Пензенский государственный университет
Кафедра «Вычислительная техника»

ОТЧЕТ

по лабораторной работе №2
по дисциплине: "Алгебраические и логические основы вычислительной техники"
на тему: "Перевод чисел из одной системы счисления в другую.

Формат с ФТ представления чисел в цифровых процессорах"

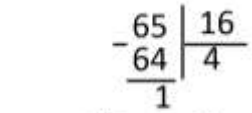
Выполнил:
xxxxxxxxxxx

Принял:
xxxxxxxxxxx

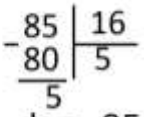
Пенза, 2021

Ход работы

1. Перевела числа a = 65 и b = -85 в двоичную систему счисления



$a = 65_{10} = 41_{16} = 1000001_2$



$b = -85_{10} = -55_{16} = -1010101_2$

2. Представила 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-ми разрядный процессор
прямой код:
$[a]_1 = 0100\ 0001$ $[b]_1 = 1101\ 0101$
дополнительный код:
$[a]_2 = 0100\ 0001$ $[b]_2 = 1010\ 1011$

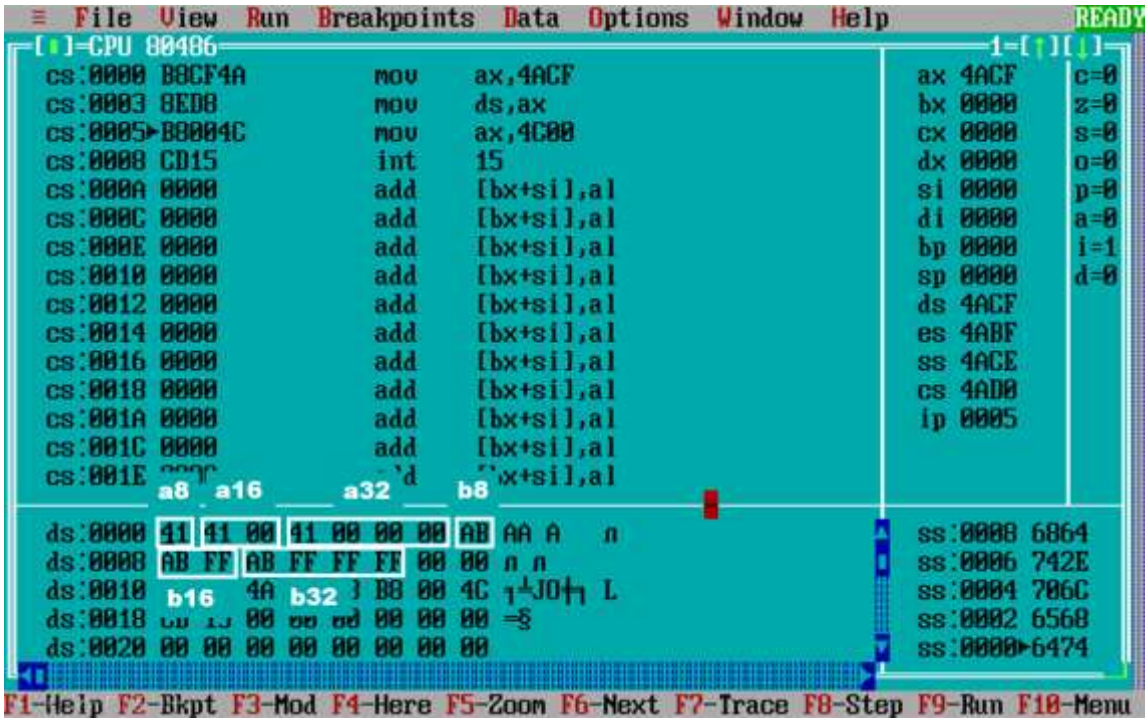
16-ти разрядный процессор
прямой код:
$[a]_1 = 0000\ 0000\ 0100\ 0001$ $[b]_1 = 1000\ 0000\ 0101\ 0101$
дополнительный код:
$[a]_2 = 0000\ 0000\ 0100\ 0001$ $[b]_2 = 1111\ 1111\ 10101011$

