Пензенский государственный университет Кафедра «Вычислительная техника»

# ОТЧЕТ

по лабораторной работе №2

по дисциплине: "Арифметические и логические основы вычислительной техники" на тему: "Перевод чисел из одной системы счисления в другую. Перевод чисел из одной системы счисления в другую. Формат представления чисел с ФТв цифровых процессорах"

Выполнил:

студент группы 23ВВВ3 бригады 3:

Полиневский Вадим

Принял:

Калиниченко Е. И

Пенза, 2024

Цель – научиться переводить числа, представленные в оперативной памяти в обычное для человека представление числа.

**Задание для студента с №"Z" в группе В3:**

п.1 a=**Z**+47; b= − (**Z**+62)

п.2 c=0,(**Z**+47); d=−0,(**Z**+62)

п.3 f=(**Z**+47),(**Z**+62); g=−(**Z**+62),(**Z**+47)

N в группе – 22;

**a** = 69; **b** = - 84; **c** = 0, 69; **d** = -0,84; **f** = 69,84; **g** = -84,69;

**Ход работы**

1. Перевёл числа a = 69 и b = -84 в двоичную систему счисления.

a = 6910 = 4516 = 100 0101 b = -8410= -5416 = **-**101 0100

1. Представил a и b как операнды в прямом и дополнительном коде в процессорах разной разрядности. Для представления в дополнительном коде двоичная запись числа b была инвертирована, затем результат был сложен с 1.

*8-разрядный процессор:*

Прямой код:

a = 0100 0101

b = 1101 0100

Дополнительный код:

a = 0100 0101

b = 1010 1100

*16-разрядный процессор:*

Прямой код:

a =0000 0000 0100 0101

b = 1000 0000 0101 0100

Дополнительный код:

a =0000 0000 0100 0101

b = 1111 1111 1010 1100

*32-разрядный процессор:*

Основной код:

a = 0000 0000 0000 0000 0000 0000 0100 0101

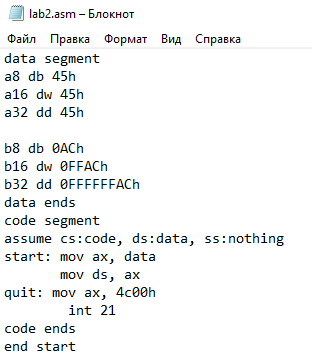
b = 1000 0000 0000 0000 0000 0000 0101 0100

Дополнительный код:

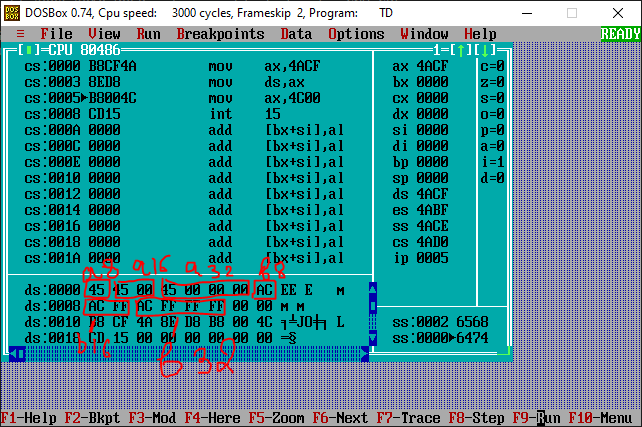
a = 0000 0000 0000 0000 0000 0000 0100 0101

b = 1111 1111 1111 1111 1111 1111 1010 1100

1. Полученные представления чисел разместил в оперативной памяти. Для этого написал следующую программу:

