«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Факультет программной инженерии и компьютерной техники  
Направление подготовки 09.03.04 Программная инженерия  
Дисциплина «Основы профессиональной деятельности»

Отчет

По лабораторной работе №3

Вариант 1701

Студент:

*Бондаренко А.А.*

*Р3110*

Преподаватель:

*Перцев Т. С.*

Санкт-Петербург, 2023 г.

**Оглавление**

[**Задание**: 3](#_Toc150813180)

[**Основные этапы выполнения** 4](#_Toc150813181)

[**Таблица команд:** 4](#_Toc150813182)

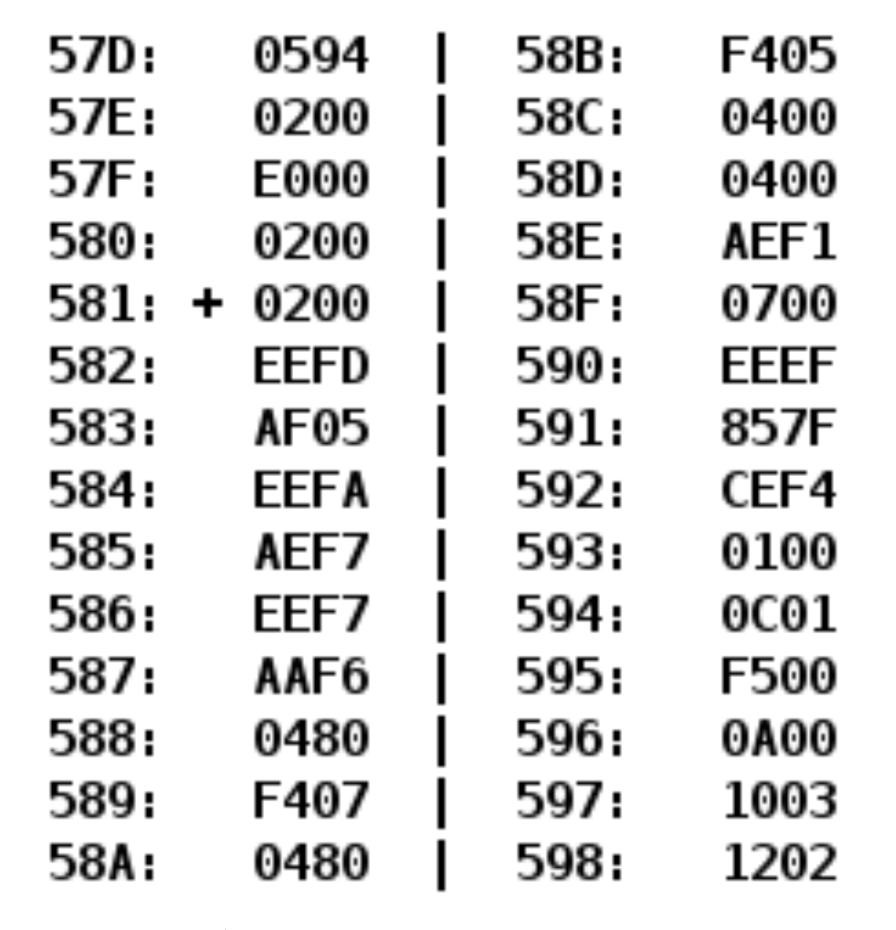
[**Описание программы:** 4](#_Toc150813183)

[**Область определения:** 4](#_Toc150813184)

[**Выводы**: 4](#_Toc150813185)

# **Задание**:

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.



