

Universal Serial Bus Type-C Cable and Connector Specification

Release 2.45
~~October 2024~~
March 2026

Copyright © 2014-~~2024~~2026, USB 3.0 Promoter Group:

**Apple Inc., HP Inc., Intel Corporation, Microsoft Corporation,
Renesas, STMicroelectronics, and Texas Instruments**

All rights reserved.

NOTE: Adopters may only use the USB Type-C® cable and connector to implement USB or third-party functionality as expressly described in this Specification; all other uses are prohibited.

LIMITED COPYRIGHT LICENSE: The USB 3.0 Promoters grant a conditional copyright license under the copyrights embodied in the USB Type-C Cable and Connector Specification to use and reproduce the Specification for the sole purpose of, and solely to the extent necessary for, evaluating whether to implement the Specification in products that would comply with the specification. Without limiting the foregoing, use of the Specification for the purpose of filing or modifying any patent application to target the Specification or USB compliant products is not authorized. Except for this express copyright license, no other rights or licenses are granted, including without limitation any patent licenses. In order to obtain any additional intellectual property licenses or licensing commitments associated with the Specification a party must execute the USB 3.0 Adopters Agreement. NOTE: By using the Specification, you accept these license terms on your own behalf and, in the case where you are doing this as an employee, on behalf of your employer.

INTELLECTUAL PROPERTY DISCLAIMER

THIS SPECIFICATION IS PROVIDED TO YOU “AS IS” WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE. THE AUTHORS OF THIS SPECIFICATION DISCLAIM ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF ANY PROPRIETARY RIGHTS, RELATING TO USE OR IMPLEMENTATION OF INFORMATION IN THIS SPECIFICATION. THE PROVISION OF THIS SPECIFICATION TO YOU DOES NOT PROVIDE YOU WITH ANY LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS.

All implementation examples and reference designs contained within this Specification are included as part of the limited patent license for those companies that execute the USB 3.0 Adopters Agreement.

USB Type-C®, USB-C® and USB4® are trademarks of the Universal Serial Bus Implementers Forum (USB-IF). DisplayPort™ is a trademark of VESA. All product names are trademarks, registered trademarks, or service marks of their respective owners.

Thunderbolt™ is a trademark of Intel Corporation. You may only use the Thunderbolt™ trademark or logo in conjunction with products designed to this specification that complete proper certification and executing a Thunderbolt™ trademark license – see <https://www.usb.org/compliance> for further information.

Contents

Specification Editor	18
Specification Work Group Contributors	18
Pre-Release Draft Industry Reviewing Companies That Provided Feedback	26
Revision History	27
1 Introduction	28
1.1 Purpose	28
1.2 Scope	28
1.3 Related Documents	28
1.4 Conventions	29
1.4.1 Precedence	29
1.4.2 Keywords	29
1.4.3 Numbering	30
1.5 Terms and Abbreviations	30
2 Overview	36
2.1 Introduction	36
2.2 USB Type-C Receptacles, Plugs and Cables	37
2.3 Configuration Process	38
2.3.1 Source-to-Sink Attach/Detach Detection	39
2.3.2 Plug Orientation/Cable Twist Detection	39
2.3.3 Initial Power (Source-to-Sink) Detection and Establishing the Data (Host-to-Device) Relationship	39
2.3.4 USB Type-C VBUS Current Detection and Usage	40
2.3.5 USB PD Communications	40
2.3.6 Functional Extensions	41
2.4 VBUS	41
2.5 VCONN	41
2.6 Hubs	42
3 Mechanical	43
3.1 Overview	43
3.1.1 Compliant Connectors	43
3.1.2 Compliant Cable Assemblies	43
3.1.3 Compliant USB Type-C to Legacy Cable Assemblies	44
3.1.4 Compliant USB Type-C to Legacy Adapter Assemblies	45
3.2 USB Type-C Connector Mating Interfaces	45
3.2.1 Interface Definition	45
3.2.2 Reference Designs	69

3.2.3	Pin Assignments and Descriptions	76
3.3	Cable Construction and Wire Assignments	77
3.3.1	Cable Construction (<i>Informative</i>).....	77
3.3.2	Wire Assignments	79
3.3.3	Wire Gauges and Cable Diameters (<i>Informative</i>).....	80
3.4	Standard USB Type-C Cable Assemblies	82
3.4.1	USB Full-Featured Type-C Cable Assembly	82
3.4.2	USB 2.0 Type-C Cable Assembly	84
3.4.3	USB Type-C Captive Cable Assembly	84
3.4.4	USB Type-C Thumb Drive Assembly	85
3.5	Legacy Cable Assemblies	85
3.5.1	USB Type-C to USB 3.1 Standard-A Cable Assembly	86
3.5.2	USB Type-C to USB 2.0 Standard-A Cable Assembly	87
3.5.3	USB Type-C to USB 3.1 Standard-B Cable Assembly	88
3.5.4	USB Type-C to USB 2.0 Standard-B Cable Assembly	89
3.5.5	USB Type-C to USB 2.0 Mini-B Cable Assembly.....	90
3.5.6	USB Type-C to USB 3.1 Micro-B Cable Assembly	91
3.5.7	USB Type-C to USB 2.0 Micro-B Cable Assembly	92
3.6	Legacy Adapter Assemblies.....	92
3.6.1	USB Type-C to USB 3.1 Standard-A Receptacle Adapter Assembly	92
3.6.2	USB Type-C to USB 2.0 Micro-B Receptacle Adapter Assembly	94
3.7	Electrical Characteristics	94
3.7.1	Raw Cable (<i>Informative</i>).....	95
3.7.2	USB Type-C to USB Type-C Passive Cable Assemblies	96
3.7.3	Mated Connector (<i>Informative – USB 3.2 Gen2 and USB4 Gen2</i>).....	118
3.7.4	Receptacle Connector SI Requirements and Testing (<i>Normative – USB4 Gen3/Gen4</i>)	122
3.7.5	USB Type-C to USB Legacy Cable Assemblies (<i>Normative</i>)	124
3.7.6	USB Type-C to USB Legacy Adapter Assemblies (<i>Normative</i>)	128
3.7.7	Shielding Effectiveness Requirements (<i>Normative</i>)	131
3.7.8	DC Electrical Requirements (<i>Normative</i>).....	133
3.8	Mechanical and Environmental Requirements (<i>Normative</i>).....	136
3.8.1	Mechanical Requirements	136
3.8.2	Environmental Requirements.....	141
3.9	Docking Applications (<i>Informative</i>)	142
3.10	Implementation Notes and Design Guides	143
3.10.1	EMC Management (<i>Informative</i>)	143
3.10.2	Stacked and Side-by-Side Connector Physical Spacing (<i>Informative</i>)	146

3.10.3	Cable Mating Considerations (<i>Informative</i>)	146
3.11	Extended Power Range (EPR) Cables	148
3.11.1	Electrical Requirements	148
3.11.2	EPR Cable Identification Requirements	148
4	Functional	149
4.1	Signal Summary	149
4.2	Signal Pin Descriptions	149
4.2.1	USB 3.2/USB4 Pins	149
4.2.2	USB 2.0 Pins	150
4.2.3	Auxiliary Signal Pins	150
4.2.4	Power and Ground Pins	150
4.2.5	Configuration Pins	150
4.3	Sideband Use (SBU)	150
4.4	Power and Ground	151
4.4.1	IR Drop	151
4.4.2	VBUS	152
4.4.3	VCONN	153
4.5	Configuration Channel (CC)	158
4.5.1	Architectural Overview	158
4.5.2	CC Functional and Behavioral Requirements	170
4.5.3	USB Port Interoperability Behavior	207
4.6	Power	224
4.6.1	Power Requirements during USB Suspend	225
4.6.2	VBUS Power Provided Over a USB Type-C Cable	226
4.7	USB Hubs	232
4.8	Power Sourcing and Charging	232
4.8.1	DFP as a Power Source	232
4.8.2	Non-USB Charging Methods	235
4.8.3	Sinking Host	235
4.8.4	Sourcing Device	235
4.8.5	Charging a System with a Dead Battery	235
4.8.6	USB Type-C Multi-Port Chargers	236
4.9	Electronically Marked Cables	240
4.9.1	Parameter Values	243
4.9.2	Active Cables	243
4.10	VCONN-Powered Accessories (VPAs) and VCONN-Powered USB Devices (VPDs)	243
4.10.1	VCONN-Powered Accessories (VPAs)	243

4.10.2	VCONN-Powered Devices (VPDs).....	244
4.11	Parameter Values	245
4.11.1	Termination Parameters	245
4.11.2	Timing Parameters	248
4.11.3	Voltage Parameters	252
5	<i>USB4</i> Discovery and Entry	259
5.1	Overview of the Discovery and Entry Process	259
5.2	<i>USB4</i> Functional Requirements	260
5.2.1	<i>USB4</i> Host Functional Requirements	260
5.2.2	USB Device Functional Requirements	260
5.2.3	<i>USB4</i> Alternate Mode Support	260
5.3	<i>USB4</i> Power Requirements	261
5.3.1	Source Power Requirements	261
5.3.2	Sink Power Requirements	261
5.3.3	Device Power Management Requirements	261
5.4	<i>USB4</i> Discovery and Entry Flow Requirements	262
5.4.1	USB Type-C Initial Connection	262
5.4.2	USB Power Delivery Contract	262
5.4.3	<i>USB4</i> Discovery and Entry Flow	262
5.4.4	<i>USB4</i> Post-Entry Operation	268
5.5	<i>USB4</i> Hub Connection Requirements	268
5.5.1	<i>USB4</i> Hub Port Initial Connection Requirements	268
5.5.2	<i>USB4</i> Hub UFP and Host Capabilities Discovery	268
5.5.3	Hub DFP Connection Requirements	269
5.5.4	Hub Ports Connection Behavior Flow Examples	270
5.5.5	Connecting to Downstream <i>USB4</i> Hubs	276
5.5.6	Fallback Functional Requirements for <i>USB4</i> Hubs	276
5.6	<i>USB4</i> Device Connection Requirements	277
5.6.1	Fallback Mapping of <i>USB4</i> Peripheral Functions of USB Device Class Types	277
5.7	Parameter Values	278
5.7.1	Timing Parameters	278
6	Active Cables	279
6.1	General Specifications for All Active Cables	279
6.1.1	Discovering Active Cable Characteristics	279
6.1.2	Electrical Requirements	281
6.1.3	Mechanical Requirements	317
6.2	Additional Specifications for Copper and Hybrid-Optical Active Cables	318

6.2.1	Active Cable Block Diagram	319
6.2.2	USB4 Asymmetric Mode Support	319
6.2.3	Active Cable USB PD Requirements	319
6.2.4	Active Cable Behaviors in Response to USB PD Events	319
6.2.5	Active Cable Power Requirements	320
6.3	Additional Specifications for Optically Isolated Active Cables	320
6.3.1	OIAC Block Diagrams	321
6.3.2	OIAC Limitations and General Requirements	323
6.3.3	OIAC Cable Power Requirements	324
6.3.4	OIAC USB PD Requirements	324
6.3.5	OIAC Connection Flow and State Diagrams	334
6.3.6	Additional Electrical Requirements for OIAC	352
6.3.7	Additional Mechanical Requirements for OIAC	355
A	Liquid Corrosion Mitigation Mode	356
A.1	Overview	356
A.2	Detail	356
A.3	Liquid Detection Methods	358
A.3.1	Liquid Measurement Method	358
A.3.2	Pulsed Measurement Method	358
A.3.3	Impedance Measurement Method	359
A.4	Liquid Detection Pins in the Connector	362
B	Debug Accessory Mode	365
B.1	Overview	365
B.2	Functional	365
B.2.1	Signal Summary	365
B.2.2	Port Interoperability	366
B.2.3	Debug Accessory Mode Entry	366
B.2.4	Connection State Diagrams	366
B.2.5	DTS Port Interoperability Behaviors	375
B.2.6	Orientation Detection	384
B.3	Security/Privacy Requirements	384
C	USB Type-C Digital Audio	385
C.1	Overview	385
C.2	USB Type-C Digital Audio Specifications	385
D	Thermal Design Considerations for Active Cables	386
D.1	Introduction	386
D.2	Model	386

