E-BOOK

An Engineer's Guide to USB Type-C®

A collection of technical content on USB Type-C and USB Power Delivery

Introduction

USB Type-C® (USB-C®) is an industry-standard connector that enables the transmission of both data and power on a single interface. USB Power Delivery (PD) is a standard using the USB-C connector to increase the capabilities and features of the USB-C interface. With USB Power Delivery, you can now transmit up to 240W of power and up to 80Gbps of data at the same time. You can also implement predefined alternate modes such as DisplayPort™ and Thunderbolt to support video and other advanced features. This e-book offers an introduction to USB Type-C and USB Power Delivery, examining various applications and their data and power requirements, and helping you understand the entire system view for implementing USB Type-C and USB Power Delivery.

Table of Contents

Basics of USB Type-C®

Abstract

Authors: Adam McGaffin, Nate Enos and Brian Gosselin

USB Type-C® (USB-C®) is an industry-standard connector that enables the transmission of both data and power on a single interface. USB Power Delivery (PD) is a standard using the USB-C connector to increase the capabilities and features of the USB-C interface. With USB PD, you can now transmit up to 240W of power and up to 80Gbps of data simultaneously. You can also implement predefined alternate modes such as DisplayPort™ and Thunderbolt to support video and other advanced features. In this chapter, we'll walk through the basics of USB-C to serve as a foundation for more advanced topics throughout this e-book.

USB-C data speeds and power levels

Table 1 lists the maximum transfer rate of each USB data transfer-related specification. The standard started with USB 1.0 and USB 1.1 supporting 1.5Mbps (low speed) and 12Mbps (full speed), respectively, but evolved to support 10Gbps (SuperSpeed+) with USB 3.1 Gen 2.

Table 1. USB specification and maximum voltage, current and power

Specification	Data rate name	Maximum transfer rate
USB 1.0 and USB 1.1	Low speed	1.5Mbps
	Full speed	12Mbps
USB 2.0	High speed	480Mbps
USB 3.0	SuperSpeed	5Gbps
USB 3.1	SuperSpeed+	10Gbps

Table 2 shows the evolution of USB power, starting with USB 2.0 through USB PD 3.0. The overall trend has been to increase the maximum power to address the growing needs of platforms and devices. Without USB PD, you can support up to 5V at 3A (15W) with just USB-C alone. However, with USB PD, you can support up to 48V at 5A (240W) within the USB-C ecosystem.

Table 2. USB-C and USB PD power levels

Specification	Maximum voltage	Maximum current	Maximum power
USB 2.0	5V	500mA	2.5W
USB 3.0 and USB 3.1	5V	900mA	4.5W
USB Battery Charging 1.2	5V	1.5A	7.5W
USB-C 1.2	5V	3A	15W
USB PD 3.0	20V	5A	100W
USB PD 3.1	48V	5A	240W

Data and power roles

There are three types of data flow in a USB connection:

- The **downstream-facing port (DFP)** sends data downstream; it is typically the port on a host or hub to which devices connect. A DFP will source VBUS power (the power path between host and device) and can also source VCONN power (to power electronically marked cables). An example of an application that may include a DFP is a docking station.
- The **upstream-facing port (UFP)**, which connects to a host or DFP of a hub, receives the data on a device or hub. These ports usually sink VBUS. An example of an application that may include a UFP is a display monitor.

- The **dual-role data (DRD)** port can operate as either a DFP (host) or a UFP (device). The port's power role at attach determines its initial role. A source port takes on the data role of a DFP, while the sink port takes on the data role of a UFP. Using USB PD data-role swap can dynamically change the port's data role, however. Example applications that may include DRD ports include laptops, tablets and smartphones.

There are three types of power flow in a USB connection:

- A **sink** is a port that when attached consumes power from VBUS. A sink is most often a device and could include USB peripherals such as USB-powered keyboards or consumer products such as headphones.
- A **source** is a port that when attached provides power over VBUS. Common sources are a host or hub DFP. An example of a source application is a USB-C wall charger.
- A **dual-role power (DRP)** port can operate as either a sink or source, and may alternate between these two states. When a DRP initially operates as a source, the port takes the data role of a DFP. Alternatively, when a DRP initially operates as a sink, the port takes the data role of a UFP. USB PD power-role swap can dynamically change the DRP's power role, however. For example, a laptop may include a DRP port that can receive power to charge the laptop's battery, but it can also deliver power to charge external accessories.

  Additionally, there are two special subclasses of a DRP:

  – A sourcing device is capable of supplying power, but not capable of acting as a DFP. One example of this subclass is a USB-C and USB PD-compatible monitor that receives data from a laptop's DFP, but also charges the laptop.
  – A sinking host is capable of consuming power, but not capable of acting as a UFP. An example could be a hub's DFP that sends data to an accessory while being powered by that accessory.

USB-C pinout and reversibility

The USB-C connector includes several new pins compared to USB Type-A and Type-B connectors. These pins enable USB-C features such as higher power, Alternate Mode and reversibility. **Figure 1** illustrates the pinout.

From left to right, **Figure 1** shows:

- GND: the return path for the signal.
- TX and RX: SuperSpeed twisted pairs for USB 3.1 data (5Gbps to 10Gbps).
- VBUS: the main system bus (5V to 48V).
- CC1 and CC2: CC lines used for cable detection, orientation and current advertisement. With USB PD, the CC lines can also communicate higher power levels and Alternate Mode. Note that one of the CC lines may become VCONN.
- SBU1 and SBU2: these are low-speed lines used only for Alternate Mode and accessory mode. For example, with DisplayPort, AUX+ and AUX– transmit over the SBU lines. For audio adapter accessory mode, these lines are used for the microphone input and analog GND.
- D+ and D–: a high-speed twisted pair for USB 2.0 data (up to 480Mbps).

Figure 1. *USB-C receptacle pinout*

A new aspect of the USB-C connector is that the pins are almost symmetrical (both vertically and horizontally). This is why the connector can be reversible. Unfortunately, it's not possible to passively realize reversibility, so additional electronics are required. **Figure 2** shows how a USB-C receptacle (top) and a USB-C plug (bottom) are essentially flipped relative to each other.

- The GND and VBUS lines are still in the same position.
- The D+ and D– pair is in the same orientation; however, the plug contains only one D+ and D– twisted pair. The USB-C specification allows shorting of the D+ and D– lines together (D+ to D+ and D– to D–) on the receptacle side. Regardless of cable orientation, the physical layer (PHY) will always see the cable's D+ and D– pair.
- The CC1 and CC2 lines are flipped and can determine the cable orientation. The orientation determines which CC line is connected and which one is left open.
- The TX and RX pairs are also flipped. Resolving this was a bit more complicated. Unlike the D+ and D– lines, you cannot simply short the common lines together because that will create a stub. At USB 2.0 speeds, a stub is acceptable, but at USB 3.1 speeds, a stub degrades signal integrity too much. To avoid this, there are two options:
  - Use two PHYs and cable-orientation detection to know which PHY to use.
  - Have a single PHY and a SuperSpeed multiplexer that switches the correct SuperSpeed lines to the PHY (given the known orientation). This is typically the more economical solution.
- The SBU lines are also flipped; however, this is typically handled within the Alternate Mode PHY (remember that these are slow-speed lines).

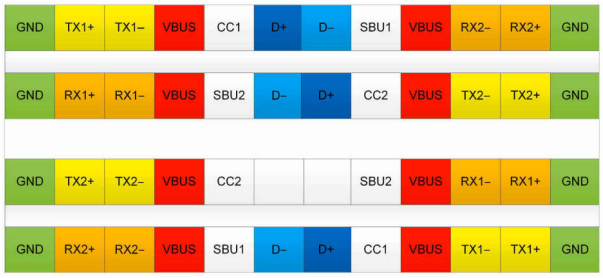


Figure 2. *USB-C pinout: receptacle (top); plug (bottom)*

USB-C cable detection and orientation

The configuration channel (CC) logic block introduced in the USB-C specification determines cable detection, cable orientation and current-carrying capability. Let's define these terms:

- **Cable detection** occurs when one of the two CC lines pulls down (see **Figure 3**). A DFP will have both of its CC pins pull up with resistor Rp, and a UFP will have both of its CC pins pull down with resistor Rd. Once a DFP processor detects that one of its CC lines is pulled down, the DFP will know that a connection has been made.
- **Cable orientation** is based on which CC line pulls down (if CC1 pulls down, the cable is not flipped; if CC2 pulls down, the cable is flipped). For nonactive cables, the remaining CC line remains open; for active cables, the remaining CC line will pull down with Ra.
- The values of Rp determine the **current-carrying capability**. USB-C can natively support either 1.5A or 3A. A DFP can advertise its current-carrying capability with a specific value pullup resistor. A UFP has a fixed-value pulldown resistor (Rd) such that when connected, it forms a voltage divider with Rp. By sensing the voltage at the center tap of the voltage divider, a UFP can detect the DFP's advertised current.

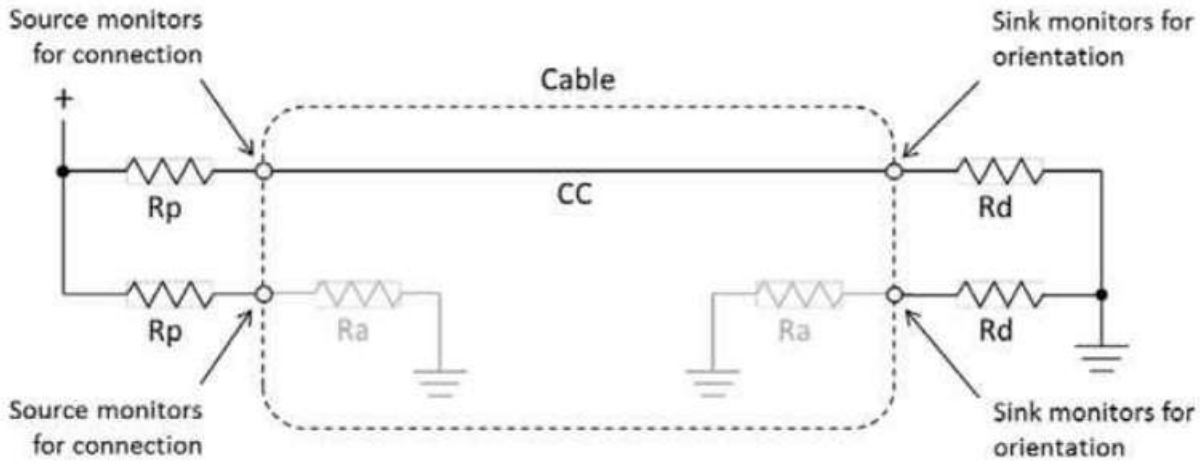


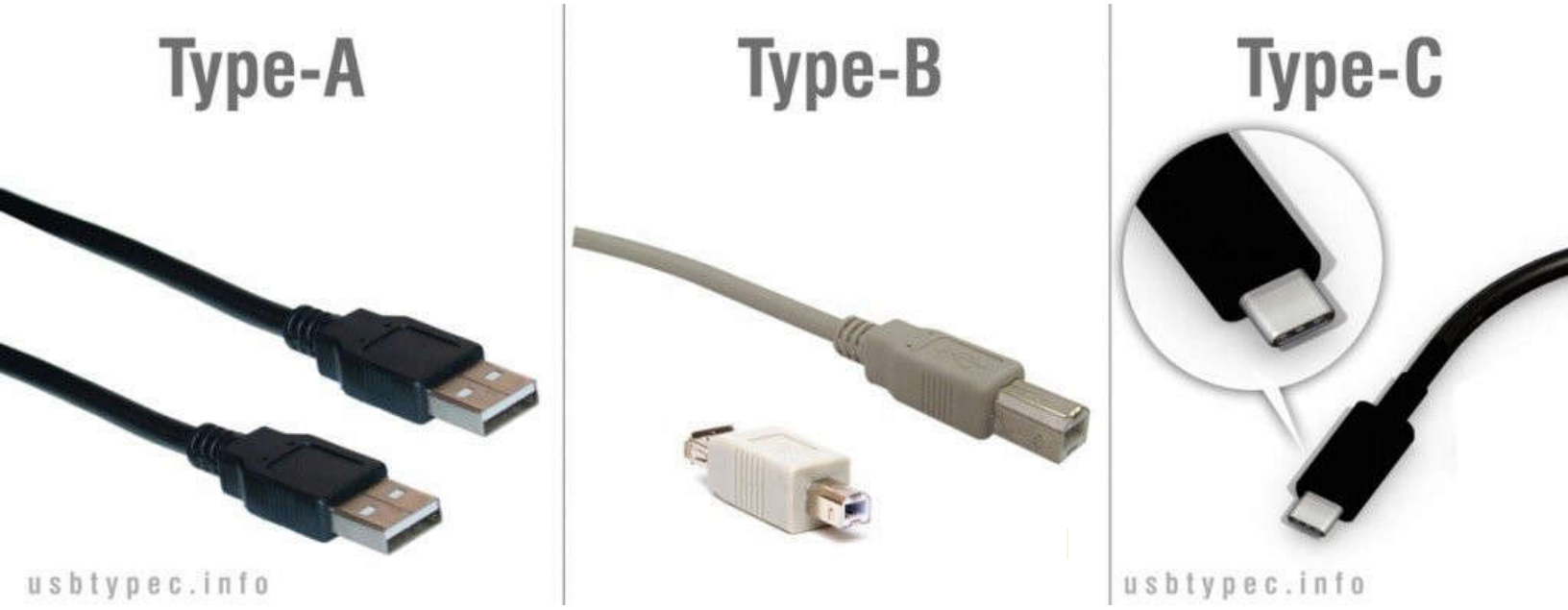
Figure 3. *CC logic pullup and pulldown termination. (Source: USB Type-C specification v1.2, Figures 4 and 5 pullup and pulldown CC model)*

When do you need a USB PD controller?

USB PD is a standard using the USB-C connector, so you may wonder when you need to use a USB PD controller. If your requirements include any of these three scenarios, you will need a USB PD controller:

- **Negotiating a voltage greater than 5V**. Even if the maximum power is less than 15W, you will need a USB PD controller to negotiate a voltage greater than 5V. For example, if you have a system that needs 15W, but you only need 5V, you do not need a USB PD controller. However, if you have a system that only needs 10W, but you need 9V, you will need a USB PD controller to negotiate the 9V contract.
- **Supporting video such as DisplayPort** on a USB-C connector requires a USB PD controller.
- **The power role and data role do not match. I**f you want to have different data and power roles, you need a USB PD controller (source and UFP). A good example is a docking station. The docking station will act as a power source for a laptop to charge, but a UFP to receive video and USB data. This happens through the use of power-role swaps and data-role swaps.

History of USB Type-C®



Abstract

Authors: Taylor Vogt

In this section, we'll discuss the history of the USB protocol in order to provide background into the changing world of USB Type-C® (USB-C®) and USB Power Delivery (PD). We'll first touch on the USB Type-A and USB Type-B definitions, and how the USB protocol evolved over time into the latest USB PD 3.1 specification.

USB connector basics

Before discussing the protocol itself, the connector is an important medium to the interface that has taken on different forms over time. This is important to distinguish, as moving forward, the ever-improving USB protocol capabilities will require the USB-C connector to attain the full feature set. **Figure 4** shows the different USB connectors leading up to USB-C.

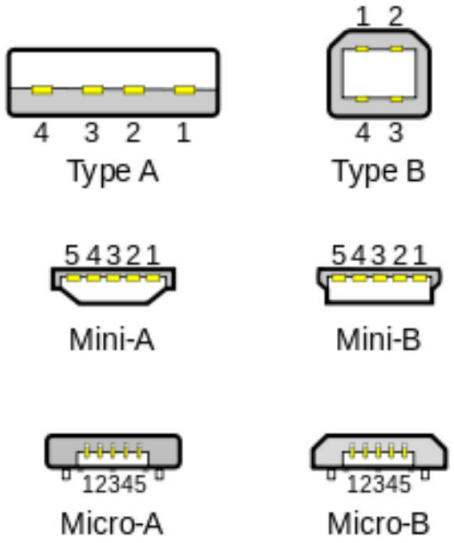


Figure 4. *USB Connector Overview*

USB Type-A

Generally more well known as the most popular and common USB standard is USB Type-A, which began use around 1996. You'll find USB Type-A ports in host devices such as desktop computers, game consoles and media players.

USB Type-B

USB Type-B connectors are on one end of typical USB cable that plugs into a peripheral device, such as a smartphone, printer or hard drive. These also debuted in 1996, typically to control a device that connected through Type-A to a PC on the other end.

USB-C

USB-C was implemented into several popular phone and notebook brands around 2015. USB-C had an innovative design in terms of connector reversibility for ease of use and downsizing to minimize its impact on thinner and sleeker devices. **Figure 5** shows the progression of USB cables from USB Type-A to USB Type-C while **Figure 6** taking a closer look into the USB-C connector itself.

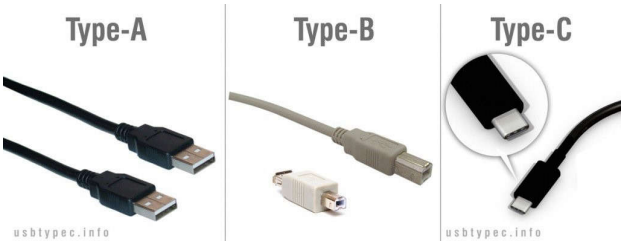


Figure 5. *USB Cable Overview*

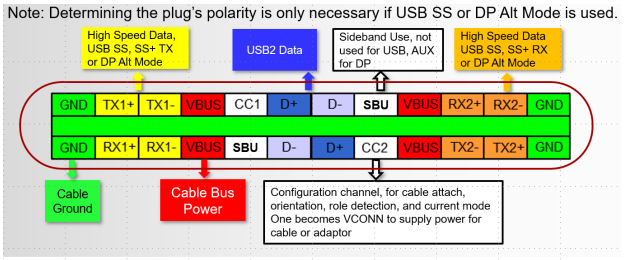


Figure 6. *USB-C Connector Overview*

USB and USB PD protocol history

In terms of the protocol itself, there are now a total of six USB specifications defined by the USB Implementers Forum (USB-IF): USB 1.0, 2.0, 3.0, 3.1, 3.2 and 4.0. However, USB 1.0 is essentially no longer used, so USB 2.0 through USB 4.0 are the standards referenced today.

In 2012, the first USB PD 1.0 specification was released, but was quickly followed in 2014 by the USB 2.0 version to stipulate the use of the USB-C connector and to clean up some of the technical details for five supportable power-supply levels: 15W, 27W, 45W, 60W and 100W.

USB PD 3.0 came out in 2018, adding some flexibility to the standard to better suit a wide range of devices. It improved the communication protocol to support features such as battery condition monitoring, enhanced security and fast role swapping. It also introduced the Programmable Power Supply (PPS) protocol, which allowed voltage levels at 20mV granular increments. This enabled custom voltage negotiation for fast-charging applications requiring a fine-tuned voltage level.

The USB PD 3.1 specification was released in 2021. This was a major update to enable the delivery of as much as 240W of power over a USB-C cable and connector. Going beyond 100W to 240W is known as Extended Power Range (EPR), while the previous USB PD range is now known as Standard Power Range (SPR).

