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

**ОТЧËТ**  
по лабораторной работе №7  
по дисциплине: «Арифметические и логические основы вычислительной техники»  
на тему «Поразрядные операции и операции сдвига в цифровых процессорах»

Выполнил студент группы 22ВВП1:  
Беляев Д.

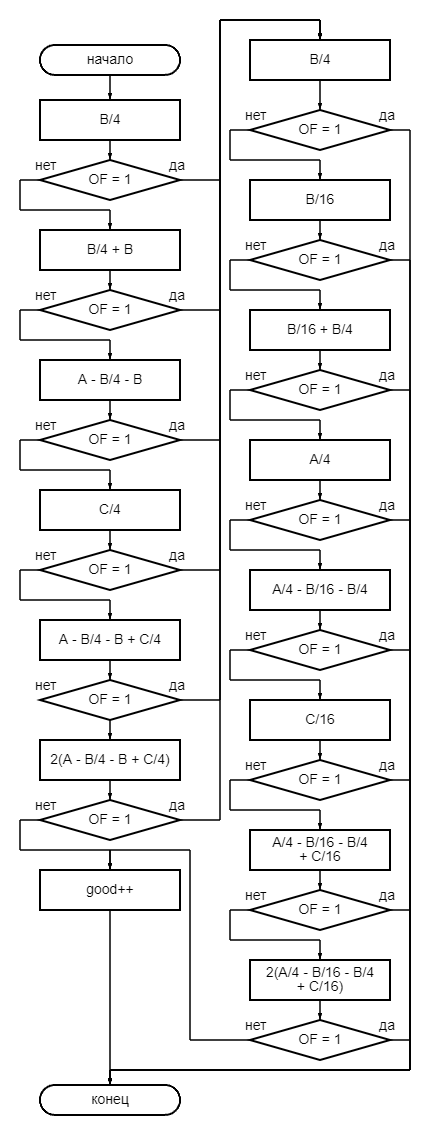
Приняли:  
Калиниченко Е. И.

Пенза 2023

**Лабораторное задание:**

Для процессора i8086 написать на языке ассемблера программу вычисления

выражения **Y = 4 \* (А − 5/4 \* В + С / 4)**. Формат переменных – 16 бит, целые со знаком (занимают одно слово). Умножение и деление на константы делать только с использованием операций сдвига, сложения и вычитания.

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

**Листинг Программы:**

data segment

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

A dw 64ECh

B dw 03F0h

C dw 0400h

Y dw ?

good db ?

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

data ends

code segment

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

start: mov ax, data

mov ds,ax

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

mov bx, B

mov cl, 2

sar bx, cl; B/4

;mov y, bx

jo ANOTHERWAY;

add bx, B; B/4 + B

mov y, bx

jo ANOTHERWAY

mov ax, A

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

mov y, ax

jo ANOTHERWAY

mov bx, C

mov cl, 2

sar bx, cl; C/4

jo ANOTHERWAY

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

mov y, ax;

jo ANOTHERWAY;

mov cl, 2

sal ax, cl; 2(A - B/4 - B + C/4)

jo ANOTHERWAY

jmp ALLGOOD

ANOTHERWAY:

mov bx, B

mov cl, 4

sar bx, cl; B/16

jo EXIT

mov y, bx

mov ax, B

mov cl, 2

sar ax, cl; B/4

jo EXIT

mov y, ax

add bx, ax; B/16 + B/4

jo EXIT

mov y, bx

mov ax, A

mov cl, 2

sar ax, cl

jo EXIT

mov y, ax

sub ax, bx; A/4 - B/16 - B/4

jo EXIT

mov y, ax

mov bx, C

mov cl, 4

sar bx, cl; C/16

jo EXIT

mov y, bx

add ax, bx; // A/4 - B/16 - B/4 + C/16

jo EXIT

mov y, ax

mov cl, 4

sal ax, cl; // 16(A/4 - B/16 - B/4 + C/16)

jo EXIT

ALLGOOD:

inc good;

mov y, ax;

EXIT:

