

# Обслуживание ввода-вывода

---

# Подсистема ввода-вывода

---

Одной из главных задач ОС является обеспечение обмена данными между приложениями и периферийными устройствами компьютера. Собственно ради выполнения этой задачи и были разработаны первые системные программы, послужившие прототипами операционных систем. В современной ОС функции обмена данными с периферийными устройствами выполняет подсистема ввода-вывода.

Основными компонентами подсистемы ввода-вывода являются драйверы, управляющие внешними устройствами, и файловая система. К подсистеме ввода-вывода можно также с некоторой долей условности отнести и диспетчер прерываний.

# Организация побайтного ВВОДА-ВЫВОДА

---

Во многих ОС имеется два типа устройств:

- ❑ устройства ввода-вывода блоками
- ❑ устройства неструктурированного или посимвольного ввода-вывода.

Примером блочных устройств ввода-вывода являются диски и ленты.

К устройствам посимвольного (побайтного) ввода вывода относятся практически все остальные устройства, в том числе дисплеи, клавиатуры и сетевое оборудование.

Таким образом, для работы с устройством существуют два драйвера:

- блочный
- посимвольный.

# Организация ввода-вывода с использованием каналов ввода-вывода

---

**Каналы ввода-вывода** (англ. IOC - input-output channel), далее КВВ, и интерфейсы обеспечивают взаимодействие центральных устройств машины и периферийных устройств.

**КВВ** — самостоятельные в логическом отношении устройства, которые работают под управлением собственных программ, находящихся в памяти.

В современных машинах КВВ называют *периферийными процессорами* или *процессорами ввода-вывода*.

КВВ и интерфейсы выполняют следующие функции:

1. Позволяют иметь машины с переменным составом периферийных устройств.
2. Обеспечивают параллельную работу периферийных устройств как между собой, так и по отношению к процессору.
3. Обеспечивают автоматическое распознавание и реакцию процессора на различные ситуации, возникающие в периферийных устройствах.

# Организация ввода-вывода с использованием каналов ввода-вывода

---

Операции ввода - вывода могут выполняться каналом ввода - вывода в одном из двух режимов: *монопольном* или *мультиплексном*.

В **монопольном** режиме периферийное устройство занимает все средства интерфейса ввода-вывода и канала и остаётся функционально связанным с каналом ввода-вывода на всё время передачи данных; никакое другое периферийное устройство не может использовать средства канала ввода-вывода и интерфейса до завершения работы предыдущего устройства.

В **мультиплексном** режиме канал ввода-вывода может обслуживать несколько одновременно работающих периферийных устройств. В этом режиме выполнение операций ввода - вывода расщепляется на короткие интервалы времени, в каждом из которых обмен данными производится с одним из периферийных устройств. Интервалы времени чередуются в соответствии с запросами от периферийных устройств; в течение каждого интервала времени канал ввода-вывода функционально связан с одним периферийным устройством.

# Организация ввода-вывода с использованием каналов ввода-вывода

---

Существует 3 вида КВВ:

*1. Мультиплексный канал.*

Сам канал быстродействующий, но обслуживает медленное периферийное устройство. При этом, подключившись к одному устройству, подаёт одно машинное слово, и после этого подключается к другому.

*2. Селекторный канал.*

Канал быстродействующий и обслуживает быстрые устройства. При этом подключившись к одному устройству, передаёт всю информацию, и после этого подключается к другому устройству.

*3. Блок-мультиплексорный канал.*

Подключившись к одному устройству, передаёт часть информации. После этого подключается к другому устройству.

# Канальная программа

---

Обмен данными между основной памятью и внешними устройствами осуществляется с помощью каналов. **Канал** представляет собой специализированное обрабатывающее устройство (процессор ввода-вывода), предназначенное для работы с устройствами ввода-вывода; как и центральный процессор, он имеет свою систему команд. Каждое из этих устройств - центральный процессор и канал - может выполнять только те команды, которые относятся к системе команд данного устройства. Из канальных команд составляется канальная программа. Одна канальная программа определяет одну операцию обмена и записывается в последовательных ячейках основной памяти машины. В процессе обмена данными канал считывает *команды канальной программы* из основной памяти и обеспечивает их выполнение.

# Система управления вводом-выводом (СУВВ)

---

Система управления вводом-выводом (СУВВ) представляет собой один или несколько системных процессов, обеспечивающих информационное и управляющее взаимодействие между внутренними и внешними процессами. Через эту подсистему происходит инициация, управление и уничтожение внешних процессов. С точки зрения программных процессов пользователей СУВВ представляет собой программный интерфейс с необходимыми для них ПУ. В рамках этого интерфейса пользователь формирует запросы на выполнение следующих действий в отношении ПУ:

- операции чтения и записи данных в отношении адресуемого ПУ;
- операции управления устройством;
- операции по проверке состояния устройства.