32-разрядный процессор
прямой код:
$[a]_1 = 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0100\ 0001$
$[b]_1 = 1000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0101\ 0101$
дополнительный код:
$[a]_2 = 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0100\ 0001$
$[b]_2 = 1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 1010\ 1011$

3. Полученные представления чисел разместила в оперативной памяти. Для этого написала следующую программу:

```
Файл  Правка  Формат  Вид  Справка
data segment
        a8 db 41h
        a16 dw 41h
        a32 dd 41h

        b8 db 0A8h
        b16 dw 0FFA8h
        b32 dd 0FFFFFFA8h
data ends

code segment
assume cs:code, ds:data, ss:nothing

start:          mov ax, data
                mov ds, ax

quit:           mov ax, 4c00h
                int 21
code ends
end start
```

Представление в оперативной памяти:



4. Увеличила в оперативной памяти каждого процессора значение младших байтов чисел на 1001_2.
   Измененные значения:

$$\begin{array}{r} +1000001_2 \\ 1001_2 \\ \hline 1001010_2 \end{array}$$

$1001010_2 = 4A_{16} = 74_{10}$

$$\begin{array}{r} +10101011_2 \\ 1001_2 \\ \hline 10110100_2 \end{array}$$

инверсия

$$\begin{array}{r} +01001011_2 \\ 1_2 \\ \hline 01001100_2 \end{array}$$

$-1001100_2 = -4C_{16} = -76_{10}$

Перевод результата для представления в памяти:

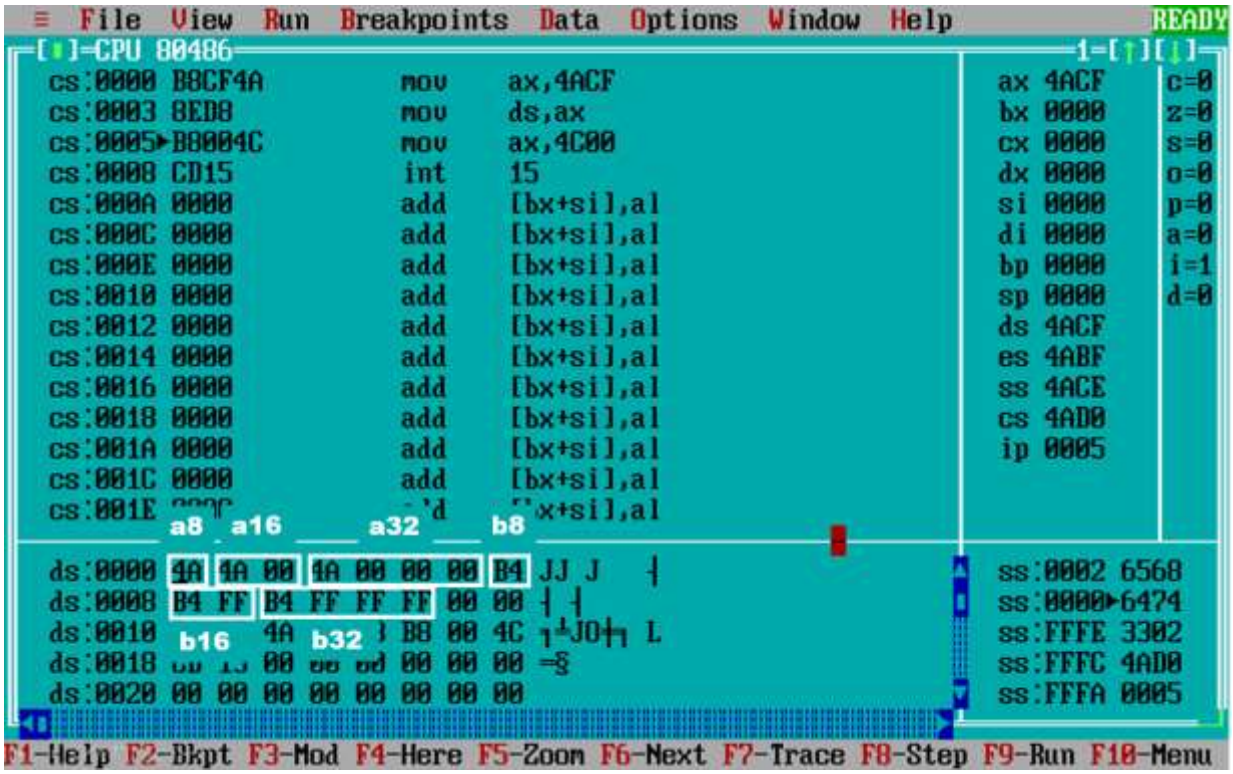
$01001100_2 \xrightarrow{\text{инверсия}} 10110011_2$

