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

Факультет Программной Инженерии и Компьютерной Техники

**Лабораторная работа №4**

По дисциплине

«Основы профессиональной деятельности»

Вариант 14289

Выполнил:

Петров Вячеслав Маркович

Группа P3108

Принял:

Вербовой Александр Александрович

Санкт-Петербург 2024

Оглавление

[Текст задания 3](#_Toc159759035)

[Описание программы 5](#_Toc159759036)

[Таблица трассировки 7](#_Toc159759037)

[Вывод 7](#_Toc159759038)

# Текст задания

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.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

|  |  |  |  |
| --- | --- | --- | --- |
| **Адрес** | **Код команды** | **Мнемоника** | **Комментарии** |
| 1E4 | 0200 | CLA | Очистка аккумулятора |
| 1E5 | EE19 | ST IP+25 | R = 0 |
| 1E6 | AE17 | LD IP+23 | Загрузка в аккумулятор |
| 1E7 | 0700 | INC | AC = Y + 1 |
| 1E8 | 0C00 | PUSH | Вызов функции |
| 1E9 | D716 | CALL 716 | F(Y + 1) |
| 1EA | 0800 | POP | Загрузка результата в аккумулятор |
| 1EB | 0700 | INC | AC = F(Y + 1) + 1 |
| 1EC | 6E12 | SUB IP+18 | AC = F(Y + 1) + 1 – R = F(Y + 1) + 1 |
| 1ED | EE11 | ST IP+17 | R = F(Y + 1) + 1 |
| 1EE | AE0E | LD IP+14 | Загрузка в аккумулятор |
| 1EF | 0740 | DEC | AC = X - 1 |
| 1F0 | 0C00 | PUSH | Вызов функции |
| 1F1 | D716 | CALL 716 | F(X - 1) |
| 1F2 | 0800 | POP | Загрузка результата в аккумулятор |
| 1F3 | 4E0B | SUB IP+11 | AC = F(X - 1) – R = F(X - 1) – (F(Y + 1) + 1) |
| 1F4 | EE0A | ST IP+10 | R = F(X - 1) – F(Y + 1) - 1 |
| 1F5 | AE06 | LD IP+6 | Загрузка в аккумулятор |
| 1F6 | 0C00 | PUSH | Вызов функции |
| 1F7 | D716 | CALL 716 | F(Z) |
| 1F8 | 0800 | POP | Загрузка результата в аккумулятор |
| 1F9 | 4E05 | ADD IP+5 | AC = F(Z) + F(X - 1) – F(Y + 1) - 1 |
| 1FA | EE04 | ST IP+4 | R = F(Z) + F(X - 1) – F(Y + 1) - 1 |
| 1FB | 0100 | HLT | Остановка программы |
| 1FC | ZZZZ | Z | Значение Z |
| 1FD | YYYY | Y | Значение Y |
| 1FE | XXXX | X | Значение X |
| 1FF | FF9E | R | Результат |

Подпрограмма:

|  |  |  |  |
| --- | --- | --- | --- |
| **Адрес** | **Код команды** | **Мнемоника** | **Комментарии** |
| 716 | AC01 | LD (SP+1) | Загрузка аргумента |
| 717 | F204 | BMI 4 | Если ≤ 0, то переход на 71C |
| 718 | F003 | BEQ 3 |
| 719 | 7E09 | CMP IP+9 | Если AC – Q ≤ 0, то переход на 720 |
| 71A | F005 | BEQ 5 |
| 71B | F804 | BLT 4 |
| 71C | 4C01 | ADD (SP+1) | Сложение с аргументом дважды |
| 71D | 4C01 | ADD (SP+1) |
| 71E | 6E05 | SUB IP+5 | Вычитание W |
| 71F | CE01 | JUMP IP+1 | Переход на 721 |
| 720 | AE02 | LD IP+2 | Загрузка Q |
| 721 | EC01 | ST (SP+1) | Сохранение результата |
| 722 | 0A00 | RET | Возврат |
| 723 | 0040 | 0040 | Константа Q = 64 |
| 724 | 0050 | 0050 | Константа W = 80 |

# Описание программы

Назначение программы: нахождения значения функции:

График:

Изображение выглядит как линия, диаграмма, График, Параллельный

Автоматически созданное описание

**Область представления**

* X, Y, Z, Q, W, R – целые знаковые шестнадцатеричные числа в дополнительном коде.

**Расположение данных в памяти**

**Основная программа:**

* 1E4-1FB – команды;
* 1FC–1FE – исходные данные;
* 1FF – итоговый результат.

**Подпрограмма:**

* 716-722 – команды;
* 723, 724 – константы.

**Область допустимых значений**

Q = 004016 = 64

W = 005016 = 80

Будем считать, что Q и W можно поменять.

