

LPT, COM, 1394, USB, USB-C

LPT

IEEE 1284, LPT (англ. Line Print Terminal; также параллельный порт, порт принтера) — международный стандарт параллельного интерфейса для подключения периферийных устройств персонального компьютера.

В основном используется для подключения к компьютеру принтера, сканера и других внешних устройств (часто использовался для подключения внешних устройств хранения данных), однако может применяться и для других целей (организация связи между двумя компьютерами, подключение каких-либо механизмов телесигнализации и телеуправления).

В основе данного стандарта лежит интерфейс Centronics и его расширенные версии (ECP, EPP).

Название LPT образовано от наименования стандартного устройства принтера LPT1 (Line Printer Terminal или Line PrinTer) в операционных системах семейства MS-DOS.

Параллельный порт Centronics — порт, используемый с 1981 года в персональных компьютерах фирмы IBM для подключения печатающих устройств, разработан фирмой Centronics Data Computer Corporation; уже давно стал стандартом де-факто, хотя в действительности официально на данный момент он не стандартизирован.

Изначально этот порт был разработан только для симплексной (однаправленной) передачи данных, так как предполагалось, что порт Centronics должен использоваться только для работы с принтером. Впоследствии разными фирмами были разработаны дуплексные расширения интерфейса (byte mode, EPP, ECP). Затем был принят международный стандарт IEEE 1284, описывающий как базовый интерфейс Centronics, так и все его расширения.

Разъемы.

Порт на стороне управляющего устройства (компьютера) имеет 25-контактный 2-рядный разъем DB-25-female («мама») (IEEE 1284-A). Не путать с аналогичным male-разъемом («папа»), который устанавливался на старых компьютерах и представляет собой 25-пиновый COM-порт. На периферийных устройствах обычно используется 36-контактный микроразъем ленточного типа Centronics (IEEE 1284-B), поэтому кабели для подключения периферийных устройств к компьютеру по параллельному порту обычно выполняются с 25-контактным разъемом DB-25-male на одной стороне и 36-контактным IEEE 1284-B на другой (AB-кабель). Изредка применяется AC-кабель с 36-контактным разъемом MiniCentronics (IEEE 1284-C).

Существуют также CC-кабели с разъемами MiniCentronics на обоих концах, предназначенные для подключения приборов в стандарте IEEE 1284-II, который применяется редко.

Длина соединительного кабеля не должна превышать 3 метров. Конструкция кабеля: витые пары в общем экране, либо витые пары в индивидуальных экранах. Изредка используются ленточные кабели.

Для подключения сканера и некоторых других устройств используется кабель, у которого вместо разъема (IEEE 1284-B) установлен разъем DB-25-male. Обычно сканер оснащается вторым интерфейсом с разъемом DB-25-female (IEEE 1284-A) для

подключения принтера (поскольку обычно компьютер оснащается только одним интерфейсом IEEE 1284). Схемотехника сканера построена таким образом, чтобы при работе с принтером сканер прозрачно передавал данные с одного интерфейса на другой.



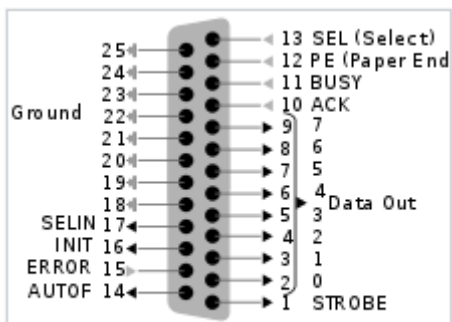
Кабельный 36-контактный разъем Centronics для подключения внешнего устройства (IEEE 1284-B)



25-контактный разъем DB-25, используемый как LPT-порт на персональных компьютерах (IEEE 1284-A)

Физический интерфейс

Базовый интерфейс Centronics является однонаправленным параллельным интерфейсом, содержит характерные для такого интерфейса сигнальные линии (8 для передачи данных, строб, линии состояния устройства).



Интерфейс разъёма

Данные передаются в одну сторону: от компьютера к внешнему устройству. Но полностью однонаправленным его назвать нельзя. Так, 5 обратных линий используются для контроля над состоянием устройства. Centronics позволяет подключать одно устройство, поэтому для совместного очередного использования нескольких устройств требуется дополнительно применять селектор.

Скорость передачи данных может варьироваться и достигать 1,2 Мбит/с.

Контакты DB-25 IEEE 1284-A	Контакты Centronics IEEE 1284-B	Обозначение	Примечание	Функция
1	1	Strobe	Маркер цикла передачи (выход)	Управление
2	2	Data 0	Сигнал 0 (выход)	Данные
3	3	Data 1	Сигнал 1 (выход)	Данные
4	4	Data 2	Сигнал 2 (выход)	Данные
5	5	Data 3	Сигнал 3 (выход)	Данные
6	6	Data 4	Сигнал 4 (выход)	Данные
7	7	Data 5	Сигнал 5 (выход)	Данные
8	8	Data 6	Сигнал 6 (выход)	Данные
9	9	Data 7	Сигнал 7 (выход)	Данные
10	10	Acknowledge	Готовность принять (вход)	Состояние
11	11	Busy	Занят (вход)	Состояние
12	12	Paper End	Нет бумаги (вход)	Состояние
13	13	Select	Выбор (вход)	Состояние
14	14	Auto Feed	Автоподача (выход)	Управление
15	32	Error	Ошибка (вход)	Состояние
16	31	Init	Инициализация (выход)	Управление

17	36	Select In	Управление печатью (выход)	Управление
18-25	16-17, 19-30	GND	Общий	Земля

Большинство фирменных расширений первоначального интерфейса впоследствии было стандартизировано индустрией, каковой процесс завершился принятием серии стандартов IEEE-1284.

Однако полного соответствия между этим стандартом и предшествующими ему фирменными расширениями нет. Наиболее известными фирменными расширениями являются расширения фирмы Hewlett-Packard. Это Bitronics, обеспечивающий двустороннюю передачу информации (применяется прежде всего для снятия расширенной информации о состоянии принтера) и протокол мультимплексирования шины от HP, предназначенный для того, чтобы подключать к одному LPT-порту множество устройств по схеме «цепочка». На основе этого протокола были разработаны стандарты 1284.3-2000 и 1284.4-2000, но полной совместимости достигнуто не было.

Стандарт позволяет использовать интерфейс в нескольких режимах:

SPP (Standard Parallel Port) — однонаправленный порт, полностью совместим с интерфейсом Centronics.

Nibble Mode — позволяет организовать двунаправленный обмен данными в режиме SPP путём использования управляющих линий (4 бит) для передачи данных от периферийного устройства к контроллеру. Исторически, это был единственный способ использовать Centronics для двустороннего обмена данными.

Byte Mode — редко используемый режим двустороннего обмена данными. Использовался в некоторых старых контроллерах до принятия стандарта IEEE 1284.

EPP (Enhanced Parallel Port) — разработан компаниями Intel, Xircom и Zenith Data Systems — двунаправленный порт, со скоростью передачи данных до 2Мбайт/сек.(1991)

ECP (Extended Capabilities Port) — разработан компаниями Hewlett-Packard и Microsoft — в дополнение появились такие возможности, как наличие аппаратного сжатия данных, наличие буфера и возможность работы в режиме DMA, также поддерживает симметричный двунаправленный обмен данными со скоростью до 2,5 Мбайт/с.

Стандарт IEEE-1284

Стал результатом длительной борьбы за обеспечение совместимости. Он включает в себя формальное описание всех вышеперечисленных режимов работы (до его принятия не было никакого формального документа, позволяющего при соблюдении его рассчитывать на корректную работу устройства во всех возможных конфигурациях).

Стандарт включает в себя следующие документы:

IEEE 1284—1994: Standard Signaling Method for a Bi-directional Parallel Peripheral Interface for Personal Computers — двунаправленная передача данных;

IEEE 1284.1-1997: Transport Independent Printer/System Interface- a protocol for returning printer configuration and status — стандартизованный метод получения информации о состоянии принтера;

IEEE 1284.2: Standard for Test, Measurement and Conformance to IEEE 1284 (not approved) — тестирование на соответствие стандарту (не был принят);

IEEE 1284.3-2000: Interface and Protocol Extensions to IEEE 1284-Compliant Peripherals and Host Adapters- a protocol to allow sharing of the parallel port by multiple peripherals (daisy chaining) — протокол и схема к одному порту многих устройств, объединённых в «цепочку» (позволяет выбрать нужное устройство и работать с ним);

IEEE 1284.4-2000: Data Delivery and Logical Channels for IEEE 1284 Interfaces — allows a device to carry on multiple, concurrent exchanges of data — протокол одновременной передачи информации многим устройствам (позволяет работать одновременно с несколькими устройствами в цепочке).

В настоящее время стандарт IEEE-1284 не развивается. Окончательная стандартизация параллельного порта совпала с началом внедрения интерфейса USB, который позволяет подключать также МФУ и обеспечивает более высокую скорость печати и надёжную работу принтера. Также альтернативой параллельному интерфейсу является сетевой интерфейс Ethernet.

RS-232

RS-232 (англ. Recommended Standard 232) — стандарт физического уровня для асинхронного интерфейса (UART). Широко известен как последовательный порт персональных компьютеров. Исторически имел широкое распространение в телекоммуникационном оборудовании. В настоящее время используется для подключения к компьютерам широкого спектра оборудования, нетребовательного к скорости обмена, особенно при значительном удалении его от компьютера и отклонении условий применения от стандартных. В компьютерах, занятых офисными и развлекательными приложениями, практически вытеснен интерфейсом USB.

