Demystifying USB-C® using Linux's USB Type-C® Connector Class

Benson Leung & Nathan Kolluru
Software Engineering, Chrome OS, Google Inc

USB Developer Days 2019 – Taipei, Taiwan
November 19, 2019

USB Type-C® and USB-C® are registered trademarks of USB Implementers Forum.
Presentation Agenda

• **USB-C® as it is now**
• Where Connector Class fits in
  • Power Delivery protocol
  • Linux’s USB Type-C connector class
  • Mockup uses
• Remaining SW work
• Where it’s headed
USB-C® is great and all, but CONFUSING
Many USB-C® Cables, > # of required logos

Table 3-1 USB Type-C Standard Cable Assemblies

<table>
<thead>
<tr>
<th>Cable Ref</th>
<th>Plug 1</th>
<th>Plug 2</th>
<th>USB Version</th>
<th>Cable Length</th>
<th>Current Rating</th>
<th>USB Power Delivery</th>
<th>USB Type-C Electronically Marked</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC2-3</td>
<td>C</td>
<td>C</td>
<td>USB 2.0</td>
<td>≤ 4 m</td>
<td>3 A</td>
<td>Supported</td>
<td>Optional</td>
</tr>
<tr>
<td>CC2-5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5 A</td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>CC3G1-3</td>
<td>C</td>
<td>C</td>
<td>USB 3.2 Gen1</td>
<td>≤ 2 m</td>
<td>3 A</td>
<td>Supported</td>
<td>Required</td>
</tr>
<tr>
<td>CC3G1-5</td>
<td></td>
<td></td>
<td>and USB4 Gen2</td>
<td></td>
<td>5 A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CC3G2-3</td>
<td>C</td>
<td>C</td>
<td>USB 3.2 Gen2</td>
<td>≤ 1 m</td>
<td>3 A</td>
<td>Supported</td>
<td>Required</td>
</tr>
<tr>
<td>CC3G2-5</td>
<td></td>
<td></td>
<td>and USB4 Gen2</td>
<td></td>
<td>5 A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CC4G3-3</td>
<td>C</td>
<td>C</td>
<td>USB4 Gen3</td>
<td>≤ 0.8 m</td>
<td>3 A</td>
<td>Supported</td>
<td>Required</td>
</tr>
<tr>
<td>CC4G3-5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5 A</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Source: USB Type-C Specification Revision 2.0
USB-C® Port Features “à la carte”

**Connector**
- Receptacle or Captive Cable

**Power**
- PD2.0 or PD3.0 FRS ask or give
- Source, Sink, or DRP Unconstrained Power
- Sources any power output from 2.5W ~ 100W
- Sinks any power input from 0.0W ~ 100W

**USB Data**
- UFP or DFP or DRD
- USB 2.0 Host or Gadget or both
- USB 3.2 Gen 1 or Gen2 Host or Gadget or both
- USB4™ PCIe and/or USB 3.x and/or DP

**Alt Modes**
- VESA DisplayPort™ Alt Mode
- Intel Thunderbolt™ 3
- (Others)

*Source: “Google Go” Sandwich Order Form*
No Way to Visually Discern Cable/Port Capabilities

What do these do? What are they?
Presentation Agenda

• USB-C® as it is now
• Where Connector Class fits in
  • Power Delivery protocol
  • Linux’s USB Type-C connector class
  • Mockup uses
• Remaining SW work
• Where it’s headed
USB Power Delivery Is the Key

Human beings can’t tell; computers know instantly.

- I can source 100w via USB PD 3.0, with Unconstrained Power
- I have Intel Thunderbolt™ 3 AltMode
- I have USB 3.2 and USB 2.0 gadget too

- I also have a USB endpoint
  I’m an external hard drive

- Sorry bud.
  - I’m only a USB 2.0 cable
    I don’t have enough wires
  - … but can carry 5a
    (at 20V USB PD 3.0)

- OK. No worries. I got this.
  - I only need 10w now, but 65w later
    My battery is full! (USB PD 3.0)
  - I can host the USB 2.0 hard drive

- Let me warn the user for you!
Linux's USB Type-C® Connector Class

/sys/class/typec/
/sys/class/typec/port0
/sys/class/typec/port0/port0-partner/
/sys/class/typec/port0/port0-cable/
/sys/class/typec/port0/port0-cable/port0-plug0
/sys/class/typec/port0/port0-cable/port0-plug1
/sys/class/typec/port0/<alternate mode>/mode1/

Bubbles up TCPC and Embedded Controller function to Userspace.


