*Факультет программной инженерии и компьютерной техники*

*Направление подготовки: 09.03.04 – Программная инженерия,*

*Системное и прикладное программное обеспечение*

*Дисциплина «Программирование»*

**Лабораторная работа по ОПД №6**

**Вариант №4682**

Выполнил:

Ткачев Денис Владимирович

Группа P3111

Преподаватели:

Остапенко Ольга Денисовна

Оглавление

[Задание 2](#_Toc192711942)

[Текст исходной программы 3](#_Toc192711943)

[Описание программы 3](#_Toc192711944)

[Текст программы на ассемблере: 3](#_Toc192711945)

[Текст исходной программы 3](#_Toc192711946)

[Назначение программы: 5](#_Toc192711947)

[Расположение в памяти БЭВМ программы, исходных данных и результатов 5](#_Toc192711948)

[Область представления 5](#_Toc192711949)

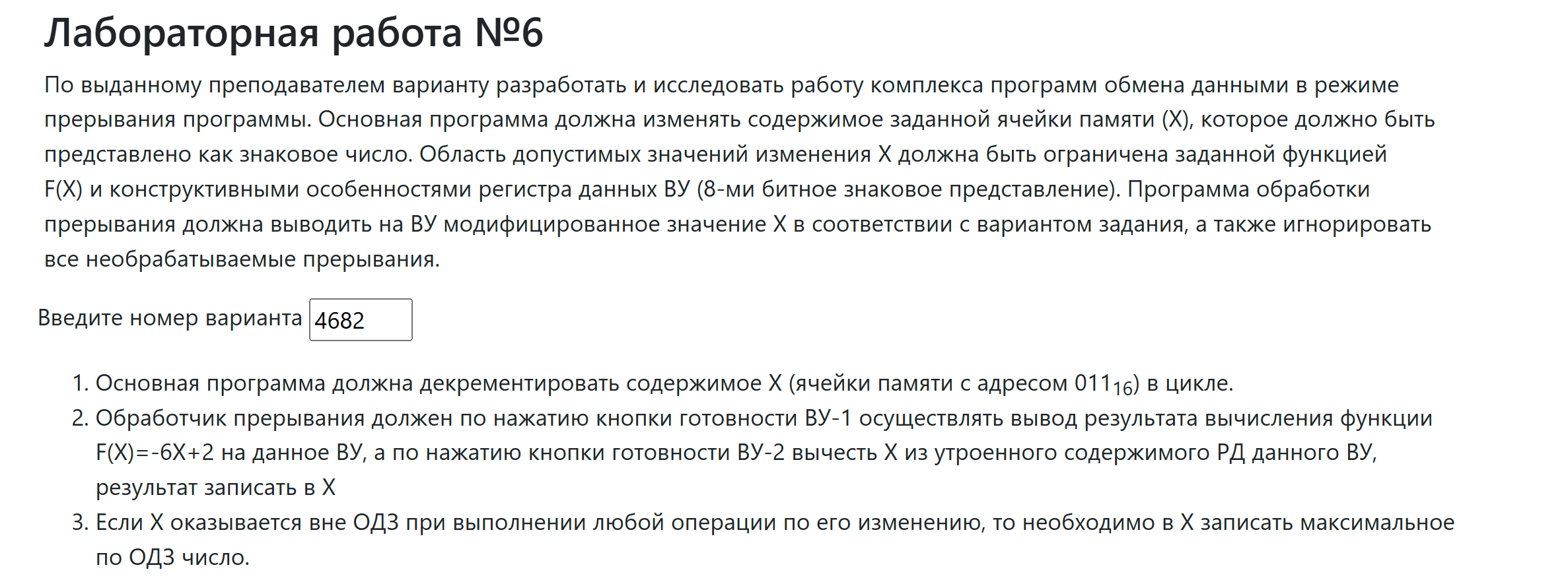
[Область допустимых значений 5](#_Toc192711950)

[Таблица трассировки 6](#_Toc192711951)

[Вывод 7](#_Toc192711952)

## 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.



1. Основная программа должна декрементировать содержимое X (ячейки памяти с адресом 01116) в цикле.
2. Обработчик прерывания должен по нажатию кнопки готовности ВУ-1 осуществлять вывод результата вычисления функции F(X)=-6X+2 на данное ВУ, a по нажатию кнопки готовности ВУ-2 вычесть Х из утроенного содержимого РД данного ВУ, 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 на ассемблере:

