Университет ИТМО

Факультет программной инженерии и компьютерной техники



Лабораторная работа по основам профессиональной деятельности №6

Вариант №3007

Группа: P3130

Студент: Толстых М. А.

Преподаватель: Перцев Т. С.

г. Санкт-Петербург

2023

**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 инкрементировать содержимое X (ячейки памяти с адресом 02316) в цикле.
2. Обработчик прерывания должен по нажатию кнопки готовности ВУ-3 осуществлять вывод результата вычисления функции F(X)=-4X-9 на данное ВУ, a по нажатию кнопки готовности ВУ-2 выполнить операцию побитового 'Исключающее ИЛИ-НЕ' 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.

**Ассемблерный код:**

|  |  |  |  |
| --- | --- | --- | --- |
| Метка | Команда | Аргумент | Комментарий |
|  | ORG | 0x0 |  |
| 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 | 0x023 |  |
| START: | DI |  | Запрет прерываний |
|  | LD | #0xA | Set interrupt control of external device 3: MR (1000 | 0011 = 1011) |
|  | OUT | 5 | Загрузка вектора прерывания в ВУ-2 |
|  | LD | #0xB |  |
|  | OUT | 7 | Загрузка вектора прерывания в ВУ-3 |
|  | CLA |  |  |
|  | LD | $X |  |
| MAIN: | DI |  | Запрет прерываний |
|  | LD X |  |  |
|  | INC |  |  |
|  | CALL | CHECK | Вызов подпрограммы проверки ОДЗ |
|  | ST X |  |  |
|  | EI |  | Разрешение прерываний |
|  | JUMP | MAIN |  |
| INT3: | HLT |  |  |
|  | ASL |  | Вычисление значения 3x+2  -4x-9 |
|  | ASL |  |
|  | NEG |  |
|  | SUB | #0x9 |
|  | OUT | 6 | Вывод результата функции на ВУ-3 |
|  | HLT |  |  |
|  | CALL | CHECK | Вызов подпрограммы проверки ОДЗ |
|  | IRET |  | Возврат из прерывания |
| INT2: | HLT |  |  |
|  | IN | 4 | Чтение значения из ВУ-2 |
|  | SXTB |  | Расширение знака |
|  | ST | $VU\_VAL | Логическая операция “Исключающее ИЛИ-НЕ” |
|  | NOT |  |
|  | ST | $NOT\_VU\_VAL |
|  | LD | $X |
|  | NOT |  |
|  | ST | $NOT\_X |
|  | AND | $NOT\_VU\_VAL |
|  | ST | $NOT\_X\_VU |
|  | LD | $X |
|  | AND | $VU\_VAL |
|  | OR | $NOT\_X\_VU |
|  | CALL | CHECK |  |
|  | HLT |  |  |
|  | IRET |  | Возврат из прерывания |
|  | CMP | $MAX | Проверка X> = MAX |
| CHECK: | BEQ | SAVE | Проверка X> = MAX  Проверка X <MIN |
|  | BLT | SETMIN |
|  | CMP | $MIN | Проверка X <MIN |
|  | BGE | SETMIN |
|  | JUMP | SAVE |
|  | LD | $MIN |  |
| SETMIN: | ST | $X |  |
| SAVE: | RET |  |  |
| EXIT\_CHECK: | PUSH |  |  |
| DEFAULT: | CLA |  |  |
|  | OUT | 0x1 | Загрузка 0 в MR не использующихся ВУ, потому что в настоящих процессорах нельзя гарантировать 0. |
|  | OUT | 0x3 | Загрузка 0 в MR не использующихся ВУ, потому что в настоящих процессорах нельзя гарантировать 0. |
|  | OUT | 0xB |
|  | OUT | 0xD |
|  | OUT | 0x11 |
|  | OUT | 0x15 |
|  | OUT | 0x19 |
|  | OUT | 0x1D |
|  | POP |  |
|  | IRET |  | Возврат из прерывания |
|  | WORD | 0x0 | Значение, прочитанное из ВУ-2 |
| VU\_VAL: | WORD | 0x0 | Отрицание значения ВУ-2 |
| NOT\_VU\_VAL: | WORD | 0x0 | Отрицание переменной X |
| NOT\_X: | WORD | 0x0 | Логическое И NOT\_VU\_VAL и NOT\_X |
| NOT\_X\_VU: | WORD | 29 | Переменная X |
| X: | WORD | -34 | Минимальное значение переменной X |
| MIN: | WORD | 29 | Максимальное значение переменной X |
| MAX: |  |  |  |