Thunderbolt and the Thunderbolt logo are trademarks of Intel Corporation or its subsidiaries in the U.S. and/or other countries.
Mockup: Phone as PD analyser

Source: “Twebkie” web-app by David Schneider (Google)
https://twebkie.org/
Mockup: Expose device capability

Source: VDM:DiscoverIdentity + DP USB2.0 Billboard Class Touch Technologies 3p Launcher for Android
• USB-C® as it is now
• Where Connector Class fits in
  • Power Delivery protocol
  • Linux’s USB Type-C connector class
  • Mockup uses
• Remaining SW work
• Where it’s headed
Remaining Work for USB Type-C® Connector Class

- Initially written and upstreamed to mainline Linux in 2017 and out of date.
- Need to sync to latest USB Type-C® 2.0 -- USB PD 3.0 R2.0 -- and USB4™ specs.

- Has no concept of SRCCAP and SNKCAP.
- Need to add Power Data Objects support to properly describe Supplies and Sinks.

- Passive and Active Cable VDO properties not fully exposed.
- UFP/DFP VDOs not exposed (new to USB PD 3.0 R2.0 spec)

- Partner needs indication of DRP, DRD, and UP
- Chrome OS support
  - cros_ec driver in development
Help Support This - Use Accurate Properties!

Vendors of Cables, Devices, Hosts, and Chargers: Accurately describe product in USB PD 3.0!

- **SRCCAP** and **SNKCAP** -- Fill out accurately!
  - UP=1 (if appropriate)
  - FRS Supported
  - Variable PDO
  - Batteries: 1 when full, 0 when empty\(^1\)
  - Static property (like DRP/DRD)
  - List wide input Vin range (and test)

- **VDM DiscoverIdentity** -- Please respond, do not use chipset vendor defaults!
  - VID:PID + bcdDevice
  - AMA VDO
  - UFP/DFP VDO
  - Cable VDOs
  - [USB-Data ⇐ ⇒ USB-PD] response
  - Check: MF, Data, Plug/receptacle, Supported Pins\(^2\)
  - Check: Current, Speed, Length (latency)

- **VDM DiscoverSVID** -- Please respond accurately!
  - Modes
  - List all AltModes

Source:
1. USB PD3.0 R2, Section 8.2.6 Use of “Unconstrained Power” bit with Batteries and AC supplies
2. VESA DisplayPort™ AltMode Specification, Section “5. Discovery and USB PD”
Presentation Agenda

- USB-C® as it is now
- Where Connector Class fits in
  - Power Delivery protocol
  - Linux’s USB Type-C connector class
  - Mockup uses
- Remaining SW work
- Where it’s headed
Call for Help - UCSI

• Intel's UCSI needs major updating too in order to implement

• **Ask would be mainly for UCSI specification:**
  • Mandate passing up to the OS raw VDOs and PDOs wherever possible
  • Would allow kernel UCSI “USB Type-C Connector driver” to decode them

• Requesting help for ecosystems:
  • Benson’s team will add FW + Kernel drivers for Chromebooks (and upstream to Linux open-source kernel)
  • UCSI developer team would need to assist for other PCs

**Audience members today are key in helping make that happen.**
Call for Help - Other Ecosystems

• Similar reporting should be done in other OSes (not Linux) for users
  • Microsoft Windows, Apple macOS, iPadOS, etc.
• These OSes likely already have a similar framework
  • Keep them up to date with the latest USB Type-C, USB PD, and USB4 specs
  • Expose the information in a readable way in Device Manager/System Profiler

Benefit and importance same -- implementation slightly different.
Presentation Agenda

- USB-C® as it is now
- Where Connector Class fits in
  - Power Delivery protocol
  - Linux’s USB Type-C connector class
  - Mockup uses
- Remaining SW work
- Where it’s headed

- BONUS TIME - Q&A
Доверяй, но проверяй. (Trust, but verify.)
Accurate DiscoverID Responses Matter!

Source: Tech Design Forum, “Combining USB Type-C and DisplayPort support in portable implementations”
Accurate DiscoverID Responses Matter!

Source: Tech Design Forum, “Combining USB Type-C and DisplayPort support in portable implementations”
Accurate DiscoverID Responses Matter!

Source: ChromeOS EC Feature: Enable wake on DP
https://chromium-review.googlesource.com/c/chromiumos/overlays/board-overlays/+/1764523

Oh no, you broke it!
Useful Tools for USB-C® Field Kit
ChromeOS Open-Source Hardware: Fluffy

Meet Fluffy

He bites
Wasn’t enough: so we made a 20 x 20 model

Meet Fluffy

He bites