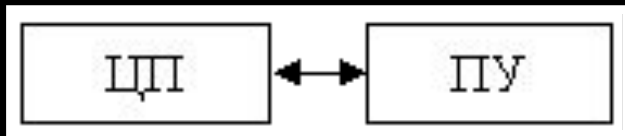


# Методы управления периферийными устройствами

В зависимости от степени автономности от ЦП можно выделить два типа управления ПУ.

**Прямой** метод основан на непосредственной связи ЦП и ПУ и предполагает наличие в составе команд процессора специальных команд по инициированию работы, проверке готовности, останову, записи информации и т.д.

Методы управления периферийными устройствами



Прямой



Косвенный

# Методы управления периферийными устройствами

---

**Косвенный** метод состоит в том, что между ЦП и ПУ помещается канал – специальный процессор, который фактически управляет вводом-выводом. С ЦП снимаются несвойственные ему функции по управлению ПУ, остаются лишь функции управления каналом. ЦП только иницирует ввод-вывод, а затем может выполнять свои программы (до момента окончания процесса ввода-вывода). При этом ЦП, канал и ПУ по мере развития внешнего процесса работают параллельно.

Для синхронизации параллельной работы ЦП и канала используют различные средства. В простейшем случае это флажок, в других случаях ЦП может быть доступна расширенная статусная информация о состоянии канала, контроллера и устройства. Такие средства предполагают некоторую периодичность проверок занятости канала со стороны ЦП.

Более совершенным механизмом является использование прерываний. Канал через систему прерываний прерывает работу ЦП всякий раз при завершении операции ввода-вывода или при возникновении ошибки. Здесь сигнал прерывания является по смыслу синхронизирующим, т.к. используется для оповещения определенного программного процесса о событии, которое произошло при работе канала или ПУ (например, при завершении печати страницы на принтере, ошибке записи на диск и т.д.).

# Методы управления периферийными устройствами

---

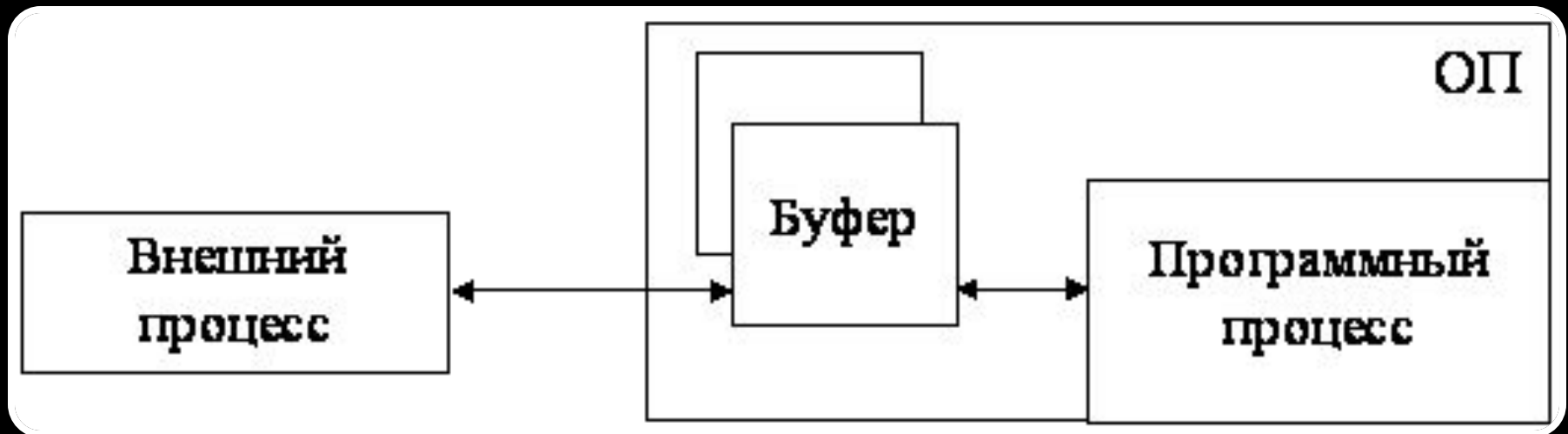
При возникновении прерывания ЦП временно «отвлекается» от основной работы. В соответствии с централизованной схемой управления ПУ после определения причины прерывания управление передается системной программе управления вводом-выводом – *супервизору ввода-вывода*. При оповещении через прерывание о событии в некотором внешнем процессе супервизор ввода-вывода планирует и осуществляет через канал дальнейшие действия по организации ввода-вывода (обновление данных, инициирование следующей операции и т.д.).

