Федеральное государственное бюджетное образовательное учреждение высшего образования

«Чувашский государственный университет им. И.Н. Ульянова»

Факультет информатики и вычислительной техники  
Кафедра вычислительной техники

Лабораторная работа №7

по дисциплине ЭВМ и периферийные устройства

ИЗУЧЕНИЕ ОБРАБОТЧИКА ПРЕРЫВАНИЯ КЛАВИАТУРЫ INT 9

Выполнил:Крупнов.

студент группы ИВТ-43-20

Проверила: доц. к.т.н.

Андреева А. А.

Чебоксары 2023 г.

**Цель работы:** изучить обработчик прерываний клавиатуры int 9 и получить практические навыки в программировании обработчика прерываний.

0000h:0417h

Бит 7: Ins включена

Бит 6: CapsLock включена

Бит 5: NumLock включена

Бит 4: ScrollLock включена

Бит 3: Alt нажата (любая Alt для функции 02h, часто только левая Alt для 12h/22h)

Бит 2: Ctrl нажата (любая Ctrl)

Бит 1: Левая Shift нажата

Бит 0: Правая Shift нажата

Байт состояния клавиатуры 2 (этот байт всегда расположен в памяти по адресу 0000h:0418h или 0040h:0018h):

Бит 7: SysRq нажата

Бит 6: CapsLock нажата

Бит 5: NumLock нажата

Бит 4: ScrollLock нажата

Бит 3: Правая Alt нажата

Бит 2: Правая Ctrl нажата

Бит 1: Левая Alt нажата

Бит 0: Левая Ctrl нажата

1. Изменил комбинацию клавиш <Ctrl+Alt> на <LeftShift+S>:

Было:

mov AL,ES:[417h]

and AL,03h

cmp AL,03h

< LeftShift+S >

0000 0011

Биты 0 и 1

Стало:

mov AL,ES:[417h]  
and AL,2h  
cmp AL,2h

1. Изменил букву L на S:

cmp AL,1Fh

1F (0001 1111) – скан-код нажатия клавиши S

1. Горячие клавиши разблокировки клавиатуры “D+A”
2. D\_make:
3. in AL, 60h
4. cmp AL,20h
5. je D\_break
6. jmp D\_make
7. D\_break:
8. in AL,60h
9. cmp AL, 0A0h
10. jne D\_break
11. A\_make:
12. in AL, 60h
13. cmp AL,0A0h
14. je A\_make
15. cmp AL, 1Eh
16. je A\_break
17. jmp D\_make
18. A\_break:
19. in AL,60h
20. cmp AL, 9Eh
21. jne A\_break

Скан-коды нажатия и отпускания клавиш

|  |  |  |
| --- | --- | --- |
| Клавиша | Скан-код нажатия | Скан-код отпускания |
| D | 20h | 0A0h |
| A | 32h | 0B2h |

1. Вывод работы: изучил обработчик прерываний клавиатуры int 9 и получил практические навыки в программировании обработчика прерываний.

Реализация прерывания:

1. Флаги проталкиваются в стек
2. Очищаются флаги IF и TF
3. Содержимое регистров IP и CS проталкиваются в стек
4. Вектор прерывания выбирается из таблицы векторов прерываний и с его помощью осуществляется переход на подпрограмму обслуживания прерывания

**Как процессор добирается до нужного обработчика прерывания:**

1. **поступление сигнала прерывания в ЦП. При этом сигнал маскируемого внешнего аппаратного прерывания поступает на вход INTR процессора,** сигнал немаскируемого внешнего аппаратного прерывания – на вход NMI, а сигнал исключения или программного прерывания поступает на внутреннюю линию процессора;
2. **собственно прерывание.** Оно выполняется ЦП по окончанию выполнения текущей машинной команды и включает действия:

* **текущее содержимое FLAGS, CS и IP помещается в ст**ек (рис.66). Последние два слова представляют собой адрес возврата в прерванную программу;
* **считывание** с внутренней шины процессора 8-битного числа – **номера прерывания N**;
* **копирование вектора прерываний с номером N в регистры IP и CS**. В результате эти регистры содержат стартовый адрес обработчика прерываний;
* **сбрасываются в нуль флажки IF и TF в регистре FLAGS**. В результате запрещаются все маскируемые внешние прерывания (кроме того, которое уже обрабатывается), а также запрещаются исключения «трассировка»;

1. **начальный этап программной обработки прерывания.** Он выполняется обработчиком прерываний и включает действия:

* **разрешение маскируемых прерываний с помощью команды sti**. В результате важные маскируемые прерывания, например от таймера, не будут более откладываться и вызовут прерывание текущего обработчика прерываний точно так, как прерывается обыкновенная программа;
* **сохранение в программном стеке содержимого тех регистров, с которыми будет работать программа обработчика;**
* **запись в сегментный регистр данных DS значения, которое соответствует адресу-сегменту данных обработчика прерываний.** Это выполняется потому, что в результате прерывания из всех сегментных регистров только CS «переключается» на обработчик прерываний, а остальные сегментные регистры по-прежнему адресуют данные прерванной программы;

1. **действия, определяемые типом прерывания.** Эти действия обработчик прерываний выполняет сам или обращается за помощью к другим подпрограммам ОС, вызывая их как обычные процедуры командами call или используя для их вызова команды программного прерывания int;
2. **завершение обработки прерывания.** Сюда относятся действия: восстановление содержимого регистров, запомненного ранее в стеке; возврат из прерывания. Его выполняет команда iret, завершающая программу обработчика прерываний. При попадании этой команды на ЦП его аппаратура выталкивает из стека в регистры IP, CS и FLAGS прежнее их содержимое. В результате следующей выполняемой на ЦП командой будет очередная команда прерванной программы.