

Раздел 4. Лекция 1.

Параллельный интерфейс IEEE
1284

Основные вопросы лекции

1. Интерфейс Centronics.
2. Характеристики, принцип действия, сигналы, протокол, особенности применения.
3. Стандарт IEEE 1284-2000.
4. Режимы SPP полубайтного ввода, двунаправленного байтного ввода, EPP, ECP.
5. Согласование режимов.

1-2. Основные характеристики

Интерфейс **Centronics** – это стандартный интерфейс ввода/вывода, разработанный в 1970-х годах для подключения принтеров и других устройств.

Характеристики интерфейса:

- параллельный (8 бит);
- симплексный;
- топология «точка-точка»;
- с побайтной передачей данных со скоростью до 150 КБ/сек;
- логика TTL, предназначенный для подсоединения механических печатающих устройств.



Он был разработан фирмой Epson в начале 80-х годов.

С появлением новых видов ПУ и повышения их быстродействия произошло развитие Centronics.

2. LPT-порт

В 1994 году появился стандарт IEEE 1284, где определялось три вида портов:

- **SPP** (Standard Parallel Port) – изначально был разработан для вывода данных на линейный (посимвольный) принтер.
- **EPP** (Enhanced Parallel Port) эффективен при работе с устройствами внешней памяти, адаптерами локальных сетей;
- **ЕСР** (Extended Capability Port) – эффективен для принтеров и сканеров, программаторов.

Стандарт IEEE 1284, допускает (с ограничениями) двунаправленную передачу с применением механизмов прерываний и DMA.

2. Стандартный порт LPT (SPP)

Традиционный (стандартный) порт SPP (Standard Parallel Port) является **однонаправленным** портом, на базе которого программно реализуется протокол обмена Centronics.

Предусматривает передачу 8 бит от компьютера к принтеру с полностью программным управлением – записью данных в выходной регистр и стробированием.

Возможна выдача прерывания по сигналу подтверждения от принтера, выдаваемого для каждого байта.

Используется протокол Centronics, разработанный для принтеров одноименной фирмы. Имеется также полубайтный режим (Nibble, или Bitronics), в котором половина информационных линий используется на прием.

2. Сигналы Centronics

Data0–Data7 – информационные линии

Strobe# - стробирование (наличие валидных данных Data0-7).

AutoLF# - сигнал авто-перевода строки (CR после LF). При низком символ CR (Carriage Return – возврат каретки), автоматически выполняет и функцию LF (Line Feed – перевод строки)

SelectIn# - сигнал использования интерфейса Centronics. Сигнализирует о включении принтера

Init# - инициализация (сброс принтера в режим параметров умолчания, возврат к началу строки)

Ack# - подтверждение от принтера - принят байт (запрос на прием следующего).

Busy – занято.

PE (Paper End) – нет бумаги .

Select – принтер включен

Error# – ошибка принтера: конец бумаги, состояние OFF-Line или внутренняя ошибка принтера

Итого 8 линий данных, 5 – состояния принтера и 4 управляющих сигнала.

SelectIn# – выбор принтера (низким уровнем). При высоком уровне принтер не воспринимает остальные сигналы интерфейса

GND – общий провод интерфейса

2. Интерфейс Sugart SA-400

Во всех массовых дисководах в целях сохранения совместимости с контроллерами FDC применялся один и тот же интерфейс, рассчитанный на модуляцию MFM и поддержку 4 устройств. Это по сути слегка видоизмененный интерфейс первого 5.25" дисковода – Sugart SA-400.

Интерфейс относится к категории интерфейсов на уровне устройства, т.к. содержит сигналы, характерные для функций устройства (Motor On - включить мотор, Index - проход индексной метки, Side 1 Select - выбор головки и т.п.). Скорость около 300 Кбит/с.

Данный интерфейс предусматривает соединение между контроллером FDC и дисководом (без контроллера) с подачей управляющих сигналов непосредственно на узлы дисковода (шаговый двигатель, шпиндель, головки) и снятие данных с датчиков.

Наиболее распространенный вариант интерфейса предусматривал использование плоского ленточного кабеля с 34 линиями и контактами. На этом кабеле крепятся разъемы для контроллера (штырьковый) и 4 дисководов – по два для 5.25" (с печатными двусторонними ламелями) и для 3.5" (с двухрядными штырьковыми контактами).