RS-232 обеспечивает передачу данных и некоторых специальных сигналов между терминалом (англ. Data Terminal Equipment, DTE) и коммуникационным устройством (англ. Data Communications Equipment, DCE) на расстояние до 15 метров на максимальной скорости (115200 бод). Т.к. этот интерфейс известен не только простотой программирования, но и неприхотливостью, в реальных условиях это расстояние увеличивают во много раз с примерно пропорциональным снижением скорости.

Протокол интерфейса предполагает два режима передачи данных синхронный и асинхронный, а также два метода управления обменом данными: аппаратный и программный. Каждый режим может работать с любым методом управления. В протоколе также предполагается вариант управления передачей данных по специальным сигналам устанавливаемых хостом (DTR - сигнал готовности состояния, DSR - сигнал готовности передачи данных).

Разъем имеет контакты:

DTR (Data Terminal Ready — готовность к приёму данных) — выход на компьютере, вход на модеме. Означает готовность компьютера к работе с модемом. Сброс этой линии вызывает почти полную перезагрузку модема в первоначальное состояние, в том числе бросание трубки (некоторые управляющие регистры выживают после такого сброса). В

UNIX это происходит в случае, если все приложения закрыли файлы на драйвере последовательного порта. Мышь использует этот провод для получения питания.

DSR (Data Set Ready — готовность к передаче данных) — вход на компьютере, выход на модеме. Означает готовность модема. Если эта линия находится в нуле — то в ряде ОС становится невозможно открыть порт как файл.

RxD (Receive Data — приём данных) — вход на компьютере, выход на модеме. Поток данных, входящий в компьютер.

TxD (Transmit Data — передача данных) — выход на компьютере, вход на модеме. Поток данных, исходящих из компьютера.

CTS (Clear to Send — готовность передачи) — вход на компьютере, выход на модеме. Компьютер обязан приостановить передачу данных, пока этот провод не будет выставлен в единицу. Используется в аппаратном протоколе управления потоком для предотвращения переполнения в модеме.

RTS (Request to Send — запрос на передачу) — выход на компьютере, вход на модеме. Модем обязан приостановить передачу данных, пока этот провод не будет выставлен в единицу. Используется в аппаратном протоколе управления потоком для предотвращения переполнения в оборудовании и драйвере.

DCD (Carrier Detect — наличие несущей) — вход на компьютере, выход на модеме. Вводится модемом в единицу после установления соединения с модемом с той стороны, сбрасывается в ноль при разрыве связи. Аппаратура компьютера может издавать прерывание при наступлении такого события.

RI (Ring Indicator — сигнал вызова) — вход на компьютере, выход на модеме. Вводится модемом в единицу после обнаружения вызывного сигнала телефонного звонка. Аппаратура компьютера может издавать прерывание при наступлении такого события.

SG (Signal Ground — сигнальная земля) — общий сигнальный провод порта, не является общей землёй, как правило, изолирован от корпуса ЭВМ или модема.

В нуль-модемном кабеле используются две перекрещенные пары: TXD/RXD и RTS/CTS.

Исходно в IBM PC и IBM PC/XT аппаратура порта была построена на микросхеме UART 8250 фирмы National Semiconductor, затем микросхема была заменена на 16450, программно совместимой с предыдущими, но позволявшей использовать скорости вплоть до 115200 бит в секунду, затем появилась микросхема 16550, содержащая двунаправленный FIFO буфер данных для снижения нагрузки на контроллер прерываний. В настоящее время включена в SuperIO микросхему на материнской плате вместе с рядом иных устройств.

Для передачи данных по интерфейсу RS-232 используется код NRZ, который не является самосинхронизирующимся, поэтому для синхронизации используется стартовый и стоповый бит, позволяющие выделить битовую последовательность и синхронизировать приёмник с передатчиком.

Изначально создавался для подключения телефонных модемов к компьютерам. В связи с такой специализацией имеет рудименты в виде, например, отдельной линии RING («звонок»). Постепенно телефонные модемы перешли на другие интерфейсы (USB), но

разъём для RS-232 имелся на всех персональных компьютерах и многие изготовители оборудования использовали его для подключения своего оборудования. Например, компьютерные мышки.

В настоящее время чаще всего используется в промышленном и узкоспециальном оборудовании, встраиваемых устройствах. На портативных компьютерах (ноутбуках, нетбуках, КПК и т. п.) широкого применения RS-232 уже не ставят, однако материнские платы стационарных персональных компьютеров обычно ещё содержат RS-232 — либо в виде разъёма на задней панели, либо в виде колодки для подключения шлейфа на плате. Также возможно использование переходников-преобразователей.

Широко используется для взаимодействия микроконтроллеров различных архитектур, имеющими в своем составе интерфейс **UART**, с другими цифровыми устройствами и периферией.

UART

Универсальный асинхронный приёмопередатчик (УАПП, англ. Universal Asynchronous Receiver-Transmitter, UART) — узел вычислительных устройств, предназначенный для организации связи с другими цифровыми устройствами. Преобразует передаваемые данные в последовательный вид так, чтобы было возможно передать их по одной физической цифровой линии другому аналогичному устройству. Метод преобразования хорошо стандартизован и широко применяется в компьютерной технике (особенно во встраиваемых устройствах и системах на кристалле (SoC)).

Представляет собой логическую схему, с одной стороны подключённую к шине вычислительного устройства, а с другой имеющую два или более выводов для внешнего соединения.

UART может представлять собой отдельную микросхему (например, Intel 18251, 18250) или являться частью большой интегральной схемы (например, микроконтроллера). Используется для передачи данных через последовательный порт компьютера, часто встраивается в микроконтроллеры.

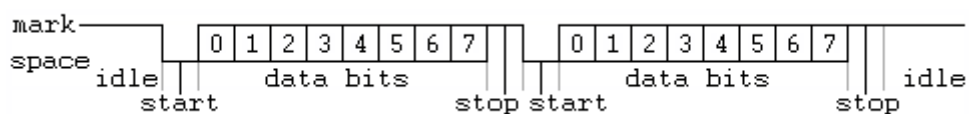
Передача данных в UART осуществляется по одному биту в равные промежутки времени. Этот временной промежуток определяется заданной скоростью UART и для конкретного соединения указывается в бодах (что в данном случае соответствует битам в секунду). Существует общепринятый ряд стандартных скоростей: 300; 600; 1200; 2400; 4800; 9600; 19200; 38400; 57600; 115200; 230400; 460800; 921600 бод. Скорость (S , бод) и длительность бита (T , секунд) связаны соотношением $T=1/S$. Скорость в бодах иногда называют сленговым словом бодреит или битреит.

Помимо информационных бит, UART автоматически вставляет в поток синхронизирующие метки, так называемые стартовый и стоповый биты. При приёме эти лишние биты удаляются из потока. Обычно стартовый и стоповый биты обрамляют один байт информации (8 бит), при этом младший информационный бит передаётся первым, сразу после стартового. Встречаются реализации UART, передающие по 5, 6, 7, или 9 информационных бит. Обрамленные стартом и стопом биты являются минимальной посылкой. Некоторые реализации UART используют два стоповых бита при передаче для уменьшения вероятности рассинхронизации приёмника и передатчика при плотном трафике. Приёмник игнорирует второй стоповый бит, воспринимая его как короткую паузу на линии.

Принято соглашение, что пассивным (в отсутствие потока данных) состоянием входа и выхода UART является логическая 1. Стартовый бит всегда логический 0, поэтому приёмник UART ждёт перепада из 1 в 0 и отсчитывает от него временной промежуток в половину длительности бита (середины передачи стартового бита). Если в этот момент на входе всё ещё 0, то запускается процесс приёма минимальной посылки. Для этого приёмник отсчитывает 9 битовых длительностей подряд (для 8-битных данных) и в каждый момент фиксирует состояние входа. Первые 8 значений являются принятыми данными, последнее значение проверочное (стоп-бит). Значение стоп-бита всегда 1, если реально принятое значение иное, UART фиксирует ошибку.

Для формирования временных интервалов передающий и приёмный UART имеют источник точного времени (тактирования). Точность этого источника должна быть такой, чтобы сумма погрешностей (приёмника и передатчика) установки временного интервала от начала стартового импульса до середины стопового импульса не превышала половины (а лучше хотя бы четверти) битового интервала. Для 8-битной посылки $0,5/9,5 = 5\%$ (в реальности не более 3%). Поскольку эта сумма ошибок приёмника и передатчика плюс возможные искажения сигнала в линии, то рекомендуемый допуск на точность тактирования UART — не более 1,5%.

Поскольку синхронизирующие биты занимают часть битового потока, то результирующая пропускная способность UART меньше скорости соединения. Например, для 8-битных посылок формата 8-N-1 синхронизирующие биты занимают 20% потока, что при физической скорости линии 115 200 бод означает полезную скорость передачи данных 92 160 бит/с или 11 520 байт/с.



Контроль чётности

Многие реализации UART имеют возможность автоматически контролировать целостность данных методом контроля битовой чётности. Когда эта функция включена, последний бит данных в минимальной посылке («бит чётности») контролируется логикой UART и содержит информацию о чётности количества единичных бит в этой минимальной посылке. Различают контроль на четность (англ. Even parity), когда сумма количества единичных бит в посылке является четным числом, и контроль на нечетность (англ. Odd parity), когда эта сумма нечетна. При приеме такой посылки UART может автоматически контролировать бит четности и выставлять соответствующие признаки правильного или ошибочного приема.

данные	количество единичных бит	бит четности	
		even	odd
0000000	0	0	1
1010001	3	1	0
1101001	4	0	1
1111111	7	1	0

Короткая запись параметров

Был выработан и прижился короткий способ записи параметров UART, таких, как количество бит данных, наличие и тип бита четности, количество стоп-бит. Выглядит как запись вида цифра-буква-цифра, где:

Первая цифра обозначает количество бит данных, например, 8.