$$\begin{array}{r} +10110011_2 \\ 1_2 \\ \hline 10110100_2 \end{array}$$

$10110100_2 = B4_{16}$

```
data segment
        a8 db 4Ah
        a16 dw 4Ah
        a32 dd 4Ah

        b8 db 0B4h
        b16 dw 0FFB4h
        b32 dd 0FFFFFFB4h
data ends
```

Представление в оперативной памяти:



5. Перевела числа c = 0,65 и d = -0,85 в двоичную систему счисления.

c = 0,65
0,65*16 = 10,4
0,40*16 = 6,4
0,40*16 = 6,4
0,40*16 = 6,4
0,40*16 = 6,4
0,40*16 = 6,4
0,40*16 = 6,4
0,40*16 = 6,4
0,40*16 = 6,4

d = -0,85
0,85*16 = 13,6
0,60*16 = 9,6
0,60*16 = 9,6
0,60*16 = 9,6
0,60*16 = 9,6
0,60*16 = 9,6
0,60*16 = 9,6
0,60*16 = 9,6
0,60*16 = 9,6

$c = 0,65_{10} = 0,A6666666_{16} = 0,1010\ 0110\ 0110\ 0110\ 0110\ 0110\ 0110\ 0110_2$

$d = -0,85_{10} = -0,D999999A_{16} = -0,1101\ 1001\ 1001\ 1001\ 1001\ 1001\ 1001\ 1010_2$

6. Представила c и d как операнды в прямом и дополнительном коде в процессорах разной разрядности.