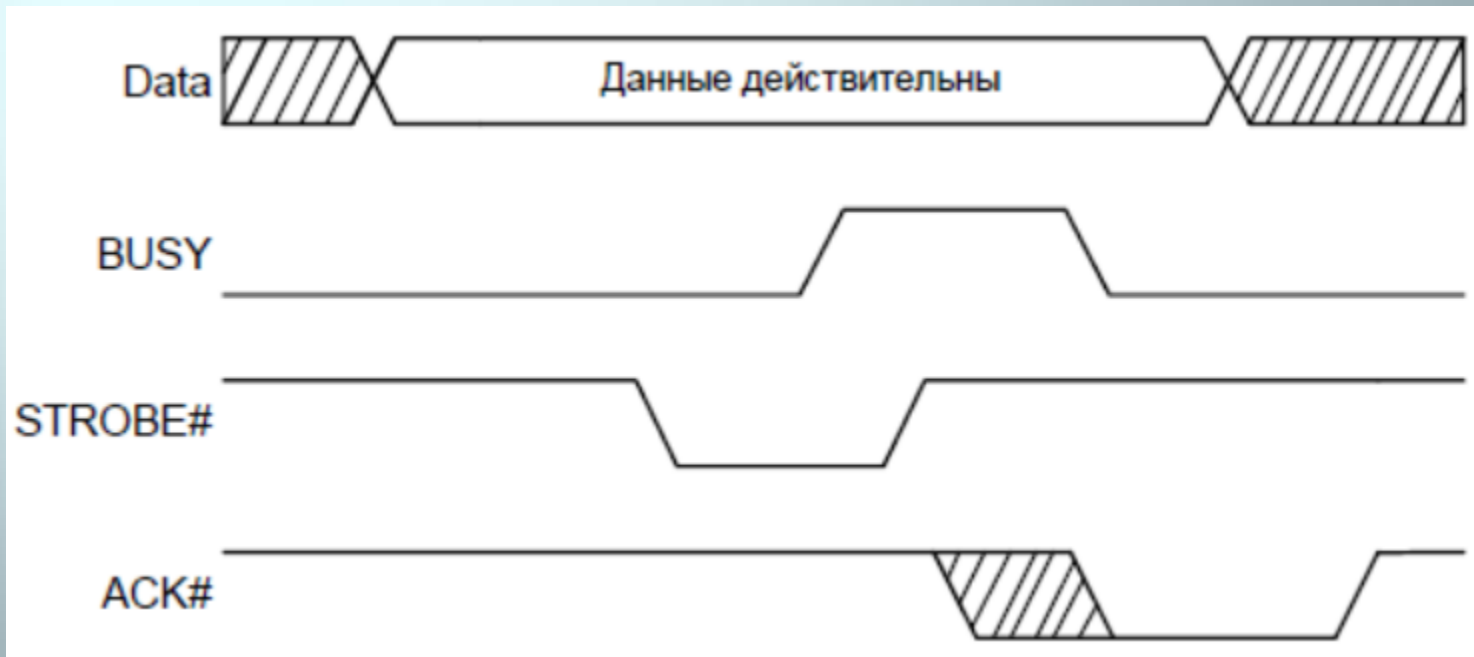
2. Временная диаграмма обмена с принтером

Data – информационные линии

Strobe# - Данные фиксируются по низкому уровню сигнала

Busy – Занято. Прием данных возможен только при низком уровне сигнала

Ack# - подтверждение от принтера - импульс подтверждения приема байта (запрос на прием следующего).



2. Программный доступ к LPT, режим SPP

Каждый порт LPT (архитектура ПК предусматривает до 3 портов) имеет 3 регистра разрядностью 8 бит, доступных через область портов в/в по стандартным адресам:

- LPT1: 0x378
- LPT2: 0x278
- LPT3: 0x3BC

Регистр DR (Data Register) – регистр данных. Данные, записанные в этот порт, выводятся на выходные линии интерфейса.

Регистр SR (Status Register) – отображение состояния входных линий плюс бит прерывания – регистр состояния (5-битный порт ввода сигналов состояния принтера).

Регистр CR – Control Register (CR) – регистр управления (4-битный порт вывода). Как и регистр данных, он допускает запись и чтение.

Использование механизма SPP требует «ручного» выставления строга, следовательно, на вывод байта нужно 5 запросов к портам (чтение SR, чтение CR, запись CR, запись DR, чтение CR).

2. Функции BIOS для LPT-порта

Поддержку LPT-порта, необходимую для организации вывода по интерфейсу Centronics, обеспечивает BIOS.

В процессе начального тестирования POST (Power-On Self-Test) BIOS проверяет наличие параллельных портов по адресам 3BCh, 378h и 278h и помещает базовые адреса обнаруженных портов в ячейки BIOS DATA AREA. Эти ячейки хранят адреса портов с логическими именами LPT1-LPT4, нулевое значение адреса является признаком отсутствия порта с данным номером.

Обнаруженные порты инициализируются – записью в регистр управления формируется и снимается сигнал Init#, после чего записывается значение 0Ch, соответствующее исходному состоянию сигналов интерфейса.

2. Программное прерывание BIOS

Программное прерывание BIOS INT 17h обеспечивает следующие функции поддержки LPT-порта:

- *00h* – вывод символа из регистра AL по протоколу Centronics (без аппаратных прерываний). Данные помещаются в выходной регистр, и, дождавшись готовности принтера (снятия сигнала BUSY), формируется строб;
- *01h* – инициализация интерфейса и принтера (установка исходных уровней управляющих сигналов, формирование импульса Init#, запрет аппаратных прерываний и переключение на вывод двунаправленного интерфейса);
- *02h* – опрос состояния принтера (чтение регистра состояния порта).

