Министерство науки и высшего образования Российской Федерации  
Пензенский государственный университет  
Кафедра вычислительная техника

**ОТЧËТ**  
по лабораторной работе №8  
по дисциплине: «Арифметические и логические основы вычислительной техники»  
на тему «Умножение в цифровых процессорах»

Выполнил студент группы 22ВВП1:  
Беляев Д.

Приняли:  
Калиниченко Е. И.

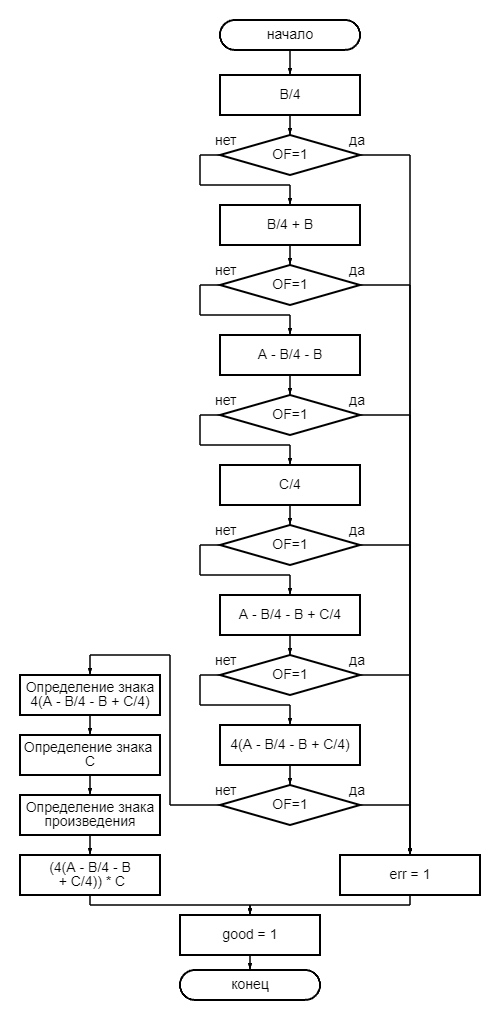
Пенза 2023

**Лабораторное задание:**

Написать на языке ассемблера программу вычисления выражения

**Y=(4\*(А−5/4\*В+С/4))\*C**. Процессор имеет разрядность – 16 бит. Исходные данные (значения переменных заданного выражения - целые, 16-битовые со знаком) располагаются в оперативной памяти, результат вычисления также поместить в оперативную память.

Общий алгоритм вычисления выражения приведен на **блок схеме**:



**Листинг Программы:**

data segment

;=============

A dw 10000d

B dw 6000d

C dw -9000d

Y1 dw ?

Y2 dw ?

good db ?

err db ?

mask dw 8000h

;=============

data ends

code segment

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

start: mov ax, data

mov ds,ax

;=============

mov good, 0

mov err, 0

mov bx, B

mov cl, 2

sar bx, cl; B/4

jo ALARM;

add bx, B; B/4 + B

jo ALARM

mov ax, A

sub ax, bx; A - B/4 - B

jo ALARM

mov bx, C

mov cl, 2

sar bx, cl; C/4

jo ALARM

add ax, bx; A - B/4 - B + C/4

jo ALARM;

mov cl, 2

sal ax, cl; 4(A - B/4 - B + C/4)

jo ALARM

; Умножение

mov bx, C

; проверка знака суммы

mov dx, mask

mov cx, dx

and dx, ax

jz PLUS\_LAST\_STEP

neg ax

mov dx, 1

PLUS\_LAST\_STEP:

; проверка знака C

and cx, bx

jz PLUS\_C

neg bx

mov cx, 1

PLUS\_C:

xor dx, cx

push dx

; Цикл умножения

xor dx, dx

mov cx, 15

MUL\_NUM:

rcr bx, 1

jnc a\_1

add dx, ax

a\_1:

rcr dx, 1

loop MUL\_NUM

rcr bx, 1

rcr dx, 1

rcr bx, 1

; Возврат знака

pop cx

test cl, 1

jz EXIT

not dx

neg bx

EXIT:

mov Y1, dx; старшая часть

mov Y2, bx; младшая часть

jmp ALLGOOD

ALARM:

mov err, 1

ALLGOOD:

