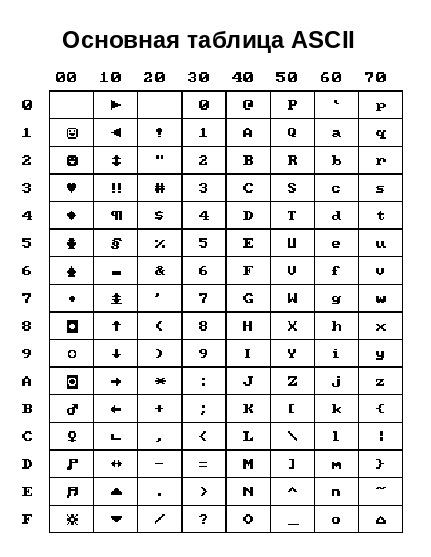
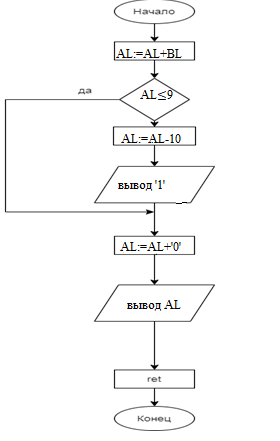
# **ПРАКТИЧЕСКОЕ ЗАНЯТИЕ 1.**

# **Подготовка, трансляция и отладка программы на языке Ассемблера в ОС MS DOS**

Подпрограмма сложения двух чисел, AL, BL от 0 до 9, результат от 0 до 18



; Подпрограмма сложения двух чисел

;Вход:AL,BL–слагаемые, выход: вывод результата на дисплей

add\_and\_show proc

add al,bl ; сложить (AL=AL+BL)

cmp al,9 ; если результат > 9,

jbe not\_carry ; то уменьшить сумму на

sub al,10 ; 10 и вывести на дисплей

push ax ; символ '1' – старшую

mov ah,2h ; цифру результата

mov dl,'1' ; c помощью функции DOS

int 21h ;

pop ax ;

not\_carry: add al,’0’ ; преобразовать результат в код

mov ah,2h ; вывести младшую цифру

mov dl,al ; результата с помощью

int 21h ; функции DOS

ret ; вернуться в вызывающую программу

add\_and\_show endp

code ends

**Описание используемых функций DOS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Функция | Входные  регистры | Выходные  регистры | Описание | Управляющие символы |
| 02h | АН=02h  DL=символ, выводимый на устройство стандартного вывода | Побочный эффект AL=DL, кроме случая, когда DL = 09h (табуляция), тогда в AL возвращается 20h | Выводит символ на экран в текущую позицию курсора с текущими атрибутами символа и фона;  обрабатывает управляющие символы. | BEL (07h) – появляется звуковой сигнал;  BS (08h) – возврат на шаг, курсор перемещается на одну позицию влево;  НТ (09h);  LF (0Ah);  CR (0Dh); |
| 01h | АН = 01h | AL = ASCII-код символа или 0. Если AL = 0, второй вызов этой функции возвратит в AL расширенный ASCII-код символа | Считаeт символ из STDIN с эхом, ожиданием и проверкой на Ctrl-Break | - |
| 09h | АН = 09h DS:DX = адрес строки, заканчивающейся символом $ (24h) | Никакого, согласно документации, но на самом деле: AL = 24h (код последнего символа) | Действие этой функции полностью аналогично действию функции 02h, но выводится не один символ, а целая строка, как в программах hello-1.asm и hello-2.asm. |  |
| 4ch | АН = 4Ch | AL = код возврата | Значение кода возврата можно использовать в пакетных файлах DOS как переменную ERRORLEVEL и определять из программы с помощью функции DOS 4Dh. | - |