3. Стандарт IEEE 1284-2000

Определяет несколько новых режимов работы порта LPT, в том числе двунаправленные и аппаратно-управляемые:

- Compatibility mode – режим SPP – однонаправленный (вывод) по протоколу Centronics
- Nibble mode – полубайтный – ввод байта в два цикла (по 4 бита), используя для приема линии состояния. Этот режим обмена может использоваться на любых адаптерах;
- Byte mode – побайтный обмен, используя для приема линии данных, направление задается битом 5 в регистре CR . Этот режим работает только на портах, допускающих чтение выходных данных (Bi-Directional или PS/2 Type 1;
- EPP (Enhanced Parallel Port) mode – режим двунаправленного обмена данными с аппаратной генерацией управляющих сигналов интерфейса во время цикла обращения к порту (чтения или записи в порт). Эффективен при работе с устройствами внешней памяти, адаптерами локальных сетей;
- ECP (Extended Capability Port) mode – режим двунаправленного обмена данными с возможностью аппаратного сжатия данных по методу RLE (Run Length Encoding) и использования FIFO-буферов и DMA. Управляющие сигналы интерфейса генерируются аппаратно. Эффективен для принтеров и сканеров.

4. Полубайтный режим ввода

Полубайтный режим предназначен для двунаправленного обмена и может работать на всех стандартных портах. Порты имеют 5 линий ввода состояния, используя которые ПУ может посылать в хост байт тетрадами (nibble — полубайт, 4 бита) за два приема.

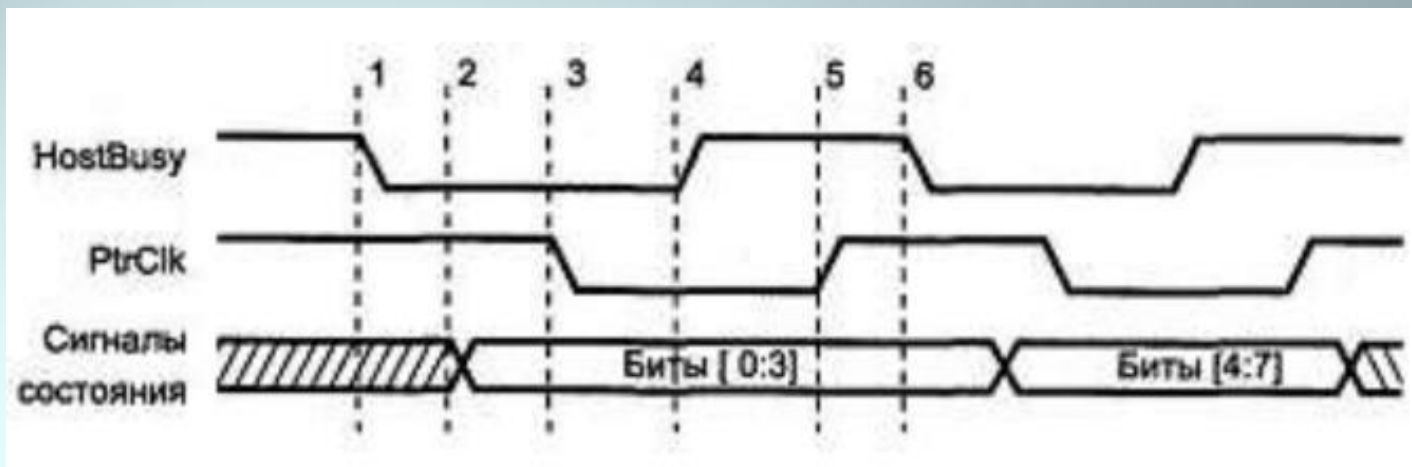
Данные на вход поступают с линий Error, Select, PE, Busy.

Сигнал Ack# - для квитирования от устройства, сигнал — AutoLF# - от компьютера: низкий уровень — готовность, высокий — подтверждение.

SelectIn выставляется в высокий уровень, сообщая об использовании режима, отличного от Centronics.

Передача начинается с понижения AutoLF#, вслед за чем устройство выставляет младшие 4 бита, понижает Ack# и ожидает повышения AutoLF# - признака приема.

4. Прием данных в полубайтном режиме



1. Хост сигнализирует о готовности приема данных установкой низкого уровня на линии HostBusy.
2. ПУ в ответ помещает тетраду на входные линии состояния.
3. ПУ сигнализирует о готовности тетрады установкой низкого уровня на линии PtrClk.
4. Хост устанавливает высокий уровень на линии HostBusy, указывая на занятость приемом и обработкой тетрады.
5. ПУ отвечает установкой высокого уровня на линии PtrClk.
6. Шаги 1-5 повторяются для второй тетрады.

