

Кафедра “компьютерные системы и технологии”

курс

Системы ввода-вывода, интерфейсы
и периферия компьютеров

часть 1

СИСТЕМЫ ВВОДА-ВЫВОДА И ИНТЕРФЕЙСЫ

Лектор профессор Чернышев Ю.А.

Тема 1.1.

Основные принципы построения систем ввода вывода и интерфейсов

Роль и место систем ввода-вывода и интерфейсов в компьютере

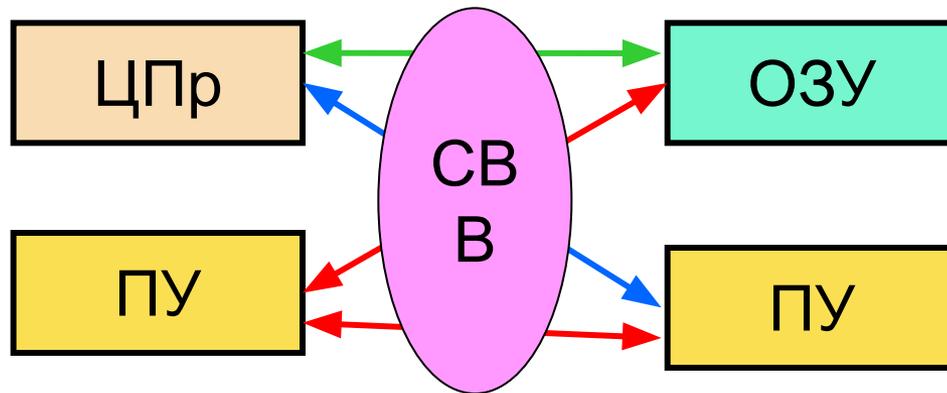


Рис.1.1. Информационное взаимодействие компьютера и внешней среды

Что такое система ввода-вывода, периферия и интерфейсы?

Система ввода-вывода (СВВ) – средства взаимосвязи и передачи информации, объединяющих между собой основные устройства компьютера:

- Центральный Процессор (ЦПр),
- Оперативное Запоминающее Устройство (ОЗУ),
- Периферийные (Внешние) Устройства (ПУ).

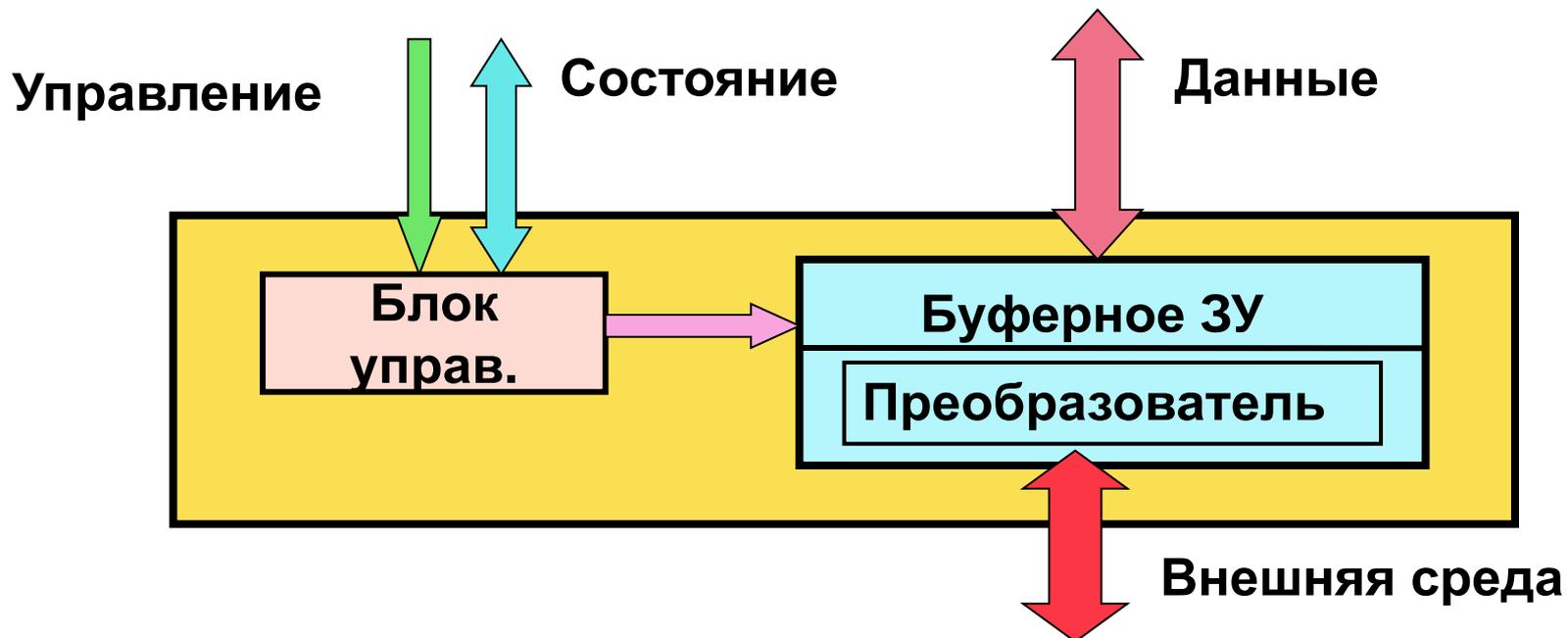


Периферийные устройства (внешние устройства)

ПУ можно разделить на две категории :

- внешние запоминающие устройства (ВЗУ);
- устройства ввода – вывода (УВВ), средства для связи компьютера с пользователями (внешним миром).

Обобщенная структура УВВ



Функционально – модульная структура компьютера

Компьютер можно представить в виде системы, состоящей из взаимосвязанных функционально законченных и конструктивно оформленных модулей : микропроцессора, ОЗУ, контроллеров (БИС, платы расширения), ВЗУ, монитор, печать и другие ПУ.

Связь между ними должна осуществляться СВВ с помощью стандартных технических средств и правил взаимодействия, т.е. стандартных интерфейсов.



Рис.1.1. Информационное взаимодействие компьютера и внешней среды

ИНТЕРФЕЙС

Средства и правила, обеспечивающие взаимосвязь устройств между собой, называются интерфейсом.

В интерфейсе стандартизируются:

- 1. Назначение и количество линий интерфейса.**
- 2. Параметры электрических сигналов.**
- 3. Протоколы обмена информацией и выполнения функций интерфейса.**
- 4. Конструктивные параметры.**

Физически интерфейс реализуется в виде электрических линий для передачи сигналов и набора микросхем, обеспечивающих выполнение основных функций интерфейса.

Некоторые понятия интерфейсов

Канал – среда передачи информации, представляемой в виде определенных сигналов. Канал реализуется с помощью тех или иных средств, зависящих от физической природы сигналов (ток, напряжение, радиосигнал, оптический сигнал и т.п.). Компьютерный интерфейс является каналом передачи информации с помощью электрических сигналов (импульсов и потенциалов).

Линия интерфейса – это электрический проводник (провод, линия печатного монтажа, контакт разъема платы), по которому распространяется электрический сигнал. При дифференциальном представлении сигналов линия содержит два проводника.

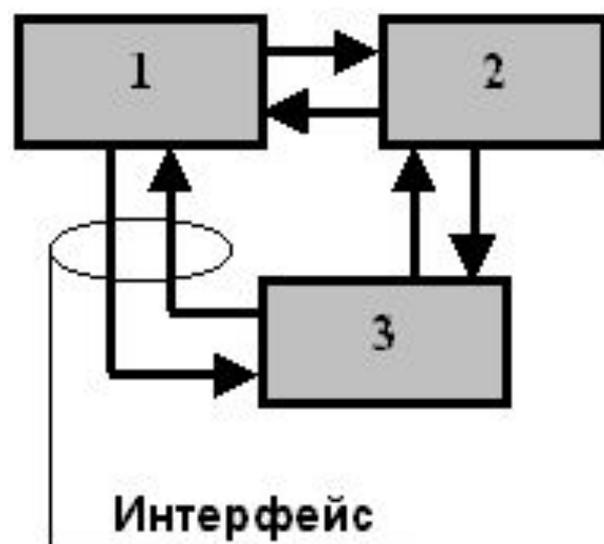
Магистраль – это совокупность всех линий интерфейса.

Шина – группа линий интерфейса, соответствующая определенному функциональному назначению (шина данных, шина адреса и т.п.)

В дальнейшем термин «магистраль» стали заменять термином «шина». В свою очередь понятие «шина» стало в определенной мере синонимом термина «интерфейс», хотя понятие шины более узкое, чем общее понятие интерфейса

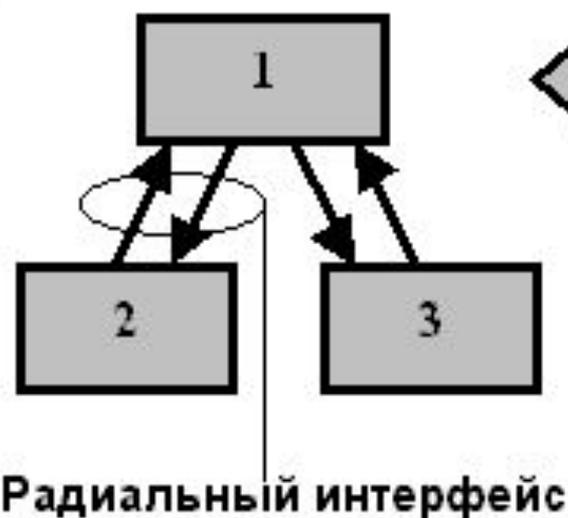
А) Каждый с каждым

Модуль 1



Б) Радиальный

Главный модуль



В) Магистральный

Общая магистраль

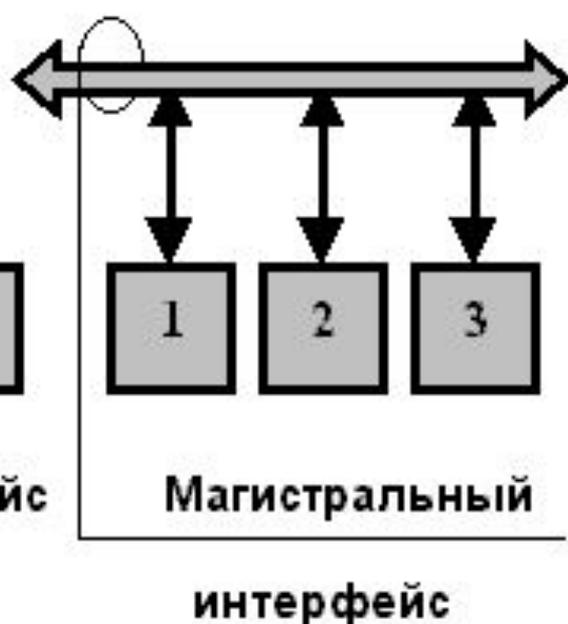
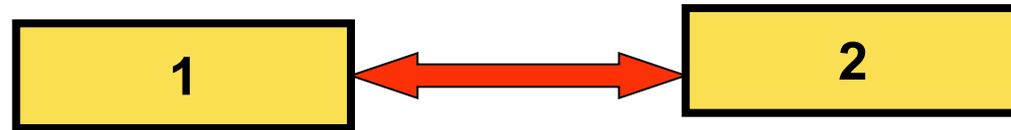


Рис. 1.2. Способы соединения модулей

Топология интерфейсов

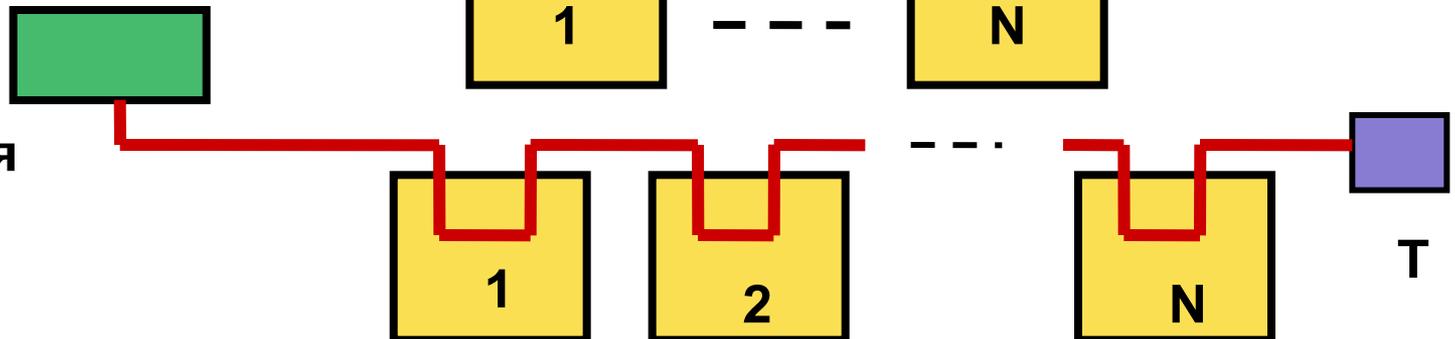
1.Точка – точка



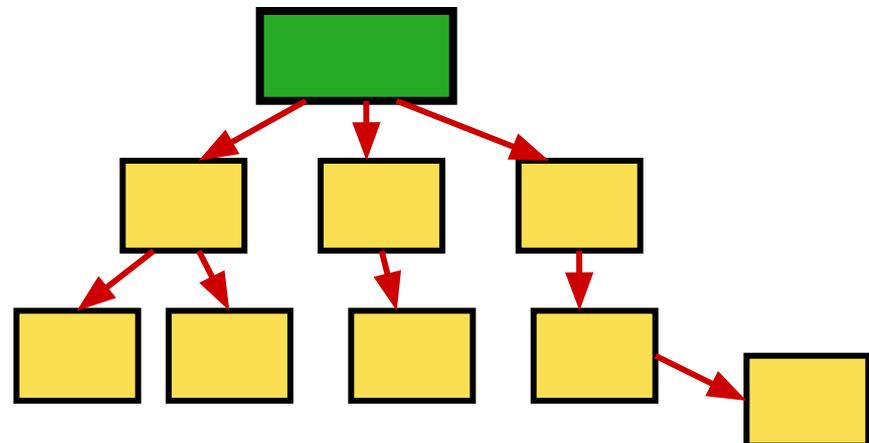
2. Магистральная



Шлейфовая



3. Многоуровневая звезда



Классификация интерфейсов

Существует множество стандартных компьютерных интерфейсов различного назначения, обладающих своими особенностями и характеристиками. Их можно классифицировать по различным признакам.

1. По способу передачи данных.

а) **Параллельные** – одновременная синхронная передача всех или части разрядов, например байта, слова, кода. Шина данных имеет столько линий сколько одновременно передается разрядов.

б) **Последовательные** – последовательная, синхронная передача бит данных. Для данных требуется всего одна линия.

Последовательные интерфейсы проще и дешевле, чем параллельные.

Для повышения производительности последовательных интерфейсов используют одновременную, параллельную передачу группы разрядов слова по нескольким последовательным каналам

2. По организации обмена.

- а) **Симплексный** – передача в одну сторону.
- б) **Полудуплексный** – передача в две стороны, но в разные моменты времени, по одним и тем же линиям.
- в) **Дуплексный** – передача в две стороны одновременно. Требуются свои линии в каждую сторону.

3. По способу синхронизации.

- а) **Синхронный** – все сигналы интерфейса переключаются по фронту синхроимпульсов.
- б) **Асинхронный** – сигналы могут переключаться внутри такта синхроимпульсов, а не только по их фронту.

3. ПО архитектуре интерфейсов.

1. Параллельный интерфейс:

а) **трехшинный** – шина данных, шина адреса, шина управления,

б) **двухшинный** – шина адреса / данных, шина управления.

Двухшинный проще, но менее производителен.

2. Последовательный интерфейс:

а) без специальных линий управления,

б) со специальными линиями управления.

4. По способу реализации.

а) **Внутренние**, для объединения электронных модулей системного блока:

-- **шина «процессор – память»**, шина переднего плана - FSB (Front-Side Bus) связь МПР и ОЗУ, скорость работы наибольшая.

-- **шина для связи МПР с кэш-памятью** второго уровня, шина заднего плана – BSB (Back – Side Bus), она быстрее, чем FSB.

- **шины ввода –вывода** (шины расширения) - предназначены для соединения ЦПР с ПУ. Они менее производительны, большего размера, важным является их стоимость и конструктив. Являются одной из основ системы ввода – вывода.
- б) **Внешние (интерфейсы ПУ)** – предназначены для подключения ПУ к системе ввода – вывода. Они могут быть :
 - **универсальные**, для подключения разных типов ПУ,
 - **специализированные** – для подключения одного типа ПУ.

Скорость передачи информации в интерфейсах

Скорость передачи измеряется в бит/сек или байт/сек.

Будем считать, что за один синхроимпульс передается слово или бит. Тогда :

при частоте синхроимпульсов f скорость передачи

- параллельный интерфейс, при N - число бит, n - байт в слове

$V_{\text{пар}} = f_{\text{пар}} N$ бит/сек = $f_{\text{пар}} n$ байт/сек, $f_{\text{пар}} < 150$ МГц

- последовательный интерфейс $V_{\text{пос}} = f_{\text{пос}}$ бит/сек

Сейчас все шире используются последовательные интерфейсы из-за их простоты и дешевизны и высоком значении $f_{\text{пос}} > 2$ ГГц

Организация передачи информации между ЦПр и ПУ

Все устройства, взаимодействующие с помощью интерфейсов, делятся на два вида : задатчики (активные устройства), начинают и управляют процессом ввода-вывода, исполнители (пассивные устройства), они принимают (вывод, запись) или выдают(ввод, чтение) информацию и выполняют приказы задатчика.

Устройство, выдающее информацию, является источником, а принимающее – приемником.

ЦПр всегда задатчик, а ОЗУ всегда исполнитель. ПУ и то и другое. Передача информации от задатчика к исполнителю реализуется операцией записи, а обратная – операцией чтения. Процесс передачи между ПУ и ЦПр называют вводом-выводом информации.

Для минимизации загрузки процессора при выполнении процессов ввода-вывода функции управления распределяются на несколько устройств: процессор, главный контроллер (канал) ввода-вывода, контроллер ПУ и блок управления ПУ.

