**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ**

**Федеральное государственное бюджетное образовательное**

**учреждение высшего образования**

**«Чувашский государственный университет имени И.Н. Ульянова»**

**Факультет информатики и вычислительной техники**

**Кафедра математического и аппаратного обеспечения информационных систем**

Отчет по дисциплине «ЭВМ и периферийные устройства»

Лабораторная работа 1

На тему «Использование макрокоманд»

Выполнил:

студент группы ИВТ-11-21

Еремеев Сергей А.

Принял: Андреева А. А.

Чебоксары, 2023

Задание:

Макрокоманда аналогичная оператору IF в Фортране для проверки числа в аккумуляторе. Eсли число отрицательное, то управление передать ячейке NEG, если оно равно 0, управление передать ячейке ZER, если же число положительное, управление передать ячейке POS.

Макрокоманда представляет собой строку, содержащую некоторое символическое имя — имя макрокоманды, предназначенную для того, чтобы быть замещенной одной или несколькими другими строками. Имя макрокоманды может сопровождаться параметрами. Для написания макрокоманды вначале необходимо задать ее шаблон-описание, который называют макроопределением.

Синтаксис макроопределения следующий:

ИмяМакрокоманды macro СписокФормальныхАргументов  
ТелоМакроопределения  
endm

Макроопределение обрабатывается транслятором следующим образом. Для того чтобы использовать описанное макроопределение в нужном месте программы, оно должно быть активизировано с помощью макрокоманды указанием следующей синтаксической конструкции:

ИмяМакрокоманды СписокФактическихАргументов

Код программы имеет вид:

Myif macro i, neg\_label, zer\_label, pos\_label

cmp i, 0 ; Сравниваем число с нулем

jl neg\_label ; Если число меньше нуля, переходим к метке NEG

jz zer\_label ; Если число равно нулю, переходим к метке ZER

jmp pos\_label ; Если число больше нуля, переходим к метке POS

endm

sseg segment stack 'stack'

dw 256 dup(?)

sseg ends

data segment

x db 0 ; Здесь хранится введенное число

neg\_msg db 10,13,' x < 0','$'

zer\_msg db 10,13,'x = 0','$'

pos\_msg db 10,13,'x > 0','$'

data ends

code segment

assume cs:code, ds:data, ss:sseg

start:

mov ax, data ; Настройка сегментного регистра DS на данные

mov ds, ax

; Вводим число (в данном коде предполагается, что число уже введено и хранится в переменной x)

Myif x, neg, zer, pos

jmp exit

neg:

mov dx, offset neg\_msg

jmp exit

zer:

mov dx, offset zer\_msg

jmp exit

pos:

mov dx, offset pos\_msg

jmp exit

exit:

mov ah, 9

int 21h

mov ah, 4ch

int 21h

code ends

end start

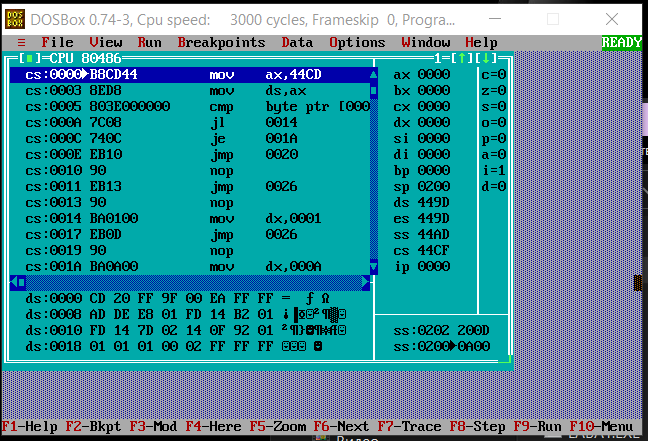


Рисунок 1. До выполнения программы

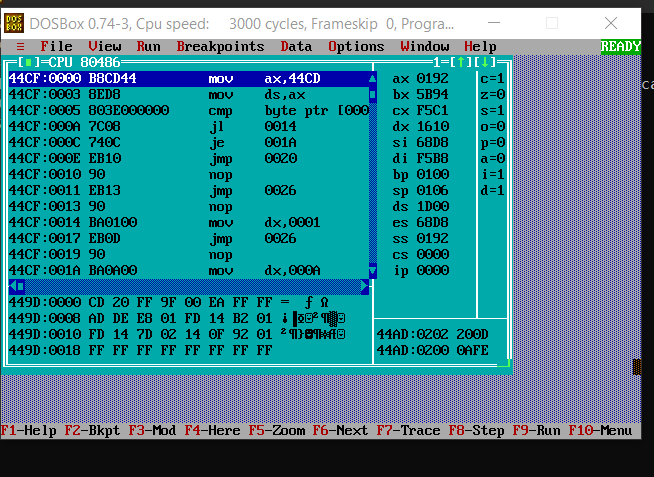
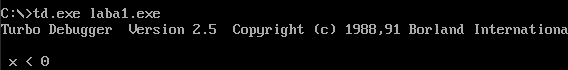
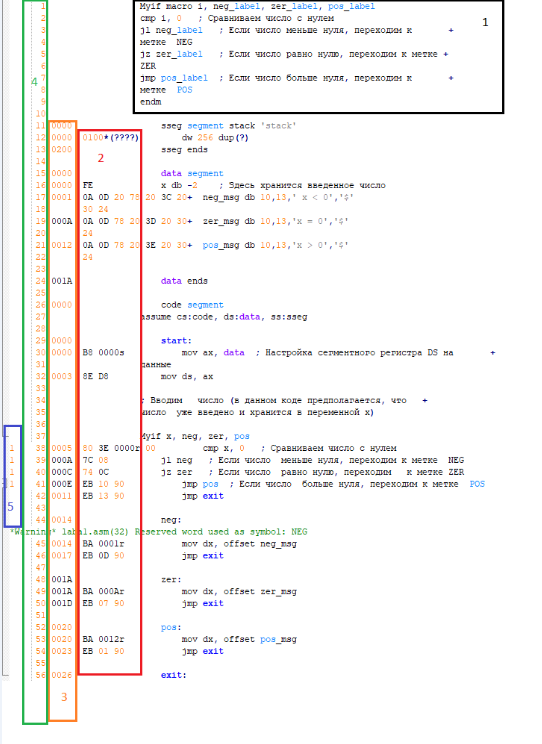


Рисунок 2. После выполнения команды



Листинг команды приведён ниже:



1 – текст программы, включая макрорасширения; 2 – машинные команды; 3 – смещение от начала сегмента; 4 – номер строки в программе; 5 – уровень вложенности.