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

quit: mov ax,4c00h

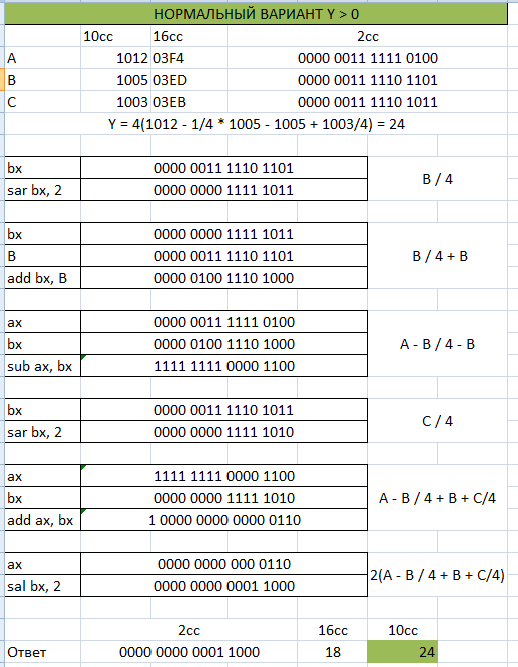
int 21

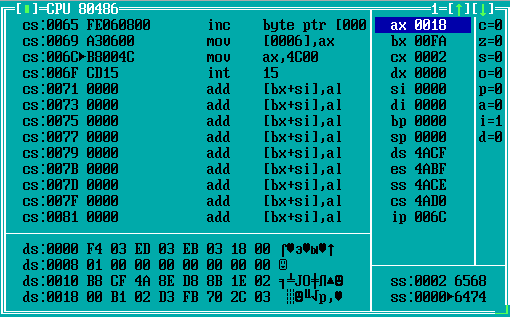
code ends

end start

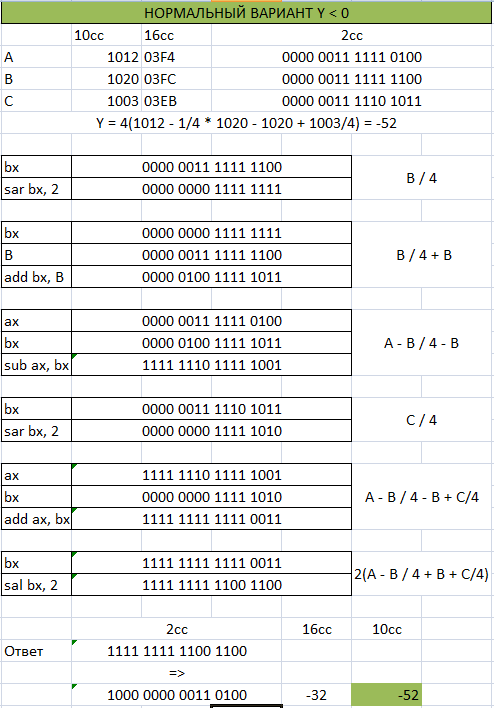
**Ход работы:**

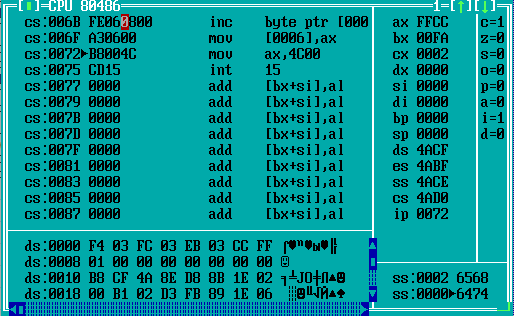
1. Нормальное решение, Y > 0

****

****

1. Нормальное решение, Y < 0

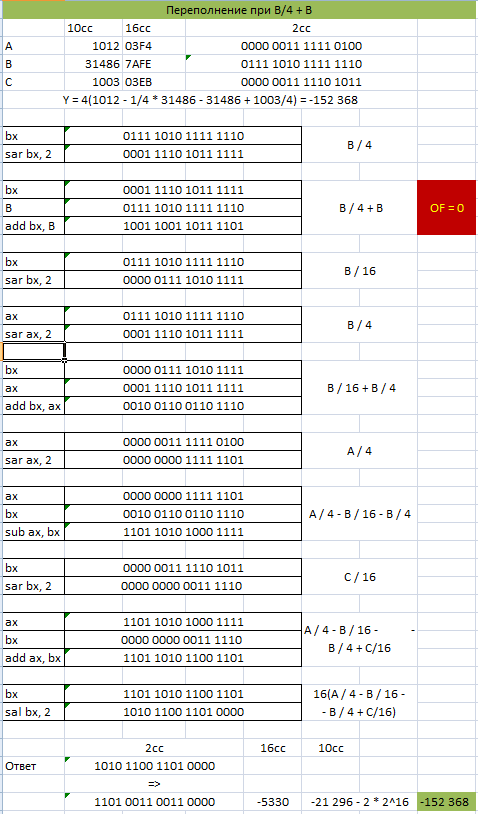
****

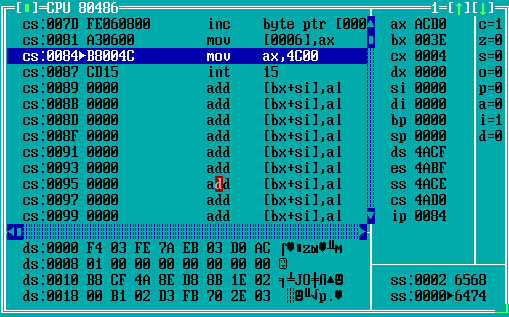
****

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

****

****

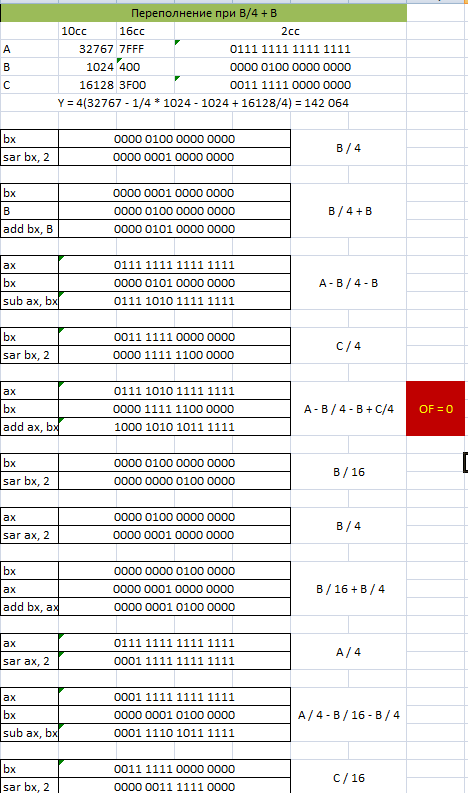
1. Переполнение при A - B/4 – B

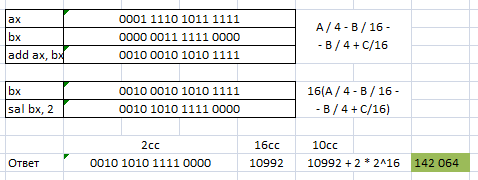
Переполнение невозможно, т.к. переполнение при вычитании возможно только при одинаковых отрицательных знаках. Число А не будет отрицательным по условию, на шаге 4-5 отрицательное число получиться не может, так как если бы B > 1000

