# Операционные системы. Лабораторная работа 01

С помощью инструмента Sourcer был получен исходный код обработчика прерывания 08h:

; Вызвать подпрограмму sub\_15  
020A:0746 call sub\_15  
; Сохранить значения регистров ES, DS, AX, DX  
020A:0749 push es  
020A:074A push ds  
020A:074B push ax  
020A:074C push dx  
; Поместить в регистр DS слово 0040h  
020A:074D mov ax, 0040h  
020A:0750 mov ds, ax  
; Поместить в регистр ES слово 0000h  
020A:0752 xor ax, ax  
020A:0754 mov es, ax  
; Инкрементировать младшее слово счётчика таймера  
020A:0756 inc word ptr ds:[006Ch]  
020A:075A jnz loc\_67  
; Инкрементировать старшее слово счётчика таймера   
020A:075C inc word ptr ds:[006Eh]  
020A:0760 loc\_67:  
; Старшее слово равно 0018h?  
020A:0760 cmp word ptr ds:[006Eh], 0018h  
020A:0765 jne loc\_68  
; Младшее слово равно 00B0h?  
020A:0767 cmp word ptr ds:[006Ch], 00B0h  
020A:076D jne loc\_68  
; Обнулить счётчик таймера  
020A:076F mov word ptr ds:[006Eh], ax  
020A:0772 mov word ptr ds:[006Ch], ax  
; Установить флаг прошествия суток  
020A:0775 mov byte ptr ds:[0070h], 1  
; Поместить в регистр AX слово 0008h  
020A:077A or al, 08h  
020A:077C loc\_68:  
; Сохранить значение регистра AX  
020A:077C push ax  
; Декрементировать счётчик отключения приводов дисковода  
020A:077D dec byte ptr ds:[0040h]  
020A:0781 jnz loc\_69  
; Сбросить флаги работы приводов дисковода  
020A:0783 and byte ptr ds:[003Fh], F0h  
; Послать команду остановки приводов в порт контроллера дисковода  
020A:0788 mov al, 0Ch  
020A:078A mov dx, 03F2h  
020A:078D out dx, al  
020A:078E loc\_69:  
; Восстановить значение регистра AX  
020A:078E pop ax  
; Установлен PF?  
020A:078F test word ptr ds:[0314h], 0004h  
020A:0795 jnz loc\_70  
; Сохранить младший байт регистра FLAGS в AH  
020A:0797 lahf  
; Поменять местами значения байтов регистра AX  
020A:0798 xchg ah, al  
; Сохранить значение регистра AX  
020A:079A push ax  
; Косвенно вызвать прерывание 1Ch  
020A:079B call dword ptr es:[0070h]  
020A:07A0 jmp short loc\_71  
020A:07A2 nop  
020A:07A3 loc\_70:  
; Вызвать прерывание 1Ch  
020A:07A3 int 1Ch  
020A:07A5 loc\_71:  
; Вызвать подпрограмму sub\_15  
020A:07A5 call sub\_15  
; Послать команду сброса в порт контроллера прерываний  
020A:07A8 mov al, 20h  
020A:07AA out 20h, al  
; Восстановить значения регистров DX, AX, DS, ES  
020A:07AC pop dx  
020A:07AD pop ax  
020A:07AE pop ds  
020A:07AF pop es  
020A:07B0 jmp loc\_50

