

# Операционные системы. Автор В.А.Серков

Подсистема управления вводом-  
выводом

# Задачи

1. Обеспечение интерфейса между устройствами ввода/вывода и остальными устройствами вычислительной системы.
2. Передача устройствам команд ввода/вывода.
3. Перехват прерываний.
4. Обработка ошибок.

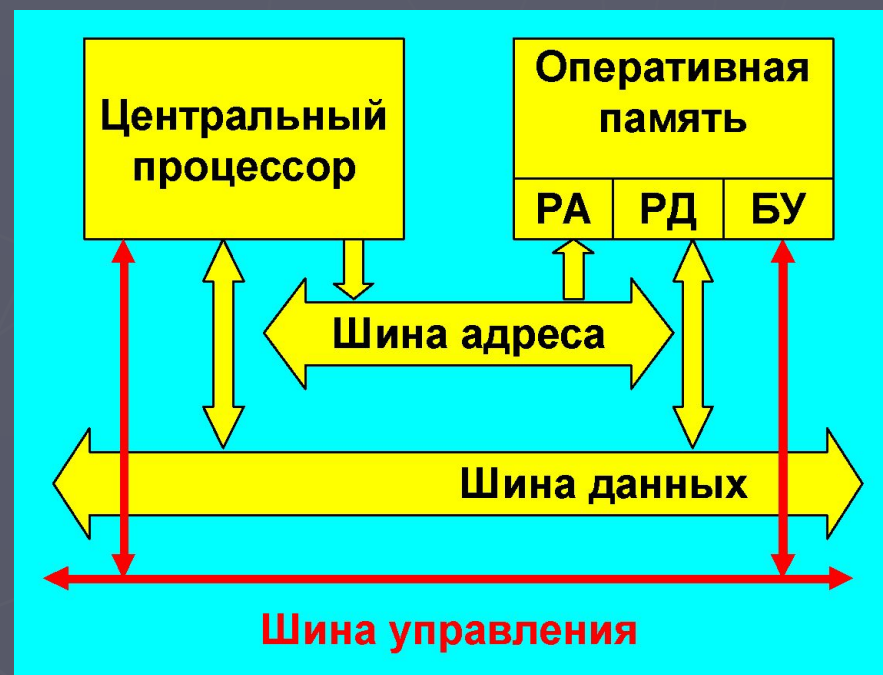
# Физическая организация устройств ввода-вывода

# Обмен данными между процессором и оперативной памятью

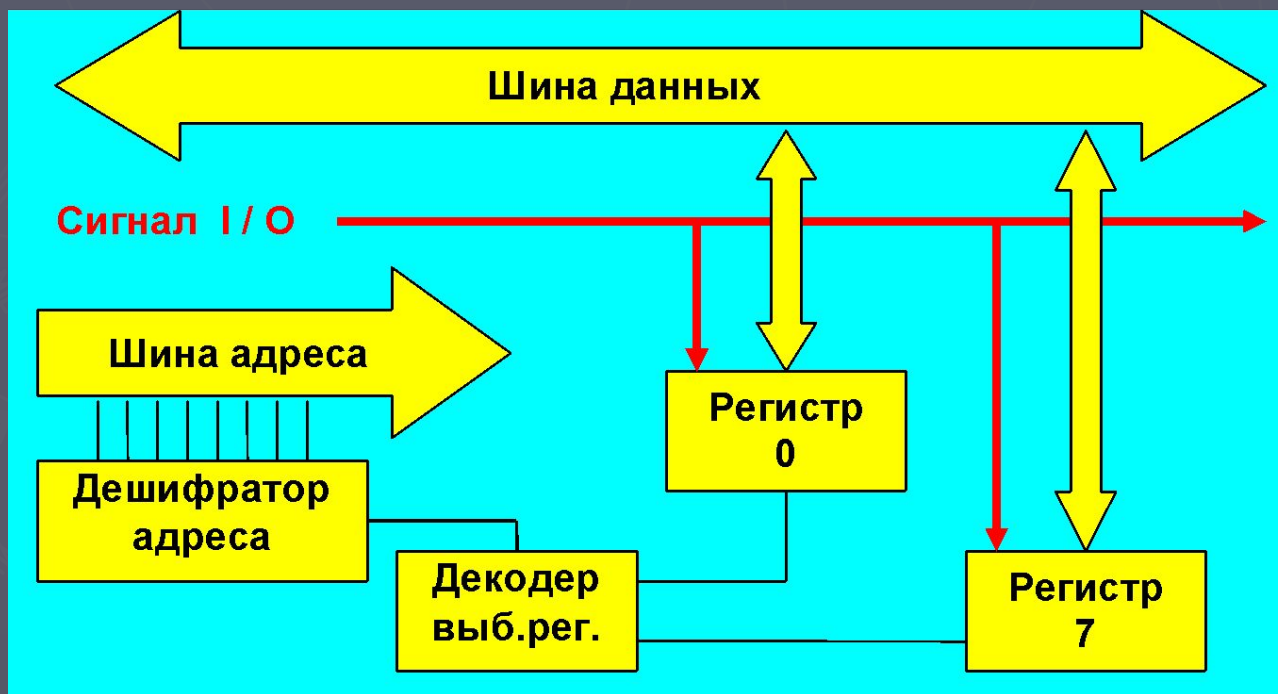
Процессор выставляет на адресную шину системного интерфейса номер ячейки, к которой происходит обращение. На шину управления подается код команды (чтения или записи).