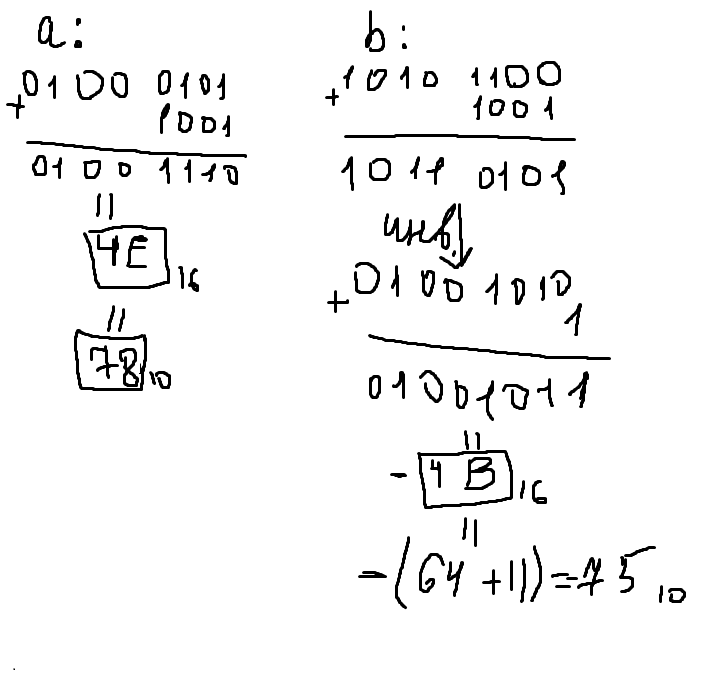


**Представление в оперативной памяти:**

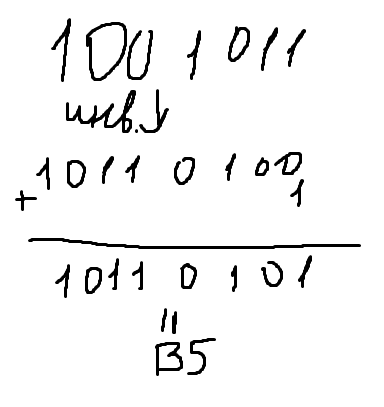


1. Увеличил в оперативной памяти каждого процессора значение младших байтов чисел на 10012.

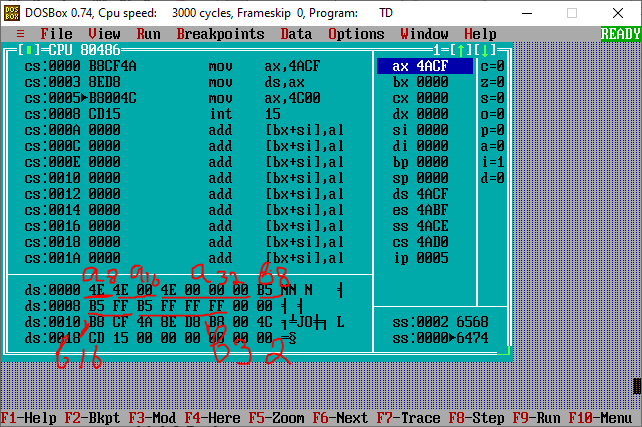
Измененные значения:



Перевод числа полученного числа b для представления в памяти:

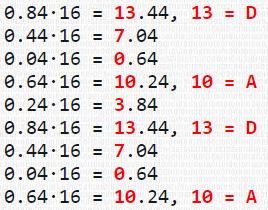
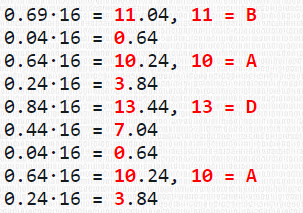


Представление в оперативной памяти:



1. Перевёл числа с = 0,69 и d = -0,84в двоичную систему счисления.

c: d:



с = 0,6910 = B0A3D70A316 ~ B0A3D70A16 = 0,1011 0000 1010 0011 1101 0111 0000 10102

d = -0,8410 = -0,D70A3D70A16 ~ -0,D70A3D7116 = -0,1101 0111 0000 1010 0011 1101 0111 00012

1. Представил с и d как операнды в прямом и дополнительном коде в процессорах разной разрядности.