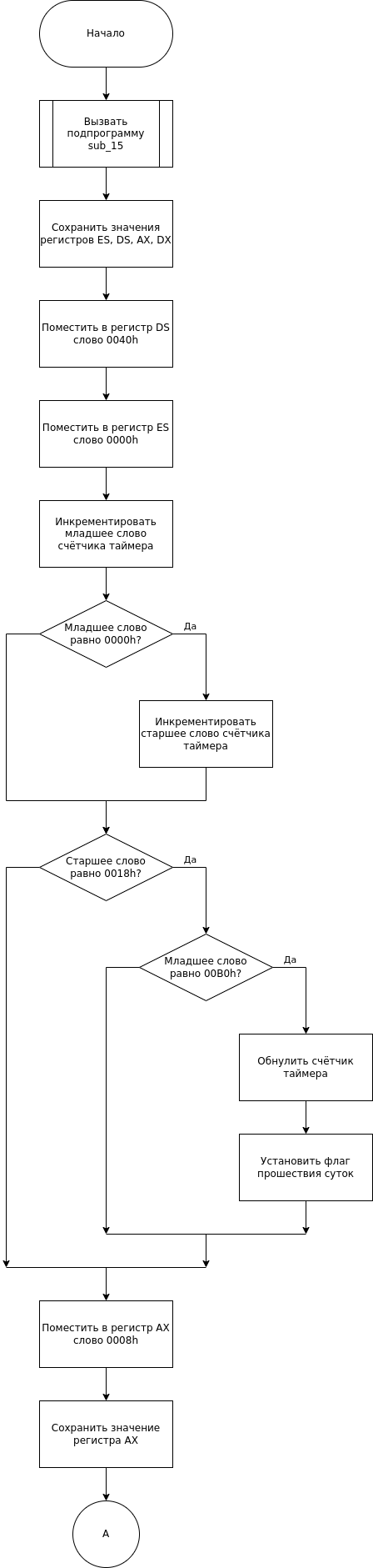
Метка loc\_50 здесь отсылает к команде iret:

020A:064C loc\_50:  
020A:064C push ds  
020A:064D push ax  
...  
020A:06AA pop ax  
020A:06AB pop ds  
; Вернуться из обработчика прерывания  
020A:06AC iret  
sub\_10 endp

Также был получен исходный код подпрограммы sub\_15:

sub\_15 proc near  
; Сохранить значения регистров DS, AX  
020A:07B9 push ds  
020A:07BA push ax  
; Поместить в регистр DS слово 0040h  
020A:07BB mov ax, 0040h  
020A:07BE mov ds, ax  
; Сохранить младший байт регистра FLAGS в AH  
020A:07C0 lahf  
; Установлен DF или старший бит IOPL?  
020A:07C1 test word ptr ds:[314h], 2400h  
020A:07C7 jnz loc\_73  
; Сбростить IF с блокировкой шины данных  
020A:07C9 lock and word ptr ds:[314h], 0FDFFh  
020A:07D0 loc\_72:  
; Восстановить младший байт регистра FLAGS из AH  
020A:07D0 sahf  
; Восстановить значения регистров AX, DS  
020A:07D1 pop ax  
020A:07D2 pop ds  
020A:07D3 jmp short loc\_74  
020A:07D5 loc\_73:  
; Запретить маскируемые прерывания  
020A:07D5 cli  
020A:07D6 jmp short loc\_72  
020A:07D8 loc\_74:  
; Вернуться из подпрограммы  
020A:07D8 retn  
sub\_15 endp

По исходному коду обработчика прерывания 08h была составлена схема его алгоритма:

Также была составлена схема алгоритма подпрограммы sub\_15:

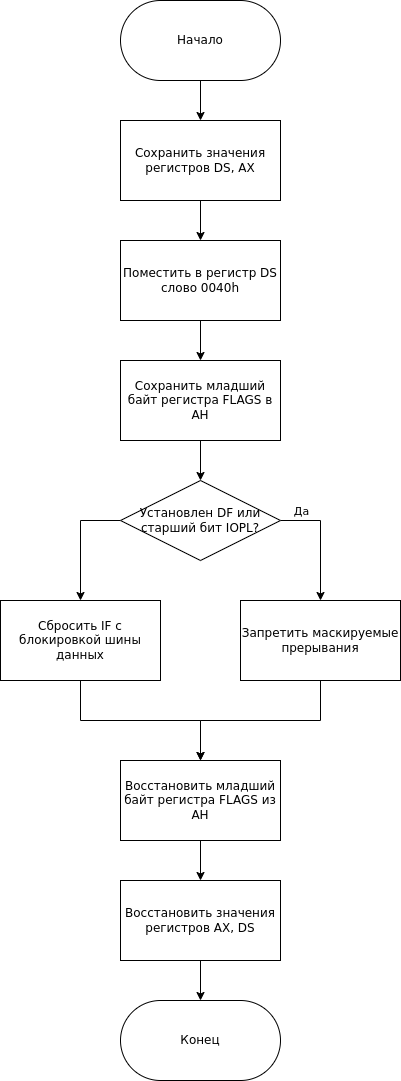


Рисунок 3. Схема алгоритма подпрограммы sub\_15