D.2.1	Assumptions.....	386
D.2.2	Model Architecture.....	387
D.2.3	Heat Sources.....	387
D.2.4	Heat Flow.....	388
D.3	<i>USB 3.2</i> Single-Lane Active Cable.....	389
D.3.1	<i>USB 3.2</i> Single-Lane Active Cable Design Considerations	389
D.4	Dual-Lane Active Cables.....	392
D.4.1	<i>USB 3.2</i> Dual-Lane Active Cable Design Considerations	392
D.4.2	<i>USB 3.2</i> Dual-Lane Active Cable in a Multiple Port Configuration	394
D.5	<i>USB 3.2</i> Host and Device Design Considerations	395
D.5.1	Heat Spreading or Heat Sinking from Host or Device.....	395
D.5.2	Motherboard Temperature Control	396
D.5.3	Wider Port Spacing for Multi-Port Applications	396
D.5.4	Power Policies	396
E	Alternate Modes.....	397
E.1	Alternate Mode Architecture.....	397
E.2	Alternate Mode Requirements.....	397
E.2.1	Alternate Mode Pin Reassignment.....	398
E.2.2	Alternate Mode Electrical Requirements.....	398
E.3	Parameter Values	401
E.4	Example Alternate Mode – USB DisplayPort™ Dock	402
E.4.1	USB DisplayPort™ Dock Example.....	402
E.4.2	Functional Overview	402
E.4.3	Operational Summary	403
E.5	Example Alternate Mode Entry Flow with Cable Warnings.....	404
E.5.1	Operational Summary	405
F	Thunderbolt™ 3 Compatibility Discovery and Entry.....	406
F.1	<i>TBT3</i> Compatibility Mode Functional Requirements	406
F.1.1	<i>TBT3</i> -Compatible Power Requirements.....	406
F.1.2	<i>TBT3</i> -Compatible Host Requirements	406
F.1.3	<i>TBT3</i> -Compatible Device Upstream Requirements.....	406
F.1.4	<i>TBT3</i> -Compatible Device Downstream Requirements.....	406
F.1.5	<i>TBT3</i> -Compatible Self-Powered Device Without Predefined Upstream Port Rules	407
F.1.6	<i>TBT3</i> -Compatible Devices with a Captive Cable	407
F.2	<i>TBT3</i> Discovery and Entry Flow.....	407
F.2.1	<i>TBT3</i> Passive Cable Discover Identity Responses	408
F.2.2	<i>TBT3</i> Active Cable Discover Identity Responses.....	411

F.2.3	<i>TBT3</i> Device Discover Identity Responses	414
F.2.4	<i>TBT3</i> Discover SVID Responses.....	415
F.2.5	<i>TBT3</i> Device Discover Mode Responses	416
F.2.6	<i>TBT3</i> Cable Discover Mode Responses	417
F.2.7	<i>TBT3</i> Cable Enter Mode Command.....	418
F.2.8	<i>TBT3</i> Device Enter Mode Command	419
F.2.9	<i>TBT3</i> Cable Functional Difference Summary	421
G	Extracting Pulse Response from Sampled Data and Calculating Non-Linearity Noise.....	422
H	<i>USB PD</i> High-Voltage Design Considerations	424
H.1	Potential for Arcing Damage During Cable Withdrawal.....	424
H.2	Arcing During USB Type-C Cable Withdrawal	424
H.3	Mitigating Arcing Damage During Cable Withdrawal Due to Sink Discharge	426
H.3.1	Limiting Sink Discharge Rate.....	427
H.3.2	Load Removal	428
H.3.3	Limiting Source Current Capability	431
I	<i>USB PD</i> Encoding Guidelines for USB Type-C Product Types	432
I.1	USB Type-C Product Type Definitions and <i>USB PD</i> Encodings.....	432
I.2	<i>USB PD</i> Encoding Guidelines Tables	433
I.3	Detailed Examples.....	440
I.3.1	<i>USB 3.2</i> Host – no DC – Power Consumer : DRP (Source or Sink) / Not DRD (DFP).....	440
I.3.2	<i>USB 3.2</i> Peripheral – Power Provider : DRP (Source or Sink) / Not DRD (UFP).....	440
I.3.3	<i>USB4</i> Host – Power Consumer : DRP (Source or Sink) / Not DRD (DFP)	441
I.3.4	<i>USB4</i> Host – <i>USB4</i> DC – Partial or no USB Equivalent : DRP (Source or Sink) / DRD.....	441
I.3.5	<i>USB4</i> Peripheral : UFP (Sink)	442
I.3.6	<i>USB4</i> Dock (Upstream) – Power Provider : DRP (Source or Sink) / Not DRD (UFP)	443
I.3.7	<i>USB4</i> Hub (Downstream) – DFP (Source)	443
J	Design Assumptions for the <i>USB4</i> Gen4 LRD Cable Specification	445

Figures

Figure 2-1	USB Type-C Receptacle Interface (Front View).....	36
Figure 2-2	USB Full-Featured Type-C Plug Interface (Front View).....	37
Figure 3-1	USB Type-C Receptacle Interface Dimensions	48
Figure 3-2	Reference Design USB Type-C Plug External EMC Spring Contact Zones.....	52
Figure 3-3	USB Full-Featured Type-C Plug Interface Dimensions.....	53
Figure 3-4	Right-Angle USB Type-C Plug Dimensional Requirements	56
Figure 3-5	Reference Footprint for a USB Type-C Vertical Mount Receptacle (<i>Informative</i>)	57
Figure 3-6	Reference Footprint for a USB Type-C Dual-Row SMT Right-Angle Receptacle (<i>Informative</i>)	58
Figure 3-7	Reference Footprint for a USB Type-C Hybrid Right-Angle Receptacle (<i>Informative</i>).....	59
Figure 3-8	Reference Footprint for a USB Type-C Mid-Mount Dual-Row SMT Receptacle (<i>Informative</i>)	60
Figure 3-9	Reference Footprint for a USB Type-C Mid-Mount Hybrid Receptacle (<i>Informative</i>).....	61

Figure 3-10 Reference Footprint for a <i>USB 2.0</i> Type-C Through Hole Right Angle Receptacle (<i>Informative</i>)	62
Figure 3-11 Reference Footprint for a <i>USB 2.0</i> Type-C Single Row Right Angle Receptacle (<i>Informative</i>)	63
Figure 3-12 <i>USB 2.0</i> Type-C Plug Interface Dimensions	65
Figure 3-13 USB Type-C Plug EMC Shielding Spring Tip Requirements	68
Figure 3-14 Reference Design of Receptacle Mid-Plate	69
Figure 3-15 Reference Design of Retention Latch	70
Figure 3-16 Illustration of the Latch Soldered to the Paddle Card Ground	70
Figure 3-17 Reference Design of the USB Full-Featured Type-C Plug Internal EMC Spring	71
Figure 3-18 Reference Design of the <i>USB 2.0</i> Type-C Plug Internal EMC Spring	72
Figure 3-19 Reference Design of Internal EMC Pad	73
Figure 3-20 Reference Design of a USB Type-C Receptacle with External EMC Springs	74
Figure 3-21 Reference Design of a USB Full-Featured Type-C Plug Paddle Card	75
Figure 3-22 Illustration of a USB Full-Featured Type-C Cable Cross Section, a Coaxial Wire Example with VCONN	78
Figure 3-23 Illustration of a USB Full-Featured Type-C Cable Cross Section, a Coaxial Wire Example without VCONN	78
Figure 3-24 USB Full-Featured Type-C Standard Cable Assembly	82
Figure 3-25 USB Type-C to <i>USB 3.1</i> Standard-A Cable Assembly	86
Figure 3-26 USB Type-C to <i>USB 2.0</i> Standard-A Cable Assembly	87
Figure 3-27 USB Type-C to <i>USB 3.1</i> Standard-B Cable Assembly	88
Figure 3-28 USB Type-C to <i>USB 2.0</i> Standard-B Cable Assembly	89
Figure 3-29 USB Type-C to <i>USB 2.0</i> Mini-B Cable Assembly	90
Figure 3-30 USB Type-C to <i>USB 3.1</i> Micro-B Cable Assembly	91
Figure 3-31 USB Type-C to <i>USB 2.0</i> Micro-B Cable Assembly	92
Figure 3-32 USB Type-C to <i>USB 3.1</i> Standard-A Receptacle Adapter Assembly	93
Figure 3-33 USB Type-C to <i>USB 2.0</i> Micro-B Receptacle Adapter Assembly	94
Figure 3-34 Illustration of Test Points for a Mated Cable Assembly	96
Figure 3-35 Recommended Differential Insertion Loss Requirement (<i>USB 3.2</i> Gen2 and <i>USB4</i> Gen2)	97
Figure 3-36 Recommended Differential Return Loss Requirement	97
Figure 3-37 Recommended Differential Crosstalk Requirement	98
Figure 3-38 Recommended Differential Near-End and Far-End Crosstalk Requirement between USB D+/D- Pair and TX/RX Pair	99
Figure 3-39 Recommended Differential Insertion Loss Requirement (<i>USB4</i> Gen3/Gen4)	99
Figure 3-40 Illustration of Insertion Loss Fit at Nyquist Frequency	100
Figure 3-41 Input Pulse Spectrum	101
Figure 3-42 IMR Limit as Function of ILfitatNq	102
Figure 3-43 IRL Limit as Function of ILfitatNq	104
Figure 3-44 Differential-to-Common Mode Conversion Requirement	104
Figure 3-45 IMR Limit as Function of ILfit at 10 GHz (<i>USB4</i> Gen3/Gen4)	108
Figure 3-46 Definition of Port, Victim, and Aggressor	109
Figure 3-47 IXT_DP and IXT_USB Limit as Function of ILfit at 10 GHz (<i>USB4</i> Gen3/Gen4)	109
Figure 3-48 IRL Limit as Function of ILfit at 10 GHz (<i>USB4</i> Gen3/Gen4)	110
Figure 3-49 Differential-to-Common Mode Conversion Requirement (<i>USB4</i> Gen3/Gen4)	110
Figure 3-50 Cable Assembly in System	111
Figure 3-51 Requirement for Differential Coupling between CC and D+/D-	114
Figure 3-52 Requirement for Single-Ended Coupling between CC and D- in <i>USB 2.0</i> Type-C Cables	114
Figure 3-53 Requirement for Single-Ended Coupling between CC and D- in USB Full-Featured Type-C Cables	115
Figure 3-54 Requirement for Differential Coupling between VBUS and D+/D-	115
Figure 3-55 Requirement for Single-Ended Coupling between SBU_A and SBU_B	116
Figure 3-56 Requirement for Single-Ended Coupling between SBU_A/SBU_B and CC	117

Figure 3-57 Requirement for Coupling between SBU_A and differential D+/D-, and SBU_B and differential D+/D-	117
Figure 3-58 Illustration of USB Type-C Mated Connector	118
Figure 3-59 Recommended Impedance Limits of a USB Type-C Mated Connector	119
Figure 3-60 Recommended Ground Void Dimensions for USB Type-C Receptacle	119
Figure 3-61 Recommended Differential Near-End and Far-End Crosstalk Limits between D+/D- Pair and TX/RX Pairs.....	121
Figure 3-62 Recommended Limits for Differential-to-Common-Mode Conversion	122
Figure 3-63 IMR Limit as Function of ILfitatNq for USB Type-C to Legacy Cable Assembly	127
Figure 3-64 IRL Limit as Function of ILfitatNq for USB Type-C to Legacy Cable Assembly	127
Figure 3-65 Cable Assembly Shielding Effectiveness Testing.....	131
Figure 3-66 Shielding Effectiveness Pass/Fail Criteria.....	132
Figure 3-67 LLCR Measurement Diagram	133
Figure 3-68 Temperature Measurement Point.....	134
Figure 3-69 Example Current Rating Test Fixture Trace Configuration	135
Figure 3-70 Example of 4-Axis Continuity Test Fixture	137
Figure 3-71 Torque Force Application Distance for Right-Angle Plugs.....	139
Figure 3-72 Example Wrenching Strength Test Fixture for Plugs without Overmold.....	140
Figure 3-73 Reference Wrenching Strength Continuity Test Fixture	140
Figure 3-74 Example of Wrenching Strength Test Mechanical Failure Point	141
Figure 3-75 Wrenching Strength Test with Cable in Fixture.....	141
Figure 3-76 USB Type-C Cable Receptacle Flange Example	143
Figure 3-77 EMC Guidelines for Side Latch and Mid-Plate	144
Figure 3-78 EMC Finger Connections to Plug Shell.....	145
Figure 3-79 EMC Pad Connections to Receptacle Shell	145
Figure 3-80 Examples of Connector Apertures.....	146
Figure 3-81 Recommended Minimum Spacing between Connectors.....	146
Figure 3-82 Recommended Minimum Plug Overmold Clearance	147
Figure 3-83 Cable Plug Overmold and an Angled Surface.....	147
Figure 4-1 Cable IR Drop.....	151
Figure 4-2 Cable IR Drop for powered cables.....	151
Figure 4-3 Logical Model for Single-Lane Data Bus Routing across USB Type-C-based Ports.....	159
Figure 4-4 Logical Model for USB Type-C-based Ports for a Single-Lane Direct Connect Device.....	159
Figure 4-5 Pull-Up/Pull-Down CC Model.....	161
Figure 4-6 Current Source/Pull-Down CC Model.....	161
Figure 4-7 Source Functional Model for CC1 and CC2	164
Figure 4-8 Source Functional Model Supporting <i>USB PD PR_Swap</i>	165
Figure 4-9 Sink Functional Model for CC1 and CC2	165
Figure 4-10 Sink Functional Model Supporting <i>USB PD PR_Swap</i> and <i>VCONN_Swap</i>	166
Figure 4-11 DRP Functional Model for CC1 and CC2.....	167
Figure 4-12 Connection State Diagram: Source.....	172
Figure 4-13 Connection State Diagram: Sink.....	173
Figure 4-14 Connection State Diagram: Sink with Accessory Support	174
Figure 4-15 Connection State Diagram: DRP	175
Figure 4-16 Connection State Diagram: DRP with Accessory and Try.SRC Support	176
Figure 4-17 Connection State Diagram: DRP with Accessory and Try.SNK Support.....	177
Figure 4-18 Connection State Diagram: Charge-Through VPD	178
Figure 4-19 Sink Power Sub-States	201
Figure 4-20 Cable eMarker State Diagram.....	203
Figure 4-21 Cable Ra Management State Diagram	204
Figure 4-22 Source to Sink Functional Model.....	207

