовность ВУ-2”.
3. Дождаться останова.
4. Записать текущее значение X, которое находится в аккумуляторе.
5. Продолжить выполнение программы, дождаться останова.
6. Записать результат обработки прерывания – содержимое AC.
7. Рассчитать и записать ожидаемое значение результата обработки прерывания.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Прерывание ВУ-1** | | | **Прерывание ВУ-2** | | |
| X | Ожидаемый результат | Полученный результат | DR ВУ-2 | Ожидаемый результат | Полученный результат |
| 2A­­­16 = 42­­­­10 | -3\*42 – 2 = -12810 = 8016 | 8016 | 5416 | 0010 1010 OR  0101 0100 =  0111 1110 = 7E16 | 7E16 |
| 0 | -3\*0 – 2 = -210 = FE16 | FE16 | 9916 | 0000 0000 OR  1001 1001 =  1001 1001 = 9916 | 9916 |
| D516 = -43­10 | -3\*(-43) – 2 = 12710 = 7F16 | 7F16 | 7A16 | 1101 0101 OR  0111 1010 =  1111 1111 = FF16 | FF16 |

## Проверка основной программы:

1. Загрузить комплекс программ в память БЭВМ.
2. Изменить все значения точки останова NOP на HLT.
3. Записать начальное значение X.
4. Запустить программу в режиме “Работа”.
5. Дождаться останова.
6. Записать текущее значение X, которое находится в аккумуляторе.
7. Продолжить выполнение программы, дождаться останова.
8. Записать результат обработки выхода за границу ОДЗ - содержимое DR AC.
9. Записать ожидаемое значение результата обработки выхода за границу ОДЗ.

|  |  |  |  |
| --- | --- | --- | --- |
| **Основная программа** | | | |
| Начальное X | X | Ожидаемый результат | Полученный результат |
| 2A­­­16 = 42­­­­10 | D4 | 2A16 | 2A16 |
| 0 | D4 | 2A16 |
| D516 = -43­10 | D4 | 2A16 |

# Вывод