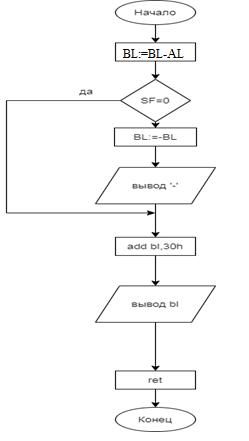
Задание. Изменить текст процедуры ADD\_AND\_SHOW так, чтобы производилось не сложение, а вычитание (SUB\_AND\_SHOW) двух одноразрядных целых десятичных чисел, причем в результате получалось одноразрядное положительное или отрицательное число (например, 8 – 3 = 5, 3 – 8 = -5). Использовать команды *SUB, JNS, NEG.*

sub bl, al; BL=BL-AL

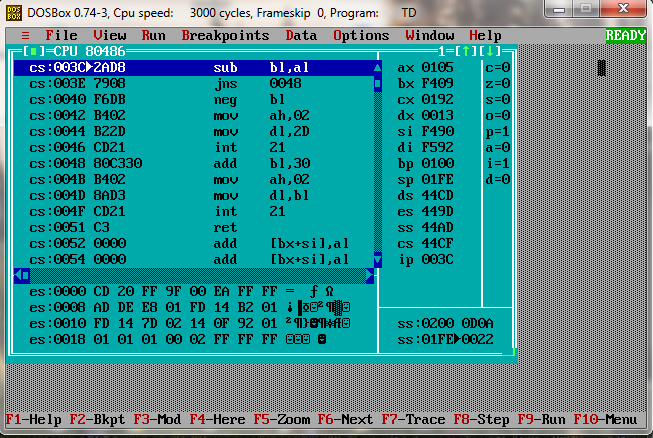
jns метка; если SF=0, то перейти на заданную метку, иначе следующая по порядку команда

neg bl; BL=-BL

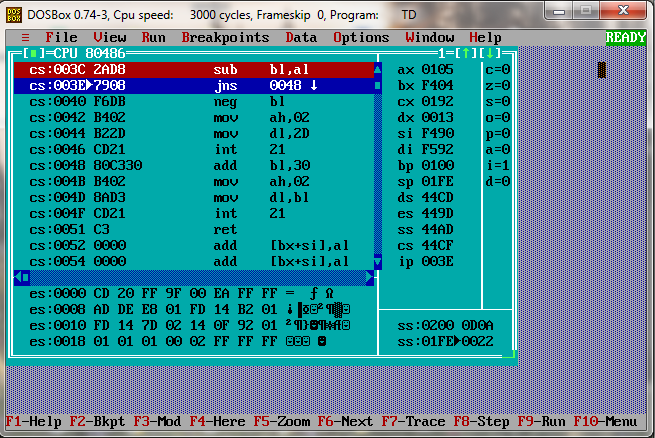
Подпрограмма вычитания двух чисел, BL=BL-AL, BL, AL от 0 до 9, результат от -9 до 9.

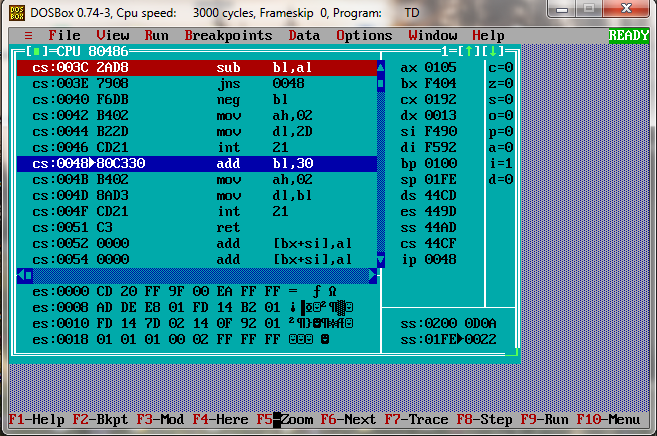


Нужно сделать покомандную прокрутку процедуры вычитания: для каждой команды объяснить ее действие, записать операнды и результат.

