

Федеральное государственное автономное образовательное  
учреждение высшего образования «Национальный исследовательский  
университет ИТМО»

**Факультет программной инженерии и  
компьютерной техники**

Лабораторная работа №2  
по дисциплине  
«Основы профессиональной деятельности»  
Вариант 3499

Выполнил:  
Жгилев Иван Игоревич  
Студент группы Р3130  
Преподаватель:  
Зайцева Ирина Сергеевна

Санкт-Петербург  
2025г

# Содержание

<b>СОДЕРЖАНИЕ.....</b>	<b>2</b>
<b>ТЕМА И ЦЕЛЬ РАБОТЫ.....</b>	<b>3</b>
<b>ЗАДАНИЕ.....</b>	<b>3</b>
<b>ВЫПОЛНЕНИЕ ЗАДАНИЯ .....</b>	<b>4</b>
Текст исходной программы .....	4
Описание программы .....	5
ОП и ОДЗ исходных данных и результата .....	5
Расположение в памяти ЭВМ программы, исходных данных и результатов; адреса первой и последней выполняемой команд программы .....	7
<b>ТРАССИРОВКА ПРОГРАММЫ.....</b>	<b>7</b>
<b>ВАРИАНТ ПРОГРАММЫ С МЕНЬШИМ ЧИСЛОМ КОМАНД.....</b>	<b>8</b>
<b>ЗАКЛЮЧЕНИЕ .....</b>	<b>9</b>

# Тема и цель работы

Тема: «Исследование работы БЭВМ»

Цель работы: изучение приемов работы на базовой ЭВМ и исследование порядка выполнения арифметических команд и команд пересылки.

## Задание

Задание представлено на Рис 1.

### Лабораторная работа №2

По выданному преподавателем варианту определить функцию, вычисляемую программой, область представления и область допустимых значений исходных данных и результата, выполнить трассировку программы, предложить вариант с меньшим числом команд. При выполнении работы представлять результат и все операнды арифметических операций знаковыми числами, а логических операций набором из шестнадцати логических значений.

**Ход работы, содержание отчета и контрольные вопросы описаны в методических указаниях**

Введите номер варианта

065: 2070  
066: + 0200  
067: 0280  
068: 2070  
069: 2065  
06A: E06F  
06B: A071  
06C: 406F  
06D: E072  
06E: 0100  
06F: 2070  
070: 0280  
071: 2070  
072: A071

### Лабораторная работа №3

По выданному преподавателем варианту восстановить текст заданного варианта программы, определить предназначение и составить описание программы, определить область представления и область допустимых значений исходных данных и результата, выполнить трассировку программы.

**Ход работы, содержание отчета и контрольные вопросы описаны в методических указаниях**

Введите номер варианта

25 ноября 2025 г.

В 15:29 (Местное время)

Рис 1 – задание

## Выполнение задания

### Текст исходной программы

Адрес	Код команды	Мнемоника	Комментарии
065	2070	WORD 2070	Переменная А исходные данные
066	0200	CLA	Очистить содержимое аккумулятора
067	0280	NOT	Побитовое отрицание аккумулятора
068	2070	AND 070 (\$C)	Выполнить операцию поразрядного И между ячейкой памяти (070) и аккумулятором
069	2065	AND 065 (\$A)	Выполнить операцию поразрядного И между ячейкой памяти (065) и аккумулятором
06A	E06F	ST 06F (\$B)	Записать содержимое аккумулятора в ячейку (06F)
06B	A071	LD 071 (\$D)	Записать содержимое ячейки (071) в аккумулятор
06C	406F	ADD 06F (\$B)	Записать сумму текущего значения аккумулятора и значения ячейки (06F) в аккумулятор
06D	E072	ST 072 (\$R)	Записать содержимое аккумулятора в ячейку (072 )
06E	0100	HLT	Остановка программы
06F	2070	WORD 2070	Переменная В промежуточное значение
070	0280	WORD 0280	Переменная С исходные данные
071	2070	WORD 2070	Переменная D исходные данные
072	A071	WORD A071	Переменная R результат

## Описание программы

Назначение программы и реализуемая ей функция (формула):

