МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

#### ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ

БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет автоматики и вычислительной техники

Кафедра радиоэлектронных средств

Отчет по дисциплине

«Цифровые устройства и микропроцессоры»

Лабораторная работа №2

«ПРИНЦИПЫ  
ВЫПОЛНЕНИЯ КОМАНД ВЕТВЛЕНИЯ, ОРГАНИЗАЦИЯ  
ЦИКЛОВ И ПОДПРОГРАММ»

Вариант №18

|  |  |  |
| --- | --- | --- |
| Выполнил: студент гр. ИНБб-3301-01-00 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Д.С.Миклин |
|  |  |  |

#### Киров 2023

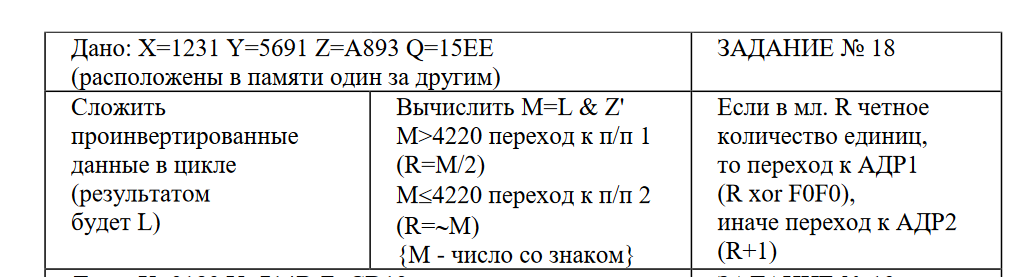
**Цель работы:** изучение принципов выполнения команд ветвления,

организации циклов и подпрограмм микропроцессоров с архитектурой x86.

**Ход работы:**

**1**.Исходные данные представлены в таблице 1.

Таблица 1 – Исходные данные



**2**. Сложить проинвертированные данные в цикле

X= 01231h

Y= 05691h

Z= 0A893h

Q=015EEh

Все переменные были помещены в массив, чтобы использовать цикл.

Скриншот выполнения данного действия в программе

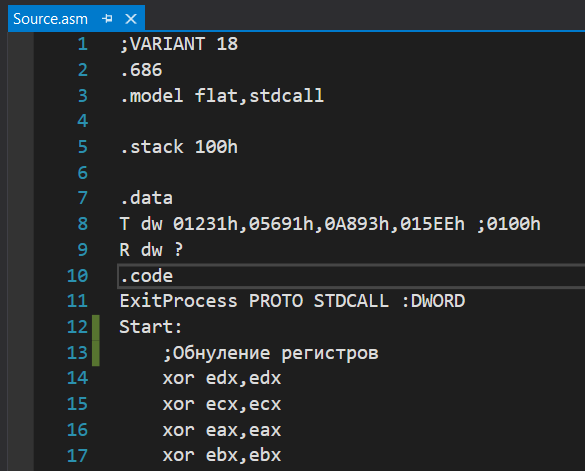
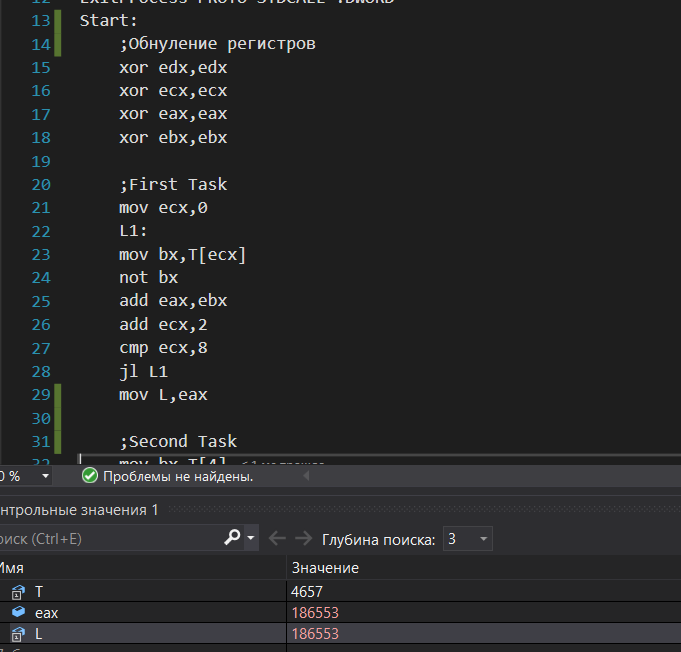
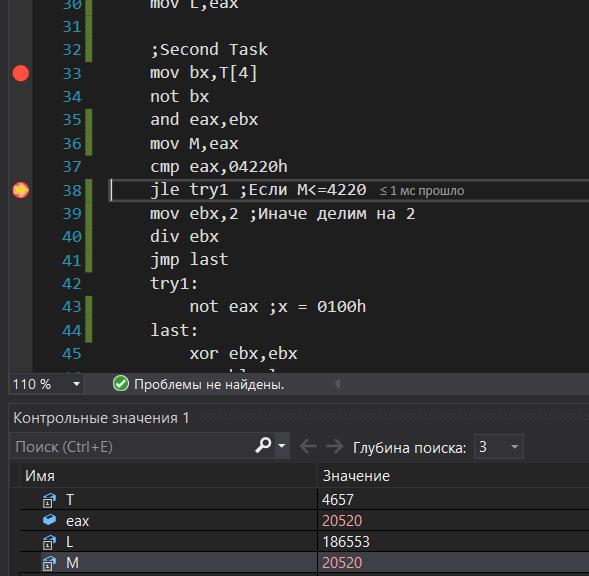


