УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Основы профессиональной деятельности»

**Отчет**

По лабораторной работе №6

Вариант 5396

Студент

*Патутин В.М*

*P3114*

Преподаватель

*Николаев В. В.*

Санкт-Петербург, 2020 г.

Текст задания:

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 04916) в цикле.
2. Обработчик прерывания должен по нажатию кнопки готовности ВУ-3 осуществлять вывод результата вычисления функции F(X)=-6X+1 на данное ВУ, a по нажатию кнопки готовности ВУ-2 выполнить операцию побитового маскирования, оставив 5-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 исходной программы:**

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

DEFAULT:

IRET

ORG 0x49

X: WORD 0x0014

MIN: WORD 0xFFEB

MAX: WORD 0x0022

MAXA: WORD 0x007F

MINA: WORD 0xFF80

START:

LD #0xA

OUT 5

LD #0xB

OUT 7

OCNOV:

DI

LD X

NOP

INC

CALL CHECK

ST X

NOP

EI

JUMP OCNOV

CHECK:

CALL CHECKUP

CALL CHECKDOWN

RET

CHECKUP:

CMP MAX

BPL SAVEMIN

RET

CHECKDOWN:

CMP MIN

BMI SAVEMIN

RET

SAVEMIN:

LD MIN

RET

INT3:

PUSH

DI

LD X

NOP

ADD X

ADD X

ADD X

ADD X

ADD X

NOT

ADD #1

ADD #1

CALL CHECK2

NOP

OUT 6

EI

POP

IRET

FIVE:

ASL

ASL

ASL

ASL

ASL

ASL

ASL

ASL

ASL

ASL

ASL

CLC

ROR

ROR

ROR

ROR

ROR

ROR

ROR

ROR

ROR

ROR

ROR

RET

INT2:

PUSH

IN 4

SXTB

AND X

CALL FIVE

CALL CHECK2

ST X

NOP

POP

IRET

CHECK2:

CALL CHECKUP2

CALL CHECKDOWN2

RET

CHECKUP2:

CMP MAXA

BPL SAVEMIN

RET

CHECKDOWN2:

CMP MINA

BMI SAVEMIN

RET

SAVEMIN:

LD MIN

RET

Назначение программы:

* Основная программа должна инкрементировать содержимое X (ячейки памяти с адресом 04916) в цикле.
* Обработчик прерывания должен по нажатию кнопки готовности ВУ-3 осуществлять вывод результата вычисления функции F(X)=-6X+1 на данное ВУ, a по нажатию кнопки готовности ВУ-2 выполнить операцию побитового маскирования, оставив 5-х младших разряда содержимого РД данного ВУ и Х, результат записать в Х
* Если Х оказывается вне ОДЗ при выполнении любой операции по его изменению, то необходимо в Х записать минимальное по ОДЗ число.

Область допустимых значений:

* Число X ϵ [EB; 15]

Область представления:

* X – 16ти разрядное целое знаковое число
* Адрес X – 11ти разрядное целое беззнаковое число
* DR КВУ – 8ми разрядное целое знаковое число

Вывод:

* В ходе выполнения лабораторной работы я изучил обмен данными с ВУ – 2 и ВУ – 3 в режиме прерываний, также изучил цикл прерывания и циклы исполнения команды IRET

Методика проверки программы:

1. Загрузить комплекс программ в память базовой ЭВМ
2. Изменить значения точек останова по адресам 052,056,071,078,07D на HLT
3. Запустить основную программу в автоматическом режиме с адреса 0x50
4. Дождаться останова
5. Записать значение переменной X, находящееся в момент останова в аккумуляторе
6. Продолжить выполнение программы
7. Дождаться останова
8. Сравнить новое значение переменной X, находящееся в момент останова в аккумуляторе, с записанным ранее значением из расчета на то, что оно должно быть равно либо записанному ранее значению переменной X, увеличенному на 1, либо минимальному допустимому значению переменной X (EB).

Проверка корректной работы подпрограммы прерывания ВУ-3:

1. Загрузить программу в память БЭВМ
2. Изменить значение точки останова по адресу 071 на HLT
3. Запустить основную программу в режиме работа с адреса 0x50
4. Установить “Готовность ВУ-3”
5. Дождаться останова
6. Записать значение переменной X, находящееся в момент останова программы в аккумуляторе
7. Продолжить выполнение программы
8. Дождаться вывода результата выполнения подпрограммы прерывания на РД ВУ-3
9. Сравнить полученное на РД ВУ значение со значением функции F(X)=-6X+1 от записанного ранее значения аккумулятора c расчетом на то, что оно должно быть равно либо значением функции F(X)=-6X+1, либо минимальному допустимому значению переменной X (EB).

Проверка корректной работы подпрограммы прерывания ВУ-2:

1. Загрузить комплекс программ в память БЭВМ
2. Изменить значение точки останова по адресу 078 и 07D на HLT
3. Запустить основную программу в режиме работа с адреса 0x50
4. Ввести некоторое значение в РД ВУ-2
5. Установить “Готовность ВУ-2”
6. Дождаться останова
7. Записать значение переменной X, находящееся в момент останова программы в ячейке памяти 0x26
8. Продолжить выполнение программы
9. Дождаться останова
10. Сравнить новое значение переменной X, находящееся в момент останова программы в аккумуляторе, с записанным ранее значением c расчетом на то, что оно должно быть равно либо операции побитового маскирования, оставив 5-х младших разрядов содержимого РД данного ВУ и Х, либо минимальному допустимому значению переменной X (EB).

|  |  |  |
| --- | --- | --- |
| Прерывания от ВУ-3 | | |
| AC(x) | Результат(DR КВУ-3) | Ожидаемый результат |
| 0001 0101 | 1000 0011 | 1000 0011 |
| 0001 0110 | 1110 1011 | 1110 1011 |
| 0000 0001 | 1111 1011 | 1111 1011 |

|  |  |  |  |
| --- | --- | --- | --- |
| Прерывания от ВУ-2 | | | |
| DR КВУ-2 | AC(X) | AC(X OR DR) | Ожидаемый результат |
| 1111 0000 | 0001 0101 | 0001 0000 | 1 0000 |
| 1111 1111 | 0001 0110 | 0001 0110 | 1 0110 |
| 0000 0000 | 0000 0001 | 0000 0000 | 0 0000 |