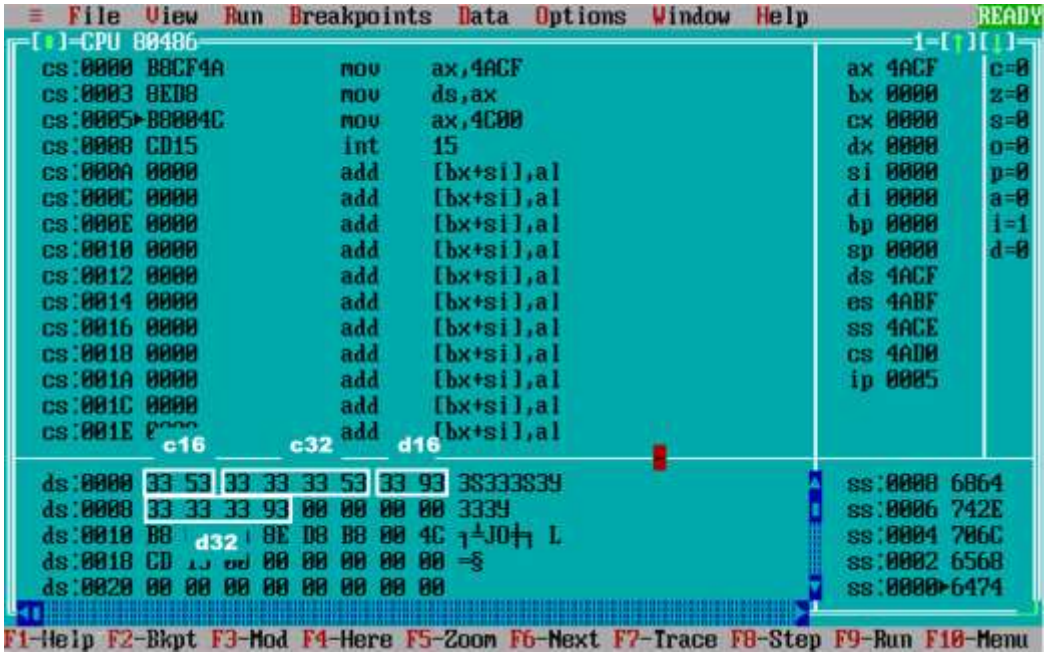
16-разрядный процессор
прямой код:
$[c]_1$ = 0101 0011 0011 0011 $[d]_1$ = 1110 1100 1100 1101
дополнительный код:
$[c]_2$ = 0101 0011 0011 0011 $[d]_2$ = 1001 0011 0011 0011

32-разрядный процессор
прямой код:
$[c]_1$ = 0101 0011 0011 0011 0011 0011 0011 0011
$[d]_1$ = 1110 1100 1100 1100 1100 1100 1100 1101
дополнительный код:
$[c]_2$ = 0101 0011 0011 0011 0011 0011 0011 0011
$[d]_2$ = 1001 0011 0011 0011 0011 0011 0011 0011

7. Полученные представления чисел разместила в оперативной памяти. Для этого написала следующую программу:

```
Файл  Правка  Формат  Вид  Справка
data segment
        c16 dw 5333h
        c32 dd 53333333h

        d16 dw 9333h
        d32 dd 93333333h
data ends

code segment
assume cs:code, ds:data, ss:nothing

start:          mov ax, data
                mov ds, ax

quit:           mov ax, 4c00h
                int 21
code ends
end start
```

Представление в оперативной памяти:



8. Увеличила в оперативной памяти каждого процессора значение младших байтов чисел на 1001_2.
   Изменение значения с:

$$+\begin{array}{r} 0101\ 0011\ 0011\ 0011_2 \\ 1001_2 \\ \hline 0101\ 0011\ 0011\ 1100_2 \end{array}$$

$C = 0{,}A678_{16} = 10*16^{-1}+6*16^{-2}+6*16^{-3}+8*16^{-4} = 0{,}65024_{10}$

$$+\begin{array}{r} 0101\ 0011\ 0011\ 0011\ 0011\ 0011\ 0011\ 0011_2 \\ 1001_2 \\ \hline 0101\ 0011\ 0011\ 0011\ 0011\ 0011\ 0011\ 1100_2 \end{array}$$

$C = 0{,}A6666678_{16} = 10*16^{-1}+6*16^{-2}+6*16^{-3}+6*16^{-4}+6*16^{-5}+6*16^{-6}+7*16^{-7}+8*16^{-8} = 0{,}65_{10}$

Изменение значения d:

$$+\begin{array}{r} 1001\ 0011\ 0011\ 0011_2 \\ 1001_2 \\ \hline 1001\ 0011\ 0011\ 1100_2 \end{array} \xrightarrow{\text{инверсия}} +\begin{array}{r} 0110\ 1100\ 1100\ 0011_2 \\ 1_2 \\ \hline 0110\ 1100\ 1100\ 0100_2 \end{array}$$

$d = -0{,}D988_{16} = -(13*16^{-1}+9*16^{-2}+8*16^{-3}+8*16^{-4}) = -0{,}849731_{10}$

