ВСанкт-Петербургский Национальный Исследовательский Университет ИТМО

Факультет Программной Инженерии и Компьютерной Техники



Вариант № 8800555

Лабораторная работа №2

по дисциплине

‘Основы профессиональной деятельности’

Выполнил студент группы P3130:

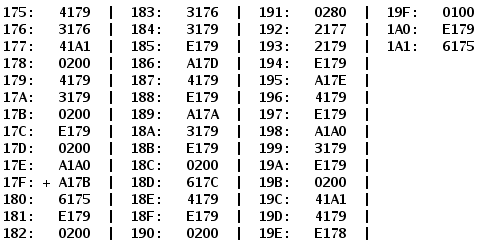
Курбанов Ф.А.

Преподаватель:  
 Перминов И.В.

Санкт-Петербург 2020 г.

**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.



**Исходная программа**

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код команды | Мнемоника | Комментарии |
| 175 | 4179 |  | Исходные данные |
| 176 | 3176 |  | Исходные данные |
| 177 | 41A1 |  | Исходные данные |
| 178 | 0200 |  | Исходные данные |
| 179 | 4179 |  | Исходные данные |
| 17A | 3179 |  | Исходные данные |
| 17B | 0200 |  | Исходные данные |
| 17C | E179 |  | Исходные данные |
| 17D | 0200 |  | Исходные данные |
| 17E | A1A0 |  | Исходные данные |
| 17F | + A17B | LD M | Загрузка M -> AC |
| 180 | 6175 | SUB M | Вычитание AC – M -> AC |
| 181 | E179 | ST M | Сохранение AC -> M |
| 182 | 0200 | CLA | Очистка аккумулятора 0 -> AC |
| 183 | 3176 | OR M | Логическое ИЛИ ^(^M & ^AC) -> AC |
| 184 | 3179 | OR M | Логическое ИЛИ ^(^M & ^AC) -> AC |
| 185 | E179 | ST M | Сохранение AC -> M |
| 186 | A17D | LD M | Загрузка M -> AC |
| 187 | 4179 | ADD M | Сложение M + AC -> AC |
| 188 | E179 | ST M | Сохранение AC -> M |
| 189 | A17A | LD M | Загрузка M -> AC |
| 18A | 3179 | OR M | Логическое ИЛИ ^(^M & ^AC) -> AC |
| 18B | E179 | ST M | Сохранение AC -> M |
| 18C | 0200 | CLA | Очистка аккумулятора 0 -> AC |
| 18D | 617C | SUB M | Вычитание AC – M -> AC |
| 18E | 4179 | ADD M | Сложение M + AC -> AC |
| 18F | E179 | ST M | Сохранение AC -> M |
| 190 | 0200 | CLA | Очистка аккумулятора 0 -> AC |
| 191 | 0280 | NOT | ^AC -> AC |
| 192 | 2177 | AND M | Логическое умножение M & AC -> AC |
| 193 | 2179 | AND M | Логическое умножение M & AC -> AC |
| 194 | E179 | ST M | Сохранение AC -> M |
| 195 | A17E | LD M | Загрузка M -> AC |
| 196 | 4179 | ADD M | Сложение M + AC -> AC |
| 197 | E179 | ST M | Сохранение AC -> M |
| 198 | A1A0 | LD M | Загрузка M -> AC |
| 199 | 3179 | OR M | Логическое ИЛИ ^(^M & ^AC) -> AC |
| 19A | E179 | ST M | Сохранение AC -> M |
| 19B | 0200 | CLA | Очистка аккумулятора 0 -> AC |
| 19C | 41A1 | ADD M | Сложение M + AC -> AC |
| 19D | 4179 | ADD M | Сложение M + AC -> AC |
| 19E | E178 | ST M | Сохранение AC -> M |
| 19F | 0100 | HLT | Останов |
| 1A0 | E179 |  | Исходные данные |
| 1A1 | 6175 |  | Исходные данные |

**Назначение программы и реализуемые ею функции**

Назначение программы:

Вычисление значения по формуле

Реализуемая формула:

R = ((((((((X1 - X2) | X3) + X4) | X5) - X6) & X7) + X8) | X9) + X10

**Область допустимых значений**

R ∈ [-215; 215 - 2], – результат

X1 ∈ [-1024; 210 - 1],

X2 ∈ [-1024; 210 - 1],

X3 ∈ [-2048; 211 - 1],

X4 ∈ [-2048; 211 - 1],

X5 ∈ [-4096; 212 - 1], – исходные данные

X6 ∈ [-4096; 212 - 1],

X7 ∈ [-213; 213 - 1],

X8 ∈ [-213; 213 - 1],

X9 ∈ [-16384; 214 - 1],

X10 ∈ [-16384; 214 - 1];