4. Двухнаправленный байтный режим

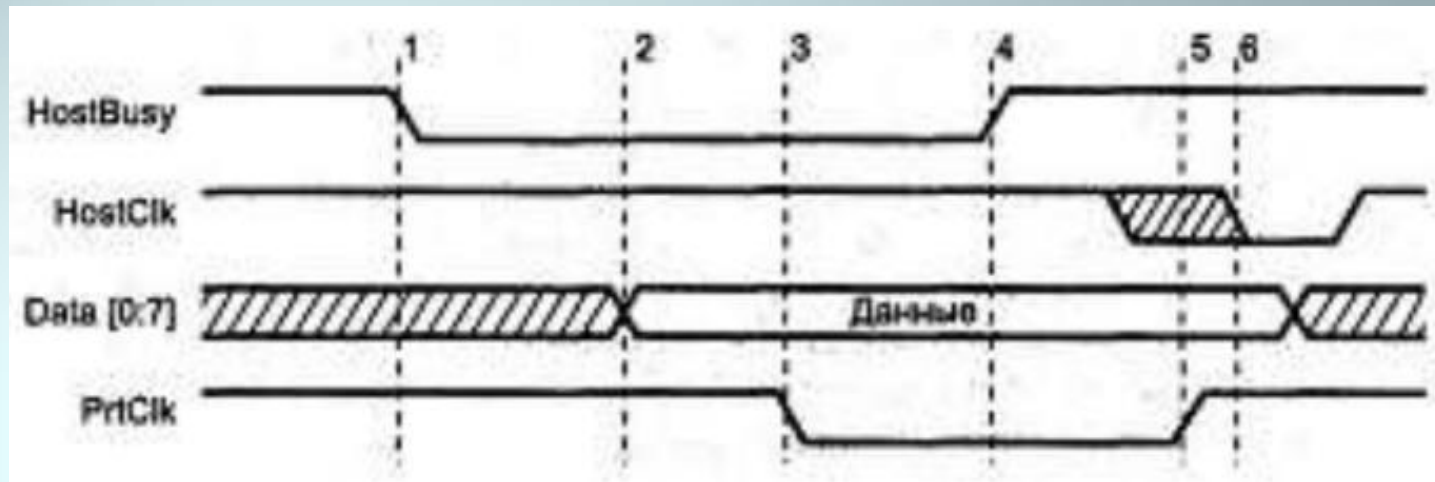
Порт может переключаться в режим приема, а не передачи, и линии `Data` используются устройством для передачи данных.

Также используется квитирование передачи, и кроме того, сигнал `Strobe` используется как запрос на следующий байт.

Назначение остальных сигналов состояния меняется.

Формирование всех выходных сигналов выполняется программно.

4. Прием данных в байтном режиме



1. Хост сигнализирует о готовности приема данных установкой низкого уровня на линии HostBusy.
2. ПУ в ответ помещает байт данных на линии Data [0:7].
3. ПУ сигнализирует о действительности байта установкой низкого уровня на линии PtrClk.
4. Хост устанавливает высокий уровень на линии HostBusy, указывая на занятость приемом и обработкой байта.
5. ПУ отвечает установкой высокого уровня на линии PtrClk.
6. Хост подтверждает прием байта импульсом HostClk.

4. Режим EPP

Предназначен для повышения производительности обмена по параллельному порту. EPP был реализован в чипсете Intel 386SL (микросхема 82360) и впоследствии принят множеством компаний как дополнительный протокол параллельного порта.

Использует аппаратное управление сигналами. Имеются сигналы выбора направления передачи (`Strobe`), стробирования для передачи данных и адресов, квитирования от устройства (`Busy`), прерывания от устройства.

Вся операция выполняется за один цикл обращения к регистрам EPP, которые дополняют стандартные регистры порта LPT.

Для устранения возможности зависания (отсутствие подтверждения от устройства) имеется механизм тайм-аута.

Регистр EPP Data может иметь разрядность 16 или 32 бита для выполнения 2 и 4 обменов соответственно за одно обращение к портам.

4. Циклы обмена протокола EPP

Протокол EPP обеспечивает четыре типа циклов обмена:

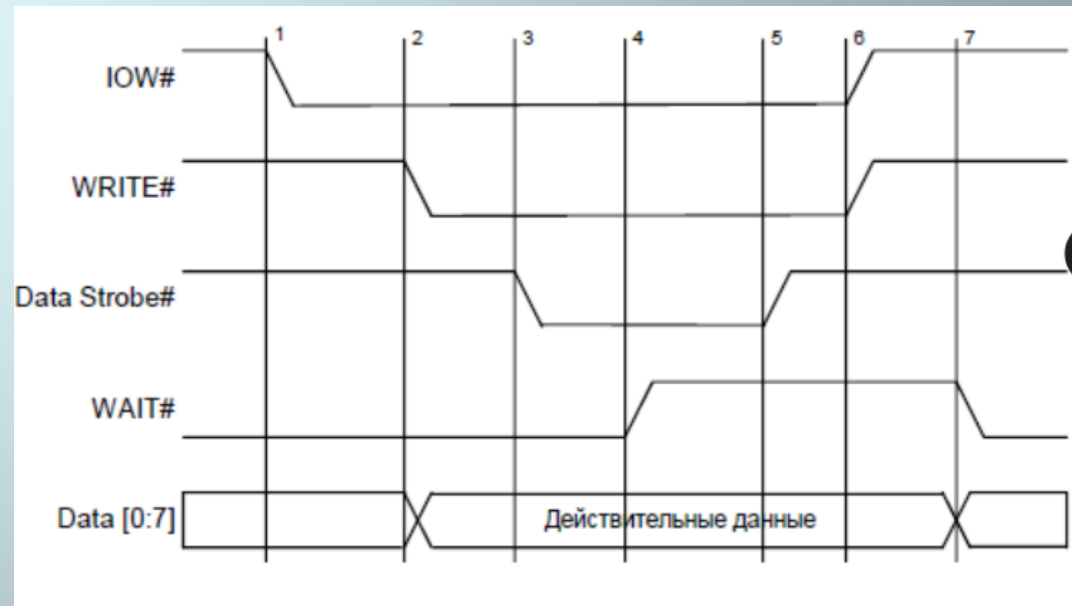
- цикл записи данных;
- цикл чтения данных;
- цикл записи адреса;
- цикл чтения адреса.