# **Основные этапы выполнения**

## **Таблица команд:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Адрес** | **Команда** | **Мнемоника** | **Комментарий** | **Описание программы** |
| **57D** | **0594** | **A** | **Значение** |  |
| **57E** | **0200** | **B** | **Значение** |  |
| **57F** | **E000** | **C** | **Значение** |  |
| **580** | **0200** | **D** | **Значение** |  |
| **581** | **0200** | **CLA** | **AC=0** | **Очистка аккумулятора, старт программы** |
| **582** | **EEFD** |  |  |  |
| **583** | **AF05** |  |  |  |
| **584** | **EEFA** |  |  |  |
| **585** | **AEF7** |  |  |  |
| **586** | **EEF7** |  |  |  |
| **587** | **AAF6** |  |  |  |
| **588** | **0480** |  |  |  |
| **589** | **F407** |  |  |  |
| **58A** | **0480** |  |  |  |
| **58B** | **0405** |  |  |  |
| **58C** | **0400** |  |  |  |
| **58D** | **0400** |  |  |  |
| **58E** | **AEF1** |  |  |  |
| **59F** | **0700** |  |  |  |
| **590** | **EEEF** |  |  |  |
| **591** | **857F** |  |  |  |
| **592** | **CEF4** |  |  |  |
| **593** | **0100** |  |  |  |
| **594** | **0C01** |  |  |  |
| **595** | **F500** |  |  |  |
| **596** | **0A00** |  |  |  |
| **597** | **1003** |  |  |  |
| **598** | **1202** |  |  |  |

## **Описание программы:**

Программа получает на вход массив из некоторых чисел, количество которых заранее записано в ячейку С, и подсчитывает количество чисел из этого массива, кратных 4. Результат подсчета записывается в ячейку

## **Область определения:**

С, (A & B) ограничиваются следующими условиями:

1. если знаки совпадают (C отрицательно, старший бит (A & B) – 1, и наоборот), значения любые (вмещающиеся в 16 бит)
2. если знаки не совпадают, оба числа входят в промежуток [-214; 214 - 1]

Правила для A, B:

1. Если старший бит обоих – единица, а C отрицательно, ограничено 16 битами
2. Если старший бит любого – ноль, а C положительно, ограничено 16 битами
3. В иных случаях первые два бита должны совпадать

Пояснение:

В первом случае модуль результата вычитания будет меньше, чем значение большего по модулю входного значения => переполнение не возникает

Во втором случае область определения C, (A | B) не может совпадать с областью допустимых значений, т.к. при выборе крайних значений, образуется ещё один разряд, который вылезет за пределы допустимого значения, т.е. [-215; 215 – 1] → [-216 + 1; 216] => сокращаем область определения до 15 бит (14 двоичных разрядов + знак).

Для A, B – в первом и втором случаях их знак совпадает со знаком C, переполнение не возникает

В третьем случае A и B могут быть любым набором из 16-ти нулей и единиц, которые в результате побитового «ИЛИ» не выходят из диапазона

[0000 0000 0000 00002; 0011 1111 1111 11112] [1100 0000 0000 00002; 1111 1111 1111 11112],

чтобы в результате проведения операции над этими данными, числовое значение, представленное в двоичном формате со знаком, не вышло за свою область определения.

## 

## Область допустимых значений

R – знаковое целое 16-разрядное число [-215; 215 - 1]

Пояснение:

R записывается 16-ю однобитными значениями, один бит отводится на знак, значит, 15 бит отводится на само число.

# **Таблица трассировки:**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Выполняемая команда** | | **Содержание регистров в процессоре после выполнения команды** | | | | | | | | **Ячейка, содержимое которой изменилось после выполнения команды** | |
| Адрес | Код | IP | CR | AR | DR | SP | BR | AC | NZVC | Адрес | Новый код |
| 103 | 3104 |  |  |  |  |  |  |  |  |  |  |
| 104 | 3105 |  |  |  |  |  |  |  |  |  |  |
| 105 | A10E | 106 | A10E | 10E | 6103 | 000 | 0105 | 6105 | 0000 |  |  |
| 106 | 0200 | 107 | 0200 | 106 | 0200 | 000 | 0106 | 0000 | 0100 |  |  |
| 107 | 3104 | 108 | 3104 | 104 | 3105 | 000 | CEFA | 3105 | 0000 |  |  |
| 108 | 3105 | 109 | 3105 | 105 | A10E | 000 | 4EF0 | B10F | 1000 |  |  |
| 109 | E103 | 10A | E103 | 103 | B10F | 000 | 0109 | B10F | 1000 | 103 | B10F |
| 10A | A10E | 10B | A10E | 10E | 6103 | 000 | 010A | 6103 | 0000 |  |  |
| 10B | 6103 | 10C | 6103 | 103 | B10F | 000 | 010B | AFF4 | 1010 |  |  |
| 10C | E10F | 10D | E10F | 10F | AFF4 | 000 | 010C | AFF4 | 1010 | 10F | AFF4 |
| 10D | 0100 | 10E | 0100 | 10D | 0100 | 000 | 010D | AFF4 | 1010 |  |  |
| 10E | E107 |  |  |  |  |  |  |  |  |  |  |
| 10F | A113 |  |  |  |  |  |  |  |  |  |  |