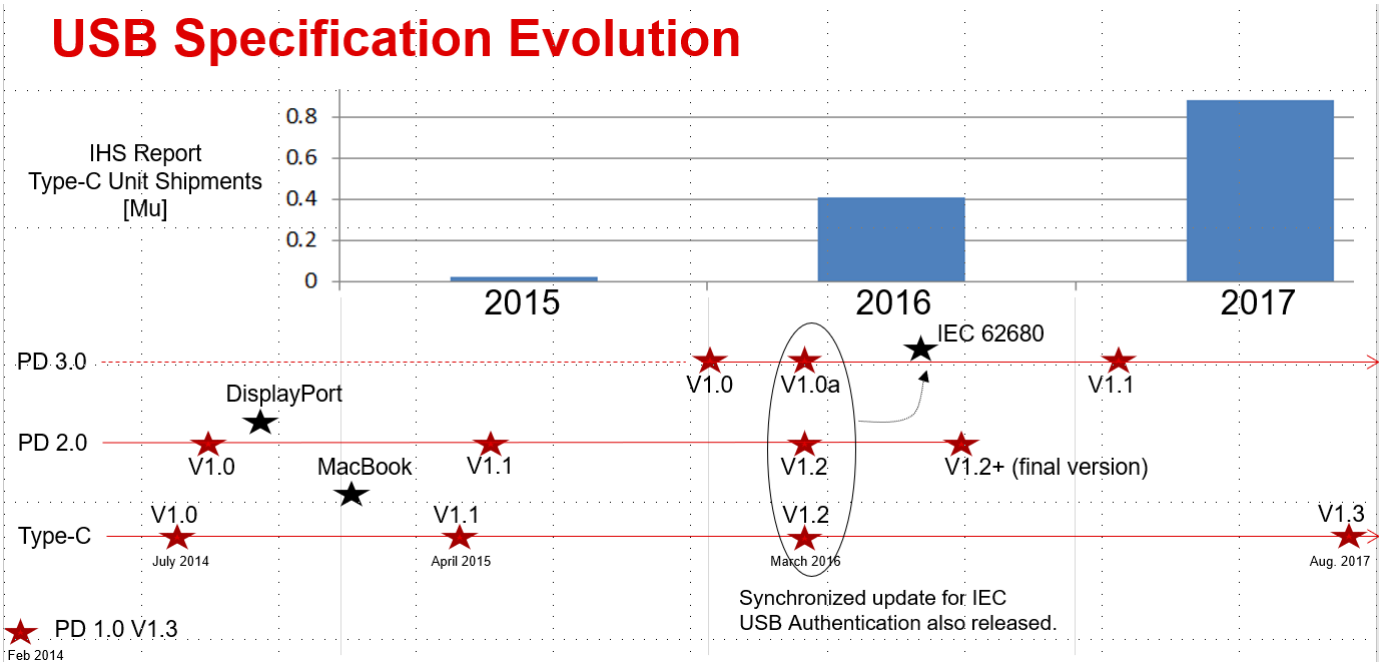


Figure 7. *USB Specification Evolution*

In contrast to the previous USB protocols, the USB 4.0 standard requires USB-C connectors, given the increase in supported features. Additionally, a new development in USB 4.0 allows for DisplayPort™ and PCI Express (PCIe).

A later chapter will dive into further details, but at a high level, here are some aspects of USB and Thunderbolt technology:

- USB 3.2:
  - USB 3.2 Gen 1 (formerly USB 3.0), SuperSpeed up to 5Gbps.
  - USB 3.2 Gen 2 (formerly USB 3.1), SuperSpeed up to 10Gbps.
  - USB 3.2 Gen 2x2 (actual USB 3.2), SuperSpeed up to 20Gbps.
  - Multilane operation with two lanes of 10Gbps for 20Gbps data rates.
  - Does not require a USB PD power contract.
- Thunderbolt 3:
  - Combines USB (2.0, 3.0 and 3.1), PCIe and DisplayPort into single interface.
  - Requires a USB PD contract.
  - Enabled once Intel's Thunderbolt 3 Alternate Mode is negotiated.
- USB 4.0:
  - Two-lane operation using existing USB-C cables and up to 40Gbps operation.
  - Backward compatibility with USB 3.2, USB 2.0 and Thunderbolt 3.
  - Requires a USB PD power contract.
  - Does not depend on Alternate Mode entry.

Figure 8 shows a visual comparison of the USB data transfer speeds.

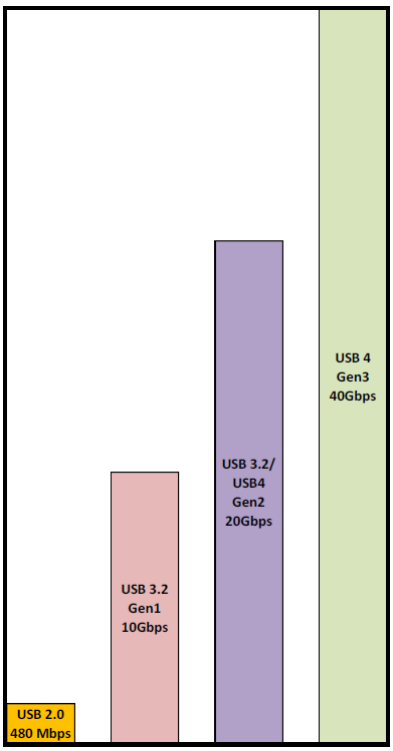


Figure 8. *USB data transfer speeds*

To simplify a USB network, there is one host and one device. Typically, the PC is the host, and a smartphone, tablet or camera is the device. In terms of data and power, power flows from the host to the device, while data can flow in both directions.

A USB 1.0 and 2.0 standard downstream port can deliver up to 500mA or 0.5A at a data rate of up to 480Mbps. USB 3.0 provides up to 900mA or 0.9A, at a data rate of. These power output specifications are a rating based on the 5V from each standard output. However, the dedicated USB 3.0 charging and charging downstream ports provide up to 1,500mA or 1.5A, which translates into 7.5W.

USB-C vs. USB PD

Fundamentally, USB-C refers directly to the connector hardware used to plug in the system, whereas USB PD refers to the protocol. On the USB-C side, it is the latest USB interface that combines power, video and data transfer. Cosmetically, USB-C is smaller than the USB Type-A connector and allows for reversibility in its connection. Because of its enhanced feature set, a single USB-C connector can replace multiple connectors of an existing device in a system.

By default, the USB-C connector works on a 5V and 3A power domain and can be used as such in applications that do not require higher power. However, the USB-C connector enables the use of the USB PD protocol, which can provide as much as twice the power of the USB Battery Charging 1.2 specification, up to 100W (20V and 5A). USB PD supports high-bandwidth video and data rates through alternate modes such as DisplayPort or Thunderbolt over the USB-C cable. **Figure 9** shows all of the different power modes and the sequence in which they are implemented.

Precedence	Mode of Operation	Nominal Voltage	Maximum Current
Highest	USB PD	Up to 20 V	Up to 5 A
	USB Type-C current @ 3A	5 V	3 A
	USB Type-C current @ 1.5A	5 V	1.5 A
	USB BC1.2	5 V	Up to 1.5 A
	USB 3.1	5V	900 mA
Lowest	USB 2.0	5V	500 mA

Figure 9. *Priority of power modes*

The USB-IF specification describes the USB PD protocol in detail, including the steps required for a system to enter into a USB PD contract. At a glance, it is possible for a USB PD controller to execute these steps, while there are separate instructions based on whether the port is a source or a sink. The source would first advertise its power capabilities through some number of desired power delivery objects (PDOs) and the sink would request one of them. The source would need to accept this request, and the sink would need to acknowledge, in order to enter the USB PD contract. **Figure 10** shows this negotiation sequence.

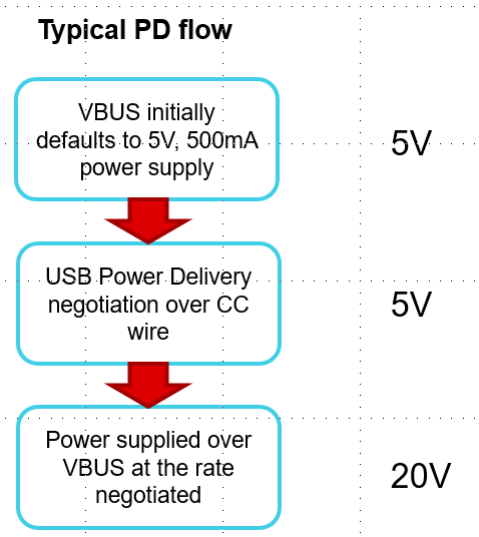


Figure 10. *USB PD power negotiation sequence*

Here are some reasons why it makes sense to use a USB PD controller to enable USB-PD:

- Higher voltages – up to 20V – while 5V, 9V, 15V also satisfy most customer needs. (The latest USB PD specification includes 28V, 36V and 48V.)
- Up to 100W power over a USB cable (240W if including EPR in the latest USB PD specification).
- The port that provides power is negotiable: the downward-facing port can be the provider or consumer, while the upward-facing port can also be the provider or consumer.
- Efficient power management across multiple peripherals.
- The downward-facing port has a cold socket to conserve power.
- Coexistence with legacy USB products.
- Enabled Alternate Mode.

Evolution of the USB PD 3.1 specification

A later chapter will go into more detail on PD3.1 and the extended power range. Here are the major features as summarized by the USB-IF:

- New 28V, 36V and 48V fixed voltages enable up to 140W, 180W and 240W power levels, respectively. An adjustable voltage supply mode allows the device being powered the ability to request intermediate voltages between 15V and higher to the maximum available fixed voltage of the charger.
- The power direction is no longer fixed. In other words, the product host or peripheral is enabled to provide power.
- Each device can negotiate the minimum power required to ensure that power is available if additional requests occur.
- Intelligent system-level power management through the use of optional hub communication with the PC.
- Allows low-power cases such as headsets to negotiate for only the power that they require.

While the future remains unclear, there is a movement in the European Union to universalize USB-C connectors across all small electronic devices, including phones and portable chargers, in order to reduce e-waste and simplify consumer ease of use.

Introduction and Overview of the USB Type-C® and USB PD Specifications

Abstract

Authors: Adam McGaffin, Eric Beljaars and Ghouse Mohiuddin

The USB Type-C® (USB-C®) and USB Power Delivery (USB PD) specifications are two protocols used in a USB-C connector. While they are two separate protocols, they are closely intertwined with one another, as you cannot negotiate a USB PD contract without first establishing a USB-C contract. In this chapter, we'll walk through both specifications and introduce complex USB PD topics.

USB-C connections

There are two configuration channel (CC) pins on USB-C connectors and cables, one for each orientation; this is what helps USB-C be reversible. The CC pins determine when a cable is connected and whether the USB-C port will assume a source or sink role; the pins also transmit and receive all USB PD messaging. Upon connection, based on the polarity, one of the CC lines is used for messaging, while the other CC line can be used for VCONN to power active cables and e-markers.

A USB-C power source will present a pullup resistor on the CC pins (Rp), while a USB-C power sink will present a pulldown resistor on the CC pins (Rd). When a source connects with a sink, Rp and Rd will form a voltage divider on the CC pin. This is how a USB-C connection is detected. In addition to the power role of the port, with standard USB-C, Rp and Rd also determine the data role. Rp is presented by the power source, which is always the downstream-facing port (DFP), while Rd is presented by the power sink, which is always the upstream facing port (UFP). After the connection is detected through the Rp and Rd resistor divider, the source must provide 5V and the sink can begin consuming current based on the implicit USB-C contract.

The resistor value of Rp will determine how much current the source can provide, as highlighted in the USB-C specification. The sink will detect the Rp resistor value through the strength of the pullup and should limit its current consumption based on the attached source's capabilities. Note that USB-C is a cold connector, meaning that there is 0V on VBUS when nothing is connected. This is different from legacy USB Type-A, which always had 5V present on VBUS. Cold connection means that each USB-C source port will need to support enabling and disabling 5V depending on if a device is attached or not, along with controlling the current across it.

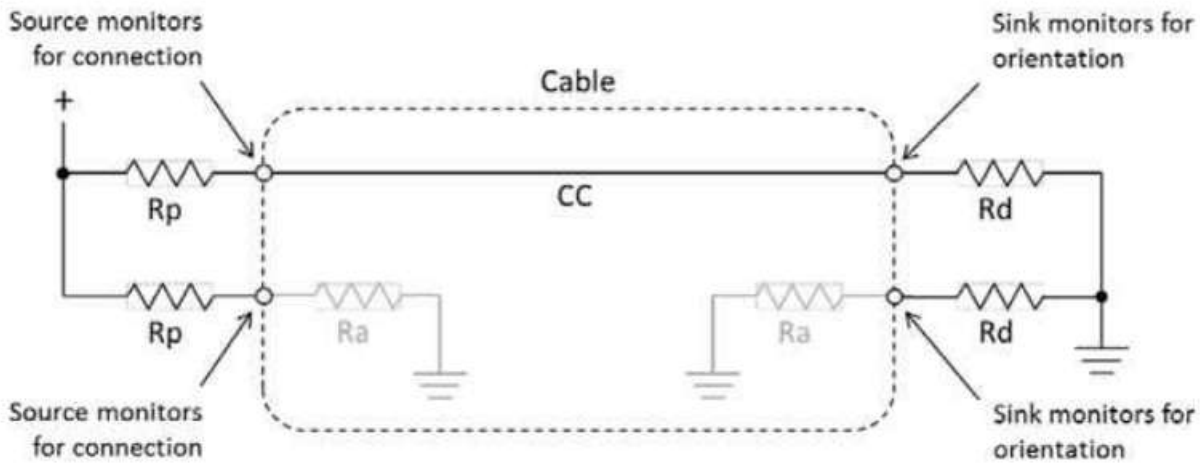


Figure 11. *CC logic pullup and pulldown termination. (Source: USB Type-C specification v1.2, Figures 4 and 5 pullup and pulldown CC model)*

VCONN and messaging types

After establishing a standard USB-C contract, the CC line can then transmit and receive USB PD messages to and from the connected device. In cases where there is >3A of current, USB 3.0 data rates, or DisplayPort™ or Thunderbolt alternate modes, the CC line can also provide power on VCONN (the opposite CC line) to power an e-marked cable or an active cable.

There are three different types of CC messages: start of packet (SOP), SOP' and SOP". The message type indicates the device to which the message is being sent:

- SOP messages move from USB PD controller to USB PD controller across the cable.
- SOP' messages transmit to the e-marker at the end of the cable that's connected to the port sending the message.
- SOP" messages transmit to the e-marker at the end of the cable opposite the port sending the message.

Figure 12 highlights where the DFP or source USB PD controller sends SOP, SOP' and SOP" messages. In this chapter, we'll focus on standard SOP messaging from one USB PD controller to another, since that's where most negotiations occur. Communication to the e-marker on either end of the cable is typically just a compatibility check to understand cable capabilities.

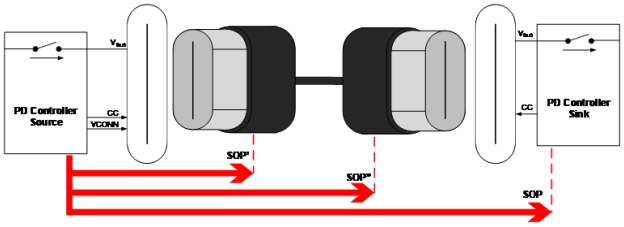


Figure 12. *SOP, SOP' and SOP" messaging*

Now that you know that signaling occurs on CC wires, and how to identify which device is sending a message, let's talk about the types of messages and what they do. There are three main categories of USB PD messages: control messages, data messages and extended messages.

Control messages are short, and used to manage the message flow between port partners or to exchange messages that require no additional data. Control messages are 16 bits in length. **Table 3** shows the full list of control message types.

Table 3. Control message types

Message type	Sent by
GoodCRC	Source, sink or cable plug
GotoMin	Source only
Accept	Source, sink or cable plug
Reject	Source, sink or cable plug
Ping	Source only
PS_RDY	Source or sink
Get_Source_Cap	Sink or dual-role power (DRP)
Get_Sink_Cap	Source or DRP
DR_Swap	Source or sink
PR_Swap	Source or sink
VCONN_Swap	Source or sink

Table 3. Control message types (continued)

Message type	Sent by
Wait	Source or sink
Soft_Reset	Source or sink
Data_Reset	Source or sink
Data_Reset_Complete	Source or sink
Not_Supported	Source, sink or cable plug
Get_Source_Cap_Extended	Sink or DRP
Get_Status	Source or sink
FR_Swap	Sink
Get_PPS_Status	Sink
Get_Country_Codes	Source or sink
Get_Sink_Cap_Extended	Source or DRP
Get_Source_Info	Sink or DRP
Get_Revision	Source or sink

Data messages are used to exchange information between a pair of port partners. Data messages range from 48 bits to 240 bits in length. There are three types:

- Those used to expose capabilities and negotiate power.
- Those used for built-in self-test (BIST).
- Those that are vendor-defined.

Table 4 shows the full list of data message types.

Table 4. Data message types

Message type	Sent by
Source_Capabilities	Source or DRP
Request	Sink only
BIST	Tester, source or sink
Sink_Capabilities	Sink or DRP
Battery_Status	Source or sink
Alert	Source or sink
Get_Country_Info	Source or sink
Enter_USB	DFP
EPR_Request	Sink
EPR_Mode	Source or sink
Source_Info	Source
Revision	Source, sink or cable plug
Vendor_Defined	Source, sink or cable plug

Like data messages, extended messages are also used to exchange information between a pair of port partners. There are several types of extended messages:

- Those used for source and battery information.
- Those used for security.
- Those used for firmware updates.

- Those that are vendor-defined.

Table 5 shows the full list of extended message types.

Table 5. Extended message types

Message type	Sent by
Source_Capabilities_Extended	Source or DRP
Status	Source, sink or cable plug
Get_Battery_Cap	Source or sink
Get_Battery_Status	Source or sink
Battery_Capabilities	Source or sink
Get_Manufacturer_Info	Source or sink
Manufacturer_Info	Source, sink or cable plug
Security_Request	Source or sink
Security_Response	Source, sink or cable plug
Firmware_Update_Request	Source or sink
Firmware_Update_Response	Source, sink or cable plug
PPS_Status	Source
Country_Info	Source or sink
Country_Codes	Source or sink
Sink_Capabilities_Extended	Sink or DRP
Extended_Control	Source or sink
EPR_Source_Capabilities	Source or DRP
EPR_Sink_Capabilities	Sink or DRP
Vendor_Defined_Extended	Source, sink or cable plug

For detailed descriptions on each message type, see the USB PD specification.

Negotiating USB PD power over CC wires

Now that you've entered a basic USB-C implicit contract through Rp and Rd, determined which CC line is for communication and which is for VCONN, and communicated with the cable to understand its capabilities through SOP' and SOP" messages, let's begin USB PD negotiation between two devices using SOP messaging.

USB PD messaging consists of 300kbps ±10% biphase mark code (BMC) signaling between two connected devices. This messaging takes place on the CC pins.

The first message sent in USB PD negotiation will be Source_Capabilities coming from the DFP and source port. The Source_Capabilities message contains the power data objects (PDOs) that the source is capable of providing to the attached device.

Between each USB PD message is a GoodCRC message coming from the device that just received the USB PD message. For example, once the DFP and source port send Source_Capabilities, the sink and UFP port will respond with a GoodCRC message before the next message. GoodCRC means that the attached device was able to receive the message. After the sink and UFP port receive the Source_Capabilities message from the attached device, it will send a Request message asking for the PDO it needs to operate. If there isn't a direct match to the sink's required operating current in the source's capabilities, the sink will toggle the Capabilities Mismatch bit in the Request message to indicate this to the source.

Once the DFP and source port receive the Request message from the attached sink, the source will send an Accept message to the sink. The source will adjust the voltage on VBUS to match the request from the sink.

Once the voltage on the source is within ±5% of the requested PDO, the source will send a PS_RDY message to indicate to the sink that the voltage is good. After the PS_RDY message, the sink can start drawing current up to the requested PDO.

Figure 13 shows how USB PD power negotiation looks on a standard USB PD analyzer. We recommend acquiring a USB PD analyzer when designing a system with USB PD in order to record CC negotiation traffic between two devices. It can be very helpful in debugging if you run into unexpected behavior.

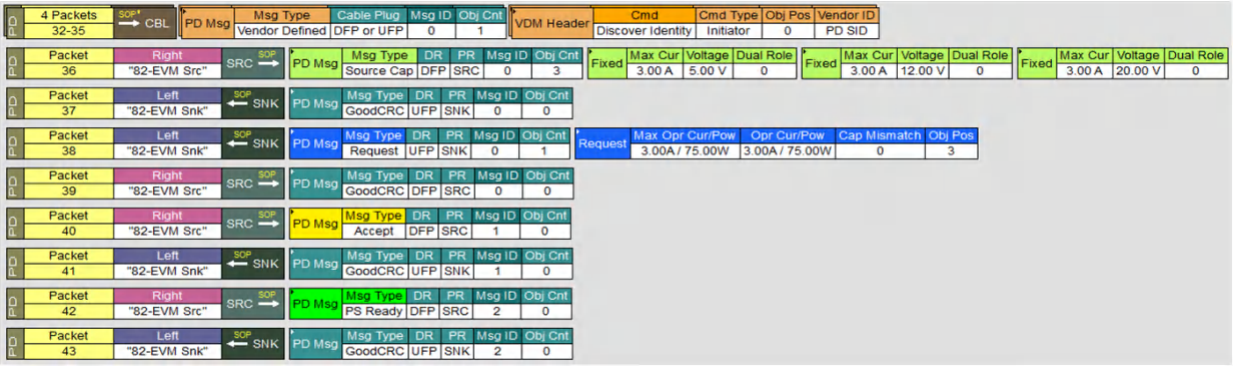


Figure 13. *USB PD power negotiation*

Figure 13 illustrates the following sequence of messages:

1. The source detects the cable capabilities or plug type if these are not already known. The source sends a Source_Capabilities message that represents the present capabilities of the power supply with an appended cyclic redundancy check (CRC).
2. The sink policy engine evaluates the Source_Capabilities message sent by the source, detects the plug type if required, and selects which power supply to use. The sink forms the data (such as a PDO) representing the request into a message and sends the message.
3. The source policy engine evaluates the request message sent by the sink and decides whether it can complete the request. The source forms and sends an accept message with an appended CRC, which triggers the following actions:
   - The sink enters the SnkStandby period and pulls less than 500mA.
   - The source begins to transition the voltage on VBUS from VBUS_old to VBUS_new, which is from 5V to 20V in this case.
4. The source device-policy manager informs the policy engine that the power supply has settled at a new operating condition and sends a PS_RDY message with an appended CRC.