При значении аргумента функции в промежутке [-215; 0] и (Q, 215 - 1], функция вернет значение 3x - 80. Функция монотонно возрастает, поэтому:

Значит возможно возникновение переполнения, значит требуется ограничить значение операнда таким образом, чтобы в результате значения функции не выходило переполнение. Поскольку аргумент умножается на 3, то надо (-32768 + W) и 32767 поделить на 3

Тогда подставляя W получим, что:

* при значении аргумента функции в промежутке [(-32768 + W)/3; 0] результат будет (в моём случае: [-10896; 0])
* при значении аргумента функции в промежутке (0; Q] результат будет
* при значении аргумента функции в промежутке (Q; 10922] результат будет

Также позже идёт сложение трёх результатов таких программ, там тоже может возникнуть переполнение, поэтому следует крайние значения ещё делить на 3.

В общем виде:

В моём случае:

В принципе можно строго ограничить

# Таблица трассировки

x=0731, y=0000, z=f8c0, q=0313, w=0090

(выдали новые числа)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Выполняемая  команда | | Содержимое регистров после выполнения команды | | | | | | | | | Ячейка, содержимое которой изменилось после выполнения команды | |
| Адрес | Код  Коман-ды | IP | CR | AR | DR | SP | BR | AC | PS | NZVC | Адрес | Новый  код |
| 1E4 | 0200 | 1E4 | 0000 | 000 | 0000 | 000 | 0000 | 0000 | 004 | 0100 |  |  |
| 1E4 | 0200 | 1E5 | 0200 | 1E4 | 0200 | 000 | 01E4 | 0000 | 004 | 0100 |  |  |
| 1E5 | EE19 | 1E6 | EE19 | 1FF | 0000 | 000 | 0019 | 0000 | 004 | 0100 | 1FF | 0000 |
| 1E6 | AE17 | 1E7 | AE17 | 1FE | 0731 | 000 | 0017 | 0731 | 000 | 0000 |  |  |
| 1E7 | 0700 | 1E8 | 0700 | 1E7 | 0700 | 000 | 01E7 | 0732 | 000 | 0000 |  |  |
| 1E8 | 0C00 | 1E9 | 0C00 | 7FF | 0732 | 7FF | 01E8 | 0732 | 000 | 0000 | 7FF | 0732 |
| 1E9 | D716 | 716 | D716 | 7FE | 01EA | 7FE | D716 | 0732 | 000 | 0000 | 7FE | 01EA |
| 716 | AC01 | 717 | AC01 | 7FF | 0732 | 7FE | 0001 | 0732 | 000 | 0000 |  |  |
| 717 | F204 | 718 | F204 | 717 | F204 | 7FE | 0717 | 0732 | 000 | 0000 |  |  |
| 718 | F003 | 719 | F003 | 718 | F003 | 7FE | 0718 | 0732 | 000 | 0000 |  |  |
| 719 | 7E09 | 71A | 7E09 | 723 | 0313 | 7FE | 0009 | 0732 | 001 | 0001 |  |  |
| 71A | F005 | 71B | F005 | 71A | F005 | 7FE | 071A | 0732 | 001 | 0001 |  |  |
| 71B | F804 | 71C | F804 | 71B | F804 | 7FE | 071B | 0732 | 001 | 0001 |  |  |
| 71C | 4C01 | 71D | 4C01 | 7FF | 0732 | 7FE | 0001 | 0E64 | 000 | 0000 |  |  |
| 71D | 4C01 | 71E | 4C01 | 7FF | 0732 | 7FE | 0001 | 1596 | 000 | 0000 |  |  |
| 71E | 6E05 | 71F | 6E05 | 724 | 0090 | 7FE | 0005 | 1506 | 001 | 0001 |  |  |
| 71F | CE01 | 721 | CE01 | 71F | 0721 | 7FE | 0001 | 1506 | 001 | 0001 |  |  |
| 721 | EC01 | 722 | EC01 | 7FF | 1506 | 7FE | 0001 | 1506 | 001 | 0001 | 7FF | 1506 |
| 722 | 0A00 | 1EA | 0A00 | 7FE | 01EA | 7FF | 0722 | 1506 | 001 | 0001 |  |  |
| 1EA | 0800 | 1EB | 0800 | 7FF | 1506 | 000 | 01EA | 1506 | 001 | 0001 |  |  |