При записи процессор передает код записываемого числа на шину данных и ОП записывает этот код в нужную ячейку.

При чтении ОП передает код читаемого числа на шину данных, а процессор считывает этот код.



Если выставленный на шине адрес соответствует адресу одного из регистров внешнего устройства, дешифратор подключает соответствующий регистр к шине данных. Таким образом, регистры устройства получают адреса в физическом адресном пространстве процессора.



Можно выделить два основных подхода к адресации регистров внешних устройств:

- отдельное адресное пространство ввода-вывода в этом случае для обращения к регистрам устройств используются специальные команды IN и OUT;
- отображенный в память ввод-вывод (memory-mapped I/O), когда память и регистры внешних устройств размещаются в одном адресном пространстве тогда могут использоваться любые команды, способные работать с операндами в памяти.

Существует два основных подхода к выделению адресов внешним устройствам:

- **фиксированная адресация**, когда одно и то же устройство всегда имеет одни и те же адреса регистров;
- **географическая адресация**, когда каждому разъему периферийной (или системной, если внешние устройства подключаются непосредственно к ней) шины соответствует свой диапазон адресов. Географически можно распределять не только адреса регистров, но и другие ресурсы — линии запроса прерывания, каналы ПДП.

Устройства ввода-вывода делятся на два типа:

**Блок-ориентированные устройства** хранят информацию в блоках фиксированного размера, каждый из которых имеет свой собственный адрес. Самое распространенное блок-ориентированное устройство - диск.

**Байт-ориентированные устройства** не адресуемы и не позволяют производить операцию поиска, они генерируют или потребляют последовательность байтов. Примерами являются терминалы, строчные принтеры, сетевые адаптеры.



# Организация программного обеспечения ввода-вывода

Основная идея организации программного обеспечения ввода-вывода состоит в разбиении его на несколько уровней, причем нижние уровни обеспечивают экранирование особенностей аппаратуры от верхних, а те, в свою очередь, обеспечивают удобный интерфейс для пользователей. Основные принципы сводятся к следующим положениям.

# Независимость от устройств

Вид программы не должен зависеть от того, читает ли она данные с гибкого диска или с жесткого диска.

# Единообразное именование

Для именовании устройств должны быть приняты единые правила.

# Обработка ошибок

Ошибки следует обрабатывать как можно ближе к аппаратуре.

Если контроллер обнаруживает ошибку чтения, то он должен попытаться ее скорректировать. Если же это ему не удастся, то исправлением ошибок должен заняться драйвер устройства.

Многие ошибки могут исчезать при повторных попытках выполнения операций ввода-вывода, например, ошибки, вызванные наличием пылинок на головках чтения или на диске.

И только если нижний уровень не может справиться с ошибкой, он сообщает об ошибке верхнему уровню.

# Использование блокирующих (синхронных) и неблокирующих (асинхронных) передач

Большинство операций физического ввода-вывода выполняется асинхронно - процессор начинает передачу и переходит на другую работу, пока не наступает прерывание.

Пользовательские программы намного легче писать, если операции ввода-вывода блокирующие - после команды READ программа автоматически приостанавливается до тех пор, пока данные не попадут в буфер программы.

ОС выполняет операции ввода-вывода асинхронно, но представляет их для пользовательских программ в синхронной форме.

# Наличие разделяемых и выделенных устройств

Диски - это разделяемые устройства, так как одновременный доступ нескольких пользователей к диску не представляет собой проблему.

Принтеры - это выделенные устройства, потому что нельзя смешивать строчки, печатаемые различными пользователями. Наличие выделенных устройств создает для операционной системы некоторые проблемы.

# Многоуровневая организация подсистемы ВВОДА-ВЫВОДА



"Операционные



# Приложения

Приложения (программы) используют для выполнения операций ввода/вывода набор стандартных функций (операторов) типа READ, WRITE.



# Библиотечные функции

Операторы ввода/вывода (READ, WRITE) на этапе компиляции программы вызывают в объектный код подпрограммы из библиотеки стандартных функций.



# Системные вызовы

Библиотечные функции в процессе работы программы порождают системные вызовы, которые через интерфейс прикладного программирования направляются к подсистеме ввода/вывода на дальнейшую обработку.



# Системные вызовы

Библиотечные функции в процессе работы программы порождают системные вызовы, которые через интерфейс прикладного программирования направляются к подсистеме ввода/вывода на дальнейшую обработку.



# Драйверы устройств

Драйвер (driver) представляет собой специализированный программный модуль, управляющий внешним устройством.

Слово driver происходит от глагола to drive (вести) и переводится с английского языка как извозчик или шофер: тот, кто ведет транспортное средство.

Драйверы обеспечивают единый интерфейс для доступа к различным устройствам, тем самым, устраняя зависимость пользовательских программ и ядра ОС от особенностей аппаратуры.



# Обработка прерываний

Прерывания должны быть скрыты как можно глубже в недрах операционной системы, чтобы как можно меньшая часть ОС имела с ними дело.

Наилучший способ состоит в разрешении процессу, инициировавшему операцию ввода-вывода, блокировать себя до завершения операции и наступления прерывания.