A = значение в ячейке памяти 065

B = значение в ячейке памяти 06F

C = значение в ячейке памяти 070

D = значение в ячейке памяти 071

R = значение в ячейке памяти 072

AC = 0

AC = 1111 1111 1111 1111

AC = C & AC

AC = A & AC

B = AC

AC = D

AC = B + AC

R = AC

$R = (A \& C) + D$

## ОП и ОДЗ исходных данных и результата

**Область представления:**

R – знаковое, 16-ти разрядное число

A, C – набор из 16 логических однобитовых значений

D – знаковое, 16-ти разрядное число

(A & C) – знаковое, 16-ти разрядное число

Для логических операций область представления: [0; 65535]

Для арифметических операций: [-32768; 32767]

**Область допустимых значений:**

$$\left\{ \begin{array}{l} D \in [-2^{14}; 2^{14} - 1] \\ (A \& C) \in [-2^{14}; 2^{14} - 1] \\ A_{15}; C_{15} = 0 \\ \forall i \in \{0, 1, 2, \dots, 14\} A_i; C_i \in \{0, 1\} \end{array} \right.$$

$$\left\{ \begin{array}{l} D \in [2^{14}; 2^{15} - 1] \\ \left\{ \begin{array}{l} A_{15}; C_{15} = 1 \\ \forall i \in \{0, 1, 2, \dots, 14\} A_i; C_i \in \{0, 1\} \end{array} \right. \\ \left\{ \begin{array}{l} A_{15} = 0; C_{15} = 0 \\ A_{15} = 1; C_{15} = 0 \\ A_{15} = 0; C_{15} = 1 \end{array} \right. \\ (A \& C) \leq 2^{15} - 1 - D \end{array} \right.$$

$$\left\{ \begin{array}{l} D \in [-2^{15}; -2^{14} - 1] \\ \left\{ \begin{array}{l} A_{15} = 0; C_{15} = 0 \\ A_{15} = 1; C_{15} = 0 \\ A_{15} = 0; C_{15} = 1 \end{array} \right. \\ \forall i \in \{0, 1, 2, \dots, 14\} A_i; C_i \in \{0, 1\} \\ \left\{ \begin{array}{l} A_{15}; C_{15} = 1 \\ (A \& C) \geq -2^{15} - D \end{array} \right. \end{array} \right.$$

Совокупность всех систем представлена на Рис 2

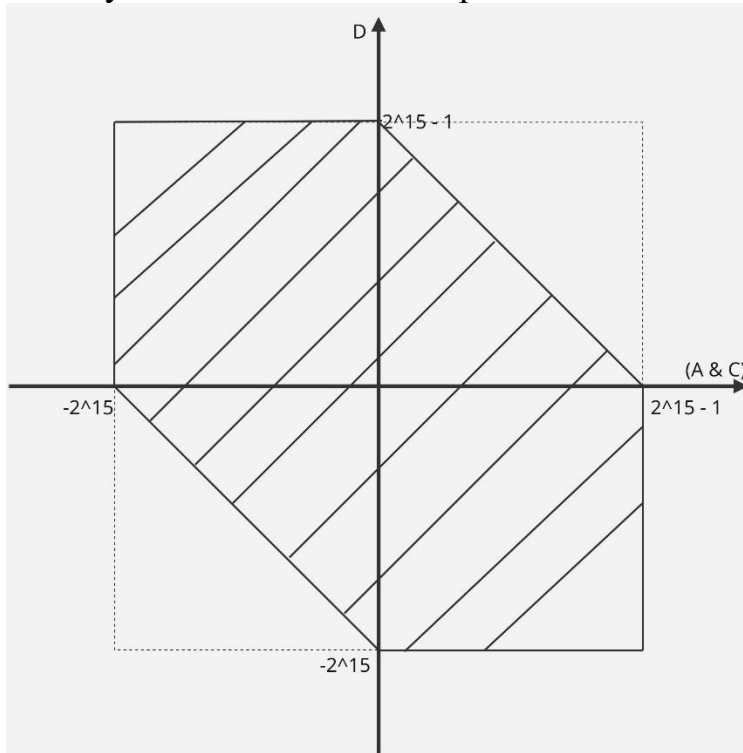


Рис 2 - область допустимых значений

## Расположение в памяти ЭВМ программы, исходных данных и результатов; адреса первой и последней выполняемой команд программы

Вся программа занимает 065 – 072 адреса.