For further details, see the USB PD specification.

Figure 14 highlights a successful fixed, variable, or battery Standard Power Range (SPR) power negotiation as described in the USB PD specification. If you're interested, you can search keywords throughout the e-book within the USB PD specification for additional details.

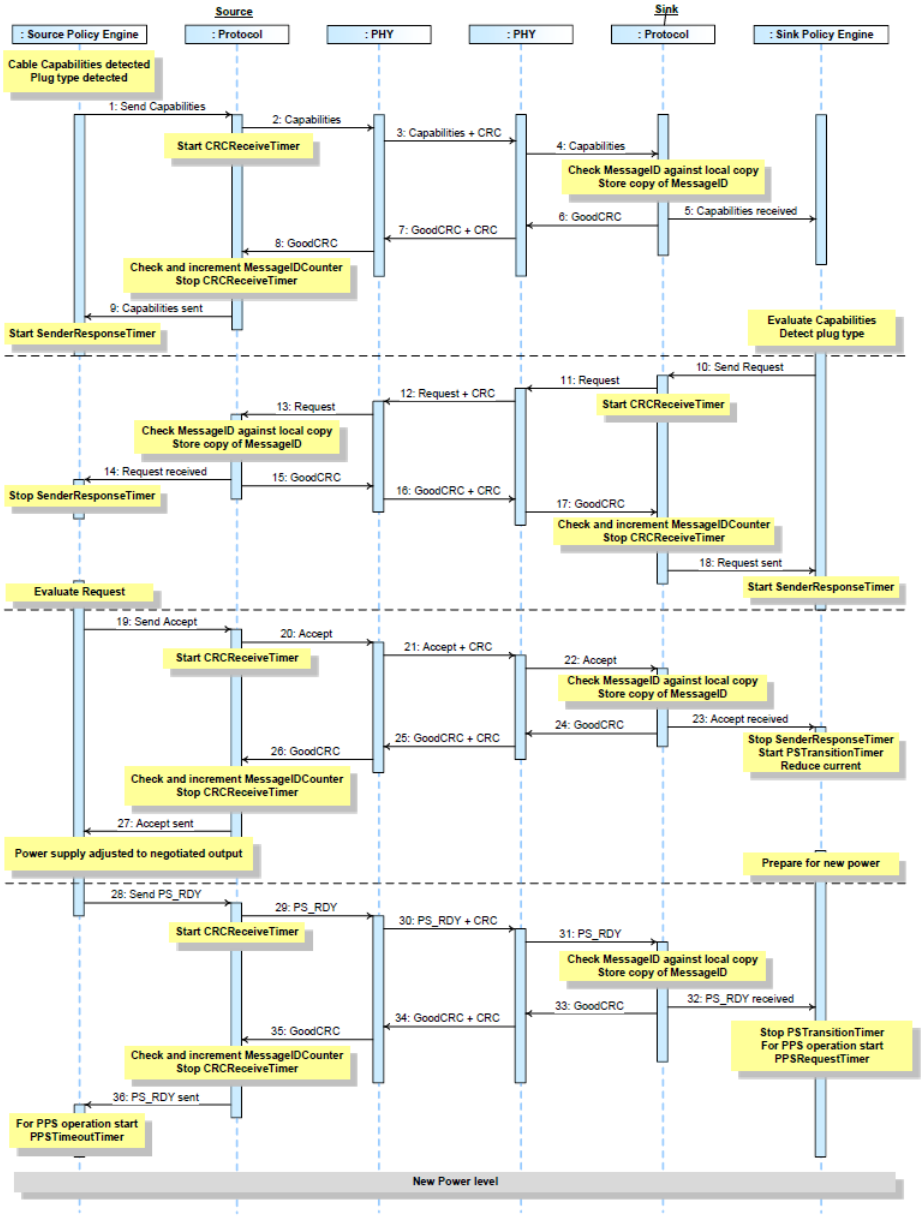


Figure 14. *Successful fixed, variable or battery SPR power negotiation*

Data-role swaps

When you enable a USB-C port with USB PD capabilities, you can mix and match data roles and power roles. With a standard USB-C connection, the data role and power role will always match which is determined by the Rp or Rd resistor presented on the CC lines. USB PD messaging adds the capabilities of completing a power-role swap or data-role swap to mix the power role and data role. This means that your USB PD port can become a DFP and sink or a UFP and source.

For example, when connecting a laptop to a docking station, you want the docking station to power and charge your laptop, but you also want to receive data from the PC in order to connect to a mouse, keyboard or displays. In this case, the docking station would need to be the power source and data UFP. To enter this state, you need either a data-role swap or a power-role swap.

Because both the laptop and docking station are typically DRP-capable, either device can end up in each original power or data role. Let's say that a laptop starts as the UFP and sink, and the docking station starts as the DFP and source. In this case, you're in the correct power role but need to complete a data-role swap to get in the correct data role. Either port partner can initiate the data-role swap. However, it's more common for the laptop to initiate the swap, since it wants to become the DFP. The sequence of events would look this:

1. The UFP and sink sends a data-role swap (DR_Swap in the USB PD specification) message to DFP and source.
2. The DFP and source receives the data-role swap message and sends back an Accept message.

At this point, the laptop would become the DFP and sink, and the data-role swap would be complete. **Figure 15** shows what this sequence looks like in the USB PD specification.

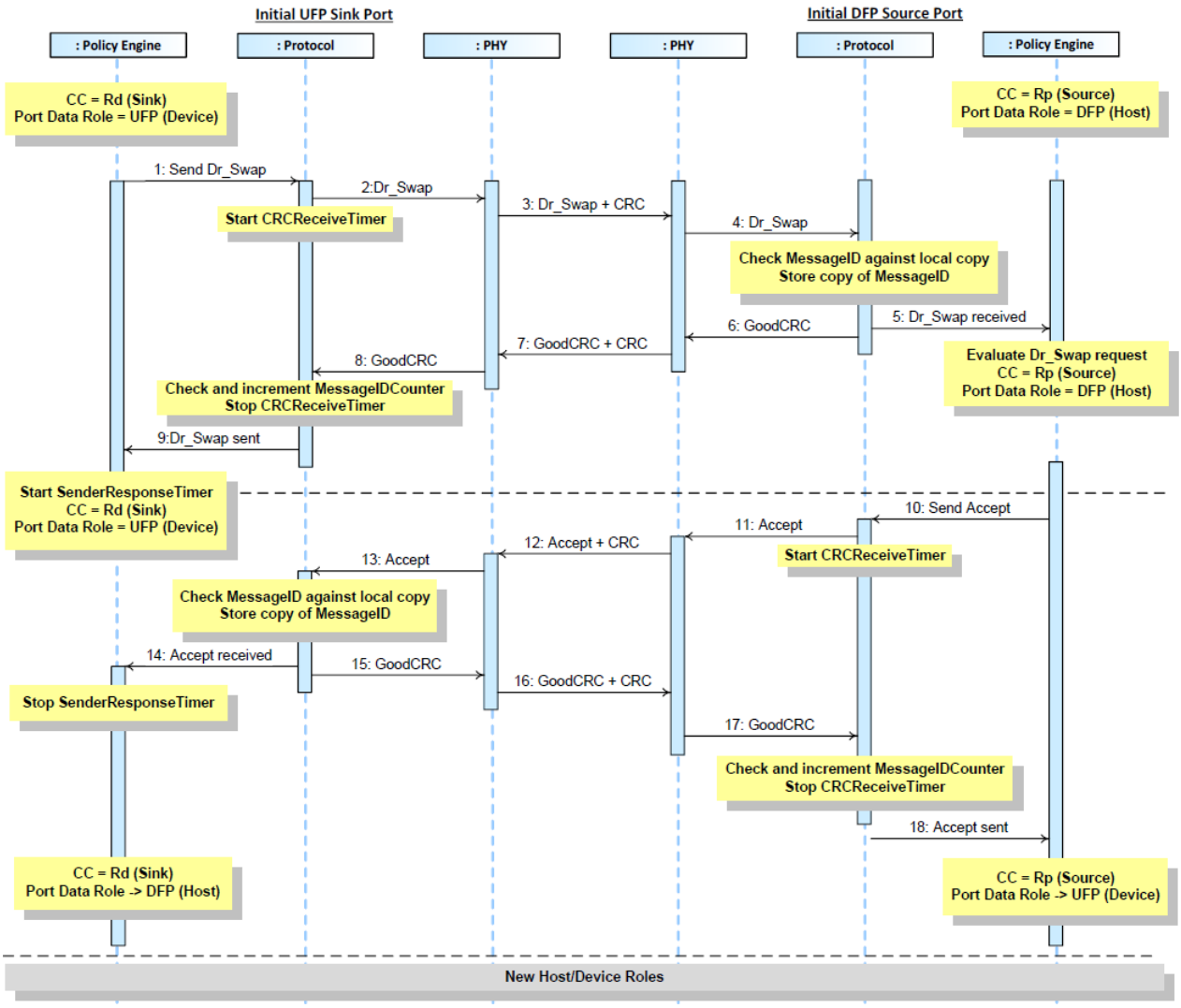


Figure 15. *Succesful data-role swap, going from UFP to DFP*

Power-role swaps

Let's next describe an example where the laptop connects as the DFP and source. In this case, you're in the correct data role, but you'll need to complete a power-role swap so that the dock can become the power source and start charging the laptop.

Here is the sequence of events that would occur when the laptop (DFP and source) initiates a power-role swap to become the power sink:

1. DFP and source sends power-role swap (PR_Swap in the USB PD specification) message to UFP and sink.
2. The UFP and sink sends an accept message back to the DFP and source.
3. The DFP and source stops sourcing power and changes the CC termination from Rp to Rd, indicating that it will become the sink. After this, the DFP and source sends a PS_RDY message to indicate that it has stopped sourcing power.
4. The initial UFP and sink receives the first PS_RDY from the initial DFP and source. The initial UFP and sink will change its CC termination from Rd to RP and begin providing 5V on VBUS.
5. When 5V is available on VBUS, the initial UFP and sink sends the second PS_RDY and the power-role swap is complete.

You'll notice that the sequence stopped with the dock providing only 5V on VBUS, after which it will begin the same USB PD power negotiation by sending its source capabilities. **Figure 16** shows the sequence of events for a power-role swap initiated by the initial sink, as highlighted in the USB PD specification.

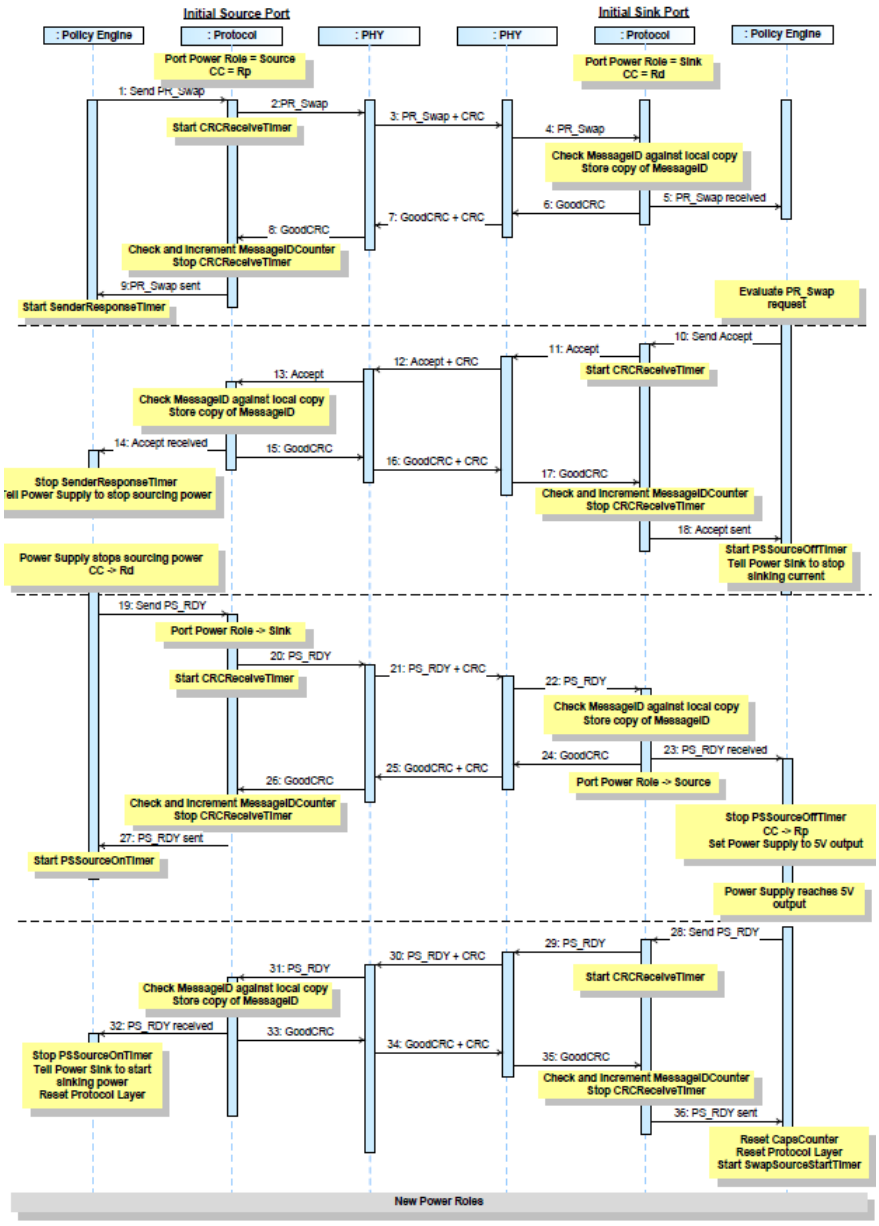


Figure 16. *Succesful power-role swap, initial sink to new source*

Introduction to USB PD alternate mode

An important benefit of USB-C is its ability to eliminate the need for nearly every cable in consumer devices (DisplayPort, Thunderbolt, power barrels, USB Type-A, USB Type-B). To do this, USB-C needed additional functionality beyond USB 3.0, which led the USB Implementers Forum (USB-IF) to define Alternate Mode. Alternate Mode enables the repurposing of USB-C pins (transmitter and receiver pairs and sideband use) for a different function. Up to this point, video has been the primary focus for Alternate Mode, with DisplayPort and Thunderbolt being the main two alternate modes for implementing video across a USB-C cable.

Introduction to EPR

Until recently, the USB PD 3.0 specification allowed for both power and data in both directions for up to 100W (20V, 5A) across an approved USB-C port and cable. The latest USB PD 3.1 specification increases the wattage up to 240W (48V/5A). To keep the terminology consistent, the USB-IF renamed the previous USB PD range to SPR, and the new specifications (between 100W and 240W) as Extended Power Range (EPR).

Table 6. USB power levels

Specification	Maximum voltage	Maximum current	Maximum power
USB 2.0	5V	500mA	2.5W
USB 3.0 and USB 3.1	5V	900mA	4.5W
USB Battery Charging 1.2	5V	1.5A	7.5W
USB-C 1.2	5V	3A	15W
USB PD 3.0	20V	5A	100W
USB PD 3.1	48V	5A	240W

USB signals over USB Type-C®

Introduction

Author: Undrea Fields

USB-Type C® (USB-C®) is compatible with USB specification versions 1.0, 1.1, 2.0, 3.2 Gen 1 (SuperSpeed USB), 3.2 Gen 2 (SuperSpeed USB 10Gbps), 3.2 Gen 2x2(SuperSpeed 20Gbps), USB 4 20Gbps, USB4 40Gbps and USB4 80Gbps.

All USB signaling over USB-C occurs over six differential pairs:

- D1+, D1– (Low Speed, Full Speed, High Speed).
- D2+, D2– (Low Speed, Full Speed, High Speed).
- TX1+, TX1– (SuperSpeed channel 1 transmit [TX]).
- Rx1+, RX1– (SuperSpeed Channel 1 receive [RX]).
- TX2+, TX2– (SuperSpeed Channel 2 TX).
- Rx2+, RX2– (SuperSpeed Channel 2 RX).

USB 2.0 Signaling Over Type-C

Per the USB specification, when implementing USB-C, USB 2.0 (Including USB 1.0 and USB 1.1) signaling must be supported in parallel with SuperSpeed signaling over the USB-C interface in order to maintain backward compatibility.

Low speed and full speed

Low-speed and full-speed signaling are 3.3V signaling. As seen in **Figure 17**, a low- or full-speed device will have a 1.5kΩ pullup resistor to 3.3V on either the D+ (full speed) or D– (low speed) signal, which allows the host to determine the required speed of the interface.

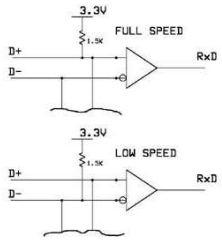


Figure 17. *Low- and full-speed pullups*

High speed

High-speed signaling comprises a combination of 3.3V and 800mV differential signaling. As seen in **Figure 18**, a high-speed device will have 45Ω terminations on both the D+ and D– signals.

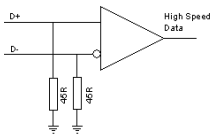


Figure 18. *High-speed termination*

Initially, a high-speed device is in an idle state, which is 3.3V on D+ preceding high-speed negotiation. Upon the successful completion of high-speed negotiation, high-speed packets (at 480Mbps) are sent with a differential voltage of approximately 400mV. This process is shown in **Figure 19**.

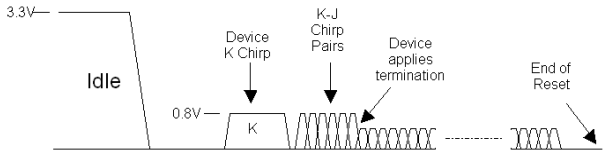


Figure 19. *High-speed negotiation*

Low-, full- and high-speed data rates

Through each revision of the USB specification, data throughput increased. **Table 7** lists the throughput for low speed, full speed (USB 1.1) and high speed (USB 2.0).

Table 7. USB 1.1 and 2.0 data rates

Name	Speed
Low speed	1.5Mbps
Full speed	12Mbps
High speed	480Mbps

USB 2.0 signal integrity

With the increased data rate of USB 2.0, there may be certain circumstances, especially on a desktop or server-type platform with longer printed circuit board traces from a host to USB connector, that may require the aid of a signal conditioner such as the TUSB211A from Texas Instruments in order to pass USB 2.0 host eye-diagram electrical testing. **Figure 20** shows an example of this and how the TUSB211A can be used as a signal conditioner when placed on top of an existing trace.

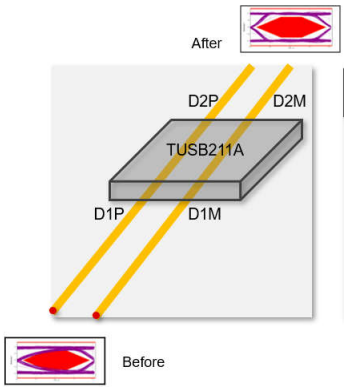


Figure 20. *TUSB211A USB 2.0 signal conditioner*

SuperSpeed Signaling over USB-C

All USB SuperSpeed signals are transmitted and received over the same differential pairs (TX1, RX1, TX2 and RX2). SuperSpeed signaling over USB-C can use either the TX1/RX1 pair or the TX2/RX2 pair; this is also known as an x1 implementation. USB over USB-C adds the ability to enable the unused TX/RX pair from the x1 implementation to effectively double the data throughput without increasing the data rate, which is known as an x2 implementation.

Table 8 shows the different USB 3.0 and USB 4.0 specification versions and associated data rates.