Представление результата в памяти: $1001\ 0011\ 0011\ 1100_2 = 933C_{16}$

$$+\begin{array}{r} 1001\ 0011\ 0011\ 0011\ 0011\ 0011\ 0011\ 0011_2 \\ 1001_2 \\ \hline 1001\ 0011\ 0011\ 0011\ 0011\ 0011\ 0011\ 1100_2 \end{array}$$

инверсия

$$+\begin{array}{r} 0110\ 1100\ 1100\ 1100\ 1100\ 1100\ 1100\ 0011_2 \\ 1_2 \\ \hline 0110\ 1100\ 1100\ 1100\ 1100\ 1100\ 1100\ 0100_2 \end{array}$$

$d = -0{,}D9999988_{16} = 13*16^{-1}+9*16^{-2}+9*16^{-3}+9*16^{-4}+9*16^{-5}+9*16^{-6}+8*16^{-7}+8*16^{-8} = -0{,}849997_{10}$

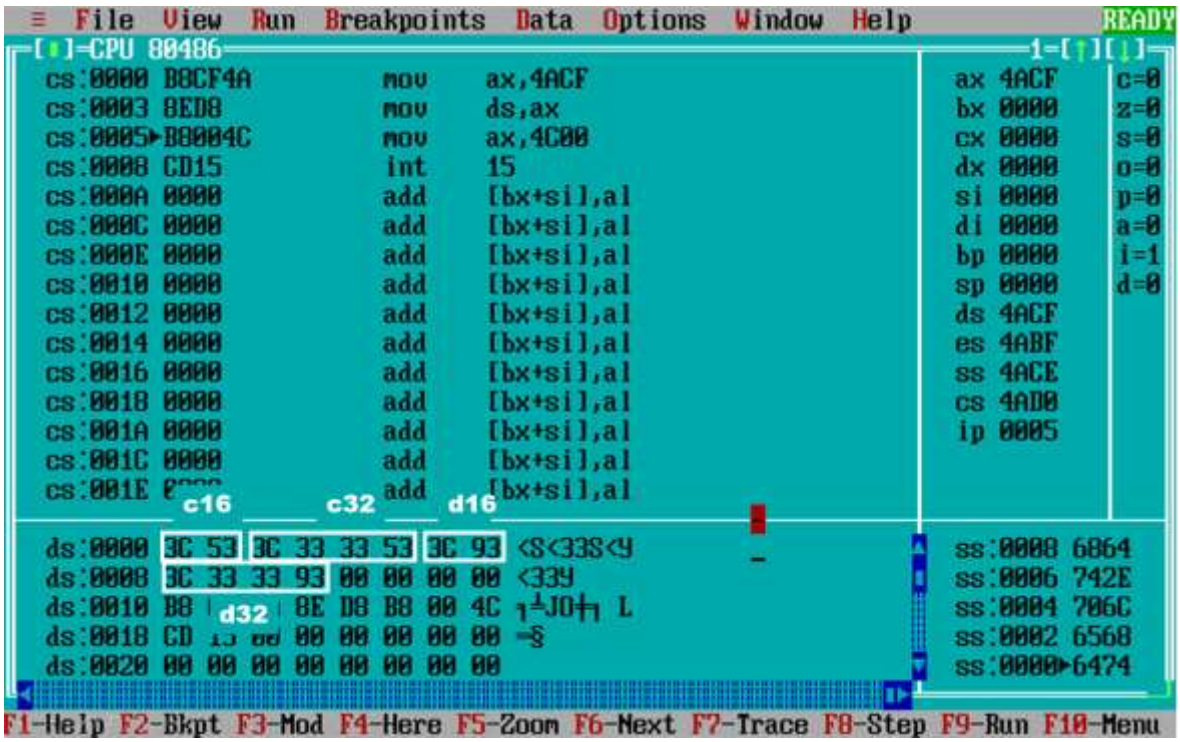
Представление результата в памяти: $1001\ 0011\ 0011\ 0011\ 0011\ 0011\ 0011\ 1100_2 = 9333333C_{16}$