Адресные циклы могут быть использованы для передачи адресной, канальной и управляющей информации. Циклы обмена данными явно отличаются от адресных циклов применяемыми стробирующими сигналами.

4. Циклы записи данных. Временная диаграмма

Цикл записи данных состоит из следующих фаз:

- 1) Программа выполняет цикл записи ($IOWR\#$) в порт 4 (EPP Data Port).
- 2) Адаптер устанавливает сигнал $Write\#$ (низкий уровень), и данные помещаются на выходную шину LPT-порта.
- 3) При низком уровне $WAIT\#$ устанавливается строб данных.
- 4) Порт ждет подтверждения от ПУ (перевода $WAIT\#$ в высокий уровень).
- 5) Снимается строб данных – внешний цикл завершается.
- 6) Завершается процессорный цикл ввода/вывода.
- 7) ПУ устанавливает низкий уровень $WAIT\#$, указывая на возможность начала следующего цикла.



4. Переопределение сигналов SPP

Протокол EPP **переопределяет** сигналы SPP

Сигналы SPP	Имя в EPP	I/O	Описание
STROBE#	WRITE#	O	Низкий уровень – признак цикл записи, высокий – чтения
AUTOFEEM#	DATASTB#	O	Строб данных. Низкий уровень устанавливается в циклах передачи данных
BUSY#	WAIT#	I	Сигнал квитирования. Низкий уровень разрешает начало цикла (установку строба в низкий уровень), переход в высокий – разрешает завершение цикла (снятие)

4. Переопределение сигналов SPR

Главной отличительной чертой EPP является выполнение внешней передачи во время одного процессорного цикла ввода/вывода. Это позволяет достигать высоких скоростей обмена (0,5-2 Мбайт/с).

Прозрачный протокол блокированного квитирования (interlocked handshakes) позволяет автоматически настраиваться на скорость обмена, доступную и хосту, и периферийному устройству. Периферийное устройство может регулировать длительность всех фаз обмена с помощью всего лишь одного сигнала WAIT#.

Протокол автоматически подстраивается и под длину кабеля – вносимые задержки только приведут к удлинению цикла.

4. Режим ECP

Протокол ECP в обоих направлениях обеспечивает два типа циклов:

- циклы записи и чтения данных;
- командные циклы записи и чтения.

Командные циклы подразделяются на два типа:

- передача канальных адресов;
- счетчика RLC (Run-Length Count).

Как и в других режимах, выполняется переопределение сигналов.