Основные принципы управления передачей информации в вычислительных системах

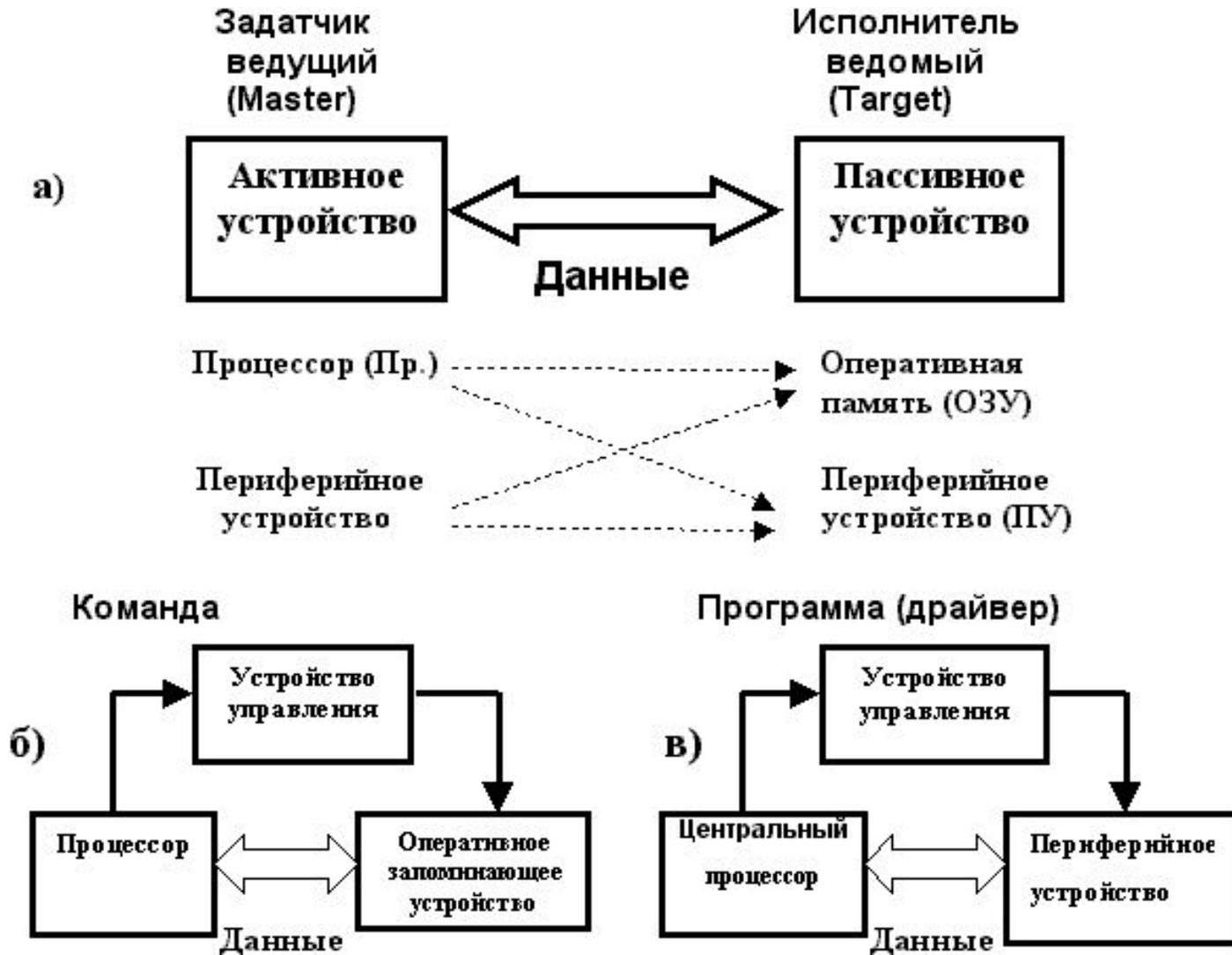


Рис.1.3. Управление обменом информации

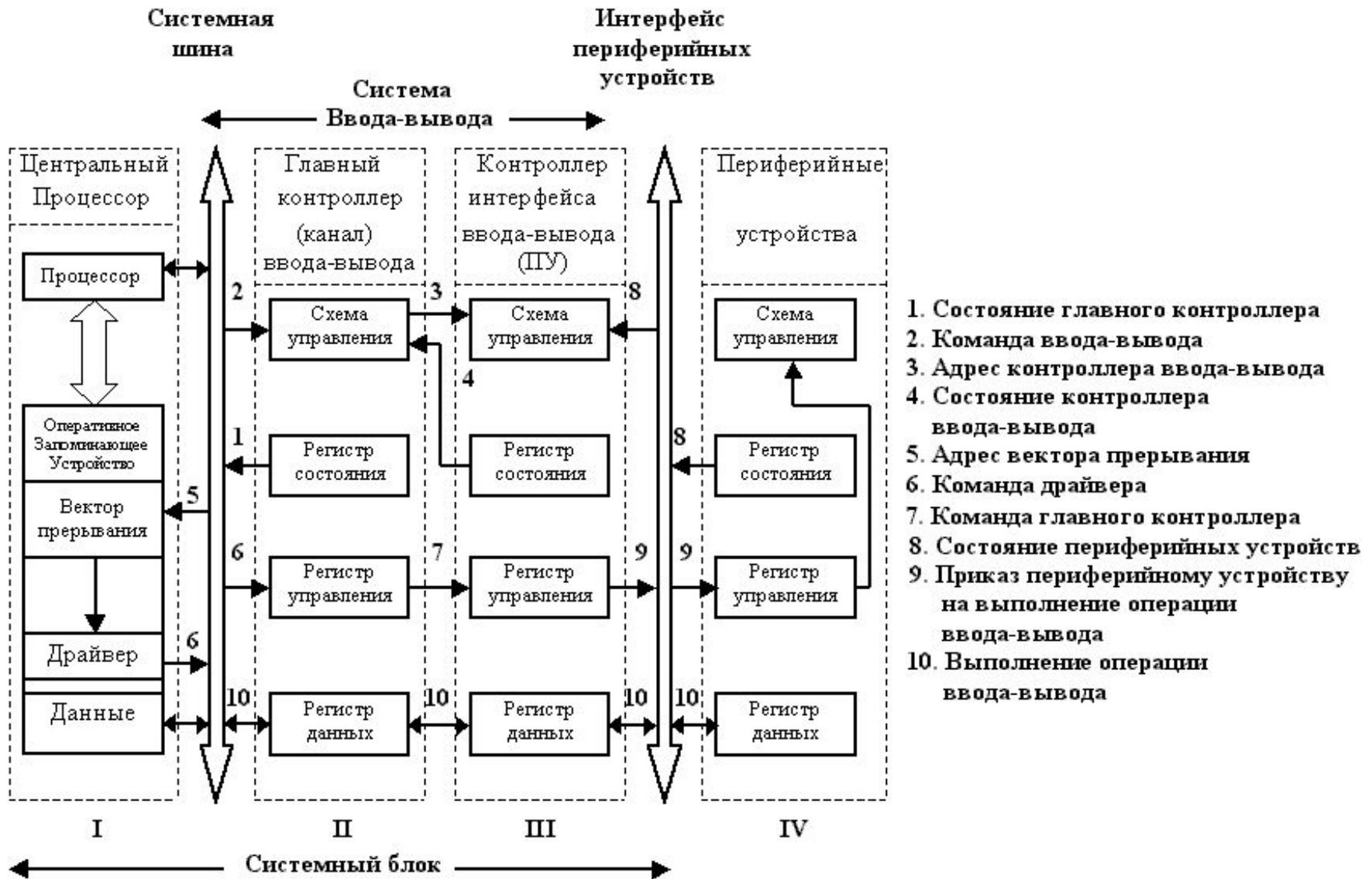


Рис. 1.4. Функциональная схема организации передачи информации в компьютере

Интерфейсы периферийных устройств

Шина	Пропускная способность	Способ передачи	Режим передачи	Разрядность данных	Частота МГц
RS-232с	от 50бит/сек до 115 Кбит/сек	последовательный	дуплексный	1	
Centronics SPP	100-150 КБайт/сек	параллельный	симплексный	8	
IEEE1284 EPP	0,5-2 Мбайт/сек	параллельный	полудуплексный	8	
IEEE1284 ECP	до 4 Мбайт/сек	параллельный	полудуплексный	8	
SCSI-3 SPI	5; 10; 20; 40; 80; (160) Мбайт/сек	параллельный	полудуплексный	8/16/32	5,10,20, 40
SCSI-3 Fibre channel SCSI	100 Мбайт/сек	последовательный	полудуплексный	1	
IEEE1394 FireWare	100; 200; 400 Мбит/сек	последовательный	полудуплексный	1	
USB	1,5; 12 Мбит/сек	последовательный	полудуплексный	1	
ATA	от 2 до 33 Мбайт/сек	параллельный	полудуплексный	8; 16	

ТАБЛИЦА 4

Структура систем ввода-вывода

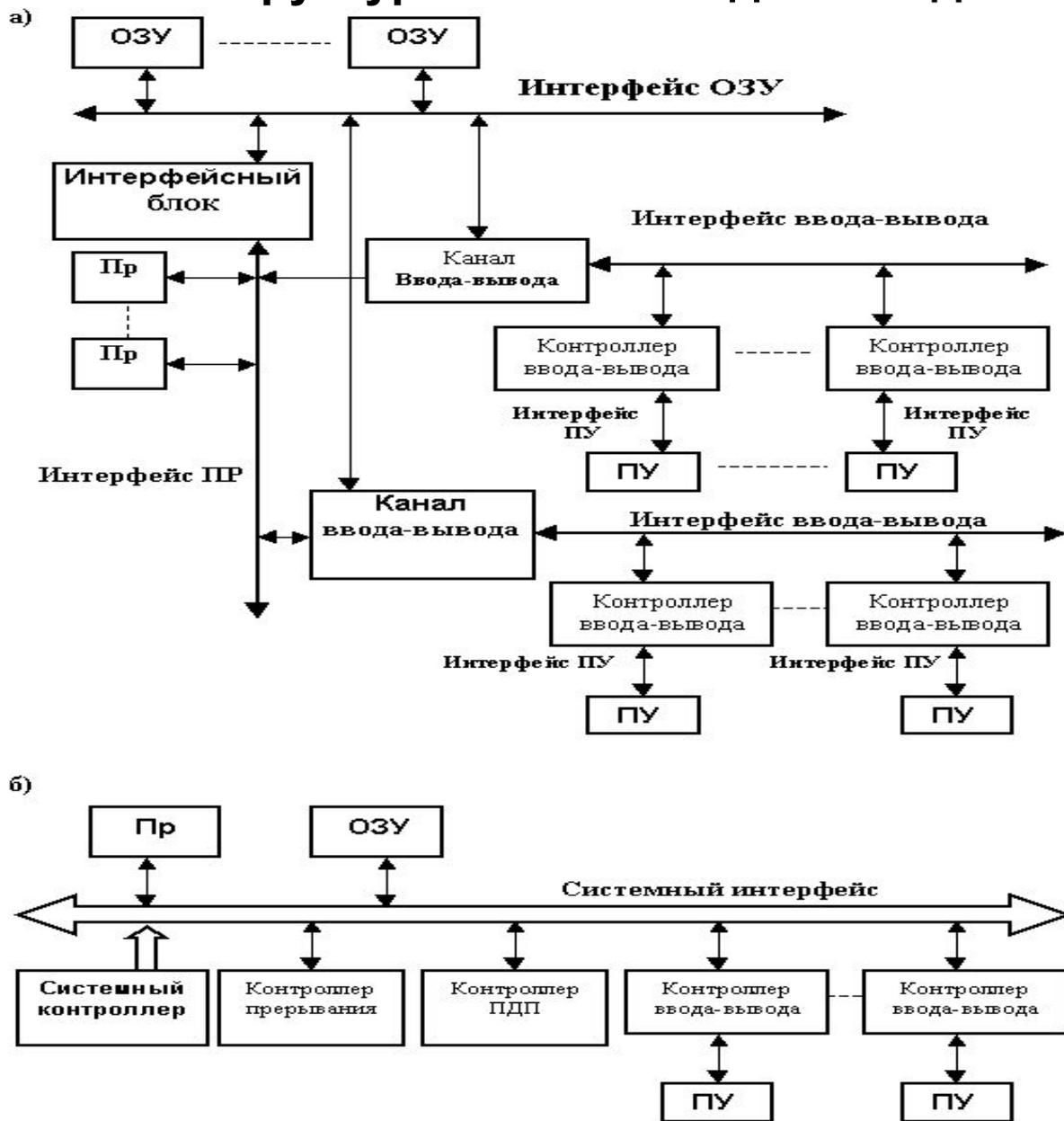


Рис. 1.8. Структура систем ввода-вывода

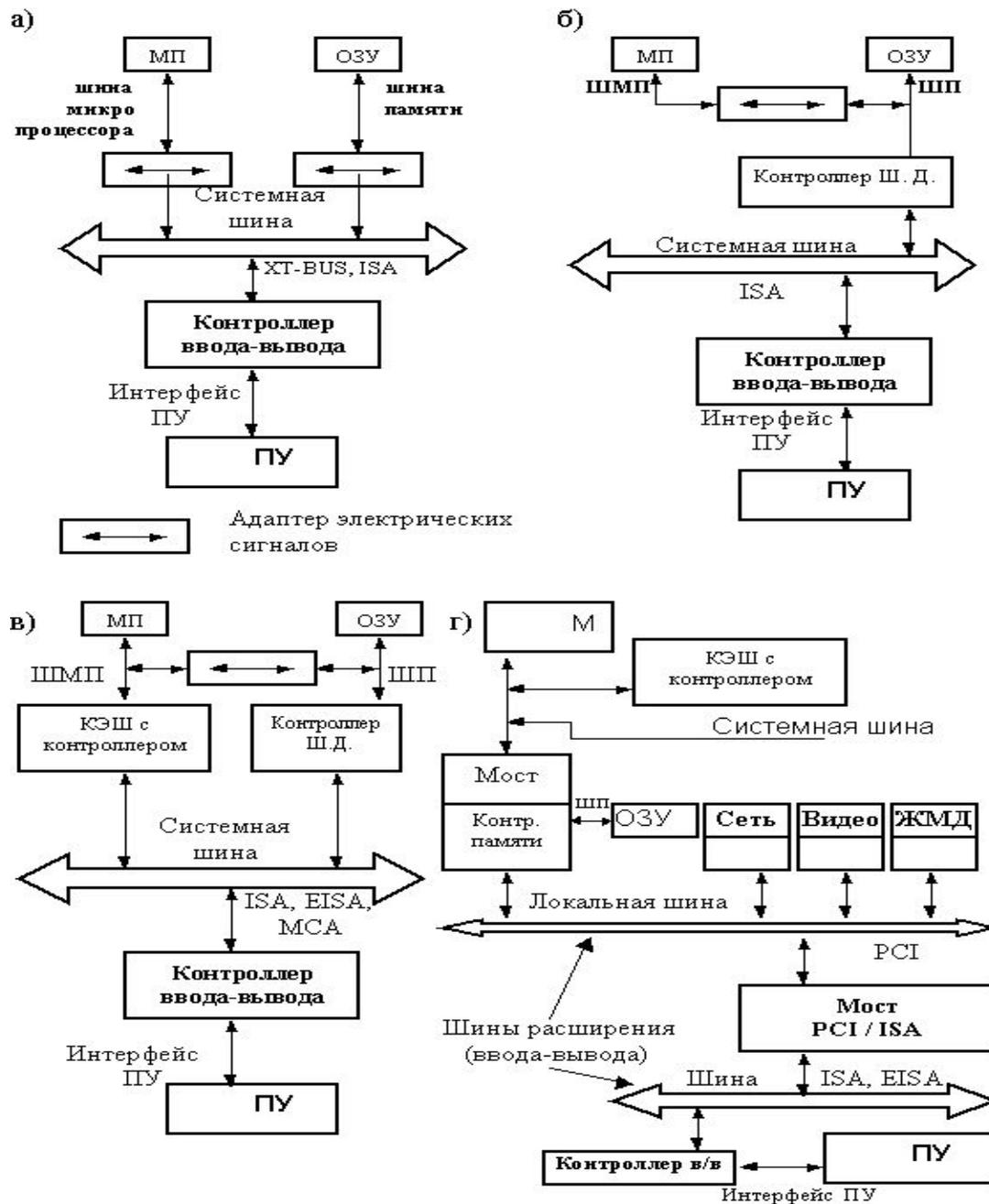


Рис. 1.9. Эволюция шинной архитектуры

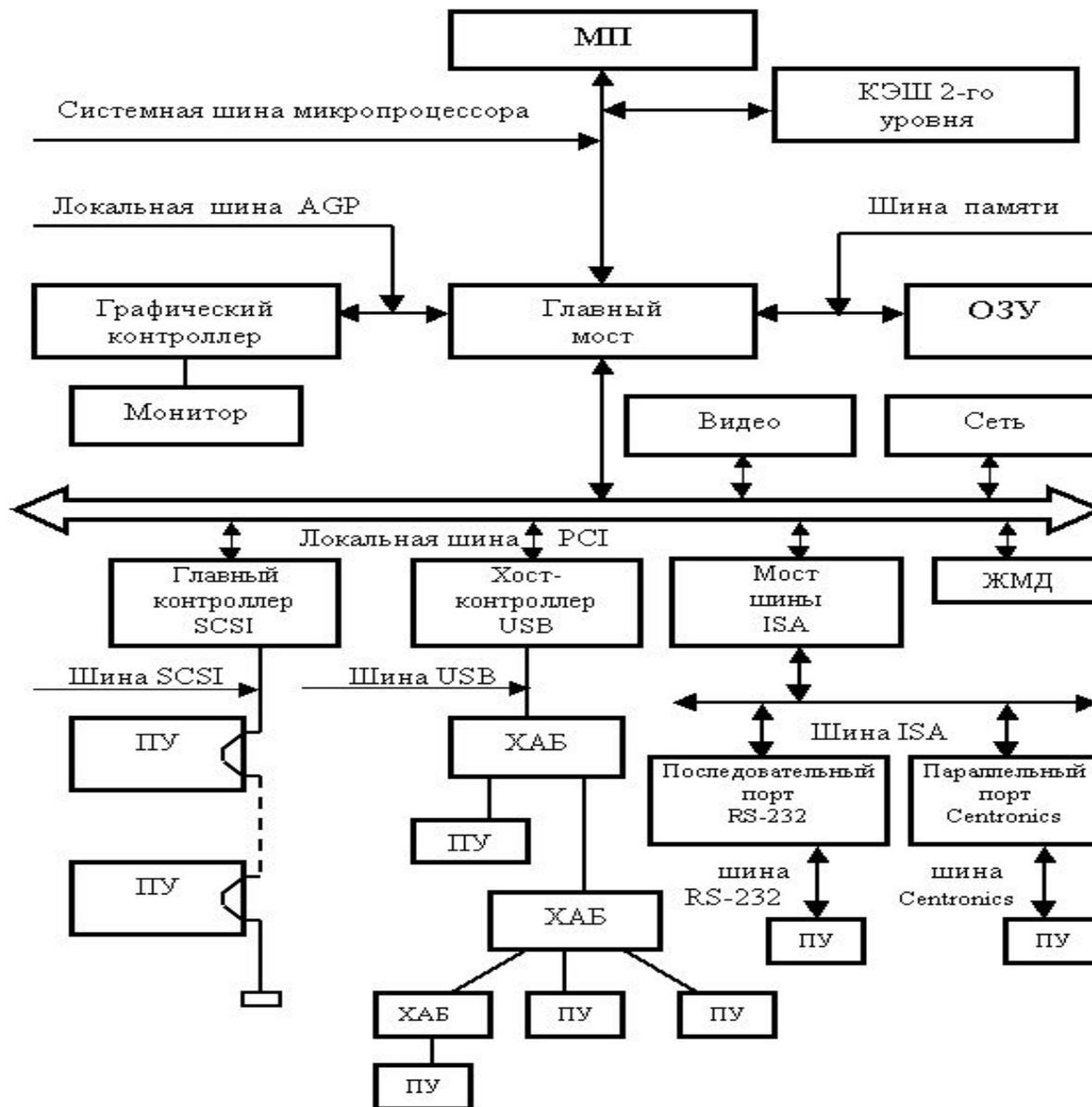


Рис. 1.10. Обобщенная структура компьютера

Тема 1.2.

**Основные функции и принципы
построения интерфейсов**

Основные функции интерфейсов

1. Передача информации.
2. Синхронизация передачи информации.
3. Арбитрация, селекция.
4. Контроль функционирования интерфейса.
5. Преобразование информации.
6. Автоконфигурация.
7. Управление питанием.
8. Горячее подключение.

Функция передача информации

Главная задача интерфейса -- организация надежной передачи информации **от источника к приемнику** в заданный момент времени. Если к одному интерфейсу с помощью специальных средств физически подключается множество взаимодействующих между собой устройств, то в каждый момент времени передача информации идет только между двумя устройствами от источника к приемнику, остальные отключены от шины.

Такая передача возможна, если между этими двумя устройствами в данный момент времени включена электрическая цепь (электрическая взаимосвязь), соединяющая выходы источника со входами приемника, по которой передаются электрические сигналы, соответствующие данной информации.

Таким образом в интерфейсах между двумя взаимосвязанными устройствами постоянно существует логический канал, который или открыт для передачи информации, образуя соединение типа «точка-точка», или закрыт, отключая устройство от шины.

Для переключения логического канала из закрытого в открытый и наоборот, а также для передачи и приема данных требуется определенное время.

Состояние линий интерфейса

Линия интерфейса может находиться в двух состояниях:

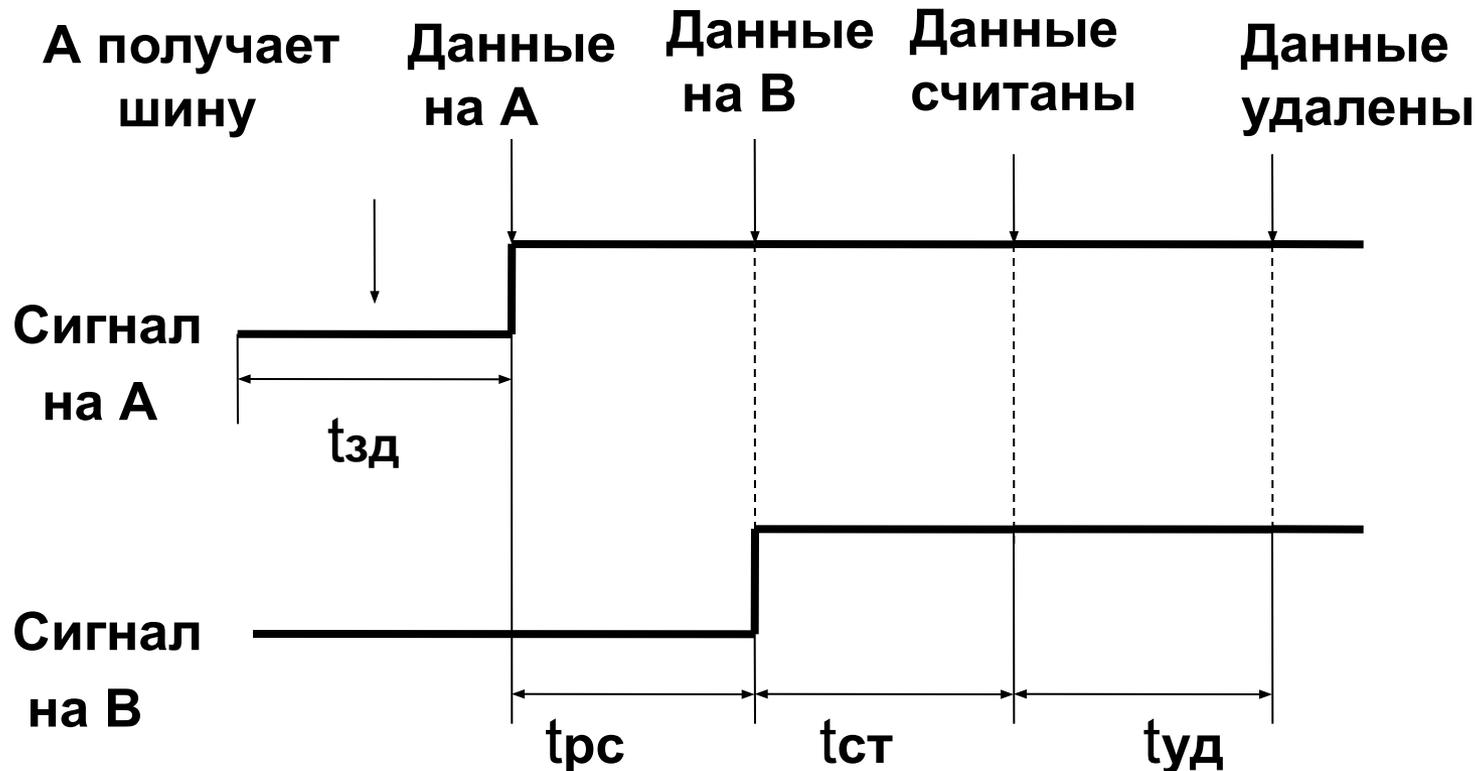
- активном, когда по линии передаются либо двоичные разряды передаваемой информации («0» или «1»), либо сигналы управления, имеющие или высокий или низкий уровни, как правило активному управляющему сигналу соответствует низкий уровень;

- состоянии холостого хода, когда на линии отсутствуют какие либо сигналы, т.е. передачи информации нет. Это состояние линий идентифицируется с помощью набора определенных управляющих сигналов на линиях шины управления, или определенным физическим состоянием линий.