Буква обозначает наличие и тип бита четности. Встречаются N (No parity) — без бита четности; E (Even parity) — с битом проверки на четность, O (Odd parity) — с битом проверки на нечетность;

Последняя цифра обозначает длительность стоп-бита. Встречаются значения 1, 1.5 и 2 для длительности стоп-бита в 1, 1.5 и 2 битовых интервала соответственно.

Например, запись 8-N-1 обозначает, что UART настроен на 8 бит данных без бита четности и один стоповый бит. Для полноты параметров эту запись снабжают указанием скорости UART, например, 9600/8-N-1.

Физический уровень

Логическая схема UART имеет входы-выходы с логическими уровнями, соответствующими полупроводниковой технологии схемы: КМОП, ТТЛ и т. д. Такой физический уровень может быть использован в пределах одного устройства, однако непригоден для коммутируемых длинных соединений по причине низкой защищённости от электрического разрушения и помех. Для таких случаев были разработаны

специальные физические уровни, такие, как токовая петля, RS-232, RS-485, LIN и тому подобные.

Изначально UART предназначался для связи двух устройств, по принципу «точка-точка». Впоследствии были созданы физические уровни, которые позволяют связывать более двух UART по принципу «один говорит — несколько слушают». Такие физические уровни называют сетевыми. Существуют реализации типа общая шина (когда все приемопередатчики подключены к одному проводу) и кольцо (когда приемники и передатчики соединяют попарно в замкнутое кольцо). Первый вариант проще и встречается гораздо чаще. Второй вариант сложнее, но надежнее и быстрее: гарантируется работоспособность всех узлов (передающий узел услышит эхо своего сообщения, только если оно успешно ретранслировано всеми узлами); любой узел может начинать передачу в любой момент, не заботясь о риске коллизии. Наиболее известные сетевые физические уровни — RS-485 и LIN.

IEEE 1394

IEEE 1394 (FireWire, i-Link) — последовательная высокоскоростная шина, предназначенная для обмена цифровой информацией между компьютером и другими электронными устройствами.

Различные компании продвигали стандарт под своими торговыми марками:

Apple — FireWire

Sony — i.LINK

Yamaha — mLAN

TI — Lynx

Creative — SB1394

В 1986 году членами Комитета по стандартам микрокомпьютеров (Microcomputer Standards Committee) принято решение объединить существовавшие в то время различные варианты последовательной шины (Serial Bus).

В 1992 году разработкой интерфейса занялась Apple.

В 1995 году принят стандарт IEEE 1394 (сама технология была разработана намного раньше, до появления Windows 95, что показывает большой потенциал данного института).

Около 1998 года содружество компаний, в том числе Microsoft, развивали идею обязательности 1394 для любого компьютера и использования 1394 внутри корпуса, а не только вне него. Существовали даже карты контроллеров с одним разъёмом, направленным внутрь корпуса. Также существовала идея Device Bay, то есть отсека для устройства со встроенным в отсек разъёмом 1394 и поддержкой горячей замены.

Такие тенденции прослеживаются в материалах Microsoft той поры, предназначенных для разработчиков компьютеров. Можно сделать вывод, что 1394 предлагали как замену ATA, то есть на роль, ныне выполняемую SATA.

Но этим идеям не суждено было воплотиться, и одной из главных причин такого исхода была лицензионная политика компании Apple, требующей выплат за каждый чип контроллера. Модели системных плат и ноутбуков, представленные на рынке с начала

2010-х годов, как правило, уже не поддерживают интерфейс FireWire. Исключения представлены в узком топовом IT-сегменте

Преимущества

Горячее подключение — возможность переконфигурировать шину без выключения компьютера.

Различная скорость передачи данных — 100, 200 и 400 Мбит/с в стандарте IEEE 1394/1394a, дополнительно 800 и 1600 Мбит/с в стандарте IEEE 1394b и 3200 Мбит/с в спецификации S3200.

Гибкая топология — равноправие устройств, допускающее различные конфигурации (возможность «общения» устройств без компьютера).

Высокая скорость — возможность обработки мультимедиа-сигнала в реальном времени
Поддержка изохронного трафика.

Поддержка атомарных операций — сравнение/обмен, атомарное увеличение (операции семейства LOCK — compare/swap, fetch/add и т. д.).

Открытая архитектура — отсутствие необходимости использования специального программного обеспечения.

Наличие питания прямо на шине (маломощные устройства могут обходиться без собственных блоков питания). До 1,5 А и напряжение от 8 до 40 вольт.

Подключение до 63 устройств.

Шина IEEE 1394 может использоваться для:

- создания компьютерной сети;
- подключения аудио- и видеоустройств;
- подключения принтеров и сканеров;
- подключения жёстких дисков, массивов RAID.

Основные сведения.

Кабель представляет собой 2 витые пары — А и В, распаянные как А к В, а на другой стороне кабеля — как В к А. Также возможен необязательный проводник питания.

Устройство может иметь до 4 портов (разъёмов). В одной топологии может быть до 64 устройств. Максимальная длина пути в топологии — 16. Топология древовидная, замкнутые петли не допускаются.

При присоединении и отсоединении устройства происходит сброс шины, после которого устройства самостоятельно выбирают из себя главное, пытаясь взвалить это «главенство» на соседа. После определения главного устройства становится ясна логическая направленность каждого отрезка кабеля — к главному или же от главного. После этого возможна раздача номеров устройствам. После раздачи номеров возможно исполнение обращений к устройствам.

Во время раздачи номеров по шине идёт трафик пакетов, каждый из которых содержит в себе количество портов на устройстве, ориентацию каждого порта — не подключён / к главному / от главного, а также максимальную скорость каждой связи (2 порта и отрезок кабеля). Контроллер 1394 принимает эти пакеты, после чего стек драйверов строит карту топологии (связей между устройствами) и скоростей (наихудшая скорость на пути от контроллера до устройства).

Операции шины делятся на асинхронные и изохронные.

Асинхронные операции — это запись/чтение 32-битного слова, блока слов, а также атомарные операции. Асинхронные операции используют 24-битные адреса в пределах каждого устройства и 16-битные номера устройств (поддержка межшинных мостов). Некоторые адреса зарезервированы под главнейшие управляющие регистры устройств. Асинхронные операции поддерживают двухфазное исполнение — запрос, промежуточный ответ, потом позже окончательный ответ.

Изохронные операции — это передача пакетов данных в ритме (строго привязанном к частоте 8 КГц), задаваемом ведущим устройством шины путём инициации транзакций «запись в регистр текущего времени». Вместо адресов в изохронном трафике используются номера каналов от 0 до 31. Подтверждений не предусмотрено, изохронные операции есть одностороннее вещание.

Изохронные операции требует выделения изохронных ресурсов — номера канала и полосы пропускания. Это делается атомарной асинхронной транзакцией на некие стандартные адреса одного из устройств шины, избранного как «менеджер изохронных ресурсов».

Помимо кабельной реализации шины, в стандарте описана и наплатная (реализации неизвестны).

Использование

Сеть поверх 1394 и FireNet

Существуют стандарты RFC 2734 — IP поверх 1394 и RFC 3146 — IPv6 поверх 1394. Поддерживались в ОС Windows XP и Windows Server 2003. Поддержка со стороны Microsoft прекращена в ОС Windows Vista, однако существует реализация сетевого стека FireNet в альтернативных драйверах от компании Unibrain (версия 6.00 вышла в ноябре 2012 года)

Поддерживается во многих ОС семейства UNIX (обычно требуется пересборка ядра с этой поддержкой).

Стандарт не подразумевает эмуляцию Ethernet над 1394 и использует совершенно иной протокол ARP. Несмотря на это, эмуляция Ethernet над 1394 была включена в ОС FreeBSD и является специфичной для данной ОС.

Внешние дисковые устройства

Существует стандарт SBP-2 — SCSI поверх 1394. В основном используется для подключения внешних корпусов с жёсткими дисками к компьютерам — корпус содержит чип моста 1394—ATA. При этом скорость передачи данных превышает скорость USB 2.0 как интерфейса к устройствам хранения данных, равную примерно 43 МБ/с, однако гораздо ниже таковой для USB 3.0.

Поддерживается в ОС семейства Windows с Windows 98 и по сей день. Также поддерживается в популярных ОС семейства UNIX.

MiniDV-видеокамеры

Исторически первое использование шины. Используется и по сей день как средство захвата фильмов с MiniDV в файлы. Возможен и захват с камеры на камеру.

Видеосигнал, идущий по 1394, идёт практически в том же формате, что и хранится на видеоленте. Это упрощает камеру, снижая требования к ней по наличию памяти.

В ОС Windows подключённая по 1394 камера является устройством DirectShow. Захват видео с такого устройства возможен в самых разнообразных приложениях — Adobe Premiere, Ulead Media Studio Pro, Windows Movie Maker. Существует также огромное количество простейших утилит, способных выполнять только этот захват. Возможно также и использование тестового инструмента Filter Graph Editor из свободно распространяемого DirectShow SDK.

Использование 1394 с miniDV положило конец проприетарным платам видеозахвата.

USB

USB (ю-эс-би, англ. Universal Serial Bus — «универсальная последовательная шина») — последовательный интерфейс для подключения периферийных устройств к вычислительной технике. Получил широчайшее распространение и фактически стал основным интерфейсом подключения периферии к бытовой цифровой технике.

Интерфейс позволяет не только обмениваться данными, но и обеспечивать электропитание периферийного устройства. Сетевая архитектура позволяет подключать большое количество периферии даже к устройству с одним разъёмом USB.