Table 8. SuperSpeed USB data rates

Mode	Name	Speed
USB 3.2 Gen 1x1	SuperSpeed USB	5Gbps
USB 3.2 Gen 2x1	SuperSpeed USB 10Gbps	10Gbps
USB 3.2 Gen 2x2	SuperSpeed USB 20Gbps	20Gbps
USB 4.0 Gen 2 x2	USB 4 20Gbps	20Gbps
USB 4.0 Gen 3 x2	USB4 40Gbps	40Gbps
USB 4.0 Gen 4	USB4 80Gbps	80Gbps

SuperSpeed startup speed negotiation

SuperSpeed devices negotiate to connect at the highest available data rate at startup by using the low-frequency periodic signaling (LFPS) signal in SuperSpeed USB, and by pulse modulating the LFPS signal (LBPM) in SuperSpeed USB 10Gbps and 20Gbps. USB 4.0 uses LBPM signaling as well as one part of a multiphase negotiation. **Figure 21** shows an example of the LBPM negotiation.

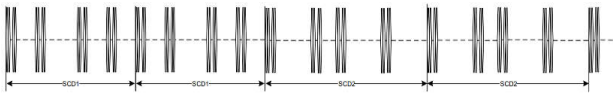


Figure 21. *Example of LBPM*

SuperSpeed signal integrity challenges

The significant increase in data rates poses a challenge on signal integrity as a signal travels across an interface. Many factors contribute to signal integrity degradation. Trace, connectors and cables are all sources of insertion loss, they also cause intersymbol interference, cross talk, noise and jitter for high-speed signals. Additionally, an impedance mismatch for any connection point will cause signal reflection. The higher the signal speed, the more likely the signal will suffer from signal degradation. **Figure 22** shows the list of potential sources that can cause signal degradation.

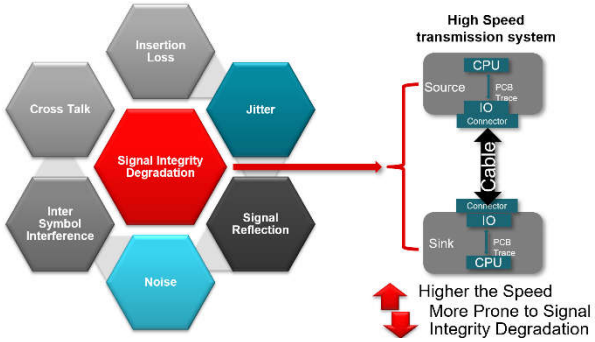
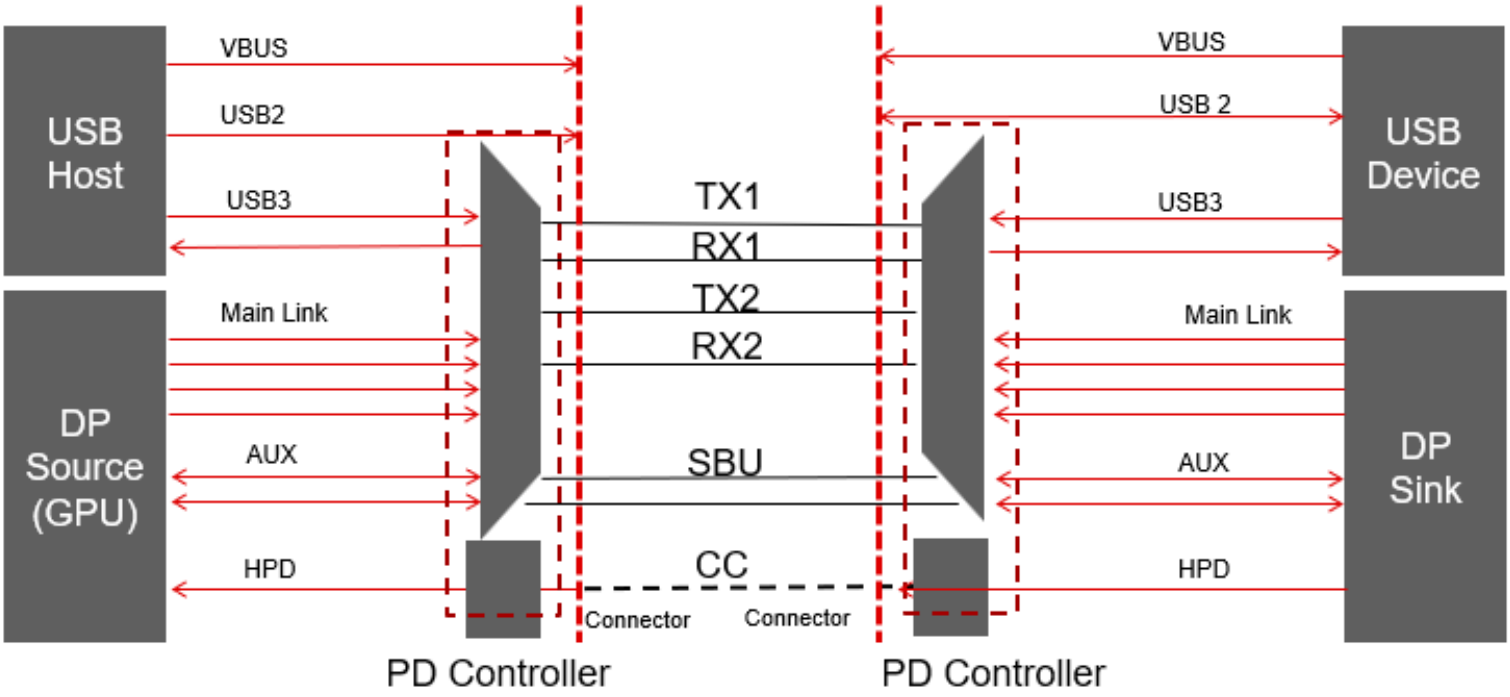


Figure 22. *Signal integrity degradation sources*

In order to overcome these signal integrity challenges, you can use a signal conditioner at the USB-C port. Signal conditioners such as the TUSB1146 linear redriver can compensate for system intersymbol interference while not interfering with link training, as any pre-shoot or de-emphasis passes through.

Signal Multiplexing for USB Type-C®

- USB-C USB 2.0
- USB-C USB 3
- USB PD DisplayPort™ alternate mode multiplexing
- DisplayPort source device (DFP_D) pin assignment C
- DisplayPort source device (DFP_D) pin assignment D
- DisplayPort source device (DFP_D) pin assignment E
- DisplayPort sink device (UFP_D) pin assignment C
- DisplayPort sink device (UFP_D) pin assignment D
- DisplayPort sink device (UFP_D) pin assignment E



USB-C USB 2.0

Author: David Liu

For USB 2.0, due to the positioning of the D+ and D- pins, signal multiplexing is typically handled by shorting together the two D+ signal pins and the two D– signal pins in the host and device receptacles.

USB-C USB 3

In the case of USB 3 or SuperSpeed signals, the data lanes require the functional equivalent of a multiplexer in both the host and device to appropriately route the SuperSpeed transmit (TX) and receive (RX) signal pairs to the connected path through the cable.

To establish the proper routing of an active USB data bus from host to device, the standard USB-C cable is wired such that a single configuration channel (CC) wire is position-aligned with the first USB SuperSpeed signal pairs (SSTXp1/SSTXn1 and SSRXp1/SSRXn1). In this way, the CC wire and USB SuperSpeed data bus wires that are used to determine the orientation and twist of the cable. By detecting which CC pin (CC1 or CC2) at the receptacle the device will terminate, the host can then detect which SuperSpeed USB signals to use for the connection, and control the functional switch for routing the SuperSpeed USB signal pairs.

In the device, detecting which CC pin the host will terminate at the receptacle enables the device to control the functional multiplexer that routes its SuperSpeed USB signal pairs. **Figure 23** shows how the routing of the SuperSpeed transmit (TX) and receive (RX) signal pairs correspond to the USB SuperSpeed signal pairs (SSTXp1/SSTXn1 and SSRXp1/SSRXn1).

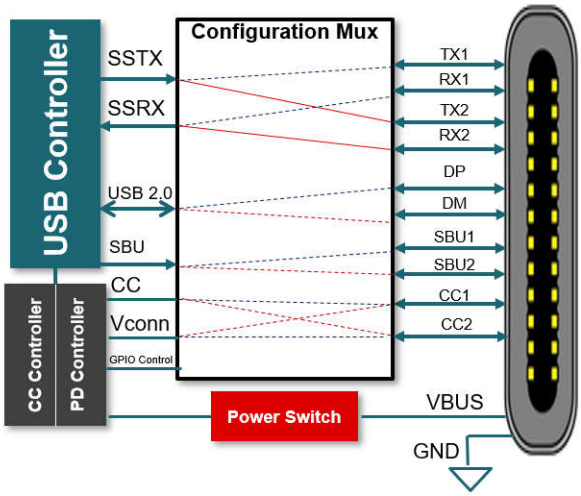


Figure 23. *USB configuration multiplexer*

USB PD DisplayPort™ alternate mode multiplexing

A USB-C port capable of transferring DisplayPort™ signals is either called USB PD DisplayPort or DisplayPort Alternate Mode. DisplayPort Alternate Mode allows you to connect video sources (PCs, Blu-Ray players) and display devices (TVs, monitors) that support DisplayPort to each other through their USB-C ports to broadcast high-definition video. **Figure 24** shows a typical block diagram for the signals used in DisplayPort Alternate Mode.

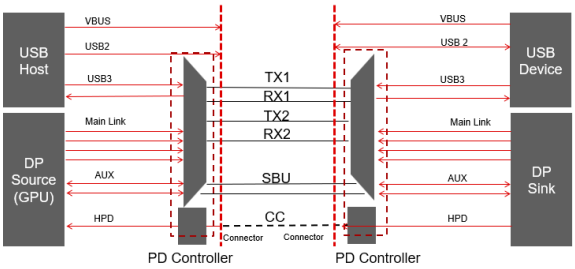


Figure 24. *USB-C DisplayPort*

Since a USB-C port can connect to a USB-C or a DisplayPort connector, there are various pin assignments to properly map the physical connection of the DisplayPort main link lanes to the USB-C connector.

A USB-C source side or DFP_D requires assignment C as shown in **Table 9**, while assignment D in **Table 9** is optional. The difference between assignments C and D is that assignment C supports four DisplayPort lanes, while assignment D supports two DisplayPort lanes. A USB-C port to DisplayPort requires assignment E support.

Table 9. USB-C source-side assignments

DFP_D	Assignment C	Assignment D	Assignment E
	USB-C to USB-C or protocol converter	USB-C to USB-C or protocol converter	USB-C to DisplayPort
USB host	Required	Optional	Required
USB device	Required	Optional	Required

A USB-C sink side or UFP_D requires assignment C as shown in **Table 10**, while assignment D is optional. The difference between assignments C and D is that assignment C supports four DisplayPort lanes, while assignment D supports two DisplayPort lanes. DisplayPort to USB-C port requires assignment E support.

Table 10. USB-C sink-side assignments

UFP_D	Assignment C	Assignment D	Assignment E
	USB-C to USB-C	USB-C to USB-C	USB-C to DisplayPort
USB host	Required	Optional	Required
USB device	Required	Optional	Required

DisplayPort source device (DFP_D) pin assignment C

Figure 25 and **Figure 26** define the configuration of the USB-C connector pins when the pins are reconfigured to support DisplayPort in normal and flipped-plug orientations. DisplayPort electrical mandates for assignment C need to comply with the DisplayPort standard for bit rates up to ultra-high bit rate (UHBR) 20, UHBR 13.5, UHBR 10, HBR 3, HBR 2, HBR or reduced bit rate (RBR), as supported by the system. The protocol is defined in the DisplayPort standard. Pin assignment C is appropriate only for USB-C-to-USB-C passive and active cables and DisplayPort sink devices with USB-C plugs.

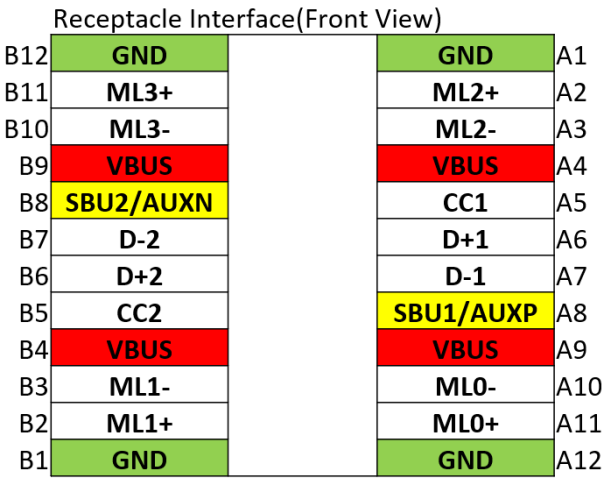


Figure 25. *DFP_D pin assignment C normal plug orientation*

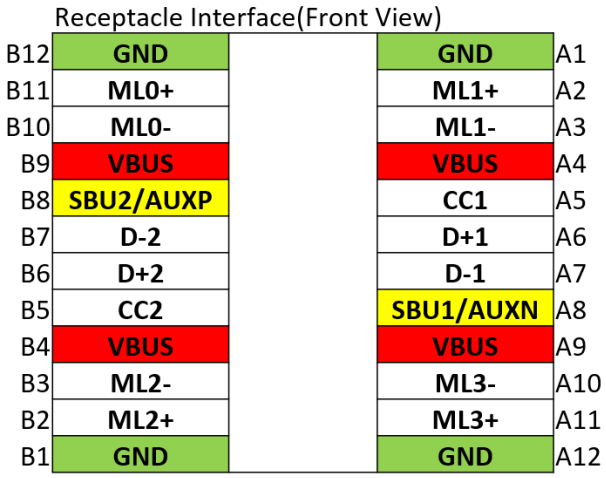


Figure 26. *DFP_D pin assignment C flipped-plug orientation*

DisplayPort source device (DFP_D) pin assignment D

DisplayPort DFP_D pin assignment D is similar to DFP_D assignment C, except that only the two lowest DisplayPort lanes are mapped to the USB-C connector. DisplayPort electrical mandates for pin assignment D need to comply with the DisplayPort standard for bit rates up to UHBR 20, UHBR 13.5, UHBR 10, HBR 3, HBR 2, HBR or RBR, as supported by the system. The protocol is defined in the DisplayPort standard. Pin assignment D is appropriate only for USB-C-to-USB-C passive and active cables and DisplayPort sink devices with USB-C plugs. USB4 does not support assignment D.

DisplayPort source device (DFP_D) pin assignment E

DisplayPort DFP_D pin assignment E is the same as DFP_D assignment C. DisplayPort electrical mandates for pin assignment E need to comply with the DisplayPort standard for bit rates up to UHBR 20, UHBR 13.5, UHBR 10, HBR 3, HBR 2, HBR or RBR, as supported by the system. Pin assignment E is appropriate only for receptacles that accept plugs on USB-C-to-DisplayPort passive and active cables, and the plugs on such cables.

DisplayPort sink device (UFP_D) pin assignment C

For UFP_D pin assignment C, **Figure 27** and **Figure 28** define the use and configuration of the USB-C connector pins when the pins are reconfigured to support DisplayPort in normal and flipped-plug orientations. DisplayPort electrical mandates for pin assignment C need to comply with the DisplayPort standard for bit rates up to UHBR 20, UHBR 13.5, UHBR 10, HBR 3, HBR 2, HBR or RBR, as supported by the system. The UFP_D pin assignment C is appropriate only for USB-C-to-USB-C passive and active cables and DisplayPort source devices with USB-C plugs.

Receptacle Interface(Front View)

B12	GND	GND	A1
B11	ML2+	ML3+	A2
B10	ML2-	ML3-	A3
B9	VBUS	VBUS	A4
B8	SBU2/AUXP	CC1	A5
B7	D-2	D+1	A6
B6	D+2	D-1	A7
B5	CC2	SBU1/AUXN	A8
B4	VBUS	VBUS	A9
B3	ML0-	ML1-	A10
B2	ML0+	ML1+	A11
B1	GND	GND	A12

Figure 27. *UFP_D pin assignment C normal plug orientation*

Receptacle Interface(Front View)

B12	GND	GND	A1
B11	ML3-	ML2-	A2
B10	ML3+	ML2+	A3
B9	VBUS	VBUS	A4
B8	SBU2/AUXP	CC1	A5
B7	D-2	D+1	A6
B6	D+2	D-1	A7
B5	CC2	SBU1/AUXN	A8
B4	VBUS	VBUS	A9
B3	ML1+	ML0+	A10
B2	ML1-	ML0-	A11
B1	GND	GND	A12

Figure 28. *UFP_D pin assignment C flipped-plug orientation*

DisplayPort sink device (UFP_D) pin assignment D

DisplayPort UFP_D pin assignment D is similar to UFP_D assignment C, except only the two lowest DisplayPort lanes are mapped to the USB-C connector. DisplayPort electrical mandates for pin assignment D need to comply with the DisplayPort standard for bit rates up to UHBR 20, UHBR 13.5, UHBR 10, HBR 3, HBR 2, HBR or RBR, as supported by the system. The protocol is defined in the DisplayPort standard. Pin assignment D is appropriate only for USB-C-to-USB-C passive and active cables and DisplayPort sink devices with USB-C plugs. USB4 does not support UFP_D assignment D.

DisplayPort sink device (UFP_D) pin assignment E

For UFP_D pin assignment E, **Figure 29** and **Figure 30** define the use and configuration of the USB-C connector pins when the pins are reconfigured to support DisplayPort in normal and flipped-plug orientations. DisplayPort electrical mandates for pin assignment E need to comply with the DisplayPort standard for bit rates up to UHBR 20, UHBR 13.5, UHBR 10, HBR 3, HBR 2, HBR or RBR, as supported by the system. Pin assignment E is appropriate only for receptacles that accept plugs on USB-C-to-DisplayPort passive and active cables and the plugs on such cables. For assignment E, the main link lane order, polarity and auxiliary polarity are swapped from assignment C.

Receptacle Interface(Front View)

B12	GND	GND	A1
B11	ML0-	ML1-	A2
B10	ML0+	ML1+	A3
B9	VBUS	VBUS	A4
B8	SBU2/AUXN	CC1	A5
B7	D-2	D+1	A6
B6	D+2	D-1	A7
B5	CC2	SBU1/AUXP	A8
B4	VBUS	VBUS	A9
B3	ML2+	ML3+	A10
B2	ML2-	ML3-	A11
B1	GND	GND	A12

Figure 29. *UFP_D pin assignment E normal plug orientation*

Receptacle Interface(Front View)

B12	GND	GND	A1
B11	ML3-	ML2-	A2
B10	ML3+	ML2+	A3
B9	VBUS	VBUS	A4
B8	SBU2/AUXP	CC1	A5
B7	D-2	D+1	A6
B6	D+2	D-1	A7
B5	CC2	SBU1/AUXN	A8
B4	VBUS	VBUS	A9
B3	ML1+	ML0+	A10
B2	ML1-	ML0-	A11
B1	GND	GND	A12

Figure 30. *UFP_D pin assignment E flipped-plug orientation*

USB4

- USB4 Overview
- USB4 discover and entry process
- USB4 System
- Sideband Communication
- USB4 lanes and data rates
- Loss Budget
- Supporting DisplayPort Alternate Mode and USB4 over SBU1 and SBU2

USB4 Overview

Author: Mike Campbell

The USB4 standard enables greater bandwidth than previous USB generations. Using the USB Type-C® (USB-C®) interface, USB4 version 1, released in 2019, increases the aggregated bandwidth to 40Gbps from the 20Gbps defined in the USB 3.2 specification. In 2022, the USB Implementers Forum (USB-IF) released USB4 version 2, enabling another speed step to 80Gbps for symmetric operation and 120Gbps for asymmetric operation. With such high bandwidth, USB4 offers the ability to tunnel multiple, independent protocols such as Peripheral Component Internconnect Express (PCIe), USB3 and DisplayPort™ over the same physical interface. Supported only on USB-C, USB4 is backward compatible with the USB2 and USB3.2 standards, as well as being backward compatible to Thunderbolt 3, enabling users to use existing products.

USB4 discover and entry process

USB4 is significantly different than previous USB generations. In a USB-C system, a USB2 or USB3.2 product operates without the need for USB Power Delivery (PD). For example, if you plug in a USB2 thumb drive into a non-USB PD-enabled USB-C port, you will be able to use the drive just like you could if the drive was plugged into a USB Type-A receptacle.