**Расположение в памяти ЭВМ программы, исходных данных и результатов**

175, …, 17E, 1A0, 1A1 – исходные данные

17F, …, 19F – команды

179, 178 – результаты

**Адреса первой и последней выполняемых команд программы**

Адрес первой команды: 17F

Адрес последней команды: 19F

**Трассировка программы**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Выполняемая команда** | | **Содержимое регистров процессора после выполнения программы.** | | | | | | | | | **Ячейки, значения которых изменились после выполнения команды** | |
| **Адрес** | **Код** | **IP** | **CR** | **AR** | **DR** | **SP** | **BR** | **AC** | **PS** | **NZVC** | **Адрес** | **Новый код** |
| **17F** | A17B | 17F | 0000 | 000 | 0000 | 000 | 0000 | 0000 | 004 | 0100 |  |  |
| **17F** | A17B | 180 | A17B | 17B | 0200 | 000 | 017F | 0200 | 000 | 0000 |  |  |
| **180** | 6175 | 181 | 6175 | 175 | 4179 | 000 | 0180 | C087 | 008 | 1000 |  |  |
| **181** | E179 | 182 | E179 | 179 | C087 | 000 | 0181 | C087 | 008 | 1000 | 179 | C087 |
| **182** | 0200 | 183 | 0200 | 182 | 0200 | 000 | 0182 | 0000 | 004 | 0100 |  |  |
| **183** | 3176 | 184 | 3176 | 176 | 3176 | 000 | CE89 | 3176 | 000 | 0000 |  |  |
| **184** | 3179 | 185 | 3179 | 179 | C087 | 000 | 0E08 | F1F7 | 008 | 1000 |  |  |
| **185** | E179 | 186 | E179 | 179 | F1F7 | 000 | 0185 | F1F7 | 008 | 1000 | 179 | F1F7 |
| **186** | A17D | 187 | A17D | 17D | 0200 | 000 | 0186 | 0200 | 000 | 0000 |  |  |
| **187** | 4179 | 188 | 4179 | 179 | F1F7 | 000 | 0187 | F3F7 | 008 | 1000 |  |  |
| **188** | E179 | 189 | E179 | 179 | F3F7 | 000 | 0188 | F3F7 | 008 | 1000 | 179 | F3F7 |
| **189** | A17A | 18A | A17A | 17A | 3179 | 000 | 0189 | 3179 | 000 | 0000 |  |  |
| **18A** | 3179 | 18B | 3179 | 179 | F3F7 | 000 | 0C00 | F3FF | 008 | 1000 |  |  |
| **18B** | E179 | 18C | E179 | 179 | F3FF | 000 | 018B | F3FF | 008 | 1000 | 179 | F3FF |
| **18C** | 0200 | 18D | 0200 | 18C | 0200 | 000 | 018C | 0000 | 004 | 0100 |  |  |
| **18D** | 617C | 18E | 617C | 17C | E179 | 000 | 018D | 1E87 | 000 | 0000 |  |  |
| **18E** | 4179 | 18F | 4179 | 179 | F3FF | 000 | 018E | 1286 | 001 | 0001 |  |  |
| **18F** | E179 | 190 | E179 | 179 | 1286 | 000 | 018F | 1286 | 001 | 0001 | 179 | 1286 |
| **190** | 0200 | 191 | 0200 | 190 | 0200 | 000 | 0190 | 0000 | 005 | 0101 |  |  |
| **191** | 0280 | 192 | 0280 | 191 | 0280 | 000 | 0191 | FFFF | 009 | 1001 |  |  |
| **192** | 2177 | 193 | 2177 | 177 | 41A1 | 000 | 0192 | 41A1 | 001 | 0001 |  |  |
| **193** | 2179 | 194 | 2179 | 179 | 1286 | 000 | 0193 | 0080 | 001 | 0001 |  |  |
| **194** | E179 | 195 | E179 | 179 | 0080 | 000 | 0194 | 0080 | 001 | 0001 | 179 | 0080 |
| **195** | A17E | 196 | A17E | 17E | A1A0 | 000 | 0195 | A1A0 | 009 | 1001 |  |  |
| **196** | 4159 | 197 | 4159 | 159 | 0000 | 000 | 0196 | A1A0 | 008 | 1000 |  |  |
| **197** | E179 | 198 | E179 | 179 | A1A0 | 000 | 0197 | A1A0 | 008 | 1000 | 179 | A1A0 |
| **198** | A1A0 | 199 | A1A0 | 1A0 | E179 | 000 | 0198 | E179 | 008 | 1000 |  |  |
| **199** | 3179 | 19A | 3179 | 179 | A1A0 | 000 | 1E06 | E1F9 | 008 | 1000 |  |  |
| **19A** | E179 | 19B | E179 | 179 | E1F9 | 000 | 019A | E1F9 | 008 | 1000 | 179 | E1F9 |
| **19B** | 0200 | 19C | 0200 | 19B | 0200 | 000 | 019B | 0000 | 004 | 0100 |  |  |
| **19C** | 41A1 | 19D | 41A1 | 1A1 | 6175 | 000 | 019C | 6175 | 000 | 0000 |  |  |
| **19D** | 4179 | 19E | 4179 | 179 | E1F9 | 000 | 019D | 436E | 001 | 0001 |  |  |
| **19E** | E178 | 19F | E178 | 178 | 436E | 000 | 019E | 436E | 001 | 0001 | 178 | 436E |
| **19F** | 0100 | 1A0 | 0100 | 19F | 0100 | 000 | 019F | 436E | 001 | 0001 |  |  |