Разработка спецификаций USB производится в рамках международной некоммерческой организации USB Implementers Forum (USB-IF), объединяющей разработчиков и производителей оборудования с шиной USB. В процессе развития выработано несколько версий спецификаций. Тем не менее разработчикам удалось сохранить высокую степень совместимости оборудования разных поколений. Спецификация интерфейса охватывает беспрецедентно широкий круг вопросов подключения и взаимодействия периферийных устройств с вычислительной системой:

- унификацию разъёмов и кабелей
- нормирование энергопотребления
- протоколы обмена данными
- унификацию функциональности и драйверов устройств

Первые спецификации для USB 1.0 были представлены в 1994—1995 годах. Разработка USB поддерживалась фирмами Intel, Microsoft, Philips, US Robotics. USB стал «общим знаменателем» под тремя не связанными друг с другом стремлениями разных компаний:

Расширение функциональности компьютера. На тот момент для подключения внешних периферийных устройств к персональному компьютеру использовалось несколько «традиционных» (англ. legacy) интерфейсов (PS/2, последовательный порт, параллельный порт, порт для подключения джойстика, SCSI), и с появлением новых внешних устройств разрабатывали и новый разъём. Предполагалось, что USB заменит их все и заодно подхлестнёт разработку нетрадиционных устройств.

Подключить к компьютеру мобильный телефон. В то время мобильные сети переходили на цифровую передачу голоса, и ни один из имеющихся интерфейсов не годился для передачи с телефона на компьютер как речи, так и данных.

Простота для пользователя. Старые интерфейсы (например, последовательный (COM) и параллельный (LPT) порты) были крайне просты для разработчика, но не давали настоящего «подключи и работай». Требовались новые механизмы взаимодействия компьютера с низко- и среднескоростными внешними устройствами — возможно, более сложные для конструкторов, но надёжные, дружелюбные и пригодные к «горячему» подключению.

Поддержка USB вышла в виде патча к Windows 95 OEM Service Release 2, в дальнейшем она вошла в стандартную поставку Windows 98. В первые годы устройств было мало, поэтому шину в шутку называли «Useless serial bus» — «бесполезная последовательная шина». Впрочем, производители быстро осознали пользу USB, и уже к 2000 году большинство принтеров и сканеров работали с новым интерфейсом.

Hewlett-Packard, Intel, Lucent (ныне Alcatel-Lucent), Microsoft, NEC и Philips совместно выступили с инициативой по разработке более скоростной версии USB. Спецификация USB 2.0 была опубликована в апреле 2000 года, и в конце 2001 года эта версия была стандартизирована USB Implementers Forum. USB 2.0 является обратно совместимой со всеми предыдущими версиями USB.

Следует отметить, что в начале 2000-х годов корпорация Apple отдавала приоритет шине FireWire, в разработке которой она принимала активное участие. Ранние модели iPod были оснащены только интерфейсом FireWire, а USB отсутствовал. Впоследствии компания отказалась от FireWire в пользу USB, оставив в некоторых моделях FireWire только для подзарядки. Однако часть выпускавшихся клавиатур и мышей, начиная со второй половины 90-х годов, имела интерфейс USB.

С середины первого десятилетия 2000-х годов BIOS'ы компьютеров массового сегмента начали поддерживать USB (поддержка USB в корпоративном сегменте началась с середины 90-х). Это позволило загружаться с флеш-дисков, например, для переустановки ОС; пропала надобность в PS/2-клавиатуре. Современные настольные материнские платы поддерживают 10 и более USB-портов. В подавляющем большинстве современных ноутбуков COM- и LPT-портов нет, всё чаще появляются настольные компьютеры без COM-портов.

Пока происходило распространение USB-портов второй версии, производители внешних жёстких дисков уже «упёрлись» в ограничение USB 2.0 — и по току, и по скорости. Потребовался новый стандарт, который и вышел в 2008 году. Уложиться в старые 4 провода не удалось, добавили 5 новых. Первые материнские платы с поддержкой USB 3.0 вышли в 2010 году. К 2013 году USB 3.0 стал массовым. Также имеются платы расширения, добавляющие поддержку USB 3.0 в старых компьютерах.

Уже в первые годы обнаружился серьёзный конструктивный недостаток разъёма USB-A: он асимметричен, но не показывает, какой стороной подключать. К тому же на сцену вышли смартфоны, а на них не стоит делать много разъёмов: так, в телефонах Samsung между пятью штырями MicroUSB добавили шесть новых. Обе эти проблемы решил симметричный разъём USB-C, появившийся в 2013 году. Одни провода продублированы на обеих сторонах, о назначении других контроллеры «договариваются» при подключении. Заодно USB-C имеет несколько резервных проводов, чтобы передавать через него, например, HDMI.

Основные сведения

Кабель USB (до 2.0 включительно) состоит из 4 медных проводников — 2 проводника питания и 2 проводника данных в витой паре, и заземлённой оплётки (экрана).

Кабели USB ориентированы, то есть имеют физически разные наконечники «к устройству» (Тип B) и «к хосту» (Тип A). Возможна реализация USB-устройства без кабеля, со встроенным в корпус наконечником «к хосту». Возможно и неразъёмное встраивание кабеля в устройство, как в мышь (стандарт запрещает это для устройств full и high speed, но производители его нарушают). Существуют (хотя и запрещены стандартом) и пассивные USB-удлинители, имеющие разъёмы «от хоста» и «к хосту».

С помощью кабелей формируется интерфейс между USB-устройствами и USB-хостом. В качестве хоста выступает программно-управляемый USB-контроллер, который обеспечивает функциональность всего интерфейса. Контроллер, как правило, интегрирован в процессор или микросхему южного моста, хотя может быть исполнен и в отдельном корпусе. Соединение контроллера с внешними устройствами происходит через USB-концентратор (другие названия — хаб, разветвитель). В силу того, что USB-шина имеет древовидную топологию, концентратор самого верхнего уровня называется корневым (root hub). Он встроен в USB-контроллер и является его неотъемлемой частью.

Для подключения внешних устройств к USB-концентратору в нём предусмотрены порты, заканчивающиеся разъёмами. К разъёмам с помощью кабеля могут подключаться USB-устройства либо USB-хабы нижних уровней. Такие хабы — активные электронные устройства (пассивных не бывает), обслуживающие несколько собственных USB-портов. С помощью USB-концентраторов допускается до пяти уровней каскадирования, не считая корневого. Сам USB-интерфейс не позволяет соединять между собой два компьютера (хост-устройства), это возможно лишь при использовании специальной электроники, имеющей два USB-входа и специализированный мост, например, эмулирующей два соединённых Ethernet-адаптера — по одному для каждой из сторон — либо использующие специализированное ПО для обмена файлами.

Устройства могут быть запитаны от шины, но могут и требовать внешний источник питания. По умолчанию устройствам гарантируется ток до 100 мА, а после согласования с хост-контроллером — до 500 мА. Поддерживается и дежурный режим для устройств и разветвителей по команде с шины со снятием основного питания при сохранении дежурного питания и включением по команде с шины.

USB поддерживает «горячее» подключение и отключение устройств. Это достигнуто увеличенной длиной заземляющего контакта разъёма по отношению к сигнальным. При подключении разъёма USB первыми замыкаются заземляющие контакты, потенциалы корпусов двух устройств становятся равны и дальнейшее соединение сигнальных проводников не приводит к перенапряжениям, даже если устройства питаются от разных фаз силовой трёхфазной сети.

На логическом уровне устройство USB поддерживает транзакции приёма и передачи данных. Каждый пакет каждой транзакции содержит в себе номер оконечной точки (endpoint) на устройстве. При подключении устройства драйверы в ядре ОС читают с устройства список оконечных точек и создают управляющие структуры данных для общения с каждой оконечной точкой устройства. Совокупность оконечной точки и структур данных в ядре ОС называется каналом (pipe).

Оконечные точки, а значит, и каналы, относятся к одному из 4 классов — поточный (bulk), управляющий (control), изохронный (isoch) и прерывание (interrupt). Низкоскоростные устройства, такие, как мышь, не могут иметь изохронных и поточных каналов.

Управляющий канал предназначен для обмена с устройством короткими пакетами «вопрос-ответ». Любое устройство имеет управляющий канал 0, который позволяет программному обеспечению ОС прочитать краткую информацию об устройстве, в том числе коды производителя и модели, используемые для выбора драйвера, и список других оконечных точек.

Канал прерывания позволяет доставлять короткие пакеты и в том, и в другом направлении, без получения на них ответа/подтверждения, но с гарантией времени доставки — пакет будет доставлен не позже, чем через N миллисекунд. Например, используется в устройствах ввода (клавиатуры/мыши/джойстики).

Изохронный канал позволяет доставлять пакеты без гарантии доставки и без ответов/подтверждений, но с гарантированной скоростью доставки в N пакетов на один период шины (1 кГц у low и full speed, 8 кГц у high speed). Используется для передачи аудио- и видеоинформации.

Поточный канал даёт гарантию доставки каждого пакета, поддерживает автоматическую приостановку передачи данных по нежеланию устройства (переполнение или опустошение буфера), но не даёт гарантий скорости и задержки доставки. Используется, например, в принтерах и сканерах.

Время шины делится на периоды, в начале периода контроллер передаёт всей шине пакет «начало периода». Далее в течение периода передаются пакеты прерываний, потом изохронные в требуемом количестве, в оставшееся время в периоде передаются управляющие пакеты и в последнюю очередь — поточные.

Активной стороной шины всегда является контроллер, передача пакета данных от устройства к контроллеру реализована как короткий вопрос контроллера и длинный, содержащий данные, ответ устройства. Расписание движения пакетов для каждого периода шины создаётся совместными усилиями аппаратуры контроллера и ПО драйвера, для этого многие контроллеры используют крайне сложный DMA со сложной DMA-программой, формируемой драйвером.

Размер пакета для оконечной точки есть вшитая в таблицу оконечных точек устройства константа, изменению не подлежит. Он выбирается разработчиком устройства из числа тех, что поддерживаются стандартом USB.

