



Инновационный Евразийский Университет

Департамент «ЭМиИТ»

Слайд-лекция

По дисциплине:

«Архитектура компьютерных систем»

Лекция 8

Организация прерываний. Аппаратные прерывания

Разработал: проф. ИнЕУ Зайцева Н.М.

Организация прерываний. Аппаратные прерывания

- Для того чтобы процессор не дожидался состояния готовности устройства ввода-вывода в цикле, а мог выполнять в это время другую работу, необходимо, чтобы устройство само умело сигнализировать процессору о своей готовности.
- Технический механизм, который позволяет внешним устройствам оповещать процессор о завершении команды вывода или команды ввода, получил название механизма прерываний.

Организация прерываний. Аппаратные прерывания

- В простейшем случае для реализации механизма прерываний необходимо к имеющимся у нас шинам локальной магистрали добавить еще одну линию, соединяющую процессор и устройства ввода-вывода – линию прерываний. По завершении выполнения операции внешнее устройство выставляет на эту линию специальный сигнал, по которому процессор после выполнения очередной команды (или после завершения очередной итерации при выполнении цепочечных команд,

Организация прерываний. Аппаратные прерывания

- т. е. команд, повторяющихся циклически со сдвигом по памяти) изменяет свое поведение.
- Вместо выполнения очередной команды из потока команд он частично сохраняет содержимое своих регистров и переходит на выполнение программы обработки прерывания, расположенной по заранее оговоренному адресу.

Организация прерываний. Аппаратные прерывания

- При наличии только одной линии прерываний процессор при выполнении этой программы должен опросить состояние всех устройств ввода-вывода, чтобы определить, от какого именно устройства пришло прерывание (polling прерываний!), выполнить необходимые действия (например, вывести в это устройство очередную порцию информации или перевести соответствующий процесс из: состояния **ожидание** в состояние **готовность**)
- и сообщить устройству, что прерывание обработано (снять прерывание).

Организация прерываний. Аппаратные прерывания

- В большинстве современных компьютеров процессор стараются полностью освободить от необходимости опроса внешних устройств, в том числе и от определения с помощью опроса устройства, сгенерировавшего сигнал прерывания.
- Устройства сообщают о своей готовности процессору не напрямую, а через специальный контроллер прерываний, при этом для общения с процессором он может использовать не одну линию, а целую шину прерываний.

Организация прерываний. Аппаратные прерывания

- Каждому устройству присваивается свой номер прерывания, который при возникновении прерывания контроллер прерывания заносит в свой регистр состояния и, возможно, после распознавания процессором сигнала прерывания и получения от него специального запроса выставляет на шину прерываний или шину данных для чтения процессором.
- Номер прерывания обычно служит индексом в специальной таблице прерываний, хранящейся по адресу, задаваемому при инициализации вычислительной системы, и содержащей адреса программ обработки прерываний – **векторы прерываний**.

Организация прерываний. Аппаратные прерывания

- Для распределения устройств по номерам прерываний необходимо, чтобы от каждого устройства к контроллеру прерываний шла специальная линия, соответствующая одному номеру прерывания.
- При наличии множества устройств такое подключение становится невозможным, и на один проводник (один номер прерывания) подключается несколько устройств.
- В этом случае процессор при обработке прерывания все равно вынужден заниматься опросом устройств для определения устройства, выдавшего прерывание, но в существенно меньшем объеме.

Организация прерываний. Аппаратные прерывания

- Обычно при установке в систему нового устройства ввода-вывода требуется аппаратно или программно определить, каким будет номер прерывания, вырабатываемый этим устройством.
- Доступ к адресам **I/O** на языке Ассемблера возможен только при помощи двух специальных команд:
- **IN** приемник, источник и
- **OUT** источник, приемник,
- используемых исключительно для данной цели. Например, в порт, номер которого записан в **dx**, выводится символ, хранящийся в **al**:
- **out dx,al**

Организация прерываний. Аппаратные прерывания

- Помимо непосредственного ввода информации из порта или вывода в порт в Ассемблере существует возможность обратиться к функциям прерывания **DOS** (Disk Operation System - дисковая операционная система) и **BIOS** (Base Input Output System- базовая система ввода-вывода), которые значительно облегчают ввод и вывод информации.

Функции прерывания DOS (INT 21h) ввода-вывода

- **Функция DOS 09h:** вывести строку на экран.
- **Вход :** AH=09h
- DS:DX=адрес строки, заканчивающейся символом '\$'
- **Действие :** выводится на экран монитора строка символов не более 256 символов из памяти компьютера.

- **Функция DOS 06h :** вывести на экран один символ.
- **Вход :** AH=06h
- DL= код символа '\$'
- **Действие :** выводится на экран монитора один символ из регистра DL.
-

Функции прерывания DOS (INT 21h) ввода-вывода

- **Функция DOS 40h:** вывести строку в файл или устройство.
- **Вход :** AH=40h
- BX=1 для вывода на экран, 0- для вывода в файл
- DS:DX=адрес строки
- CX=длина строки
- **Выход :** CF=0
- AX= число записанных байтов
- **Действие:** выводится строка символов на экран или в файл.

