Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное

учреждение высшего образования

«Воронежский государственный лесотехнический университет

имени Г.Ф. Морозова»

Реферат

По теме:

«Интерфейс USB и поддерживаемый протокол обмена»

|  |  |  |
| --- | --- | --- |
| Студент группы ИС2-191-ОБ  (номер группы)  Руководитель, к.т.н. доцент  (ученая степень, ученое звание) | \_\_\_\_\_\_\_\_\_\_  (подпись)  \_\_\_\_\_\_\_\_\_\_  (подпись) | Величко В. А.  (инициалы и фамилия)  Заревич А. И.  (инициалы и фамилия) |

Воронеж 2022

**Оглавление**

[1. Общая характеристика 3](#_Toc106562528)

[2. Структура USB 4](#_Toc106562529)

[3. Физический интерфейс 7](#_Toc106562530)

[4. Модель передачи данных 10](#_Toc106562531)

[5. Типы передачи данных 12](#_Toc106562532)

[6. Протокол 14](#_Toc106562533)

[7. Форматы пакетов 15](#_Toc106562534)

[8. Системное конфигурирование 17](#_Toc106562535)

[9. Устройства USB - функции и хабы 19](#_Toc106562536)

[10. Хост-контроллер 23](#_Toc106562537)

1. Общая характеристика

USB (Universal Serial Bus — универсальная последовательная шина) является промышленным стандартом расширения архитектуры PC, ориентированным на интеграцию с телефонией и устройствами бытовой электроники. Версия 1.0 была опубликована в январе 1996 года. Архитектура USB определяется следующими критериями:

Легко реализуемое расширение периферии PC.

Дешевое решение, поддерживающее скорость передачи до 12 Mбит/с.

Полная поддержка в реальном времени передачи аудио и (сжатых) видеоданных.

Гибкость протокола смешанной передачи изохронных данных и асинхронных сообщений.

Интеграция с выпускаемыми устройствами.

Доступность в PC всех конфигураций и размеров.

Обеспечение стандартного интерфейса, способного быстро завоевать рынок.

Создание новых классов устройств, расширяющих PC.

С точки зрения конечного пользователя, привлекательны следующие черты USB:

Простота кабельной системы и подключений.

Скрытие подробностей электрического подключения от конечного пользователя.

Самоидентифицирующиеся ПУ, автоматическая связь устройств с драйверами и конфигурирование.

Возможность динамического подключения и конфигурирования ПУ.

С середины 1996 года выпускаются PC со встроенным контроллером USB, реализуемым чипсетом. Уже появились модемы, клавиатуры, сканеры, динамики и другие устройства ввода/вывода с поддержкой USB, а также мониторов с USB-адаптерами - они играют роль концентраторов для подключения других устройств.

2. Структура USB

USB обеспечивает одновременный обмен данными между хост-компьютером и множеством периферийных устройств (ПУ). Распределение пропускной способности шины между ПУ планируется хостом и реализуется им с помощью посылки маркеров. Шина позволяет подключать, конфигурировать, использовать и отключать устройства во время работы хоста и самих устройств.

Ниже приводится авторский вариант перевода терминов из спецификации "Universal Serial Bus Specification", опубликованной Compaq, DEC, IBM, Intel, Microsoft, NEC и Northern Telecom. Более подробную и оперативную информацию можно найти по адресу: http://www.usb.org/.

Устройства (Device) USB могут являться хабами, функциями или их комбинацией. Хаб (Hub) обеспечивает дополнительные точки подключения устройств к шине. Функции (Function) USB предоставляют системе дополнительные возможности, например подключение к ISDN, цифровой джойстик, акустические колонки с цифровым интерфейсом и т. п. Устройство USB должно иметь интерфейс USB, обеспечивающий полную поддержку протокола USB, выполнение стандартных операций (конфигурирование и сброс) и предоставление информации, описывающей устройство. Многие устройства, подключаемые к USB, имеют в своем составе и хаб, и функции. Работой всей системы USB управляет хост-контроллер (Host Controller), являющийся программно-аппаратной подсистемой хост-компьютера.