Figure 4-23 Source to DRP Functional Model.....	208
Figure 4-24 DRP to Sink Functional Model.....	209
Figure 4-25 DRP to DRP Functional Model – CASE 1	210
Figure 4-26 DRP to DRP Functional Model – CASES 2 & 3.....	211
Figure 4-27 Source to Source Functional Model	213
Figure 4-28 Sink to Sink Functional Model.....	213
Figure 4-29 DRP to VPD Model	214
Figure 4-30 Example DRP to Charge-Through VPD Model.....	215
Figure 4-31 Source to Legacy Device Port Functional Model	222
Figure 4-32 Legacy Host Port to Sink Functional Model.....	222
Figure 4-33 DRP to Legacy Device Port Functional Model.....	223
Figure 4-34 Legacy Host Port to DRP Functional Model.....	224
Figure 4-35 Sink Monitoring for Current in Pull-Up/Pull-Down CC Model.....	227
Figure 4-36 Sink Monitoring for Current in Current Source/Pull-Down CC Model.....	228
Figure 4-37 <i>USB PD</i> over CC Pins	229
Figure 4-38 <i>USB PD</i> BMC Signaling over CC	229
Figure 4-39 USB Type-C Cable's Output as a Function of Load for Non-PD-based USB Type-C Charging.....	234
Figure 4-40 0 – 3 A <i>USB PD</i> -based Charger USB Type-C Cable's Output as a Function of Load	234
Figure 4-41 3 – 5 A <i>USB PD</i> -based Charger USB Type-C Cable's Output as a Function of Load	235
Figure 4-42 Electronically Marked Cable with VCONN connected through the cable	240
Figure 4-43 Electronically Marked Cable with SOP' at both ends	241
Figure 4-44 Electronically Marked Direct Connect Device with a Captive Cable SOP' Powered by VCONN.....	241
Figure 4-45 Electronically Marked Direct Connect Device with a Captive Cable SOP' Powered by VBUS or Local Power.....	242
Figure 4-46 Electronically Marked Direct Connect Device with No Cable with SOP' Powered by VCONN.....	242
Figure 4-47 Electronically Marked Direct Connect Device with No Cable with SOP' Powered by VBUS or Local Power.....	243
Figure 4-48 Example Charge-Through VCONN-Power USB Device Use Case	245
Figure 4-49 Example Source Implementation.....	248
Figure 4-50 Exiting from <i>Attached.SRC</i> with slow discharging VBULK	249
Figure 4-51 Exiting from <i>Attached.SRC</i> with fast discharging VBULK	249
Figure 4-52 DRP Timing.....	250
Figure 4-53 Pull-Up/Pull-Down Voltage Detection Model.....	252
Figure 4-54 Current Source Voltage Detection Model	252
Figure 5-1 <i>USB4</i> Discovery and Entry Flow Model.....	263
Figure 5-2 <i>USB4</i> Hub with <i>USB4</i> Host and Device Connection Flow Alignment	271
Figure 5-3 <i>USB4</i> Hub with <i>USB 3.2</i> Host and <i>USB4</i> Device Connection Flow Alignment	272
Figure 5-4 <i>USB4</i> Hub with <i>USB4</i> Host and <i>USB 3.2</i> Device Connection Flow Alignment	273
Figure 5-5 <i>USB4</i> Hub with <i>USB 3.2</i> Host and Device Connection Flow Alignment	274
Figure 5-6 <i>USB4</i> Hub with <i>USB4</i> Host and DP Alt Mode Device Connection Flow Alignment	275
Figure 5-7 <i>USB4</i> Hub with <i>USB 3.2</i> Host and DP Alt Mode Device Connection Flow Alignment	276
Figure 6-1 Active Cable Topologies	282
Figure 6-2 SuperSpeed USB Electrical Test Points	285
Figure 6-3 SuperSpeed USB Compliance Test Setup	285
Figure 6-4 Compliance Points Definition.....	289
Figure 6-5 RX Differential Return-Loss Mask.....	290
Figure 6-6 Re-timer-based Active Cable Compliance Test Setup	291
Figure 6-7 Example of Transmitter Frequency Variation During Clock Switching.....	294
Figure 6-8 Active Cable Functional Test Setup.....	294
Figure 6-9 Linear Re-driver-based Active Cable Compliance Setup.....	296
Figure 6-10 Gain Parameters Specified for the Linear Re-driver Active Cable.....	299

Figure 6-11	OUTPUT_NOISE Limit Versus <i>ILfitatNq</i>	300
Figure 6-12	Gain Parameters for <i>USB4</i> Gen4 LRD Active Cables	303
Figure 6-13	<i>USB4</i> LRD Active Cable Snooping of TxFFE Negotiation	309
Figure 6-14	<i>USB4</i> State Machine per Channel (<i>Informative</i>)	310
Figure 6-15	<i>USB4</i> Gen4 CL1/CL2 Exit Flow with LRD Active Cable	312
Figure 6-16	<i>USB4</i> Symmetric to Asymmetric Transition.....	313
Figure 6-17	<i>USB4</i> Asymmetric to Symmetric Transition.....	316
Figure 6-18	OIAC <i>USB PD</i> Message Forwarding	329
Figure 6-19	OIAC Successful Data Role Swap	332
Figure 6-20	OIAC Rejected Data Role Swap.....	332
Figure 6-21	OIAC Wait Data Role Swap.....	333
Figure 6-22	OIAC Initiator Reject Data Role Swap	333
Figure 6-23	OIAC Initiator Wait Data Role Swap.....	334
Figure 6-24	OIAC Discovery – Phase 1.....	336
Figure 6-25	OIAC Reboot – Phase 2	337
Figure 6-26	OIAC Plug-A Configured as DFP – Phase 3	338
Figure 6-27	OIAC Plug-A Configured as UFP – Phase 3	339
Figure 6-28	OIAC Plug-A No Connection Possible Billboard – Phase 3.....	340
Figure 6-29	OIAC Plug-A State Diagram Part 1 (Phase 1 and 2)	341
Figure 6-30	OIAC Plug-A State Diagram Part 2 (Phase 3)	342
Figure 6-31	OIAC Plug-B State Diagram.....	348
Figure 6-32	Illustration of Usages for OIAC that Require an Adapter or Hub	354
Figure A-1	Electrolytic Liquid Corrosion Example.....	356
Figure A-2	Corrosion Mitigation in Pull-Up/Pull-Down CC Model.....	357
Figure A-3	Corrosion Mitigation in Current Source/Pull-Down CC Model.....	357
Figure A-4	Corrosion Mitigation in Current Source/Pull-Down CC Model with Alternate Mitigation.....	357
Figure A-5	Liquid Detection by Leakage Measurement	358
Figure A-6	Liquid Detection by Pulsed Measurement	359
Figure A-7	Example Measurement of a Dry Port.....	360
Figure A-8	Example Measurement of a Port with Reverse Osmosis Water	360
Figure A-9	Example Measurement of a Port with Tap Water	361
Figure A-10	Example Measurement of a Port with Artificial Sweat	361
Figure A-11	Example Measurement of a Port with Ocean Water.....	362
Figure A-12	Example Liquid Detect Pin(s)/Pad(s) Location Area	363
Figure A-13	Example Liquid Detect Pad Along All Connector Pins.....	363
Figure A-14	Example Liquid Detect Pins Adjacent to VBUS/SBU/CC.....	364
Figure A-15	Example Liquid Detect Connector Surface Mount View	364
Figure B-1	USB Type-C Debug Accessory Layered Behavior	365
Figure B-2	DTS Plug Interface.....	365
Figure B-3	Connection State Diagram: DTS Source.....	367
Figure B-4	Connection State Diagram: DTS Sink.....	368
Figure B-5	Connection State Diagram: DTS DRP	369
Figure B-6	TS Sink Power Sub-States	373
Figure D-1	Active Cable Model (Single Port, Top Mount Receptacle)	387
Figure D-2	Model Architecture.....	387
Figure D-3	Heat Sources and Heat Flow Paths.....	388
Figure D-4	Vertically Stacked Horizontal Connectors 3x1 Configuration (VERT).....	389
Figure D-5	Horizontally Stacked Vertical Connectors 1x3 Configuration (HZ90).....	390
Figure D-6	Horizontally Stacked Horizontal Connectors 1x3 Configuration (HORZ).....	390
Figure D-7	<i>USB 3.2</i> Single-Lane 3A Active Cable in a 3-Port Configuration.....	391
Figure D-8	<i>USB 3.2</i> Single-Lane 5A Active Cable in a 3-Port Configuration.....	391

Figure D-9 Impact of Overmold Power P_0 and Thermal Boundary Temperature T_{MB} at 3 A VBUS in a Single Port Configuration	393
Figure D-10 Impact of Overmold Power P_0 and Thermal Boundary Temperature T_{MB} at 5 A VBUS in a Single Port Configuration	393
Figure D-11 USB 3.2 Active Cable Dongle Design (One End Shown)	394
Figure D-12 USB 3.2 Dual-Lane 3A Active Cable in a 3-Port Configuration.....	394
Figure D-13 USB 3.2 Dual-Lane 5A Active Cable in a 3-Port Configuration.....	395
Figure D-14 Example: Additional Heat Spreader on Receptacle in Host or Device	396
Figure D-15 Example: Heat Sinking on Chassis of Host or Device.....	396
Figure E-1 Pins Available for Reconfiguration over the Full-Featured Cable	398
Figure E-2 Pins Available for Reconfiguration for Direct Connect Applications	398
Figure E-3 Alternate Mode Implementation using a USB Type-C to USB Type-C Cable	400
Figure E-4 Alternate Mode Implementation using a USB Type-C to Alternate Mode Cable or Device.....	400
Figure E-5 USB DisplayPort Dock Example	402
Figure E-6 USB4 DFP Mode Entry Flow Example.....	404
Figure F-1 TBT3 Discovery Flow	408
Figure H-1 Arcing Damage to USB Type-C VBUS Contacts	424
Figure H-2 Arcing Due to Discharge.....	425
Figure H-3 Arcing Prevention During Sink Discharge by Limiting Slew Rate	427
Figure H-4 Arcing Prevention During Sink Discharge by Load Removal.....	429

Tables

Table 2-1 Summary of power supply options.....	41
Table 3-1 USB Type-C Standard Cable Assemblies.....	44
Table 3-2 USB Type-C Legacy Cable Assemblies	45
Table 3-3 USB Type-C Legacy Adapter Assemblies	45
Table 3-4 USB Type-C Receptacle Interface Pin Assignments.....	76
Table 3-5 USB Type-C Receptacle Interface Pin Assignments for USB 2.0-only Support	77
Table 3-6 USB Type-C Standard Cable Wire Assignments	79
Table 3-7 USB Type-C Cable Wire Assignments for Legacy Cables/Adapters.....	80
Table 3-8 Reference Wire Gauges for standard USB Type-C Cable Assemblies.....	81
Table 3-9 Reference Wire Gauges for standard USB Type-C to Legacy Cable Assemblies.....	81
Table 3-10 USB Full-Featured Type-C Standard Cable Assembly Wiring.....	83
Table 3-11 USB 2.0 Type-C Standard Cable Assembly Wiring.....	84
Table 3-12 USB Type-C to USB 3.1 Standard-A Cable Assembly Wiring	86
Table 3-13 USB Type-C to USB 2.0 Standard-A Cable Assembly Wiring	87
Table 3-14 USB Type-C to USB 3.1 Standard-B Cable Assembly Wiring	88
Table 3-15 USB Type-C to USB 2.0 Standard-B Cable Assembly Wiring	89
Table 3-16 USB Type-C to USB 2.0 Mini-B Cable Assembly Wiring.....	90
Table 3-17 USB Type-C to USB 3.1 Micro-B Cable Assembly Wiring	91
Table 3-18 USB Type-C to USB 2.0 Micro-B Cable Assembly Wiring	92
Table 3-19 USB Type-C to USB 3.1 Standard-A Adapter Assembly Wiring	93
Table 3-20 USB Type-C to USB 2.0 Micro-B Receptacle Adapter Assembly Wiring	94
Table 3-21 Differential Insertion Loss Examples for USB TX/RX with Twisted Pair Construction	95
Table 3-22 Differential Insertion Loss Examples for USB TX/RX with Coaxial Construction	96
Table 3-23 Key Parameters in COM Configuration File	112
Table 3-24 Electrical Requirements for CC and SBU Wires	113
Table 3-25 Coupling Matrix for Low Speed Signals.....	113
Table 3-26 Maximum Mutual Inductance (M) between VBUS and Low Speed Signal Lines	116