Считывание приемником значений разрядов производится в момент времени, когда сигналы, соответствующие 0 или 1 примут стабильное значение, в этот момент времени подается специальный строб импульс, по которому информация записывается в приемник.

Время передачи информации

Пусть : А - источник, В - приемник.



$t_{зд}$ - задер. захв. шины, $t_{рс}$ - распр. сигн., $t_{ст}$ -стабильный сигнал

$t_{уд}$ - удерж. стаб. сиг., $t_{п}$ - время пер. **$t_{п} = t_{зд} + t_{рс} + t_{ст} + t_{уд}$**

Транзакции и операции интерфейса

Данные по линиям интерфейса передаются блоками. Блок, содержащий в начале адрес первого слова, а затем сами данные называется *транзакцией*. Использование транзакций повышает производительность интерфейса за счет однократной передачи адреса. Транзакция имеет признаки ее начала и конца.

При выполнении транзакции чтения после установки адреса требуется время на переключение направления передачи информации : было от задатчика к исполнителю, требуется от исполнителя к задатчику.

Передача информации по шине выполняется с помощью специальных команд (операций) интерфейса, каждой из которых соответствует определенный протокол работы шины. Основными являются операции ЗАПИСИ - WR(write) и ЧТЕНИЯ – RD(read). Они разные при обращении к разным полям памяти: ОЗУ, регистры ввода – вывода, регистры автоконфигурации.

Функция синхронизации

При передаче информации источник должен знать, когда можно начинать передачу, а приемник должен знать, когда началась передача информации. Это делается с помощью системы синхронизации.

Синхронизация – согласование процессов взаимодействия при передаче информации от источника к приемнику.

В зависимости от типа единицы информации можно выделить три уровня синхронизации:

1. На уровне бит – реализуется аппаратно.
2. На уровне слов (байт) - реализуется аппаратно.
3. На уровне массивов – реализуется программно.

Два основных принципа синхронизации:

Синхронный – когда смена состояний источника и приемника взаимонезависима и выполняется через одинаковые фиксированные интервалы времени.

Два основных принципа синхронизации:

Синхронный – когда смена состояний источника и приемника взаимонезависима и выполняется через одинаковые фиксированные интервалы времени, если в заданный момент времени передача не произошла, информация теряется. Т.е. задается постоянный фиксированный период синхронизации $T_{\text{син}}$.

$$T_{\text{син}} \geq t_{\text{рл}} + t_{\text{расп}} + t_{\text{фик}}$$

$t_{\text{рл}}$ – время распространения сигнала в линии

$t_{\text{расп}}$ – время распознавания

$t_{\text{фик}}$ – время фиксации сигнала

$T_{\text{син}}$ – определяется по самому медленному устройству.

Асинхронный – когда смена состояний источника и приемника взаимозависимы, т.е. момент смены состояния источника зависит от момента смены состояния приемника. В этом случае $T_{\text{син}}$ является переменным и зависит от быстродействия устройства. Здесь требуется обратная связь от приемника к источнику.

Последовательная передача данных

Синхронный принцип синхронизации

Использование специальной линии синхронизации от генератора синхроимпульсов (ГСИ), задающего период синхронизации $T_{\text{син}}$.



Использование двух ГСИ на источнике и приемнике, запускаемых одновременно при начале передачи данных



Использования принципа самосинхронизации. При этом сигналы 0 и 1 физически должны быть разными. Например 1 – изменение сигнала от низкого до высокого уровня, а 0 – наоборот.

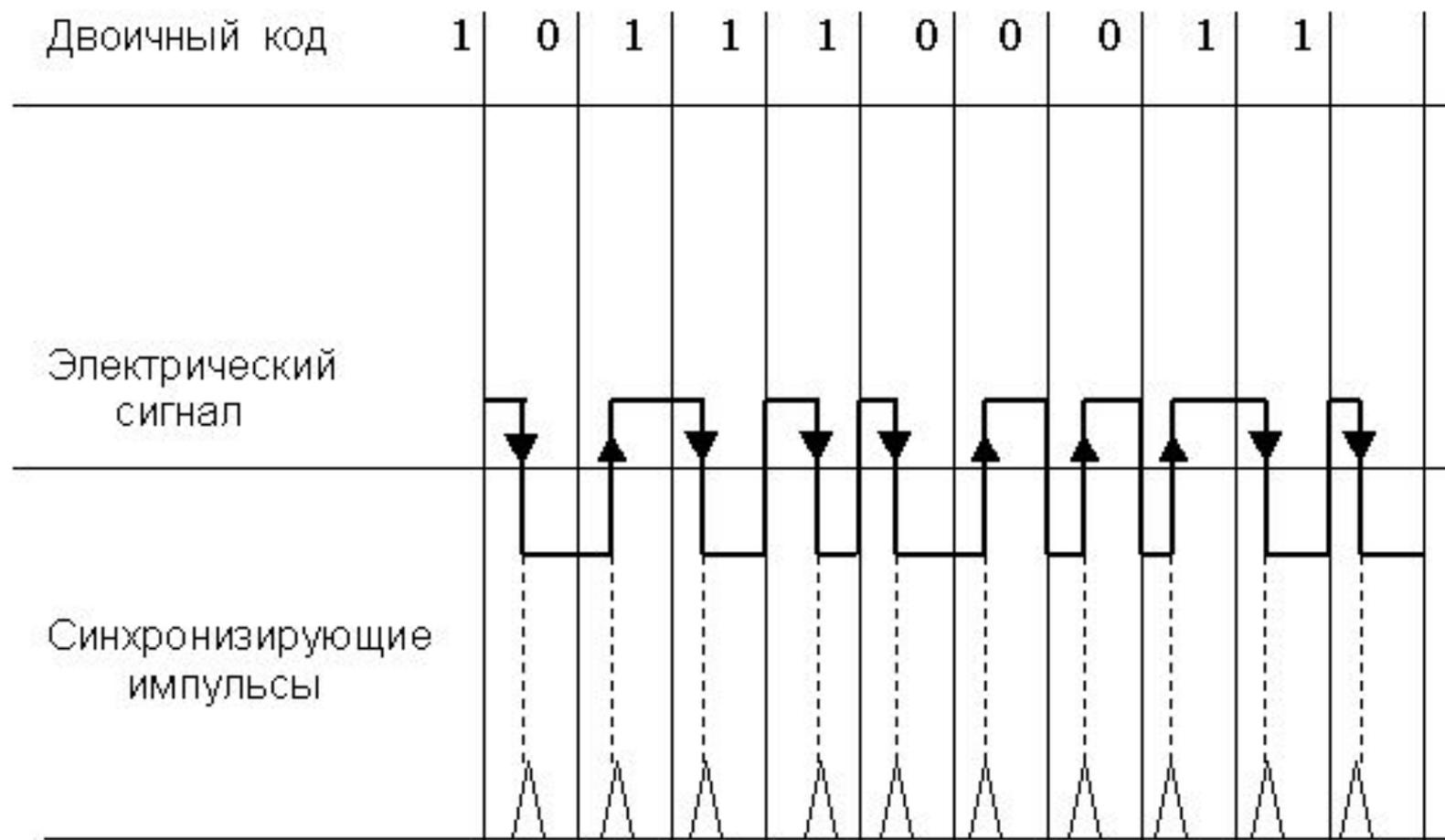


Рис. 1.14. Самосинхронизация

Асинхронный принцип синхронизации

Используется при передаче блока (массива) данных. Начало и конец блока помечается специальными кодами или сигналами.

Например, старт и стоп битами при передаче символа, в состоянии холостого кода линия находится в состоянии 1.

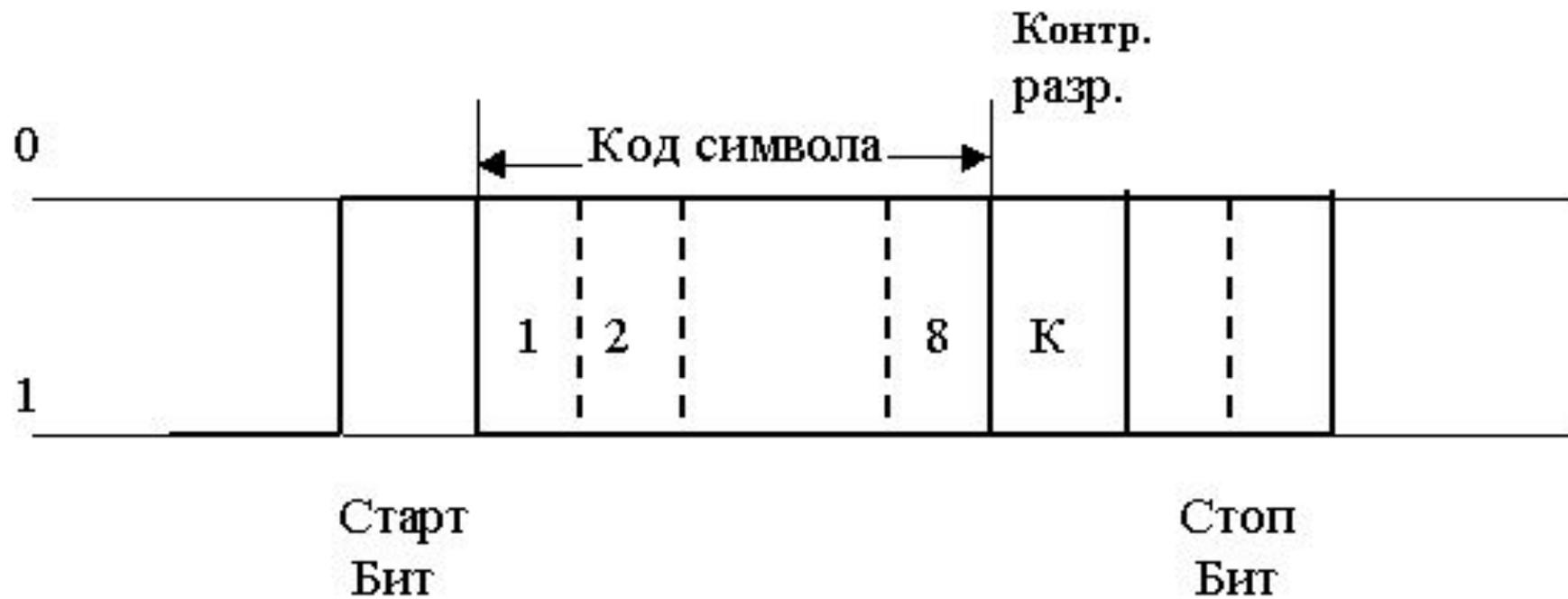


Рис 1.15 Асинхронная передача символа

Параллельная передача данных

При такой передаче обязательно используется строб – импульс, посредством которого считываются(записываются) одновременно – но все разряды передаваемого слова(кода), когда все сигналы разрядов примут стабильное значение.

При синхронном принципе синхронизации период синхронизации задает строб – импульс.

При асинхронном принципе синхронизации кроме stroba от приемника к источнику подается сигнал готовности приемника.

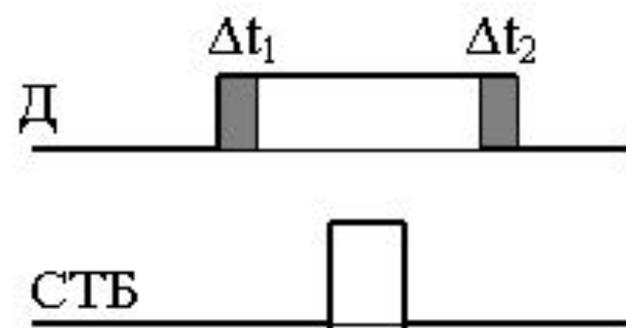


Рис. 1.16. Синхронная параллельная передача данных

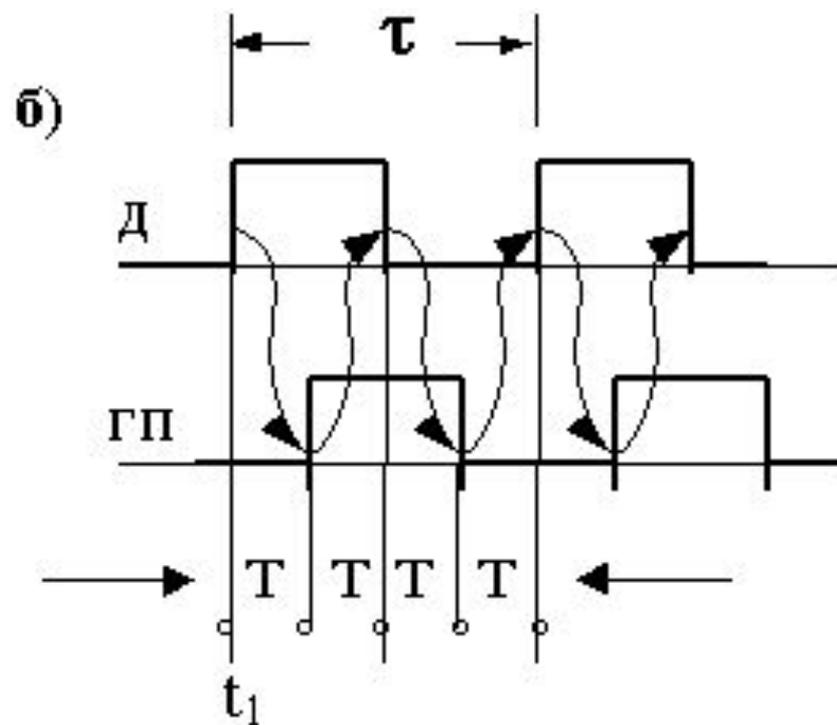
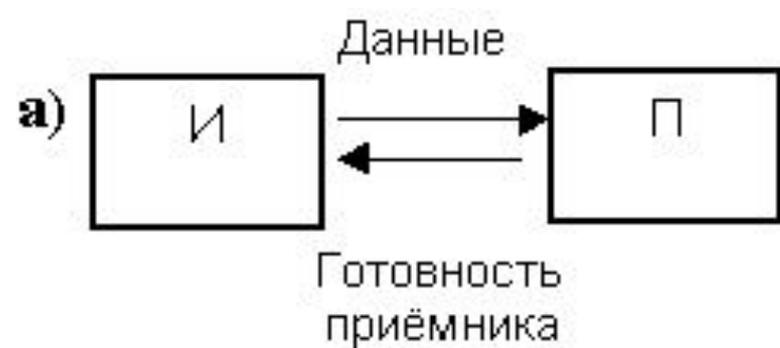


Рис.1.13. Асинхронный принцип синхронизации

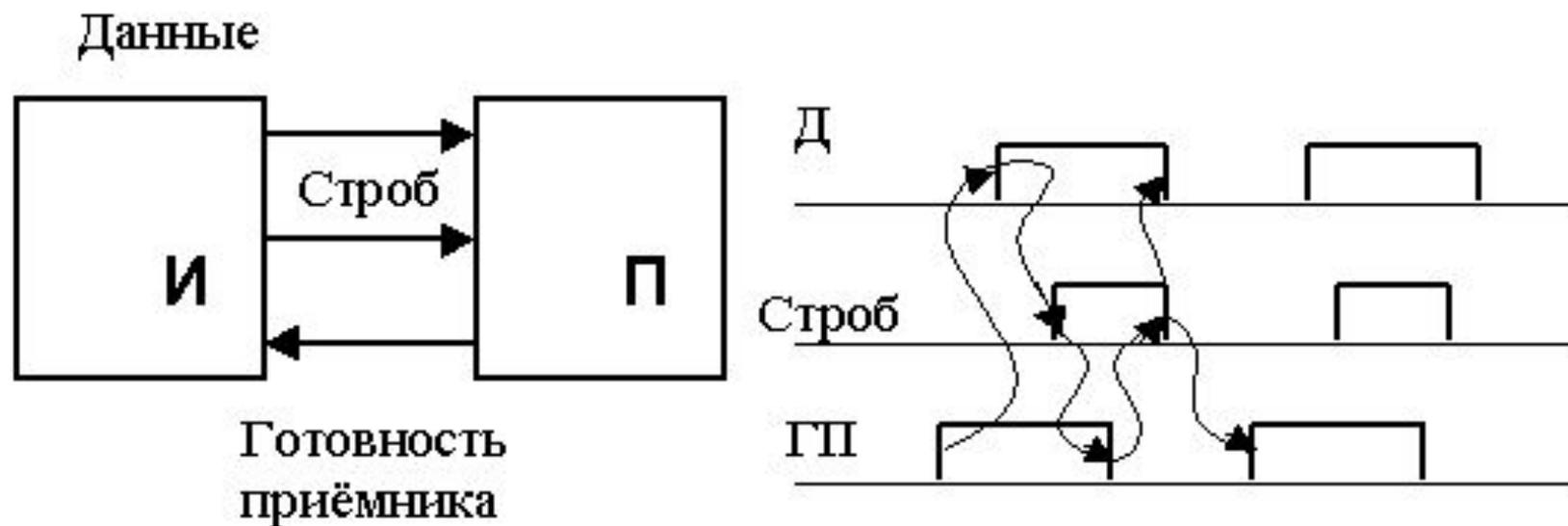


Рис. 1.17. Асинхронная параллельная передача данных

Функция арбитражи и селекции

Способ разрешения коллизий, возникающих при обращении нескольких задатчиков к одной шине назовем арбитражией, а способ разрешения коллизий, возникающих при обращении нескольких ПУ к ЦПр назовем селекцией. Принципы их реализации одни и теже. Устройство для реализации функции арбитражи – арбитр, для селекции – контроллер прерываний.

