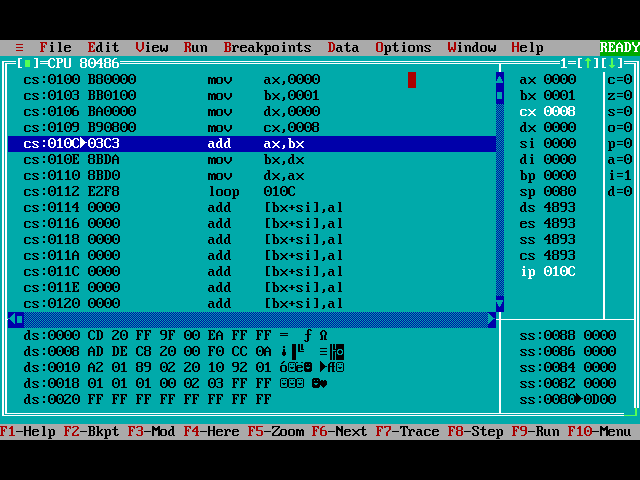
Лабораторная работа 1. **Освоение отладочных средств операционной системы ЭВМ.**

Цель: освоение правил использования ЭВМ для решения задач и освоение средств отладки программ.

Задание 1. Программа вычисления N чисел Фибоначчи.

Наблюдения с пояснениями



Мы написали программу, цикл loop выполняется 8 раз, т.е. до обнуления сх, после чего программа переходит на следующую строку, стоящую за loop.

Программный код с пояснениями:

mov ax,0; инициализация ax

mov bx,1; инициализация bx

mov dx,0; инициализация dx

mov cx,8; инициализация cx (cx-счетчик)

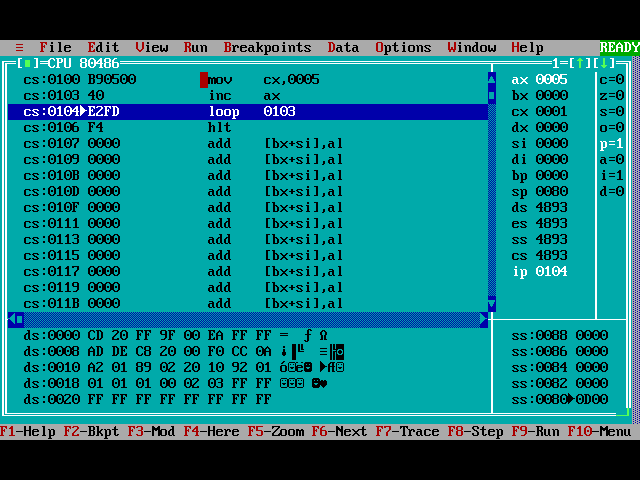
add ax, bx; в ax добавляем bx

mov bx,dx; копирование содержимого регистра dx в bx

mov dx,ax; копирование содержимого регистра ax в dx

loop 010С; команда цикла, количество повторений цикла задается значением в сx.

**Задача 1**



**Программный код:**

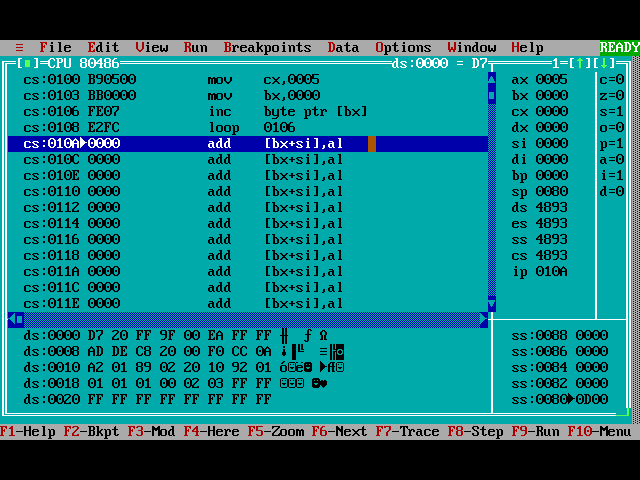
mov cx,5 ; инициализация сх

inc ax; увеличение ах на 1

loop 103; прохождение метки 103 сх раз(т.е. 5 раз)