*part1.asm – Блокнот

Файл Правка Формат Вид Справка

```
data segment
        c16 dw 533Ch
        c32 dd 5333333Ch

        d16 dw 933Ch
        d32 dd 9333333Ch
data ends

code segment
```

Представление в памяти:



9. Перевела числа f = 65,85 и g = -85,65 в двоичную систему счисления.

f = 65,85

65	16
64	4
1	

0,85*16 = 13,6
0,60*16 = 9,6
0,60*16 = 9,6
0,60*16 = 9,6
0,60*16 = 9,6
0,60*16 = 9,6
0,60*16 = 9,6
0,60*16 = 9,6
0,60*16 = 9,6

d = -85,65

85	16
80	5
5	

0,65*16 = 10,4
0,40*16 = 6,4
0,40*16 = 6,4
0,40*16 = 6,4
0,40*16 = 6,4
0,40*16 = 6,4
0,40*16 = 6,4
0,40*16 = 6,4
0,40*16 = 6,4

$f = 41,D9999A_{16} = 1000001,1101\ 1001\ 1001\ 1001\ 1001\ 1010_2$

$g = -55,A66666_{16} = -1010101,\ 1010\ 0110\ 0110\ 0110\ 0110\ 0110_2$

10. Представила f и g как операнды в прямом и дополнительном коде в процессорах разной разрядности.

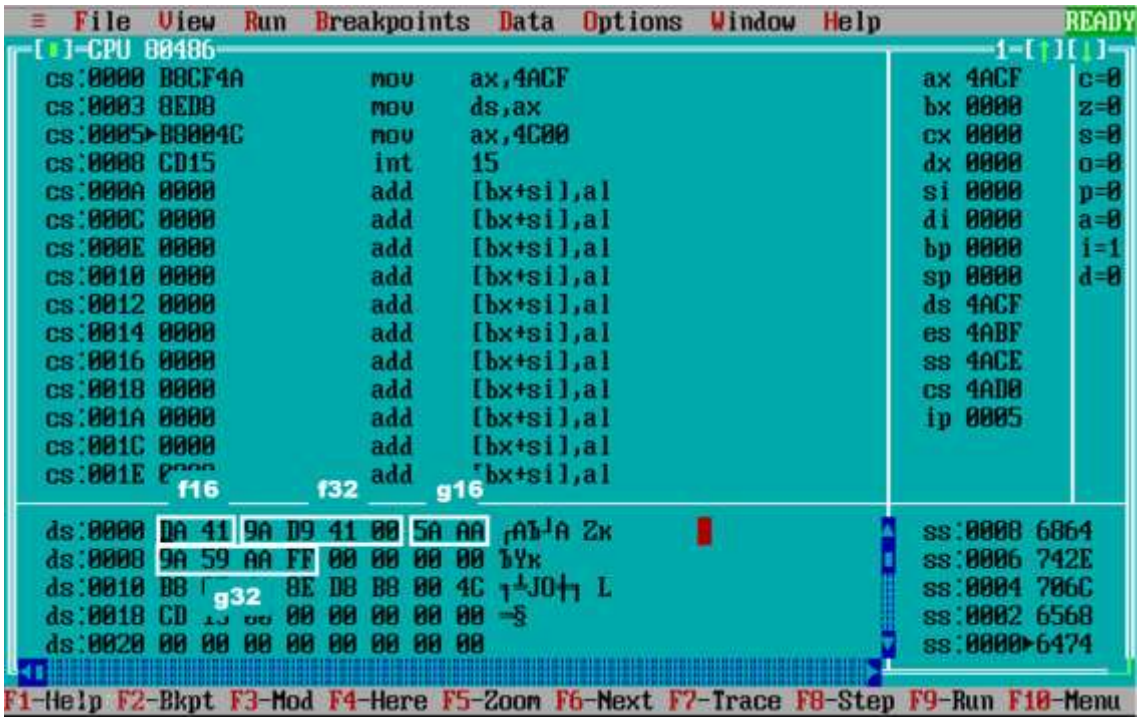
16-разрядный процессор
прямой код:
$[f]_1$ = 0100 0001 1101 1010 $[g]_1$ = 1101 0101 1010 0110
дополнительный код:
$[f]_2$ = 0100 0001 1101 1010 $[g]_2$ =1010 1010 0101 1010