Функция арбитражи и селекции

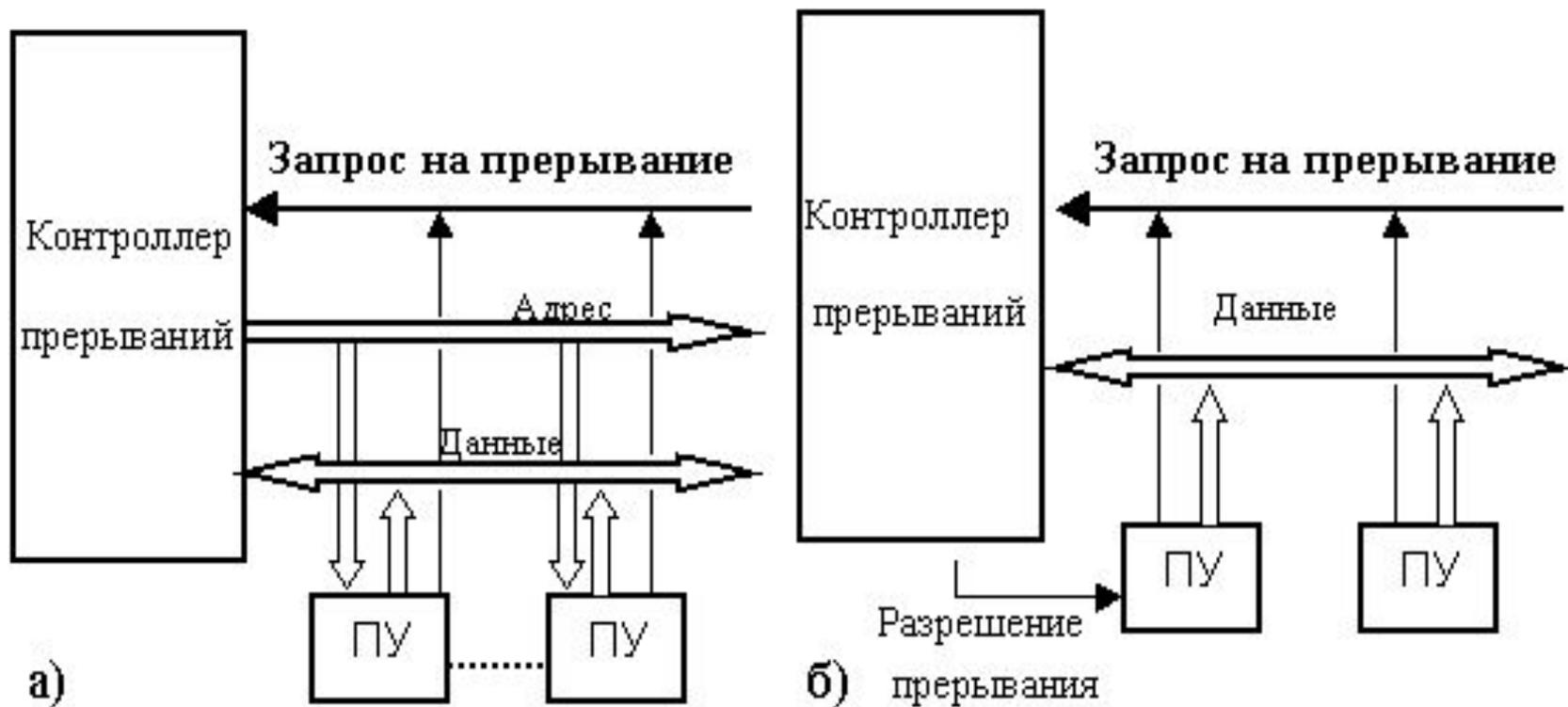


Рис. 1.18. Одноуровневая система селекции

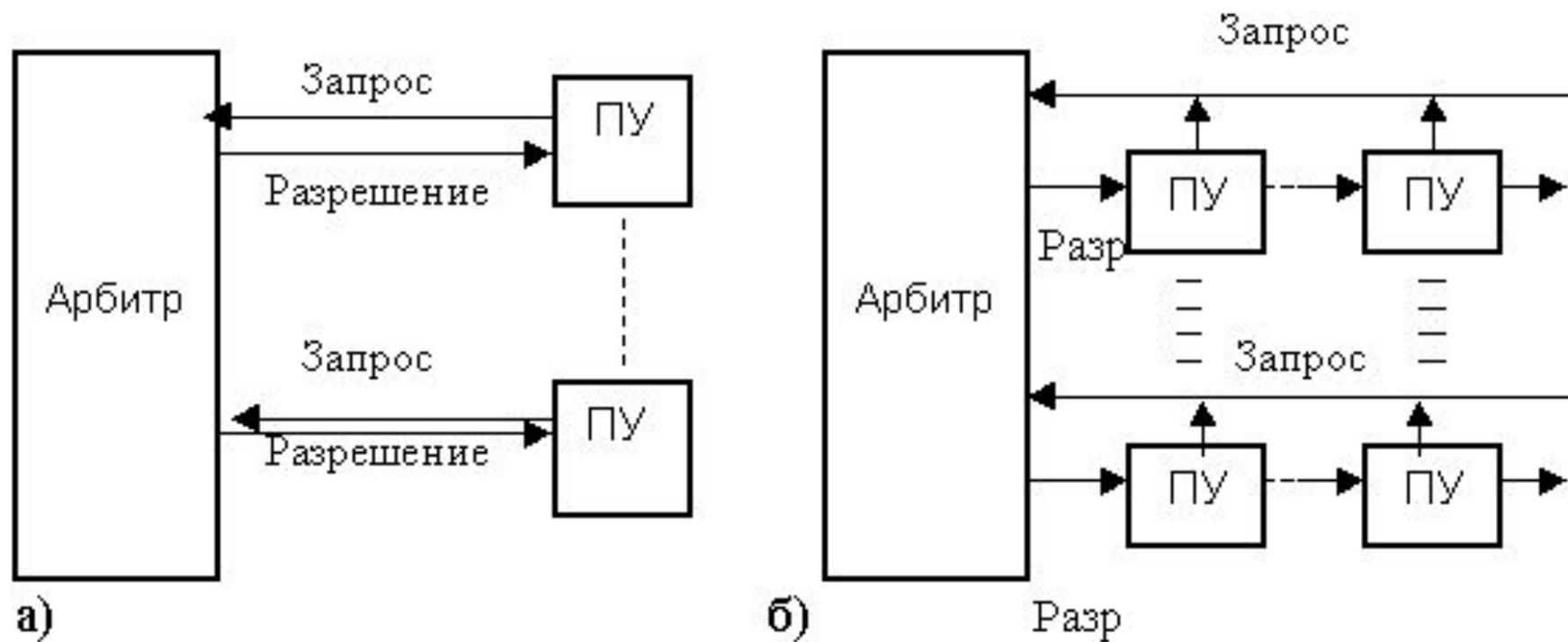


Рис. 1.19. Многоуровневая система арбитражи

Функция контроля: избыточное кодирование и режим тайм – аут.

Функция преобразования: из параллельного кода в последовательный и наоборот.

Функция автоконфигурации: автоматическая конфигурация периферийных устройств при их подключении к компьютеру.

Функция управления питанием: управление электропотреблением при разных состояниях компьютера, спецификация ACPI, технология OnNow

- С точки зрения ACPI, всего имеется 4 состояния PC:
- G0 – обычное рабочее состояние
- G1 – suspend, спящий режим
- G2 – soft-off, режим, когда питание отключено, но блок питания находится под напряжением, и машина готова включиться в любой момент
- G3 – mechanical off – питание отключено напрочь

Протоколы передачи данных в компьютерных интерфейсах

Алгоритмы протоколов передачи данных.

Два этапа: захват шины и передача данных.

1–ый этап: Арбитрация

- 1. Запрос задатчика арбитру на захват шины.**
- 2. Арбитр анализирует приоритеты и выдает разрешение задатчику с наибольшим приоритетом.**
- 3. Задатчик захватывает шину при ее переходе в состояние холстого хода и выставляет сигнал занятости шины.**

Селекция

- 1. Запрос ПУ к контроллеру прерываний на выполнение операции ввода - вывода**
- 2. Контроллер прерываний анализирует приоритет ПУ и выдает разрешение ПУ с наибольшим приоритетом.**
- 3. ЦПр анализирует сигнал от контроллера прерываний, прерывает свою работу и переходит к процессу ввода – вывода.**

- 2-ой этап:**
- 1. Задатчик устанавливает адрес исполнителя.**
 - 2. Задатчик устанавливает код (сигнал) команды.**
 - 3. При готовности задатчика и исполнителя начинается передача данных.**
 - 4. После передачи последнего слова шина переходит в состояние холостого хода.**

Параллельный интерфейс

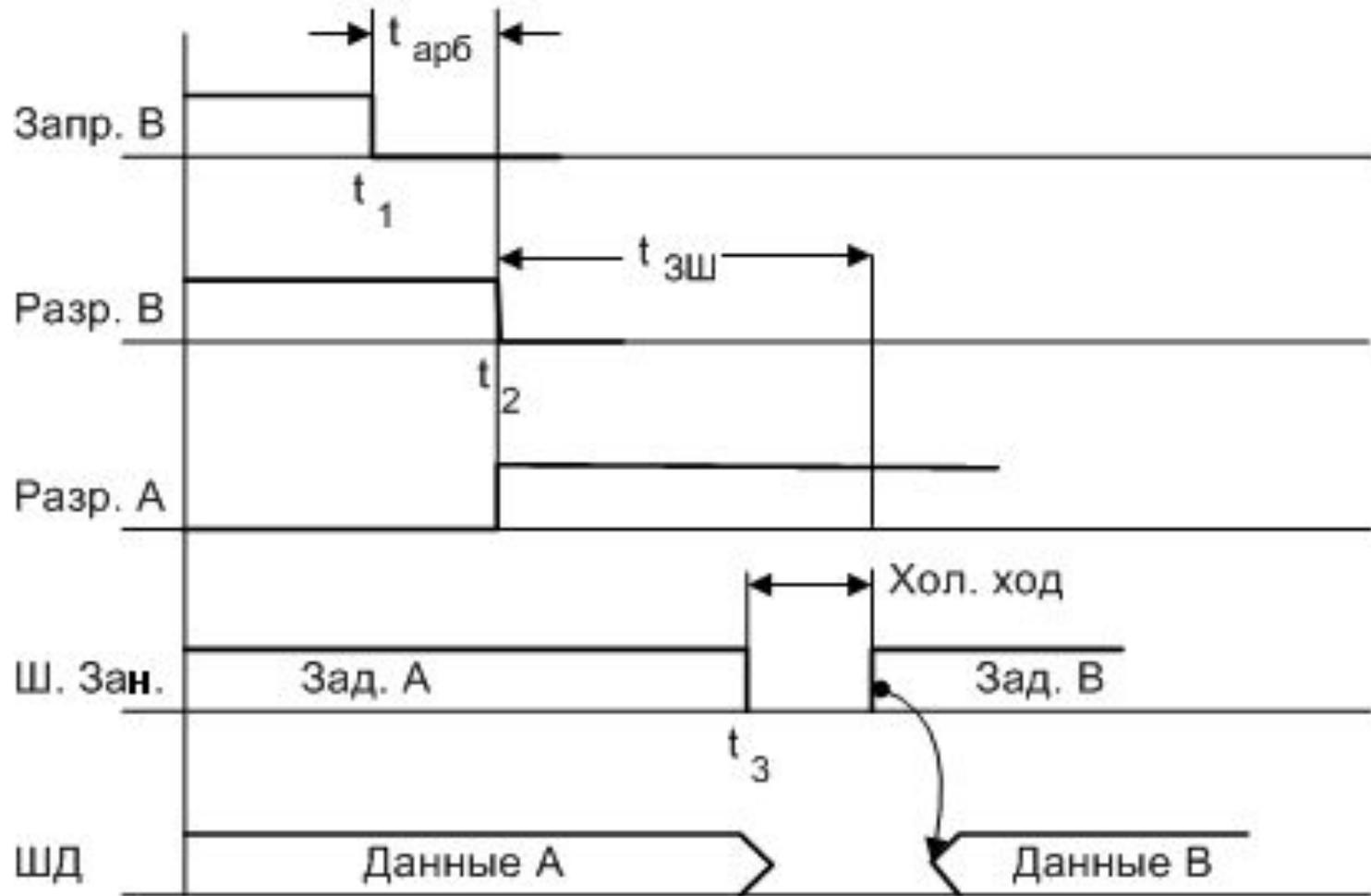


Рис. 1.20. Протокол арбитражи

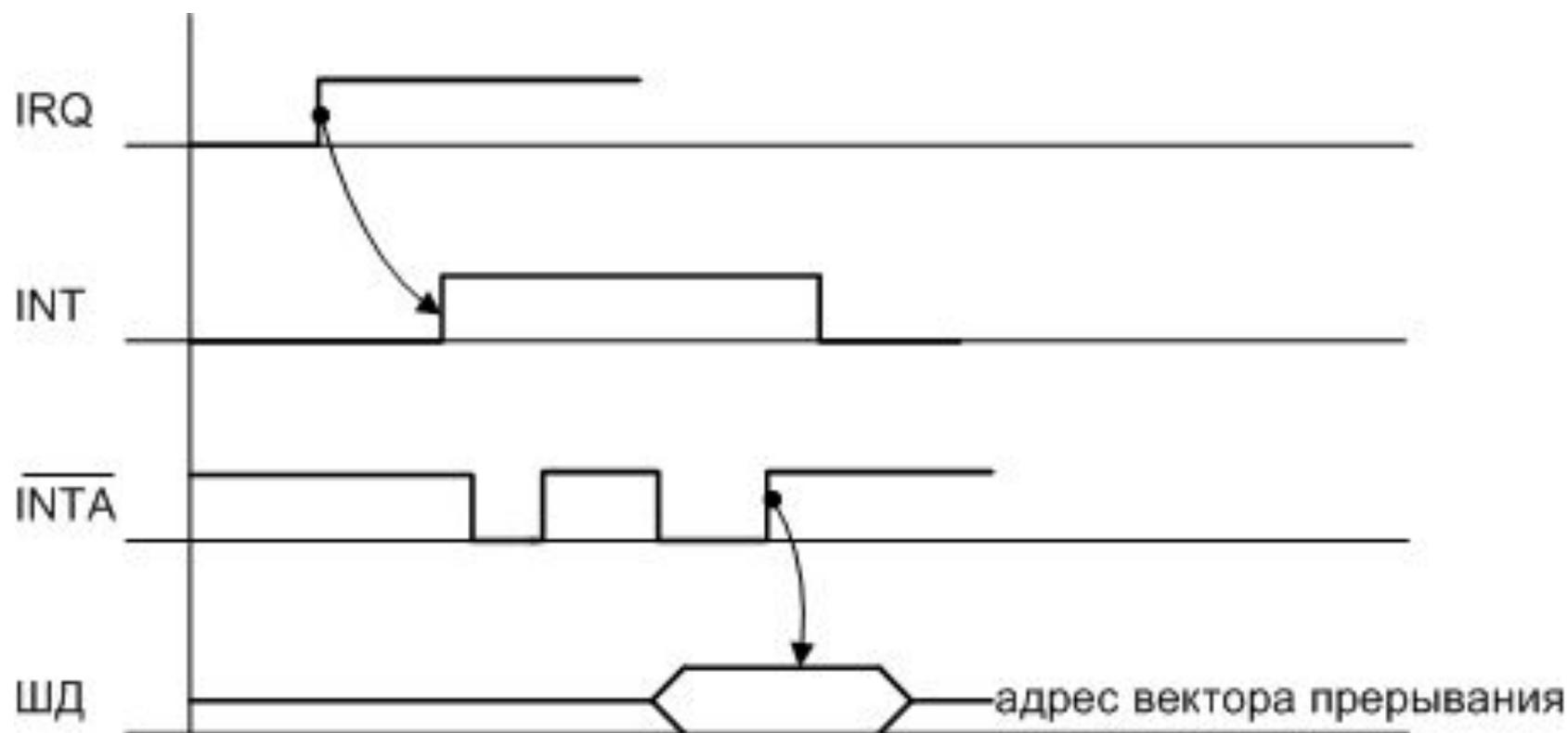
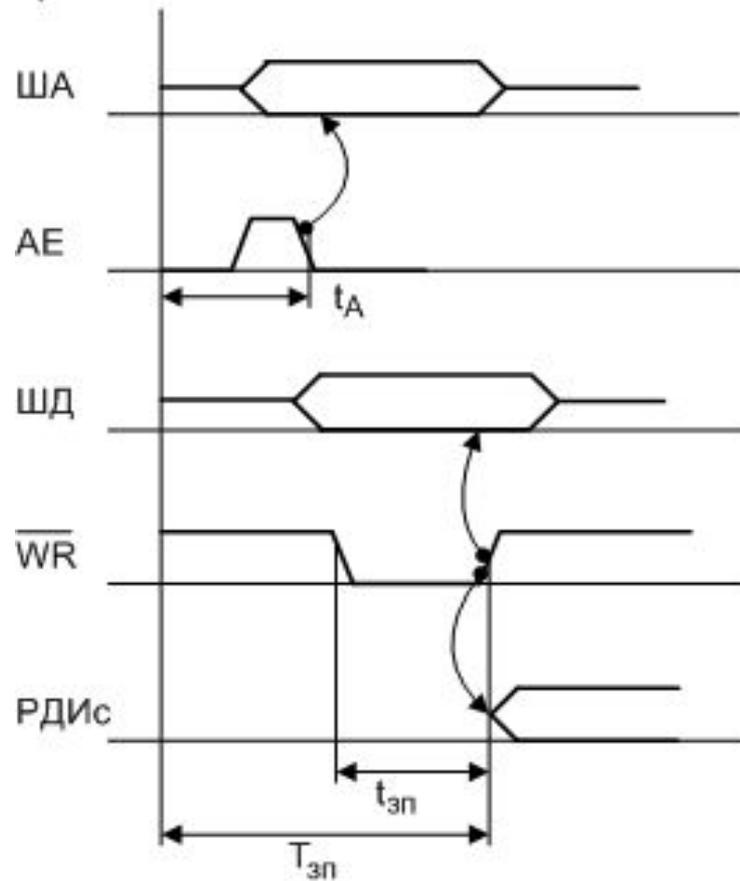


Рис. 1.21. Протокол режима прерываний

а) запись



б) чтение

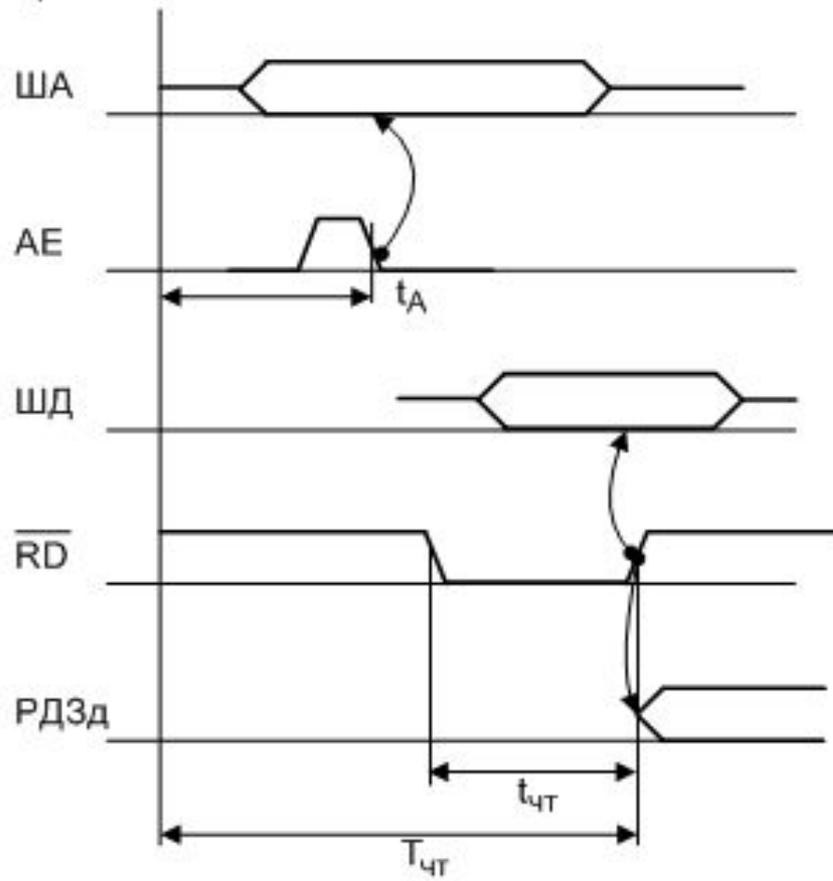
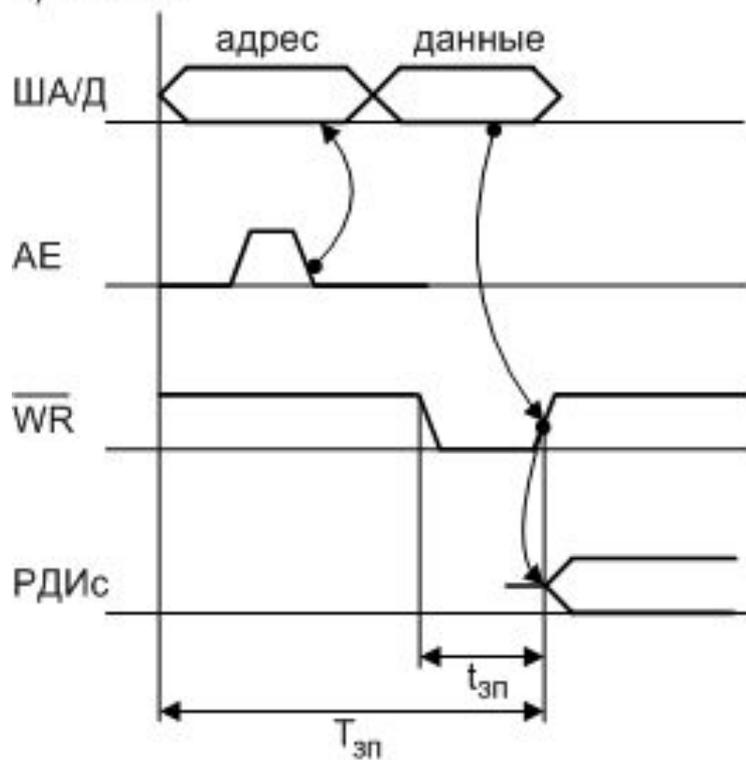


