*Факультет программной инженерии и компьютерной техники*

*Дисциплина «Основы профессиональной деятельности»*



**Отчет**

**По лабораторной работе №6**

**Вариант №92851**

Выполнил:

Саранча Павел Александрович

Группа: Р3109

Преподаватель:

Саржевский Иван Анатольевич

Г. Санкт-Петербург, 2024 г.

# 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 инкрементировать содержимое X (ячейки памяти с адресом 03816) в цикле.
2. Обработчик прерывания должен по нажатию кнопки готовности ВУ-1 осуществлять вывод результата вычисления функции F(X)= -6X-10 на данное ВУ, a по нажатию кнопки готовности ВУ-2 выполнить операцию побитового маскирования, оставив 3-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.

# Описание программы

**Назначение программы**

1. Основная программа должна увеличивать на 1 содержимое X (ячейки памяти с адресом 03816) в цикле.
2. Обработчик прерывания должен по нажатию кнопки готовности ВУ-1 осуществлять вывод результата вычисления функции F(X)= -6X-10 = -(6X+10) на данное ВУ, a по нажатию кнопки готовности ВУ-2 выполнить операцию побитового маскирования, оставив 3-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.

**Текст программы**

|  |
| --- |
| ORG 0x0  V0: WORD $default, 0X180 ; бит P – в программе и бит W – работа, без этого стоп.  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  ORG 0x038  X: WORD ?  max: WORD 0x001B ; 27, максимальное значение Х  min: WORD 0xFFE8 ; -24, минимальное значение Х  default: IRET ; Обработка прерывания по умолчанию  START: DI  CLA  OUT 0x1 ; Запрет прерываний для неиспользуемых ВУ  OUT 0x7  OUT 0xB  OUT 0xD  OUT 0x11  OUT 0x15  OUT 0x19  OUT 0x1D  LD #0x9 ; Загрузка в аккумулятор MR (1000|0001=1001)  OUT 3 ; Разрешение прерываний для 1 ВУ  LD #0xA ; Загрузка в аккумулятор MR (1000|0010=1010)  OUT 5 ; Разрешение прерываний для 2 ВУ  EI  main: DI ; Запрет прерываний чтобы обеспечить атом. операции  LD X  NOP  INC ; Инкремент как сказано в задании  CALL check  ST X  NOP  EI  JUMP main  int1: DI ; Обработка прерывания на ВУ-1  LD X  NOP  ASL  ASL  ADD X  ADD X  ADD #10  NEG  NOP  OUT 2  EI  IRET  int2: DI ; Обработка прерывания на ВУ-2  IN 4  NOP  AND #0x07  AND X  ST X  NOP  EI  IRET  check: ; Проверка принадлежности X к ОДЗ  check\_min: CMP min ; Если x > min переход на проверку верхней границы  BPL check\_max  JUMP ld\_min ; Иначе загрузка min в аккумулятор  check\_max: CMP max ; Проверка пересечения верхней границы X  BMI return ; Если x < max переход  ld\_min: LD min ; Загрузка минимального значения в X  return: RET ; Метка возврата из проверки на ОДЗ |

**Область допустимых значений**

* Число X ϵ [FFE8; 001B]

**Расположение данных в памяти**

* Вектор прерываний: 0x000 – 0x00F
* Переменные: 0x038 – 0x042
* Программа: 0x03C – 0x06D

**Область представления**

* X, min, max – знаковое 16-ричное целое число;
* DR КВУ – 8-ми разрядное целое знаковое число.

# 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.

# Методика проверки программы

**Методика проверки ВУ-1:**

1. Изменить значение переменной X в коде на нужное вам для проверки значение. Требуется, чтобы данное число находилось в данном промежутке [-2416 ; 2716]
2. Подготовить точки остановы (в int1)
3. Скомпилировать код
4. Поменять STOP на RUN
5. Запустить программу в автоматическом режиме с адреса 0x52 (F7)
6. Установить “Готовность ВУ-1”
7. Дождаться останова
8. Проверить содержимое X в аккумуляторе
9. Продолжить исполнение (Continue F8)
10. Выписать содержимое аккумулятора (Результат после выполнения формулы) в момент останова программы
11. Сравнить выписанное значение с формулой Y = 5X-3
12. Программа остановлена. Вы выполнили проверку ВУ-1

**Методика проверки ВУ-2:**

1. Подготовить точки остановы (в int2)
2. Изменить значение переменной X в коде на нужное вам для проверки значения.

Требуется, чтобы данное число находилось в данном промежутке [-2416; 2716]

1. Запомнить число, которое было введено в X
2. Скомпилировать код
3. Поменять STOP на RUN
4. Записать число в DR ВУ-2. Требуется, чтобы вводимое число находилось в промежутке [-2416 ; 2716]
5. Установить “Готовность ВУ-2”
6. Запустить программу в автоматическом режиме с адреса 0x5F (F7)
7. Дождаться останова
8. Записать значение AC(переменная X = DR ВУ-2)
9. Продолжить исполнение (Continue F8)
10. Дождаться останова
11. Записать значение AC (результат выполнения функции DR & 0x7 & X)
12. Программа остановлена. Вы выполнили проверку ВУ-2

**Методика проверки основной программы:**

1. Записать в переменную X любое значение в диапазоне ОДЗ
2. Подготовить точки остановы в main
3. Запустить программу в режиме работы с дефолтного адреса START = 0x3C.
4. Дождаться останова
5. Записать значение AC – переменной X без инкремента
6. Нажать кнопку продолжить
7. Дождаться останова
8. Проверить AC, возможны два исхода: (Либо AC = AC+1 либо сброс AC на min значение -24 = FFE816 в случае, если инкрементируется max значение 27 и случится выход за рамки ОДЗ)
9. Можно убедиться, что программа работает исправно. Проверка выполнена

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Прерывание ВУ-1 | | | Прерывание ВУ-2 | | |
| AC  (0...7) | Ожидание  -(6\*X+10) | DR | переменX  (0…7) | AC = DR  КВУ-2 | AC  (DR & 0x7 & X) |
| 1316 (19) | 8416 (-124) | 8416(-124) | 916 (9) | (136) | 016 (0) |
| FF16 (-1) | FC16 (-4) | FC16 (-4) | 916 (9) |  | 116 (1) |
| 0A16 (10) | BA16 (-70) | BA16 (-70) | 1B16 (27) | (175) | 316 (3) |

|  |  |  |
| --- | --- | --- |
| Основная программа | | |
| AC | Ожидание | AC |
| 1316 (19) | 1416 (20) | 1416 (20) |
| 1B16 (27) | FFE816 (-24) | FFE816 (-24) |
| 1C16 (28) | FFE816 (-24) | FFE816 (-24) |