4. Описание основных сигналов

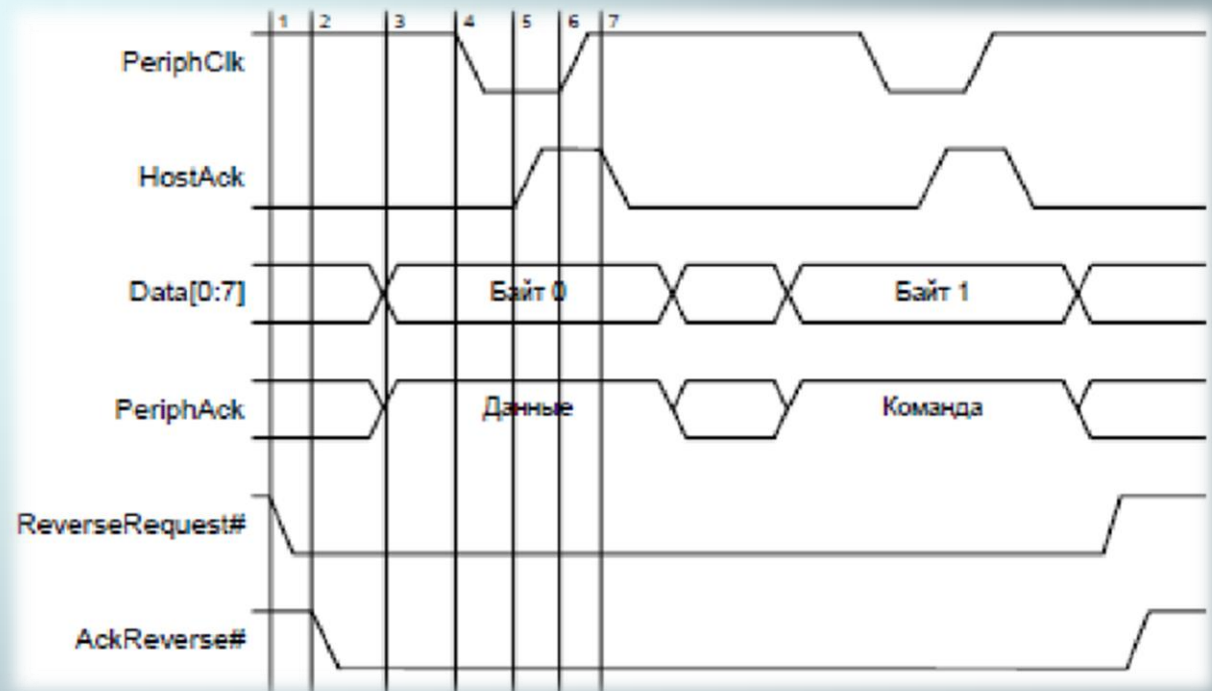
Конт	Сигнал	Имя в ECP	I/O	Описание
1	STROBE#	HostClk	O	Используется в паре с PeriphAck для передачи в прямом направлении (вывод)
14	AUTOFEEDS	HostAck	O	Индицирует тип команда/данные при передаче в прямом направлении. Используется в паре с PeriphClk для передачи в обратном направлении
17	SELECTING	1284Active	O	Высокий уровень указывает на обмен в режиме IEEE 1284. (В режиме SPP уровень низкий)
16	INIT#	ReverseRequest#	O	Низкий уровень переключает канал на передачу в обратном направлении
10	ACK#	PeriphClk	I	Используется в паре с HostAck для передачи в обратном направлении
11	BUSY	PeriphAck	I	Используется в паре с HostClk для передачи в обратном направлении. Индицирует тип команда/данные при передаче в обратном направлении
12	PE	AckReverse#	I	Переводится в низкий уровень как подтверждение сигналу ReverseRequest#
13	SELECT	Xflag*	I	Флаг расширяемости Extensibility flag

4. Диаграмма двух циклов прямой передачи

За циклом данных следует командный цикл.

Тип цикла задается уровнем на линии HostAck: в цикле данных – высокий, в командном цикле – низкий. В командном цикле байт может содержать каналный адрес или счетчик RLE.

Отличительным признаком является бит 8 (старший): если он нулевой, то биты 1- 7 содержат счетчик RLE (0-127), если единичный – то каналный адрес.