Рис. 1.22. Трехшинная архитектура синхронной передачи

а) запись



б) чтение

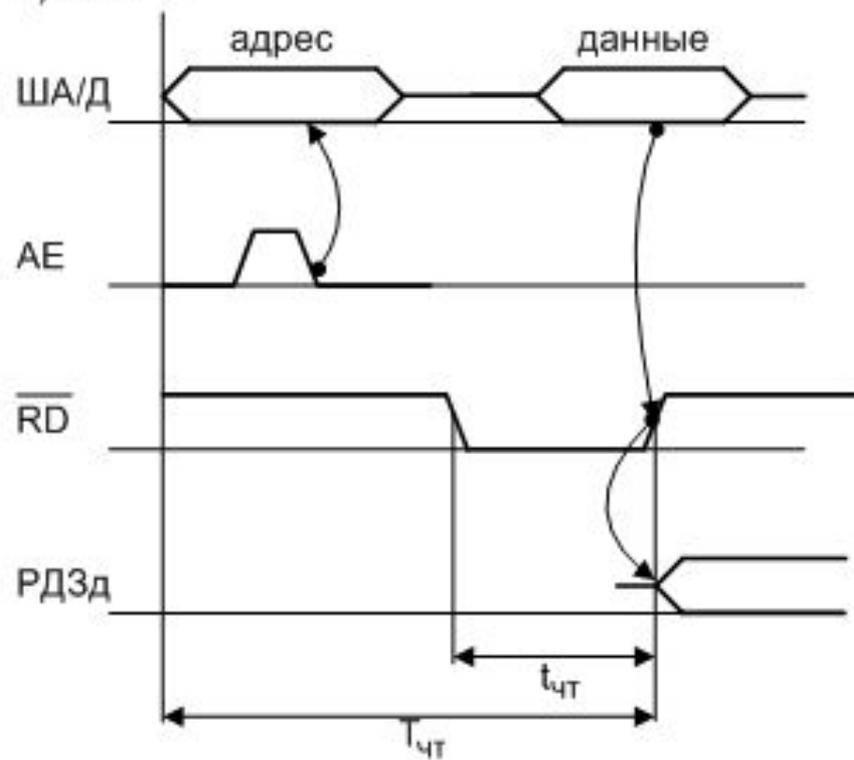


Рис. 1.23. Двухшинная архитектура синхронной передачи

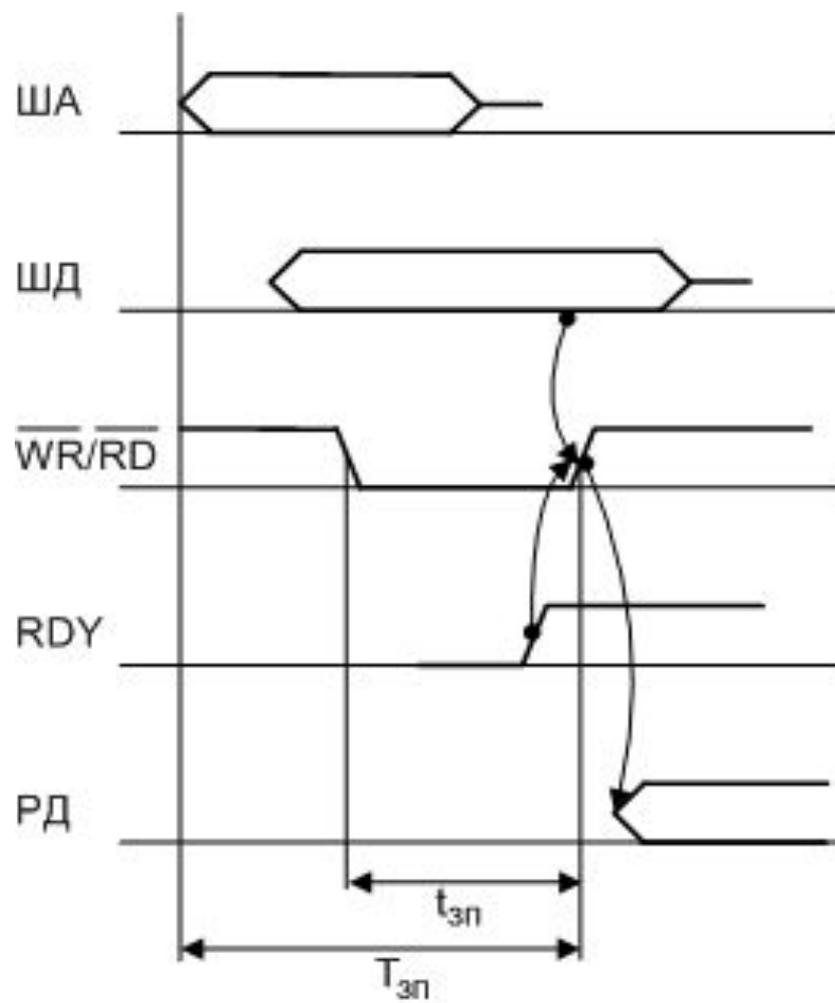


Рис. 1.24. Асинхронная передача

Последовательный интерфейс



Рис. 1.25. Байт-ориентированный протокол

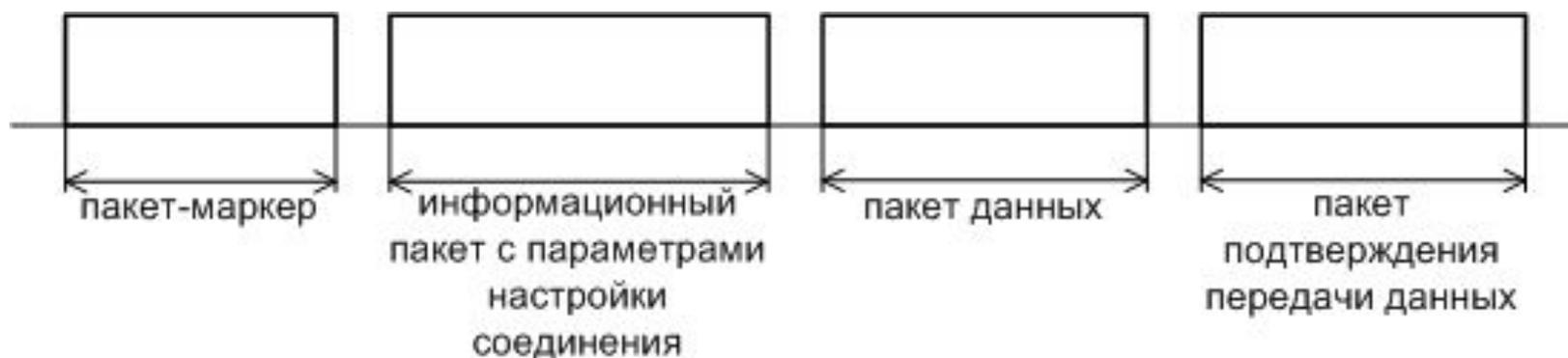


Рис. 1.26. Протокол с установлением соединения

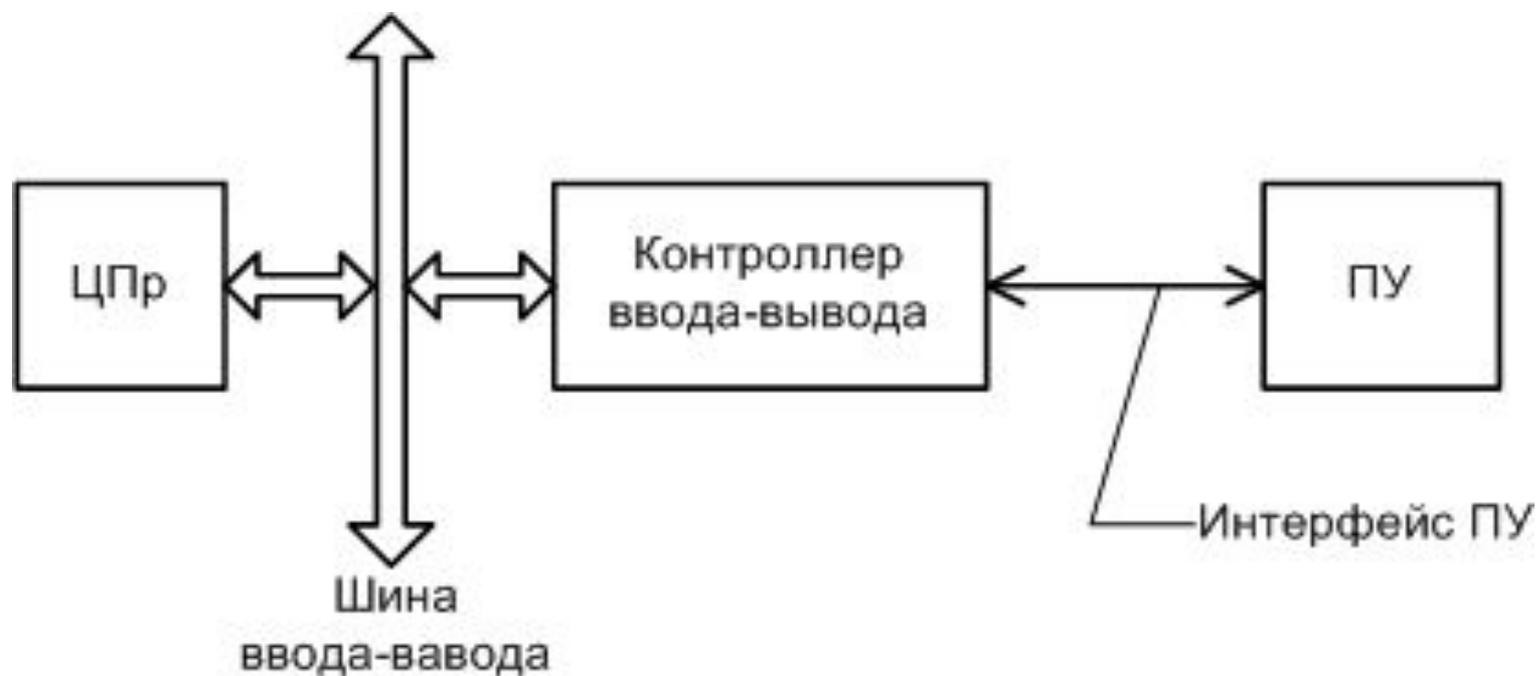
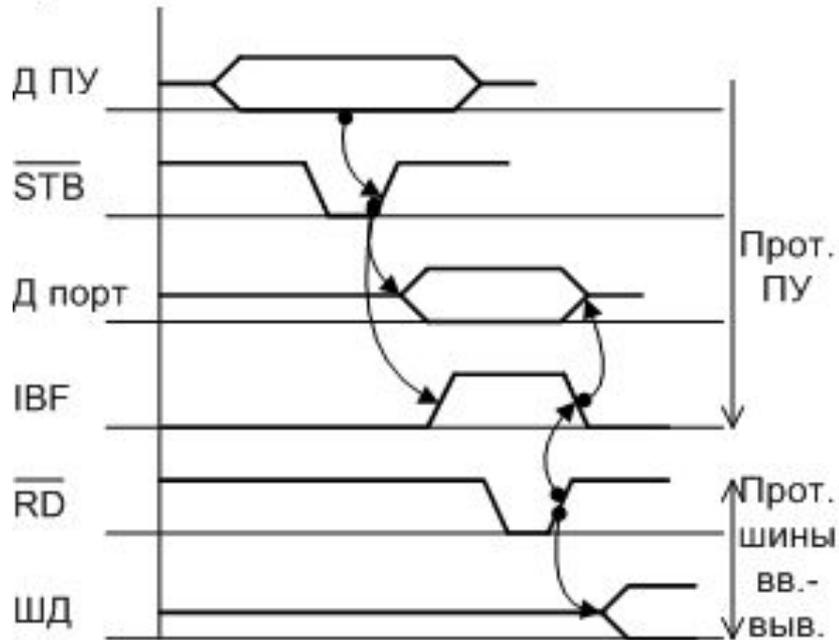


Рис. 1.27. Принцип взаимодействия центрального процессора (ЦПр) и периферийных устройств (ПУ)

Парллельный интерфейс

а) чтение



б) запись

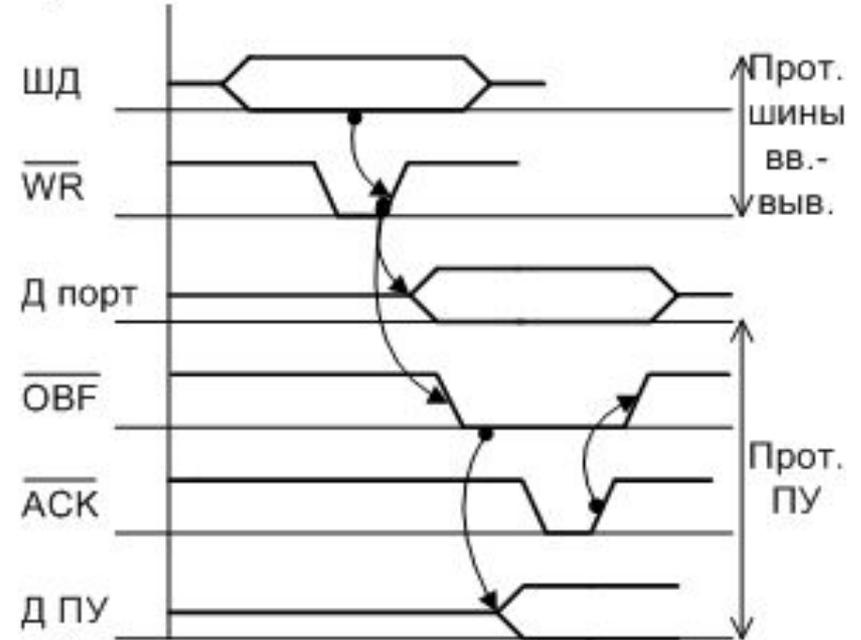


Рис. 1.28. Протоколы передачи между ПУ и шиной ввода-вывода

ТЕМА 1.3

Системные интерфейсы, шины расширения

Шина PCI

(Peripheral Component Interconnect)

Базовая версия PCI:

- Тактовая частота шины 33 МГц, используется синхронная передача данных;
- Пиковая пропускная способность 133 МБ в секунду;
- Параллельная шина данных шириною 32-бита;
- Адресное пространство 32-бита (4 ГБ);
- Сигнальный уровень 3.3 или 5 вольт.

Позже появляются следующие ключевые модификации шины:

- PCI 2.2 — допускается 64-бит ширина шины и/или тактовая частота 66 МГц, т.е. пиковая пропускная способность до 533 МБ/сек.;
- PCI-X, 64-бит версия PCI 2.2 с увеличенной до 133 МГц частотой (пиковая пропускная полоса 1066 МБ/сек.);
- PCI-X 266 (PCI-X DDR), DDR версия PCI-X (эффективная частота 266 МГц, реальная 133 МГц с передачей по обоим фронтам тактового сигнала, пиковая пропускная полоса 2.1 ГБ/сек);
- PCI-X 533 (PCI-X QDR), QDR версия PCI-X (эффективная частота 533 МГц, пиковая пропускная полоса 4.3 ГБ/сек.);
- Mini PCI — PCI с разъемом в стиле SO-DIMM, применяется преимущественно для миниатюрных сетевых, модемных и прочих карточек в ноутбуках;
- Compact PCI — стандарт на форм фактор (модули вставляются с торца в шкаф с общей шиной на задней плоскости) и разъем, предназначенные в первую очередь для промышленных компьютеров и других критических применений; развитие – PXI.

Архитектура шины

- многопроцессорная;
- двухшинная архитектура;
- 32, 64 – разрядная адресация данных;
- синхронная шина;
- производительность 133 Мбайт/сек – 4,3 Гбайт/сек , частота 33 МГц – 133 МГц, эффективная частота до 533 МГц, использование технологий DDR и QDR;
- пакетная передача данных, транзакции;
- арбитрация;
- таймер-задержка;
- два метода адресации;
- три адресных пространства: ОЗУ, регистры ввода/вывода, регистры автоконфигурации;
- поддержка автоконфигурации;
- контроль четности.

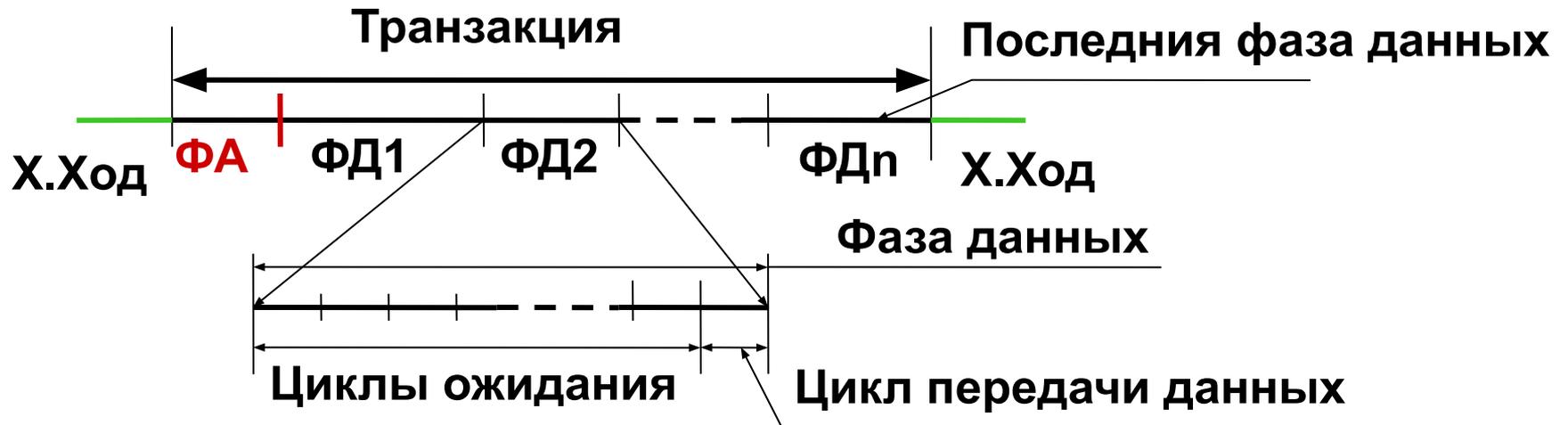
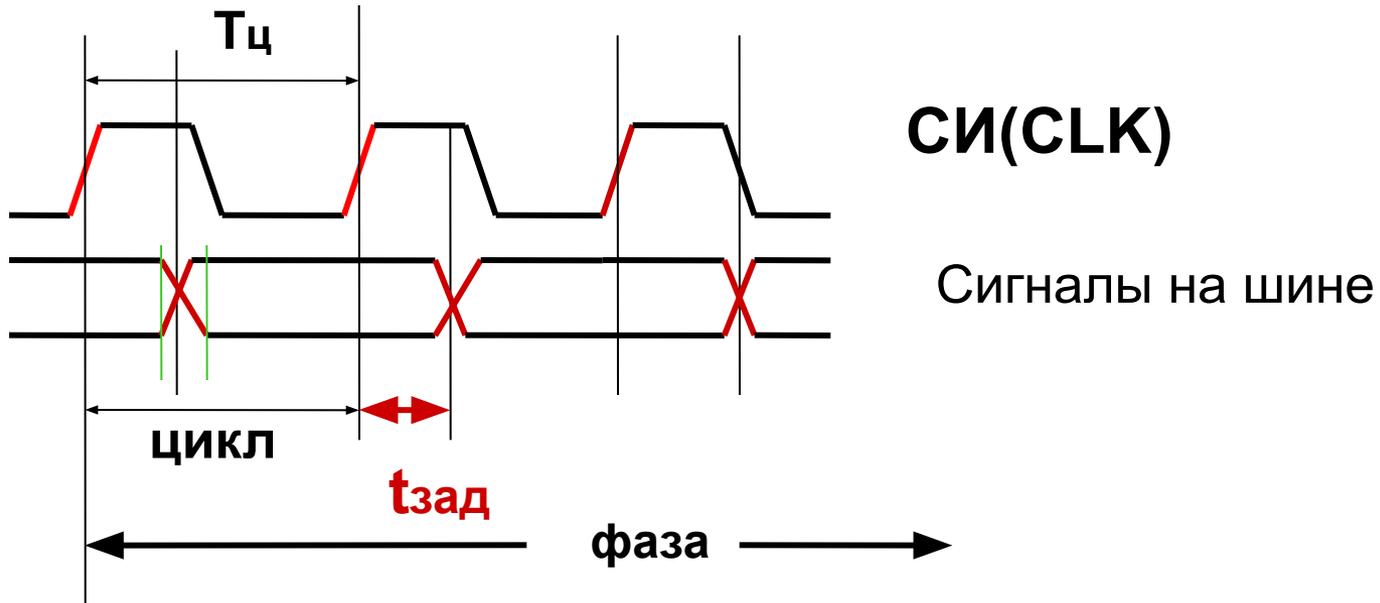
Базовые сигналы шины PCI

- AD[31-0] – адрес – данные;
- C/BE[3-0] – код команды, указатель байт;
- #FRAME – начало и конец транзакции;
- #DEVSEL – исполнитель найден;
- #IRDY – готовность задатчика, строб данных;
- #TRDY – готовность исполнителя, строб данных;
- #STOP – прерывание транзакции от исполнителя;
- #LOCK – выполнение нескольких транзакций;
- #REQ – запрос на захват шины;
- #GNT – разрешение на захват шины;
- IDSEL – выбор устройства при автоконфигурации;
- PAR – контроль по четности A/D, C/BE;
- #PERR – ошибка паритета;
- #SERR – системная ошибка;
- #RST – сброс;
- CLK – синхроимпульс.

