

**Министерство науки и высшего образования Российской Федерации**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
ИТМО»**

**Факультет ПИиКТ**

**Дисциплина: Основы профессиональной деятельности**

**Лабораторная работа №6  
Обмен данными с ВУ  
по прерыванию**

**Вариант 3289**

**Выполнил: Михайлов Петр Сергеевич**

**Группа: Р3111**

**Преподаватель: Остапенко Ольга Денисовна**

**Санкт-Петербург 2025г.**

## Содержание

Задание .....	3
Определение функции, вычисляемой программой .....	4
1. Текст исходный программы.....	<b>Error! Bookmark not defined.</b>
2. Описание программы.....	6
3. Расположение в БЭВМ программы, исходных данных и результатов.....	<b>Error! Bookmark not defined.</b>
4. Область представления .....	6
5. Область допустимых значений.....	6
Трассировка программы .....	<b>Error! Bookmark not defined.</b>
Заключение.....	9

## 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.

1. Основная программа должна увеличивать на 3 содержимое  $X$  (ячейки памяти с адресом  $031_{16}$ ) в цикле.
2. Обработчик прерывания должен по нажатию кнопки готовности ВУ-1 осуществлять вывод результата вычисления функции  $F(X) = -3X$  на данное ВУ, а по нажатию кнопки готовности ВУ-3 прибавить содержимое РД данного ВУ к  $X$ , 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.

# Определение функции, вычисляемой программой

## 1. Текст исходный программы / Код программы на Ассемблере

Таблица 1: Текст исходной программы.

<b>Готовность ВУ-1: <math>F(X) = -3X \rightarrow \text{РДВУ-1}</math>, Готовность ВУ-3: <math>\text{РДВУ-3} + X \rightarrow X</math>.</b> <b>Основной цикл: <math>X + 3 \rightarrow X</math>.</b>		
	ORG 0x0	; Инициализируем векторы прерываний.
V0:	WORD \$DEFAULT, 0x180	; Вектор 0 – стандартный обработчик.
V1:	WORD \$INT1, 0x180	; Вектор 1 установлен на обр. INT1.
V2:	WORD \$DEFAULT, 0x180	; Вектор 2 – стандартный обработчик.
V3:	WORD \$INT3, 0x180	; Вектор 3 установлен на обр. INT3.
V4:	WORD \$DEFAULT, 0x180	; Вектор 4 – стандартный обработчик.
V5:	WORD \$DEFAULT, 0x180	; Вектор 5 – стандартный обработчик.
V6:	WORD \$DEFAULT, 0x180	; Вектор 6 – стандартный обработчик.
V7:	WORD \$DEFAULT, 0x180	; Вектор 7 – стандартный обработчик.
DEFAULT:	IRET	; Стандартный обработчик: просто ; возвращает из прерывания.
START:	ORG 0x020 DI	; На время инициализации векторов. ; запретим какие-либо прерывания.
	LD #0x9	; Инициализируем прерывание ВУ-1.
	OUT 0x3	; на вектор 1 (и разрешим его).
	LD #0xB	; Инициализируем прерывание ВУ-3.
	OUT 0x7	; на вектор 3 (и разрешим его).
	CLA	; Остальные прерывания ВУ будут.
	OUT 0x1	; назначены на вектор 0.
	OUT 0x3	; (и вообще запрещены на уровне КВУ).
	OUT 0xB	; ....
	OUT 0xE	; ....
	OUT 0x12	; ....
	OUT 0x16	; ....
	OUT 0x1A	; ....
	OUT 0x1E	; ....
	JUMP \$PROG	; Векторы назначены. Двигаемся в ; основной цикл...
ADDR_X:	ORG 0x030 WORD 0x031	; Указатель на X.
X:	WORD ?	; Основная переменная программы.
PROG:	EI	; Разрешаем прерывания.
	LD ADDR_X	; Загрузим адрес X.
	PUSH	; И положим адрес в стек.
INCLP:	CALL \$AT_INC	; Вызываем атомарную операцию +3.
	JUMP INCLP	; Это вся суть основного цикла.