**Назначение комплекса программ:**

Обмен данными с ВУ в режиме прерывания

**Назначение программы:**

Постоянное увеличивание переменной X на 1 и проверка на вхождение в ОДЗ

**Назначение подпрограммы INT3:**

Вывод результата вычисления функции F(X) = -4X - 9 на ВУ-3

**Назначение подпрограммы INT2:**

Чтение данных из ВУ-2, выполнение логической операции “Исключающее ИЛИ-НЕ”, проверка на вхождение результат в диапазон и запись результата в X

**Назначение подпрограммы CHECK:**

Проверка выхождения X за диапазон [MIN, MAX], и запись в X минимального числа из ОДЗ если X в него не входит

**Назначение подпрограммы DEFAULT:**

Деактивация не использующихся внешних устройств

**Область представления и область допустимых значений исходных данных и результата:**

* Область представления:

X, MIN, MAX, VU\_VAL, NOT\_VU\_VAL, NOT\_X, NOT\_X\_VU: [-215; 215 – 1]

* Область допустимых значений исходных данных и результата:

MIN = -34, MAX = 29

**Расположение в памяти ЭВМ программы, исходных данных и результатов:**

* Инициализация векторов прерывания: [0; 00F]
* Исходные данные программы: [023; 029]
* Программа: [02A; 02E]
* Подпрограмма обработки прерывания ВУ-3: [02F; 037]
* Подпрограмма обработки прерывания ВУ-2: [038; 048]
* Подпрограмма проверки числа во вхождение в ОДЗ: [049; 04E]
* Подпрограмма обработки прерываний для непредусмотренных ВУ: [052; 05D]

# Методика проверки

Убедиться, что выставлен режим работы “Run” и запустить программу, нажав на кнопку “Start”

1. **Основная программа**

1. Загрузить текст программы в БЭВМ.

2. Записать в переменную X максимальное по ОДЗ значение (29)

3. Запустить программу в режиме остановки.

4. Пройти нужное количество шагов программы, убедиться, что при увеличении

X на 1, до того момента, когда он равен 29, происходит сброс значения в

Мин. по ОДЗ.

1. **Обработка прерывания ВУ-3**
   1. Открыть ВУ, выставить готовность ВУ нажав кнопку “Ready”
   2. Дождаться остановки программы
   3. Из аккумулятора (AC) записать значение X в таблицу
   4. Продолжить выполнение программы, нажав кнопку “Continue”
   5. Дождаться остановки программы
   6. Открыть ВУ, записать посчитанное значение F(X) в таблицу
   7. Посчитать ожидаемый результат функции по формуле и записать результат в таблицу
   8. Нажать кнопку “Continue”, чтобы основная программа продолжила выполняться

**Обработка прерывания ВУ-2**

* 1. Записать число для подсчета логической операции в таблицу
  2. Открыть ВУ, и ввести число, которое ранее было введено в таблицу
  3. Выставить готовность ВУ нажав кнопку “Ready”
  4. Дождаться остановки программы
  5. Из аккумулятора (AC) записать значение X в таблицу
  6. Продолжить выполнение программы, нажав кнопку “Continue”
  7. Дождаться остановки программы
  8. Из аккумулятора (AC) записать значение “Исключающее ИЛИ-НЕ” в таблицу
  9. Посчитать “Исключающее ИЛИ-НЕ” введенного в ВУ значения и значения X (если результат выходит за границы [-34, 29], то результат считать равным -34)
  10. Нажать кнопку “Continue”, чтобы основная программа продолжила выполняться