Декодирование команд шины PCI

C/BE[3:0]	Тип команды
0000	Interrupt Acknowledge — подтверждение прерывания
0001	Special Cycle — специальный цикл
0010	I/O Read — чтение порта ввода-вывода
0011	I/O Write — запись в порт ввода-вывода
0100	Зарезервировано
0101	Зарезервировано
0110	Memory Read — чтение памяти
0111	Memory Write — запись в память
1000	Зарезервировано
1001	Зарезервировано
1010	Configuration Read — конфигурационное считывание
1011	Configuration Write — конфигурационная запись
1100	Multiple Memory Read — множественное чтение памяти
1101	Dual Address Cycle (DAC) — двухадресный цикл
1110	Memory-Read Line — чтение строки памяти
1111	Memory Write and Invalidate — запись с инвалидацией

Циклы шины



Шина PCI Express

- Последовательная системная шина общего назначения;
- Имя — **PCI Express**, на стадии проектирования была также известна как **3GIO** (Ввод-вывод третьего поколения) или по кодовому имени рабочей группы и проекта «Arapahoe», причем оба названия (3GIO и PCI Express) являются зарегистрированными торговыми марками PCISIG;
- Дата рождения — 22 июля 2002 года — опубликована базовая спецификация протокола и сигнального уровня, а также базовая спецификация на форм-фактор и энергопотребление карт и разъемы;
- Фактически — совокупность независимых самостоятельных последовательных каналов передачи данных;
- Сигнальный уровень 0.8 вольт. Каждый канал состоит из двух дифференциальных сигнальных пар (необходимо только 4 контакта):

- Используется избыточное защищенное от помех кодирование — каждый байт при передаче представляется десятью битами;
- Пропускная способность 2.5 Гигабита (250 МБ) в секунду для одного канала в каждом направлении одновременно (полный дуплекс), однако, следует учесть, что эффективная скорость передачи данных за вычетом избыточного кодирования составляет 2 Гигабита (200 МБ) ровно;
- Стандартизированы 1, 2, 4, 8, 16 и 32 канальные варианты (до 6.4 эффективных Гигабайт в секунду соответственно, при передаче в одну сторону и вдвое больше при передаче в обоих направлениях). При передаче данных они передаются параллельно (но не синхронно) по всем доступным каналам:

Тема 1.4.

Интерфейсы периферийных устройств (Centronics, RS232, SCSI, USB)

Параллельный интерфейс: LPT-порт

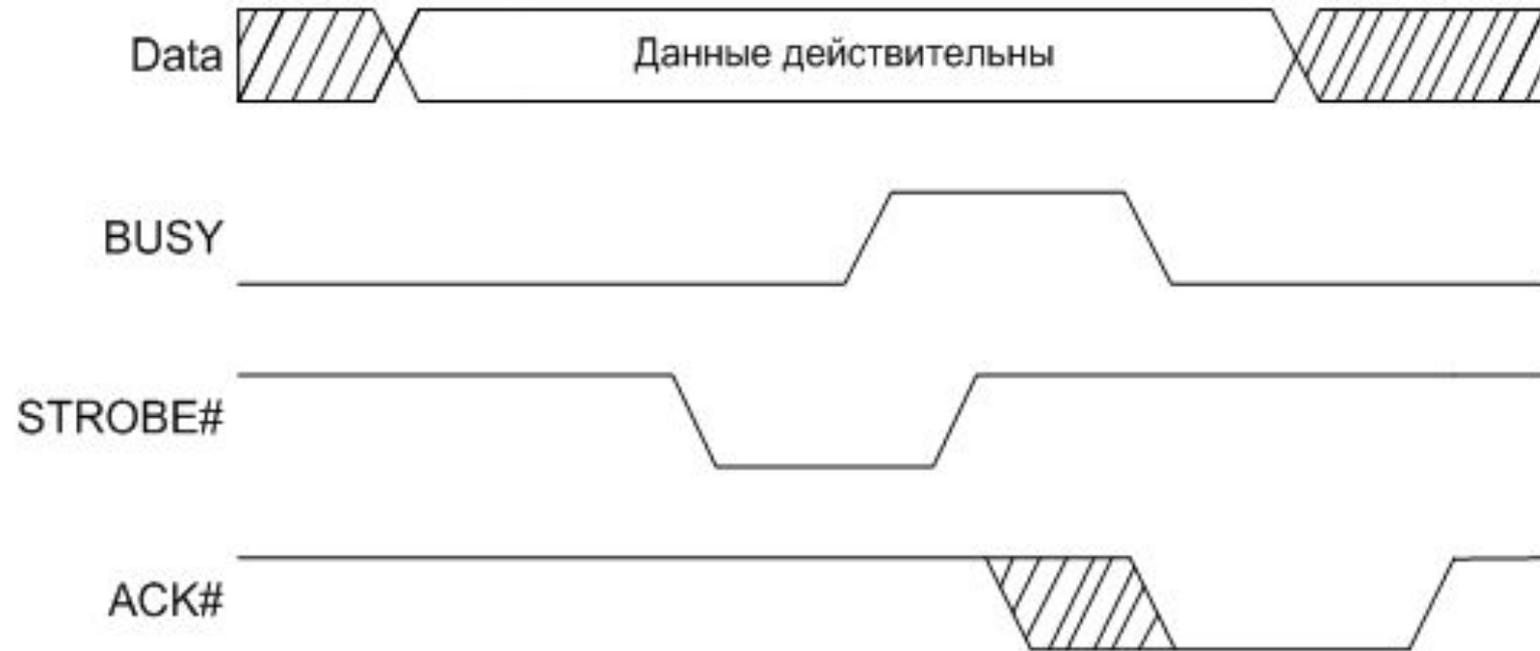


Рис. 1.

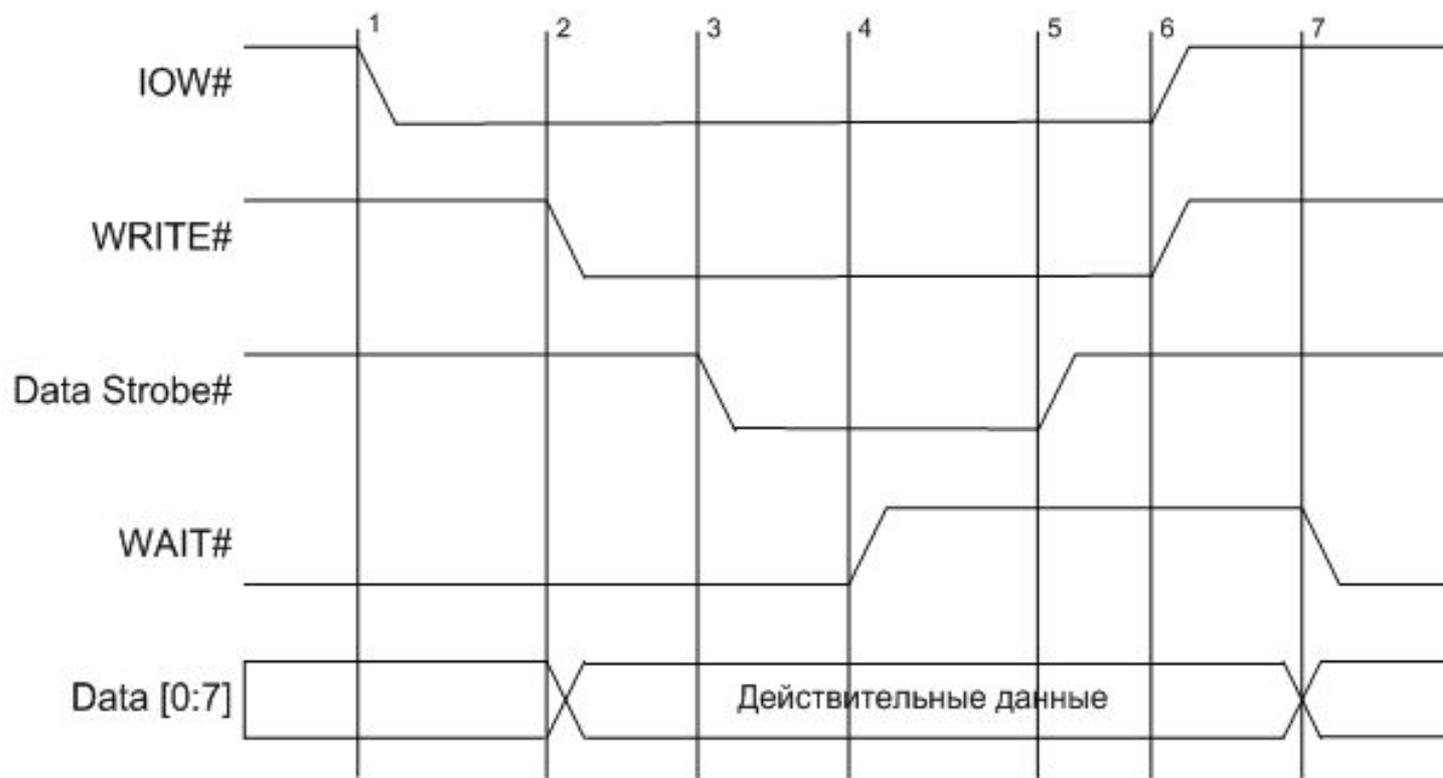


Рис. 2. Цикл записи данных.

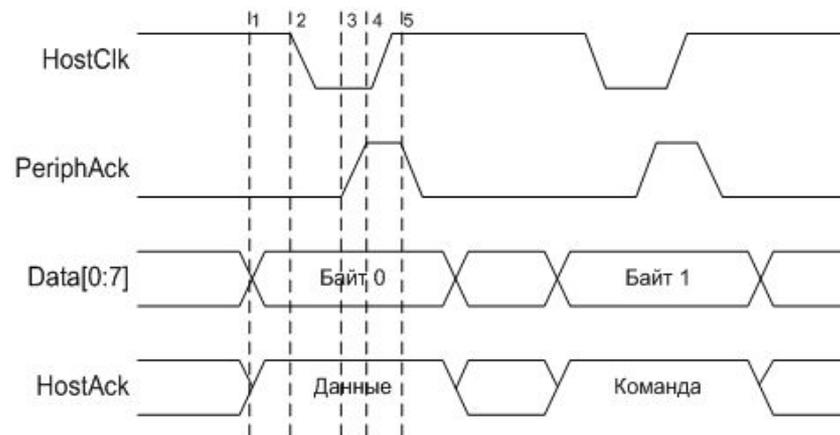


Рис. 3,а.

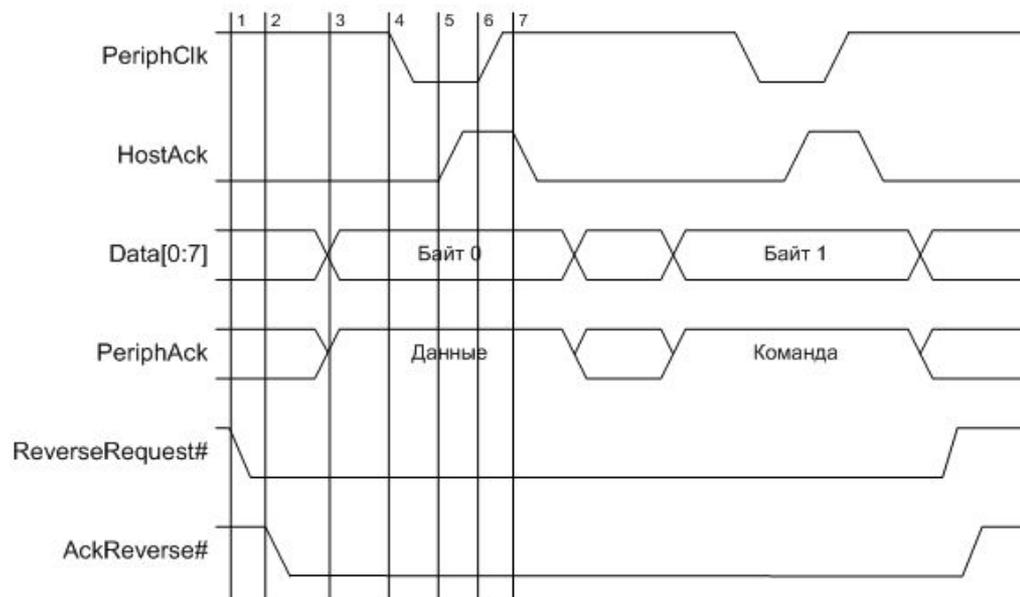


Рис. 3,б.

Последовательные интерфейсы: COM-порт



Рис. 1. Формат асинхронной передачи.

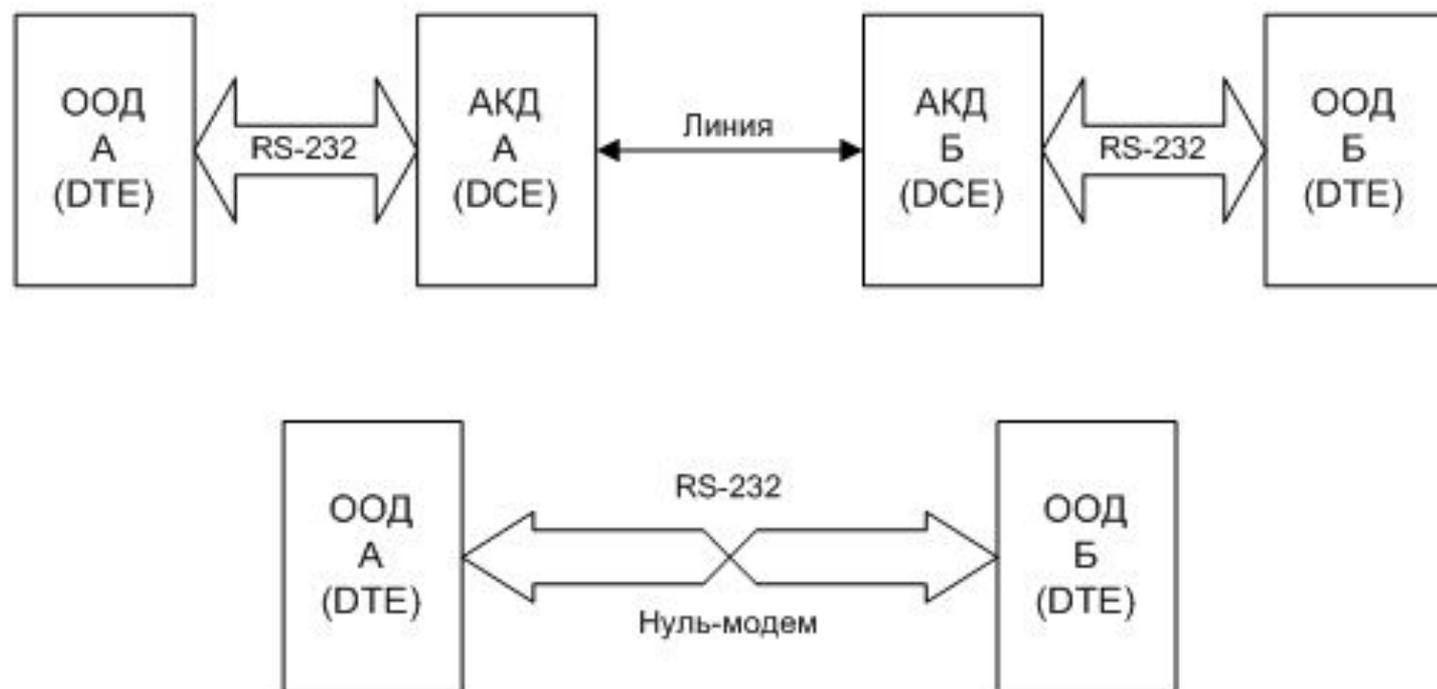


Рис. 2. Соединение по RS-232C нуль-модемным кабелем.

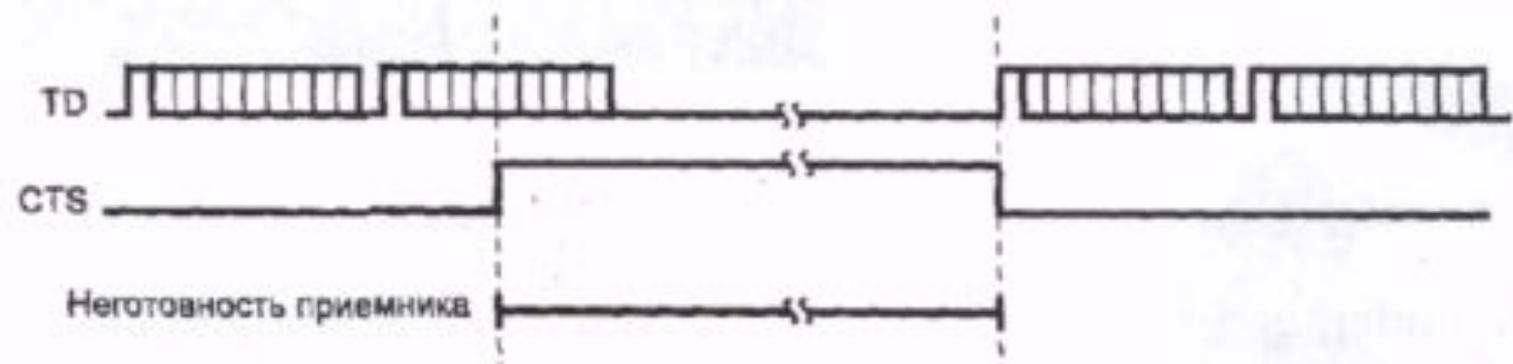


Рис. 5.

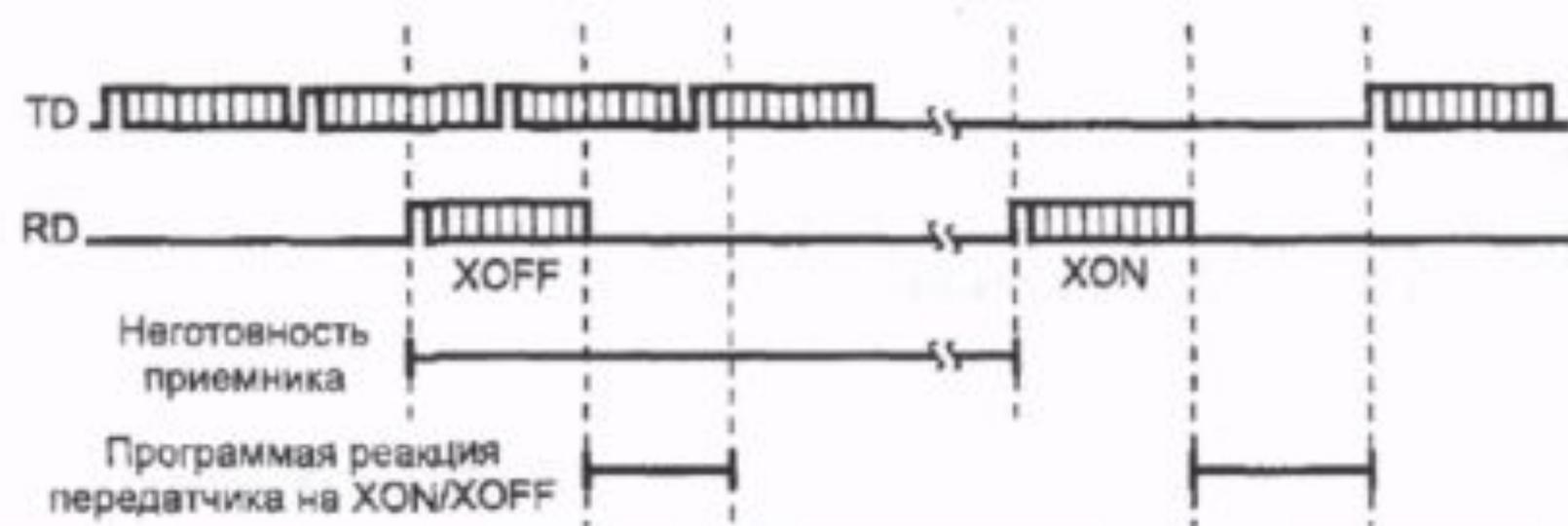


Рис. 6.

Шина USB

(Universal Systems Interface)

Архитектура шины USB

1. Последовательная передача данных.
2. Полудуплексный режим передачи.
3. Принцип действия – сетевой с маркерным доступом.
4. Максимальное количество подключаемых ПУ – 127.
5. Скорость передачи: USB1 - 1,5 Мбит/сек, 12 Мбит/сек
USB2 - 480 Мбит/сек
USB3 – 5 Гбит/сек
6. Наличие автоконфигурации.
7. Горячее подключение.
8. Возможно питание от шины.
9. Топология шины – многоуровневая звезда.

Структура шины

Структура имеет топологию – многоуровневой звезды.

В ней используются сетевые принципы построения шины. В основе структуры лежат устройства шины: хабы и функции. Хабы представляют из себя сетевые концентраторы, являющиеся центрами многоуровневой звезды, к которым кабелем подключаются функции(ПУ).

Могут быть комбинированные устройства, содержащие как хаб так и функцию(ПУ).

Компьютер имеет хост – контроллер, содержащий корневой хаб, имеющий одну или несколько точек подключения.

Хаб имеет один восходящий и несколько низходящих портов.

Функции хаба.

Распознает подключение или отключение устройства к порту.

Управляет подачей питания на подключенное устройство.

Может разрешить или запретить использование порта.

Обеспечивает изоляцию сегментов с низкой скоростью от высокоскоростных.