Исходный код программы занимает 066 – 06E адреса.

Переменные занимают 065 и 06F – 072 адреса.

Промежуточный результат хранится в ячейке 06F.

Итоговый результат хранится в ячейке 072.

Адрес первой команды выполняемой программой: 066

Адрес последней команды выполняемой программой: 06E

## Трассировка программы

Выполняемая команда		Содержимое регистров процессора после выполнения команды								Ячейка, содержимое которой изменилось после выполнения команды	
Адрес	Код	IP	CR	AR	DR	SP	BR	AC	NZVC	Адрес	Новый код
066	0200	067	0200	066	0200	000	0066	0000	0100		
067	0280	068	0280	067	0280	000	0067	FFFF	1000		
068	2070	069	2070	070	0280	000	0068	0280	0000		
069	2065	06A	2065	065	2070	000	0069	0000	0100		
06A	E06F	06B	E06F	06F	0000	000	006A	0000	0100	06F	0000
06B	A071	06C	A071	071	2070	000	006B	2070	0000		
06C	406F	06D	406F	06F	0000	000	006C	2070	0000		
06D	E072	06E	E072	072	2070	000	006D	2070	0000	072	2070
06E	0100	06F	0100	06E	0100	000	006E	2070	0000		

## Вариант программы с меньшим числом команд

066	A070	LD 070 (\$C)	Записать содержимое ячейки (070) в аккумулятор
067	2065	AND 065 (\$A)	Выполнить операцию поразрядного И между ячейкой памяти (065) и аккумулятором
068	406F	ADD 071 (\$D)	Записать сумму текущего значения аккумулятора и значения ячейки (06F) в аккумулятор
069	E072	ST 072 (\$R)	Записать содержимое аккумулятора в ячейку (072 )
06A	0100	HLT	Остановка программы

$$R = (A \& C) + D$$

## Трассировка с новыми значениями

$$A = CCAA_{16}$$

$$C = 3FDA_{16}$$

$$D = 1288_{16}$$

Выполняемая команда		Содержимое регистров процессора после выполнения команды								Ячейка, содержимое которой изменилось после выполнения команды	
Адрес	Код	IP	CR	AR	DR	SP	BR	AC	NZVC	Адрес	Новый код
066	0200	067	0200	066	0200	000	0066	0000	0100		
067	0280	068	0280	067	0280	000	0067	FFFF	1000		
068	2070	069	2070	070	3FDA	000	0068	3FDA	0000		
069	2065	06A	2065	065	CCAA	000	0069	0C8A	0000		
06A	E06F	06B	E06F	06F	0C8A	000	006A	0C8A	0000	06F	0C8A
06B	A071	06C	A071	071	1288	000	006B	1288	0000		
06C	406F	06D	406F	06F	0C8A	000	006C	1F12	0000		
06D	E072	06E	E072	072	1F12	000	006D	1F12	0000	072	1F12
06E	0100	06F	0100	06E	0100	000	006E	1F12	0000		

Проверка трассировки вручную

$$A = (1100\ 1100\ 1010\ 1010)_2$$

$$C = (0011\ 1111\ 1101\ 1010)_2$$

$$D = (0001\ 0010\ 1000\ 1000)_2$$

$$(A \& C) = (0000\ 1100\ 1000\ 1010)_2$$

$$(A \& C) + D = (0001\ 1111\ 0001\ 0010)_2$$

Результат, полученный во время выполнения программы, совпадает с результатом, полученным при ручной проверке, значит значения заданы в области допустимых значений



## **Заключение**

Во время выполнения лабораторной работы я научился работать с некоторыми командами БЭВМ с абсолютной адресацией, определять функцию, реализуемую программой. Изучил, как определять область представления и область допустимых значений исходных данных и результата. Я научился составлять таблицу трассировки и переписывать исходный код с меньшим количеством команд.