Физическое соединение устройств осуществляется по топологии многоярусной звезды. Центром каждой звезды является хаб, каждый кабельный сегмент соединяет две точки - хаб с другим хабом или с функцией. В системе имеется один (и только один) хост-контроллер, расположенный в вершине пирамиды устройств и хабов. Хост-контроллер интегрируется с корневым хабом (Root Hub), обеспечивающим одну или несколько точек подключения - портов. Контроллер USB, входящий в состав чипсетов, обычно имеет встроенный двухпортовый хаб. Логически устройство, подключенное к любому хабу USB и сконфигурированное (см. ниже), может рассматриваться как непосредственно подключенное к хост-контроллеру.

Функции представляют собой устройства, способные передавать или принимать данные или управляющую информацию по шине. Типично функции представляют собой отдельные ПУ с кабелем, подключаемым к порту хаба. Физически в одном корпусе может быть несколько функций со встроенным хабом, обеспечивающим их подключение к одному порту. Эти комбинированные устройства для хоста являются хабами с постоянно подключенными устройствами-функциями.

Каждая функция предоставляет конфигурационную информацию, описывающую возможности ПУ и требования к ресурсам. Перед использованием функция должна быть сконфигурирована хостом - ей должна быть выделена полоса в  
канале и выбраны опции конфигурации.

Примерами функций являются:

Указатели - мышь, планшет, световое перо.  
 Устройства ввода - клавиатура или сканер.

Устройство вывода - принтер, звуковые колонки (цифровые).

Телефонный адаптер ISDN.

Хаб - ключевой элемент системы РпР в архитектуре USB.  
Хаб является кабельным концентратором. Точки подключения называются портами хаба. Каждый хаб преобразует одну точку подключения в их множество. Архитектура допускает соединение нескольких хабов.

У каждого хаба имеется один восходящий порт (Upstream Port), предназначенный для подключения к хосту или хабу верхнего уровня. Остальные порты являются нисходящими (Downstream Ports), предназначенными для подключения функций или хабов нижнего уровня. Хаб может распознать подключение устройств к портам или отключение от них и управлять подачей питания на их сегменты. Каждый из портов может быть разрешен или запрещен и сконфигурирован на полную или ограниченную скорость обмена. Хаб обеспечивает изоляцию сегментов с низкой скоростью от высокоскоростных.

Хабы могут управлять подачей питания на нисходящие порты; предусматривается установка ограничения на ток, потребляемый каждым портом.

Система USB разделяется на три уровня с определенными правилами взаимодействия. Устройство USB содержит интерфейсную часть, часть устройства и функциональную часть. Хост тоже делится на три части - интерфейсную, системную и ПО устройства. Каждая часть отвечает только за определенный круг задач, логическое и реальное взаимодействие между ними иллюстрирует.

В рассматриваемую структуру входят следующие элементы:

Физическое устройство USB - устройство на шине, выполняющее функции, интересующие конечного пользователя.

Client SW - ПО, соответствующее конкретному устройству, исполняемое на хост-компьютере. Может являться составной частью ОС или специальным продуктом.

USB System SW - системная поддержка USB, независимая от конкретных устройств и клиентского ПО.

USB Host Controller - аппаратные и программные средства для подключения устройств USB к хост-компьютеру.

3. Физический интерфейс

Стандарт USB определяет электрические и механические спецификации шины. Информационные сигналы и питающее напряжение 5 В передаются по четырехпроводному кабелю. Используется дифференциальный способ передачи сигналов D+ и D- по двум проводам. Уровни сигналов передатчиков в статическом режиме должны быть ниже 0,3 В (низкий уровень) или выше 2,8 В (высокий уровень). Приемники выдерживают входное напряжение в пределах - 0,5...+3,8 В. Передатчики должны уметь переходить в высокоимпедансное состояние для двунаправленной полудуплексной передачи по одной паре проводов.