**Трассировка программы 1**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Адр** | **Знчн** | **IP** | **CR** | **AR** | **DR** | **SP** | **BR** | **AC** | **PS** | **NZVC** | **Адр** | **Знчн** |
| **00B** | A000 | 00B | 0000 | 000 | 0000 | 000 | 0000 | 0000 | 004 | 0100 |  |  |
| **00B** | A000 | 00C | A000 | 000 | FC00 | 000 | 000B | FC00 | 008 | 1000 |  |  |
| **00C** | 6001 | 00D | 6001 | 001 | 03FF | 000 | 000C | F801 | 009 | 1001 |  |  |
| **00D** | 3002 | 00E | 3002 | 002 | 0000 | 000 | 07FE | F801 | 009 | 1001 |  |  |
| **00E** | 4003 | 00F | 4003 | 003 | 0FFF | 000 | 000E | 0800 | 001 | 0001 |  |  |
| **00F** | 3004 | 010 | 3004 | 004 | E000 | 000 | 17FF | E800 | 009 | 1001 |  |  |
| **010** | 6005 | 011 | 6005 | 005 | E000 | 000 | 0010 | 0800 | 001 | 0001 |  |  |
| **011** | 2006 | 012 | 2006 | 006 | 0000 | 000 | 0011 | 0000 | 005 | 0101 |  |  |
| **012** | 4007 | 013 | 4007 | 007 | C000 | 000 | 0012 | C000 | 008 | 1000 |  |  |
| **013** | 3008 | 014 | 3008 | 008 | 0000 | 000 | 3FFF | C000 | 008 | 1000 |  |  |
| **014** | 4009 | 015 | 4009 | 009 | C000 | 000 | 0014 | 8000 | 009 | 1001 |  |  |
| **015** | E00A | 016 | E00A | 00A | 8000 | 000 | 0015 | 8000 | 009 | 1001 | 00A | 8000 |
| **016** | 0100 | 017 | 0100 | 016 | 0100 | 000 | 0016 | 8000 | 009 | 1001 |  |  |

**Трассировка программы 2**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Адр** | **Знчн** | **IP** | **CR** | **AR** | **DR** | **SP** | **BR** | **AC** | **PS** | **NZVC** | **Адр** | **Знчн** |
| **00B** | A000 | 00B | 0000 | 000 | 0000 | 000 | 0000 | 0000 | 004 | 0100 |  |  |
| **00B** | A000 | 00C | A000 | 000 | 03FF | 000 | 000B | 03FF | 000 | 0000 |  |  |
| **00C** | 6001 | 00D | 6001 | 001 | FC00 | 000 | 000C | 07FF | 000 | 0000 |  |  |
| **00D** | 3002 | 00E | 3002 | 002 | 07FF | 000 | F800 | 07FF | 000 | 0000 |  |  |
| **00E** | 4003 | 00F | 4003 | 003 | 07FF | 000 | 000E | 0FFE | 000 | 0000 |  |  |
| **00F** | 3004 | 010 | 3004 | 004 | 0FFF | 000 | F000 | 0FFF | 000 | 0000 |  |  |
| **010** | 6005 | 011 | 6005 | 005 | F000 | 000 | 0010 | 1FFF | 000 | 0000 |  |  |
| **011** | 2006 | 012 | 2006 | 006 | FFFF | 000 | 0011 | 1FFF | 000 | 0000 |  |  |
| **012** | 4007 | 013 | 4007 | 007 | 1FFF | 000 | 0012 | 3FFE | 000 | 0000 |  |  |
| **013** | 3008 | 014 | 3008 | 008 | 3FFF | 000 | C000 | 3FFF | 000 | 0000 |  |  |
| **014** | 4009 | 015 | 4009 | 009 | 3FFF | 000 | 0014 | 7FFE | 000 | 0000 |  |  |
| **015** | E00A | 016 | E00A | 00A | 7FFE | 000 | 0015 | 7FFE | 000 | 0000 | 00A | 7FFE |
| **016** | 0100 | 017 | 0100 | 016 | 0100 | 000 | 0016 | 7FFE | 000 | 0000 |  |  |