Спецификация	Скорость	Стандарт USB
Low-Speed	до 1,5 Мбит/с	USB 1.0
Full-Speed	до 12 Мбит/с	USB 1.1
High-speed	до 480 Мбит/с	USB 2.0

SuperSpeed	до 5 Гбит/с	USB 3.0 / USB 3.1 Gen 1
SuperSpeed 10Gbps	до 10 Гбит/с	USB 3.1 Gen 2

Некоторые реализованные модификации.

On-The-Go Supplement 1.3: извещение выпущено в декабре 2006 года. USB On-The-Go делает возможным связь двух USB-устройств друг с другом без отдельного USB-хоста. На практике одно из устройств играет роль хоста для другого.

В USB одно устройство всегда хост, другое — периферия. Смартфонам, цифровым фотоаппаратам и прочим мобильным устройствам приходится быть то хостом, то периферией: при подключении к компьютеру фотоаппарат — периферия, а при подключении к фотопринтеру — хост.

USB OTG (аббр. от On-The-Go) сделал возможными такие устройства двойного назначения: они сами определяют, кем им быть. OTG-устройства можно подключать к компьютеру, и к таким устройствам через тот же порт можно подключать USB-периферию: обычно флеш-накопители, цифровые фотоаппараты, клавиатуры, мыши и другие устройства, не требующие дополнительных драйверов.

Ранг устройства определяется кабелем: со стороны хоста замыкаются контакты 4 (ID) и 5 (Ground) в штекере. Со стороны периферии ID никуда не подключается.

USB 3.0

Окончательная спецификация USB 3.0 появилась в 2008 году. Созданием USB 3.0 занимались компании Intel, Microsoft, Hewlett-Packard, Texas Instruments, NEC и NXP Semiconductors.

Спецификация USB 3.0 повышает максимальную скорость передачи информации до 5 Гбит/с, что на порядок больше 480 Мбит/с, которые может обеспечить USB 2.0. Таким образом, скорость передачи возрастает с 60 МБ/с (30 МБ/с эффективных) до 600 МБ/с.

Версия 3.0 отличается не только более высокой скоростью передачи информации, но и увеличенной силой тока с 500 мА до 900 мА. Таким образом, от одного порта можно запитывать большее количество устройств, а также отпадает необходимость использования внешнего питания для некоторых устройств. В спецификации USB 3.0 разъёмы и кабели обновлённого стандарта физически и функционально совместимы с USB 2.0, причём для однозначной идентификации разъёмы USB 3.0 принято изготавливать из пластика синего цвета (у некоторых производителей — красного цвета). Кабель USB 2.0 содержит в себе четыре линии — пару для приёма/передачи данных, плюс и ноль питания, разъём «А» имеет 4 контакта. Для передачи высокоскоростных SuperSpeed сигналов в USB 3.0 добавлено ещё четыре линии связи (две витые пары) и один контакт сигнальной земли (GND_DRAIN), в результате чего кабель стал гораздо толще. Новые контакты в разъёмах USB 3.0 расположены отдельно от старых в другом контактном ряду.

Хост-контроллер USB 3.0 (xHCI) обеспечивает аппаратную поддержку потоков для команд, статусов, входящих и исходящих данных, что позволяет более полно использовать пропускную способность USB-шины. Потоки были добавлены к протоколу USB 3.0 SuperSpeed для поддержки UASP.

Linux поддерживает USB 3.0, начиная с версии ядра 2.6.31.[10] В Windows 8 и 10 интерфейс USB 3.0 поддерживается без установки дополнительных драйверов.

USB 3.1

31 июля 2013 года USB 3.0 Promoter Group объявила о принятии спецификации следующего интерфейса, USB 3.1, скорость передачи которого может достигать 10 Гбит/с. Компактный разъем USB Type-C, используемый с данной версией, является симметричным, позволяя вставлять кабель любой стороной, как это ранее сделала Apple в разъемах Lightning.

Пользователи получили возможность передавать данные со скоростью до 10 Гбит/с. Новый стандарт также позволяет в отдельных случаях передавать мощность до 100 Вт.

После выхода стандарта USB 3.1 организация USB-IF объявила, что разъемы USB 3.0 со скоростью до 5 Гбит/с (SuperSpeed) теперь будут классифицироваться как USB 3.1 Gen 1, а новые разъемы USB 3.1 со скоростью до 10 Гбит/с (SuperSpeed USB 10Gbps) — как USB 3.1 Gen 2.

В USB 3.1 входит два стандарта:

SuperSpeed USB (USB 3.1 Gen 1) со скоростью до 5 Гбит/с, такой же, как и у USB 3.0; SuperSpeed USB 10Gbps (USB 3.1 Gen 2) со скоростью до 10 Гбит/с, удвоенная USB 3.0. В USB 3.1 Gen 2, помимо увеличения скорости до 10 Гбит/с, были снижены издержки кодирования до 3 % переходом на схему кодирования 128b/132b.

Стандарт USB 3.1 обратно совместим с USB 3.0 и USB 2.0.

USB 3.2

Организация USB 3.0 Promoter Group выпустила предварительный вариант спецификаций нового стандарта USB 3.2. Обновление принесёт вдвое большую скорость передачи данных по сравнению с USB 3.1 за счёт использования двух линий на 5 Гбит/с или 10 Гбит/с, выдавая в результате 10 или 20 Гбит/с. Современные кабели USB Type-C, имеющиеся в наличии, уже поддерживают такой «двухлинейный» режим, так что покупать новые кабели не придётся.

Устройства с поддержкой USB 3.2 ожидаются не ранее 2018 г.

Inter-Chip USB



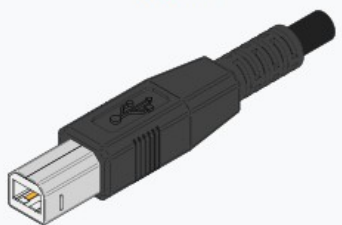
Inter-Chip USB (IC-USB) и High Speed Inter-Chip USB (HSIC) — упрощённые версии USB 2.0 для некоммутируемого соединения микросхем в одном устройстве. Упрощение достигается за счёт замены физического уровня USB с асинхронного на синхронный, отказа от возможности смены скорости и определения подключения, отказа от электрической защиты драйверов и уменьшения их мощности. Логическая часть USB неизменна (в том числе логика состояний шины). IC-USB определяет соединение Full Speed (12 МБ/с) устройств; HSIC определяет соединение High Speed (480 МБ/с) устройств.

Первая версия стандарта IC-USB была принята в 2006 году. Первая версия стандарта HSIC была принята в 2007 году[17]. HSIC использует две цифровых линии с логическими уровнями LVCMOS (1,2 вольта): STROBE и DATA. Максимальная длина проводников 10 см. Синхронный интерфейс обеспечивает пропускную способность 480 Мбит/с при тактовой частоте 240 МГц. Драйвер физического уровня HSIC потребляет на 50 % меньше энергии и занимает на 75 % меньше места на кристалле, чем традиционный драйвер USB 2.0.

В 2012 году была принята первая версия спецификаций Inter-Chip USB для USB 3.0

Кабели и разъёмы USB 1.x и 2.0

Спецификация 1.0 регламентировала два типа разъёмов: А — на стороне контроллера или концентратора USB и В — на стороне периферийного устройства. Впоследствии были разработаны миниатюрные разъёмы для применения USB в переносных и мобильных устройствах

	Обычный	Mini	Micro
Тип А	4×12 мм 	3×7 мм 	2×7 мм 
Тип В	7×8 мм 	3×7 мм 	2×7 мм 

Сигналы USB (версии ≤2.x) передаются по двум проводам экранированного четырёхпроводного кабеля.

Распиновка Тип-А и -В

Номер контакта	Обозначение	Цвет провода		Описание
1	V _{BUS}	Красный, или	Оранжевый	+5 V
2	D-	Белый, или	Золотой	Данные-
3	D+	Зелёный		Данные+
4	GND	Чёрный, или	Синий	Земля

Распиновка Mini/Micro-A и -В

Номер контакта	Обозначение	Цвет провода	Описание
1	V _{BUS}	Красный	+5 V
2	D-	Белый	Данные-
3	D+	Зелёный	Данные+
4	ID	No wire	On-The-Go ID определяет конец кабеля: <ul style="list-style-type: none"> • А (хост): подключён к земле • В (устройство): не подключён
5	GND	Чёрный	Земля

Здесь GND — цепь «корпуса» для питания периферийных устройств, а VBus — +5 вольт, также для цепей питания. Данные передаются дифференциально по проводам D- и D+ (diff0 и diff1 соответственно, в терминологии официальной документации). Состояния «0» и «1» определяются по разности потенциалов между линиями более 0,2 В и при условии, что на одной из линий (D- в случае diff0 и D+ при diff1) потенциал относительно GND выше 2,8 В. Дифференциальный способ передачи является основным, но не единственным (например, при инициализации устройство сообщает хосту о режиме, поддерживаемом устройством (англ. Full-Speed или англ. Low-Speed), подтягиванием одной из линий данных к V_BUS через резистор 1,5 кОм (D- для режима Low-Speed и D+ для режимов Full-Speed и High-Speed).

Для соблюдения достаточного уровня сигнала в кабеле и недопускания его затухания требуется коррелировать длину кабеля с сечением проводников. Принята практика указания толщины сечения провода в AWG, например «28 AWG/1P.....».

Примерное соответствие: маркировка кабеля (указание толщины провода в AWG) и соответствующая ей длина кабеля:

AWG	Длина, не больше (см)
28	81
26	131
24	208
22	333
20	500

Ограничения длины кабеля также связаны с задержкой сигнала в линии. В спецификациях USB 2.0 оговорена величина задержки: она должна быть менее 5,2 наносекунды на метр для кабеля длиной 5 м. Максимально допустимая задержка сигнала в линии — 1,5 микросекунды для низкоскоростного режима. Таким образом, для обеспечения режима Hi Speed линия должна гарантировать задержку менее 26 наносекунд, а Low Speed — 1,5 микросекунды.