Функции прерывания DOS (INT 21h) ввода-вывода

- **Функция DOS 0Ah:** ввести строку с клавиатуры.
- **Вход :** AH=0Ah DS:DX=адрес строки'
- **Действие :** вводится строка символов с клавиатуры в память компьютера, причем область памяти, куда производится ввод должна иметь определенную структуру(смотри ниже) .
- Функция предоставляет удобный способ ввода информации пользователю. Она использует строку ввода следующим образом:
- **Str db 28;** максимальное число символов строки 28
- **Str_l db ?;** введенное число символов
- **Buf db 28 dup(?);** память для вводимой строки символов

Функции прерывания BIOS (INT 16h, INT 10h) ввода-вывода

- **BIOS** предоставляет больше возможностей по сравнению с **DOS** для работы с экраном и клавиатурой. Так для работы с клавиатурой существует ряд функций прерывания **int 16h**. Приведем некоторые из них
- **Функция BIOS 0h**: чтение символа с ожиданием.
- **Вход** : AH=0h; (83/84- key), 10h (101/102-key), 20h (122-key)
- **Выход** : AL=ASCII-код символа
- AH= скан-код нажатой клавиши или расширенный ASCII-код

Функции прерывания BIOS (INT 16h, INT 10h) ввода-вывода

- Не каждой клавише на клавиатуре соответствует ASCII-код (ASCII-код имеют клавиши с цифрами, буквами или символами). Например такие клавиши как Esc, F1-F12, Shift и т.п., его не имеют, но каждой клавише на клавиатуре соответствует так называемый скан-код, соответствующий только этой клавише. Этот код позволяет определить какая именно клавиша или сочетание клавиш были нажаты.
- **Функция BIOS 02h, 12h , 22h:** чтение состояния клавиатуры
- **Вход :** AH=02h; (83/84- key), 12h (101/102- key), 22h (122-key)

Функции прерывания BIOS (INT 16h, INT 10h) ввода-вывода

- **Выход** : AL=байт состояния клавиатуры
- Структура байта:
 - Бит 7: **Ins** включена
 - Бит 6: **CapsLock** включена
 - Бит 5: **NumLock** включена
 - Бит 4: **ScrollLock** включена
 - Бит 3: **Alt** нажата
 - Бит 2: **Ctrl** нажата
 - Бит 1: правая **Shift** нажата
 - Бит 0: левая **Shift** нажата
- **BIOS** позволяет переключать видеоадаптер в различные текстовые и графические режимы.

Функции прерывания BIOS (INT 16h, INT 10h) ввода-вывода

- Режимы отличаются друг от друга разрешением (для графических) и количеством строк и столбцов (для текстовых), а также количеством возможных цветов.
- Цветные адаптеры имеют два основных режима работы: текстовый (алфавитно-цифровой) и графический. По умолчанию используется текстовый режим. Для установки графического режима или возврата в текстовый режим используется прерывание **BIOS int 10h**, как это показано в двух следующих примерах:
 - **mov ah,00**; функция выбора режима
 - **mov al,03**;цветной текстовый
 - **int 10h**

Функции прерывания BIOS (INT 16h, INT 10h) ввода-вывода

- **mov ah, 00**; Функция выбора режима
- **mov al, 04**; графический среднего разрешения
- **int 10h** ;
- Вызов функции **ah=00** приводит к тому, что экран устанавливается в режим, помещенный в регистр **al**. Номера текстовых режимов:
 - 0 и 1 - 16-цветные режимы с 40 символами в строке и 25 строк;
 - 2 и 3 - 16-цветные режимы с 80 символами в строке и 25 строк;
 - 7 - монохромный режим с 80 символами в строке и 25 строк.

Функции прерывания BIOS (INT 16h, INT 10h) ввода-вывода

- В приведенном ниже примере с помощью функции установки курсора курсор убирается с экрана:
- **mov ah,02h** ;Установить курсор
- **mov bh,0** ; видеостраница 0
- **mov dh,26** ;на строку 26, т.е. за пределы экрана
- **mov dl,1** ;колонка 0
- **int 10** ;Курсора на экране нет

Функции прерывания BIOS (INT 16h, INT 10h) ввода-вывода

- Для установки видеорежимов SVGA используется функция **ah=4Fh** с параметром, помещенным в регистр **al=02h** и с указанием в **bx** номера режима в младших 13 битах:
 - 108 h -.80×60
 - 109 h -.132×25
 - 10A h -.132×43
 - 10B h -.132×50
 - 10C h -.132×60
- Если бит 15 регистра **VX** установлен в 1, то видеопамять не очищается.
- Для управления положением курсора существует специальная функция.

Функции прерывания BIOS (INT 16h, INT 10h) ввода-вывода

- Для вызова этой функции необходимо поместить в регистр **ah** ее номер **02h**, в регистр **dh** - номер строки, **dl**- номер столбца, **bh**- номер видеостраницы. Номера видеостраниц: 0-3(для режимов 2 и 3) и 0-7(для режимов 0 и 1).