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

«Техника микропроцессорных систем»

Лабораторная работа №2

«ПРИНЦИПЫ ВЫПОЛНЕНИЯ КОМАНД ВЕТВЛЕНИЯ, ОРГАНИЗАЦИЯ ЦИКЛОВ И ПОДПРОГРАММ»

Вариант №14

|  |  |  |
| --- | --- | --- |
| Выполнил: студент группы ИКТб– 31 |  | А.В. Волков |
|  |  |  |
| Проверил: доцент кафедры РЭС |  | М.А. Земцов |

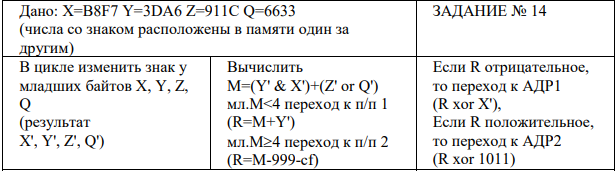
#### Киров 2022

**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**.Текст программыпредставлен ниже:

.686

.model flat,stdcall

.stack 100h

.data

arr dw 0B8F7h,03DA6h,0911Ch,06633h ; массив исходных данных

m dw 0000000010000000b ; обьявление маски для инверсии младших байтов у X Y Z Q

.code

ExitProcess PROTO STDCALL :DWORD

Start:

xor eax,eax ;очистка регистра EAX

xor ebx,ebx ;очистка регистра EBX

xor ecx,ecx ;очистка регистра ECX

xor edx,edx ;очистка регистра EDX

lea esi,[arr] ; для вычисления адреса

lea edi,[arr] ; для вычисления адреса

mov cx,4 ; указываем в регистре ECX 4

mov bx,m ; записал маску в bx

@cycle: ; начало цикла

lodsw ; прочитал число из строки

xor ax,bx ; умножил число на маску, которая находится в bx

stosw ; записал число в массив

loop @cycle ; конец цикла

lea esi,[arr]

mov edx,eax ; значение в регистре EAX в регистр EDX Q

lodsw

mov edi,eax ; значение в регистре EAX в регистр EDI X

lodsw

mov ebx,eax ; значение в регистре EAX в регистр EBX Y

lodsw

mov ecx,eax ; значение в регистре EAX в регистр ECX Z

mov eax,edi ; значение в регистре EAX в регистр EDI X

and eax,ebx ; X & Y

or ecx,edx ; Z or Q

add eax,ecx ; сложение результатов (М)

mov ecx,4

xor edx,edx ; очищение EDX

mov dl,al ; значение младшего байта записывается в EDX

cmp edx,ecx ; условие (мл M < 4)

jb pp1

jae pp2

pp1:

mov ecx,4h ; значение 4 записывается в регистр ECX

add eax,ebx ; выполнение 1 условия

;mov eax,-2 ; для проверки перехода к adr1

jmp foma

jmp exit

pp2:

mov edx,999h ; значение 999 записывается в регистр EDX

sub eax,edx ; вычитается из M значение 999

mov edx,eax ; значение в регистре EAX в регистр EDX

xor eax,eax ; очищение регистра EAX

xor ecx,ecx ; очищение регистра EСX

lahf ; запись значений флагов в регистр EAX

mov cl,ah ; значение в регистре EAX в регистр ECX

shl cl,7 ; сдвиг влево на 7 битов

shr cl,7 ; сдвиг вправо на 7 битов

sub edx,ecx ; вычитание из М значение флага CY

mov eax,edx ; значение в регистре EDX в регистр EAX

jmp foma

foma:

mov ebx,0

cmp eax,ebx ; сравнение (R < 0)

jl adr1

jg adr2

adr1:

xor eax,edi

jmp exit

adr2:

xor eax,1011h

exit:

Invoke ExitProcess,1

End Start

**3**. Расчет вручную.

В цикле изменить знак у младших байтов X, Y, Z, Q (результат X', Y', Z', Q')

X= 10111000 11110111=B8F7

Y= 00111101 10100110=3DA6

Z= 10010001 00011100=911C

Q= 01100110 00110011=6633

Тогда

X`= 10111000 01110111= B877

Y`= 00111101 00100110 = 3D26

Z`= 10010001 10011100 = 919C

Q`= 01100110 10110011=66B3

Вычислить M=(Y' & X')+(Z' or Q')

(Y`&X`)=3826

Z`orQ`=F7BF

M=3826+F7BF=12FE5 = 10010111111100101

мл. М =E5=11100101,

M>4, значит переходим к п/п2

R=M-999-cf = 1264C

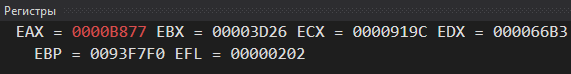
Cf=0

R>0, значит переход к АДР2

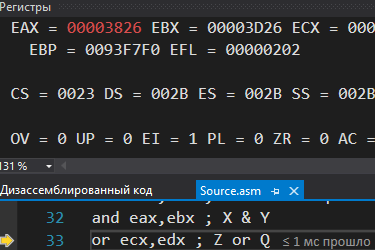
R xor 1011 = 1365D

Скриншоты состояния регистров во время выполнения программы представлены ниже:

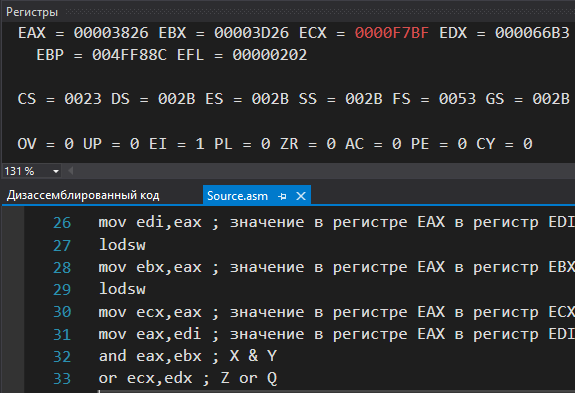
Переменные, измененные в цикле



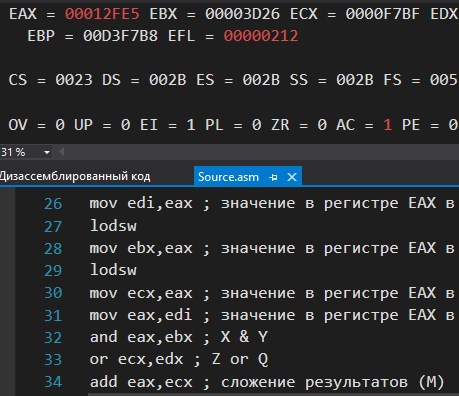
(X`&Y`)=3826



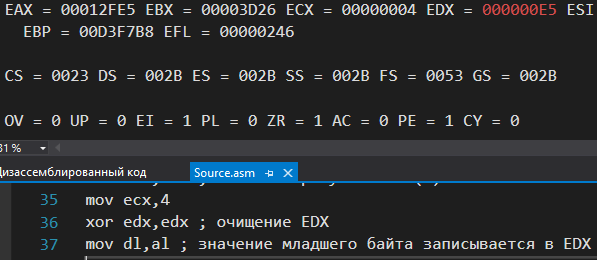
(Z' or Q')=F7BF



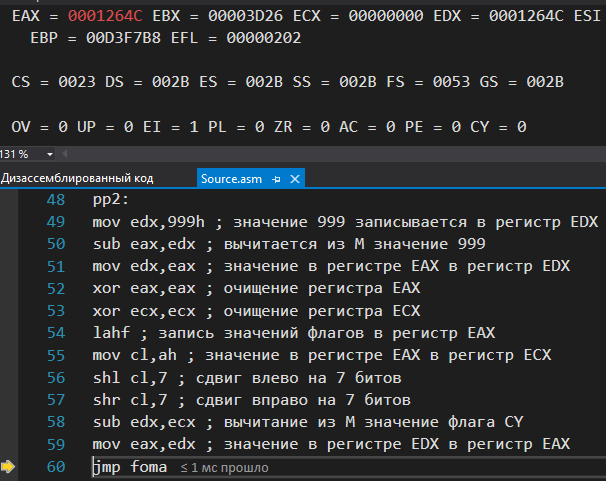
M=(Y' & X')+(Z' or Q')=12FE5= 1 0010111111100101



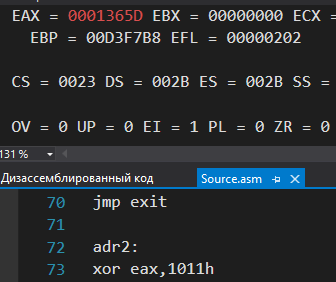
Младший байт M=E5



Выполнение п/п2:



Выполнение АДР2:



**Верификация программы:**

Для верификации программы используем другие входные данные:

X= AF82

Y= C386

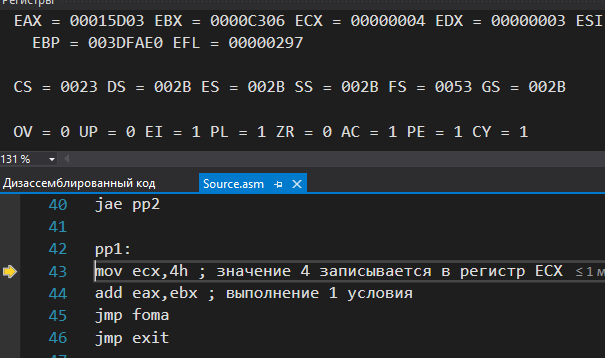
Z= DA81

Q= 8A80

Младший байт млM<4

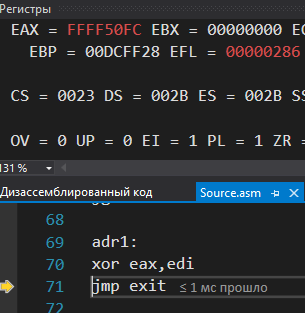
3<4

Выполнение п/п1



Для проверки АДР1 установим отрицательное значение eax вручную командой mov eax, -2.

Далее, соответственно переход к АДР1



**Вывод:** в ходе лабораторной работы были изучены команды: cmp, lea, lodsw, stosw, так же был изучен принципы работы цикла и условий перехода, получены навыки программирования на языке assembler.