Рисунок 1 - скриншот массива

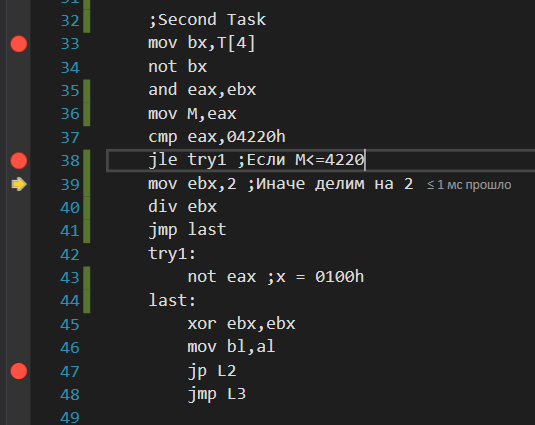
После инвертирования и сложения всех переменных получаем значение L = 186553



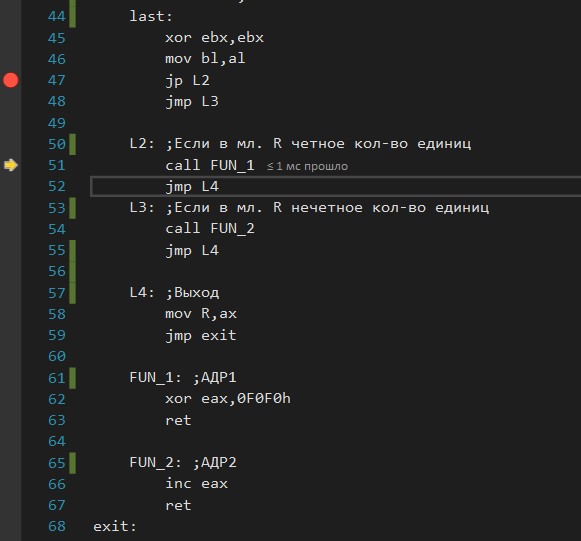
После выполнения M=L & Z' получаем M = 20520



Так как 20520 = 05028h, то произойдет переход к п/п1



В младшем разряде R четное кол-во единиц, следовательно произойдет переход к АДР1.



R на выходе равняется 55524.

***Вывод:*** изучены принципы выполнения команд ветвления,

организации циклов и подпрограмм микропроцессоров с архитектурой x86.

**3. Код программы:**

;VARIANT 18

.686

.model flat,stdcall

.stack 100h

.data

T dw 01231h,05691h,0A893h,015EEh ;0100h

R dw ?

L dd ?

M dd ?

.code

ExitProcess PROTO STDCALL :DWORD

Start:

;Обнуление регистров

xor edx,edx

xor ecx,ecx

xor eax,eax

xor ebx,ebx

;First Task

mov ecx,0

L1:

mov bx,T[ecx]

not bx

add eax,ebx

add ecx,2

cmp ecx,8

jl L1

mov L,eax

;Second Task

mov bx,T[4]

not bx

and eax,ebx

mov M,eax

cmp eax,04220h

jle try1 ;Если M<=4220

mov ebx,2 ;Иначе делим на 2

div ebx

jmp last

try1:

not eax ;x = 0100h

last:

xor ebx,ebx

mov bl,al

jp L2

jmp L3

L2: ;Если в мл. R четное кол-во единиц

call FUN\_1

jmp L4

L3: ;Если в мл. R нечетное кол-во единиц

call FUN\_2

jmp L4

L4: ;Выход

mov R,ax

jmp exit

FUN\_1: ;АДР1

xor eax,0F0F0h

ret

FUN\_2: ;АДР2

inc eax

ret

exit:

Invoke ExitProcess,1

End Start