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

**ОТЧЕТ**

по лабораторной работе №2

по дисциплине «ЭВМ и переферийные устройства»

на тему «Работа процессора в линейных программах»

Выполнили:

студенты группы 22ВВП1

Хоссейни Нежад С.А.С.М.

Захаров А. С.

Приняли:

Никишин К.И.

Патунин Д.В.

Пенза 2025

**Цель работы**

Изучение способов адресации и арифметико-логических и посылочных операций.

**Задания**

Задание 1: Составить программу вычисления по формуле (см. варианты). Операнды и результат разместить в памяти. Один из операндов A,B, С адресовать, используя базовую адресацию, остальные - базовую со смещением, константу адресовать непосредственно, результат - прямо.



Задание 2: Используя листинг программы, полученный при выполнении задания 1, разобрать первой записанной в сегменте кода команды add (четные варианты) или команды sub (нечетные варианты) и записать бинарные коды всех полей ее формата.

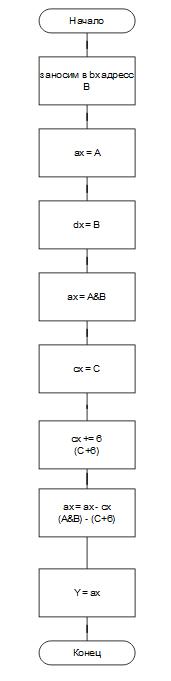
**Ход работы**

Задание 1

Написали программу удовлетворяющую условию:

* Y=A&B-(C+6)= 7&(-9) - (14 + 6) = - 13
* Базовая адрессация для операнда B
* Базовая адрессация со смещением для операндов A и C
* Непосредственная адрессация для константы – ‘6’
* Прямая адрессация для результата

Блок-схема:



Листинг:

stseg segment para stack

dw 16 dup(?)

stseg ends

dseg segment para

B dw -9

A dw 7

C dw 14

Y dw ?

dseg ends

cseg segment para

lab2 proc far

assume cs:cseg, ds:dseg, ss:stseg

push ds

mov ax, dseg

mov ds, ax

mov bx, offset B

mov ax, [bx+2]

mov dx, [bx]

and ax, dx

mov cx, [bx+4]

add cx, 6

sub ax, cx

mov Y, ax

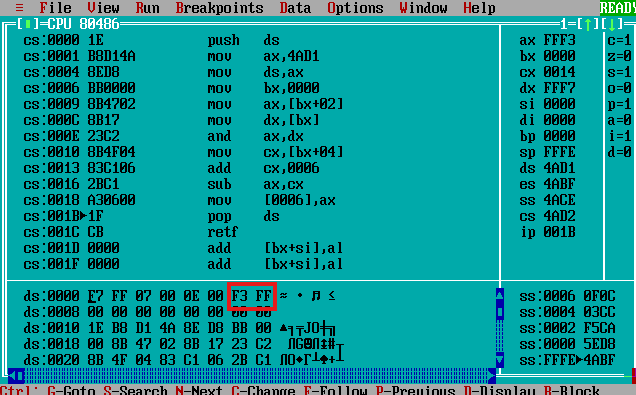
pop ds

ret

lab2 endp

cseg ends

end lab2

****

Преобразуем в десятичный вид:

Y в доп коде: FFF316 = 1111 1111 1111 00112

инвертируем: 1000 0000 0000 11002

Прибавляем 1: 1000 0000 0000 11012 = -1310

Результаты ручного и программного метода совпали.

Задание 2

В нашем случае код первой операции sub это 2BC1, где:

**2B** в шестнадцатеричном виде → **0010 1011** в двоичном виде

**C1** в шестнадцатеричном виде → **1100 0001** в двоичном виде

### Первый байт: 2B

* 2B — это opcode инструкции.  
  В 8086 этот opcode с суффиксом /r соответствует инструкции  
  SUB r16, r/m16, то есть вычитает из регистра-операнда (dest) значение, хранящееся в регистре или по адресу (source).

### Второй байт: C1

* C1 в шестнадцатеричном виде → двоичный вид: 1100 0001  
  Разобьём его по полям:
  + mod (бит7-6): 11  
    Значение 11 означает, что оба операнда – регистры.
  + reg (бит5-3): 000  
    Код регистра 000 соответствует регистру AX.
  + r/m (бит2-0): 001  
    Код 001 соответствует регистру CX.

**Вывод**

Изучили способы адресации и арифметико-логические и посылочные операций.