Передача по двум проводам в USB не ограничивается дифференциальными сигналами. Кроме дифференциального приемника каждое устройство имеет линейные приемники сигналов D+ и D-, а передатчики этих линий управляются  
индивидуально. Это позволяет различать более двух состояний линии, используемых для организации аппаратного интерфейса. Состояния Diff0 и Diff1 определяются по разности потенциалов на линиях D+ и D- более 200 мВ при условии, что на одной из них потенциал выше порога срабатывания VSE. Состояние, при котором на обоих входах D+ и D- присутствует низкий уровень, называется линейным нулем (SEO - Single-Ended Zero). Интерфейс определяет следующие состояния:

Data J State и Data К State - состояния передаваемого  
бита (или просто J и К), определяются через состояния  
Diff0 и Diff1.

Idle State - пауза на шине.

Resume State - сигнал "пробуждения" для вывода устройства из "спящего" режима.

Start of Packet (SOP) - начало пакета (переход из Idle State в К).

End of Packet (EOP) - конец пакета.  
 Disconnect - устройство отключено от порта.  
 Connect - устройство подключено к порту.  
 Reset - сброс устройства.

Состояния определяются сочетаниями дифференциальных и линейных сигналов; для полной и низкой скоростей состояния DiffO и Diff1 имеют противоположное назначение.  
В декодировании состояний Disconnect, Connect и Reset учитывается время нахождения линий (более 2,5 мс) в определенных состояниях.

Шина имеет два режима передачи. Полная скорость передачи сигналов USB составляет 12 Мбит/с, низкая - 1,5 Мбит/с. Для полной скорости используется экранированная витая пара с импедансом 90 Ом и длиной сегмента до 5 м, для низкой - невитой неэкранированньгй кабель до 3 м. Низкоскоростные кабели и устройства дешевле высокоскоростных. Одна и та же система может одновременно использовать оба режима; переключение для устройств осуществляется прозрачно.  
Низкая скорость предназначена для работы с небольшим количеством ПУ, не требующих высокой скорости. Скорость, используемая устройством, подключенным к конкретному порту, определяется хабом по уровням сигналов  
на линиях D+ и D-, смещаемых нагрузочными резисторами R2 приемопередатчиков.

Сигналы синхронизации кодируются вместе с данными по методу NRZI (Non Return to Zero Invert), его работу иллюстрирует. Каждому пакету предшествует поле синхронизации SYNC, позволяющее приемнику настроиться на частоту передатчика. Кабель также имеет линии VBus и GND для передачи питающего напряжения 5 В к устройствам. Сечение проводников выбирается в соответствии с длиной сегмента для обеспечения гарантированного уровня сигнала и питающего напряжения.

Разъемы типа "А" применяются для подключения к хабам (Upstream Connector). Вилки устанавливаются на кабелях, не отсоединяемых от устройств (например, клавиатура, мышь и т. п.). Гнезда устанавливаются на нисходящих портах (Downstream Port) хабов. Разъемы типа "В" (Downstream Connector) устанавливаются  
на устройствах, от которых соединительный кабель может отсоединяться (принтеры и сканеры). Ответная часть (вилка) устанавливается на соединительном кабеле, противоположный конец которого имеет вилку типа "А".

Разъемы типов "А" и "В" различаются механически что исключает недопустимые петлевые соединения портов хабов. Четырехконтактные разъемы имеют ключи, исключающие неправильное присоединение. Конструкция разъемов обеспечивает позднее соединение и раннее отсоединение сигнальных цепей по сравнению с питающими. Для распознавания разъема USB на корпусе устройства ставится стандартное символическое обозначение.

Питание устройств USB возможно от кабеля (Bus-Powered Devices) или от собственного блока питания (Self-Powered Devices). Хост обеспечивает питанием непосредственно подключенные к нему ПУ. Каждый хаб, в свою очередь, обеспечивает питание устройств, подключенных к его нисходящим портам. При некоторых ограничениях топологии допускается применение хабов, питающихся от шины. На приведен пример схемы соединения устройств USB. Здесь клавиатура, перо и мышь могут питаться от шины.

4. Модель передачи данных

Каждое устройство USB представляет собой набор независимых конечных точек (Endpoint), с которыми хост-контроллер обменивается информацией. Конечные точки описываются следующими параметрами:

требуемой частотой доступа к шине и допустимыми задержками обслуживания;

требуемой полосой пропускания канала;

номером точки;

требованиями к обработке ошибок;

максимальными размерами передаваемых и принимаемых  
пакетов;

типом обмена;

направлением обмена (для сплошного и изохронного обменов).