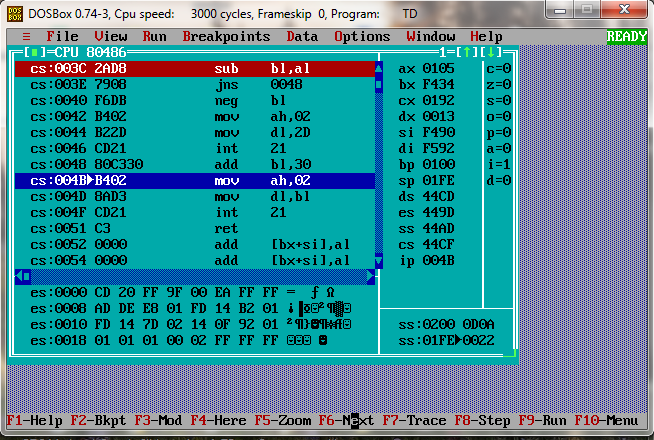


Были введены числа 9 и 5, они записаны в регистры al, bl. При нажатии на F7 произойдет вычитание (bl-al) и результат запишется в BL (будет 04).

Так как число не отрицательное(s=0), то ip поменяется на 0048 и далее будет выполнена команда с этим адресом.

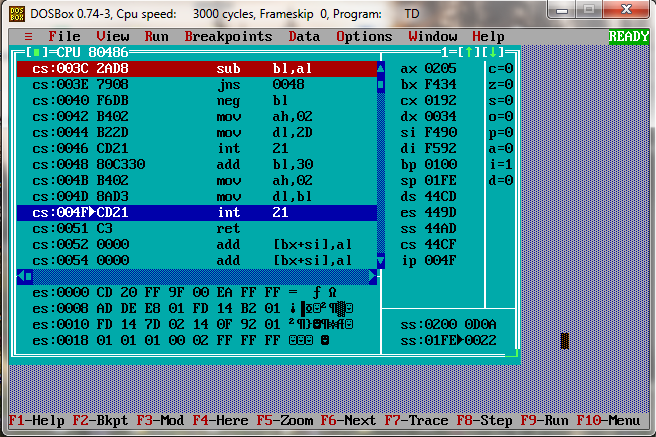


Далее, у нас есть результат вычитания, но нам нужно превратить его в десятичное число, для этого к результату мы прибавляем 30, чтобы в таблице ASCII код соответствовал числу. BL=34.

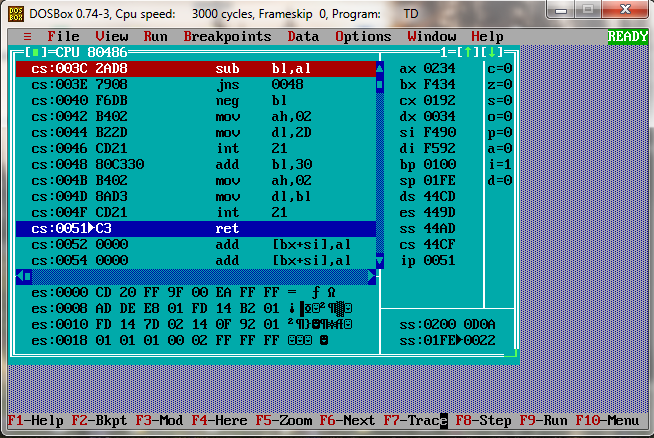


Меняем старший байт в AX на 02, получим AX=0205.

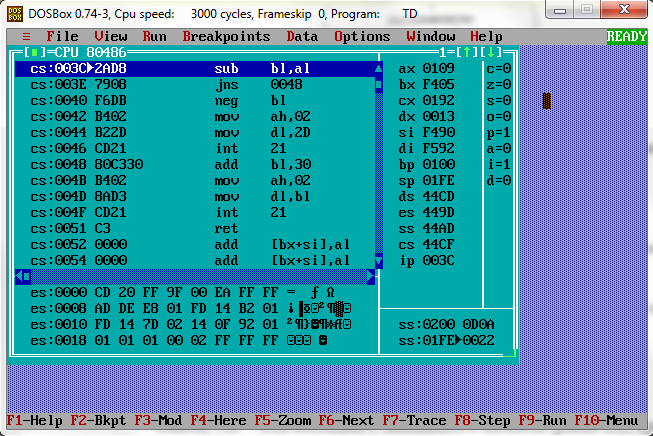
После выполнения команды mov dl,bl DL=34.