Taking advantage of all of the capabilities of a USB4 product requires USB PD. During the discovery process, if both port partners and as well as the cable support USB4, then the downstream-facing port (DFP) issues the USB PD Enter_USB message to the cable and port partner. If a USB4 product is inserted into a USB-C port that doesn't support USB PD, then the USB4 product operates in a legacy USB mode (either USB2 or USB3.2).

USB4 System

The USB4 system comprises a host, hub device, peripheral device and dock, each of which contain a router. The router maps tunneled protocol traffic to USB4 packets and routes packets through the USB4 fabric. **Figure 31** shows the block diagrams for a USB4 host, hub, and device.

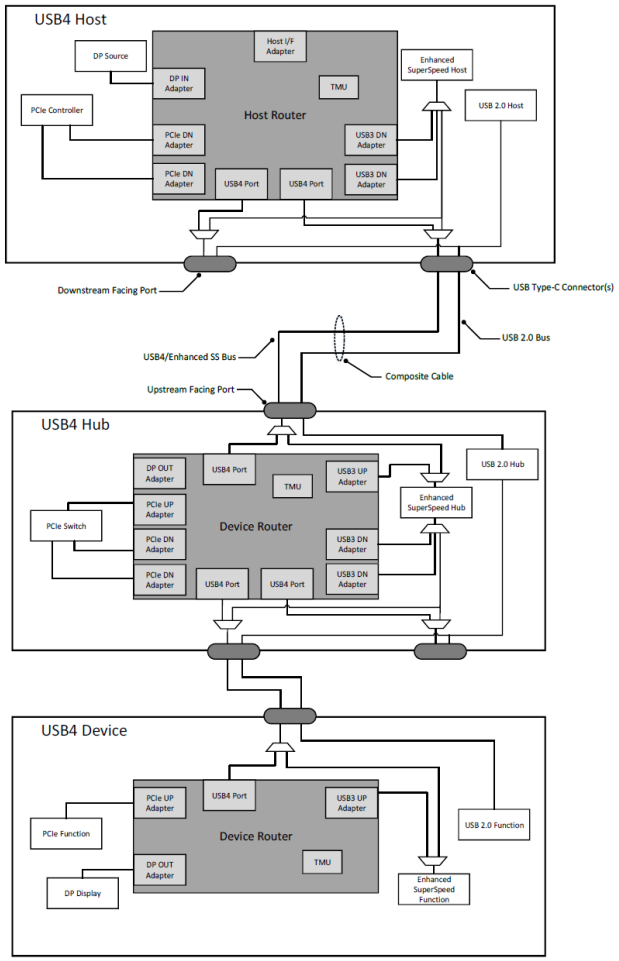


Figure 31. *USB4 Host, Hub, and Device*

The host contains the host router, an USB host controller (both USB3.2 and USB2), and a DisplayPort source. The host may have more than one DFP. The host may optionally incorporate a PCIe controller or PCIe switch if PCIe tunneling is supported. Residing in a host is the connection manager software, which enumerates, configures and manages the entire USB4 domain. **Figure 32** takes a closer look at a USB4 hub.

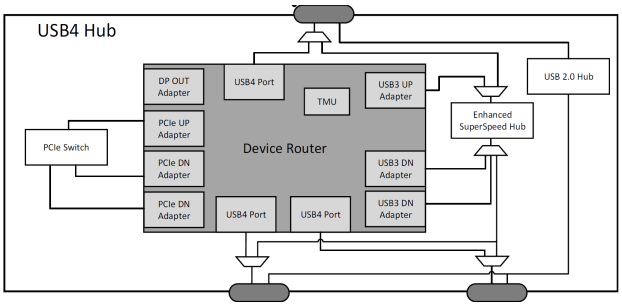


Figure 32. *USB4 Hub*

A hub has a single upstream-facing port (UFP) and, like a host, can have more than one DFP. A hub router supports USB2, USB3, a PCIe switch and DisplayPort™ tunneling. **Figure 33** takes a closer look at a USB4 peripheral device.

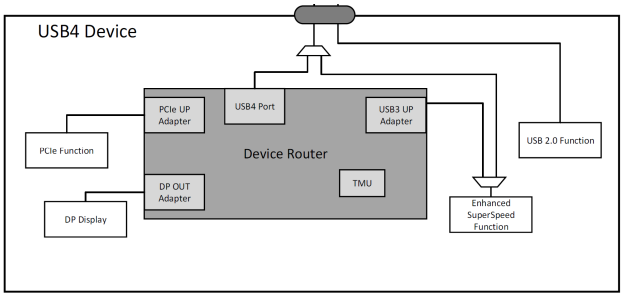


Figure 33. *USB4 Peripheral Device*

A peripheral device has a single UFP and no DFP, and may optionally contain one or more enhanced SuperSpeed hubs, enhanced SuperSpeed functionality, PCIe switches or endpoints, or DisplayPort source or sink functionality. **Figure 34** takes a closer look at a USB4 dock

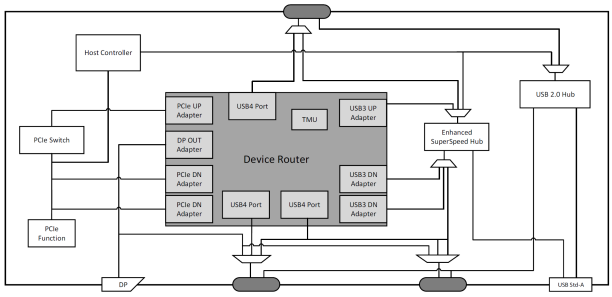


Figure 34. *USB4 Dock*

A USB4 dock is similar to a USB4 hub with a single UFP and one or more DFPs. Unlike a USB4 hub, the dock contains one or more protocol adapters such as DisplayPort.

All USB4 components connect using USB-C passive or active cables in a spanning tree with the host at the top, a hub in the middle and a peripheral device at end of tree. There can be as many as five hubs between the host and the peripheral router. All of these products are configured using a sideband channel.

Sideband Communication

Sideband communication is performed using a 3.3V low-voltage complementary metal-oxide semiconductor (LVCMOS)-level 1Mbps universal asynchronous universal transmitter (UART) comprising 10 bits: 1 start bit, 8 bits of data payload and 1 stop bit. Sideband transactions are transmitted over the SBTX pin and received from the SBRX pin and use the USB-C receptacle's SBU1 and SBU2 pins. Sideband communication is used for these purposes:

- Whether or not a port is connected (SBRX high > 25µs) or disconnected (SBRX low > 14µs).
- To identify router manufacturer and product information.
- To configure parameters of the USB4 link such as data rate (Gen2, Gen3, Gen4) and to enable or disable lanes.
- Asymmetric support, initiation and decision.
- Transmitter feed-forward equalization link-training handshake and whether or not routers are locked.

USB4 lanes and data rates

USB4 Gen2, Gen3 and Gen4 support up to two lanes (lane 0 and lane 1), where each lane comprises a transmit (TX) and receive (RX) path. USB4 version 2 allows asymmetric operation when operating at USB4 Gen4. In asymmetric operation, lane 0 remains unchanged, but lane 1 is split into two lanes (lane 1 and lane 2), where both lanes can be configured as either TX or RX. A USB4 Gen4 port operating in asymmetric mode having one TX and three RX or three TX and one RX produces a data rate of 120Gbps. **Table 11** summarizes all of the USB4 generations and their corresponding lane and data rate capabilities.

Table 11. USB4 data rates

	Enabled lanes	Total data rate (Gbps)
USB4 Gen2	1	10
	2	20
USB4 Gen3	1	20
	2	40
USB4 Gen 4 symmetric	2	80
USB4 Gen4 asymmetric	3	120

Loss Budget

All USB4 products from the router to the USB-C receptacle must keep their insertion loss within loss budget defined by the specification. The budget for USB4 is detailed in the table below.

Table 12. USB Type-C Insertion Loss Budget for USB4

	Host (dB)	Cable (dB)	Device (dB)	Total
USB4 Gen 2	5.5	12	5.5	23dB at 5GHz
USB4 Gen 3	7.5	7.5	7.5	23dB at 10GHz
USB4 Gen 4	9.5	9.5	9.5	28.5dB at 12.8GHz

(1) USB4 Gen2 supports up to 2m of USB-C passive cable. USB4 Gen3 and USB4 support up to 0.8m of passive USB-C cable.

In some cases, the insertion loss between router and USB-C receptacle is too large and a signal conditioner, called a retimer (RT), is needed to overcome the deficiency. The USB4 specification allows up to two retimers between the router and the USB-C receptacle. Instead of a second retimer, the USB4 specification allows the use of a linear redriver (LRD) between the router and retimer near USB-C receptacle. A few examples of different use cases are shown in **Figure 35**.

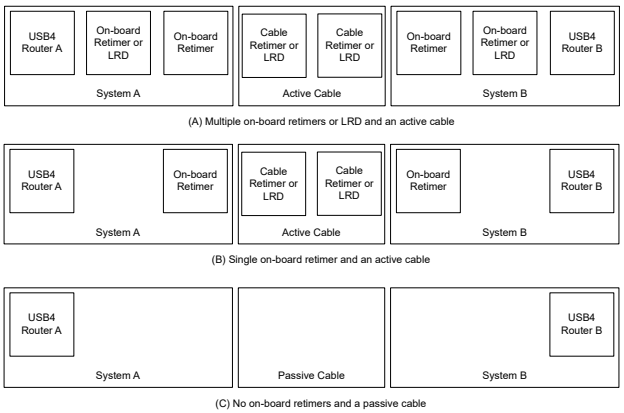


Figure 35. *Possible System and cable uses cases*

Uses cases (A) and (B) show the use of a RT or LRD in a system and in an active cable. Use case (C) shows no signal conditioners in the system or cable.

The use of a signal conditioner like a RT or LRD in a cable is intended to extend the Type-C cable to lengths that are not achievable using only passive components like copper wire. This type of cable is called an active cable and is defined in the USB-C specification.

Supporting DisplayPort Alternate Mode and USB4 over SBU1 and SBU2

A USB4 host must be able to support legacy USB products as well as DisplayPort Alternate Mode products such as USB-C docking stations. In order to support both types of products, the host needs to multiplex both USB4 sideband (SBTX and SBTX) and DisplayPort sideband (AUXP and AUXP) over the USB-C SBU1 and SBU2 pins. **Figure 36** shows an example of how to multiplex USB4 and DisplayPort sideband signals over SBU1 and SBU2.

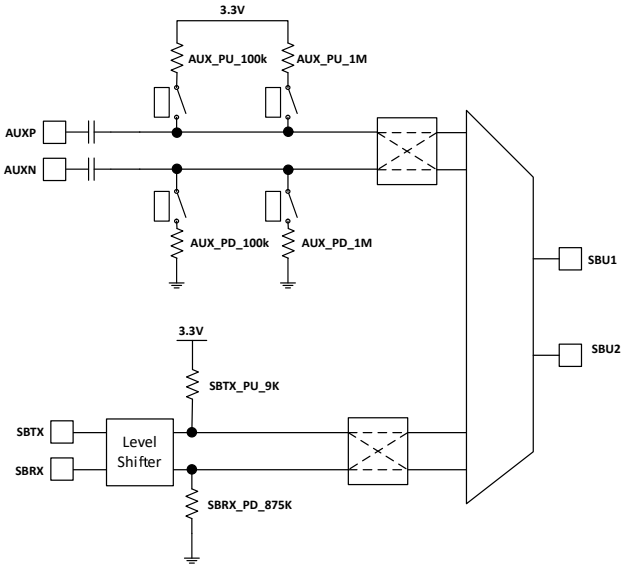


Figure 36. *USB4 and DisplayPort sideband multiplexing*

Figure 36 shows the auxiliary biasing required by the DisplayPort specification and the necessary SBTX pullup resistor and SBRX pulldown resistor required by USB4 specification. **Table 13** lists these resistors for reference. The level shifter on SBRX and SBTX may be necessary if the USB4 router or retimer doesn't support the 3.3V LVCMOS levels present at the USB-C receptacle. Some or all of the components in **Figure 36** may be integrated in a retimer or a discrete sideband multiplexer.

Table 13. USB4 and DisplayPort pullup and pulldown resistors

Parameter	Minimum (kΩ)	Maximum (kΩ)
SBTX pullup resistor	7	10.5
SBRX pulldown resistor	700	1050
DisplayPort source AUXP pulldown resistor	10	105
DisplayPort source AUXN pullup resistor	10	105
DisplayPort sink AUXP pullup resistor	800	1,200
DisplayPort sink AUXN pulldown resistor	800	1,200

Introduction to eUSB2

Abstract

Author: Nicholaus Malone

As discussed in previous chapters, the USB 3.2 and 4.0 specifications allow for high-speed data transfer through the USB bus. The SuperSpeed lanes on a USB Type-C® (USB-C®) connector support these newer data rates; however, the D+ and D– lanes on a USB-C connector still support the original USB 2.0 specification released in 2000.

Despite having a maximum speed of 480Mbps, the reliability and interoperability with many USB devices make USB 2.0 one of the most popular interfaces in the world. USB 2.0 also has the highest signaling levels of all modern interfaces with 3.3V amplitudes, which can cause technical challenges in some modern applications. The higher voltage can damage the metal oxide in technology nodes where oxide dielectric thickness is 7nm and below. Thus, in 2014 the USB Implementers Forum released an embedded USB 2.0 (eUSB2) Physical Layer (PHY) Supplement to the USB Revision 2.0 Specification to address this problem. The eUSB2 specification enables USB 2.0 communication using low-voltage signaling in order to avoid damaging sensitive components.

eUSB2 overview

The creation of the eUSB2 specification introduced a number of features not available in traditional USB 2.0:

- Process scalability provides a low-voltage USB 2.0 PHY solution to eliminate 3.3V input/output (I/O) signaling, allowing USB 2.0 to scale with decreasing process technology nodes.
- I/O power efficiency to improve link active and idle power efficiency.
- Reduced PHY analog content and employing digital mechanisms for PHY functionality.
- Support for USB 2.0 devices – although eUSB2 and USB 2.0 are not electrically compatible, the specification defines a mechanism to enable USB 2.0 support.

eUSB2 is a PHY supplement only, which means that eUSB2 has different electrical parameters than USB 2.0. They do both share the same protocol layer specification. The eUSB2 packet structure is the same as USB 2.0, and supports the same standard data rates of low speed, full speed and high speed. However, as mentioned previously, the signaling voltage levels are lower. Low- and full-speed communication drops from 3.3V to either 1.2V or 1.0V, and the high-speed differential swing is half of USB 2.0. The similarity between USB 2.0 and eUSB2 interfaces facilitates implementation simplicity, but the different signaling levels create an interoperability problem with conventional USB 2.0 devices. In native mode, an eUSB2 device can only communicate with another eUSB2 device.

eUSB2 modes

There are two primary modes of communication between eUSB2 devices: native mode and repeater mode. In eUSB2, the two channels that carry electrical data signals are eD+ and eD–. The eD+ and eD– channels of the host are connected directly to the transceiver of the eUSB2 peripheral device while in native mode. The two eUSB2 transceivers communicate directly with each other as shown in **Figure 37**.

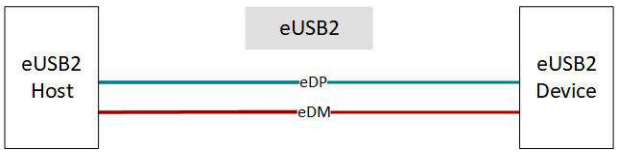


Figure 37. *Native mode*

Native mode is primarily used for interchip interconnect and does not support interoperability with USB 2.0 devices. If support for a USB 2.0 communication is necessary, the eUSB2 specification defines an eUSB2 repeater to enable backward compatibility.

An eUSB2 repeater is a device that “translates” between eUSB2 and USB signaling. It enables communication between eUSB2 and USB 2.0 devices. Unlike USB 2.0, the signaling for eUSB2 low- and full-speed communication is single-ended. The switch from differential to single-ended signaling means that an eUSB2 repeater is not simply a 3.3V to 1.2V voltage level shifter.

An eUSB2 repeater must be configurable as either a host repeater or a peripheral repeater. Configuring a repeater establishes its role in the USB architecture. There is only one eUSB2 host allowed per USB bus. If an eUSB2 repeater is connected to the host via the eUSB2 interface, it is configured by the host as a host repeater as shown in **Figure 38**. After configuration, the eUSB2 host can communicate to downstream USB 2.0 devices connected to the repeater.

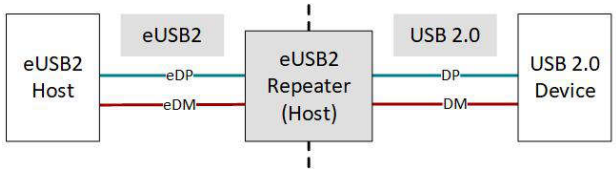


Figure 38. *Host repeater mode*

Unlike the USB host, there may be many peripherals in a USB bus. A eUSB2 repeater must be configured as a peripheral repeater for each eUSB2 peripheral to communicate to a USB 2.0 host, as shown in **Figure 39**.

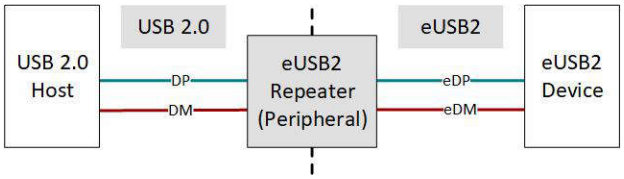


Figure 39. *Peripheral repeater mode*

The ability to dynamically be configured as either a host or peripheral repeater allows eUSB2 repeaters to be used in dual-role applications where the attached eUSB2 device can act as either a host or peripheral. This is dual-role capability is commonly used in mobile applications.

It is also possible to have both a eUSB2 host and eUSB2 peripheral in a single USB 2.0 link. A repeater may be required for both the host and peripheral in applications that require a USB cable. The eUSB2 interface is designed for interchip interconnect, and not compatible with USB cables. Assuming both the host and peripheral are eUSB2-capable, this can result in a back-to-back repeater application as shown in **Figure 40**.

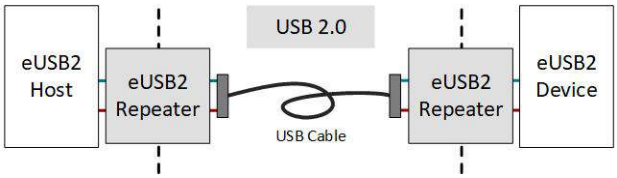


Figure 40. *Back-to-back repeater application*

This is most common in mobile electronics, where a laptop, mobile phone, or tablets are commonly dual-role capable. A repeater enables both the eUSB2 host and the eUSB2 peripheral to communicate to each other using the USB 2.0 interface over a USB cable.

Other features

An eUSB2 repeater must respond to control messages, which are eUSB2 signal patterns that control the repeater state machine after configuration. With the notable exception of CM.RAP, control messages are not used in native mode. They are not meant to be forwarded to the USB 2.0 bus and are meant to be received and interpreted only by an eUSB2 repeater or peripheral.