Рис. 2-1 Топология шины USB



СТРУКТУРА ХАБА

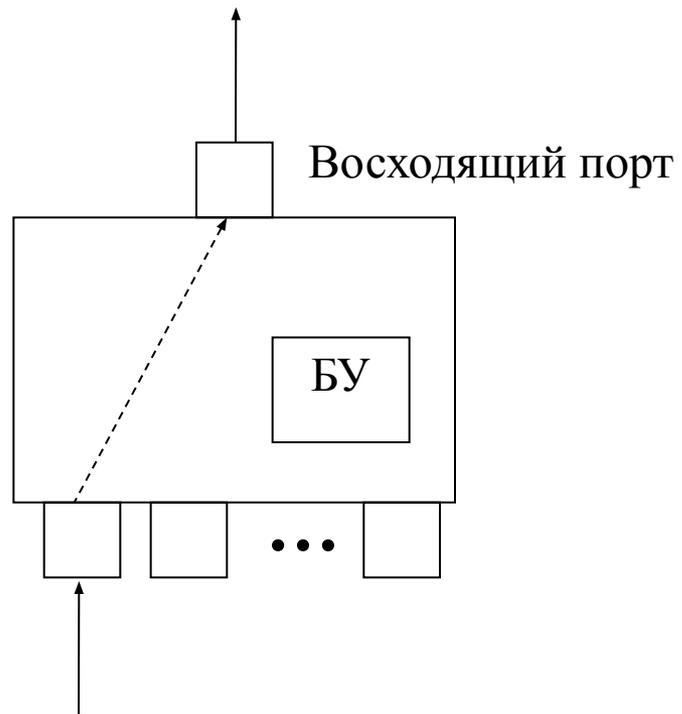
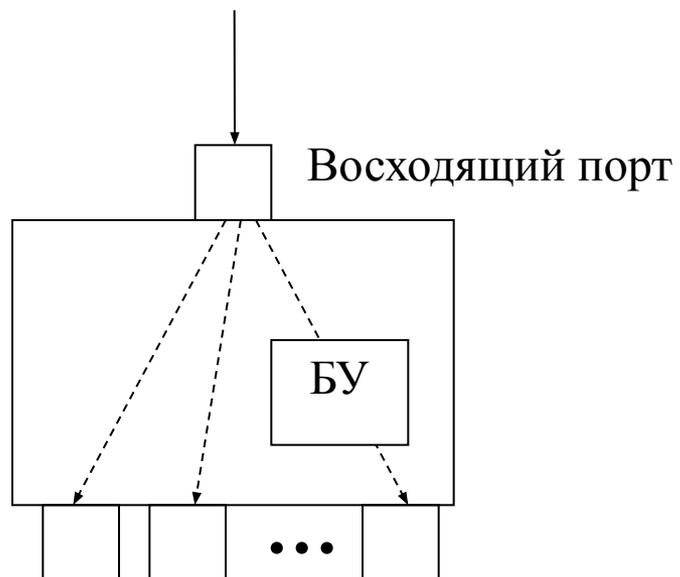


Рис. 3-1 Простейшее хост/устройство взаимодействие



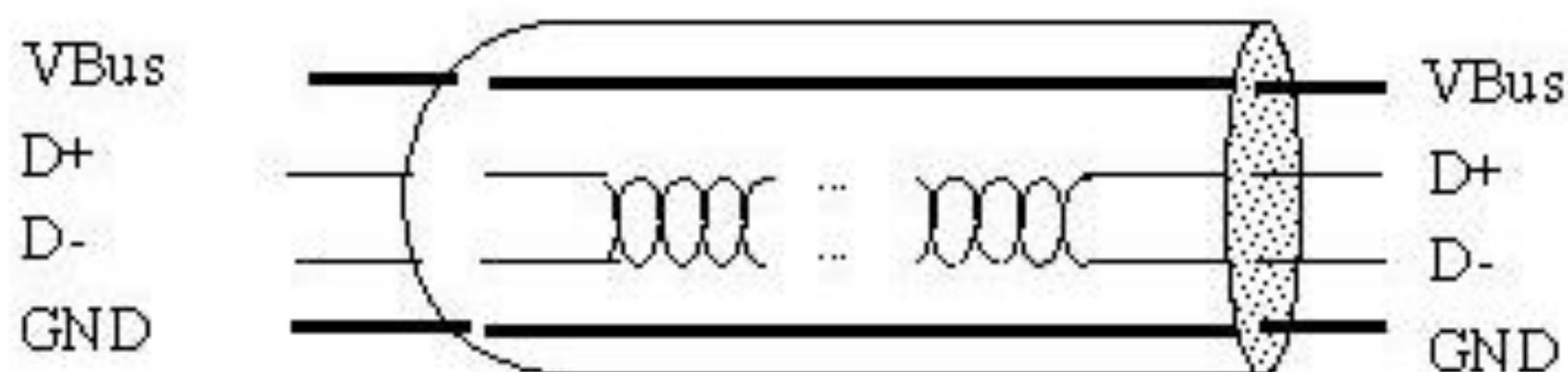
Физический интерфейс USB

Кабель имеет 4 провода: два для передачи сигналов (Д+ и Д-), два для подачи питания (5в). Дифференциальный способ передачи сигналов. Передатчик должен иметь 3-е высокоимпедансное состояние, для реализации полудуплексного режима передачи данных.

Кроме дифференциального сигнала приемник и передатчик могут работать и с линейными сигналами. Это дает возможность иметь более двух состояний линий. Следующие состояния линий:

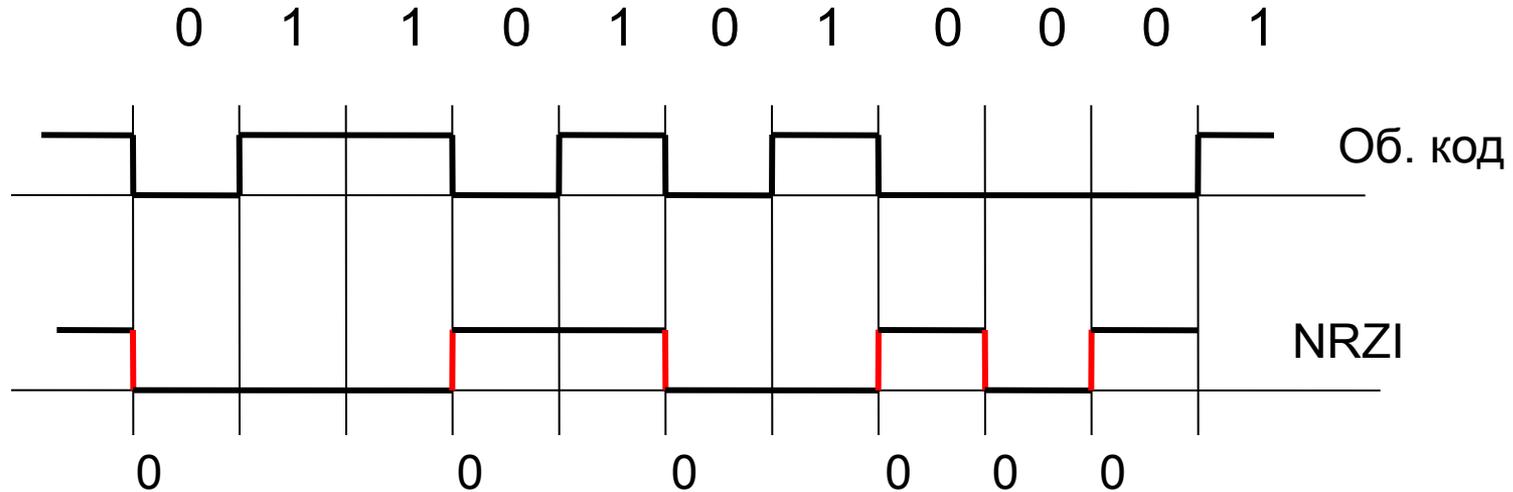
- линейный ноль (SEO, Singl-Endet Zero): – Д + и Д – низкий уровень,
- состояние передаваемых бит : Data J State(J), Data K State(K),
- Пауза на шине: Idle State,
- сигнал «пробуждения» : Resume State,
- начало пакета: Start of Packet – переход из Idle State в состояние K,
- конец пакета: End of Packed,
- устройство отключено от порта: Disconnect,
- устройство подключено к порту: Connect,
- сброс устройства: Reset.

Рис. 2-4 Стандартный кабель USB



Кодирование информации

Используется NRZI – кодирование.



Если подряд идет более 6 единиц, то после шестой вставляется 0.

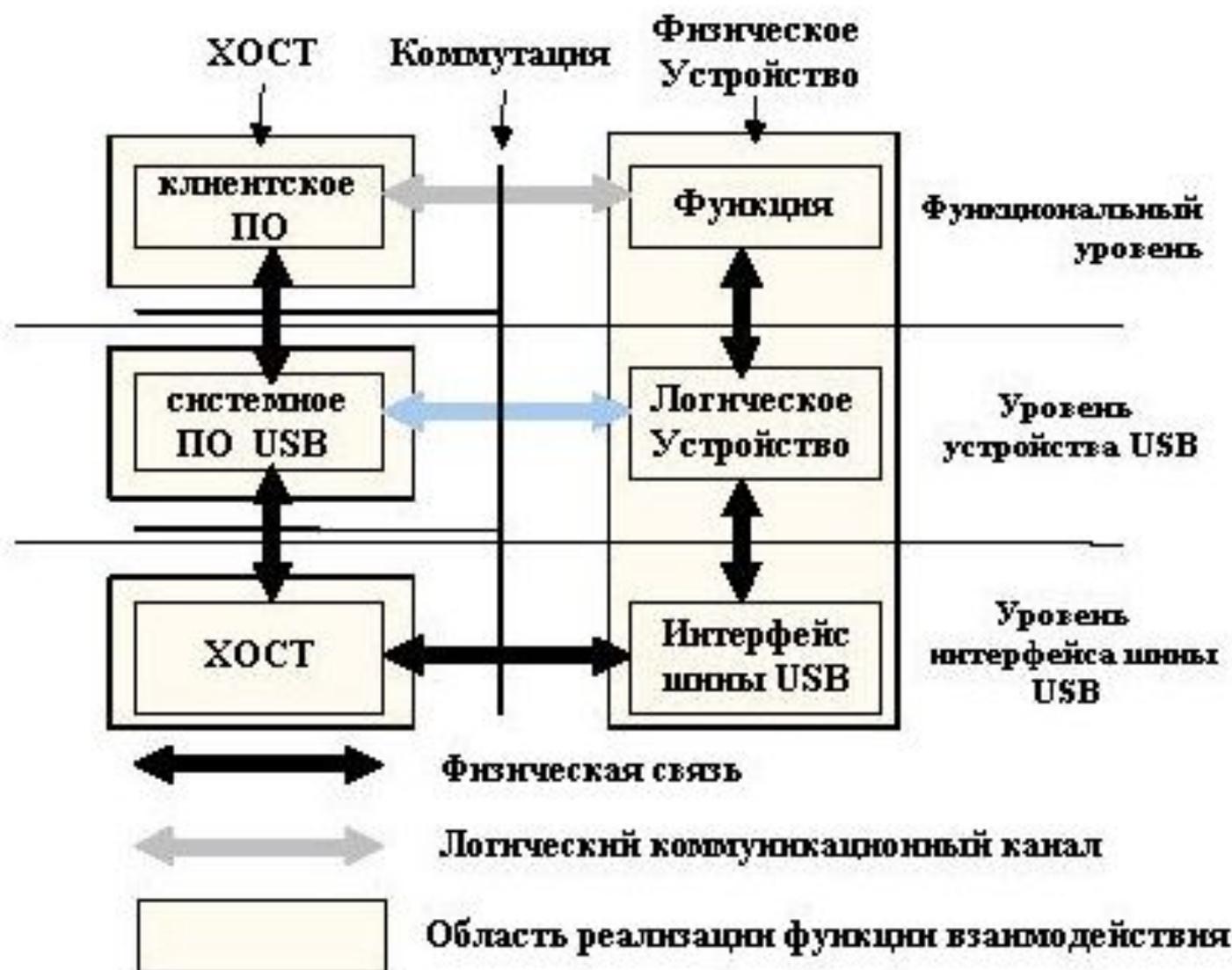


Рис. 3-2 Взаимодействие между хостом и функциональным устройством.

Типы передачи данных

1. **Управляющие** – конфигурирование и управление устройствами. Количество данных – не более 64 байт, асинхронная передача с квитированием. Control transfer.
2. **Сплошные** – передача больших блоков без жестких требований по времени доставки. Поле данных пакета не более 64байт. Асинхронная с квитированием. Bulk data transfer.
3. **С прерываниями** – короткие передачи типа символа. Асинхронная с квитированием. Interrupt.
4. **Изохронные** – непрерывная передача в реальном времени с заданной задержкой доставки, например аудио, видео.

ТИПЫ ПЕРЕДАЧИ ДАННЫХ

□ **Управляющие посылки** (Control Transfers);

Используются для конфигурирования во время подключения и в процессе работы, для управления устройствами. Протокол обеспечивает гарантированную доставку данных. Длина поля данных управляющей посылки не превышает 64 байт на полной скорости и 8 байт на низкой.

□ **Сплошные передачи** (Bulk Data Transfers);

Передача сравнительно больших пакетов без жестких требований ко времени доставки. Передачи занимают всю свободную полосу пропускания шины. Пакеты имеют поле данных размером 8, 16, 32 или 64 байт. Приоритет этих передач самый низкий, они могут приостанавливаться при большой загрузке шины. Допускаются только на полной скорости передачи.

□ **Прерывания** (Interrupt);

Короткие (до 64 байт на полной скорости, до 8 байт на низкой) передачи типа вводимых символов или координат.

Прерывания имеют спонтанный характер и должны обслуживаться не медленнее, чем того требует устройство. Предел времени обслуживания устанавливается в диапазоне 1-255 мс для полной скорости и 10-255 мс — для низкой.

□ **Изохронные передачи** (Isochronous Transfers);

Непрерывные передачи в реальном времени, занимающие предварительно согласованную часть пропускной способности шины и имеющие заданную задержку доставки.

Модель передачи данных

Данные передаются между хост– контроллером и функцией(ПУ) в виде потоков информации. Управляет всеми передачами хост – контроллер.

Возможны передачи только между хост – контроллером и ПУ.

Функция представляется в виде логического устройства, состоящего из набора конечных точек (End Point, EP) – регистров. Каждое логическое устройство имеет свой адрес, каждая конечная точка идентифицируется своим номером. В низкоскоростных устройствах 3, а в высокоскоростных до 16 EP. Каждое устройство должно иметь EP с номером 0, через которую осуществляется конфигурирование и управление устройством.

Для решения задачи в устройстве имеется набор из несколько EP, называемых интерфейсом задачи. Если выполняется несколько задач, устройство имеет несколько интерфейсов. Набор одновременно поддерживаемых интерфейсов составляет конфигурацию устройства.

Каждая конечная точка(EP) имеет набор характеристик.

Характеристики конечной точки

1. Требуемая частота доступа и задержка обслуживания.
2. Требуемая частота пропускания канала.
3. Номер точки.
4. Требования к обработке ошибок.
5. Максимальный размер пакетов.
6. Тип обмена.
7. Направление обмена.

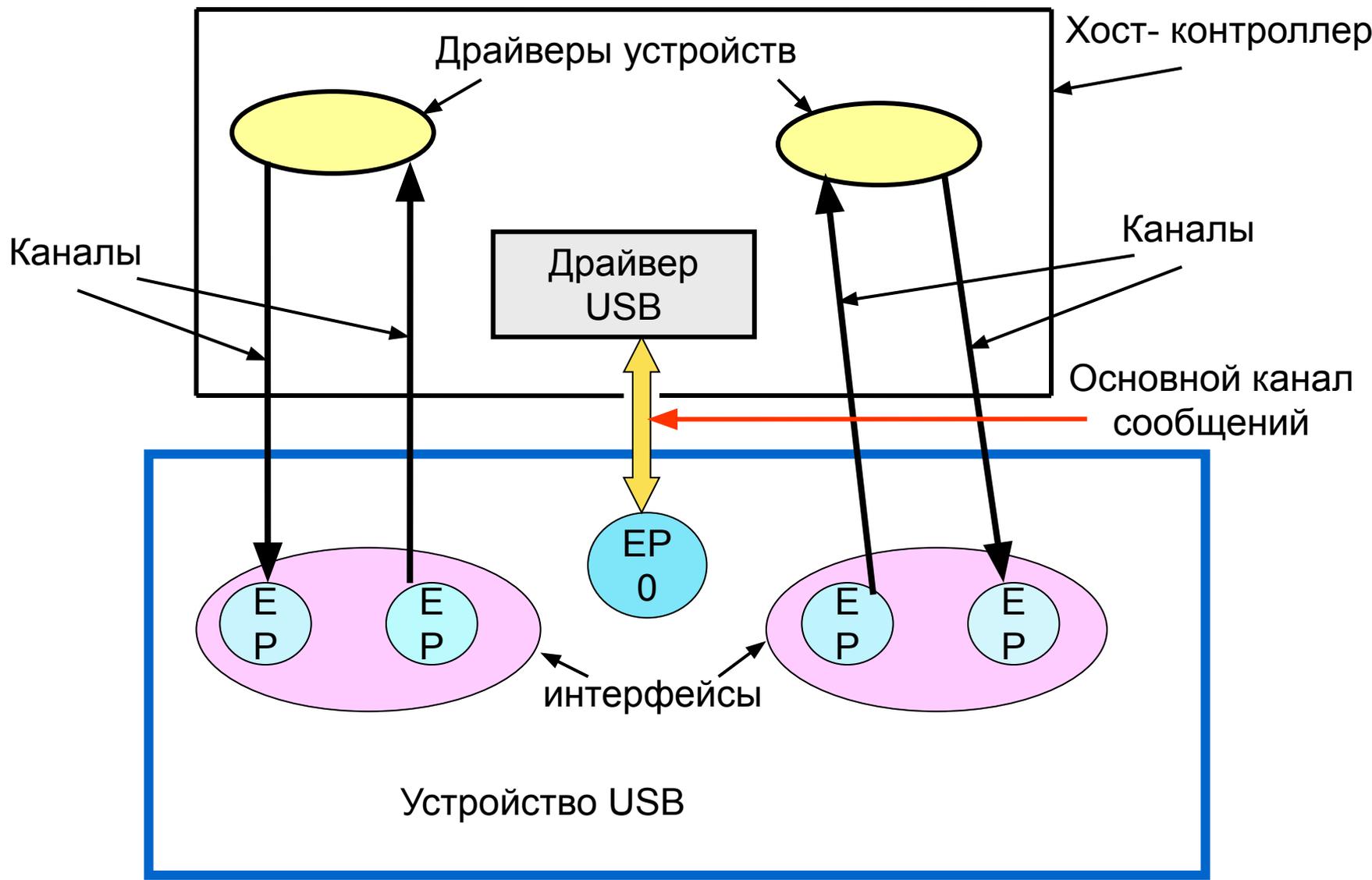
При конфигурации между драйвером устройства и конечной точкой устанавливаются коммуникационные каналы (communication pipe). Передача информации между ними производится в виде транзакций.

Имеется два типа каналов:

Потоковый – однонаправленный, реализует сплошной, с прерываниями и изохронный тип передачи, без определенной структуры.

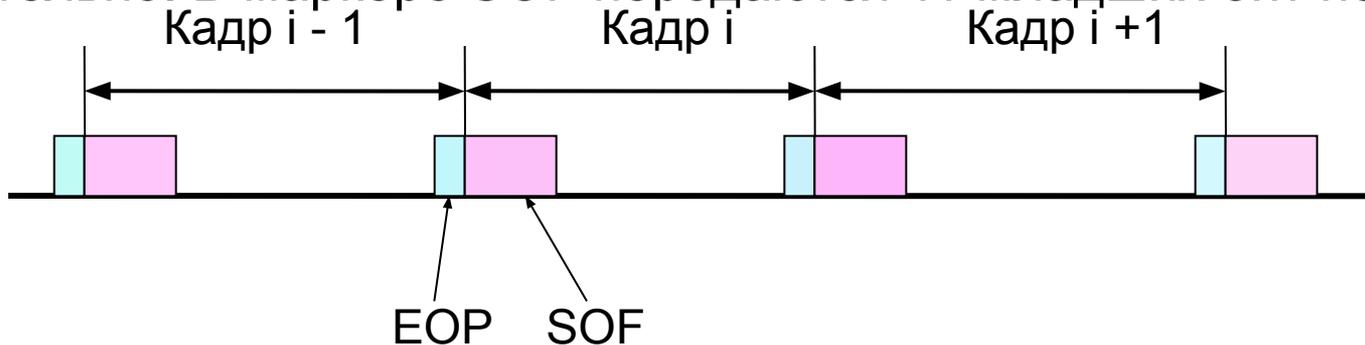
Сообщений – двунаправленный, реализует управляющий тип передачи, имеет формат определяемый спецификацией USB.

Структура модели



Протокол шины USB

Протокол шины USB обеспечивает обмен данными между хостом и устройством. Используется мультиплексирование данных с временным уплотнением. Единицей передаваемой информации является транзакция, состоящая из двух - трех пакетов. Хост-контроллер циклически с периодом 1мс(USB1) или 125 мкс(USB2) для обмена с устройствами формирует кадры, содержащие несколько транзакций. Кадры передаются согласно плану распределения ресурсов. Каждый кадр начинается с пакета-маркера начала кадра SOF (Start Of Frame), а заканчивается интервалом времени EOF(End Of Frame). Кадры нумеруются последовательно. В маркере SOF передаются 11 младших бит номера кадра



Транзакции и пакеты

Транзакция содержит определенную последовательность пакетов.