16-разрядный процессор

прямой код:

с = 0101 1000 0101 0001

d = 1110 1011 1000 0101

дополнительный код:

с = 0101 1000 0101 0001

d = 1001 0100 0111 1011

32-разрядный процессор

прямой код:

c = 0101 1000 0101 0001 1110 1011 1000 0101

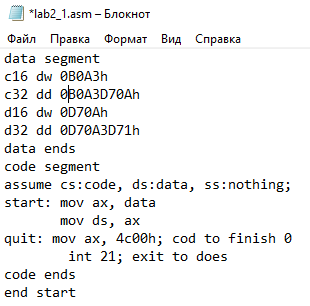
d = 1110 1011 1000 0101 0001 1110 1011 1000

дополнительный код:

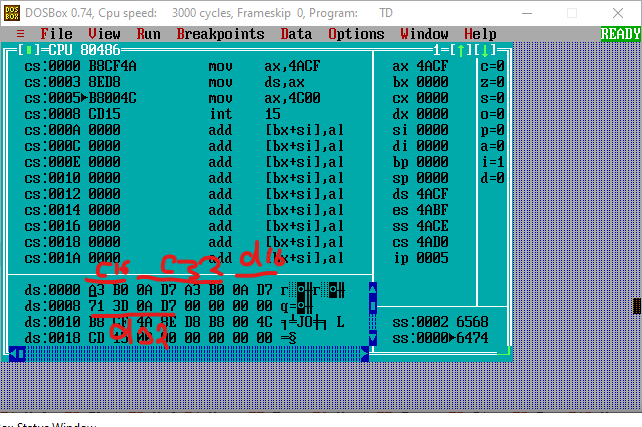
c = 0101 1000 0101 0001 1110 1011 1000 0101

d = 1001 0100 0111 1010 1110 0001 0100 1000

1. Полученные представления чисел разместил в оперативной памяти. Для этого написал следующую программу:



Представление в оперативной памяти:



1. Увеличил в оперативной памяти каждого процессора значение младших байтов чисел на 10012.

Изменение значения c в 16:

|  |
| --- |
| 0101 1000 0101 10102 |
| 10012 |
| 0101 1000 0110 00112 |

с = 0, 1011 0000 1011 01002

c ­= B0B4­16­ = 11 \* 16-1 + 0\*16-2 +11\*16-3+4\*16-4=0,6902410

Изменение значения c в 32:

|  |
| --- |
| 0101 1000 0101 0001 1110 1011 1000 0101 |
| 10012 |
| 0101 1000 0101 0001 1110 1011 1000 11102 |

с = 0101 1000 0101 0001 1110 1011 1000 0101

0,1011 0000 1010 0011 1101 0111 0001 1110 2= B0A3B71E = 0.689998

Изменение значения d в 16:

|  |
| --- |
| 1001 0100 0111 10112 |
| 10012 |
| 1001 0100 1000 01002 |

инверсия-------> 1110 1011 0111 1011

|  |
| --- |
| 1110 1011 0111 10112 |
| 12 |
| 1110 1011 0111 10112 |

d =-0. 1101 0110 1111 10002 = -0.EB7B16=0.9198410

1101 0110 1111 0110 = D6F616 = 0.8396

Изменение значения d в 32:

|  |
| --- |
| 1001 0100 0111 1010 1110 0001 0100 10002 |
| 10012 |
| 1001 0100 0111 1010 1110 0001 0101 00012 |

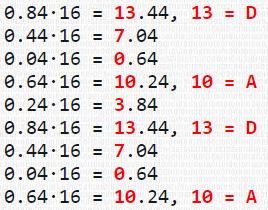
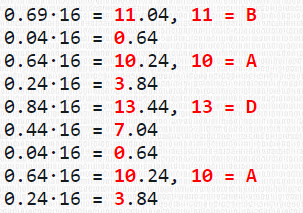
инверсия+1 ------->