	ORG 0x040	
TEMP_ADDR:	WORD ?	; Врем. ячейка для адреса аргумента.
AT_INC:	DI	; Атомарное увел. яч. памяти на 3.
	LD &1	; Загрузим адрес операнда.
	ST TEMP_ADDR	; Сохраним во временную ячейку.
	LD (TEMP_ADDR)	; Загрузим операнд.
	NOP	; Точка отладки 1.
	INC	; Увеличим его на 3.
	INC	; Прерывания были
	INC	; запрещены до этого.
	CALL \$AAV_CHECK	; Сверим с ОДЗ полученный результат.
	ST (TEMP_ADDR)	; Сохраним его в ячейку памяти.
	NOP	; Точка отладки 2.
	EI	; Снова разрешим прерывания.
	RET	; И вернем логику управления.
INT1:	ORG 0x050	; Обработчик вектора прерываний 1.
	NOP	; Точка отладки 3.
	PUSH	; Сохраним состояние АС.
	LD \$X	; Загрузим X в АС.
	ASL	; Арифметический сдвиг влево ( $X*2$ ).
	ADD \$X	; $2*X + X \rightarrow AC$ ( $3*X$ ).
	NEG	; $AC(3*X) \rightarrow AC(-3*X)$ .
	OUT 2	; Итоговый результат $F(X)$ на ВУ-1.
	POP	; Вернем состояние АС.
	NOP	; Точка отладки 4.
	IRET	; Выйдем из текущего прерывания.
INT3:	ORG 0x060	; Обработчик вектора прерываний 3
	NOP	; Точка отладки 5.
	PUSH	; Сохраним состояние АС.
	IN 0x6	; Произведем чтение с ВУ-3.
	ADD \$X	; Прибавим X к РДВУ-3 ( $РДВУ-3 + X$ ).
	ST \$X	; Сохраним в X.
	POP	; Вернем состояние АС.
	NOP	; Точка отладки 6.
	IRET	; Выйдем из текущего прерывания.
AAV_CHECK:	ORG 0x070	
	CMP MAX_VAL	; Проверим верхнюю границу ОДЗ.
	BEQ AAV_RET	; Они равны? Да – на выход.
	BGE RESET	; Результат больше? Сбрасываем его.
	CMP MIN_VAL	; Проверим нижнюю границу ОДЗ.
	BGE AAV_RET	; Результат больше? На выход.
RESET:	LD MIN_VAL	; Иначе сбросим до минимального числа.
AAV_RET:	RET	; Тот самый выход!
MIN_VAL:	WORD 0xFFD5	; Нижняя граница ОДЗ.
MAX_VAL:	WORD 0x002A	; Верхняя граница ОДЗ.

Окончание таблицы.

## 2. Описание программы

Назначение основной программы: увеличение значения ячейки памяти на 3. Нажатие на кнопку «Готов» на ВУ-1 выведет в РДВУ-1 значение  $F(X) = -3X$ . Нажатие на кнопку «Готов» на ВУ-3 обновит значение ячейки памяти добавленным значением с РДВУ-3.

## 3. Область представления

$X$  – основная ячейка памяти.

$X$  – знаковое, 16-разрядное число.

## 4. Область допустимых значений

$$\begin{aligned}F(X) &= -3X \\-128 &\leq -3X \leq 127 \\-127 &\leq 3X \leq 128 \\-42 &\leq X \leq 42 \\42_{10} &= 0000\ 0000\ 0010\ 1010_2 = 002A_{16} \\-42_{10} &= 1111\ 1111\ 1101\ 0101_2 = FFD5_{16} \\ \text{Число } X &\in [FFD5, 002A]\end{aligned}$$

## 5. Расположение в памяти данных

- Программный комплекс располагается в следующих ячейках памяти:
  - Векторы прерываний: между ячейками 000 и 00F включительно
  - Стандартный обработчик прерываний: между ячейками 010 и 01F включительно
  - Инициализация векторов прерываний: между ячейками 020 и 02E включительно
  - Основной цикл программы: между ячейками 030 и 036 включительно
  - Подпрограмма атомарного увеличения ячейки на 3: между ячейками 040 и 04D включительно
  - Обработчик прерываний вектора 1: между ячейками 050 и 059 включительно
  - Обработчик прерываний вектора 3: между ячейками 060 и 067 включительно
  - Подпрограмма проверки на входение ячейки в ОДЗ: между ячейками 070 и 078 включительно
- Исходные данные должны располагаться в ячейках памяти:
  - $X - 031$
- Результат работы программы должен располагаться в ячейке памяти 020.
- В программе используются следующие неизменяемые значения (константы):
  - По адресу 000 – значение 0x0010
  - По адресу 001 – значение 0x0180
  - По адресу 002 – значение 0x0010
  - По адресу 003 – значение 0x0180
  - По адресу 004 – значение 0x0050
  - По адресу 005 – значение 0x0180
  - По адресу 006 – значение 0x0060
  - По адресу 007 – значение 0x0180
  - По адресу 008 – значение 0x0010
  - По адресу 009 – значение 0x0180
  - По адресу 00A – значение 0x0010

- По адресу 00B – значение 0x0180
- По адресу 00C – значение 0x0010
- По адресу 00D – значение 0x0180
- По адресу 00E – значение 0x0010
- По адресу 00F – значение 0x0180
- По адресу 077 – значение 0xFFD5
- По адресу 078 – значение 0x002A
- В программе также используются вспомогательные ячейки, находящиеся по адресам 030 и 040.
- Первая команда располагается в ячейке по адресу 020.