Пакет начинается с синхропоследовательности, один байт(USB1) –

В коде NRZI: 10101011. Две 1 означает начало информационной части пакета SOP (Start Of Packet). Пакет заканчивается сигналом конца пакета EOP (End Of Packet) – переводом обеих линий данных в низкое состояние SEO (Single-Ended Zero).

В зависимости от назначения пакет имеет 3 формата.

Маркер, пакет данных и пакет квитирования. Каждый пакет начинается с их идентификатора PID. Код PID задает тип пакета, т.е. его назначение и функцию. Пакет квитирования имеет только поле PID.

Рис. 4-5 Структура маркерного пакета

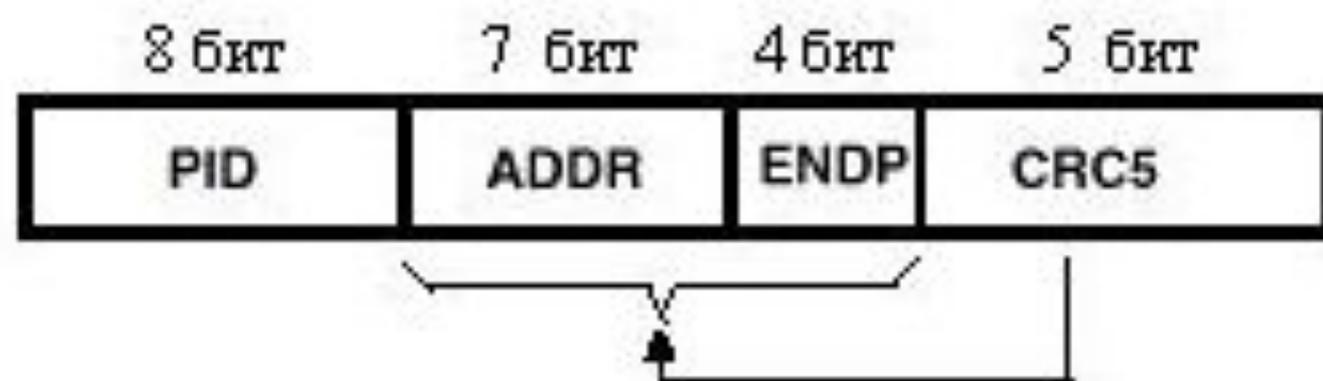


Рис. 4-1 Формат поля PID

Мл. бит

Ст. бит

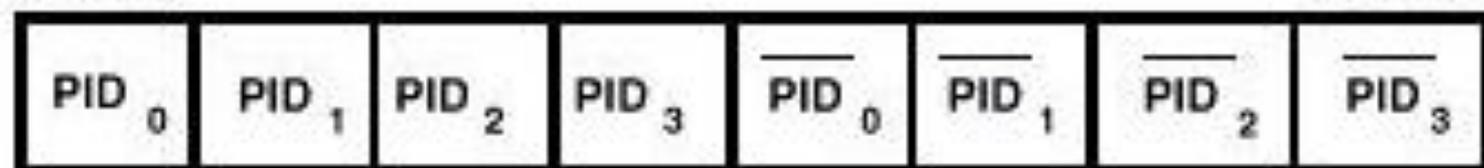


Рис. 4-2 Поле адреса функции

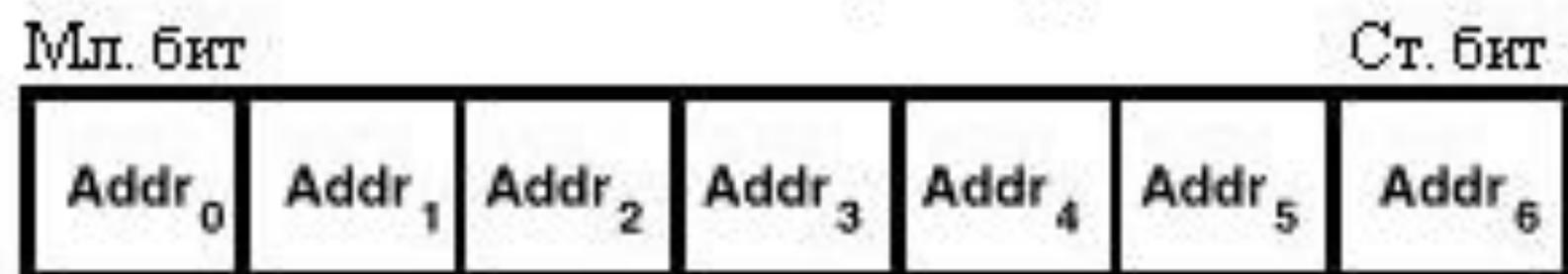


Рис. 4-3 Поле адреса конечной точки

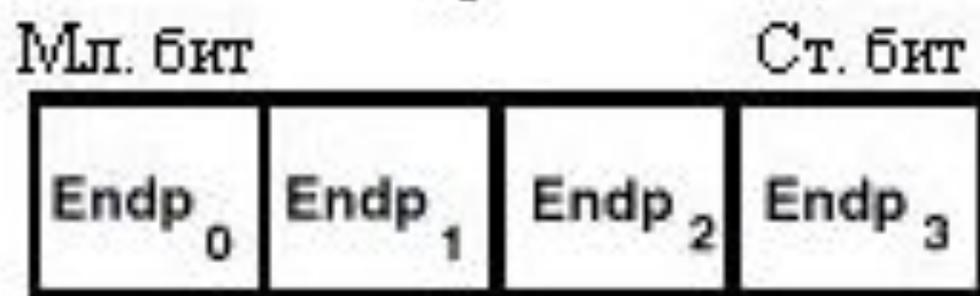


Рис. 4-6 Структура маркерного пакета типа SOF

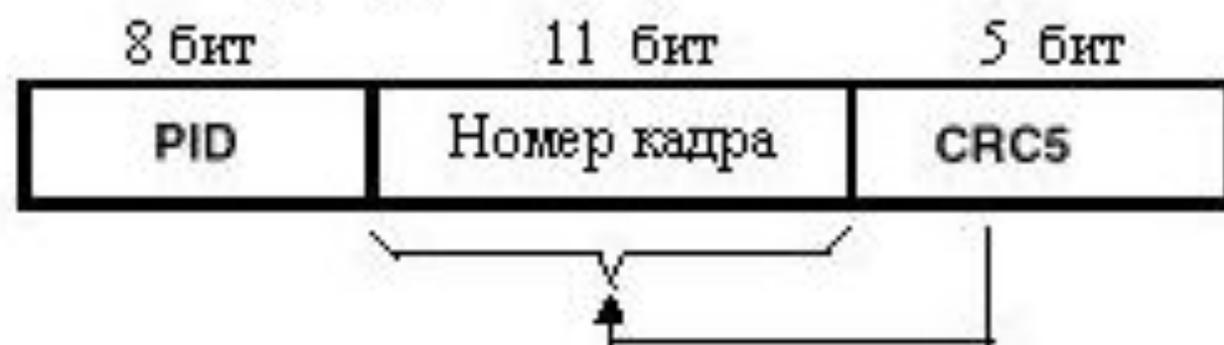


Рис. 4-7 Формат пакета данных

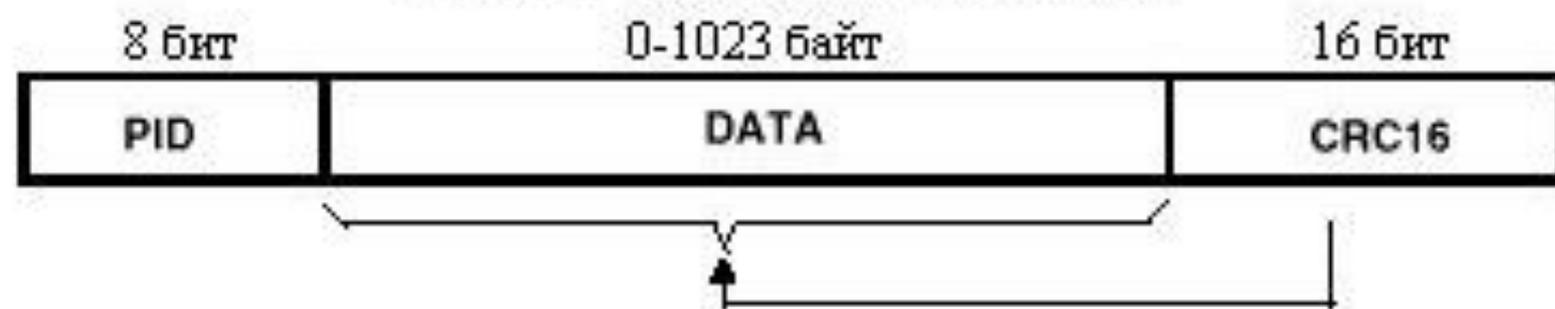


Рис. 4-4 Поле данных

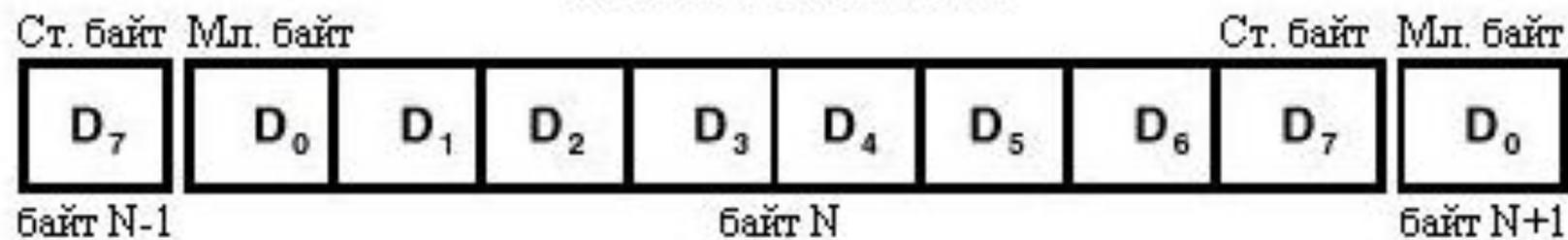
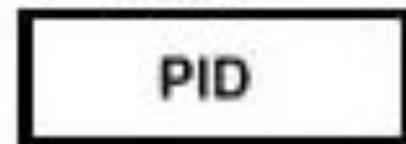


Рис. 4-8 Формат пакета подтверждения

8 бит



Типы пакетов квитирования

ACK – подтверждение безошибочного приема пакета.

NAK – индикация занятости конечной точки.

STALL – конечная точка требует вмешательство хоста. Серьезная ошибка.

NYET – безошибочный прием, но отсутствует место для следующего пакета.

Типы пакетов маркеров

OUT, IN, SETAP, SOF.

Устойчивость к ошибкам

Все принимаемые пакеты проверяются на ошибки.

Дублирование PID.

CRC код.

Вставка бит.

Счетчик тайм-аута

Рис. 4-9 Формат сплошной транзакции

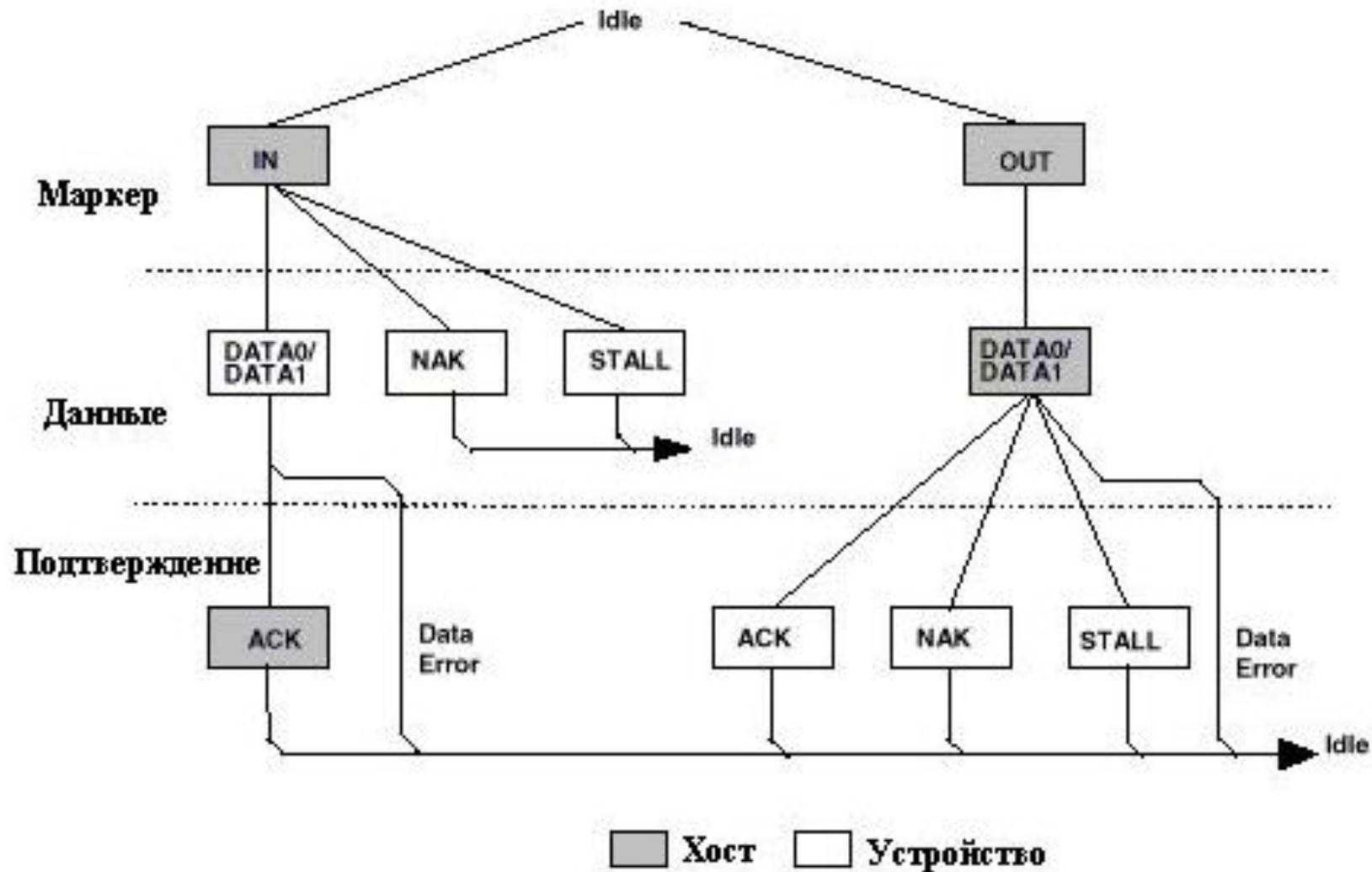


Рис. 4-10 Формат транзакции типа SETUP

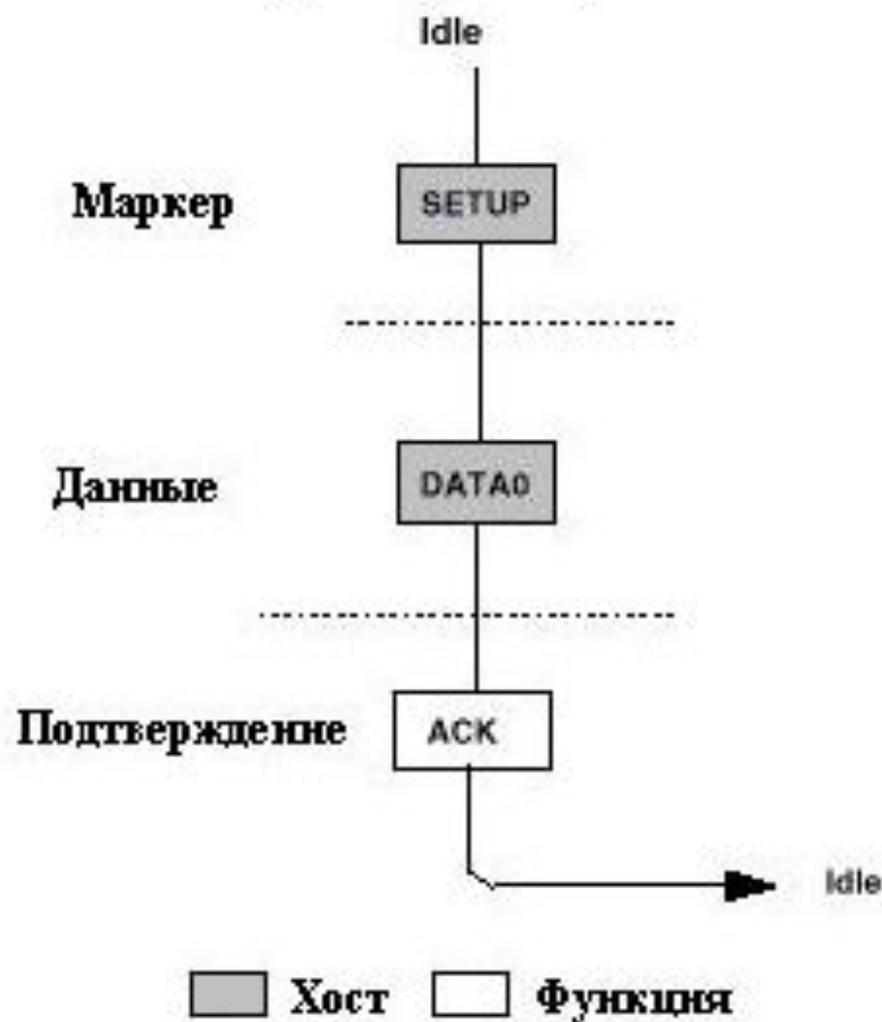


Рис. 4.106. Формат изохронных передач

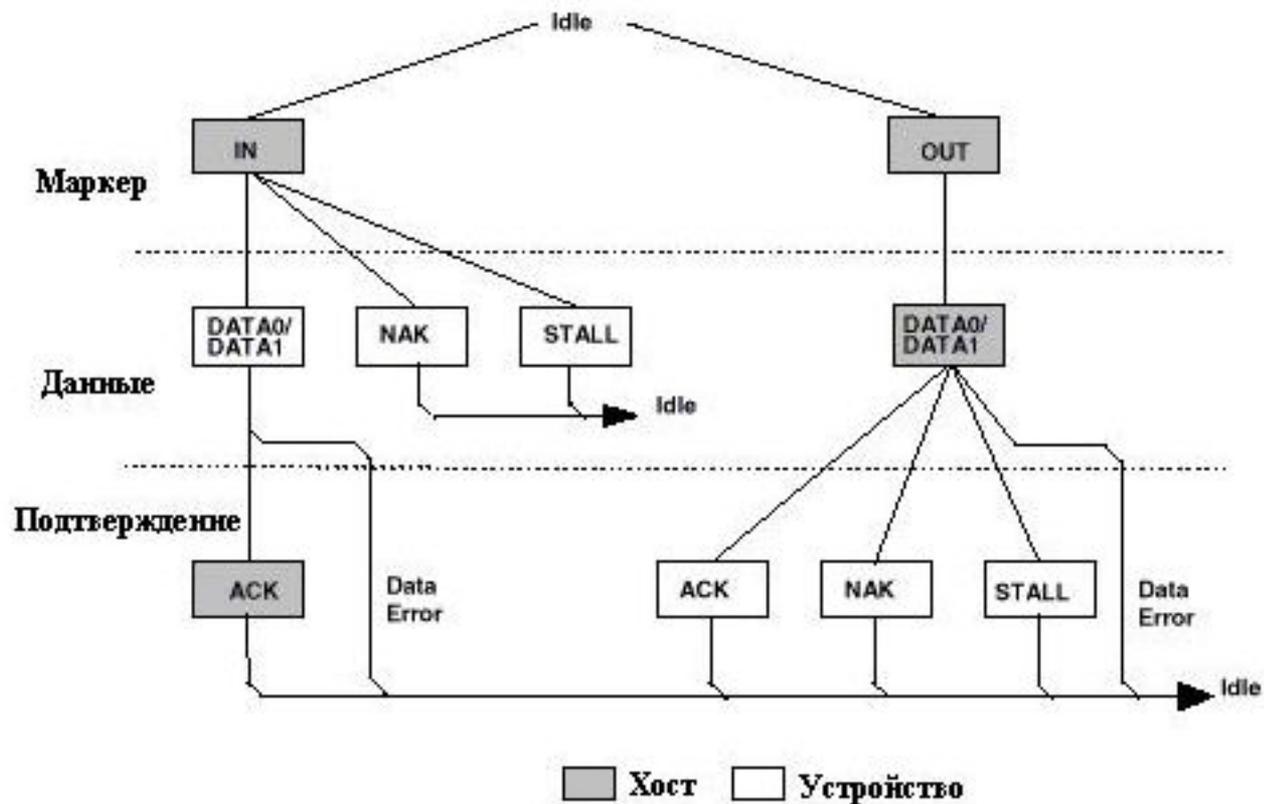


Рис. 4-11 Управляющие посылки