Через буфер данные либо посылаются от некоторого программного процесса к адресуемому внешнему, либо от внешнего процесса передаются программному. На супервизор ввода-вывода возлагаются функции выделения и уничтожения буферов в оперативной памяти, определения их количества, размеров и назначения (для ввода или для вывода).

Супервизор ввода-вывода производит синхронизацию программных и внешних процессов, взаимодействующих через буфер – устраняет возможность одновременного обращения этих процессов к буферу.

# Супервизор

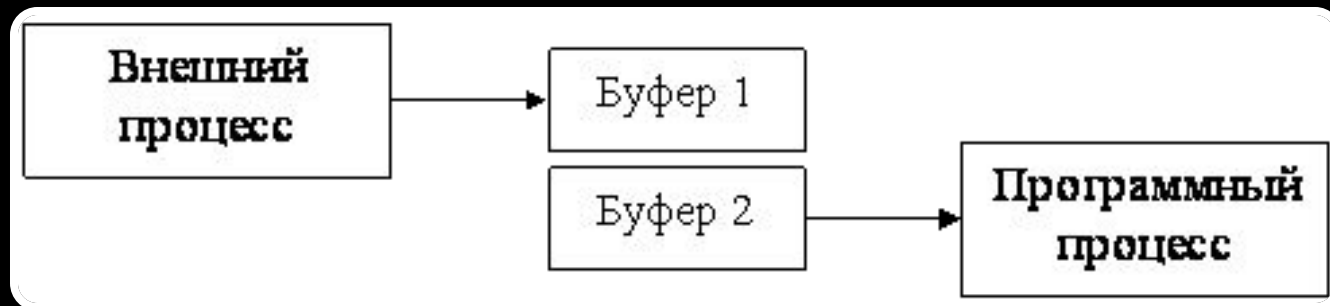
Использование буферов для организации информационного взаимодействия внешнего и программного процессов.



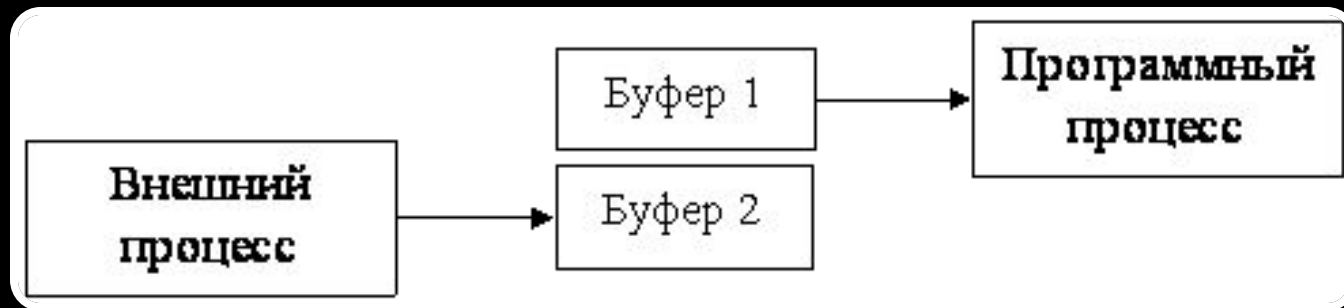
# Супервизор

Для устранения задержек в ожидании наполнения буфера используется несколько буферов. Например, с точки зрения временных затрат для операции чтения рационально использовать два буфера. Пока один из них наполняется, другой в это время освобождается.

А)



Б)



Поочередное использование двух буферов.

# Алгоритм обработки прерываний по вводу-выводу

---

Прерывание вызывает ряд событий, которые происходят как в аппаратном, так и в программном обеспечении. На рисунке показана типичная последовательность этих событий. После завершения работы устройства ввода-вывода происходит следующее:

- Устройство посылает процессору сигнал прерывания.
- Перед тем как ответить на прерывание, процессор должен завершить исполнение текущей команды.
- Процессор производит проверку наличия прерывания, обнаруживает его и посылает устройству, приславшему это прерывание, уведомляющий сигнал об успешном приеме. Этот сигнал позволяет устройству снять свой сигнал прерывания.

Далее в программный счетчик процессора загружается адрес входа программы обработки прерываний, которая отвечает за обработку данного прерывания. В зависимости от архитектуры компьютера и устройства операционной системы может существовать как одна программа для обработки всех прерываний, так может быть и своя программа обработки для каждого устройства и каждого типа прерываний. Если для обработки прерываний имеется несколько программ, то процессор должен определить, к какой из них следует обратиться. Эта информация может содержаться в первоначальном сигнале прерывания; в противном случае для получения необходимой информации процессор должен по очереди опросить все устройства, чтобы определить, какое из них отправило прерывание.