32-разрядный процессор
прямой код:
$[f]_1$ = 0000 0000 0100 0001 1101 1001 1001 1010
$[g]_1$ = 1000 0000 0101 0101 1010 0110 0110 0110
дополнительный код:
$[f]_2$ = 0000 0000 0100 0001 1101 1001 1001 1010
$[g]_2$ = 1111 1111 1010 1010 0101 1001 1001 1010

11. Полученные представления чисел разместила в оперативной памяти. Для этого написала следующую программу:

```
Файл Правка Формат Вид Справка
data segment
        f16 dw 41DAh
        f32 dd 0041D99Ah

        g16 dw 0AA5Ah
        g32 dd 0FFAA599Ah
data ends

code segment
assume cs:code, ds:data, ss:nothing

start:          mov ax, data
                mov ds, ax

quit:           mov ax, 4c00h
                int 21
code ends
end start
```

Представление в оперативной памяти:



12. Увеличила в оперативной памяти каждого процессора значение младших байтов чисел на 1001_2.
    Изменение значения f:

$$\begin{array}{r} 0100\ 0001\ \ 1101\ 1010_2 \\ + \qquad\qquad\quad 1001_2 \\ \hline 0100\ 0001\ \ 1110\ 0011_2 \end{array}$$

$f = 41,E3_{16} = 4*16^{1}+1*16^{0}+14*16^{-1}+3*16^{-2} = 65,88672_{10}$

$$\begin{array}{r} 0000\ 0000\ 0100\ 0001\ \ 1101\ 1001\ 1001\ 1010_2 \\ + \qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad 1001_2 \\ \hline 0000\ 0000\ 0100\ 0001\ \ 1101\ 1001\ 1010\ 0011_2 \end{array}$$

$f = 41,D9A3_{16} = 4*16^{1}+1*16^{0}+13*16^{-1}+9*16^{-2}+10*16^{-3}+3*16^{-4} = 65,85014_{10}$

Изменение значения g:

$$\begin{array}{r} 1010\ 1010\ 0101\ 1010_2 \\ + \qquad\qquad 1001_2 \\ \hline 1010\ 1010\ 0110\ 0011_2 \end{array} \xrightarrow{\text{инверсия}} + \begin{array}{r} 0101\ 0101\ 1001\ 1100_2 \\ \qquad\qquad 1_2 \\ \hline 0101\ 0101\ 1001\ 1101_2 \end{array}$$

$g = -55,9D_{16} = -(5*16^{1}+5*16^{0}+9*16^{-1}+13*16^{-2}) = -85,61328_{10}$

Представление результата в памяти: $1010\ 1010\ 0110\ 0011_2 = AA63_{16}$

$$\begin{array}{r} 1111\ 1111\ 1010\ 1010\ 0101\ 1001\ 1001\ 1010_2 \\ + \qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad 1001_2 \\ \hline 1111\ 1111\ 1010\ 1010\ 0101\ 1001\ 1010\ 0011_2 \end{array}$$

инверсия

$$\begin{array}{r} 0000\ 0000\ 0101\ 0101\ 1010\ 0110\ 0101\ 1100_2 \\ + \qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad 1_2 \\ \hline 0000\ 0000\ 0101\ 0101\ 1010\ 0110\ 0101\ 1101_2 \end{array}$$