Каждое устройство обязательно имеет конечную точку с номером 0, используемую для инициализации, общего управления и опроса его состояния. Эта точка всегда сконфигурирована при включении питания и подключении устройства к шине. Оно поддерживает передачи типа "управление" (см. далее).

Кроме нулевой точки, устройства-функции могут иметь дополнительные точки, реализующие полезный обмен данными. Низкоскоростные устройства могут иметь до двух дополнительных точек, полноскоростные - до 16 точек ввода и 16 точек вывода (протокольное ограничение). Точки не могут быть использованы до их конфигурирования (установления согласованного с ними канала).

Каналом (Pipe) в USB называется модель передачи данных между хост-контроллером и конечной точкой (Endpoint) устройства. Имеются два типа каналов: потоки (Stream) и сообщения (Message). Поток доставляет данные от одного конца канала к другому, он всегда однонаправленный. Один и тот же номер конечной точки может использоваться для двух поточных каналов - ввода и вывода. Поток может реализовывать следующие типы обмена: сплошной, изохронный и прерывания. Доставка всегда идет в порядке "первым вошел - первым вышел" (FIFO); с точки зрения USB, данные потока неструктурированы. Сообщения имеют формат, определенный спецификацией USB. Хост посылает запрос к конечной точке, после которого передается (принимается) пакет сообщения, за которым следует пакет с информацией состояния конечной точки. Последующее сообщение нормально не может быть послано до обработки предыдущего, но при отработке ошибок возможен сброс необслуженных сообщений. Двухсторонний обмен сообщениями адресуется к одной и той же конечной точке. Для доставки сообщений используется только обмен типа "управление".

С каналами связаны характеристики, соответствующие конечной точке (полоса пропускания, тип сервиса, размер буфера и т. п.). Каналы организуются при конфигурировании устройств USB. Для каждого включенного устройства существует канал сообщений (Control Pipe 0), по которому передается информация конфигурирования, управления и состояния.

5. Типы передачи данных

USB поддерживает как однонаправленные, так и двунаправленные режимы связи. Передача данных производится между ПО хоста и конечной точкой устройства. Устройство может иметь несколько конечных точек, связь с каждой из них (канал) устанавливается независимо.

Архитектура USB допускает четыре базовых типа передачи данных:

Управляющие посылки (Control Transfers), используемые 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) 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. В случае обнаружения ошибки изохронные данные передаются без повтора - недействительные пакеты игнорируются. Пример - цифровая передача голоса. Пропускная способность определяется требованиями к качеству передачи, а задержка доставки может быть критичной, например, при реализации телеконференций.

Полоса пропускания шины делится между всеми установленными каналами. Выделенная полоса закрепляется за каналом, и если установление нового канала требует такой полосы, которая не вписывается в уже существующее распределение, запрос на выделение канала отвергается.

Архитектура USВ предусматривает внутреннюю буферизацию всех устройств, причем чем большей полосы пропускания требует устройство, тем больше должен быть его буфер. USB должна обеспечивать обмен с такой скоростью, чтобы задержка данных в устройстве, вызванная буферизацией, не превышала нескольких миллисекунд.

Изохронные передачи классифицируются по способу синхронизации конечных точек - источников или получателей  
данных - с системой: различают асинхронный, синхронный  
и адаптивный классы устройств, каждому из которых соответствует свой тип канала USB.

6. Протокол

Все обмены (транзакции) по USB состоят из трех пакетов. Каждая транзакция планируется и начинается по инициативе контроллера, который посылает пакет-аркер (Token Packet). Он описывает тип и направление передачи, адрес ус-тройства USB и номер конечной точки. В каждой транзакции возможен обмен только между адресуемым устройством (его конечной точкой) и хостом. Адресуемое маркером устройство распознает свой адрес и готовится к обмену. Источник данных (определенный маркером) передает пакет данных (или уведомление об отсутствии данных, предназначенных для передачи). После успешного приема пакета приемник данных посылает пакет подтверждения (Handshake Packet).