# **Укороченная программа**

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код команды | Мнемоника | Комментарии |
| 104 | 3105 | - | Переменная A |
| 105 | A10E | - | Переменная B |
| 106 | 4105 | ADD 105 | Добавляем в аккумулятор значение ячейки 105(B) |
| 107 | 3104 | OR 104 | Проводим побитовую операцию «ИЛИ» между значением ячейки 104(A) и аккумулятором |
| 108 | B10C | SWAM 10C | Сохранить значение аккумулятора в ячейку 10C, а содержимое ячейки 10C поместить в аккумулятор. |
| 109 | 610C | SUB 10C | Вычитает из аккумулятора значение ячейки 10C |
| 10A | E10D | ST 10D | Сохранить значение аккумулятора в ячейку 10D(R) |
| 10B | 0100 | HLT | Останов. Останавливает выоплнение программы |
| 10C | 6103 | - | Переменная С |
| 10D | E10F | - | Переменная R |

# Дополнительное задание:

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код команды | Мнемоника | Комментарии |
| 103 | 3104 | - | Переменная T (промежуточный результат) |
| 104 | 90EA | - | Переменная A |
| 105 | 8089 | - | Переменная B |
| \*106 | 0200 | CLA | Очистка аккумулятора |
| 107 | 3104 | OR 104 | Проводим побитовую операцию «ИЛИ» между значением ячейки 104 и аккумулятором. |
| 108 | 3105 | OR 105 | Проводим побитовую операцию «ИЛИ» между значением ячейки 105 и аккумулятором. |
| 109 | E103 | ST 103 | Сохранить значение аккумулятора в ячейку 103 (T) |
| 10A | A10E | LD 10E | Добавляет значение ячейки 10Е(С) в аккумулятор |
| 10B | 6103 | SUB 103 | Вычитает из аккумулятора значение ячейки 103(T) |
| 10C | E10F | ST 10F | Сохранить значение аккумулятора в ячейку 10F (R) |
| 10D | 0100 | HLT | Останов. Останавливает выоплнение программы |
| 10E | 7FFF | - | Переменная С |
| 10F | E10F | - | Переменная R |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Выполняемая команда** | | **Содержание регистров в процессоре после выполнения команды** | | | | | | | | **Ячейка, содержимое которой изменилось после выполнения команды** | |
| Адрес | Код | IP | CR | AR | DR | SP | BR | AC | NZVC | Адрес | Новый код |
| 106 | 0200 | 107 | 0200 | 106 | 0200 | 000 | 0106 | 0000 | 0100 |  |  |
| 107 | 3104 | 108 | 3104 | 104 | 90EA | 000 | 6F15 | 90EA | 1000 |  |  |
| 108 | 3105 | 109 | 3105 | 105 | 8089 | 000 | 6F14 | 90EB | 1000 |  |  |
| 109 | E103 | 10A | E103 | 103 | 90EB | 000 | 0109 | 90EB | 1000 | 103 | 90EB |
| 10A | A10E | 10B | A10E | 10E | 7FFF | 000 | 010A | 7FFF | 0000 |  |  |
| 10B | 6103 | 10C | 6103 | 103 | 90EB | 000 | 010B | EF14 | 1010 |  |  |
| 10C | E10F | 10D | E10F | 10F | EF14 | 000 | 010C | EF14 | 1010 | 10F | EF14 |
| 10D | 0100 | 10E | 0100 | 10D | 0100 | 000 | 010D | EF14 | 1010 |  |  |

Результат в Одз не попадает из-за возникающего переполнения.

**Выводы**:

В ходе работы над лабораторной работой я познакомился со структурой БЭВМ, узнал, как устроены и связаны его основные элементы, научился определять ОДЗ, узнал структуру и виды команд.