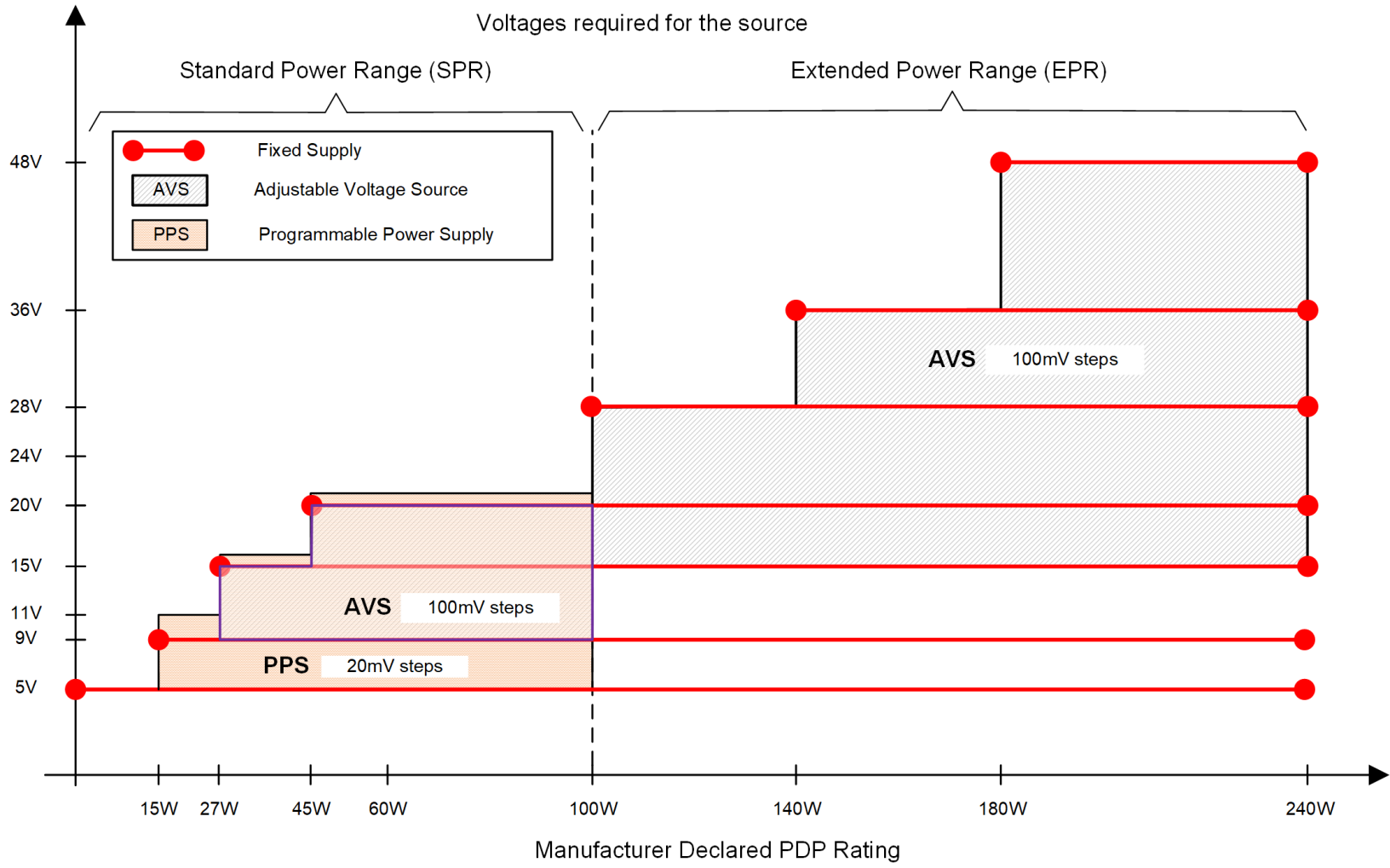
Register Access Protocol (RAP) is an optional feature introduced in eUSB2 to further simplify implementation. CM.RAP, shown in Table 1, is the control message used to access the RAP interface. It's common to use the I^2C interface for configuration. In lieu of implementing a separate I^2C interface, RAP can access the register space of an eUSB2 repeater or peripheral directly through eUSB2. This vendor-defined register space can be modified by issuing the CM.RAP command, along with the operand and target register address. RAP enables system designers to exclude components and general-purpose I/O pins typically required for I^2C communication to reduce system cost and footprint.

Table 14. eUSB2 control messages

Control message name	Description
CM.FS	Revert to full-speed terminations
CM.L1	Enter L1 power state
CM.L2	Enter L2 power state
CM.Reset	USB 2.0 bus reset
CM.Test	Force-enable high-speed terminations
CM.RAP	Start of RAP access

Although it is still a relatively new interface, eUSB2 hosts, devices and repeaters are only expected to grow as modern processor technology nodes continue to shrink below 7nm.

Extended Power Range (EPR)



Abstract

Authors: Taylor Vogt, Adam McGaffin

In this section, we'll highlight some of the latest developments to enable higher-power applications using the USB-C® interface. Added to the USB Power Delivery (PD) 3.1 USB Implementers Forum (IF) specification[CCH1] in 2021, Extended Power Range (EPR) increases the maximum power for a USB-C connector from 100W to 240W. We'll first touch on the basics of EPR and how it differs from standard USB-C, and then detail its technical specifications and power negotiation sequence

What is EPR?

Until recently, the USB PD 3.0 specification allowed both power and data in both directions at levels as high as 100W (20V, 5A) across an approved USB-C port and cable. The latest USB PD 3.1 specification increases this limit up to 240W (48V/5A) through a USB-C cable.

The previous USB PD range was Standard Power Range (SPR), while the new power range (between 100W and 240W) is EPR. This translates to a hefty improvement, enabling power-hungry devices to implement the USB-C interface. There is also the potential to reduce the number of heavy power adapter bricks and migrate to one single USB-C cable. **Table 15** shows how the new EPR compares to the original SPR.

Table 15. SPR and EPR fixed voltage ranges

Power Range	Available Current and Voltages	PDP Range	Notes
Standard Power Range (SPR)	3A: 5V, 9V, 15V, 20V 5A(1): 20V	15 - 60W >60 - 100W	
Extended Power Range (EPR)	3A(2): 5V, 9V, 15V, 20V 5A(2): 20V 5A(2): 28V, 36V, 28V	15 - 60W >60 - 100W >100 - 240W	Requires entry into EPR Mode.

(1) Requires 5A cable.
(2) Requires EPR cable.

Technical specifications

EPR allows support for up to 240W of power (28V, 36V and 48V at 5A). EPR mode has the same requirement for sink devices; to evaluate and respond to a new source capabilities messages as in a typical USB PD contract negotiation. After entering EPR mode, a port negotiates a power delivery object (PDO) of up to 240W (48V, 5A). This 48V represents a practical limit when considering design safety margins.

On top of a fixed voltage-level extension in the extended range, the supply must also follow the specifications for an adjustable voltage supply (AVS). Within EPR mode, an AVS enables the sink to fine-tune the optimum voltage between 9V and 48V in steps of 100mV for improved performance and thermal efficiency, giving the sink flexibility to receive voltages from any charger. With this, designers can avoid custom adapters and create a consistent user experience with their electronic ecosystem. A programmable power supply (PPS) is similar to an AVS but has 20mV steps between 5V and 20V. While the two are similar, they are not intended for the same use cases. A PPS requires a current limit (50mA steps, ±5%), whereas a USB-IF certified charger does not need to include a PPS. The USB-IF certifies a "fast charger" with a PPS and a "charger" without a PPS.

Figure 41 shows the relationship between SPR and EPR power levels as well as how AVS and PPS overlap with these power ranges.

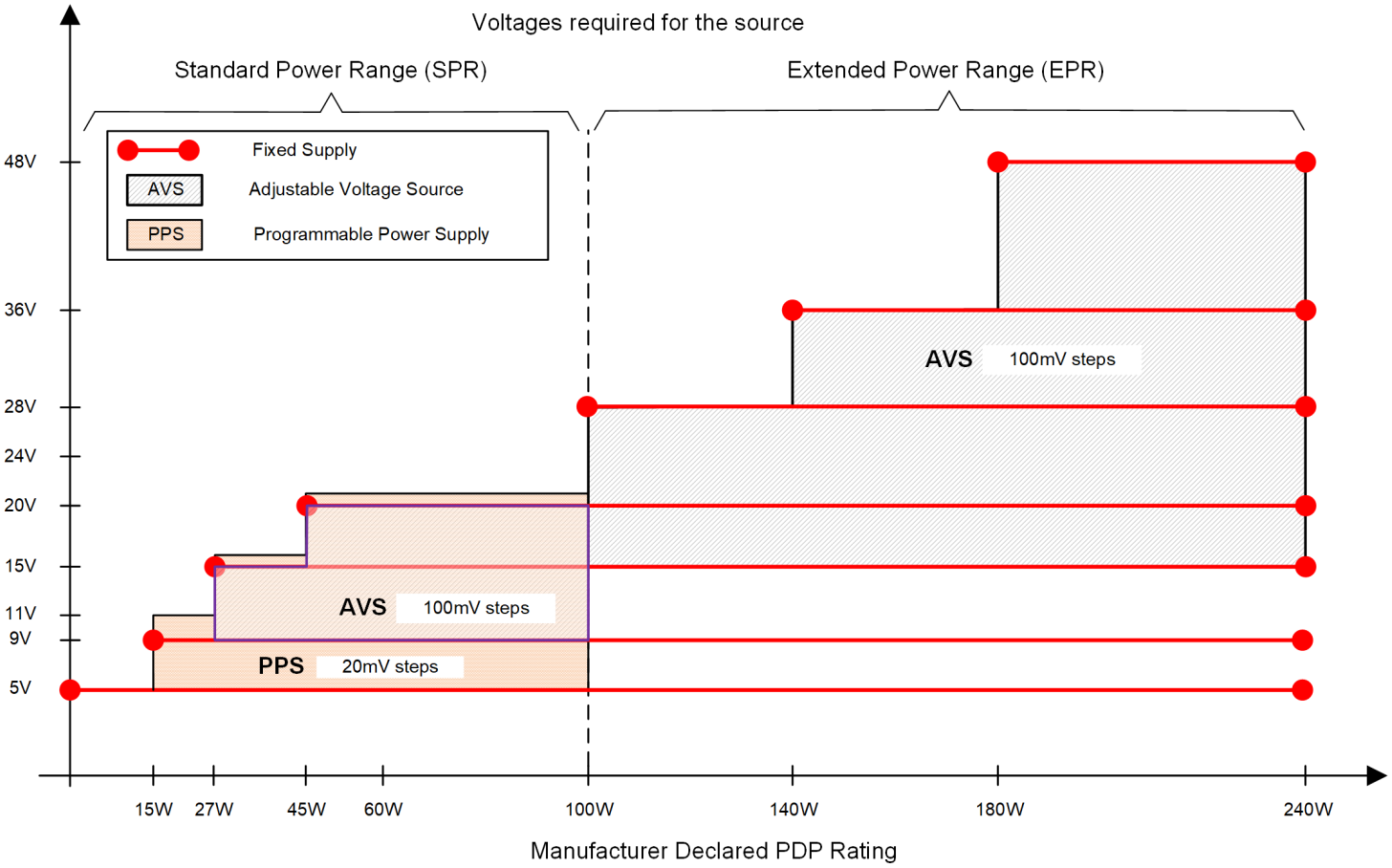


Figure 41. *AVS and PPS ranges for SPR and EPR*

Safety implications >100W

Beyond the supply and source requirements, there are additional mechanisms to meet safety standards when using >100W applications. Specifically, there is an additional requirement on the cable to support EPR. It is no longer optional to have the EPR-compatible cables visibly marked using the logos shown in **Figure 42**; instead, consumers can identify cable supports up to 240W safely through the logo on the cable itself. Typically, this means that the cable design may include higher-voltage-rated bypass capacitors and additional snubber circuits to reduce transients and ringing that may be more likely to occur at the higher voltages. It also means that 5A-rated cables will become obsolete; eventually, the USB-IF will no longer certify the original 20V/5A cables. This decision is will move the market to a state where all 5A cables are EPR-capable, and reduce consumer confusion around cabling issues. The cables are backward-compatible with older USB ecosystem devices, assuming the use of a USB-IF-certified USB PD controller.

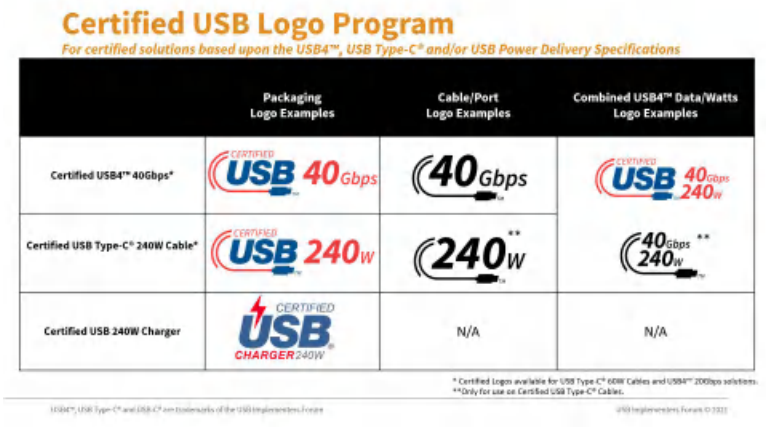


Figure 42. *Certified USB Logo Program*

With >100W applications, it's important to highlight the restrictions for entering EPR mode before accepting the contract. The purpose of the new handshake guarantees that a single mistake in the handshake cannot lead to an EPR contract (in order to satisfy safety regulators). This new handshake ensures that legacy sinks cannot accidentally request an EPR contract.

Handling power negotiation with TI's PD controllers

The EPR power negotiation is similar to today's USB PD sequence in terms of offer, request, and accept or reject. However, you must first enter into an SPR contract before entering into an EPR contract. The new EPR negotiation process includes discovering whether the cable is EPR-capable and whether the source can deliver these extended power levels. **Figure 43** shows an example of the general contract handshake flow.

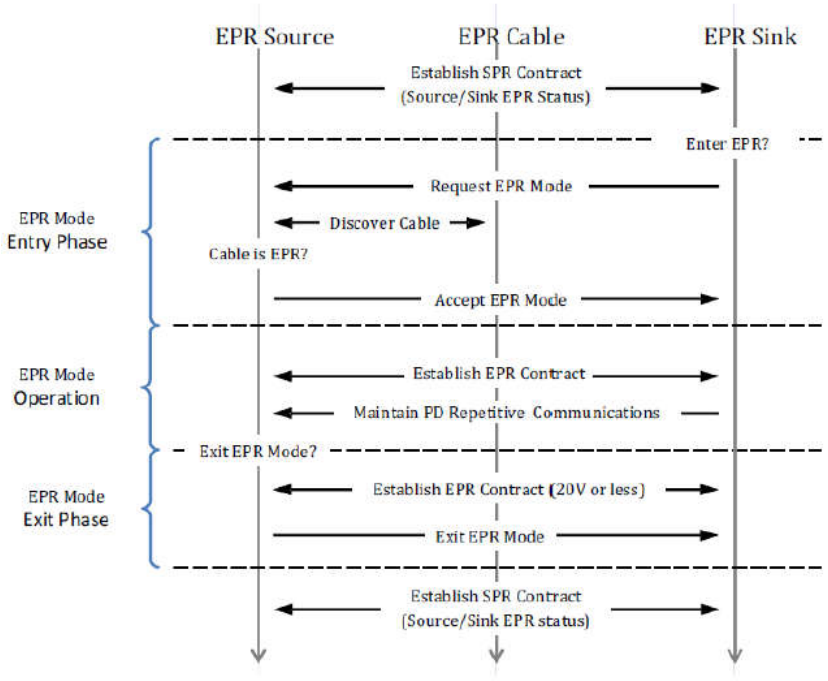


Figure 43. *EPR negotiation sequence*

The sink side drives the entry process into EPR mode. The source side cannot force EPR entry if the sink does not first request and lead entry into this mode. Afterward, the source verifies that the sink and cable are EPR-capable before sourcing any extended voltages. To verify this, the EPR request message from the sink side includes a copy of PDO to verify its capabilities and that the cable and sink are EPR capable. After entering EPR mode, the sink must continually send “keep alive” messages roughly every half second to confirm that both sides are open and active. If the source does not receive this message within about a second, the source initiates a hard reset, and then drops into the default SPR mode at 5V operation.

TI worked closely with the USB-IF to ensure that this new technology is safe. To help validate the safety for the new EPR negotiation process, several functional safety experts independently audited the 240W functionality. Most importantly, a source cannot offer the voltages available in EPR until entering EPR mode. EPR requires using newly defined USB PD messages in a specific sequence in order to successfully negotiate a USB PD contract above 20V, ensuring that no single messaging error could result in negotiating a voltage greater than 20V.

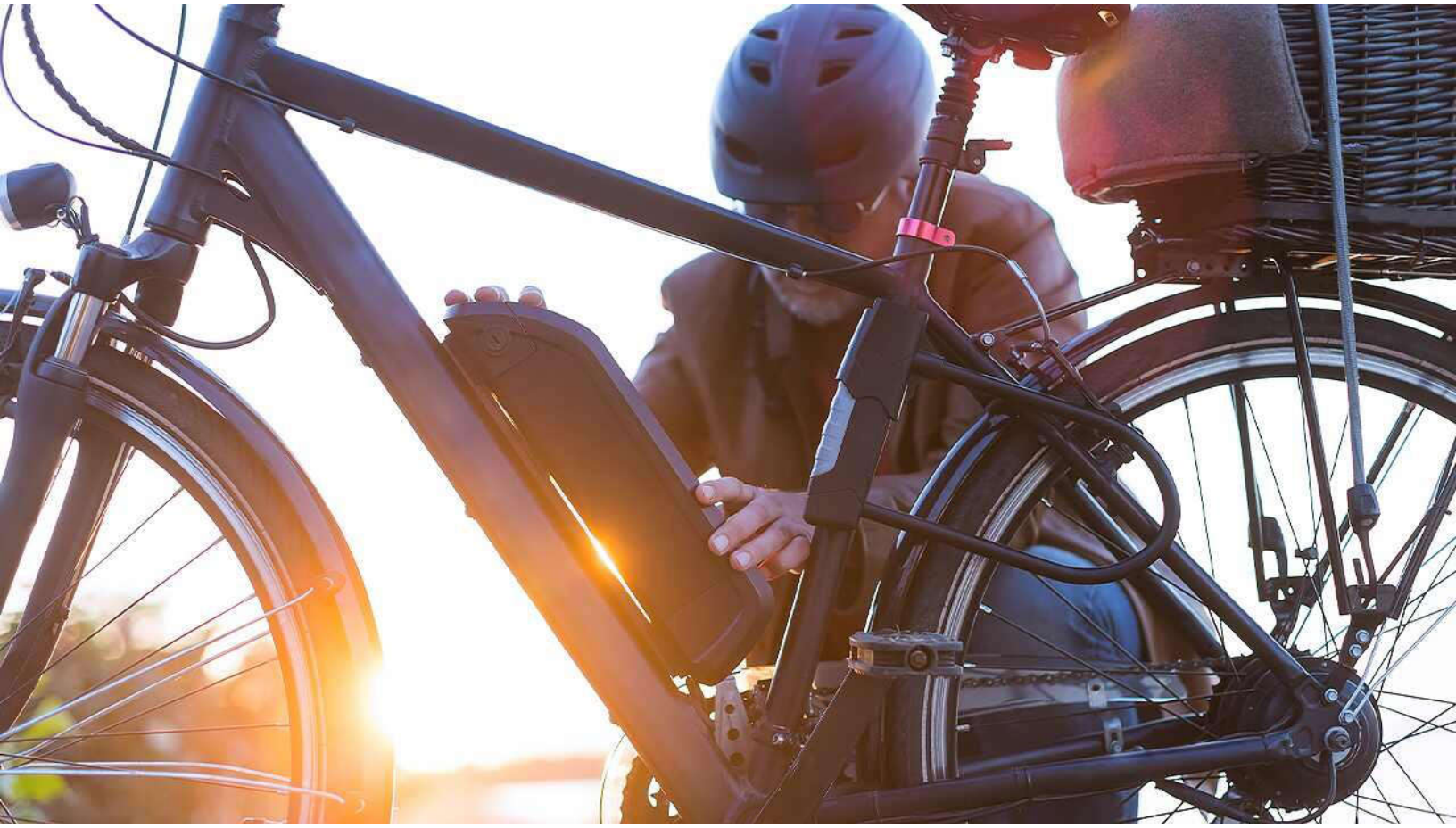
For additional details on the new EPR negotiation sequence, please review the **USB-PD Specification**.

Conclusion

EPR is a defining new development within the USB-C ecosystem. These extended voltage levels allow for higher-power applications, which will broaden the use cases for USB-C interfaces. There are some challenges in terms of new cabling specifications and power path design, but ultimately the flexibility and backward compatibility should pave a strong path to universalize the USB-C interface.

USB Type-C® and USB power delivery common use cases and block diagrams

- 5V USB-C source-only port (no USB PD)
- Basic functional blocks
- 5V USB-C source-only port with USB 3.0 data (no USB PD)
- 5V USB-C sink-only port (no USB PD)
- 5V USB-C DRP (no USB PD)
- 20V USB-C source-only port with USB PD
- 20V USB-C sink-only port with USB PD
- 5V source, 20V sink USB-C port with USB PD and DisplayPort™ Alternate Mode
- 20V USB-C DRP with USB PD and a battery charger



5V USB-C source-only port (no USB PD)

Authors: Eric Beljaars, Adam McGaffin

When converting a legacy USB Type-A port to a USB-C port, the product gains the benefits of a slim, bidirectional connector, as well as more power output. A standard USB-C source port can support up to 15W (5V at 3A). But there are also several solutions offering 5V at 1.5A capabilities.