Table 3-27 USB D+/D– Signal Integrity Requirements for USB Type-C to USB Type-C Passive Cable Assemblies.....	118
Table 3-28 USB Type-C Mated Connector Recommended Signal Integrity Characteristics (<i>Informative</i>)	120
Table 3-29 USB Type-C Receptacle Connector Signal Integrity Characteristics for <i>USB4</i> Gen3 (<i>Normative</i>) ..	123
Table 3-30 USB D+/D– Signal Integrity Requirements for USB Type-C to Legacy USB Cable Assemblies (<i>Normative</i>).....	124
Table 3-31 Design Targets for USB Type-C to <i>USB 3.1</i> Gen2 Legacy Cable Assemblies (<i>Informative</i>)	125
Table 3-32 USB Type-C to <i>USB 3.1</i> Gen2 Legacy Cable Assembly Signal Integrity Requirements (<i>Normative</i>)	126
Table 3-33 USB D+/D– Signal Integrity Requirements for USB Type-C to Legacy USB Adapter Assemblies (<i>Normative</i>).....	128
Table 3-34 Design Targets for USB Type-C to <i>USB 3.1</i> Standard-A Adapter Assemblies (<i>Informative</i>)	129
Table 3-35 USB Type-C to <i>USB 3.1</i> Standard-A Receptacle Adapter Assembly Signal Integrity Requirements (<i>Normative</i>).....	130
Table 3-36 Current Rating Test PCB.....	135
Table 3-37 Maximum DC Resistance Requirement (<i>Normative</i>).....	135
Table 3-38 Force and Moment Requirements	138
Table 3-39 Environmental Test Conditions	141
Table 3-40 Reference Materials	142
Table 4-1 USB Type-C List of Signals	149
Table 4-2 VBUS Source Characteristics	152
Table 4-3 VBUS Sink Characteristics	153
Table 4-4 USB Type-C Source Port’s VCONN Requirements Summary.....	154
Table 4-5 VCONN Source Characteristics	154
Table 4-6 Cable VCONN Sink Characteristics.....	155
Table 4-7 VCONN-Powered Accessory (VPA) Sink Characteristics.....	156
Table 4-8 VCONN-Powered Device (VPD) Sink Characteristics	157
Table 4-9 USB Type-C Port Interoperability.....	160
Table 4-10 Source Perspective	162
Table 4-11 Source (Host) and Sink (Device) Behaviors by State	162
Table 4-12 Recommended Implementation of Try.SRC and Try.SNK for Dual-Role Ports with Preferred Roles	169
Table 4-13 <i>USB PD</i> Swapping Port Behavior Summary.....	169
Table 4-14 Use of Power and Data Role Swaps for Dual-Role Ports with Preferred Roles.....	170
Table 4-15 Power Role Behavioral Model Summary	170
Table 4-16 Source Port CC Pin State.....	179
Table 4-17 Sink Port CC Pin State.....	179
Table 4-18 Mandatory and Optional States	205
Table 4-19 Precedence of Power Source Usage	225
Table 4-20 USB Type-C Current Advertisement and PDP Equivalent	227
Table 4-21 Sink Maximum Current Limit When Attached to CTVPD	230
Table 4-22 Example Charge-Through VPD Sink Maximum Currents based on VBUS Impedance and GND Impedance	231
Table 4-23 Examples of 4-port Shared Capacity Group and Power Sharing Policies.....	237
Table 4-24 eMarker Timing	243
Table 4-25 Charge-Through VPD CC Impedance (RccCON) Requirements	245
Table 4-26 CTVPD Charge-Through Port VBUS Bypass Requirements	245
Table 4-27 Source CC Termination (Rp) Requirements	246
Table 4-28 Sink CC Termination (Rd) Requirements	246
Table 4-29 Powered Cable Termination Requirements	246
Table 4-30 CC Termination Requirements for Disabled state, ErrorRecovery state, and Unpowered Source	247

Table 4-31	SBU Termination Requirements	247
Table 4-32	VBUS and VCONN Timing Parameters	248
Table 4-33	DRP Timing Parameters	250
Table 4-34	CC Timing Parameters	251
Table 4-35	Sink CC Pin Voltages for Connect Detection for Rd and Clamp Voltage $\pm 20\%$	255
Table 4-36	Sink CC Pin Voltages for Connect and Current Advertisement Detection for Rd $\pm 10\%$	255
Table 4-37	Source CC Pin Voltages for All Current Advertisements using a Pull-Up current	256
Table 4-38	Source CC Pin Voltages for Pull-Up Resistance to 5 V	257
Table 4-39	Source CC Pin Voltages for Pull-Up Resistance to 3.3 V	258
Table 4-40	CC Pin Clamping Voltage	258
Table 5-1	Certified Cables Where <i>USB4</i> -compatible Operation is Expected	265
Table 5-2	Fallback Mapping <i>USB4</i> Functions to USB Device Class Types	277
Table 5-3	USB Billboard Device Class Availability Following <i>USB4</i> Device Entry Failure	278
Table 6-1	<i>USB4</i> Cable Identity Summary	280
Table 6-2	Active Cable Features	281
Table 6-3	Active Cable SBU Requirements	282
Table 6-4	Active Cable Power-on Requirements	283
Table 6-5	Maximum <i>USB 3.2</i> U0 Delay	284
Table 6-6	<i>USB 3.2</i> U-State Requirements	284
Table 6-7	Active Cable <i>USB 3.2</i> Stressed Source Swing, TP1	286
Table 6-8	Active Cable <i>USB 3.2</i> Stressed Source Jitter, TP1	287
Table 6-9	Active Cable <i>USB 3.2</i> Input Swing at TP2 (<i>Informative</i>)	287
Table 6-10	Active Cable <i>USB 3.2</i> Output Swing at TP3 (<i>Informative</i>)	288
Table 6-11	Compliance Points Definition	289
Table 6-12	<i>USB4</i> CL-State Requirements for Active Cables	290
Table 6-13	Re-timer-based <i>USB4</i> Gen2 and Gen3 Active Cable Output Specifications Applied for All Speeds (at TP3')	292
Table 6-14	Stressed Receiver Conditions for <i>USB4</i> Gen2 and Gen3 Cable Compliance Testing (at TP2)	294
Table 6-15	Linear Re-driver-based Active Cable Output Parameters	297
Table 6-16	Input Signal at TP2 for Compliance Testing	298
Table 6-17	<i>USB4</i> Gen4 LRD-based Active Cable ILfitMask Limits	302
Table 6-18	<i>USB4</i> Gen4 LRD-based Active Cable Wiring Diagram for Compliance	304
Table 6-19	<i>USB4</i> Active Cable SBx Transaction Snooping	305
Table 6-20	<i>USB4</i> LRD Active Cable Logical Timing Parameters	307
Table 6-22	Cable Temperature Requirements	318
Table 6-23	OIAC <i>USB PD</i> Message Handling	325
Table 6-24	Recommended OIAC Sink_Capabilities PDO (SOP) on Initial Connection	326
Table 6-25	Recommended OIAC Sink_Capabilities PDO (SOP) on Active Connection	326
Table 6-26	Recommended OIAC Active Sink RDO (SOP)	327
Table 6-27	OIAC <i>USB PD</i> Message Timing	330
Table 6-28	Port and Plug Capabilities	335
Table 6-29	<i>USB 3.2</i> U-State Requirements	354
Table 6-30	<i>USB4</i> CL-State Requirements for OIAC	355
Table A-1	Example Measurement Test Conditions	359
Table B-1	DTS to TS Port Interoperability	366
Table B-2	Rp/Rp Charging Current Values for a DTS Source	373
Table B-3	Mandatory and Optional States	375
Table D-1	Heat Sources and Heat Dissipation Example (1.5 W cable and 5 A)	388
Table D-2	<i>USB 3.2</i> Active Cable Design Single Port Case Study at 35 °C Ambient and 60 °C Thermal Boundary (Single Lane)	389

Table D-3 <i>USB 3.2</i> Active Cable Design Single Port Case Study at 35 °C Ambient and 60 °C Thermal Boundary (Dual Lane).....	392
Table E-1 USB Safe State Electrical Requirements	401
Table E-2 USB Billboard Device Class Availability Following Alternate Mode Entry Failure	401
Table E-3 Alternate Mode Signal Noise Ingression Requirements.....	401
Table F-1 <i>TBT3</i> Passive Cable Discovery Identity VDO Responses.....	409
Table F-2 <i>TBT3</i> Passive Cable VDO for <i>USB PD</i> Revision 2.0, Version 1.3.....	410
Table F-3 <i>TBT3</i> Passive Cable VDO for <i>USB PD</i> Revision 3.0, Version 1.2.....	410
Table F-4 <i>TBT3</i> Active Cable Discovery Identity VDO Responses	411
Table F-5 <i>TBT3</i> Active Cable VDO for <i>USB PD</i> Revision 2.0, Version 1.3	412
Table F-6 <i>TBT3</i> Active Cable VDO 1 for <i>USB PD</i> Revision 3.0, Version 1.2	412
Table F-7 <i>TBT3</i> Active Cable VDO 2 for <i>USB PD</i> Revision 3.0, Version 1.2	413
Table F-8 <i>TBT3</i> Device Discovery Identity VDO Responses.....	414
Table F-9 <i>TBT3</i> Discover SVID VDO Responses.....	415
Table F-10 <i>TBT3</i> Device Discover Mode VDO Responses	416
Table F-11 <i>TBT3</i> Cable Discover Mode VDO Responses.....	417
Table F-12 <i>TBT3</i> Cable Enter Mode Command.....	418
Table F-13 <i>TBT3</i> Device Enter Mode Command	419
Table F-14 <i>TBT3</i> Cable Functional Difference Summary	421
Table G-1 Linear Fit Pulse Extraction Parameters.....	423
Table I-1 USB Type-C Product Example Clarifying Notes.....	434
Table I-2 <i>USB PD</i> Encoding Guidelines for Example Products.....	437

Specification Editor

SpecWerkz

Brad Saunders

Specification Work Group Contributors

The following is a list of contributor companies and the individuals that were members of the work group roster at the time of this release (October 2024). Note: For historical reasons, the following list also includes individuals that were members of the work group and associated with their company affiliations at the time of each release going back to Release 1.0.

Acroname Inc.	Grant Fritz	Connor Goss	Matt Krugman
Advanced-Connectek, Inc. (ACON)	Glen Chandler	Conrad Choy	Alan Tsai
	Victory Chen	Vicky Chuang	Wayne Wang
	Bobo Chen	Same Dai	Stephen Yang
	Dennis Cheung	Jessica Feng	Sunney Yang
	Jeff Chien	Aven Kao	Kevin Yao
	Lee (Dick Lee) Ching	Danny Liao	
Advanced Micro Devices	Steve Capezza	Jason Hawken	Joseph Scanlon
	Michael Comai	Juan Martinez	Peter Teng
	Walter Fry	Gerald Merits	Sujan Thomas
	Will Harris	Tim Perley	
Allion Labs, Inc.	Howard Chang	Yan Ken	Brian Shih
	Alex Chuang	Lexus Lee	Chester Tsai
	Maroco Fan	Stan Lin	Chou Tsungbo
	Jesse Jiang	Denver Mishima	Liam Wu
	Terry Kao	Minoru Ohara	
Amphenol Corporation	Louis Chan	Terry Ke	Shawn Wei
	Zhinennng Fan	Martin Li	Alan Yang
	Jesse Jaramillo	Lino Liu	
Agilent Technologies, Inc.	James Choate		
Analogix Semiconductor, Inc.	Mehran Badii	Haijian Sui	Yueke Tang
	Greg Stewart		
Apple Inc. (USB Promoter company)	Colin Abraham	Brian Follis	Nathan Ng
	Mahmoud Amini	Zheng Gao	James Orr
	Sree Anantharaman	Derek Iwamoto	Keith Porthouse
	Brian Baek	Scott Jackson	Breton Saunders
	Paul Baker	Girault Jones	Reese Schreiber
	Michael Bonham	Keong Kam	Jay Sha
	Jonathan Brown	Kevin Keeler	Sascha Tietz
	Carlos Calderon	Min Kim	Jennifer Tsai
	Jason Chung	Woopoung Kim	Colin Whitby-Strevens
	David Conroy	Alan Kobayashi	Jeff Wilcox
	Bill Cornelius	Alexei Kosut	Eric Wiles
	Christophe Daniel	Christine Krause	Dan Wilson
	Raju Desai	Chris Ligtenberg	Dennis Yarak
	William Ferry	Matthew Mora	
Aptiv Consumer Connectivity	Andrew Burchett	Moe Elghrawi	
ASMedia Technology Inc.	Kuo Lung Li		
Bizlink Technology, Inc.	Alex Chou	CY Tsai	Giovanni Wang
	Morphy Hsieh	Kevin Tsai	