# Методики проверки

## 1. Проверка основного цикла

1. Загрузить комплекс программ в память Базовой ЭВМ.
2. Изменить значение отладочной точки 1 и отладочной точки 2 по адресам 047 и 053 на НЛТ.
3. Переключить тумблер в режим «РАБОТА», отключить потактовое исполнение, нажать кнопку «ПУСК».
4. Дождаться остановки работы ЭВМ.
5. Записать текущее значение счетчика команд (IP).
6. Ввести в клавишный регистр (IR) значение 0000.0000.0011.0001 (0x0031).
7. Нажать кнопку «ВВОД АДРЕСА».
8. Нажать кнопку «ЧТЕНИЕ».
9. Записать значение регистра данных (DR), если это сделано второй раз, то перейти к шагу 15.
10. Ввести в клавишный регистр (IR) ранее записанное значение счётчика команд (IP).
11. Нажать кнопку «ВВОД АДРЕСА».
12. Не меняя состояние тумблеров, нажать кнопку «ПРОДОЛЖЕНИЕ».
13. Дождаться остановки работы ЭВМ.
14. Повторить пункты 5–9 включительно.
15. Сравнить полученные 2 записанных значения.
  - а. Второе значение либо должно быть больше первого на 3.
  - б. Либо равняться минимальному значению согласно ОДЗ исходных данных, в случае если первое значение было больше, чем 0x27.

## 2. Проверка прерывания ВУ-1

1. Загрузить комплекс программ в память Базовой ЭВМ.
2. Изменить значение отладочной точки 3 и отладочной точки 4 по адресам 058 и 066 на НЛТ.
3. Переключить тумблер в режим «РАБОТА», отключить потактовое исполнение, нажать кнопку «ПУСК».
4. Установить «Готовность ВУ-1»
5. Дождаться остановки работы ЭВМ.
6. Записать текущее значение счетчика команд (IP).
7. Ввести в клавишный регистр (IR) значение 0000.0000.0011.0001 (0x0031).
8. Нажать кнопку «ВВОД АДРЕСА».
9. Нажать кнопку «ЧТЕНИЕ».
10. Записать значение регистра данных (DR).
11. Ввести в клавишный регистр (IR) ранее записанное значение счётчика команд (IP).
12. Нажать кнопку «ВВОД АДРЕСА».
13. Не меняя состояние тумблеров, нажать кнопку «ПРОДОЛЖИТЬ».
14. Дождаться остановки работы ЭВМ.
15. Вычислить функцию  $F(X) = -3X$  от заданного значения  $X$  в начале программы.
16. Сравнить его с записанным значением РДВУ-1. Убедиться, что значения равны.

## 3. Проверка прерывания ВУ-3

1. Загрузить комплекс программ в память Базовой ЭВМ.



2. Изменить значение отладочной точки 5 и отладочной точки 6 по адресам 070 и 076 на HLT.
3. Переключить тумблер в режим «РАБОТА», отключить потактовое исполнение, нажать кнопку «ПУСК».
4. Придумать любое число и записать его в РДВУ-3.
5. Установить «Готовность ВУ-3».
6. Дождаться остановки работы ЭВМ.
7. Записать текущее значение счетчика команд (IP).
8. Ввести в клавишный регистр (IR) значение 0000.0000.0011.0001 (0x0031).
9. Нажать кнопку «ВВОД АДРЕСА».
10. Нажать кнопку «ЧТЕНИЕ».
11. Записать значение регистра данных (DR), если это сделано второй раз, то перейти к шагу 17.
12. Ввести в клавишный регистр (IR) ранее записанное значение счётчика команд (IP).
13. Нажать кнопку «ВВОД АДРЕСА».
14. Не меняя состояние тумблеров, нажать кнопку «ПРОДОЛЖЕНИЕ».
15. Дождаться остановки работы ЭВМ.
16. Повторить пункты 7–11 включительно.
17. Сравнить полученные 2 записанных значения: второе значение должно быть получено из суммы первого и значения из пункта 4.

#### 4. Сводная таблица результатов проверок по соответствующим методикам

Таблица 2: Результаты проверки работы программного комплекса.

№	Основной цикл			Прерывание ВУ-1			Прерывание ВУ-3		
	Исходное	Подсчитанное	Полученное	Исходное	Подсчитанное	Полученное	Исходное	Подсчитанное	Полученное
1	DR: 0x0000	0x0003	0x0003	DR: 0xFFFF	F(X)=-3X	F(X)=-3X	КВУ-3: 0x0088	РДВУ-3 + DR	РДВУ-3 + DR
1				РДВУ-1	0x0003	0x0003	DR: 0x0001	0x0089	0x0089
2	DR: 0x0027	0x002A	0x002A	DR: 0x0001	F(X)=-3X	F(X)=-3X	КВУ-3: 0x00FF	РДВУ-3 + DR	РДВУ-3 + DR
2				РДВУ-1	0x00FD	0x00FD	DR: 0x0100	0x01FF	0x01FF
3	DR: 0x0030	0xFFD5	0xFFD5	DR: 0x5555	F(X)=-3X	F(X)=-3X	КВУ-3: 0x0010	РДВУ-3 + DR	РДВУ-3 + DR
3				РДВУ-1	0x0001	0x0001	DR: 0xFFFF	0x0001	0x0001

Окончание таблицы.

## Заключение

Во время выполнения данной лабораторной работы я изучил процесс прерывания программы и исследовал функционирование Базовой ЭВМ при обмене данными в режиме прерывания программы, а также научился писать методики проверки программных комплексов и следовать им.