mov good, 1

;=============

quit: mov ax,4c00h

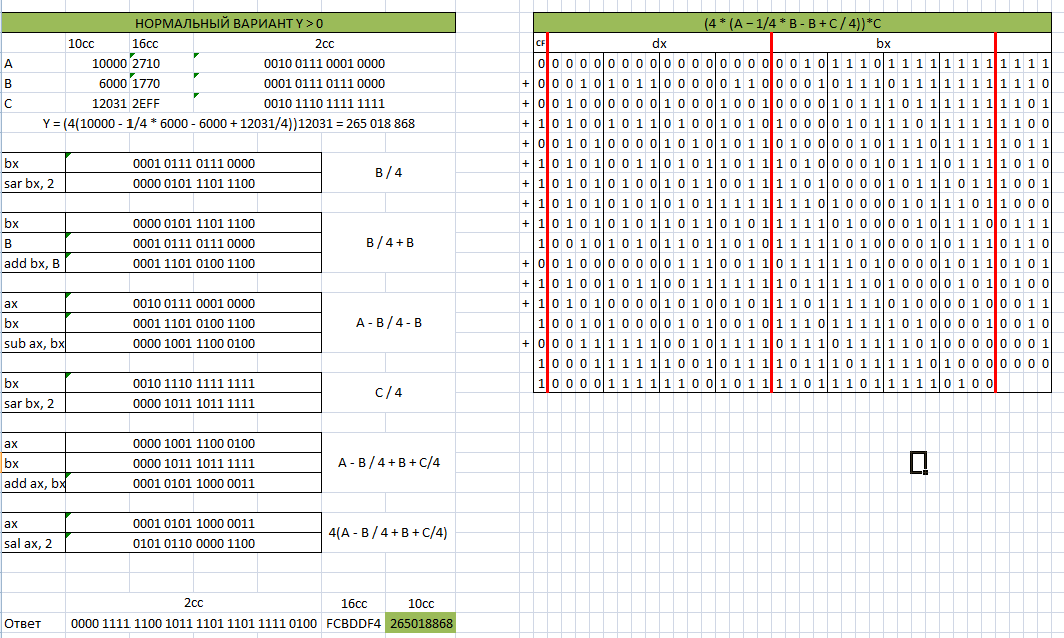
int 21

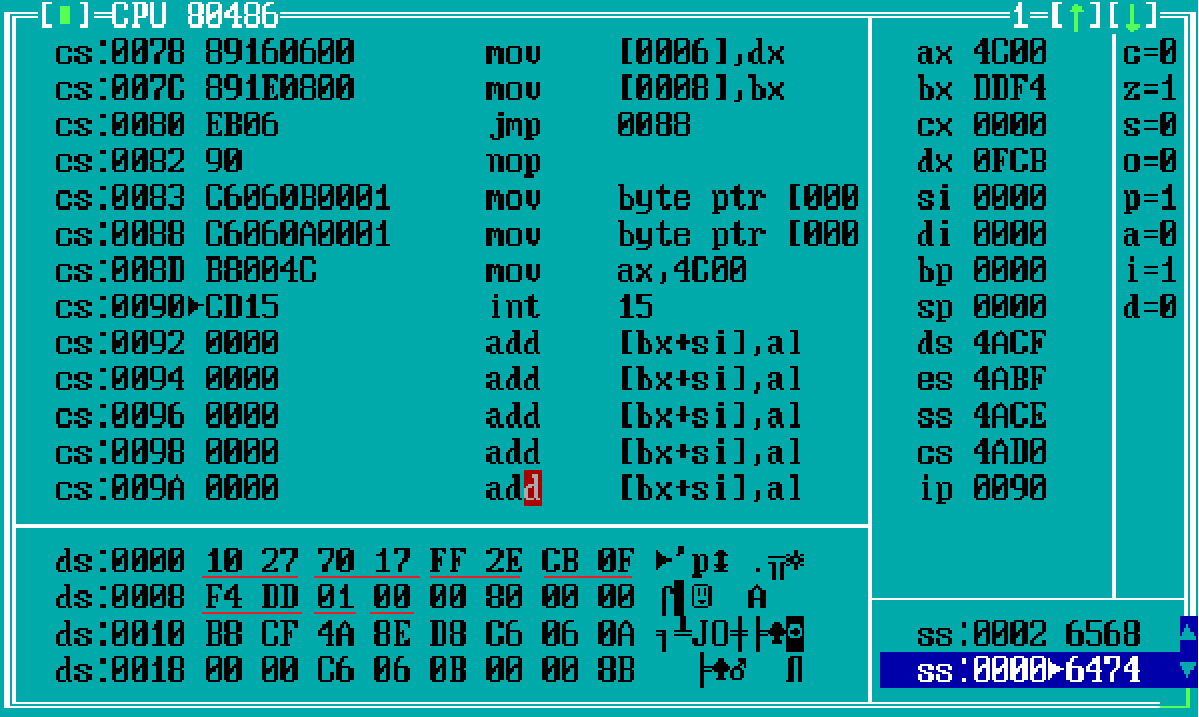
code ends

end start

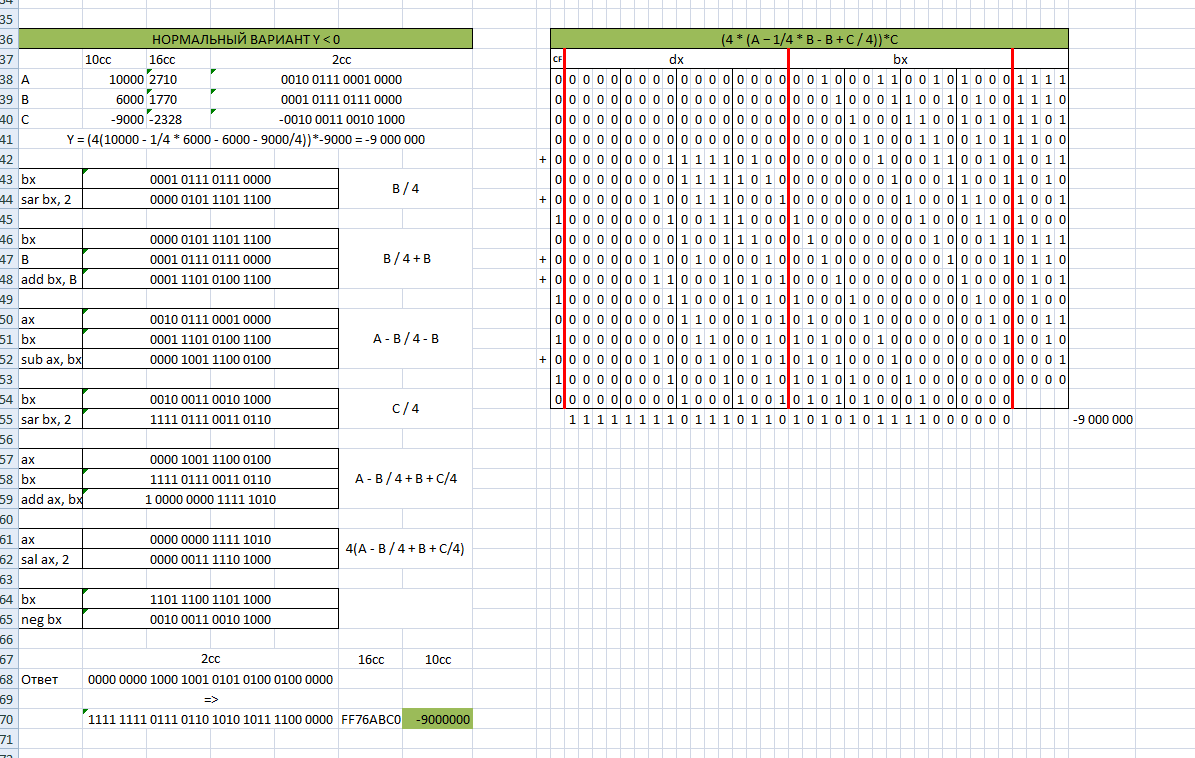
**Ход работы:**

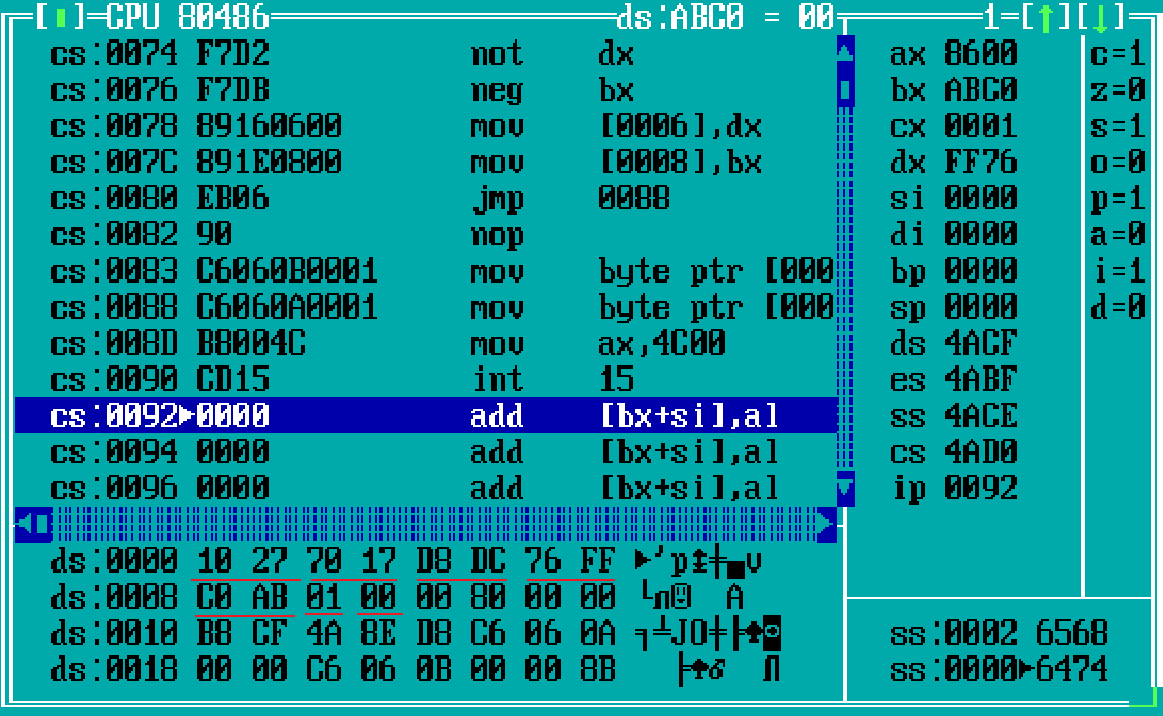
1. Нормальное решение, Y > 0

****

****

1. Нормальное решение, Y < 0

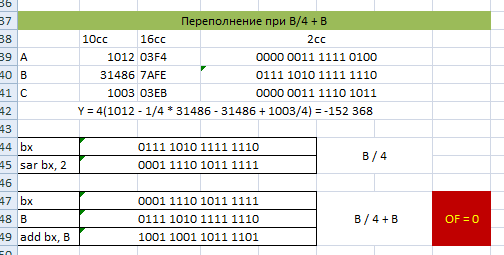