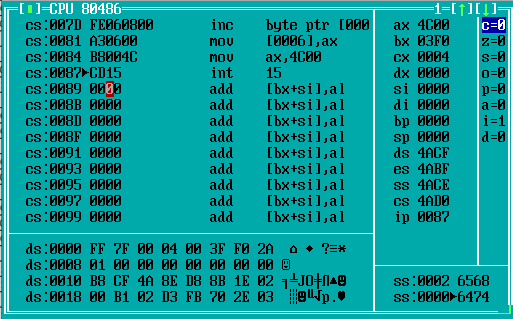
1. Переполнение при C/4

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

1. Переполнение при A - B/4 - B + C/4

****

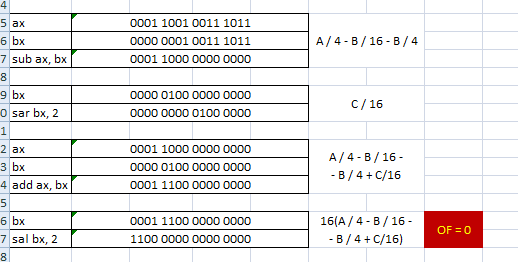
****

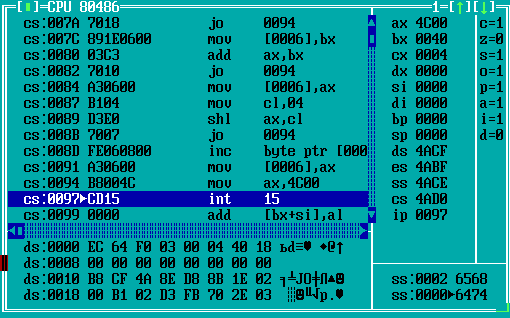
****

1. Переполнение при 2(A - B/4 - B + C/4)

Если произошло переполнение на этом шаге, то переполнение будет в любом случае, т.к. на этом шаге число сдвигается пропорционально





****

**Вывод:**

Создали программу на языке ассемблера для вычисления выражения

**Y = 4 \* (А − 5/4 \* В + С / 4)**. Умножение и деление на константы реализовано операциями сдвига, сложения и вычитания. Исключение – переполнение - обработано для пересчета и получения верного результата.