**Вариант с меньшим числом команд**

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код команды | Мнемоника | Комментарии |
| 000 | //X1 |  | Исходные данные |
| 001 | //X2 |  | Исходные данные |
| 002 | //X3 |  | Исходные данные |
| 003 | //X4 |  | Исходные данные |
| 004 | //X5 |  | Исходные данные |
| 005 | //X6 |  | Исходные данные |
| 006 | //X7 |  | Исходные данные |
| 007 | //X8 |  | Исходные данные |
| 008 | //X9 |  | Исходные данные |
| 009 | //X10 |  | Исходные данные |
| 00A | //R |  | Исходные данные |
| 00B | + A000 | LD X1 | Загрузка 000 -> AC |
| 00C | 6001 | SUB X2 | Вычитание AC – 001 -> AC |
| 00D | 3002 | OR X3 | Логическое ИЛИ ^(^002 & ^AC) -> AC |
| 00E | 4003 | ADD X4 | Сложение 003 + AC -> AC |
| 00F | 3004 | OR X5 | Логическое ИЛИ ^(^004 & ^AC) -> AC |
| 010 | 6005 | SUB X6 | Вычитание AC – 006 -> AC |
| 011 | 2006 | AND X7 | Логическое умножение 006 & AC -> AC |
| 012 | 4007 | ADD X8 | Сложение 007 + AC -> AC |
| 013 | 3008 | OR X9 | Логическое ИЛИ ^(^008 & ^AC) -> AC |
| 014 | 4009 | ADD X10 | Сложение 009 + AC -> AC |
| 015 | E00A | ST R | Сохранение AC -> 00A |
| 016 | 0100 | HTL | Останов |

**Вывод**

В результате лабораторной работы я познакомился со структурой ЭВМ, узнал как устроены и связаны его основные элементы. Узнал структуру и виды команд, а также то, как представлены данные в памяти БЭВМ.

**Числа для трассировки:**

R = ((((((((X1 - X2) | X3) + X4) | X5) - X6) & X7) + X8) | X9) + X10

Пытаемся сломать ОДЗ в отрицательную сторону:

X1 = -102410 = 1111 1100 0000 00002 = FC0016

X2 = 102310 = 0000 0011 1111 11112 = 03FF16

X3 = 0

X4 = -204810 = 1111 1000 0000 00002 = F80016

X5 = 0

X6 = 409510 = 0000 1111 1111 11112 = 0FFF16

X7 = -819210 = 1110 0000 0000 00002 = E00016

X8 = -819210 = 1110 0000 0000 00002 = E00016

X9 = 0

X10 = -1638410 = 1100 0000 0000 00002 = C00016

R = -3276810

При сложении/вычитании выбираем числа из ОДЗ так, чтобы получалось минимальное число. При логическом ИЛИ мы можем только увеличить число либо никак на него не повлиять – выбираем число которое не увеличит наш результат (0). При логическом умножении мы можем либо уменьшить результат, либо никак не повлиять на него, поэтому выбираем такое отрицательное число, у которого больше всего нулей в старших разрядах – оно сильнее всего уменьшит наш результат.

Пытаемся сломать ОДЗ в положительную сторону:

X1 = 102310 = 0000 0011 1111 11112 = 03FF16

X2 = -102410 = 1111 1100 0000 00002 = FC0016

X3 = 204710 = 0000 0111 1111 11112 = 07FF16

X4 = 204710 = 0000 0111 1111 11112 = 07FF16

X5 = 409510 = 0000 1111 1111 11112 = 0FFF16

X6 = -409610 = 1111 0000 0000 00002 = F00016

X7 = -110 = 1111 1111 1111 11112 = FFFF16

X8 = 819110 = 0001 1111 1111 11112 = 1FFF16

X9 = 1638310 = 0011 1111 1111 11112 = 3FFF16

X10 = 1638310 = 0011 1111 1111 11112 = 3FFF16

R = 3276610

При сложении/вычитании выбираем числа из ОДЗ так, чтобы получалось максимально число. При логическом ИЛИ мы можем только увеличить число либо никак на него не повлиять – выбираем число которое увеличит наш результат – максимальное число из промежутка для данной переменной. При логическом умножении мы можем либо уменьшить результат, либо никак не повлиять на него, поэтому выбираем такое число, которое никак не повлияет на наше число (-1)