$g = -(55,A65D)_{16} = -(5*16^{1}+5*16^{0}+10*16^{-1}+6*16^{-2}+5*16^{-3}+13*16^{-4}) = -85,64986_{10}$

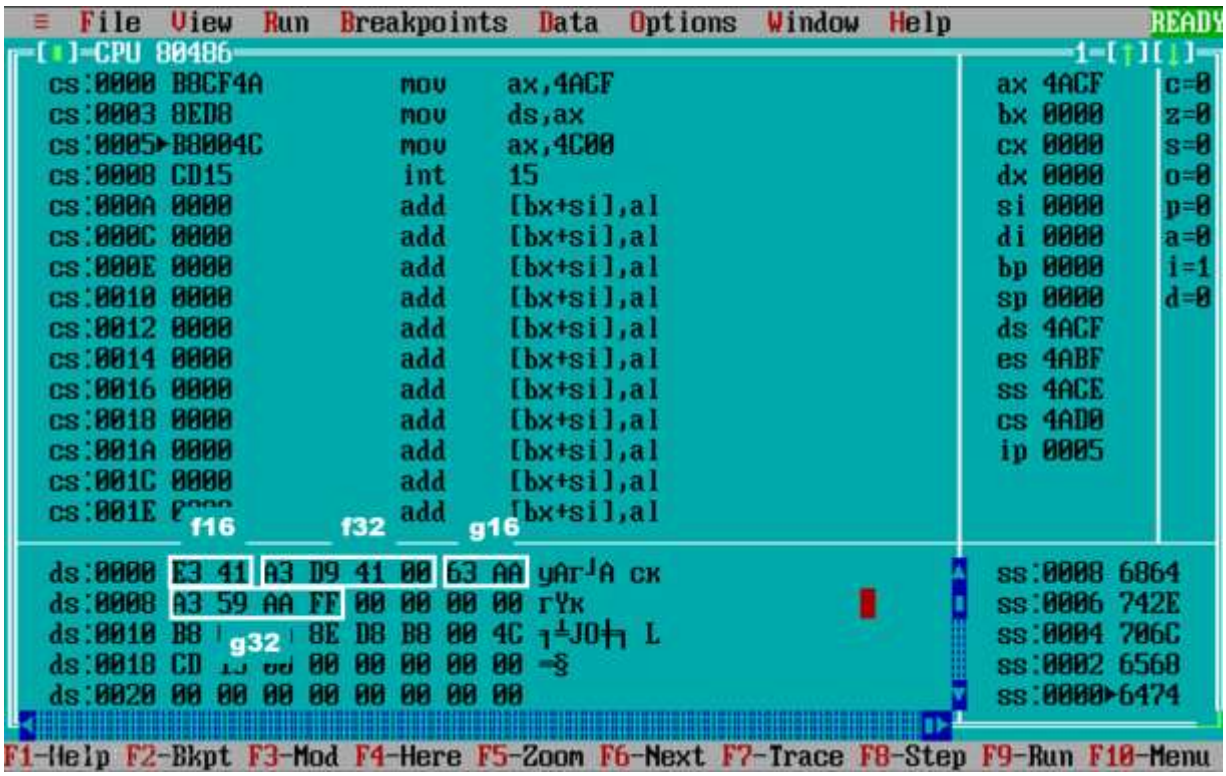
Представление результата в памяти: $1111\ 1111\ 1010\ 1010\ 0101\ 1001\ 1010\ 0011_2 = FFAA59A3_{16}$

Представление в памяти:

part1.asm – Блокнот

Файл Правка Формат Вид Справка

```
data segment
        f16 dw 41E3h
        f32 dd 0041D9A3h

        g16 dw 0AA63h
        g32 dd 0FFAA59A3h
data ends

code segment
```



Вывод: получила 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.