Cadence Design Systems, Inc.	Marcin Behrendt Huzaifa Dalal Pawel Eichler Sanjai Gandhi Sathish Kumar Ganesan	Dariusz Kaczmarczyk Tomasz Klimek Sanjeet Kumar Jie Min Asila Nahas Uyen Nguyen	Neelabh Singh Michal Staworko Fred Stivers Mark Summers Claire Ying
Canova Tech	Piergiorgio Beruto Andrea Maniero	Michael Marioli Antonio Orzelli	Paola Pilla Nicola Scantamburlo
China Telecommunications Technology Labs	Ming Wei		
Cirrus Logic Inc.	Sean Davis	Darren Holding	Brad Lambert
Corning Optical Communication LLC	Wojciech Giziewicz	Ian McKay	Jamie Silva
Cosemi Technologies Inc.	Samir Desai	Devang Parekh	
Credo Semiconductor, Inc.	Jim Bartenslager Yifei Dai Kevin Gao	Yasuo Hidaka Justin Lee Phil Sun	Ping Xiong
Cypress Semiconductor	Chia Hua Chang Mark Fu Naman Jain Savan Javia	Rushil Kadakia Benjamin Kropf Venkat Mandagulathur Anup Nayak	Jagadeesan Raj Sanjay Sancheti Subu Sankaran Anita Thimma Govarthanarajan
Dell	Mohammed Hijazi Rose Hou David Meyers Sean O'Neal	Ken Nicholas Marcin Nowak Ernesto Ramirez	Siddhartha Reddy Thomas Voor Merle Wood
Dialog Semiconductor (UK) Ltd.	Yimin Chen		
Diodes Incorporated	Kay Annamalai Justin Lee Paul Li	Bob Lo Jaya Shukla Qun Song	Jin-sheng Wang Ada Yip
DisplayLink (UK) Ltd.	Pete Burgers		
DJI Technology Co., Ltd.	Steve Huang		
eEver Technology, Inc.	Chien-Cheng Kuo		
Electronics Testing Center, Taiwan	Sophia Liu		
Elka International Ltd.	Roy Ting		
Ellisys	Abel Astley Rick Bogart	Wenhui Luo Mario Pasquali	Chuck Trefts Tim Wei
Etron Technology, Inc.	Chien-Cheng Kuo		
EverPro Technologies Company, Ltd.	Eva Chen	Hui Jiang	Silane Zhou
Feature Integration Technologies Inc.	Jacky Chan Chen Kris Yulin Lan	KungAn Lin Yuchi Tsao	Paul Yang Amanda Ying

Foxconn / Hon Hai	Patrick Casher	Fred Fons	Christine Tran
	Asroc Chen	Chris Goralka	Pei Tsao
	Brandon Chen	Bob Hall	AJ Yang
	Joe Chen	Chien-Ping Kao	Yuan Zhang
	Allen Cheng	Ji Li	Jessica Zheng
	Jason Chou	Ann Liu	Jie Zheng
	Edmond Choy	Terry Little	Andy Yao
	Shruti Deore	Steve Sedio	
Foxlink/Cheng Uei Precision Industry Co., Ltd.	Robert Chen	Armando Lee	Steve Tsai
	Sunny Chou	Dennis Lee	Wen Yang
	Carrie Chuang	Justin Lin	Wiley Yang
	Wen-Chuan Hsu	Robert Lu	Patrick Yeh
	Alex Hsue	Tse Wu Ting	Junjie Yu
Fresco Logic Inc.	Brian Collins	Bob McVay	Christopher Meyers
Google	Mehdi Abrishamchian	Mark Hayter	Vincent Palatin
	Naga Viswanadha	Eric Herrmann	Dylan Reid
	Udaya Kiran Ammu	Nithya Jagannathan	Adam Rodriguez
	Lucasz Bartosik	Nathan Kolluru	David Schneider
	Alec Berg	Lawrence Lam	Stephan Schooley
	Joshua Boilard	Adam Langley	Ketan Shringarpure
	Alec Berg	Benson Leung	Toshak Singhal
	Todd Broch	Abraham Levkoy	Bartosz Szpila
	Charlie Costakis	Ingrid Lin	Jameson Thies
	Nehemiah Dureus	Prashant Malani	Radu Vele
	Chao Fei	George-Daniel Matei	Ken Wu
	Jim Guerin	Richard Palatin	Songping Wu
	Jeffrey Hayashida		Diana Zigterman
Granite River Labs	Yung Han Ang	Mike Engbretson	Kristof Mommen
	Velmurugan Ayyakkannu	Mark Jiang	Krishna Murthy
	Rajesh B	Vishal Kakade	Amy Peng
	Pascal Berten	Jesson Li	Johnson Tan
	Hariprasad Bhat	Caspar Lin	Annie Tao
	Sandy Chang	Olivia Lu	Vasudev Uttharahalli
	Allen Chen	Bala M	Chin Hun Yaep
	Swee Guan Chua	Prasannakumar Marikunte Nagaraju	Mike Yang
	Alan Chuang	Winson Mok	Yun Yang
GuangDong OPPO Mobile Telecommunications Corp, Ltd.	Jerry Qin		
Honor Device Co., Ltd.	Wang Feng		
Hirose Electric Co., Ltd.	Jeremy Buan	Hyejun Lee	Gourgen Oganessyan
	Kohei Hida	Zhili Liang	Youngbon Park
	Kazunori Ichikawa	Sang-Muk Lim	Eungsoo Shin
	Sukkyu Jang	William MacKillop	Sid Tono
	William Kysiak		
Hosiden Corporation	Takahisa Otsuji	Fumitake Tamaki	

HP Inc. (USB Promoter company)	Lee Atkinson Srinath Balaraman Rami Bathaniah Roger Benson Alan Berkema	Robin Castell Steve Chen Michael Krause Rahul Lakdawala	Jim Mann Linden McClure Mike Pescetto Asjad Shamim
Huawei Technology Co., Ltd.	Jianhong He Zhengbing Li	Gang Liang Lauri Soderbacka	Ruinan Sun Liansheng Zheng
Hynetek Semiconductor Co., Ltd.	Yingyang Ou	James Xie	
I-PEX (Dai-ichi Seiko)	Alan Kinningham	Jack Ozeki	Ro Richard
Indie Semiconductor	Ian Board	Jim Wilshire	
Infineon Technologies	Godwin Gerald Arulappan Pradeep Baipai Dominic Gutierrez Anton Gutsul Naman Jain	Behzad Jamasbi Benjamin Kropf Venkat Mandagulathur Aniket Shashikant Mathad	Geona Mary Pullankannappally Devasia Shopitham Ram Gayathri Vasudevan Tue Fatt David Wee
Insight Test Labs	Caspar Lin		
Intel Corporation (USB Promoter company)	Dave Ackelson Nirmala Bailur Mike Bell Dmitriy Berchanskiy Brad Berlin Pierre Bossart Huimin Chen Kuan-Yu Chen Hengju Cheng Oded David Sourabh Das Jhuda Dayan Hem Doshi Paul Durley Saranya Gopal Venkataramani Gopalakrishnan Yaniv Hayat Howard Heck Hao-Han Hsu Seppo Ingalsuo Abdul (Rahman) Ismail James Jaussi	Luke Johnson Ziv Kabiry Vijaykumar Kadgi Jerzy Kolinski Venkataramana Kotakonda Rolf Kuhnisi Henrik Leegaard Edmond Lau Xiang Li Yun Ling Guobin Liu Steve McGowan Sankaran Menon Udaya Natarajan Aruni Nelson Chee Lim Nge Asutosh Pathak Sagar Pawar Duane Quiet Amardeep Rai Kannappan Rajaraman Sridharan Ranganathan	Rajaram Regupathy Oren Salomon Brad Saunders Tomer Savariego Ehud Shoor Amit Srivastava Einat Suriyan Eliahu Suued Ron Swartz Chuen Ming Tan David Thompson Pranav Tipnis Karthi Vadivelu Venkataramani Gopalakrishnan Tsion Vidal Stephanie Wallick Rafal Wielicki Devon Worrell Gal Yedidia Li Yuan Reza M. Zamani Vitaly Zhivov
Japan Aviation Electronics Industry Ltd. (JAE)	Kenji Hagiwara Hiroaki Ikeda Masaki Kimura Toshio Masumoto Kenta Minejima Toshiyuki Moritake Joe Motojima Ron Muir Norihiko Nakamura	Tadashi Okubo Kazuhiro Saito Kimiaki Saito Yuichi Saito Mark Saubert Toshio Shimoyama Tatsuya Shioda Atsuo Tago Masaaki Takaku	Jussi Takaneva Junichi Takeuchi Tomohiko Tamada Kentaro Toda Kouhei Ueda Takakazu Usami Masahide Watanabe Youhei Yokoyama

JPC/Main Super Inc.	Sam Tseng	Ray Yang	
Kandou Bus SA	Amanda Dong Majid Foodeei	Hitaish Sharma David Stauffer	Paul Wilson
Keysight Technologies Inc.	Soon Tatt Beh Fang Huey Lim	Jit Lim Pedro Merlo	Abhijeet Shinde Xiao Zhang
Kinetic Technologies, Inc.	Ramesh Dandapani Akihiro Moto	Jay Slivkoff Sireesha Vemulapalli	Kei Yasuno
Leadtrend	Yao-Wei Hsieh		
LeCroy Corporation	Mike Engbretson Daniel H. Jacobs	Tyler Joe Chetan Kopalle	Giuseppe Leccia
Lenovo	Rob Bowser Tomoki Harada	Jianye Li Wei Liu	Howard Locker Munefumi Nakata
LG Electronics Inc.	Won-Jong Choi Do Kyun Kim	Jeeyoung Kim Minji Kim	Seung Hyun Yoo
Lintes Technology Co., Ltd.	Tammy Huang Charles Kaun RD Lintes	Guobin Liu Max Lo CT Pien	JinYi Tu Jason Yang
Logitech Inc.	Niclas Granqvist		
Lotes Co., Ltd.	Ariel Delos Reyes Ernest Han Mark Ho Regina Liu-Hwang	Charles Kaun Chi-Chang Lin James Lin Max Lo	John Lynch Scott Shuey JinYi Tu Jason Yang
LSI Corporation	Dave Thompson		
Luxshare-ICT	Josue Castillo Daniel Chen Kenny Chen Lisen Chen Sally Chiu Blue Ho CY Hsu Alan Kinningham	Peter Jhou Antony Lin Gorden Lin John Lin Stone Lin Alan Liu Sean O'Neal	Scott Shuey James Stevens Carr Wang Yangli Wang Eric Wen Wayne Yeh Pat Young
Maxim Integrated Products	Forrest Christo Ken Helfrich	Sang Kim Jeff Lo	Michael Miskho Jacob Scott
MCCI Corporation	Terry Moore		
MediaTek Inc.	Henry Chen Yu-Yi Chu Yishu Hsieh Jason Hsu	Liang-Yu Hsu Jing-Hao Huang Shih Jyun-Yang Akan Lin	Alex YC Lin Tung-Sheng Lin Colin Tsai Chasel Wang
MegaChips Corporation	Alan Kobayashi	Satoru Kumashiro	
Mercedes-Benz Research & Development, North America, Inc.	Hans Wickler		
Microchip (SMSC)	Josh Averyt Mark Bohm Shannon Cash Thomas Farkas Fernando Gonzalez	Matthew Kalibat Donald Perkins Richard Petrie Mohammed Rahman Andrew Rogers	John Sisto Anthony Tarascio Kiet Tran Christopher Twigg Prasanna Vengateshan

Microsoft Corporation (USB Promoter company)	Randy Aull Jim Belesiu Michelle Bergeron Fred Bhesania Anthony Chen Philip Froese Vivek Gupta David Hargrove Robbie Harris Teemu Helenius Robert Hollyer Lily Huang Dan Iatco	Kai Inha Jayson Kastens Andrea Keating Shoaib Khan Patrick Law Eric Lee Ivan McCracken Arvind Murching Gene Obie Toby Nixon Arjun Padmanabhan Praveen Kumar Palacharla	Rahul Ramadas Srivatsan Ravindran Kiran Shastry Nathan Sherman Ugan Sivagnanenthirarajah Bala Sivakumar Timo Toivola Vahid Vassey David Voth Andrew Yang Panu Ylihaavisto
Molex LLC	Adib Al Abaji	Alan MacDougall	
Monolithic Power Systems	Junyong Gong Di Han Yuncong Jiang	Istvan Nagy Chris Sporck Ao Sun	Greg Virnig Vincent Wu
MQP Electronics Ltd.	Sten Carlsen	Pat Crowe	
Multilane Inc.	Fadi Daou	Rita Khawaja	Elias Khoury
NEC Corporation	Kenji Oguma		
Newnex Technology Corp.	Sam Liu	Nimrod Peled	
Nokia Corporation	Daniel Gratiot Pekka Leinonen	Samuli Makinen Pekka Talmola	Timo Toivola Panu Ylihaavisto
Nuvoton Technology Corp.	Nimrod Peled		
NVIDIA	Jamie Aitken	Jason Tsai	
NXP Semiconductors	Mahmoud EL Sabbagh Dennis Ha Jia Guo Ken Jaramillo	Nate Johnson Vijendra Kuroodi Roger Lo	Guru Prasad Wouter Sijm Krishnan TN
Oculus VR LLC	Amish Babu	Marty Evans	Joaquin Fierro
ON Semiconductor	Edward Berrios Rod Comer Alex Cornwall Eduardo De Reza	Alan Finkel Oscar Freitas Christian Klein Amir Lahooti	Eric Maier Rick Pierce Michael Smith
Parade Technologies, Inc.	Tom Burton Jian Chen Brian Collins	Reginald Conley Robert McVay Craig Wiley	Paul Xu Alan Yuen Jingfan Zhang
Phison Electronics Corp.	Wilson Wen		
Power Integrations	Shruti Anand Daryll Bonavente Rahul Joshi	Aditya Kulkarni Akshay Nayaknur	Amruta Patra Kaushik Raam

