

ГУАП

КАФЕДРА № 42

ОТЧЕТ  
ЗАЩИЩЕН С ОЦЕНКОЙ \_\_\_\_\_  
ПРЕПОДАВАТЕЛЬ

ассистент

\_\_\_\_\_  
должность, уч. степень, звание

\_\_\_\_\_  
подпись, дата

И.Д. Свеженин

\_\_\_\_\_  
инициалы, фамилия

## ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №2

Представление данных. Арифметико-логические операции

по курсу: Архитектура ЭВМ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. № 4329

\_\_\_\_\_  
подпись, дата

Д.С. Шаповалова

\_\_\_\_\_  
инициалы, фамилия

Санкт-Петербург 2025

## 1. Цель работы:

Изучение архитектуры МП Intel 8086, изучение структуры простейшей ассемблерной программы, ознакомление с системой арифметико-логических команд процессора, организация вычислений на языке ассемблера.

## 2. Задание:

1. Определить исходные данные в соответствии с номером варианта.
2. Перевести значения величин X1-X4 в шестнадцатеричную систему счисления.
3. Провести трассировку заданного алгоритма с использованием заданных исходных данных.
4. Составить программу заданного алгоритма в мнемокодах.
5. Оформить отчет по лабораторной работе.
6. В учебной лаборатории проверить результаты выполнения программы в программе-отладчике, сравнивая их с результатами ручной трассировки алгоритма.

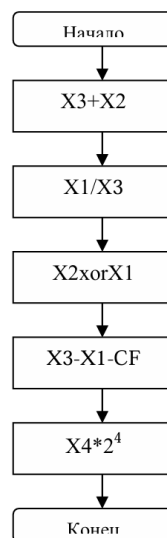


Рисунок 1.1 – Вариант 7.

Значения исходных данных, которые должны храниться в сегменте данных, определяются выражениями:

$$X1 = N^{\circ}B * (-1)^{N^{\circ}B} = -17$$

$$X2 = (-1)^{N^{\circ}B+1} * (N^{\circ}\Gamma * N^{\circ}B) = 73593$$

$$X3 = (-1)^{N^{\circ}B+2} * (N^{\circ}\Gamma * N^{\circ}B + N^{\circ}\Gamma) = -77922$$

$$X4 = (-1)^{N^{\circ}B+3} * N^{\circ}\Gamma = 4329$$

где  $N^{\circ}B$  – номер варианта,  $N^{\circ}\Gamma$  – номер группы.

### 3. Ход работы:

#### 1. Определение исходных данных:

$$X1 = N^oB * (-1)^{N^oB} = 17 * (-1)^{17} = 17 * (-1) = -17$$

$$X2 = (-1)^{N^oB+1} * (N^o\Gamma * N^oB) = (-1)^{(17+1)} * (29 * 17) = 493$$

$$X3 = (-1)^{N^oB+2} * (N^o\Gamma * N^oB + N^o\Gamma) = (-1)^{(17+2)} * (29 * 17 + 29) = -522$$

$$X4 = (-1)^{N^oB+3} * N^o\Gamma = (-1)^{(17+3)} * 29 = (1) * 29 = 29$$

#### 2. Перевод величин X1-X4 в шестнадцатеричную СС:

$$X1 = -17 \rightarrow \text{FFEFh}$$

$$X2 = 493 \rightarrow \text{01EDh}$$

$$X3 = -522 \rightarrow \text{FDF6h}$$

$$X4 = 29 \rightarrow \text{1Dh}$$

#### 3. Ручная трассировка алгоритма с использованием исходных данных

$$\text{Шаг 1: } X3 + X2 = -29$$

$$\text{Шаг 2: } X1 / X3$$

$$-17$$

$$\div -522$$

---

$$\text{Частное} = 0, \text{Остаток} = 17$$

$$\text{Результат: } AX = 0, DX = 17 = \text{FFEFh}$$

$$\text{Шаг 3: } X2 \text{ xor } X1$$

$$493 \text{ (00000000111101101)}$$

$$\text{XOR}$$

$$-17 \text{ (1111111111101111)}$$

---

$$-510 \text{ (11111111000000010)} = \text{FE02h}$$

$$\text{Результат:}$$

$$\text{Шаг 4: } X3 - X1 - CF$$

$$-522$$

$$- -17$$

$$- 0 \text{ (CF=0)}$$

---

$$-505$$

Шаг 5:  $X4 * 2^4 = 29 \times 16 = 464 = 01D0h$

Операция	Результат	Шестнадцатеричный
$X3 + X2$	-29	FFE3h
$X1 / X3$	0 (AX)	0000h
$X2 \text{ xor } X1$	-510	FE02h
$X3 - X1 - CF$	-505	FE07h
$X4 * 2^4$	464	01D0h

После каждой операции результат сохраняется в регистре AX (в случае деления, частное в AX, остаток в DX).

#### *4. Алгоритм в мнемокодах*

```
.MODEL SMALL
```

```
Sseg SEGMENT STACK
```

```
    DB 256 DUP (?)
```

```
Sseg ENDS
```

```
Dseg SEGMENT
```

```
    X1 DW -17
```

```
    X2 DW 73593
```

```
    X3 DW -77922
```

```
    X4 DW 4329
```

```
Dseg ENDS
```

```
Cseg SEGMENT
```

```
    ASSUME CS:Cseg, DS:Dseg, SS:Sseg
```

```
Main PROC FAR
```

```
    PUSH DS
```

```
    XOR AX, AX
```

```
    PUSH AX
```

```
    MOV AX, Dseg
```

```
    MOV DS, AX
```

```
    MOV AX, X3
```

```
    ADD AX, X2
```

```
    MOV AX, X1
```

```
    CWD
```

```
    IDIV X3
```

```
MOV AX, X2
XOR AX, X1

MOV AX, X3
SBB AX, X1

MOV AX, X4
MOV CL, 4
SHL AX, CL

RET
Main ENDP

Cseg ENDS
END Main
```

#### 4. Вывод:

В ходе лабораторной работы были успешно освоены основы программирования на языке ассемблера для процессора Intel 8086. На практике изучены различные арифметико-логические операции, включая сложение, вычитание, умножение, деление, побитовые операции и сдвиги.

Особое внимание было уделено работе с разными форматами данных - байтовыми и словными операндами, а также правилам выполнения операций с учетом знака чисел. Экспериментально подтверждена важность контроля флагов процессора, особенно флага переноса (CF), который критически влияет на результаты операций с многобайтными числами.