



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное  
образовательное учреждение высшего образования  
“Национальный исследовательский университет ИТМО”

**ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ**

## **ЛАБОРАТОРНАЯ РАБОТА №6**

по дисциплине  
“ОСНОВЫ ПРОФЕССИОНАЛЬНОЙ ДЕЯТЕЛЬНОСТИ”  
Тема: «Обмен данными с ВУ по прерыванию».

Вариант: 1366.

*выполнил:*  
Студент группы Р3130  
**Птицын Максим Евгеньевич**  
*Преподаватель*  
**Ткешелашвили Нино Мерабиевна**

г. Санкт-Петербург  
2022 г.

# 1 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 (X), которое должно быть представлено как знаковое число. Область допустимых значений изменения X 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 X 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 02E<sub>16</sub>) в цикле.
2. Обработчик прерывания должен по нажатию кнопки готовности ВУ-1 осуществлять вывод результата вычисления функции  $F(X)=3X-5$  на данное ВУ, а по нажатию кнопки готовности ВУ-2 выполнить операцию побитового маскирования, оставив 4-х младших разряда содержимого РД данного ВУ и X, результат записать в X
3. Если X оказывается вне ОДЗ при выполнении любой операции по его изменению, то необходимо в X записать минимальное по ОДЗ число.

## 2 Программа

### 2.1 Assembler

```

                                ORG      0x0
V1:  WORD    $INT1, 0x180
V2:  WORD    $INT2, 0x180
V3:  WORD    $INT3, 0x180
V5:  WORD    $INT5, 0x180
V6:  WORD    $INT6, 0x180
V7:  WORD    $INT7, 0x180
V8:  WORD    $INT8, 0x180
V9:  WORD    $INT9, 0x180
                                ORG      0x2E
X:    WORD    0x0
LB:   WORD    0xFFD7
HB:   WORD    0x2C
START: DI
      LD      0x3
      OUT     0x7
      INC
      OUT     0xF
      INC
      OUT     0x13
      INC
      OUT     0x17
      INC
      OUT     0x1B
      INC
      OUT     0x1F
      LD      #0x8
      OUT     0x3
      LD      #0x9
      OUT     0x5
      EI
MAIN:  DI
      LD      $X
      INC
      CALL    $CHECK
      ST      $X
      EI
      JUMP    $MAIN
CHECK: CMP     $LB
      BPL     PART2
      JUMP    LOAD
PART2: CMP     $HB
      BMI     RETN
      BEQ     RETN
LOAD:  LD      LB
RETN:  RET
INT2:  NOP
      IN      4
      AND     #0xF
      AND     $X
      CALL    $CHECK
      ST      $X
      NOP
      IRET
INT1:  ASL
      ADD     $X
      SUB     #0x5
      OUT     2
      NOP
      IRET
INT3:  IN      0x7
      IRET
INT5:  IN      0xF
      IRET
INT6:  IN      0x13
      IRET
INT7:  IN      0x17
      IRET
INT8:  IN      0x1B
      IRET
INT9:  IN      0x1F
      IRET
```

## 2.2 Описание программы:

Программа циклически увеличивает значение ячейки памяти на 1 и обрабатывает прерывания.

## 3 Область представления данных и область допустимых значений

### 3.1 Область представления:

X, LB, HB - знаковое двухразрядное 16-ричное целое число.

### 3.2 ОДЗ

#### 3.2.1 F(x):

$$-128 \leq F(x) \leq 127$$

### 3.3 X:

$$\begin{cases} X \leq \frac{127+5}{3}, \\ X \geq \frac{-128+5}{3}, \\ -41 \leq X \leq 44 \end{cases}$$

## 4 Расположение программы в памяти БЭВМ:

Вектор прерываний - 0x000 - 0x00F .

Переменные - 0x02E-0x030 .

Программа - 0x031-0x62

## 5 Методика проверки.

Проверка обработки прерываний:

1. Загрузить текст программы в БЭВМ во вкладку Assembler.
2. Заменить NOP в тексте программы на HLT.
3. Запустить программу в режиме РАБОТА.
4. Установить «Готовность ВУ-1».
5. Дождаться остановки программы.
6. Записать текущее значение X из памяти БЭВМ:
  - (a) Запомнить текущее состояние счетчика команд.
  - (b) Ввести в клавишный регистр значение 0x02E
  - (c) Нажать «Ввод адреса».
  - (d) Нажать «Чтение».
  - (e) Записать значение регистра данных.
  - (f) Вернуть счетчик команд в исходное состояние.

7. Записать результат обработки прерывания – содержимое DR контроллера ВУ-1
8. Рассчитать ожидаемое значение обработки прерывания по формуле  $3 * X - 5$ , записать его
9. Сверить ожидаемые данные с полученными.
10. Нажать «Продолжение».
11. Ввести в ВУ-2 произвольный набор нулей и единиц, записать его
12. Установить «Готовность ВУ-2».
13. Дождаться остановки программы.
14. Записать текущее значение X из памяти БЭВМ (аналогично п.6).
15. Нажать «Продолжение».
16. Записать текущее значение X из памяти БЭВМ (аналогично п.6).
17. Рассчитать ожидаемое значение переменной X после обработки прерывания  
(Если X будет меньше минимального из ОДЗ, то запишется нижняя граница.)
18. Проверить правильность маскирования полученного числа последними  
Четырьмя битами из введённого набора на ВУ.

Проверка основной программы:

1. Загрузить текст программы в БЭВМ.
2. Записать в переменную X значение, меньшее минимального доступного по ОДЗ ( $X < -41$ )
  - (a) Записать значение IP.
  - (b) Ввести адрес 0x2E.
  - (c) Ввести желаемое значение в Input Register.
  - (d) Нажать "Запись".
  - (e) Вернуть в IR исходный IP.
  - (f) Нажать "Ввод адреса".
3. Запустить программу в режиме попрограммного выполнения.
4. Убедиться, что при первой итерации в X записывается минимальное значение по ОДЗ (FFD7)  
(аналогично п.6 из предыдущей инструкции)
5. Пройти нужное количество шагов программы, убедиться, что при увеличении X на 1, до того момента, когда он равен 45, происходит сброс значения в минимальное по ОДЗ (FFD7).

Проверка обработки прерываний с ВУ-1:

X	Предполагаемый результат	Полученный результат	Верно?
0x4	$3 * 4 - 5 = 7$	0x7	да.

Проверка обработки прерываний с ВУ-2:

Введённые данные с ВУ-2	$X_{orig}$	Предполагаемый результат	$X_{res}$	Верно?
1010 0101	0xB (1011)	1 (0001)	1(0001)	да.

Проверка работы основной программы (случай выхода из ОДЗ):

Значение икса перед выходом из ОДЗ	Какое значение должно было быть без проверки	Какой получился	Верно?
0x2C	0x2D	0xFFD7	да.
0xFFD0	0xFFD1	0xFFD7	да.