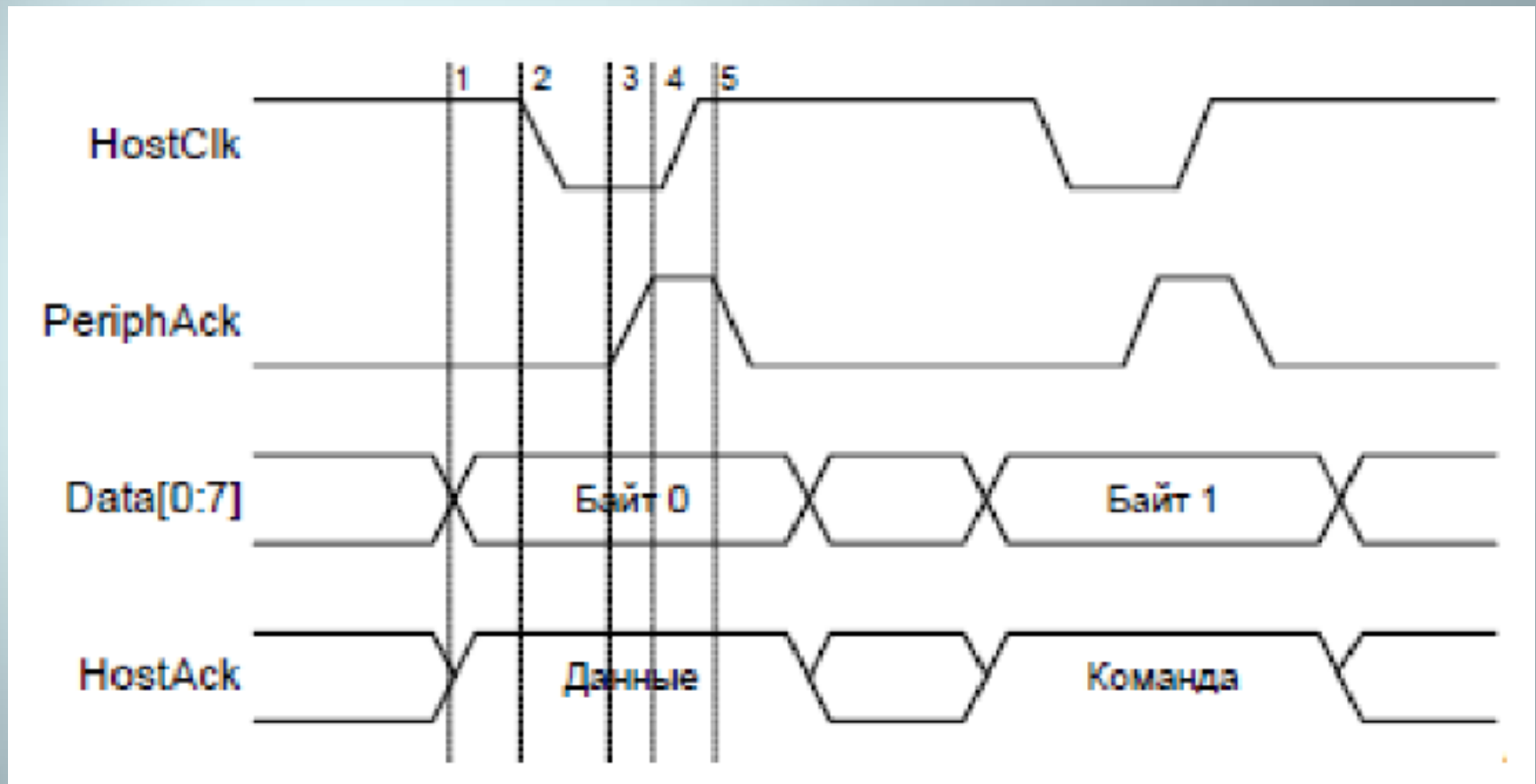


4. Прямая передача данных

Прямая передача данных на внешнем интерфейсе состоит из следующих шагов:

1. Хост помещает данные на шину канала и устанавливает признак цикла данных (высокий уровень) или команды (низкий уровень) на линии `HostAck`.
2. Хост устанавливает низкий уровень на линии `HostClk`, указывая на действительность данных.
3. ПУ отвечает установкой высокого уровня на линии `PeriphAck`.
4. Хост устанавливает высокий уровень линии `HostClk`, и этот перепад может использоваться для фиксации данных в ПУ.
5. ПУ устанавливает низкий уровень на линии `PeriphAck` для указания на готовность к приему следующего байта.

4. Пара циклов обратной передачи



4. Обратная передача данных

Обратная передача данных состоит из следующих шагов:

1. Хост запрашивает изменение направления канала, устанавливая низкий уровень на линии `ReverseRequest#`.
2. ПУ разрешает смену направления установкой низкого уровня на линии `AckReverse#`.
3. ПУ помещает данные на шину канала и устанавливает признак цикла данных (высокий уровень) или команды (низкий уровень) на линии `PeriphAck`.
4. ПУ устанавливает низкий уровень на линии `PeriphClk`, указывая на действительность данных.
5. Хост отвечает установкой высокого уровня на линии `HostAck`.
6. ПУ устанавливает высокий уровень линии `PeriphClk`, и этот перепад может использоваться для фиксации данных хостом.
7. Хост устанавливает низкий уровень на линии `HostAck` для указания на готовность к приему следующего байта.

4. Отличие ECP от ERP

Протокол ERP позволяет драйверу чередовать циклы прямой и обратной передачи, не запрашивая подтверждения на смену направления.

В ECP смена направления должна быть согласована: хост запрашивает реверс установкой `ReverseRequest#`, после чего он должен дожидаться его подтверждения сигналом `AckReverse#`. Только после этого возможна передача данных в другом направлении.

4. Физический и электрический интерфейс

IEEE 1284 определяет два уровня интерфейсной совместимости:

- ❖ первый уровень (Level I) определен для устройств, не претендующих на высокоскоростные режимы обмена, но использующих возможности смены направления передачи данных;
- ❖ второй уровень (Level II) определен для устройств, работающих в расширенных режимах, с высокими скоростями и длинными кабелями.

К передатчикам предъявляются следующие требования:

- уровни сигналов без нагрузки не должны выходить за пределы - 0,5...+5,5 В;
- уровни сигналов при токе нагрузки 14 мА должны быть не ниже +2,4 В для высокого уровня и не выше +0,4 В для низкого уровня на постоянном токе;
- выходной импеданс, измеренный на разъеме, должен составлять 50(±)5 Ом на уровне VoH-VoL. Для обеспечения заданного импеданса в некоторых случаях используют последовательные резисторы в выходных цепях передатчика. Согласование импеданса передатчика и кабеля снижает уровень импульсных помех;
- скорость нарастания (спада) импульса должна находиться в пределах 0,05-0,4 В/нс.

4. Конфигурирование LPT-портов

Управление параллельным портом разделяется на два этапа – предварительное конфигурирование (Setup) аппаратных средств порта и текущее (оперативное) переключение режимов работы прикладным или системным ПО. Оперативное переключение возможно только в пределах режимов, разрешенных при конфигурировании. Таким образом обеспечивается возможность согласования аппаратуры и программного обеспечения и блокирования ложных переключений, вызванных некорректными действиями программы.

Порт, расположенный на системной плате, обычно конфигурируется через BIOS Setup.

Конфигурированию подлежат следующие параметры:

1. Базовый адрес, который может иметь значение 3BCh, 378h и 278h.
2. Используемая линия запроса прерывания: для LPT1 обычно используется IRQ7, для LPT2 – IRQ5.
3. Использование канала DMA для режимов ECP и Fast Centronics – разрешение и номер канала DMA.