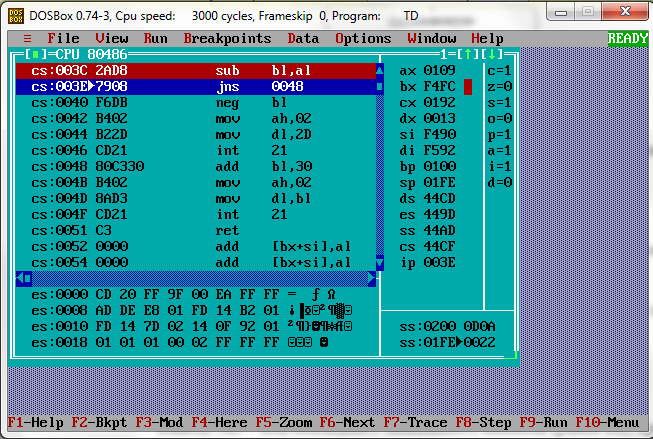


Следующим шагом будет вызов прерывания и возврат в основную программу.

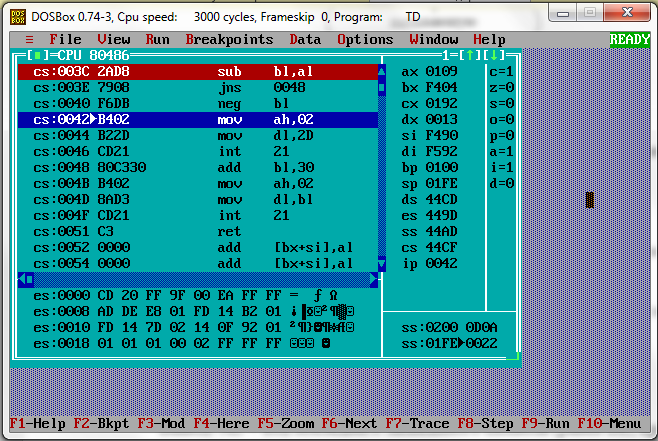


Рассмотрим случай, когда результат будет отрицательным.

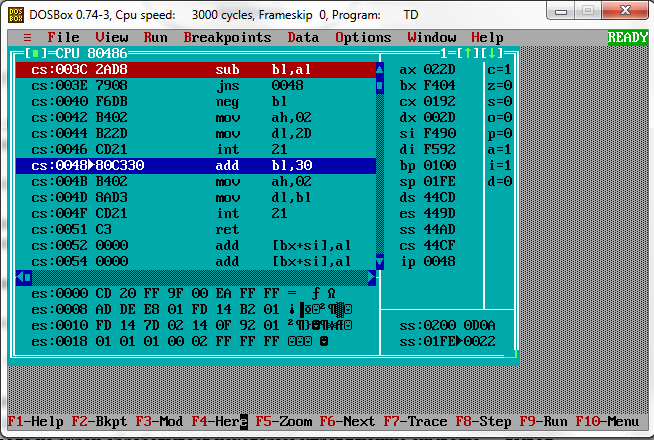
Если ответ будет отрицательным, то S=1. Числа записаны в регистры al, bl. 



Результат вычитания получился FC, чтобы получить его с плюсом, нужно сделать (0-bl). BL=04



Теперь нужно вывести знак ‘-‘ перед числом. Для этого ah=02. В dl записываем код этого символа dl=2D и выводим символ минуса.



Далее идет то же самое, что и в случае с положительным результатом.