Кабели и разъёмы USB 3.0 и их совместимость с USB 2.0

Все разъёмы USB, имеющие возможность входить в соединение друг с другом, рассчитаны на совместную работу. Также это достигается за счёт электрической совместимости всех контактов разъёма USB 2.0 с соответствующими контактами разъёма USB 3.0. При этом разъём USB 3.0 имеет дополнительные контакты, не имеющие соответствия в разъёме USB 2.0, и, следовательно, при соединении разъёмов разных

версий «лишние» контакты не будут задействованы, обеспечивая нормальную работу соединения версии 2.0.

Все гнёзда и штекеры между USB 3.0 Тип А и USB 2.0 Тип А рассчитаны на совместную работу.

Размер гнезда USB 3.0 Тип В несколько больше, чем это могло бы потребоваться для штекера USB 2.0 Тип В и более ранних. При этом предусмотрено подключение в эти гнёзда и такого типа штекеров. Соответственно, для подключения к компьютеру периферийного устройства с разъёмом USB 3.0 Тип В можно использовать кабели обоих типов, но для устройства с разъёмом USB 2.0 Тип В — только кабель USB 2.0.

Гнёзда eSATA, обозначенные как eSATA/USB Combo, то есть имеющие возможность подключения к ним штекера USB, имеют возможность подключения штекеров USB Тип А: USB 2.0 и USB 3.0, но в скоростном режиме USB 2.0.

Штекер eSATA ни в какую версию простого гнезда USB войти не может.

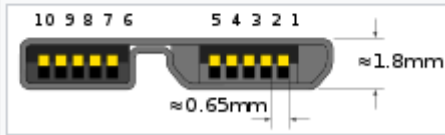
Штекер eSATA может подключаться к гнезду eSATA/USB Combo.

Изображения разъёмов USB 3.0

	Обычный	Mini	Micro
Тип А			
Тип В			
Тип С			

Расположение выводов соединителей USB 3.0 типа А

№ контакта	A	B	micro B
1	VBUS (VCC)	VBUS (VCC)	VBUS (VCC)
2	D-	D-	D-
3	D+	D+	D+
4	GND	GND	ID
5	StdA_SSTX-	StdA_SSTX-	GND
6	StdA_SSTX+	StdA_SSTX+	StdA_SSTX-
7	GND_DRAIN	GND_DRAIN	StdA_SSTX+
8	StdA_SSRX-	StdA_SSRX-	GND_DRAIN
9	StdA_SSRX+	StdA_SSRX+	StdA_SSRX-
10			StdA_SSRX+
Экран	Экран	Экран	Экран



Расположение контактов вилки
USB 3.0 Micro-B

USB Type-C

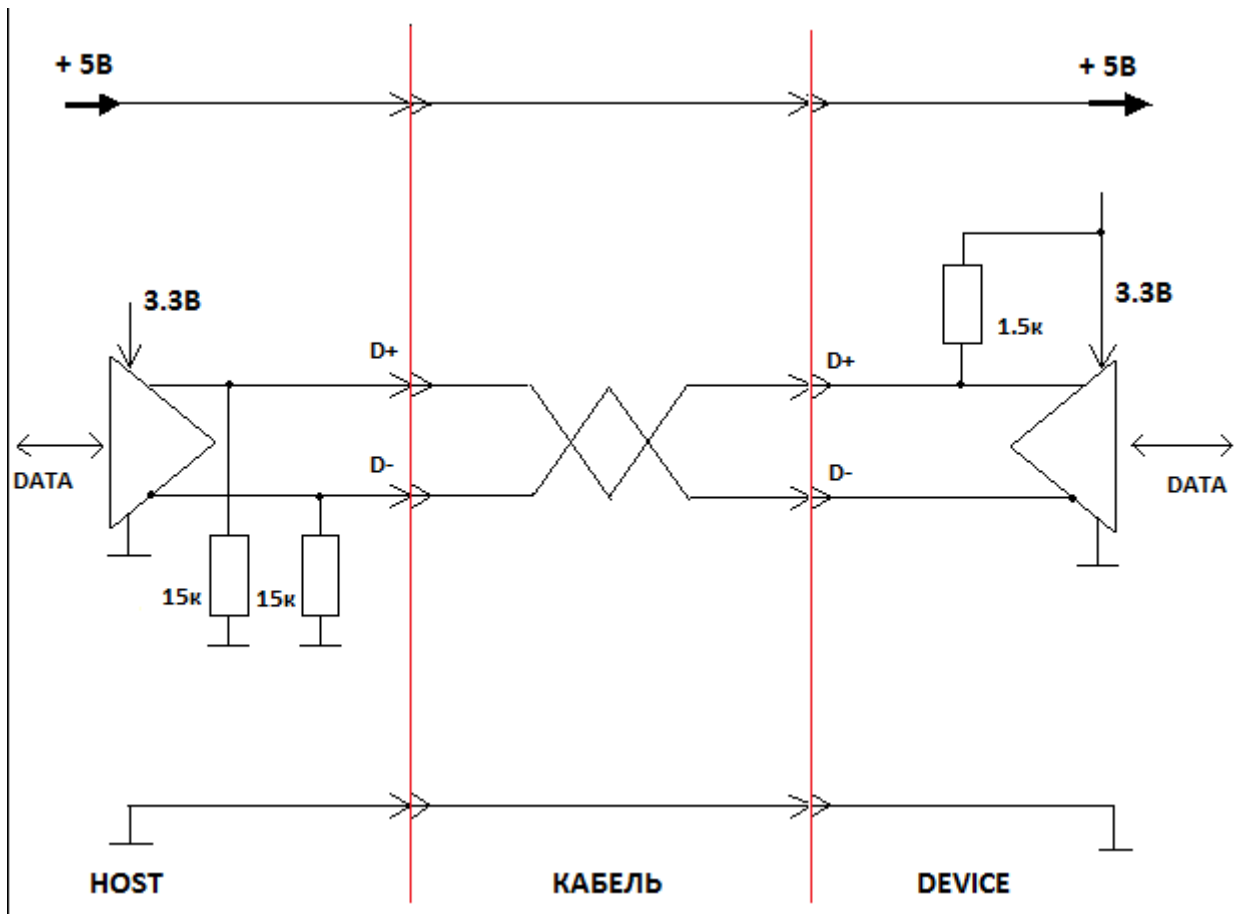
Назначение контактов в разъёме USB Type-C

Контакт	Название	О п и с а н и е	Контакт	Название	О п и с а н и е
A1	GND	Общая земля	B12	GND	Общая земля
A2	SSTXp1	SuperSpeed дифференциальная пара #1, TX, positive	B11	SSRXp1	SuperSpeed дифференциальная пара #1, RX, positive
A3	SSTXn1	SuperSpeed дифференциальная пара #1, TX, negative	B10	SSRXn1	SuperSpeed дифференциальная пара #1, RX, negative
A4	V _{BUS}	Линия питания	B9	V _{BUS}	Линия питания
A5	CC1	Конфигурационный канал	B8	SBU2	Полоса пропускания

					данных (SBU)
A6	Dp1	USB 2.0 дифференциальная пара, position 1, positive	B7	Dn2	USB 2.0 дифференциальная пара, position 2, negative
A7	Dn1	USB 2.0 дифференциальная пара, position 1, negative	B6	Dp2	USB 2.0 дифференциальная пара, position 2, positive
A8	SBU1	Полоса пропускания данных (SBU)	B5	CC2	Конфигурационный канал
A9	V _{BUS}	Bus power	B4	V _{BUS}	Bus power
A10	SSRXn2	SuperSpeed дифференциальная пара #2, RX, negative	B3	SSTXn2	SuperSpeed дифференциальная пара #2, TX, negative
A11	SSRXp2	SuperSpeed дифференциальная пара #2, RX, positive	B2	SSTXp2	SuperSpeed дифференциальная пара #2, TX, positive
A12	GND	Общая земля	B1	GND	Общая земля

Физический уровень

Упрощённая электрическая схема USB-соединения показана на рисунке. Когда к хосту никто не подключён, обе сигнальные линии D+ и D- подтянуты резисторами 15 кОм к минусу питания. При подключении устройства одна из линий подтягивается к +3,3 В через резистор 1,5 кОм. Устройства Low Speed подтягивают линию D-, а устройства Full Speed — D+. Таким образом хост определяет факт подключения и тип подключённого устройства. Устройства High Speed в момент подключения работают как Full Speed, переключаясь в высокоскоростной режим после обмена визитками.

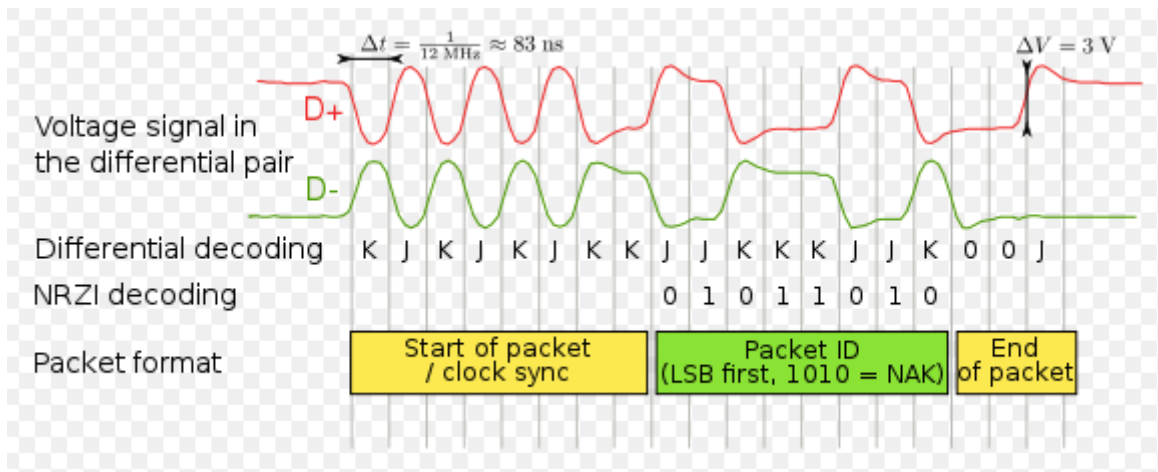


Состояние дифпары, определённое резисторами подтяжки, в спецификации именуется Idle. То же состояние при включённом драйвере обозначается буквой J. Противоположное ему состояние — буквой K. Замыкание обеих линий на минус именуется Single Ended 0, сокращенно SE0; замыкание на плюс — SE1.