hlt; остановка машины

**Задача 2**



**Программный код:**

mov cx,5 ; инициализация сх

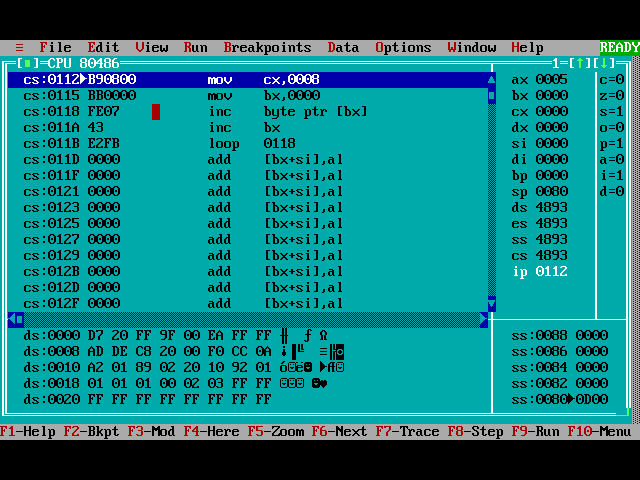
mov bx,0; инициализация bx

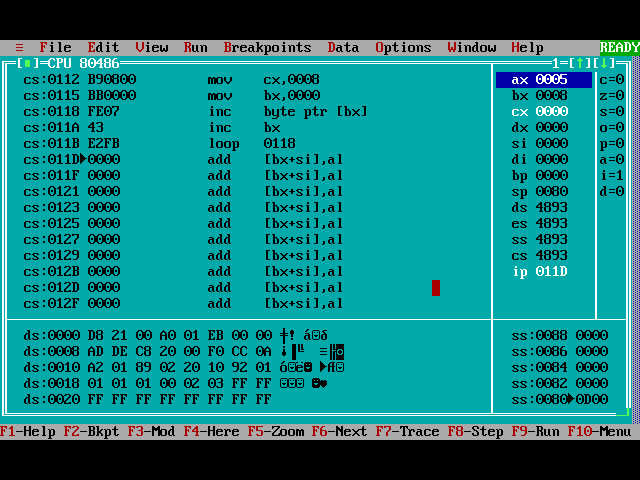
inc byte ptr [bx]; увеличение адреса bx на 1

loop 106; выполнения цикла до обнуления сх

После выполнения программы мы видим, что нулевой элемент в области данных становится 1(s=1)

**Задача 3**





**Программный код**

mov cx,8; инициализация сx

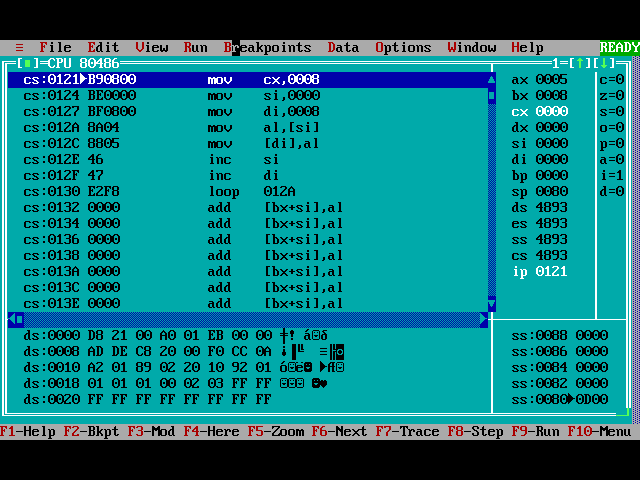
mov bx,0; инициализация bx

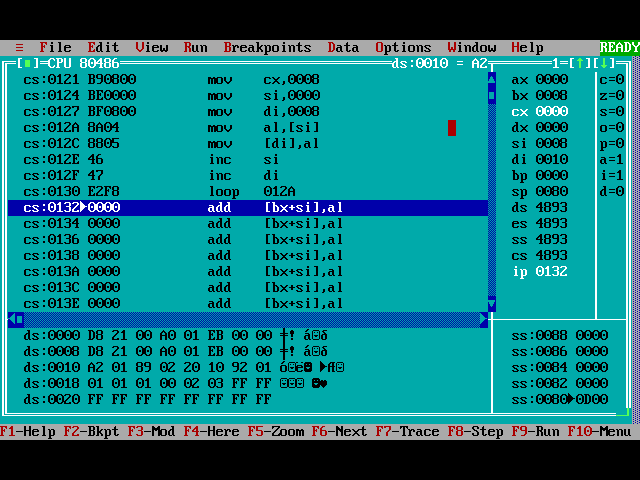
inc byte ptr[bx] ; увеличение на 1 адреса bx.

inc bx; увеличение bx на 1

loop 118; продолжение цикла сх раз с метки 118

**Задача 4**





mov cx,8; инициализация сx

mov si,0; инициализация si( индекса источника)

mov di,8; инициализация di (индекса приемника)

mov al,[si]; копирование si в аl

mov [di], al; копирование al в di

inc si; увеличение на единицу si

inc di; увеличение на единицу di

loop 012A; цикл

**Вывод:**

* **loop** - это команда цикла, она всегда использует вспомогательный регистр сх ,размещается в конце цикла ,уменьшает сх на 1 ,и если сх не равно 0 ,то передает управление по адресу в ней указанному,а если сх равно 0,то передает управление на команду после loop.
* **Числа Фибоначчи**— бесконечная числовая последовательность Fn, в которой каждое число есть сумма двух предыдущих:
* Fn=Fn−1+Fn−2.
* **jmp** - используется в программе для организации безусловного перехода как внутри текущего сегмента команд, так и за его пределы.