| 1EB | 0700 | 1EC | 0700 | 1EB | 0700 | 000 | 01EB | 1507 | 000 | 0000 |  |  |
| 1EC | 6E12 | 1ED | 6E12 | 1FF | 0000 | 000 | 0012 | 1507 | 001 | 0001 |  |  |
| 1ED | EE11 | 1EE | EE11 | 1FF | 1507 | 000 | 0011 | 1507 | 001 | 0001 | 1FF | 1507 |
| 1EE | AE0E | 1EF | AE0E | 1FD | 0000 | 000 | 000E | 0000 | 005 | 0101 |  |  |
| 1EF | 0740 | 1F0 | 0740 | 1EF | 0740 | 000 | 01EF | FFFF | 008 | 1000 |  |  |
| 1F0 | 0C00 | 1F1 | 0C00 | 7FF | FFFF | 7FF | 01F0 | FFFF | 008 | 1000 | 7FF | FFFF |
| 1F1 | D716 | 716 | D716 | 7FE | 01F2 | 7FE | D716 | FFFF | 008 | 1000 | 7FE | 01F2 |
| 716 | AC01 | 717 | AC01 | 7FF | FFFF | 7FE | 0001 | FFFF | 008 | 1000 |  |  |
| 717 | F204 | 71C | F204 | 717 | F204 | 7FE | 0004 | FFFF | 008 | 1000 |  |  |
| 71C | 4C01 | 71D | 4C01 | 7FF | FFFF | 7FE | 0001 | FFFE | 009 | 1001 |  |  |
| 71D | 4C01 | 71E | 4C01 | 7FF | FFFF | 7FE | 0001 | FFFD | 009 | 1001 |  |  |
| 71E | 6E05 | 71F | 6E05 | 724 | 0090 | 7FE | 0005 | FF6D | 009 | 1001 |  |  |
| 71F | CE01 | 721 | CE01 | 71F | 0721 | 7FE | 0001 | FF6D | 009 | 1001 |  |  |
| 721 | EC01 | 722 | EC01 | 7FF | FF6D | 7FE | 0001 | FF6D | 009 | 1001 | 7FF | FF6D |
| 722 | 0A00 | 1F2 | 0A00 | 7FE | 01F2 | 7FF | 0722 | FF6D | 009 | 1001 |  |  |
| 1F2 | 0800 | 1F3 | 0800 | 7FF | FF6D | 000 | 01F2 | FF6D | 009 | 1001 |  |  |
| 1F3 | 4E0B | 1F4 | 4E0B | 1FF | 1507 | 000 | 000B | 1474 | 001 | 0001 |  |  |
| 1F4 | EE0A | 1F5 | EE0A | 1FF | 1474 | 000 | 000A | 1474 | 001 | 0001 | 1FF | 1474 |
| 1F5 | AE06 | 1F6 | AE06 | 1FC | F8C0 | 000 | 0006 | F8C0 | 009 | 1001 |  |  |
| 1F6 | 0C00 | 1F7 | 0C00 | 7FF | F8C0 | 7FF | 01F6 | F8C0 | 009 | 1001 | 7FF | F8C0 |
| 1F7 | D716 | 716 | D716 | 7FE | 01F8 | 7FE | D716 | F8C0 | 009 | 1001 | 7FE | 01F8 |
| 716 | AC01 | 717 | AC01 | 7FF | F8C0 | 7FE | 0001 | F8C0 | 009 | 1001 |  |  |
| 717 | F204 | 71C | F204 | 717 | F204 | 7FE | 0004 | F8C0 | 009 | 1001 |  |  |
| 71C | 4C01 | 71D | 4C01 | 7FF | F8C0 | 7FE | 0001 | F180 | 009 | 1001 |  |  |
| 71D | 4C01 | 71E | 4C01 | 7FF | F8C0 | 7FE | 0001 | EA40 | 009 | 1001 |  |  |
| 71E | 6E05 | 71F | 6E05 | 724 | 0090 | 7FE | 0005 | E9B0 | 009 | 1001 |  |  |
| 71F | CE01 | 721 | CE01 | 71F | 0721 | 7FE | 0001 | E9B0 | 009 | 1001 |  |  |
| 721 | EC01 | 722 | EC01 | 7FF | E9B0 | 7FE | 0001 | E9B0 | 009 | 1001 | 7FF | E9B0 |
| 722 | 0A00 | 1F8 | 0A00 | 7FE | 01F8 | 7FF | 0722 | E9B0 | 009 | 1001 |  |  |
| 1F8 | 0800 | 1F9 | 0800 | 7FF | E9B0 | 000 | 01F8 | E9B0 | 009 | 1001 |  |  |
| 1F9 | 4E05 | 1FA | 4E05 | 1FF | 1474 | 000 | 0005 | FE24 | 008 | 1000 |  |  |
| 1FA | EE04 | 1FB | EE04 | 1FF | FE24 | 000 | 0004 | FE24 | 008 | 1000 | 1FF | FE24 |
| 1FB | 0100 | 1FC | 0100 | 1FB | 0100 | 000 | 01FB | FE24 | 008 | 1000 |  |  |

# Вывод

Во время выполнения лабораторной работы я узнал, как работает стек в БЭВМ, научился вызывать подпрограммы, а также изучил тактовое выполнение команд PUSH и POP.