# 

# Таблица для тестирования программного комплекса

|  |  |
| --- | --- |
| **ВУ-3** |  |
| Значение X | 0000 0000 0000 0000 (0) |
| Посчитанное программой значение F(X) | 1111 1111 1111 0111(-9) |
| Посчитанное пользователем значение F(Х) | 1111 1111 1111 0111(-9) |
|  |  |
| Значение X | 0000 0000 0000 0010 (2) |
| Посчитанное программой значение F(X) | 1111 1111 1110 1111 (-17) |
| Посчитанное пользователем значение F(Х) | 1111 1111 1110 1111 (-17) |
|  |  |
| **ВУ-2** |  |
| Введенное значение | 0000 0000 0001 0001 (17) |
| Значение Х | 1111 1111 1111 0111(-9) |
| Результат, посчитанный программой | 0000 0000 0001 1001 (25) |
| Результат, посчитанный пользователем | 0000 0000 0001 1001 (25) |
|  |  |
| Введенное значение | 1111 1111 1000 1000 () |
| Значение Х | 1111 1111 1110 1111 (-17) |
| Результат, посчитанный программой | 1111 1111 1001 1000 (-104)  Но запишется в X значение -34 |
| Результат, посчитанный пользователем | 1111 1111 1001 1000 (-104) |

**Код для вставки:**

**ORG 0x0**

**V0: WORD $DEFAULT, 0x180**

**V1: WORD $DEFAULT, 0x180**

**V2: WORD $INT2, 0x180**

**V3: WORD $INT3, 0x180**

**V4: WORD $DEFAULT, 0x180**

**V5: WORD $DEFAULT, 0x180**

**V6: WORD $DEFAULT, 0x180**

**V7: WORD $DEFAULT, 0x180**

**ORG 0x023**

**START: DI**

**LD #0xA**

**OUT 5**

**LD #0xB**

**OUT 7**

**CLA**

**LD $X**

**MAIN: DI**

**LD X**

**INC**

**CALL CHECK**

**ST X**

**EI**

**JUMP MAIN**

**INT3: HLT**

**ASL**

**ASL**

**NEG**

**SUB #0x09**

**OUT 6**

**HLT**

**CALL CHECK**

**IRET**

**INT2: HLT**

**IN 4**

**SXTB**

**ST $VU\_VAL**

**NOT**

**ST $NOT\_VU\_VAL**

**LD $X**

**NOT**

**ST $NOT\_X**

**AND $NOT\_VU\_VAL**

**ST $NOT\_X\_VU**

**LD $X**

**AND $VU\_VAL**

**OR $NOT\_X\_VU**

**CALL CHECK**

**HLT**

**IRET**

**CHECK: CMP $MAX**

**BGE SETMIN**

**CMP $MIN**

**BEQ SAVE**

**BLT SETMIN**

**JUMP SAVE**

**SETMIN: LD $MIN**

**SAVE: ST $X**

**EXIT\_CHECK: RET**

**DEFAULT: PUSH**

**CLA**

**OUT 0x1**

**OUT 0x3**

**OUT 0xB**

**OUT 0xD**

**OUT 0x11**

**OUT 0x15**

**OUT 0x19**

**OUT 0x1D**

**POP**

**IRET**

**VU\_VAL: WORD 0x0**

**NOT\_VU\_VAL: WORD 0x0**

**NOT\_X: WORD 0x0**

**NOT\_X\_VU: WORD 0x0**

**X: WORD 0x0000**

**MIN: WORD 0xFFDE**

**MAX: WORD 0x001D**

**Вывод:**

Изучила организацию процесса управления программой при управляемом прерываниями вводе-выводе.

0000 0000 0000 0001

1111 1111 1111 0000

1111 1111 1000 0100

0000 0000 0000 0001

0000 0000 0111 1010