Basic functional blocks

Presenting resistor pullups or pulldowns on the configuration channel (CC) lines establishes a USB-C connection. A USB-C source-only port requires a pullup resistor on the CC line, known as Rp. The value of Rp will change depending on how much current you would like to advertise. The most common current levels supported by a USB-C source-only port are default USB power (500mA for USB2 and 900mA for USB3), 1.5A and 3A. The USB Type-C specification's Termination Parameters section lists the corresponding Rp resistor values for each of these current values.

In general, when designing a system for a 5V USB-C source-only port, a CC controller IC will ensure the presentation of the correct value of Rp on the CC line automatically.

In addition to presenting Rp on the CC line, a 5V USB-C source-only port will also need to be able to protect against noncompliant sink devices that draw more current than negotiated by the Rp. For example, when presenting a 3A Rp, the connected sink device must ensure that its current draw does not exceed 3A. Although the sink is responsible for ensuring that it does not exceed the negotiated current level, the source is actually responsible for implementing overcurrent protection in the event that a noncompliant sink device draws more current than negotiated.

There are several ways to implement this current limit. For example, you could design a discrete power path with some form of current measurement, or use a load switch with an integrated current limit. The simplest way is to use a CC controller with an integrated power path, which would ensure automatic implementation of the current limit based on the value of Rp presented.

Implementing a 5V USB-C source-only port requires two key blocks: a CC controller and a 5V load switch. See **Figure 44**.

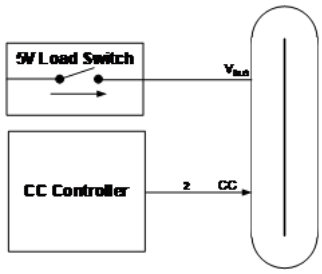


Figure 44. *5V source only block diagram*

5V USB-C source-only port with USB 3.0 data (no USB PD)

Implementing USB 3.0 data rates on a 5V USB-C source-only port introduces the need for an additional system block, a USB 3.0 multiplexer. USB 3.0 uses the SSTX and SSRX pins of the USB-C connector. In order to route the correct pins to the USB 3.0 physical layer (PHY) within the system, a multiplexer controlled by a CC controller handles the

bidirectional flip of the USB-C connector. CC controllers will have a general-purpose input/output (GPIO) for indicating the CC polarity. This GPIO output from the CC controller connects to the multiplexer polarity control input.

In addition to requiring a multiplexer, a USB-C port supporting USB 3.0 data is also required to support V_{CONN}, since USB 3.0 cables typically include an e-marker or active redriver, which are powered by V_{CONN}. Note that V_{CONN} is provided on the unused CC pin. Although it is possible to implement V_{CONN} discretely through another load switch or power path, the simplest way to add V_{CONN} capabilities is to use a CC controller with an integrated V_{CONN} power path, as illustrated in **Figure 45**.

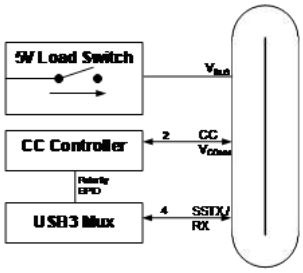


Figure 45. *5V source with USB3 mux block diagram*

When supporting USB 3.0 data in any direction, the requirements for supporting V_{CONN} and including a USB 3.0 multiplexer remain. The type of USB 3.0 multiplexer will vary, depending on whether you need an upward-facing port or downward-facing port. However, the GPIO interface between the CC controller and the USB 3.0 multiplexer remain the same (single GPIO to indicate polarity). Because these hierarchical requirements are identical, the remaining sections on 5V USB-C source-only ports without USB PD will focus on the differences in power architectures.

5V USB-C sink-only port (no USB PD)

Implementing a USB-C port for sinking power up to 15W can look quite a bit different from a source only solution, depending on the requirements of the system onto which the USB-C port is going. Typically, if you're implementing a USB-C sink-only port, you will probably power the entire system off of the power coming into that USB-C port.

When implementing a 5V USB-C sink-only port, you can in some cases get away with just two Rd resistors on the CC lines. In other cases, the required blocks look a lot like the required blocks on the 5V USB-C source-only port implementation. The two variables that most affect the solution are the amount of capacitance on V_{BUS} and whether you need to support USB 3.0 data.

In its simplest state, you can enable a 5V USB-C sink-only port by tying Rd resistors (5.1kΩ) to each of the CC pins on the USB-C connector. This is only possible, however, if the amount of capacitance you will have on V_{BUS} is less than 10µF. If your system exceeds 10µF of capacitance, you will need to isolate this additional capacitance through a power path or load switch, enabled upon the establishment of a USB-C connection. Although this connection can be detected discretely, the easiest way to detect a USB-C connection as a sink would be through a CC controller. Typical CC controllers have a GPIO output for controlling a power path based on connection status.

Another instance requiring a CC controller would be if you need support for USB 3.0 data. As with the USB-C source-only with USB 3.0 data example, a separate USB 3.0 multiplexer is required here too. To control this multiplexer, you will likely need a CC controller.

Figure 46 highlights the various blocks needed to implement a 5V USB-C sink-only port, depending on the V_{BUS} capacitance and USB 3.0 requirements.

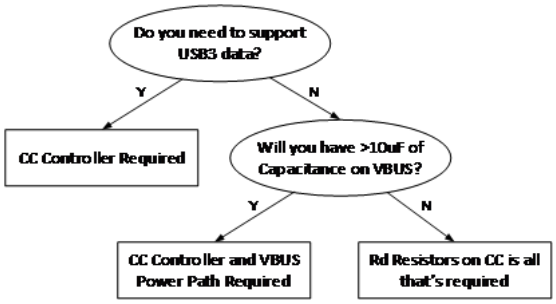


Figure 46. *CC controller selection*

5V USB-C DRP (no USB PD)

If the USB-C port must be able to provide outgoing power (source) but also consume power (sink), then you will have to implement a USB-C dual-role port (DRP). Again, you will likely need a CC controller. When supporting DRP capabilities, the CC line is actively toggling between presenting Rp and Rd before making a connection, which is known as a CC toggle. Essentially, if you were to plug a dual-role port into a sink-only device where Rd is present on CC, the CC toggle would stop and establish a connection once it toggles to Rp. The opposite applies when connecting a source-only port. If you connect two DRP ports together, there's a bit of randomness regarding which port connects as the source and which port connects as the sink, depending on where the two toggling CC lines land. If your system requires a particular power state, it might be best to implement a USB-C port where you can always send power-role or data-role swaps if you end up in the wrong port role after connecting to another DRP.

In terms of the power architecture, DRP systems might prefer to have two separate power paths, controlled by a DRP CC controller. In such cases, a dedicated 5V rail in the system connects to the source power path, while a separate sink power path can connect to something such as a battery charger or other system power rail. See the system block diagram in **Figure 47**. If you need USB 3.0 data, you will also need V_{CONN} and a USB 3.0 multiplexer in the system

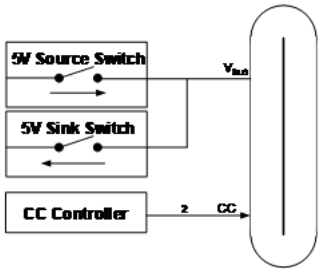


Figure 47. *5V dual-role port block diagram*

In some instances, you could get away with using a bidirectional power path if the system's power architecture allows for it. For example, some battery chargers support an on-the-go mode that allows them to be bidirectional. In instances like this, it makes sense to use a bidirectional power path.

20V USB-C source-only port with USB PD

Once you exceed 15W of power, you advance from requiring a standard USB-C controller to needing a USB PD controller. Enabling >15W of power requires USB PD negotiation between the port partners in order to ensure that both ends can support the amount of power.

Many types of end equipment – AC/DC adapters, docking stations, automotive charging ports and wall sockets – can provide 20V of power in a source-only implementation. Anything designed to provide power to a connected device would end up implementing their USB PD system in this fashion. Although these applications may have different requirements from a data standpoint, they all share the common requirement of providing >15W of power across the USB-C cable.

When designing a 20V USB-C source-only port with USB PD, there are some specifications from the USB-IF to keep in mind that impact system-level designs.

For a USB PD source-only port, the initial connection and requirement for Rp on the CC lines is the same as it is for the 5V USB-C source-only port without USB PD. To enable voltages beyond 5V, USB PD negotiation will need to occur. Depending on which voltage gets negotiated, the USB PD controller will need to adjust the output of a variable supply in order to provide the required output voltage.

The USB PD specification requires that the output voltage be ±5% from the negotiated power data object (PDO). **Figure 48** is a typical block diagram for a 20V USB-C source-only port with USB PD design. You can see that there is now a USB PD controller present instead of a CC controller, since a USB PD controller is necessary to handle USB PD negotiation. Additionally, there is a 20V load switch as just with the 5V source-only design, the source side is still responsible for managing overvoltage, undervoltage and overcurrent protections from the USB PD specification.

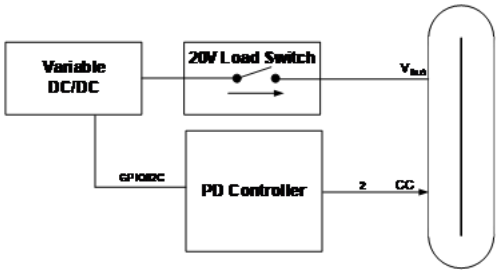


Figure 48. *20V USB PD source only block diagram*

The USB PD controller is expected to control a variable DC/DC supply based on the voltage negotiated with a connected device. The USB PD controller can do this through either GPIOs or I^2C. Several TI USB PD evaluation modules have examples of how USB PD controllers and DC/DC supplies interface with one another to enable this.

Another design consideration is how to ensure coverage for the required PDO voltages, depending on the power offered by the 20V USB-C source-only port. The discrete voltage levels required are 5V, 9V, 15V and 20V. The current can vary continuously, depending on the required power level (up to 3A). Any given power level still requires a source to support all previous voltages and power levels. For example, a 60W source must be able to supply 20V at 3A, 15V at 3A, 9V at 3A and 5V at 3A. This is an update in version 3.0 of the USB PD specification, in order to ensure that higher power supplies can support lower-powered devices.

Depending on the system architecture, it is possible to use a different supply. For example, for an AC/DC wall adapter, the USB PD controller can typically control the DC output voltage of the AC/DC converter directly, thus ensuring that the voltage is at the right level based on the USB PD negotiation. For systems with a DC voltage rail, you could use either a buck DC/DC converter or a buck or boost DC/DC converter.

To support >3A across a USB-C cable, there are a few additional system-level considerations. If your system will have a USB-C port with a removable USB-C cable, you will need to be able to provide V_{CONN} on the CC pins. In order to support >3A, the cable you connect will need to be able to support >3A. Cables that support >3A will contain an e-marker chip inside of them that stores the capabilities of the cable. Powering this e-marker and reading back the contents will require V_{CONN}. Many USB PD controllers integrate a dedicated V_{CONN} power path and switch V_{CONN} on automatically during the USB PD negotiation process.

If your 20V USB-C source-only port with USB PD design has a hard-wired cable, or what's known as a captive cable, you can support >3A without V_{CONN}. A captive cable is a nonremovable cable. In this case, you don't need to worry about supporting various different cables, and you always know the capabilities of the permanently connected cable.

When adding support for USB 3.0 data, follow the same approach as described in the 5V USB-C Source-Only Port with USB 3.0 Data (No USB PD) section: add a USB 3.0 multiplexer to control the flipping of the USB 3.0 SuperSpeed data pins within the system.

20V USB-C sink-only port with USB PD

A 20V USB-C sink-only port with USB PD is relatively easier to design than a source port. According to the USB PD specification, the source port is required to manage most of the power protections and control the voltage across the cable. The sink side mainly requests the connected source for voltage or current in the form of a PDO and completes USB PD negotiation.

In this implementation, the system likely receives its power from the USB-C port. With the expansion of USB-C and some countries now mandating adoption of the USB-C connector, devices that have historically been powered by a barrel jack, coaxial cable or other proprietary DC input must now have charge via a USB-C port. A few end-equipment examples would be Wi-Fi® adapters, *Bluetooth*® speakers, and e-bikes and e-scooters.

Similar to the 5V USB-C sink-only port, a USB PD sink-only port could also be required to implement a power path. The decision to add a power path will depend on the amount of capacitance exposed on V_{BUS}. From the USB-IF specification, you're only allowed 10µF of capacitance on V_{BUS}, so if your system requires more, you'll need to add a power path to isolate the system capacitance from V_{BUS}.

Figure 49 is a basic system-level block diagram for implementing a 20V USB-C sink-only port with USB PD design. The main difference between a USB-C 5V sink-only design and a 20V USB-C sink-only port with USB PD design is that the load switch scales up to 20V, and you will need a USB PD controller instead of a CC controller.

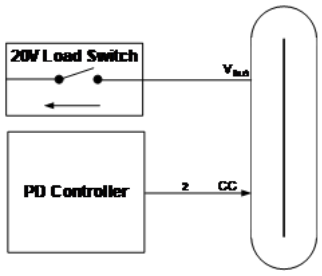


Figure 49. *20V USB PD sink only block diagram*

Again, support for USB 3.0 data necessitates the addition of a multiplexer.

5V source, 20V sink USB-C port with USB PD and DisplayPort™ Alternate Mode

In a notebook or PC implementation, a single USB-C port has the ability to sink USB PD voltages in order to charge the battery; provide at least 5V out to power small connected devices such as mice, keyboards and flash drives; and connect a monitor. You can quickly see how the required capabilities of a USB-C port is robust and flexible enough to meet the end user's expectations for certain end equipment.

Figure 50 shows the power architecture for this type of system. There are typically separate power paths in the system: one for sourcing the 5V and another for sinking up to 20V. If you only have one USB-C port in the system, you could implement a single power path rather than two separate ones. In such cases, you would need the battery charger to have bidirectional support, and include on-the-go support. Most systems that require 5V source and 20V sink as well as DisplayPort Alternate Mode support will have more than one USB port.

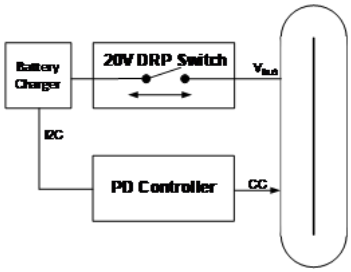


Figure 50. *5V source 20V sink USB PD block diagram*

If the end equipment will have multiple USB ports, a shared 5V rail can provide the sourcing power for both USB Type-A and USB-C ports. You will need to calculate the power budget of this 5V DC/DC supply based on the maximum current supported by each USB port when sourcing 5V to connected devices.

Connecting the sink power path to the battery charger isolates the capacitance from the battery charger from V_{BUS}, while also making sure that the battery charger receives power when the user connects an AC/DC adapter.

As in the previous examples, the USB PD controller will have integrated power path or provide a method to control them through GPIOs. Some USB PD controllers offer a N-channel field-effect transistor (NFET) gate driver to drive external NFETs directly.

Figure 50 also shows that the USB PD controller has the ability to supply V_{CONN}. The USB PD source-only design requires V_{CONN} when exceeding 3A of current. But adding support for DisplayPort Alternate Mode requires V_{CONN} in order to determine the data capabilities of the cable, not the power capabilities. Similar to the 5V USB-C source-only

port with USB 3.0 data example, it's important to confirm that the connected cable has capabilities for supporting DisplayPort Alternate Mode as well. V_{CONN} needs to power the e-marker in the cable in order to read back its capabilities.

Figure 51 shows a more complete block diagram that includes both the power and data blocks for implementing a 5V source, 20V sink USB-C port with USB PD and DisplayPort™ Alternate Mode design.

The last block in **Figure 51** is the DisplayPort Alternate Mode multiplexer. As in previous cases, DisplayPort Alternate Mode also uses the SuperSpeed pins on the USB-C connector to transmit video data.

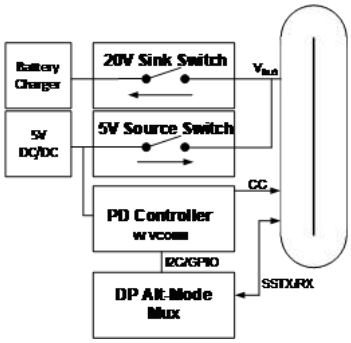


Figure 51. *5V source 20V sink with DisplayPort™ USB PD block diagram*

DisplayPort Alternate Mode includes several different pin configurations that help determine the distribution of the SuperSpeed pins between supporting USB 3.0 or DisplayPort video data. There are pin configurations that enable maximum bandwidth for DisplayPort by dedicating all SuperSpeed pairs for DisplayPort, and pin configurations that split the SuperSpeed pairs to enable both USB 3.0 and DisplayPort data simultaneously. The DisplayPort specification has further details about the pin configurations supported on USB-C.

The DisplayPort Alternate Mode multiplexer will multiplex the SuperSpeed pins to either the DisplayPort host or USB host, depending on the pin assignment negotiated within DisplayPort Alternate Mode. As with other peripheral devices, the USB PD controller is expected to communicate over I^2C or with GPIOs to the DisplayPort Alternate Mode multiplexer to configure it accordingly. If the port partner also supports DisplayPort Alternate Mode, the USB PD controller will automatically negotiate and enter DisplayPort Alternate Mode with the connected device. Based on this negotiation, the USB PD controller will then configure the DisplayPort Alternate Mode multiplexer through either I^2C or GPIO. During USB PD negotiation, power always gets negotiated first, followed by alternate modes such as DisplayPort.

20V USB-C DRP with USB PD and a battery charger

Historically, battery powered end-equipment has used either a barrel jack, coaxial cable or proprietary cable to charge the product. Moving to USB-C with USB PD enables you to both source and sink power, which then enables the conversion of a battery-powered device to a power bank. In other words, end users can both charge connected devices through USB-C and also have a device charged through the same USB-C connector. To accomplish these requirements, you would likely implement a DRP architecture with a bidirectional battery charger. Although this implementation may sound complex, it is typically a two-chip self-contained solution.

Figure 52 is a block diagram for a 20V USB DRP with USB PD design leveraging a bidirectional battery-charger IC. In this case, the battery charger charges the batteries when the end user connects a charger device. The battery charger will also provide the correct voltage on V_{BUS} when the end user connects a device that needs charging. In this case, the USB PD controller will also communicate to the charger IC over I^2C. When operating as a power source, the USB PD controller will communicate to the battery charger whether there's a connection, what voltage to provide, and where to set the current limit. The battery charger will need to have the correct voltage to meet the tolerances of the USB PD specification, and also ensure the voltage transitions (from 5V to 20V, for example) to meet the timing requirements of the USB PD specification. Typically, if a battery charger is designed for USB PD applications, it would be designed to meet those specifications, or provide configurable settings to adjust the voltage transitions in order to tune the output for USB PD compliance.

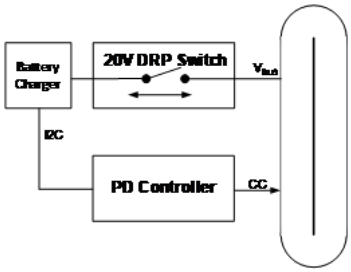


Figure 52. *5V source 20V sink USB PD block diagram*

When operating as a power sink, the USB PD controller will communicate what power is available to the battery charger and enable the battery charger to start charging the battery. **Figure 52** includes an additional 20V DRP power path because most battery charger ICs will require more than 10µF of capacitance on the input pin for stability. If you plan to use a charger IC that can operate with less than 10µF of capacitance, you can remove the 20V DRP power path from the system.

End equipment-specific block diagrams

Abstract

Author: Adam McGaffin

USB Type-C® (USB-C®) is quickly becoming the universal connector, enabling a wide variety of products to benefit from its technology. In this chapter, I'll describe some of the most popular types of end equipment that currently implement USB-C, and explain the high-level blocks that make up the subsystems surrounding the USB-C connector.

Laptops and industrial PCs

Laptops were the first products to implement USB-C, and a majority of the laptops that ship today have at least one USB-C port. With the ability to charge the system, power external peripherals, and transmit video, laptops benefit greatly from USB-C and USB PD. **Figure 53** is a typical block diagram for a notebook or industrial PC platform.

The only difference between a standard notebook PC and an industrial PC is the operating temperature range requirements. However, the core components between the two are the same, which is why they can share a common block diagram.