4. Режимы работы порта

Режим работы порта может быть задан в следующих вариантах:

- *SPP* – порт работает только в стандартном однонаправленном программно-управляемом режиме;
- *PS/2*, он же *Bi-Directional* – отличается от *SPP* возможностью реверса канала (с помощью установки CR.5=1);
- *Fast Centronics* – аппаратное формирование протокола Centronics с использованием FIFO-буфера и, возможно, DMA;
- *EPP* – в зависимости от использования регистров, порт работает в режиме *SPP* или *EPP*;
- *ECP* – по умолчанию включается в режим *SPP* или *PS/2*, записью в *ЕСК* может переводиться в любой режим *ECP*- *ECP+EPP* – то же, что и *ECP*, но запись в *ECR* кода режима 100 переводит порт в режим *EPP*.

4. Использование параллельных портов

Наиболее распространенным применением LPT-порта является, естественно, **подключение принтера**.

- *двунаправленный режим (Bi-Di)* не повышает производительность, но дает дополнительные возможности для сообщения состояния и параметров принтера. Скоростные режимы (Fast Centronics) существенно повышают производительность практически любого принтера (особенно лазерного), но могут потребовать более качественного кабеля. От принтера этот режим не требует каких-либо дополнительных — интеллектуальных способностей;
- *режим ECP* потенциально самый эффективный, и он имеет системную поддержку во всех вариантах Windows. Однако он реализует свои способности (включая аппаратную компрессию) не на всех принтерах. Из распространенных семейств ECP поддерживают принтеры HP DeskJet моделей 6xx, LaserJet начиная с 4-го, современные модели фирмы Lexmark. Требуется применения кабеля, по частотным свойствам соответствующего IEEE 1284.

4. Параллельный порт и PnP

Большинство современных периферийных устройств, подключаемых к LPT-порту, поддерживает стандарт 1284 и функции PnP. Для поддержки этих функций компьютером с аппаратной точки зрения достаточно иметь контроллер интерфейса, поддерживающий стандарт 1284.

Если подключаемое устройство поддерживает PnP, оно по протоколу согласования режимов 1284 способно — договориться с портом, представляющим — интересы компьютера, о возможных режимах обмена.

Далее, для работы PnP подключенное устройство должно сообщить операционной системе все необходимые сведения о себе. Как минимум это идентификаторы производителя, модели и набор поддерживаемых команд. Более развернутая информация об устройстве может содержать идентификатор класса, подробное описание и идентификатор устройства, с которым обеспечивается совместимость.

В соответствии с принятой информацией операционная система может предпринять действия по установке требуемого программного обеспечения поддержки данного устройства.

5. Согласование режимов IEEE 1284

ПУ в стандарте IEEE 1284 обычно не требуют от контроллера реализации всех предусмотренных этим стандартом режимов. Для определения режимов и методов управления конкретным устройством стандарт предусматривает последовательность согласования (negotiation sequence).

Последовательность построена таким образом, что старые устройства, не поддерживающие IEEE 1284 и на ее запрос не ответят, и контроллер останется в стандартном режиме. Периферия IEEE 1284 может сообщить о своих возможностях, и контроллер установит режим, удовлетворяющий и хост, и ПУ.

- Во время фазы согласования контроллер выставляет на линии данных байт расширяемости (extendibility byte), запрашивая подтверждение на перевод интерфейса в требуемый режим или прием идентификатора ПУ.
- Идентификатор передается контроллеру в запрошенном режиме (любой режим обратного канала, кроме EPP).
- ПУ использует сигнал Xflag (Select в терминах SPP) для подтверждения запрошенного режима обратного канала, кроме полубайтного, который поддерживается всеми устройствами IEEE 1284. Бит Extensibility Link request послужит для определения дополнительных режимов в будущих расширениях стандарта.

5. Последовательность согласования

1. Хост выводит байт расширяемости на линии данных.
2. Хост устанавливает высокий уровень сигнала `SelectIn*` и низкий — `AutoFeed*`, что означает начало последовательности согласования.
3. ПУ отвечает установкой низкого уровня сигнала `Ack#` и высокого — `Error*`, `PaperEnd` и `Select`. Устройство, «не понимающее» стандарта 1284, ответа не даст, и дальнейшие шаги не выполняются.
4. Хост устанавливает низкий уровень сигнала `Strobe*` для записи байта расширяемости в ПУ.
5. Хост устанавливает высокий уровень сигналов `Strobe*` и `AutoLF*`.
6. ПУ отвечает установкой в низкий уровень сигналов `PAPEREnd` и `Error*`, если ПУ имеет обратный канал передачи данных. Если запрошенный режим поддерживается устройством, на линии `Select` устанавливается высокий уровень, если не поддерживается — низкий.
7. ПУ устанавливает высокий уровень на линии `Ack#` для указания на завершение последовательности согласования, после чего контроллер задает требуемый режим работы.

5. Последовательность согласования. Временная диаграмма