Qualcomm, Inc.	Lior Amarilio	James Goel	Vatsal Patel
	Vara Prasad Arikatla	Amit Gupta	Jack Pham
	Aris Balatsos	Philip Hardy	Vamsi Samavedam
	Nicholas Cadieux	Will Kun	Matthew Sienko
	David Chang	Jonathan Luty	Dmitrii Vasilchenko
	Tomer Ben Chen	Adi Menachem	Joshua Warner
	Richard Burrows	Lalan Mishra	Chris Wiesner
	Amit Gil	George Paparrizos	
Realtek Semiconductor Corp.	Marco Chiu	Ray Lee	Chuting Su
	Tsung-Peng Chuang	Jay Lin	Kyle Tsai
	Charlie Hsu	Ryan Lin	Jason Wang
	Fan-Hau Hsu	Terry Lin	Changhung Wu
	Ty Kingsmore		
Renesas Electronics Corp. (USB Promoter company)	Kai Bao	Philip Leung	Yosuke Sasaki
	Yen-Mo Chen	Mengfei Liu	Toshifumi Yamaoka
	Yimin Chen	Kiichi Muto	Fengshuan Zhou
	Nobuo Furuya	Ziba Nami	Yuhuan Zhou
	Tony Lai	Hajime Nozaki	
Resillion (Belgium NV)	Bart Grispen		
Richtek Technology Corp.	Ben Chiang	Max Huang	Ken Shih
	Tzuhsien Chuang	Leo Kang	Alex Yang
	Bryan Huang	Roger Lo	
Rohde & Schwarz GmbH & Co. KG	Curtis Donahue	Kai Uwe Schmidt	Randy White
	Patrick McKenzie	Martin Stumpf	
Rohm Co., Ltd.	Mark Aaldering	Yusuke Kondo	Yoshinori Ohwaki
	Kris Bahar	Arun Kumar	Takashi Sato
	Ruben Balbuena	Chris Lin	Hiroshi Yoshimura
	Nobutaka Itakura	Kazuomi Nagai	
Samsung Electronics Co., Ltd.	Jaedeok Cha	Sangwoo Kang	Jun Bum Lee
	KangSeok Cho	Wonseok Kang	Kyunghwan Lee
	Junhwa Choi	Koyoungwon Kim	Kyungwoo Lee
	Woojin Choi	Sangju Kim	Jinyoung Oh
	Yeongbok Choi	Soondo Kim	Chahoon Park
	Cheolyoon Chung	Woonki Kim	Chulwoo Park
	Ilhyung Chung	Jagoun Koo	Youngjin Park
	JaeRyong Han	Termi Kwon	Jung Waneui
	Jaehyeok Jang	Cheolho Lee	Sunggeun Yoon
	Sung Geun Joo	Edward Lee	
SB C&S Corp.	Toshinari Nakamura	Eiji Sakai	Kenji Watanabe
	Morito Ogikubo		
Scosche Industries	Josh Aguilera	Perry DeGuzman	Kevin Trejo
Seagate	Alvin Cox	Tony Priborsky	Dan Smith
	Emmanuel Lemay	Tom Skaar	Curtis Stevens
Shenzhen Deren Electronic Co., Ltd.	Smak (Zhudong) Huo	Yang Lirong	Lucy Zhang
	Wen Fa Lei	Valerie Yang	
Silicon Line GmbH	Ian Jackson	Boleslaw Wojtowicz	Zhiying Zhang

SiliConch Systems Private Limited	Jaswanth Ammineni Pavitra Balasubramanian Kaustubh Kumar Aniket Mathad	Shubham Paliwal Jinisha Patel Vinay Patel Rakesh Polasa	Vishnu Pusuluri Ranjith S Abhishek Sardeshpande Satish Anand Verkila
Simula Technology Inc.	John Chang Voss Cheng Thomas Li	Jung Lin Jyunming Lin Doris Liu	Richard Liu CK Wang Alice Yu
Softnautics LLP	Bhaves Desai Hetal Jariwala	Dipakkumar Modi Ishita Shah	Ujjwal Talati
Sony Corporation	Shinichi Hirata	Shigenori Tagami	
Spectra7 Microsystems Corp.	Andrew Kim	James McGrath	John Mitchell
SpecWerkz	Robert Dunstan Amanda Hosler	Diane Lenox Steve McGowan	Brad Saunders
STMicroelectronics (USB Promoter company)	Jerome Bach Nathalie Ballot Filippo Bonaccorso Guenael Cadier Dominique Chaillot Christophe Cochard Nicolas Florenchie	Cedric Force Gregory Gosciniak Chekib Hammami Joel Huloux Christophe Lorin Yohann Martiniault Patrizia Milazzo	Federico Musarra Pascal Legrand Richard O'Connor Massimo Panzica Nicolas Perrin Mathieu Rouviere
Sumitomo Electric Ind., Ltd.	Takeshi Inoue Yasuhiro Maeda	Wataru Sakurai Sainer Siagian	Masaki Suzuki Mitsuaki Tamura
Synaptics Inc.	Daniel Bogard Dan Ellis	Jeff Lukanc	Prashant Shamaraao
Synopsys, Inc.	Prishkrit Abrol Subramaniam Aravindhnan	Morten Christiansen Jaspreet Gambhir Nivin George	Satya Patnala John Stonick
Taiwan Testing and Certification Center	Dennis Chang	Sophia Liu	
Tektronix, Inc.	Sourabh Das Mark Guenther	Nitin Jhanwar Abhijeet Shinde	Randy White
Texas Instruments (USB Promoter company)	Jawaid Ahmad Kasthuri Annamalai Mike Campbell Prajith Cheerakkoda Greg Collins Gary Cooper Biju Erayamkot Panayamthatta Anant Gole GP Gopalakrishnan Craig Greenberg Richard Hubbard Katherine Huckabay Nate Johnson	Michael Koltun IV Kevin Kosta Yoon Lee Grant Ley Win Maung Shafiuddin Mohammed Lauren Moore Jacob Ontiveros Brian Parten Martin Patoka Jason Peck John Perry	Louis Peryea Brian Quach Sai Karthik Rajaraman Wes Ray Dafydd Roche Anwar Sadat Cory Stewart Neeta Thawani Sue Vining Bill Waters Deric Waters Gregory Watkins Roy Wojciechowski
The Silanna Group Pty. Ltd.	Tod Wolf		
Thine Electronica, Inc.	Yuseke Fujita	Shuhei Yamamoto	
Total Phase	Chris Yokum		

Trinity Co., Ltd.	Tetsushi Hoshikawa	Kinya Kishita	
Tyco Electronics Corp. (TE Connectivity Ltd.)	Max Chao	Mike Lockyer	Hiroshi Shirai
	Robert E. Cid	Jeff Mason	Hidenori Taguchi
	Calvin Feng	Takeshi Nakashima	Nathan Tracy
	Kengo Ijiri	Luis A. Navarro	Bernard Vetten
	Eiji Ikematsu	Masako Saito	Ryan Yu
	Joan Leu	Yoshiaki Sakuma	Noah Zhang
	Clark Li	Gavin Shih	Sjoerd Zwartkruis
UL LLC	Leo Chung	Dylan Su	Eric Wall
	Michael Hu	Henry Tsou	Lance Yang
	Terry Kao	Paul Vanderlaan	Chien-Wei Yeh
Unigraf OY	Steven Chen	Alexander Gushchin	Topi Lampiranta
	Sergey Grushin		
Varjo Technologies	Kai Inha		
Ventev Mobile	Brad Cox	Colin Vose	
VIA Technologies Inc.	Rex Hsu	Terrance Shih	Fong-Jim Wang
	Einsent Huang	Jay Tseng	
Vivo Mobile Communications Co. Ltd.	Benny Dong	Junchen Wei	Ziji Zhou
	Fangding Luo		
Weltrend Semiconductor	Hung Chiang	Wayne Lo	Eric Wu
	Jeng Cheng Liu	Ho Wen Tsai	
Western Digital	Dave Landsman	Rotem Sela	Prithviraj Sengupta
	Larry McMillan		
Xiaomi Communications Co., Ltd.	Canfeng Chen	Xiaoxing Yang	Qi Zhu
	Zhe Wang	Juejia Zhou	
Zhuhai iSmartWare Technology Co., Ltd.	Yuanchao Liang	Liu Qiong	Long Zhang

Pre-Release Draft Industry Reviewing Companies That Provided Feedback

Aces	JST Mfg. Co., Ltd.	Pericom
Fairchild Semiconductor	Korea Electric Terminal	Semtech Corporation
Fujitsu Ltd.	Marvell Semiconductor	Silicon Image
Industrial Technology Research Institute (ITRI)	Motorola Mobility LLC	SMK Corporation
Joinsoon Electronics Mfg. Co. Ltd.	PalCONN/PalNova (Palpilot International Corp.)	Toshiba Corporation

Revision History

Release	Date	Description
1.0	August 11, 2014	Initial Release
1.1	April 3, 2015	Reprint release including incorporation of all approved ECNs as of the revision date plus editorial clean-up.
1.2	March 25, 2016	Reprint release including incorporation of all approved ECNs as of the revision date plus editorial clean-up.
1.3	July 14, 2017	Reprint release including incorporation of all approved ECNs as of the revision date plus editorial clean-up.
1.4	March 29, 2019	Reprint release including incorporation of all approved ECNs as of the revision date plus editorial clean-up.
2.0	August 2019	New release primarily for enabling USB4 over USB Type-C connectors and cables. Also includes incorporation of all approved ECNs as of the revision date plus editorial clean-up.
2.1	May 2021	New release primarily for enabling Extended Power Range (EPR) and defining EPR cables aligning with USB Power Delivery Specification R3.1 V1.0. Also includes incorporation of all approved ECNs as the revision date plus editorial clean-up.
2.2	October 2022	New release primarily for enabling USB4 Version 2.0 (80 Gbps) over USB Type-C connectors and cables. Also includes incorporation of all approved ECNs as of the revision date plus editorial clean-up.
2.3	October 2023	New release primarily for <i>deprecating</i> the Audio Adapter Accessory Mode and <i>replacing it with</i> the Liquid Corrosion Mitigation Mode , and for updating the Multi-port Charger Shared Capacity definition and behaviors. Also includes incorporation of all other approved ECNs as of the revision date. Note: this release was created using a newly developed document template that includes some style adjustments and editorial clean-up.
2.4	October 2024	Reprint release including incorporation of all approved ECNs as of the revision date plus editorial clean-up.
<u>2.5</u>	<u>March 2026</u>	<u>Reprint release including incorporation of all approved ECNs as of the revision date plus editorial clean-up.</u>

1 Introduction

With the continued success of the USB interface, there exists a need to adapt USB technology to serve newer computing platforms and devices as they trend toward smaller, thinner, and lighter form-factors. Many of these newer platforms and devices are reaching a point where existing USB receptacles and plugs are inhibiting innovation, especially given the relatively large size and internal volume constraints of the Standard-A and Standard-B versions of USB connectors. Additionally, as platform usage models have evolved, usability and robustness requirements have advanced, and the existing set of USB connectors were not originally designed for some of these newer requirements. This specification establishes a new USB connector ecosystem that addresses the evolving needs of platforms and devices while retaining all the functional benefits of USB that form the basis for this most popular computing device interconnect.

1.1 Purpose

This specification defines the USB Type-C® receptacles, plug and cables.

The USB Type-C Cable and Connector Specification is guided by the following principles:

- Enable new and exciting host and device form-factors where size, industrial design and style are important parameters
- Work seamlessly with existing USB host and device silicon solutions
- Enhance ease of use for connecting USB devices with a focus on minimizing user confusion for plug and cable orientation

The USB Type-C Cable and Connector Specification defines a receptacle, plug, cable, and detection mechanisms that are compatible with existing USB interface electrical and functional specifications. This specification covers the following aspects that are needed to produce and use this new USB cable/connector solution in newer platforms and devices, and that interoperate with existing platforms and devices:

- USB Type-C receptacles, including electro-mechanical definition and performance requirements
- USB Type-C plugs and cable assemblies, including electro-mechanical definition and performance requirements
- USB Type-C to legacy cable assemblies and adapters
- USB Type-C-based device detection and interface configuration, including support for legacy connections
- **USB Power Delivery** optimized for the USB Type-C connector

The USB Type-C Cable and Connector Specification defines a standardized mechanism that supports **Alternate Modes**, such as repurposing the connector for docking-specific applications.