Данные кодируются по методу NRZI (Non-return-to-zero inverted). По этому методу каждому нулевому бит входных данных соответствует изменение состояния дифпары (J→K или K→J), а при единице изменения нет. Чтобы исключить потерю синхронизации на длительных единичных последовательностях, применяют битстаффинг, то есть принудительно вставляют в поток данных ноль на каждые 6 единиц подряд.

Состояние шины SE0 дольше 10 мс трактуется устройством как сброс (Reset) и требует от устройства реинициализации USB-стека. Состояние Idle дольше 3 мс подряд трактуется устройством как останов шины (Suspend) и формально требует от устройства самоограничения в потреблении электроэнергии от шины USB. Выход из Suspend происходит либо по возобновлению активности хоста, либо устройство может при необходимости подать специальный сигнал Resume. Сигнал Resume состоит из состояния K на несколько миллисекунд, завершаемое последовательностью SE0, SE0, J, где каждое состояние длится один битовый интервал согласно скоростному режиму устройства.

Структура пакетов



Обмен происходит короткими пакетами. Каждый пакет начинается с последовательности Start of Packet, для Low и Full Speed это KJKJKJKK. Далее всегда идёт специальный идентификатор пакета PID (Packet IDentifier), указывающий на тип пакета. Всего имеется 16 разных типов пакетов, поэтому PID имеет размерность 4 бита. Однако для надёжности значение этого поля дублируется в инверсном виде, поэтому длина поля PID в пакете 8 бит. Заканчивается пакет последовательностью End of Packet: SE0,SE0,J. Минимальный межпакетный интервал ~0,1 мкс (для Full Speed).

В зависимости от типа пакета, между PID и EoP может содержаться ряд других полей с параметрами пакета и/или данными. Все эти поля (включая PID) передаются младшим битом вперёд (LSB first).

Типы пакетов USB представлены в таблице.

Тип	Значение ПИД (старшим битом вперед)	Передаваемый байт (младшим битом вперед)	Имя	Описание
<i>Reserved</i>	0000	0000 1111		
Token	1000	0001 1110	SPLIT	High-bandwidth (USB 2.0) split transaction
	0100	0010 1101	PING	Check if endpoint can accept data (USB 2.0)
Special	1100	0011 1100	PRE	Уведомление хабу что следующая транзакция будет осуществляться в режиме Low Speed
Handshake			ERR	Split transaction error (USB 2.0)
	0010	0100 1011	ACK	Подтверждение приёма пакета с данными
	1010	0101 1010	NACK	Неготовность обслужить предыдущий пакет прямо сейчас, пакет игнорируется.
	0110	0110 1001	NYET	Data not ready yet (USB 2.0)
	1110	0111 1000	STALL	Предыдущий пакет обратился к несуществующему или выключенному функционалу.
	Token	0001	1000 0111	OUT
	1001	1001 0110	IN	Хост уведомляет устройство, что ожидает от устройства пакет с данными.

	0101	1010 0101	SOF	Маркер временного фрейма.
	1101	1011 0100	SETUP	Хост уведомляет устройство, что следующий пакет будет содержать конфигурационные данные от хоста к устройству.
Data	0011	1100 0011	DATA0	Чётный пакет с данными. При обычном обмене пакеты с данными чередуются чётный-нечётный.
	1011	1101 0010	DATA1	Нечётный пакет с данными.
	0111	1110 0001	DATA2	Data packet for high-bandwidth isochronous transfer (USB 2.0)
	1111	1111 0000	MDATA	Data packet for high-bandwidth isochronous transfer (USB 2.0)

Пакеты типа IN, OUT, SETUP являются заголовками многопакетной транзакции с обменом данными. Они содержат поля адреса устройства для транзакции и номера Endpoint, с которым будет обмен. Целостность пакетов удостоверяет поле CRC5.

Пакеты типа DATA содержат поле данных и поле контроля целостности данных CRC16. Стандарт ограничивает максимальную разрешённую длину данных: 8 байт для неконфигурированных устройств, 64 байта для устройств Low Speed, 1023 байта для устройств Full Speed и 1024 байта для устройств High Speed. Устройство может установить свою максимальную длину данных, меньшую разрешённой. Хост обязан поддерживать максимальную разрешённую длину данных.

Пакеты типа ACK, NACK, STALL не содержат никаких дополнительных полей.

Адрес

USB является сетью, то есть к одному хосту может подключаться несколько устройств. Каждому устройству в процессе начального конфигурирования в момент подключения назначается уникальный адрес. Размерность адреса 7 бит, нулевое значение зарезервировано — соответственно, к одному хосту может подключаться до 127 устройств. Поле адреса содержат только те пакеты, что начинают транзакцию (IN, OUT, SETUP).

Endpoint

Помимо адресации физически подключённых устройств, USB предлагает логическую адресацию внутри устройства. Логическая адресация позволяет разделить потоки данных по разному функционалу внутри одного устройства. Например, клавиатура с тачпадом

может иметь один канал данных для нажатий клавиш, а другой — для данных тачпада. В стеке TCP/IP имеется прямая аналогия endpoint — порты.

Поле endpoint имеет размерность 4 бита, то есть возможны до 16 endpoint. Каждый endpoint может независимо работать как приёмный и как передающий, поэтому иногда их насчитывают 32. Поле endpoint содержат только те пакеты, что начинают транзакцию (IN, OUT, SETUP). В момент подключения в рамках начального конфигурирования устройство обязано передать хосту информацию о задействованных endpoint и их назначении. Эта информация должна согласовываться с соответствующими каналами данных программного драйвера устройства хоста. Обращение к незадействованному endpoint вызывает ответ STALL. Пакеты SETUP могут приходить только на нулевой endpoint.

Фреймы

Спецификация USB содержит понятия временных фреймов и микрофреймов. Для Low и Full Speed устройств каждую миллисекунду хост передаёт специальный пакет SOF (Start of Frame). Этот период называется фрейм. Для High Speed этот пакет передаётся каждые 125 мкс, такой период называется микрофрейм. Спецификация USB требует, чтобы поддерживалось такое планирование транзакций и пакетов, чтобы периодичность рассылки SOF не нарушалась.

Принципы обмена данными

Обмен данными происходит так называемыми транзакциями — неразрывными последовательностями из нескольких пакетов. Инициатором обмена всегда является хост. Он передаёт короткий пакет (token), уведомляющий о начале новой транзакции. В этом пакете-токене хост указывает направление транзакции (IN или OUT), адрес устройства и номер endpoint. Например, токен OUT означает, что немедленно за токеном последует пакет с данными от хоста к устройству (DATA0 или DATA1). Пакетов с данными может быть несколько в одной транзакции, если каждый из них имеет максимально допустимую для этого устройства длину данных. Окончание пересылки данных определяется по длине пакета, не равной максимальной. Как только приходит укороченный пакет, устройство немедленно передаёт ответный пакет-подтверждение (handshake), например ACK (всё успешно принято), NACK (не смог принять: например, переполнен входной буфер), STALL (данные адресованы отключённому endpoint). Все пакеты в транзакции передаются практически слитно, максимальная пауза между пакетами не должна превышать ~1 мкс (для Full Speed), иначе транзакция будет признана ошибочной.

Аналогично происходит передача данных от устройства к хосту. Хост инициирует передачу токеном IN. Если устройство не имеет готовых данных для передачи, то отвечает NACK и транзакция заканчивается. Если данные готовы, устройство начинает передавать пакеты DATA0/DATA1. Принцип окончания передачи аналогичен: неполная длина пакета с данными. Получив неполный пакет, хост отвечает устройству пакетом-подтверждением ACK.

Транзакция с токеном SETUP полностью аналогична транзакции OUT, отличия лишь в логике восприятия данных устройством: это параметры соединения, которые управляют работой USB стека устройства.

Control, Interrupt, Bulk, Isochronous

Спецификация USB предоставляет несколько методов обмена данными. Каждому включённому endpoint должен быть сопоставлен какой-либо из методов. Control, Interrupt и Bulk используют протокол обмена с подтверждениями, описанный чуть выше. Метод

bulk позволяет хосту свободно обмениваться данными с устройством по своему усмотрению. Метод control аналогичен bulk, но обменивается с устройством только специальными данными, управляющими работой USB-протокола в соответствии со спецификацией (в рамках транзакций типа SETUP). Поскольку периферийные устройства не могут инициировать обмен, то для передачи внезапно возникающих у устройства данных придумали метод interrupt, который позволяет опрашивать устройство с заданным периодом. Метод interrupt широко применяется для опроса клавиатур и мышек. Особняком стоит метод isochronous, позволяющий зарезервировать часть полосы пропускания USB-шины для таких данных, как аудио или видео. Isochronous не поддерживает контроля целостности передачи (пакеты ACK и NACK не передаются), а значит, не предусмотрены повторы в случае ошибок: неверно принятые данные пропадают.