## Аппаратное обеспечение



## Программное обеспечение



# Подсистема ввода-вывода

---

Подсистема ввода-вывода ОС при обмене данными с внешними устройствами должна решать ряд общих задач:

- Организация параллельной работы устройства ввода-вывода и процессора;
- Согласование скоростей обмена и кэширования данных;
- Разделение устройств и данных между процессами;
- Обеспечение удобного логического интерфейса между устройствами и остальной частью системы;
- Поддержка широкого спектра драйверов с возможностью простого включения в систему нового драйвера;
- Динамическая загрузка и выгрузка драйверов;
- Поддержка файловых систем;
- Поддержка синхронных и асинхронных операций ввода-вывода.



# Поддержка широкого спектра драйверов с возможностью простого включения в систему нового драйвера

Достоинством подсистемы ввода-вывода операционной системы является разнообразие устройств, поддерживаемых данной ОС.

Для создания драйверов необходимо наличие удобного и открытого интерфейса между драйверами и другими компонентами ОС.

Драйвер взаимодействует, с одной стороны, с модулями ядра ОС, а с другой стороны – с контроллерами внешних устройств. Драйвер имеет два интерфейса:

1. DKI (driver kernel interface)
2. DDI (driver device interface).



# Динамическая загрузка и выгрузка драйверов

---

Другой проблемой работы с устройствами ввода-вывода является проблема включения драйвера в состав работающей ОС – динамическая загрузка/выгрузка драйверов.

Способность системы автоматически загружать и выгружать из оперативной памяти требуемый драйвер повышает универсальность ОС.

Альтернативой динамической загрузке драйверов при изменении текущей конфигурации внешних устройств является повторная компиляция кода ядра с требуемым набором драйверов. Пример – некоторые версии UNIX.

# Поддержка файловых систем

---

Внешняя память вычислительной системы представляет собой периферийные устройства, на которых хранится большая часть пользовательской информации и системных данных.

Для организации хранения информации на внешних носителях используется файловая модель.

Для обеспечения доступа к данным используется специальный программный слой, обеспечивающий поддержку работы с конкретной файловой системой – драйверы файловой системы.

Для обеспечения возможности работы с несколькими файловыми системами применяется подход, основанный на применении специального слоя, с которым взаимодействуют приложения ОС – например, слой VFS (virtual file system) в некоторых версиях UNIX.

# Поддержка синхронных и асинхронных операций ввода-вывода

---

Операции ввода-вывода по отношению к программному приложению выполняются в синхронном или асинхронном режимах.

**Синхронный режим** — приложение приостанавливает свою работу и ждет отклика от устройства.

**Асинхронный режим** — приложение продолжает работу, параллельно с ожиданием отклика от устройства.

Операционные системы для разных приложений должны обеспечить синхронную и асинхронную работу с устройствами.

# Менеджеры ввода-вывода

---

Для координации работы драйверов в подсистеме ввода-вывода выделяется специальный модуль, называемый менеджером ввода-вывода.

Верхний слой менеджера составляют системные вызовы ввода-вывода, которые получают запросы от приложений и переадресуют их определенным драйверам.

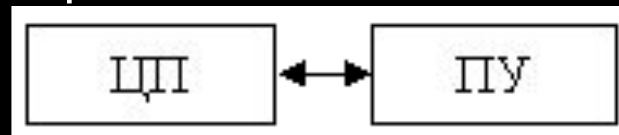
Нижний слой реализует взаимодействие с контроллерами внешних устройств, экранируя драйверы от особенностей аппаратной платформы компьютера.

Еще одна функция менеджера ввода-вывода – организация взаимодействия модулей ввода-вывода с модулями других подсистем (управление процессами, виртуальной памятью и т. д.).

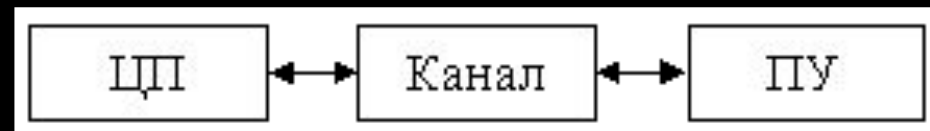
# КОНТРОЛЬНЫЕ ВОПРОСЫ

---

1. Что такое супервизор?
2. Для чего предназначены КВВ?
3. Как вы думаете какой режим работы является усовершенствованным: А)



Б)



4. Какие есть основные задачи у подсистемы ввода-вывода?
5. Закончите предложения:
  - 1) Сегодня я узнал ....
  - 2) До этого я не знал ....., но сегодня узнал, что ....
  - 3) Было интересно ....