1.2 Scope

This specification is intended as a supplement to the existing **USB 2.0**, **USB 3.2**, **USB4®** and **USB Power Delivery** specifications. It addresses only the elements required to implement and support the USB Type-C receptacles, plugs and cables.

Normative information is provided to allow interoperability of components designed to this specification. **Informative** information, when provided, may illustrate possible design implementations.

1.3 Related Documents

USB 2.0 Universal Serial Bus Revision 2.0 Specification

This includes the entire document release package.

USB 3.2 Universal Serial Bus Revision 3.2 Specification

This includes the entire document release package.

USB 3.1 Legacy Cable and Connector Specification, Revision 1.0

USB4 USB4 Specification, Version 2.0, June 2023

(including posted errata and ECNs)

TBT3 Chapter 13 of USB4 Specification, Version 2.0, June 2023

USB PD USB Power Delivery Specification, Revision 2.0, Version 1.3, January 12, 2017

USB Power Delivery Specification, Revision 3.2, Version 1.1, ~~October 2024~~ 2, January 2026

(including posted errata and ECNs)

USB BB USB Billboard Device Class Specification, Revision 1.2.2, January 29, 2021

USB BC Battery Charging Specification, Revision 1.2, March 15, 2012

(including posted errata and ECNs)

DP AM DisplayPort™ Alt Mode on USB Type-C Standard, Version 2.1a, August 2024

All USB-specific documents are available for download at <http://www.usb.org/documents>.

The **DisplayPort Alt Mode** specification is available from VESA (<http://www.vesa.org>).

1.4 Conventions

1.4.1 Precedence

If there is a conflict between text, figures, and tables, the precedence **shall** be tables, figures, and then text.

1.4.2 Keywords

The following keywords differentiate between the levels of requirements and options.

1.4.2.1 Informative

Informative is a keyword that describes information with this specification that intends to discuss and clarify requirements and features as opposed to mandating them.

1.4.2.2 May

May is a keyword that indicates a choice with no implied preference.

1.4.2.3 May Not

May not is a keyword that is the inverse of **May**. Indicates a choice to not implement a given feature with no implied preference.

1.4.2.4 N/A

N/A is a keyword that indicates that a field or value is not applicable and has no defined value and **shall not** be checked or used by the recipient.

1.4.2.5 Normative

Normative is a keyword that describes features that are mandated by this specification.

1.4.2.6 Optional and Optionally

Optional and **Optionally** are equivalent keywords that describe features not mandated by this specification. However, if an **optional** feature is implemented, the feature **shall** be implemented as defined by this specification (**optional normative**).

1.4.2.7 Reserved

Reserved is a keyword indicating reserved bits, bytes, words, fields, and code values that are set-aside for future standardization. Their use and interpretation may be specified by future extensions to this specification and, unless otherwise stated, **shall not** be utilized, or adapted by vendor implementation. A reserved bit, byte, word, or field **shall** be set to zero by the sender and **shall** be ignored by the receiver. **Reserved** field values **shall not** be sent by the sender and, if received, **shall** be ignored by the receiver.

1.4.2.8 Shall

Shall is a keyword indicating a mandatory (**normative**) requirement. Designers are mandated to implement all such requirements to ensure interoperability with other compliant Devices.

1.4.2.9 Shall Not

Shall not is a keyword that is the inverse of **Shall** indicating non-compliant operation.

1.4.2.10 Should

Should is a keyword indicating flexibility of choice with a preferred alternative. Equivalent to the phrase “it is recommended that ...”.

1.4.2.11 Should Not

Should not is a keyword that is the inverse of **Should**. Equivalent to the phrase “it is recommended that implementations do not ...”.

1.4.3 Numbering

Numbers that are immediately followed by a lowercase “b” (e.g., 01b) are binary values. Numbers that are immediately followed by an uppercase “B” are byte values. Numbers that are immediately followed by a lowercase “h” (e.g., 3Ah) are hexadecimal values. Numbers not immediately followed by either a “b”, “B”, or “h” are decimal values.

1.5 Terms and Abbreviations

Term	Description
Accessory Mode	A reconfiguration of the connector based on the presence of Rd/Rd on CC1/CC2, respectively.
Active cable	Active cables are USB Full-Featured Type-C Cables that incorporate repeaters in the USB 3.2 data path. All active cables, regardless of length, are expected to comply with this specification, the USB 3.2 Appendix E, and the USB 3.2 active cable CTS. Active cables may incorporate repeaters in both ends of the cable, one end, or anywhere in the cable. See Chapter 6.
Alternate Mode	Operation defined by a vendor or standards organization that is associated with a SVID assigned by the USB-IF . Entry and exit into and from an Alternate Mode is controlled by the USB PD Structured VDM Enter Mode and Exit Mode commands. See Appendix E.
Alternate Mode Adapter (AMA)	A USB PD Device which supports Alternate Modes and acts as a UFP.

Term	Description
Assured Capacity Port	A charger port that, in terms has a dedicated allocation of USB PD, is either a Guaranteed Capability or Managed Capability power, i.e., the port that is always capable isn't part of delivering its Port Maximum PDPa Shared Capacity group of ports.
Audio Adapter Accessory Mode	The Accessory Mode defined by the presence of Ra/Ra on CC1/CC2, respectively. This mode is deprecated and replaced by Liquid Corrosion Mitigation Mode .
BMC	Biphase Mark Coding used for USB PD communication over the CC wire.
Cable Port Partner	The USB Type-C DRP, Source, or Sink connected to the cable plug.
Captive cable	A cable that is terminated on one end with a USB Type-C plug and has a vendor-specific connect means (hardwired or custom detachable) on the opposite end.
CC	Configuration Channel (CC) used in the discovery, configuration, and management of connections across a USB Type-C cable. See Section 4.5.
Charge-Through VPD (CTVPD)	A VCONN-Powered USB Device that has the mechanism to pass power and CC communication from one port to the other without any reregulation. See Section 4.10.2.
Configuration Lane	The USB 3.2 Configuration Lane is used to establish and manage dual-lane SuperSpeed USB operation. The Configuration Lane is specifically the SuperSpeed USB TX1/RX1 differential signal set in the cable/plug.
Debug Accessory Mode (DAM)	The Accessory Mode defined by the presence of Rd/Rd or Rp/Rp on CC1/CC2, respectively. See Appendix B.
Debug and Test System (DTS)	The combined hardware and software system that provides a system developer debug visibility and control when connected to a Target System in Debug Accessory Mode .
Default VBUS	VBUS voltage as defined by the USB 2.0 and USB 3.2 specifications. Note: where used, 5V VBUS connotes the same meaning.
DFP	Downstream Facing Port, specifically associated with the flow of data in a USB connection. Typically, the ports on a host or the ports on a hub to which devices are connected. In its initial state, the DFP sources VBUS and VCONN, and supports data. A charge-only DFP port only sources VBUS.
Direct connect device	A device with either a captive cable or just a USB Type-C plug (e.g., thumb drive).
DRD (Dual-Role-Data)	The acronym used in this specification to refer to a USB port that can operate as either a DFP (Host) or UFP (Device). The role that the port initially takes is determined by the port's power role at attach. A Source port takes on the data role of a DFP and a Sink port takes on the data role of a UFP. The port's data role may be changed dynamically using USB PD Data Role Swap.
DRP (Dual-Role-Power)	The acronym used in this specification to refer to a USB port that can operate as either a Source or a Sink. The role that the port offers may be fixed to either a Source or Sink or may alternate between the two port states. Initially when operating as a Source, the port will also take on the data role of a DFP and when operating as a Sink, the port will also take on the data role of a UFP. The port's power role may be changed dynamically using USB PD Power Role Swap.
DR_Swap	USB PD Data Role Swap.

Term	Description
Dual-lane (x2)	Dual-lane operation is defined as simultaneously signaling on both sets of USB transmit and receive differential pairs (TX1/RX1 and TX2/RX2 in the cable/plug).
Electronically Marked Cable	A USB Type-C cable that uses USB PD to provide the cable's characteristics. See Section 4.9.
eMarker	The element in an Electronically Marked Cable that returns information about the cable in response to a USB PD Discover Identity command.
Guaranteed Capability Port	As defined by the USB PD specification.
Hybrid Optical Active Cable	A cable that uses an intermediate optical transmission line for the high-speed signaling path (TX/RX) while retaining a metallic conductor-based solution for the rest of the defined interfaces, e.g., CC, USB 2.0 , SBUs, etc.
Initiator	The port initiating a Vendor Defined Message. It is independent of the port's PD role (e.g., Provider, Consumer, Provider/Consumer, or Consumer/Provider). In most cases, the Initiator will be a host.
Internal Temperature	In reference to an active cable, the temperature measured inside a plug. It is not the skin temperature. There is a relationship between the plug's internal temperature and the skin temperature, but that relationship is design dependent.
Liquid Corrosion Mitigation Mode	The mode defined by the presence of Ra/Ra on CC1/CC2, respectively. See Appendix A.
Local Plug	The cable plug being referred to.
Managed Capability Port	As defined by the USB PD specification.
Optically Isolated Active Cable (OIAC)	A cable that uses an intermediate optical transmission line for all signaling. This cable has no metallic conductors and is electrically isolated between the two plugs. This cable has a USB Type-C Plug on each end with one Cable Plug supporting SOP' and the other supporting SOP''.
Passive cable	A cable that does not incorporate any electronics to condition the data path signals. A passive cable may or may not be electronically marked.
Port Partner	Refers to the port (device or host) a port is attached to.
Power Bank	A device with a battery whose primary function is to charge or otherwise extend the runtime of other USB Type-C devices.
Power Delivery Mode	A mode where the port partners are in a USB PD power contract (either Explicit or Implicit).
Port Maximum PDP	As defined by the USB PD specification.
Port Present PDP	As defined by the USB PD specification.
Power Sinking Devices (PSD)	Sink which draws power but has no other USB or Alternate Mode communication function, e.g., a USB-powered light.
Powered cable	A cable with electronics in the plug that requires VCONN indicated by the presence of Ra between the VCONN pin and ground.
PR_Swap	USB PD Power Role Swap.

Term	Description
Re-driver	Re-driver refers to an analog component that operates on the signal without re-timing it. This <i>may</i> include equalization, amplification, and transmitter. The re-driver does not include a clock-data recovery (CDR) circuit. Re-drivers are beyond the scope of this document.
Remaining Shared Capacity	The remaining power available for a Shared Capacity group of ports after power has been allocated to one or more of its ports.
Remote Plug	A remote cable plug in the context of OIAC plugs is the plug at the other end of the Optically Isolated Active Cable.
Repeater	Repeater refers to any active component that acts on a signal in order to increase the physical lengths and/or interconnect loss over which the signal can be transmitted successfully. The category of repeaters includes both re-timers and re-drivers.
Responder	The port responding to the Initiator of a Vendor Defined Message (VDM). It is independent of the port's USB PD role (e.g., Provider, Consumer, Provider/Consumer, or Consumer/Provider). In most cases, the Responder will be a device.
Re-timer	Re-timer refers to a component that contains a clock-data recovery (CDR) circuit that “retimes” the signal. The re-timer latches the signal into a synchronous memory element before re-transmitting it. It is used to extend the physical length of the system without accumulating high frequency jitter by creating separate clock domains on either side of the re-timer. Re-timers are defined in USB 3.2 Appendix E and USB4 .
SBU	Sideband Use.
Shared Capacity Port	A charger port that, in terms of USB PD, is a Managed Capability port that is not always capable of delivering where its Port Maximum PDP allocation of power may be <u>constrained</u> due to it being part of a group of ports that share a common <u>power source</u> that is less than the sum of the individual port's Port Maximum PDP <u>power ratings</u> .
Shared Port Power Available	The power available, up to the port's PDP, to an unattached port in a Shared Capacity group of ports. This power represents what is available to each port in the group when a Sink is attached after considering power that is already allocated to ports with connected Sinks. This power will be a minimum of 7.5 W, initially offered as USB Type-C Current @ 1.5 A.
Short Active Cable (SAC)	A cable with a USB Type-C Plug on each end, at least one of which is a Cable Plug supporting SOP'. Cable length up to 5 meters.
SID	A Standard ID (SID) is a unique 16-bit value assigned by the USB-IF to identify an industry standard.
Single-lane (x1)	USB 3.2 single-lane operation is defined as signaling on only one set of SuperSpeed USB transmit and receive differential pairs (TX1/RX1 in the cable/plug).
Sink	Port asserting Rd on CC and when attached is consuming power from VBUS; most commonly a Device.
Skin Temperature	In reference to an active cable, the temperature of a plug's over-mold.
Source	Port asserting Rp on CC and when attached is providing power over VBUS; most commonly a Host or Hub DFP.
SVID	General reference to either a SID or a VID. Used by USB PD Structured VDMs when requesting SIDs and VIDs from a device.