Инициализация устройств

В момент подключения хост запрашивает у устройства ряд стандартизованных сведений (дескрипторов), на основании которых принимает решение, как с этим устройством работать. Дескрипторы содержат сведения о производителе и типе устройства, на основании которых хост подбирает программный драйвер. Таблицы дескрипторов и назначение полей подробно описаны в главе 9 спецификации USB.

После этого хост производит смену скорости (если устройство High Speed) и назначает устройству адрес.

Отладка и сертификация

Для отладки протоколов и контроля соответствия стандарту разработчиками устройств могут использоваться различные инструменты, позволяющие наблюдать процессы обмена на шине. Эти инструменты могут быть чисто программными, извлекающими события шины из драйверов USB компьютера. Однако такие инструменты не показывают аппаратно обрабатываемые либо ошибочные сигналы в шине. Для всеобъемлющего независимого контроля используются специализированные аппаратные сканеры и анализаторы протоколов. Использование аппаратного анализатора рекомендуется USB консорциумом для прохождения сертификации и при подготовке выпуска приборов в серийное производство.

Формально для получения права на размещение логотипов USB на продукции необходима её сертификация на соответствие стандарту. Организация USB-IF предлагает услуги по сертификации USB устройств, а также поддерживает список сторонних сертифицирующих лабораторий.

Plug and Play

Разработчики спецификации USB уделили внимание вопросу автоматического определения функциональности USB устройств, чтобы избавить пользователя от рутинных действий при подключении USB устройств. Для этого в стандарте предусмотрено два механизма:

Устройство сообщает хосту свои атрибуты, куда входит идентификатор разработчика устройства (VID) и идентификатор изделия (PID). На основании этих идентификаторов хост (компьютер) ищет методы работы с этим устройством (обычно это выражается в требовании установить драйвера, поставляемые разработчиком устройства).

Устройство сообщает хосту идентификатор стандартизованного класса устройств. В рамках концепции USB разработан ряд спецификаций стандартных классов устройств, в

рамках которых унифицирована работа с устройствами определённой функциональности. Например, широко известны устройства класса Human Interface Device, HID (это мышки, клавиатуры, игровые манипуляторы и т. п.) и устройства Mass Storage («флешки», дисководы). Для популярных классов устройств в компьютерах имеются готовые драйвера, поэтому подключение таких устройств происходит незаметно для пользователя.

Помимо стандартных решений USB, некоторые компании и энтузиасты предлагают иные решения. Например, в среде Windows популярны предустановленные драйвера WinUSB с доступным стороннему разработчику API.

Стандартные классы устройств

Назначение USB-устройств может определяться кодами классов, которые сообщаются USB-хосту для загрузки необходимых драйверов. Коды классов позволяют унифицировать работу с однотипными устройствами разных производителей. Устройство может поддерживать один или несколько классов, максимальное количество которых определяется количеством доступных endpoints.

Описание кодов классов

Код	Название	Примеры использования/примечание
00h	N/A	Не задано
01h	Audio	Звуковая карта, MIDI
02h	Communication Device (CDC)	Модем, сетевая карта, COM-порт
03h	Human Interface Device (HID)	Клавиатура, мышь, джойстик
05h	Physical Interface Device (PID)	Джойстик с поддержкой Force feedback
06h	Image	Веб-камера, сканер
07h	Printer	Принтер
08h	Mass Storage Device (MSD)	USB-накопитель, карта памяти, кардридер, цифровая фотокамера
09h	USB hub	USB-хаб
0Ah	CDC Data	Используется совместно с классом CDC
0Bh	Smart Card Reader (CCID)	Считыватель смарт-карт
0Dh	Content security	Биометрический сканер
0Eh	Video Device Class	Веб-камера
0Fh	Personal Healthcare	Индикатор пульса, медицинское оборудование
DCh	Diagnostic Device	Используется для проверки совместимости с USB

E0h	Wireless Controller	Bluetooth-адаптер
EFh	Miscellaneous	ActiveSync-устройства
FEh	Application-specific	IrDA-устройства, режим обновления прошивки (DFU)
FFh	Vendor-specific	На усмотрение производителя

Электропитание

В стандарте USB предусмотрена возможность снабжения подключённых устройств небольшой электрической мощностью. Первоначально стандарт USB 2.0 допускал максимальный потребляемый устройством ток до 0,5 А при напряжении 5 В. USB 3.0 увеличил максимальный ток до 0,9 А при том же напряжении. Эти стандарты позволяют хосту контролировать потребление подключённых к шине устройств. Для этого в момент подключения и инициализации устройство сообщает хосту свои энергетические потребности. Хост оценивает энергетические возможности этого сегмента сети и разрешает или запрещает устройству работу.

В попытках стандартизировать запросы энергоёмких устройств в 2007 году USB-IF принял спецификацию USB Battery Charging, которая в рамках кабельной инфраструктуры USB 2.0/3.0 позволяла увеличить потребляемый устройством ток до 5А. Позже была принята отдельная спецификация USB Power Delivery, которая предусматривает гораздо большую гибкость в управлении питанием.

USB Battery Charging

Первая попытка стандартизировать повышенное потребление гаджетов и источники питания с выходным разъёмом USB привела к появлению спецификации USB Battery Charging. Первая версия вышла в 2007 году. Актуальная версия USB BC 1.2 опубликована в 2010 году.

Спецификация разрешала существование специально обозначенных разъёмов USB-A с повышенной отдачей по току (до 1,5 А). Протокол начального конфигурирования USB дополнялся возможностью «договориться» о расширенном потреблении. Конечное устройство могло увеличить потребление только после «договорённости» с хостом.

Также разрешались разъёмы USB-A с не подключёнными линиями данных, например на зарядных устройствах. Такие зарядные устройства идентифицировались гаджетом по замкнутым между собой контактам D+ и D-. Таким устройствам разрешалось отдавать ток до 5 А.

Для малогабаритных потребителей электроэнергии спецификация рекомендовала разъём MicroUSB-B.

USB Power Delivery

В новом стандарте USB Power Delivery концепция электропитания была значительно переработана. Теперь разработчики как хоста, так и устройства получили возможность

гибко управлять питанием через шину USB. Решение о том, кто является источником, кто потребителем, о возможностях источника и кабеля принимаются в ходе диалога между устройствами по отдельному каналу связи. Предусматривалась возможность, что в процессе диалога устройство могло потребовать, а хост согласиться на повышение напряжения питания с целью передачи по существующей кабельной инфраструктуре больших мощностей. Повышенное напряжение выдавалось хостом на провод питания Vusb. Для совместимости со старыми устройствами хост возвращал напряжение к старым значениям 5 Вольт, как только обнаруживал отсоединение устройства.

USB PD Rev.1

В 2012 году представлена первая ревизия USB PD. Использовалась стандартная разъёмная и кабельная инфраструктура USB 2.0 и 3.0. Управление питанием осуществлялось путём диалога между потребителем и источником по независимому каналу связи, организованному по проводу питания стандартного кабеля USB (Vbus). Использовалась частотная модуляция с несущей 24 МГц.

USB PD Rev.2

Вторая ревизия стандарта вышла в 2014 году вместе со спецификацией USB 3.1 и привязана к новому разъёму Type C. В частности, теперь для выделенного канала связи между источником мощности и потребителем используется отдельный провод в кабеле (Configuration Channel). Также поддерживается определение типа кабеля и его возможностей по передаче мощности. Источник менее жёстко ограничен требованиями профилей первой ревизии и может гибче подходить к выбору максимального тока нагрузки в зависимости от располагаемой мощности.

Профили источника USB PD rev. 2

Мощность источника, Вт	Ток, А			
	+5 В	+9 В	+15 В	+20 В
0,5-15	0,1-3	Нет	Нет	Нет
15-27	3 (15 Вт)	1,7-3		
27-45		1,8-3		
45-60		3 (27 Вт)	3 (45 Вт)	2,25-3
60-100			3-5	

Критика

Разъёмы Mini- и особенно Micro-USB, имеют конструктивные просчёты.

Протокол требует от конечного устройства поддержания достаточно сложного алгоритмического стека как для непосредственно обмена по шине, так и для поддержания сопутствующих функций типа инициализации или ответов на служебные сообщения. Ввиду своей сложности и разнообразности устройства зачастую аппаратно выполняют лишь базовые уровни протокола, оставляя верхние на откуп программному коду. Это приводит к заметным непроизводительным затратам программной памяти и времени, а также содержит угрозу ошибок и попыток избыточно упростить программный код в ущерб соответствию стандарту.

Код производителя (VID) выдаётся разработчику устройства лишь после бюрократической процедуры и денежных затрат порядка 5000 USD. Дополнительно организация разработчиков стандарта USB-IF негативно относится к перепродаже владельцами кодов производителя кодов устройств (PID). Всё это ограничивает доступность шины для мелких производителей и независимых разработчиков. Свободно доступные коды для устройств, реализующих стандартные функциональности (напр., порт обмена, устройство памяти или аудиоустройство) создатели стандарта не предоставляют.

Список классов и подклассов устройств частями непоследователен, чрезмерно раздут, подклассы одного уровня зачастую неравноценны и содержат устаревшую функциональность. Как результат поддержка определённого стандартного класса зачастую требует избыточного кода, не нужного для непосредственного

функционирования, как со стороны устройства, так и хоста (компьютера). То же относится и к типам передаваемых пакетов, часть из которых имеет скорее историческое значение.

Несмотря на заявленную универсальность, многие устройства, даже принадлежащие стандартным классам, большей частью требуют программной поддержки и отдельных драйверов на хосте. Так, современная операционная система Windows при подключении внешнего COM-порта или GPS-навигатора (которые относятся к одному стандартному классу коммуникационных устройств) требует для каждого из устройств отдельного драйвера. Это налагает на производителей отдельные обязанности по созданию и, возможно, подписыванию драйверов и содержит риск неработоспособности устройства на операционной системе другой версии.