1110 1011 1000 0101 0001 1110 1010 11102

1101 0111 0000 1010 0011 1101 0101 1100= 0.D70A3D5C16 = 0.8399910

1. Перевел числа f = 69,84 и g = -84,69 в двоичную систему:



f = 45, D70A3D70A16 = 0100 0101, 1101 0111 0011 1101 0111 0001

g = -54, B0A3D70A316 = -0101 0100, 1011 0000 1010 0011 1011 0111 0000 1011

1. Представил с и d как операнды в прямом и дополнительном коде в процессорах разной разрядности:

16-разрядный процессор:

прямой код:

f = 0100 0101, 0110 1011

g = 0101 0100, 1101 1000

дополнительный код:

f = 0100 0101, 0110 1011

g = 1010 1100, 1010 0101

1. разрядный процессор:

прямой код:

f = 0000 0000 0100 0101, 0110 1011 1001 1110

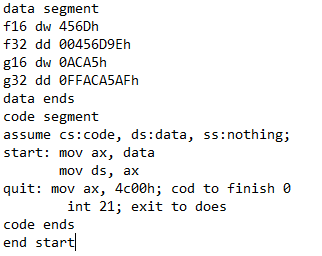
g = 1111 1111 0101 0100, 1101 1000 0101 0001

Дополнительный код:

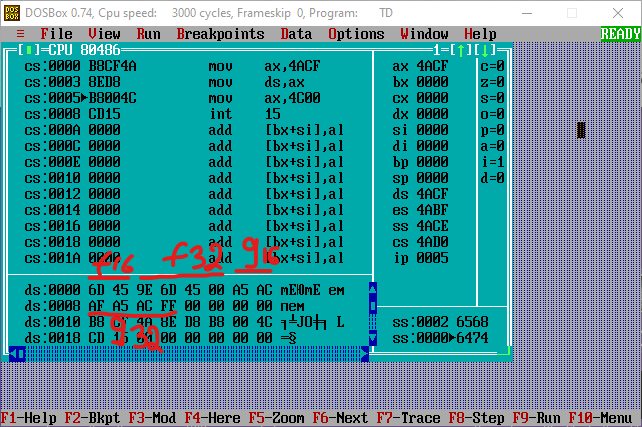
f = 0000 0000 0100 0101, 0110 1011 1001 1110

g = 1111 1111 1010 1100, 1010 0111 1010 1111

Написал следующую программу:



Размещение в оперативной памяти:



1. Увеличил в оперативной памяти каждого процессора значение младших байтов чисел на 10012

Изменение значения f в 16:

|  |
| --- |
| 0100 0101, 0110 10112 |
| 10012 |
| 0100 0101, 0111 01002 |

f = 45, 7416 = 69,E8= 69.90

Изменение значения f в 32:

|  |
| --- |
| 0000 0000 0100 0101, 0110 1011 1001 11102 |
| 10012 |
| 0000 0000 0100 0101, 0110 1011 1010 01112 |

f = 45, D74E16 = 69,841

Изменение значения g в 16:

|  |
| --- |
| 1010 1100, 1010 01012 |
| 10012 |
| 1010 1100, 1010 11102 |

инверсия+1 в обоих частях ------->

-0101 0100, 0101 00102

g = -54, A416 = 84.6410

Изменение значения g в 32:

|  |
| --- |
| 1111 1111 1010 1100, 1010 0111 1010 11112 |
| 10012 |
| 1111 1111 1010 1100, 1010 0111 1011 10002 |

инверсия+1 в обоих частях ------->

-0000 0000 0101 0100, 0101 1000 0100 0111

g = -54, B08E16 = 84.689610

Вывод: получил опыт в переводе целых, дробных и смешанных чисел из десятичной системы счисления в двоичную и из двоичной системы в десятичную с использованием шестнадцатеричной системы счисления в качестве промежуточной; рассмотрела формат представления чисел с фиксированной точкой в цифровых процессорах.