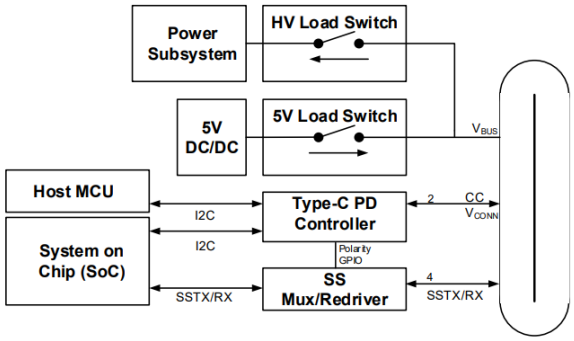


Figure 53. *Notebook and industrial PC block diagram*

Negotiating the high-power contracts needed to power the system as well as transmitting video over the connector requires a USB PD controller. From a power perspective, a majority of platforms implement two different power paths. One power path is dedicated to sinking high-voltage contracts in order to charge the laptop. The other power path is dedicated to 5V, in order to charge connected peripherals such as mice, keyboards and phones. The USB PD controller will either integrate these power paths directly or have the ability to control external power paths such as dedicated eFuses and load switches.

Transmitting video or data over the USB-C connector requires the implementation of a multiplexer or redriver in order to route the SuperSpeed lanes correctly. Controlling the multiplexer is also usually handled by the USB PD controller, toggling the polarity of the multiplexer either through general-purpose input/output or I^2C, based on the orientation of the connected cable.

Finally, it is common for these applications to implement some type of central microprocessor in order to handle the system-level events that occur on the USB-C port. A majority of USB PD controllers implement the I^2C interface, allowing for communication with the central microcontroller. Port partner information, power contracts and current video output speed are all examples of information that you can obtain from a USB PD controller.

Docking station

The most common accessory paired with a notebook is a docking station, as it enables the notebook to charge and transmit data using a single port. **Figure 54** is a block diagram for a docking station.

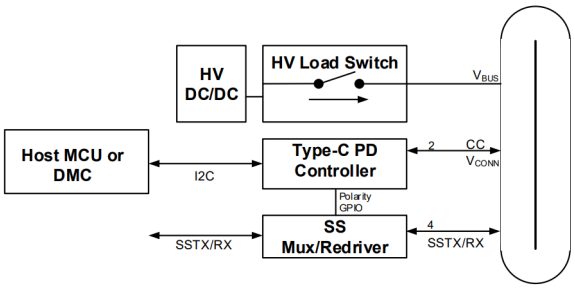


Figure 54. *Docking station block diagram*

Like a notebook, a docking station requires a USB PD controller in order to implement high-power contracts and negotiate video alternate modes. Unlike a notebook, however, you will only need one power path – a high-voltage power path connected to a high-voltage variable DC/DC so that the docking station can charge connected notebooks and other peripherals. The docking station will never sink power from this port, which is why a single power path is enough.

Bluetooth® speaker

Portable devices such as Bluetooth speakers have the ability to benefit from USB-C by allowing their product to charge and power connected peripherals from a single port, giving users the ability to turn their portable product into a power bank. **Figure 55** and **Figure 56** show two typical block diagrams for a Bluetooth speaker, which are often applicable to other portable battery-powered devices.

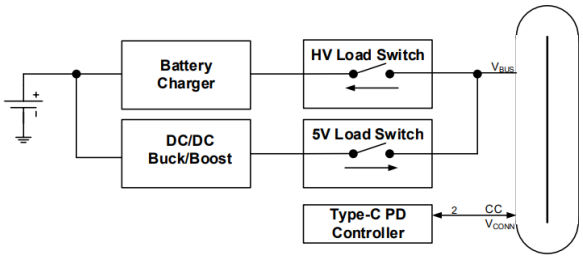


Figure 55. *Bluetooth speaker block diagram 1*

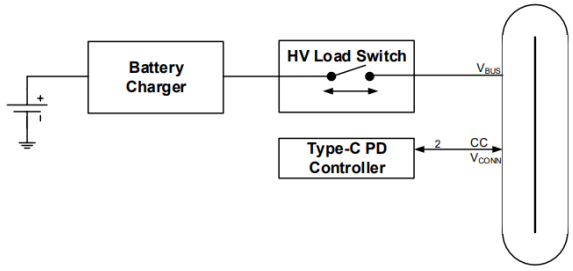


Figure 56. *Bluetooth speaker block diagram 2*

There are two potential power architectures when designing a Bluetooth speaker or any portable battery-powered device:

- Two separate power paths for sinking and sourcing power from the system.
- A single power path where the battery charger also acts as the source in on-the-go mode.

Instances when the first option make sense include if there is a common 5V rail used across multiple devices within Bluetooth speaker, and designer would prefer to have the sink power path isolated from the rest of the power architecture. The second option may be better for systems that want to sink and source voltages greater than 5V. In order for a system to source 9V from the USB-C port, you must implement a variable DC/DC that has the ability to start at 5V and then transition to 9V. Instead of having to implement a separate variable DC/DC for this functionality, you can use the on-the-go functionality integrated in many battery chargers.

Wi-Fi® routers and smart speakers

Traditional barrel-jack power devices such as Wi-Fi routers and smart speakers are transitioning to USB-C. **Figure 57** illustrates the transition from a proprietary barrel-jack connection to a USB-C port.

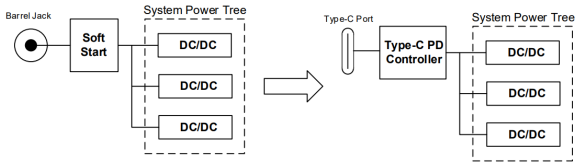


Figure 57. *Transitioning barrel jack to USB-C*

As shown in **Figure 57**, the core system power architecture can remain the same, as you can maintain the same voltage typically powered on the barrel jack negotiated on the USB-C port. Another benefit is that a majority of barrel-jack applications implement soft-start circuitry to limit the inrush current during the initial connection. USB-C ports can implement this same functionality through the built-in soft-start capabilities of the load switches used in a USB-C system.

Figure 58 is a more detailed illustration of the USB-C block, and what's required to implement USB-C in a Wi-Fi router or smart speaker application.

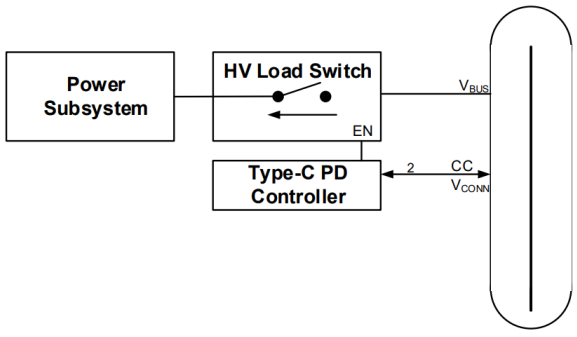


Figure 58. *Wi-Fi routers and smart speakers block diagram*

It is possible to implement a sink-only USB-C system such as a Wi-Fi router or smart speaker in one or two blocks. A USB PD controller will negotiate the required voltage with the connected charger, and a load switch or power path is also required. This power path connects to the system power architecture and subsystem that would power all of the voltage rails within the device.

Power tools

Power tools and similar portable power packs to power and charge laptops, phones and speakers on the go are another type of end equipment adopting USB-C. **Figure 59** and **Figure 60** show typical block diagrams for these types of systems.

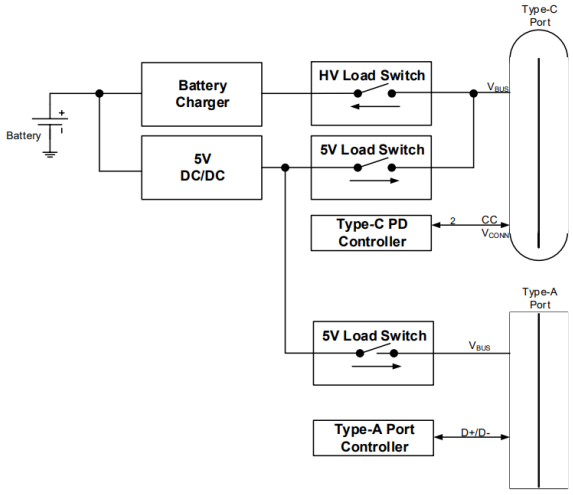


Figure 59. *Power tools block diagram 1*

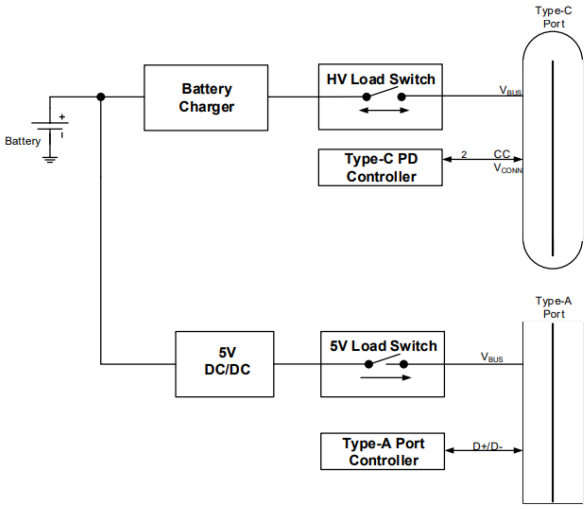


Figure 60. *Power tools block diagram 2*

While many designers are converting all of their USB Type-A ports to USB-C ports, some end equipment continues to include the older technology in order to power their customers' older products. One example of this is with power tools. Like Bluetooth speakers, the power tool battery pack has the ability to source and sink power from a single USB-C port. The USB Type-A port would only source power to charge connected devices.

There are two different implementations from a power perspective:

- The USB-C and USB Type-A port share the same 5V DC/DC power supply, with the output of the USB-C port limited to 5V.
- The USB-C ports implement a single power path, with the battery charger acting in on-the-go mode. The USB Type-A port has a dedicated 5V DC/DC power supply.

Another similarity between a power tool and a Bluetooth speaker is that the main reason to chose one topology over another is predicated on the output voltage on the USB-C port. If the USB-C port only needs a maximum of 5V, then a shared DC/DC topology may be the most beneficial, with the battery charger that's tied to the USB-C port acting only as a power sink. If the USB-C port is expected to supply >5V, then having the battery charger act as both the source and sink, relying on the on-the-go functionality built into the charger, is likely the best path forward.

Benefits of a TI PD Controller

Abstract

Authors: Joe Li, Adam McGaffin

Ever since its release in 2013, USB-C PD is a valuable technology enabling applications to support high data rates and higher power transfer across a single cable. However, there are some design challenges with PD adoption that slowed down the growth of USB-C PD technology. TI PD controllers directly address these problems and provide end users a complete solution.

TI solutions to common design challenges

TI offers highly integrated solution

Challenge: Most PD controllers do not integrate all necessary power paths required into their solutions, yielding inferior over current protection (OCP), over voltage protection (OVP) and reverse current protection (RCP). End users need to either design a discrete power path or purchase a 5V or high voltage (HV) load switch. This results to end users facing a large total solution size and a high total bill of materials (BOM).

Solution: TI offers complete, standalone USB-C PD solutions that integrate all necessary power paths and miscellaneous functions that system designers would need to design a USB-C PD system. **Figure 61** shows a simplified block diagram of a TI PD controller. Note the integrated field-effect transistors (FETs) serving as internal power paths, saving end users the trouble of designing extra 5V or HV power paths. At the same time, the integrated dead battery low dropout (LDO) generates a 3.3V rail to help power some of the system in a dead battery scenario. The internal power path protection offers robust defense against RCP, OVP and OCP. Additionally, TI PD controllers integrate 26-V tolerant CC pins for powerful protection when connected to non-compliant device. With a highly integrated design, TI PD controllers eliminate the need for firmware development or an external micro-controller.

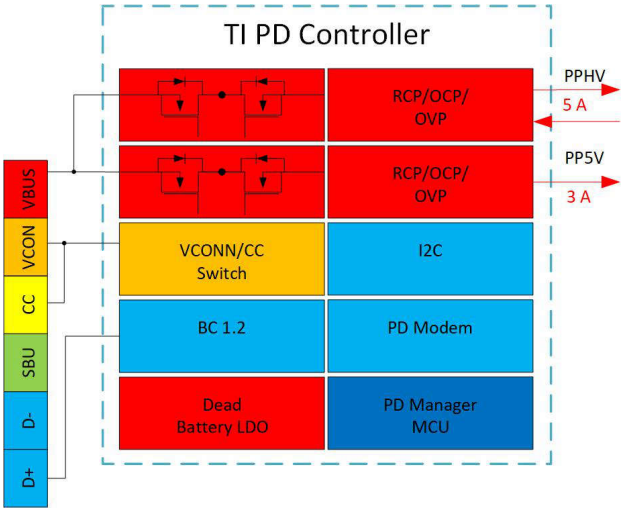


Figure 61. *TI PD controller design*

On the other hand, typical PD controller products are often less integrated and require additional design effort and materials from system designers in order to complete the system. **Figure 62** shows a typical PD controller design. With far fewer integrated functions, these types of solutions leave much of the work needed to implement a complete USB-C PD solution up to the system designer. With no internal power path, no power path protections, no dead battery LDO

and oftentimes no BC 1.2 protocol module, it is up to the end users to complete the design. With multiple additional chips and FETs, using a typical PD controller can turn out to be an inefficient, complex, and expensive solution.

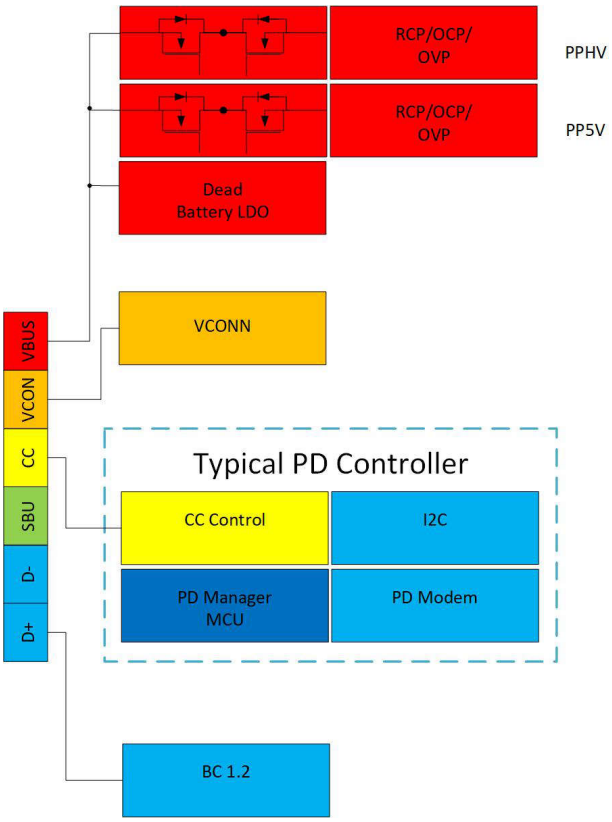


Figure 62. *Typical PD controller design*

Overall, with highly integrated designs, TI is able to provide one of the smallest total solution sizes in the industry while keeping one of the lowest total BOM. With built-in power path management and miscellaneous functions, TI offers a complete one-chip solution that saves end users the trouble and cost of searching for additional parts to complete their USB-C PD system.

TI offers simple configuration tool

Challenge: The ability to configure PD controllers before first-time use is very important. For example, an end user should easily be able to change the power role of a PD controller from source only to dual role power or to sink only. However, this is often not the case when designing with typical PD controllers as the configuration process can be confusing and time-consuming. Most of the configuration software requires manual coding, complex firmware scripting, or even in-depth knowledge of I2C registers to enable charger support, creating a steep learning curve and making the process even less user-friendly.

Solution: The configuration process for TI PD controllers is simple and requires minimum coding. For the TPS6598X series, end users can download a TI application customization graphical user interface (GUI) tool to easily change configuration settings or interact with the PD controller over I2C. **Figure 63** is a screenshot of using the TPS6598X series GUI tool to configure general-purpose input/output (GPIO) settings.

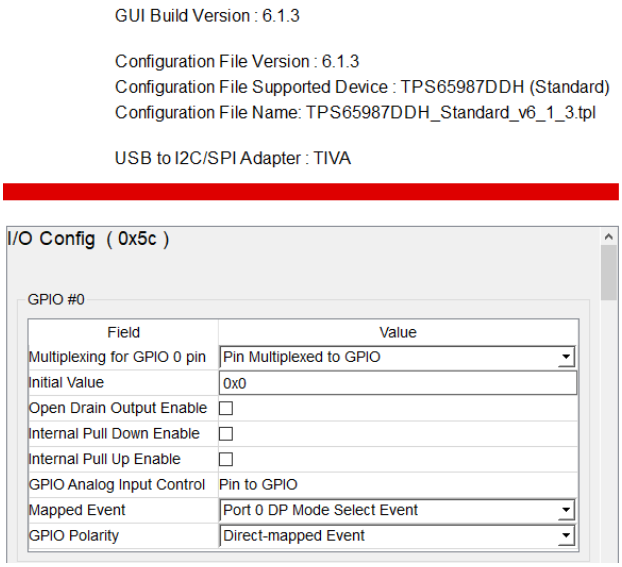


Figure 63. *TPS6598X series GUI tool*

For current and future PD controllers, TI offers an easy-to-use, web-based Q&A-styled GUI tool for end users to configure their PD controllers. End users can generate firmware/configuration image by simply answering some questions. See **Figure 64** for the easy configuration process.

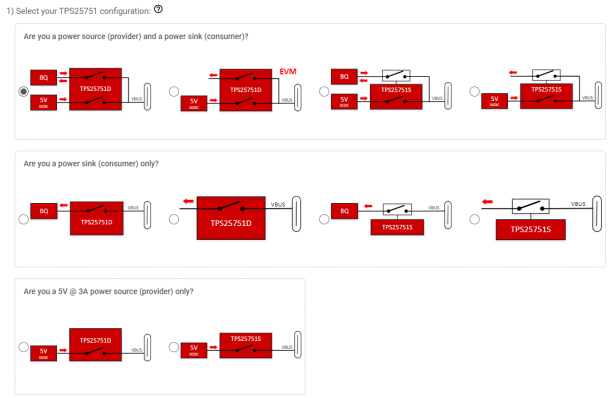


Figure 64. *Web-based GUI screenshot*

Figure 65. *Web-based GUI screenshot*

TI products are rigorously validated and USB-IF certified

Challenge: It is common that some PD controllers are not fully compliant with USB Implementers Forum (USB-IF) specifications. The lack of certification from USB-IF can cause issues when trying to interoperate with other USB-C end products or even failures of conventional USB-C functionalities; this can include DisplayPort Alternate Mode failure.

Solution: To ensure the reliability of TI products, all TI PD controllers undergo extensive validation before each release, including both firmware verification and system validation. With each new firmware release or firmware update, TI adds new tests accordingly to verify any new features and conducts full regression testing to certify all existing functionalities are not affected. After firmware testing, the product undergoes system validation which includes compliance, interop, fault, power measurement and functional tests to ensure that the product meets the standards for compliance and functionality with its accompanying devices. Furthermore, TI continues to upgrade and validate its PD controllers according to user inputs after each release.

TI also has a long-standing partnership with USB-IF, the committee in charge of setting the USB specifications. TI's collaboration with USB-IF drives TI to constantly work to enable compliance. TI has representatives that serve on the current USB-IF board of directors, making TI an important member of the USB community. All TI PD products undergo testing and certification following the USB-IF specification. This ensures that TI PD controllers implement USB Power Delivery specification properly, allowing for seamless compatibility with any other USB-IF certified system, not just one that implements a TI solution.

Other benefits of using TI PD controllers

TI offers complete reference design

TI is dedicated to making the end users adoption of USB-C PD technology as seamless as possible, which is why TI offers complete reference designs for most TI PD controllers covering various use cases. Visit the following website for a complete list of reference designs and use the parametric search tool to quickly locate the desired product: **https://www.ti.com/reference-designs/index.html#search?famid=361,658,3391**.

TI offers great customer support

To further improve end users' experience, TI offers 1st in class customer support. By simply navigating to **www.ti.com/usb-c**, end users can find all relevant training videos, reference designs and evaluation modules (EVMS) to develop and debug their USB-C PD systems. On top of the existing resources, TI's online E2E forum links individual end user directly with a TI expert for additional technical support.

Conclusion

Switching to a USB-C power system can seem complicated and expensive, but by selecting a TI PD controller, end users minimize their design effort, time to market, and system cost. TI strives to make their USB-PD products the industry's gold standard. TI identifies and addresses the biggest concerns regarding USB-C PD products by offering one of the most integrated PD solutions, developing easy to use GUI configuration tool and providing first-class customer support.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265