1. ORG 0x0 ; Инициализация векторов прерывания

2. V0: WORD $DEFAULT 0x180 ; вектор прерывания #0

3. V1: WORD $INT1, 0x180 ; вектор прерывания #1

4. V2: WORD $INT2, 0x180 ; вектор прерывания #2

5. V3: WORD $DEFAULT 0x180 ; вектор прерывания #3

6. V4: WORD $DEFAULT 0x180 ; вектор прерывания #4

7. V5: WORD $DEFAULT 0x180 ; вектор прерывания #5

8. V6: WORD $DEFAULT 0x180 ; вектор прерывания #6

9. V7: WORD $DEFAULT 0x180 ; вектор прерывания #7

10.

11. ORG 0x11

12. X: WORD ? ; переменная

13.

14. MAX: WORD 0x0016 ; 21, Максимальное значение X, которое удовлетворяет ОДЗ

15. MIN: WORD 0xFFEB ; -20,  Минимальное значение X, которое удовлетворяет ОДЗ

16. DEFAULT: IRET ; Обработка прерывания по умолчанию, просто возврат прерывания

17.

18. START: DI

19.        CLA ; по ТЗ, программа должна игнорировать все необработанные прерывания

20.        OUT 0x1 ; Значение MR КВУ-0 равно 0 - на вектор 0, запрет прерываний для неиспользуемых ВУ

21.        OUT 0x7 ; MR КВУ-3 на вектор 0, запрет прерываний для неиспользуемых ВУ

22.        OUT 0xB ; MR КВУ-4 на вектор 0, запрет прерываний для неиспользуемых ВУ

23.        OUT 0xD ; MR КВУ-5 на вектор 0, запрет прерываний для неиспользуемых ВУ

24.        OUT 0x11 ; MR КВУ-6 на вектор 0, запрет прерываний для неиспользуемых ВУ

25.        OUT 0x15 ; MR КВУ-7 на вектор 0, запрет прерываний для неиспользуемых ВУ

26.        OUT 0x19 ; MR КВУ-8 на вектор 0, запрет прерываний для неиспользуемых ВУ

27.        OUT 0x1D ; MR КВУ-9 на вектор 0, запрет прерываний для неиспользуемых ВУ

28.

29.        LD #0x9 ; Для загрузки  из аккумулятора в MR ВУ1 (1000|0001=1001)

30.        OUT 3 ; Разрешение прерываний для ВУ 1

31.        LD #0xA ; Для загрузки  из аккумулятора в MR ВУ2 (1000|0010=1010)

32.        OUT 5 ; Разрешение прерываний для ВУ 2

33.        EI ; Разрешение прерывания

34.        JUMP $MAIN

35.

36. ; Основная программа

37. MAIN: DI ; Запрет прерываний, для того, чтобы обеспечить полное выполнение программы

38.     LD X ; Загрузка переменной X

39.     DEC ; Декрементация в цикле

40.     CALL CHECK ; Проверка значения на ОДЗ

41.     ST X ; Сохранение значения X

42.     EI ; Разрешение прерывания

43.     JUMP MAIN ; Возврат в цикл

44.

45. ; Обработка прерывания ВУ 1

46. INT1:  DI ; Запрет на прерывание во время обработки прерывания

47.        LD X ; Загрузка переменной X

48.        NOP ; Отладочная точка остановки (NOP/HLT) для отладки. Всегда можно NOP заменить на HLT

49.        ASL ; X --> 2X

50.        ASL ; 2X --> 4X

51.        ASL ; 4X --> 8X

52.        SUB X ; 8X --> 7X

53.        SUB X ; 7X --> 6X

54.        ADD #2 ; -6X --> -6X + 2

55.        NOP

56.        OUT 2 ; Вывод F(x) = -6X + 2 в ВУ 1

57.        EI ; Разрешение прерывания

58.        IRET ; Возврат из обработки прерывания

59.

60. TMP: WORD ?

61. ; Обработка прерывания ВУ 2

62. INT2: DI ; Запрет на прерывание во время обработки прерывания

63.       CLA

64.       IN 4 ; Считывание DR ВУ 2

65.       NOP

66.       LD TMP ; Записываем в TMP значение DR ВУ 2

67.       ASL ; TMP --> 2TMP

68.       ADD TMP ; 2TMP --> 3TMP

69.       SUB X ;

70.       CALL CHECK ; Проверка значения на ОДЗ

71.       ST X ;

72.       NOP

73.       EI ; Разрешение прерывания

74.       IRET ; Возврат из обработки прерывания

75.