Планирование транзакций обеспечивает управление поточными каналами. На аппаратном уровне использование отказа от транзакции (NAck) при недопустимой интенсивности передачи предохраняет буферы от переполнения сверху и снизу. Маркеры отвергнутых транзакций повторно передаются в свободное для шины время. Управление потоками позволяет гибко планировать обслуживание одновременных разнородных потоков данных.

Устойчивость к ошибкам обеспечивают следующие свойства USB:

Высокое качество сигналов, достигаемое благодаря дифференциальным приемникам/передатчикам и экранированным кабелям.

Защита полей управления и данных CRC-кодами.

Обнаружение подключения и отключения устройств и конфигурирование ресурсов на системном уровне.

Самовосстановление протокола с тайм-аутом при потере пакетов.

Управление потоком для обеспечения изохронности и управления аппаратными буферами.

Независимость функций от неудачных обменов с другими функциями.

Для обнаружения ошибок передачи каждый пакет имеет контрольные поля CRC-кодов, позволяющие обнаруживать все одиночные и двойные битовые ошибки. Аппаратные средства обнаруживают ошибки передачи, а контроллер автоматически производит трехкратную попытку передачи. Если повторы безуспешны, сообщение об ошибке передается клиентскому ПО.

7. Форматы пакетов

Байты передаются по шине последовательно, начиная с младшего бита. Все посылки организованы в пакеты. Каждый пакет начинается с поля синхронизации Sync, которое представляется последовательностью состояний KJKJKJKK (коди-  
рованную по NRZI), следующую после состояния Idle. Последние два бита (КК) являются маркером начала пакета SOP, используемым для идентификации первого бита идентификатора пакета PID. Идентификатор пакета является 4-битным полем PID[3:0], идентифицирующим тип пакета (табл. 7.2), за которым в качестве контрольных следуют те же 4 бита, но инвертированные.

В пакетах-маркерах IN, SETUP и OUT следующими являются адресные поля: 7-битный адрес функции и 4-битный адрес конечной точки. Они позволяют адресовать до 127 функций USB (нулевой адрес используется для конфигурирования) и по 16 конечных точек в каждой функции.

В пакете SOF имеется 11-битное поле номера кадра (Frame Number Field), последовательно (циклически) увеличиваемое для очередного кадра.

Поле данных может иметь размер от 0 до 1023 целых байт. Размер поля зависит от типа передачи и согласуется при установлении канала.

Поле СКС-кола присутствует во всех маркерах и пакетах данных, оно защищает все поля пакета, исключая PID. CRC для маркеров (5 бит) и данных (11 бит) подсчитываются по разным формулам.

Каждая транзакция инициируется хост-контроллером посылкой маркера и завершается пакетом квитирования. Последовательность пакетов в транзакциях иллюстрирует.

Хост-контроллер организует обмены с устройствами согласно своему плану распределения ресурсов. Контроллер циклически (с периодом 1 мс) формирует кадры (Frames), в которые укладываются все запланированные транзакции. Каждый кадр начинается с посылки маркера SOF (Start Of Frame), который является синхронизирующим сигналом для всех устройств, включая хабы. В конце каждого кадра выделяется интервал времени EOF (End Of Frame), на время которого хабы запрещают передачу по направлению к контроллеру. Каждый кадр имеет свой номер. Хост-контроллер оперирует 32-битным счетчиком, но в маркере SOF передает только младшие 11 бит. Номер кадра увеличивается (циклически) во время EOF. Хост планирует загрузку кадров так, чтобы в них всегда находилось место для транзакций управления и прерывания. Свободное время кадров может заполняться сплошными передачами (Bulk Transfers).

Для изохронной передачи важна синхронизация устройств и контроллера. Есть три варианта:

синхронизация внутреннего генератора устройства с маркерами SOF;

подстройка частоты кадров под частоту устройства;

согласование скорости передачи (приема) устройства с частотой кадров.

Подстройка частоты кадров контроллера возможна, естественно, под частоту внутренней синхронизации только одного устройства. Подстройка осуществляется через механизм обратной связи, который позволяет изменять период кадра в пределах ±1 битового интервала.

8. Системное конфигурирование

USB поддерживает динамическое подключение и отключение устройств. Нумерация устройств шины является постоянным процессом, отслеживающим изменения физической топологии.