****

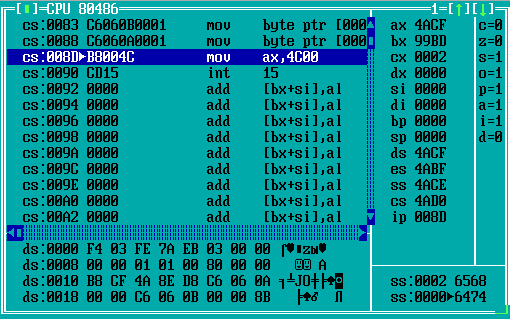
****

1. Переполнение при B/4

Переполнение невозможно, так как Флаг OF изменяется только в случае сдвига на 1 бит

1. Переполнение при B/4 + B

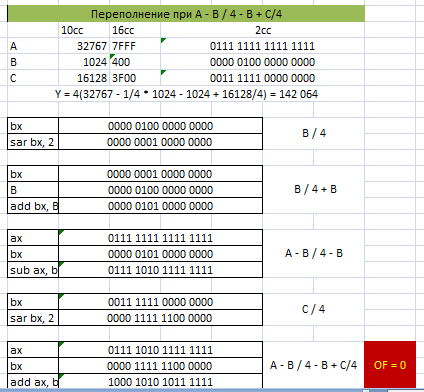
****

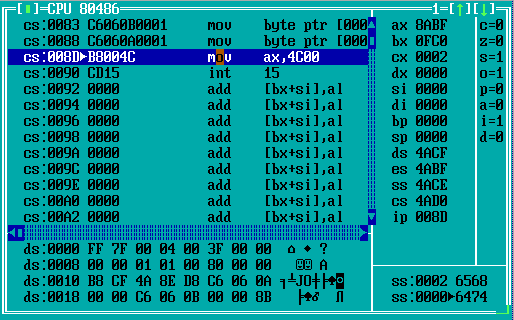
****

1. Переполнение при C/4

Переполнение невозможно, так как Флаг OF изменяется только в случае сдвига на 1 бит

1. Переполнение при A - B/4 - B + C/4

****

****

1. Переполнение при 4(A - B/4 - B + C/4)

Переполнение невозможно, так как Флаг OF изменяется только в случае сдвига на 1 бит

**Вывод:**

Создали программу на языке ассемблера для вычисления выражения

**Y=(4\*(А−5/4\*В+С/4))\*C**. Умножение было реализовано через операции сдвига, сложения и вычитания. Возможные исключения обработаны.