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

Факультет программной инженерии и компьютерной техники

Основы профессиональной деятельности

Лабораторная работа №2

Исследование работы БЭВМ

Вариант 3003

Выполнил: Свечников Константин Денисович, P3130

Проверил: Карташев В. С., преподаватель практики

**Оглавление**

[Задание 2](#_Toc185176934)

[Ход работы 3](#_Toc185176935)

[Описание программы: 3](#_Toc185176936)

[Трассировка программы 5](#_Toc185176937)

[Вариант программы с меньшим числом команд 6](#_Toc185176938)

[Заключение 7](#_Toc185176939)

Задание

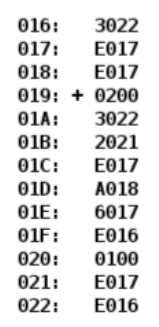
По выданному преподавателем варианту определить функцию, вычисляемую программой, область представления и область допустимых значений исходных данных и результата, выполнить трассировку программы, предложить вариант с меньшим числом команд. При выполнении работы представлять результат и все операнды арифметических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

Рисунок 1

Ход работы

Текст исходной программы описан ниже:

|  |  |  |  |
| --- | --- | --- | --- |
| **Адрес** | **Код команды** | **Мнемоника** | **Комментарий** |
| 016 | 3022 |  | Значение R = D – (A & B) |
| 017 | E017 |  | Значение C = A & B |
| 018 | E017 |  | Значение D |
| 019 | 0200 | CLA | Очистить аккумулятор. AC = 0 |
| 01A | 3022 | OR 022 | Выполнить над содержимым аккумулятора и содержимым ячейки памяти **022** операцию логического сложения. AC = (AC OR A) = (0 OR A) = A |
| 01B | 2021 | AND 021 | Выполнить над каждым разрядом содержимого аккумулятора и содержимым ячейки памяти **021** операцию логического умножения. AC = AC & B = A & B |
| 01C | E017 | ST 017 | Сохранить значение аккумулятора в ячейку памяти **017** (**AC → 017**). C = AC = A & B |
| 01D | A018 | LD 018 | Загрузить содержимое ячейки памяти **018** в аккумулятор (**018 → AC**). AC = D |
| 01E | 6017 | SUB 017 | Вычесть из значения аккумулятора содержимое ячейки памяти **017.** AC = AC – C = D – (A & B) |
| 01F | E016 | ST 016 | Сохранить значение аккумулятора в ячейку памяти **016** (**AC → 016**). R = AC |
| 020 | 0100 | HLT | Остановить программу, перейти в пультовый режим. |
| 021 | E017 |  | Значение B |
| 022 | E016 |  | Значение A |

## Описание программы:

* Программа вычисляет формулу **R = D – (A & B)** и сохраняет в памяти результат. Сначала производится логическое умножение A и B, запись результата в память, затем разность D и (A & B) (значение C), с последующим сохранением результата в ячейку памяти.
* Исходные данные – два набора логических значений A и B и целое знаковое 16-битное число D. Логическое произведение A и B при выполнении вычитания считается целым знаковым 16-битным числом.
* Область допустимых значений результата R: -215 ≤ R ≤ 215 – 1.

Исходную формулу можно привести к виду R = (D + 1) + (~A | ~B)

Рассмотрим три случая:

Случай 1.

Если ограничить разрядность слагаемых, то переполнения не возникает. Однако тогда теряется половина возможных значений D+1 и (~A | ~B).

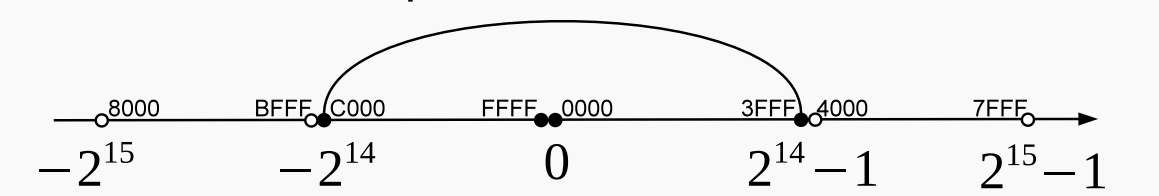


Рисунок 2 - случай 1

Случай 2.

Пусть 214 ≤ (D + 1) ≤ 215 – 1

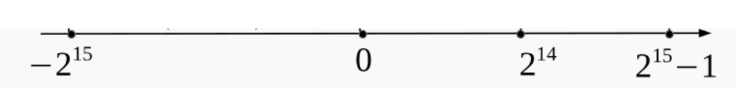


Рисунок 3 - случай 2

Случай 3.