Все устройства подключаются через порты хабов. Хабы определяют подключение и отключение устройств к своим портам и сообщают состояние портов при запросе от контроллера. Хост разрешает работу порта и адресуется к устройству через канал управления, используя нулевой адрес - USB Default Address. При начальном подключении или после сброса все устройства адресуются именно так.

Хост определяет, является новое подключенное устройство хабом или функцией, и назначает ему уникальный адрес USB. Хост создает канал управления (Control Pipe) с этим устройством, используя назначенный адрес и нулевой номер точки  
назначения.

Если новое устройство является хабом, хост определяет подключенные к нему устройства, назначает им адреса и устанавливает каналы. Если новое устройство является функцией, уведомление о подключении передается диспетчером  
USB заинтересованному ПО. Когда устройство отключается, хаб автоматически запрещает соответствующий порт и сообщает об отключении контроллеру, который удаляет сведения о данном устройстве из всех структур данных. Если отключается хаб, процесс удаления выполняется для всех подключенных к нему устройств. Если отключается функция, уведомление посылается заин-  
тересованному ПО. Нумерация устройств, подключенных к шине (Bus Enumeration), осуществляется динамически по мере их подключения (или включения их питания) без какого-либо вмешательства пользователя или клиентского ПО. Процедура  
нумерации выполняется следующим образом:

1. Хаб, к которому подключилось устройство, информирует хост о смене состояния своего порта ответом на опрос состояния. С этого момента устройство переходит в состояние Attached (подключено), а порт, к которому оно подключилось, в состояние Disabled.

2. Хост уточняет состояние порта.

3. Узнав порт, к которому подключилось новое устройство, хост дает команду сброса и разрешения порта.

4. Хаб формирует сигнал Reset для данного порта (10 мс) и переводит его в состояние Enabled. Подключенное устройство может потреблять от шины ток питания до 100 мА. Устройство переходит в состояние Powered (питание подано), все его регистры переводятся в исходное состояние, и оно отзывается на обращение по нулевому адресу.

5. Пока устройство не получит уникальный адрес, оно доступно по дежурному каналу, по которому хост-контроллер определяет максимально допустимый размер поля данных пакета.

6. Хост сообщает устройству его уникальный адрес, и оно переводится в состояние Addressed (адресовано).

7. Хост считывает конфигурацию устройства, включая заявленный потребляемый ток от шины. Считывание может затянуться на несколько кадров.

8. Исходя из полученной информации, хост конфигурирует все имеющиеся конечные точки данного устройства, которое переводится в состояние Configured (сконфигурировано). Теперь хаб позволяет устройству потреблять от шины полный ток, заявленный в конфигурации. Устройство готово.

Когда устройство отключается от шины, хаб уведомляет об этом хост и работа порта запрещается, а хост обновляет свою текущую топологическую информацию.

9. Устройства USB - функции и хабы

Возможности шины USB позволяют использовать ее для подключения разнообразных устройств. Не касаясь "полезных" свойств ПУ, остановимся на их интерфейсной части, связанной с шиной USB. Все устройства должны поддержи-  
вать набор общих операций, перечисленных ниже. Динамическое подключение и отключение. Эти события отслеживаются хабом, который сообщает о них хост-контроллеру и выполняет сброс подключенного устройства. Устройство после сигнала сброса должно отзываться на нулевой адрес, при этом оно не сконфигурировано и не приостановлено. После назначения адреса, за которое отвечает хост-контроллер, устройство должно отзываться только на свой уни-  
кальный адрес.

Конфигурирование устройств, выполняемое хостом, является необходимым для их использования. Для конфигурирования обычно используется информация, считанная из самого устройства. Устройство может иметь множество интерфейсов, каждому из которых соответствует собственная конечная точка, представляющая хосту функцию устройства. Интерфейс в конфигурации может иметь альтернативные наборы характеристик; смена наборов поддерживается протоколом. Для поддержки адаптивных драйверов дескрипторы устройств и интерфейсов имеют поля класса, подкласса и протокола.

Передача данных возможна посредством одного из четырех типов передач (см. выше). Для конечных точек, допускающих разные типы передач, после конфигурирования доступен только один из них.