76. ; Проверка принадлежности X к ОДЗ

77. CHECK: CMP MAX ; Если X < MAX, то

78.        BMI CHECK\_MIN ; проверка на нижнюю границу X > MIN

79.        JUMP LD\_MAX ; X не входит в ОДЗ, поэтому X будет принимать максимальное значение

80. CHECK\_MIN: CMP MIN ; Проверка X > MIN

81.            BPL RETURN ; Если X > MIN, то X принадлежит ОДЗ, возвращаемся в основную программу

82.            JUMP LD\_MAX

83. LD\_MAX: LD MAX ; Загрузка максимального числа в AC, Если X не принадлежит к ОДЗ

84. RETURN: RET ; Метка возврата из проверки на ОДЗ

85.

Назначение программы:

Основная программа выполняет декрементацию переменной X, в бесконечном цикле.

По команде готовности ВУ-1 вычисляет значение функции F(x) = -6x+2 и записывает результат на ВУ-1

По команде готовности ВУ-2, программа достаёт значение РД данного ВУ, вычисляет значение функции F(x, y)=3y-x, где y – значение РД ВУ 2. Затем сохраняет результат в переменную Х.

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

0x0-0xF – Векторы прерывания

0x11-0x13 – Переменные X, MIN и MAX ОДЗ.

0x14-0x4e - Программа

## Область представления

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

DR КВУ – 8-ми разрядное целое знаковое число.

## Область допустимых значений

Число X ϵ [FFEC; 0015]

## **Методика проверки программы**

Проверка обработки прерываний:

1. Загрузить текст программы в БЭВМ.

2. Заменить NOP по нужному адресу на HLT.

3. Запустить программу в режиме РАБОТА.

4. Установить «Готовность ВУ-1».

5. Дождаться остановки.

6. Записать текущее значение X из памяти БЭВМ:

1. Запомнить текущее состояние счетчика команд.

2. Ввести в клавишный регистр значение 0x011

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 остановки.

13. Записать текущее значение X из памяти БЭВМ, также, как и в пункте 6.

14. Нажать «Продолжение».

15. Записать текущее значение X из памяти БЭВМ, также, как и в пункте 6.

16. Рассчитать ожидаемое значение переменной X после обработки прерывания

Проверка основной программы:

1. Загрузить текст программы в БЭВМ.

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

3. Запустить программу в режиме остановки.

4. Пройти нужное количество шагов программы, убедиться, что при уменьшении X на 1, то после момента, когда он равен -21, происходит сброс значения в максимальное по ОДЗ (21).

|  |  |  |
| --- | --- | --- |
| Основная программа | | |
| Старое значение AC(0..7) | Ожидаемое значение | Новое значение AC(0..7) |
| 0x1 (1) | 0x0(0) | 0x0(0) |
| 0x10 (16) | 0xF (15) | 0xF (15) |
| 0xFFFF (-1) | 0xFFFE (-2) | 0xFFFE (-2) |

|  |  |  |
| --- | --- | --- |
| Проверка ВУ-1 | | |
| Значение AC (0..7) | Ожидаемое значение в DR ВУ1 -6+2 | Результат АС (0..7) |
| 0x1 (1) | 0xFFFС (-4) | 0xFFFC (-4) |
| 0x10 (16) | 0xFFA2 (-94) | 0xFFA2 (-94) |
| 0xFFFF (-1) | 0x8 (8) | 0x8 (8) |

|  |  |  |
| --- | --- | --- |
| Проверка ВУ-2 | | |
| Значение АС(0..7) | Значение DR ВУ-1 | Результат X |
| 0x1 (1) | 0x5 (5) | 0xE (14) |
| 0x10 (16) | 0x12 (18) | 0x26 (38) |
| 0xFFFF (-1) | 0xFFFA (-6) | 0xFFEF (-17) |

## Вывод

В ходе выполнения лабораторной работы я изучил обмен данными с ВУ-1 и ВУ-2 в режиме прерываний, также изучил цикл прерывания и циклы исполнения новых команд. Также закрепил знания в написании программ на ассемблере БЭВМ.