Пусть -215 ≤ (D + 1) ≤ -214 – 1

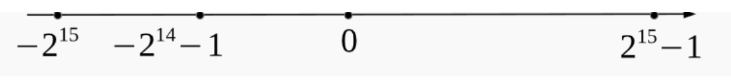


Рисунок 4 - случай 3

* По адресам 019 – 020 расположена программа. Исходные данные C, B и A содержатся по адресам 018, 021 и 022 соответственно. Промежуточный результат C сохраняется по адресу 017. Конечный результат R сохраняется по адресу 016.
* Адрес первой команды программы – 019. Последней – 020

## Трассировка программы

Заданные числа: A = 0xE016, B = 0xE017, D = 0xE017

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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 |
| 016 | 3022 |  |  |  |  | 000 |  |  |  |  |  |
| 017 | E017 |  |  |  |  | 000 |  |  |  |  |  |
| 018 | E017 |  |  |  |  | 000 |  |  |  |  |  |
| 019 | +0200 | 01A | 0200 | 019 | 0200 | 000 | 0019 | 0000 | 0100 |  |  |
| 01A | 3022 | 01B | 3022 | 022 | E016 | 000 | 1FE9 | E106 | 1000 |  |  |
| 01B | 2021 | 01C | 2021 | 021 | E017 | 000 | 001B | E016 | 1000 |  |  |
| 01C | E017 | 01D | E017 | 017 | E016 | 000 | 001C | E016 | 1000 | 017 | E016 |
| 01D | A018 | 01E | A018 | 018 | E017 | 000 | 001D | E017 | 1000 |  |  |
| 01E | 6017 | 01F | 6017 | 017 | E016 | 000 | 001E | 0001 | 0001 |  |  |
| 01F | E016 | 020 | E016 | 016 | 0001 | 000 | 001F | 0001 | 0001 | 016 | 0001 |
| 020 | 0100 | 021 | 0100 | 020 | 0100 | 000 | 0020 | 0001 | 0001 |  |  |
| 021 | E017 |  |  |  |  | 000 |  |  |  |  |  |
| 022 | E016 |  |  |  |  | 000 |  |  |  |  |  |

## Вариант программы с меньшим числом команд

|  |  |  |  |
| --- | --- | --- | --- |
| **Адрес** | **Код команды** | **Мнемоника** | **Комментарий** |
| 016 | A01C | LD 01C | Загрузить содержимое ячейки памяти 01B в аккумулятор. AC = A |
| 017 | 201D | AND 01D | Выполнить над каждым разрядом содержимого аккумулятора и содержимым ячейки памяти 01D операцию логического умножения. AC = AC & B = A & B |
| 018 | 0780 | NEG | Изменить знак содержимого аккумулятора. AC = -AC + 1 |
| 019 | 401E | ADD 01E | Добавить содержимое ячейки памяти 01E к аккумулятору. AC = AC + D = - (A & B) + D |
| 01A | E01F | ST 01F | Сохранить значение аккумулятора в ячейку памяти 01F. R = AC = D – (A & B) |
| 01B | 0100 | HLT | Остановить программу, перейти в пультовый режим. |
| 01C | E016 |  | Значение A |
| 01D | E017 |  | Значение B |
| 01E | E017 |  | Значение D |
| 01F | 0000 |  | Значение R = D – (A & B) |

Трассировка по упрощенной программе

Заданные числа: A = 0x0111, B = 0x0333, D = 0x000F

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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 |
| 016 | +A01C | 017 | A01C | 01C | 0111 | 000 | 0016 | 0111 | 0000 |  |  |
| 017 | 201D | 018 | 201D | 01D | 0333 | 000 | 0017 | 0111 | 0000 |  |  |
| 018 | 0780 | 019 | 0780 | 018 | 0780 | 000 | 0018 | FEEF | 1000 |  |  |
| 019 | 401E | 01A | 401E | 01E | 000F | 000 | 0019 | FEFE | 1000 |  |  |
| 01A | E01F | 01B | E01F | 01F | FEFE | 000 | 001A | FEFE | 1000 | 01F | FEFE |
| 01B | 0100 | 01C | 0100 | 01B | 0100 | 000 | 001B | FEFE | 1000 |  |  |
| 01C | 0111 |  |  |  |  |  |  |  |  |  |  |
| 01D | 0333 |  |  |  |  |  |  |  |  |  |  |
| 01E | 000F |  |  |  |  |  |  |  |  |  |  |
| 01F | 0000 |  |  |  |  |  |  |  |  |  |  |

Заключение

В ходе выполнения лабораторной работы я познакомился с архитектурой базовой ЭВМ, изучил ее команды, способы представления данных в ее памяти, а также научился разбирать программы БЭВМ и делать их трассировку.