Управление энергопотреблением является весьма развитой функцией USB. Для устройств, питающихся от шины, мощность ограничена. Любое устройство при подключении не должно потреблять от шины ток, превышающий 100 мА.  
Рабочий ток (не более 500 мА) заявляется в конфигурации, и если хаб не сможет обеспечить устройству заявленный ток, оно не конфигурируется и, следовательно, не может быть использовано.

Устройство USB должно поддерживать приостановку (Suspended Mode), в котором его потребляемый ток не превышает 500 мкА. Устройство должно автоматически приостанавливаться при прекращении активности шины.

Возможность удаленного пробуждения (Remote Wakeup) позволяет приостановленному устройству подать сигнал хосткомпьютеру, который тоже может находиться в приостановленном состоянии. Возможность удаленного пробуждения описывается в конфигурации устройства. При конфигурировании эта функция может быть запрещена.

Хаб в USB выполняет коммутацию сигналов и выдачу питающего напряжения, а также отслеживает состояние подключенных к нему устройств, уведомляя хост об изменениях. Хаб состоит из двух частей - контроллера (Hub Controller) и повторителя (Hub Repeater). Повторитель представляет собой управляемый ключ, соединяющий выходной порт со входным. Он имеет средства поддержки сброса и приостановки передачи сигналов. Контроллер содержит регистры для взаимодействия с хостом. Доступ к регистрам осуществляется по специфическим командам обращения к хабу. Команды позволяют конфигурировать хаб, управлять нисходящими портами и наблюдать их состояние.

Нисходящие (Downstream) порты хабов могут находиться в следующих состояниях:

Powered ((питание отключено) - на порт не подается питание (возможно только для хабов, коммутирующих питание). Выходные буферы переводятся в высокоимпедансное состояние, входные сигналы игнорируются.

Disconnected (отсоединен) - порт не передает сигналы ни в одном направлении, но способен обнаружить подключение устройства (по отсутствию состояния SEO в течение 2,5 мкс). Тогда порт переходит в состояние Disabled, а по уровням входных сигналов {DiffO или Diff1 в состоянии Idle) он определяет скорость подключенного устройства.

Disabled (запрещен) - порт передает только сигнал сброса (по команде от контроллера), сигналы от порта (кроме обнаружения отключения) не воспринимаются. По обнаружении отключения (2,5 мкс состояния SEO) порт переходит в состояние Disconnect, а если отключение обнаружено "спящим" хабом, контроллеру будет послан сигнал Resume.

Enabled (разрешен) - порт передает сигналы в обоих направлениях. По команде контроллера или по обнаружении ошибки кадра порт переходит в состояние Disabled, а по обнаружении отключения - в состояние Disconnect.

Suspended (приостановлен) - порт передает сигнал перевода в состояние останова ("спящий" режим). Если хаб находится в активном состоянии, сигналы через порт не пропускаются ни в одном направлении. Однако "спящий" хаб воспринимает сигналы смены состояния незапрещенных портов, подавая "пробуждающие" сигналы от активизировавшегося устройства даже через цепочку "спящих" хабов. Состояние каждого порта идентифицируется контроллером  
хаба с помощью отдельных регистров. Имеется общий регистр, биты которого отражают факт изменения состояния каждого порта (фиксируемый во время EOF). Это позволяет хост-контроллеру быстро узнать состояние хаба, а в случае  
обнаружения изменений специальными транзакциями уточнить состояние.

10. Хост-контроллер

Хост-компьютер общается с устройствами через контроллер.  
Хост имеет следующие обязанности:

обнаружение подключения и отсоединения устройств USB;

манипулирование потоком управления между устройствами и хостом;

управление потоками данных;

сбор статистики;

обеспечение энергосбережения подключенными ПУ.

Системное ПО контроллера управляет взаимодействием между устройствами и их ПО, функционирующим на хост-компьютере, для согласования:

нумерации и конфигурации устройств;

изохронных передач данных;

асинхронных передач данных;

управления энергопотреблением;

информации об управлении устройствами и шиной.

По возможности ПО USB использует существующее системное

ПО хост-компьютера - например, Advanced Power Management для управления энергопотреблением.