

## Тема 5. Драйверы и администраторы устройств

*Драйвер устройства* –системная программа, выполняющая под управлением ОС все операции с конкретным периферийным устройством.

### *Задачи, решаемые драйвером:*

- обеспечение возможности стандартного обращения к любому устройству, без учета специфических особенностей этого устройства;
- максимально эффективное использование функциональных возможностей и особенностей конкретного устройства.

Возможность стандартными средствами работать с разными устройствами важная особенность ОС, так как прикладной программист в этом случае может не учитывать особенности конкретных устройств. Например, при организации печати можно не учитывать особенности конкретной марки принтера. Все особенности устройства учитываются в одном – в драйвере устройства.

Полностью скрыть все различия между устройствами, однако невозможно, например нельзя приравнять, скажем, диск к клавиатуре, принтер к сканеру, но. В большинстве ОС различаются, как минимум, два разных типа драйверов: для символьных и для блочных устройств.

Обращаясь к драйверу, ОС указывает функцию, которую требуется выполнить. Список этих функций общий для драйверов различных устройств, при этом каждый драйвер может реализовать только те функции, которые имеют смысл для данного устройства. Наиболее общими являются функции чтения данных, записи данных, инициализации устройства (вызывается системой один раз, сразу после загрузки), открытия и закрытия устройства (если символьное устройство открывается как файл). Для учета разнообразия возможных операций, в число функций драйвера вводят функцию «выполнение специальных функций», которая для каждого устройства определяет набор специальных функций.

Типичный драйвер устройства содержит три основных блока:

- заголовок драйвера;
- блок стратегии;
- блок прерываний.

**Заголовок** содержит различную информацию о данном драйвере и об управляемом устройстве. Он может включать имя устройства, тип устройства, число однотипных устройств, обслуживаемых одним драйвером, объем памяти на устройстве и т.п. Заголовок содержит также адреса блока стратегии и блока прерываний.

**Блока стратегии** отвечает за прием заявок на выполнение операции, ведение очереди заявок (в многозадачных системах, а также при асинхронных операциях, выполнения могут дожидаться несколько заявок), а также запуск операции и ее завершение. Заявка на выполнение операции представляет собой

стандартную запись, формируемую системой перед обращением к драйверу. Заявка содержит код требуемой функции драйвера и сведения об адресе данных в памяти и на устройстве, о количестве передаваемых данных. Заявка также содержит поле, в которое драйвер должен будет записать код завершения операции (обычно 0 – нормально выполненная операция, другие значения – коды ошибок).

Система вызывает **блок прерываний**, когда получает сигнал прерывания от устройства, обслуживаемого драйвером. Закончив выполнение заявки, блок прерываний возвращает управление блоку стратегии для завершения операции.

Помимо основных блоков драйвер может содержать блок инициализации, блок изменения параметров драйвера и др.

При использовании многоуровневой схемы использования драйверов кроме низкоуровневых драйверов аппаратуры используются высокоуровневые драйверы, расположенные между драйверами аппаратуры и остальной частью ОС. Высокоуровневый драйвер не содержит блока прерываний, он принимает заявки от системы, преобразует данные тем или иным образом, а затем вызывает низкоуровневый драйвер для работы с устройством.

В Windows используется многоуровневая структура драйверов, в которой высокоуровневые драйверы могут играть роль фильтров, выполняющих специальную обработку данных, полученных от драйвера низкого уровня или передаваемых такому драйверу.

Выделяют специальные типы драйверов, отличающихся функциональным назначением:

- **Драйверы GDI** (Graphic Device Interface) представляют собой высокоуровневые драйверы графических устройств (мониторов, принтеров, плоттеров). Эти драйверы выполняют трансляцию графических вызовов Windows в команды, выполняющие соответствующие действия на конкретном устройстве. Выдача этих команд на устройство выполняется уже другим, низкоуровневым драйвером. Благодаря наличию драйверов GDI одна и та же программа может посылать графическое изображение на разные устройства.

- **Драйверы клавиатуры и мыши** генерируют сообщения о событиях на соответствующем устройстве (нажатие и отпускание клавиши, перемещение мыши, нажатие и отпускание кнопок мыши) и помещают их в системную очередь сообщений. Затем система переправляет каждое сообщение процессу, которому оно было предназначено, для дальнейшей обработки.

- **Драйверы виртуализации устройств** (VxDдрайверы) предназначены для разделения устройств между процессами. Драйвер организует очередь заявок от процессов, переключает устройство в нужный для очередного процесса режим и т.п.

В большинстве случаев программы работают не непосредственно с устройствами. Вместо этого для выполнения требуемых операций используются API-функции более высокого уровня, а обращения к устройствам выполняются системой по мере надобности.

В UNIX различаются два основных типа драйверов: символьные и блочные.

Для символьных устройств используются только символьные драйверы. Для каждого блочного устройства обычно имеется два разных драйвера: блочный и символьный. Блочный драйвер позволяет выполнять операции только с целым числом блоков. Символьный драйвер блочного устройства является более высокоуровневой программой, которая имитирует выполнение операций чтения и записи произвольного количества байт, на самом деле используя обращения к блочному драйверу.

Помимо драйверов реальных физических устройств, система может включать драйверы «псевдоустройств». Примером может служить драйвер, обеспечивающий обращение программ к содержимому системной памяти.

При загрузке системы формируются две таблицы, для символьных и для блочных драйверов. Строки таблицы соответствуют конкретным драйверам, а столбцы – функциям, которые должен уметь выполнять драйвер, так что в ячейках таблицы содержатся адреса, по которым вызываются функции драйвера. Набор функций для символьных и для блочных драйверов слегка разнится, поэтому используются две разных таблицы.

К наиболее важным функциям драйвера относятся следующие:

- **открытие устройства.** Увеличение счетчика текущих обращений к устройству, что позволяет ставить обращения в очередь, если устройство занято;
- **закрытие устройства;**
- **обработка прерывания** – ввод или вывод очередной порции данных, когда устройство переходит в состояние готовности;
- **опрос устройства** – выполняется для устройств, не генерирующих прерываний. Опрос выполняется с некоторым периодом, по прерыванию от таймера.
- **чтение данных с устройства;**
- **запись данных на устройство;**
- **вызов стратегии** – это способ выполнения операций ввода/вывода, характерный для блочных устройств;
- **выполнение специальных функций.** Набор этих функций зависит от конкретного устройства.

В UNIX для работы с периферийными устройствами прикладные программы используют те же средства, что для работы с файлами. Устройства в UNIX представлены как специальные файлы, вписанные в каталог файловой системы наравне с обычными файлами. Каждому драйверу устройства соответствует отдельный специальный файл, символьный или блочный, в зависимости от типа драйвера. Чтобы начать работу с устройством, программа должна вызвать функцию открытия файла, указав ей имя специального файла.

При этом происходит обращение к функции открытия из драйвера соответствующего устройства.

С каждым специальным файлом связаны два числа, называемые старшим и младшим номерами устройства. Старший номер определяет номер строки в таблице символьных либо блочных драйверов. Младший номер передается драйверу как дополнительный параметр. Он может означать, например, номер конкретного дискового устройства.