Term	Description
Target System (TS)	The system being debugged in <i>Debug Accessory Mode</i> .
Total Shared Capacity	The total power available for a Shared Capacity group of ports. This is the overall power available when none of the ports in the group are connected to Sinks.
Type-A	A general reference to all versions of USB “A” plugs and receptacles.
Type-B	A general reference to all versions of USB “B” plugs and receptacles.
Type-C Plug	A USB plug conforming to the mechanical and electrical requirements in this specification.
Type-C Port	The USB port associated to a USB Type-C receptacle. This includes the USB signaling, CC logic, multiplexers, and other associated logic.
Type-C Receptacle	A USB receptacle conforming to the mechanical and electrical requirements of this specification.
UFP	Upstream Facing Port, specifically associated with the flow of data in a USB connection. The port on a device or a hub that connects to a host or the DFP of a hub. In its initial state, the UFP sinks VBUS and supports data.
USB 2.0 Type-C Cable	A USB Type-C to Type-C cable that only supports USB 2.0 data operation. This cable does not include USB 3.2 or SBU wires.
USB 2.0 Type-C Plug	A USB Type-C plug specifically designed to implement the USB 2.0 Type-C cable.
USB Full-Featured Type-C Cable	A USB Type-C to Type-C cable that supports USB 2.0 , USB 3.2 and USB4 data operation. This cable includes SBU wires and is an Electronically Marked Cable.
USB Full-Featured Type-C Plug	A USB Type-C plug specifically designed to implement the USB Full-Featured Type-C cable.
USB4 Hub	A USB4 hub product is used for USB port expansion, includes only USB upstream and downstream ports, and does not include any additional capability that exposes other connector types or functions except as defined in Section 5.2.3 (<i>Alternate Modes Support</i>).
USB4 -based Dock	A USB4 -based dock product combines a USB4 hub (including at least one exposed USB Type-C downstream port) with additional capabilities that either exposes other connector types and/or includes other user-visible functions, e.g., storage, networking, etc. Examples of functions that are not considered user-visible include firmware update and device authentication.
USB Safe State	The USB Safe State as defined by the USB PD specification.
<i>VCONN-Powered Accessory (VPA)</i>	An accessory that is powered from VCONN to operate in an <i>Alternate Mode</i> . <i>VPAs</i> cannot implement the charge-through mechanism described for <i>VPDs</i> , and instead must intermediate by negotiating USB Power Delivery with both the connected host and source in order to enable similar functionality. See Section 4.10.1.
<i>VCONN-Powered USB Device (VPD)</i>	A USB direct-connect or captive-cable device that can be powered solely from either VCONN or VBUS. <i>VPDs may optionally</i> support the <i>VPD</i> charge-through capability. See Section 4.10.2.
<i>VCONN_Swap</i>	USB PD VCONN Swap.
VDM	Vendor Defined Message as defined by the USB PD specification.

Term	Description
VID	A Vendor ID (VID) is a unique 16-bit value assigned by the USB-IF to identify a vendor.
<i>vSafe0V</i>	VBUS “0 volts” as defined by the USB PD specification.
<i>vSafe5V</i>	VBUS “5 volts” as defined by the USB PD specification.
x1	See Single-lane.
x2	See Dual-lane.

2 Overview

2.1 Introduction

The USB Type-C® receptacle, plug and cable provide a smaller, thinner, and more robust alternative to legacy USB interconnect (Standard and Micro USB cables and connectors). This solution targets use in very thin platforms, ranging from ultra-thin notebook PCs down to smart phones where existing Standard-A and Micro-AB receptacles are deemed too large, difficult to use, or inadequately robust.

Some key specific enhancements include:

- The USB Type-C receptacle may be used in very thin platforms as its total system height for the mounted receptacle is under 3 mm
- The USB Type-C plug enhances ease of use by being plug-able in either upside-up or upside-down directions
- The USB Type-C cable enhances ease of use by being plug-able in either direction between host and devices

While the USB Type-C interconnect no longer physically differentiates plugs on a cable by being an A-type or B-type, the USB interface still maintains such a host-to-device logical relationship. Determination of this host-to-device relationship is accomplished through a Configuration Channel (CC) that is connected through the cable. In addition, the Configuration Channel is used to set up and manage power and Alternate/Accessory Modes.

Using the Configuration Channel, the USB Type-C interconnect defines a simplified 5 volt VBUS-based power delivery and charging solution that supplements what is already defined in the **USB 3.2** Specification. More advanced power delivery and battery charging features over the USB Type-C interconnect are based on the **USB Power Delivery** specification. As a product implementation improvement, the USB Type-C interconnect shifts the **USB PD** communication protocol from being communicated over VBUS to being delivered across the USB Type-C Configuration Channel.

The USB Type-C receptacle, plug and cable designs are intended to support future USB functional extensions. As such, consideration was given to frequency scaling performance, pin-out arrangement and the configuration mechanisms when developing this solution. The definition of future USB functional extensions is not in the scope of this specification but rather will be provided in future releases of the base USB Specification, i.e., beyond the existing **USB4** Specification.

Figure 2-1 illustrates the comprehensive functional signal plan for the USB Full-Featured Type-C receptacle, not all signals shown are required in all platforms or devices. As shown, the receptacle signal list functionally delivers both **USB 2.0** (D+ and D-) and either **USB 3.2** or **USB4** (TX and RX pairs) data buses, USB power (VBUS) and ground (GND), Configuration Channel signals (CC1 and CC2), and two Sideband Use (SBU) signal pins. Multiple sets of USB data bus signal locations in this layout facilitate being able to functionally map the USB signals independent of plug orientation in the receptacle. For reference, the signal pins are labeled. For the **USB 2.0** Type-C receptacle, neither the **USB 3.2** nor **USB4** signals are implemented.

Figure 2-1 USB Type-C Receptacle Interface (Front View)

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12
GND	TX1+	TX1-	VBUS	CC1	D+	D-	SBU1	VBUS	RX2-	RX2+	GND
GND	RX1+	RX1-	VBUS	SBU2	D-	D+	CC2	VBUS	TX2-	TX2+	GND
B12	B11	B10	B9	B8	B7	B6	B5	B4	B3	B2	B1

Figure 2-2 illustrates the comprehensive functional signal plan for the USB Type-C plug. Only one CC pin is connected through the cable to establish signal orientation, and the other CC pin is repurposed as VCONN for

powering electronics in the USB Type-C plug. Also, only one set of **USB 2.0** D+/D– wires are implemented in a USB Type-C cable. For USB Type-C cables that only intend to support **USB 2.0** functionality, the TX/RX and SBU signals are not implemented. For the USB Type-C Power-Only plug (intended only for USB Type-C Sink applications), only nine contacts are implemented to support CC, VBUS, and GND.

Figure 2-2 USB Full-Featured Type-C Plug Interface (Front View)

A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
GND	RX2+	RX2–	VBUS	SBU1	D–	D+	CC	VBUS	TX1–	TX1+	GND
GND	TX2+	TX2–	VBUS	VCONN			SBU2	VBUS	RX1–	RX1+	GND
B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12

2.2 USB Type-C Receptacles, Plugs and Cables

Cables and connectors, including USB Type-C to USB legacy cables and adapters, are explicitly defined within this specification. These are the only connectors and cables that are authorized by the licensing terms of this specification. All licensed cables and connectors are required to comply with the compliance and certification requirements that are developed and maintained by the **USB-IF**.

The following USB Type-C receptacles and plugs are defined.

- USB Full-Featured Type-C receptacle for **USB 2.0**, **USB 3.2**, **USB4** and full-featured platforms and devices.
- **USB 2.0** Type-C receptacle for **USB 2.0** platforms and devices.
- USB Type-C Power-Only receptacle.
- USB Full-Featured Type-C plug.
- **USB 2.0** Type-C plug.
- USB Type-C Power-Only plug.

The following USB Type-C cables are defined.

- USB Full-Featured Type-C cable with a USB Full-Featured Type-C plug at both ends for **USB 3.2**, **USB4** and full-featured applications.
- **USB 2.0** Type-C cable with a **USB 2.0** Type-C plug at both ends for **USB 2.0** applications.
- Captive cable with either a USB Full-Featured Type-C plug or **USB 2.0** Type-C plug at one end.
- Active cables as defined in Chapter 6.

All of the defined USB Type-C receptacles, plugs and cables (except OIAC) support USB charging applications, including support for the **optional** USB Type-C-specific implementation of the **USB Power Delivery** specification (See Section 4.6.2.4).

All USB Full-Featured Type-C cables are electronically marked. **USB 2.0** Type-C cables **may** be electronically marked. See Section 4.9 for the requirements of Electronically Marked Cables.

The following USB Type-C to USB legacy cables and adapters are defined.

- **USB 3.2** Type-C to Legacy Host cable with a USB Full-Featured Type-C plug at one end and a **USB 3.1** Standard-A plug at the other end – this cable supports use of a USB Type-C-based device with a legacy USB host.

- **USB 2.0** Type-C to Legacy Host cable with a **USB 2.0** Type-C plug at one end and a **USB 2.0** Standard-A plug at the other end – this cable supports use of a USB Type-C-based device with a legacy **USB 2.0** host (primarily for mobile charging and sync applications).
- **USB 3.2** Type-C to Legacy Device cable with a USB Full-Featured Type-C plug at one end and a **USB 3.1** Standard-B plug at the other end – this cable supports use of legacy **USB 3.1** hubs and devices with a USB Type-C-based host.
- **USB 2.0** Type-C to Legacy Device cable with a **USB 2.0** Type-C plug at one end and a **USB 2.0** Standard-B plug at the other end – this cable supports use of legacy **USB 2.0** hubs and devices with a USB Type-C-based host.
- **USB 2.0** Type-C to Legacy Mini Device cable with a **USB 2.0** Type-C plug at one end and a **USB 2.0** Mini-B plug at the other end – this cable supports use of legacy devices with a **USB 2.0** Type-C-based host.
- **USB 3.2** Type-C to Legacy Micro Device cable with a USB Full-Featured Type-C plug at one end and a **USB 3.1** Micro-B plug at the other end – this cable supports use of legacy **USB 3.1** hubs and devices with a USB Type-C-based host.
- **USB 2.0** Type-C to Legacy Micro Device cable with a **USB 2.0** Type-C plug at one end and a **USB 2.0** Micro-B plug at the other end – this cable supports use of legacy **USB 2.0** hubs and devices with a USB Type-C-based host.
- **USB 3.2** Type-C to Legacy Standard-A adapter with a USB Full-Featured Type-C plug at one end and a **USB 3.1** Standard-A receptacle at the other end – this adapter supports use of a legacy USB “thumb drive” style device or a legacy USB ThinCard device with a **USB 3.2** Type-C-based host.
- **USB 2.0** Type-C to Legacy Micro-B adapter with a **USB 2.0** Type-C plug at one end and a **USB 2.0** Micro-B receptacle at the other end – this adapter supports charging a USB Type-C-based mobile device using a legacy USB Micro-B-based chargers, either captive cable-based or used in conjunction with a legacy **USB 2.0** Standard-A to Micro-B cable.

USB Type-C receptacle to USB legacy adapters are explicitly not defined or allowed. Such adapters would allow many invalid, and potentially unsafe, cable connections to be constructed by users.

2.3 Configuration Process

The USB Type-C receptacle, plug and cable solution incorporates a configuration process to detect a downstream facing port to upstream facing port (Source-to-Sink) connection for VBUS management and host-to-device connected relationship determination.

The USB Type-C port configuration process is used for the following:

- Source-to-Sink attach/detach detection,
- plug orientation/cable twist detection,
- initial power (Source-to-Sink) detection and establishing the data (Host-to-Device) relationship,
- detect if cable requires VCONN,
- USB Type-C VBUS current detection and usage,
- **USB PD** communication, and
- discovery and configuration of functional extensions.

Two pins on the USB Type-C receptacle, CC1 and CC2, are used for this purpose. Within a standard USB Type-C cable, only a single CC pin position within each plug of the cable is connected through the cable.

2.3.1 Source-to-Sink Attach/Detach Detection

Initially, Source-to-Sink attach is detected by a host or hub port (Source) when one of the CC pins at its USB Type-C receptacle senses a specified resistance to GND. Subsequently, Source-to-Sink detach is detected when the CC pin that was terminated at its USB Type-C receptacle is no longer terminated to GND.

Power is not applied to the USB Type-C host or hub receptacle (VBUS or VCONN) until the Source detects the presence of an attached device (Sink) port. When a Source-to-Sink attach is detected, the Source is expected to enable power to the receptacle and proceed to normal USB operation with the attached device. When a Source-to-Sink detach is detected, the port sourcing VBUS removes power.

2.3.2 Plug Orientation/Cable Twist Detection

The USB Type-C plug can be inserted into a receptacle in either one of two orientations, therefore the CC pins enable a method for detecting plug orientation in order to determine the lane ordering of the USB TX/RX data signal pairs functionally connected through the cable and identify the Configuration Lane for dual-lane operation when supported. This allows for signal routing, if needed, within a host or device to be established for a successful connection.

2.3.3 Initial Power (Source-to-Sink) Detection and Establishing the Data (Host-to-Device) Relationship

Unlike existing USB Type-A and USB Type-B receptacles and plugs, the mechanical characteristics of the USB Type-C receptacle and plug do not inherently establish the relationship of USB host and device ports. The CC pins on the receptacle also serve to establish an initial power (Source-to-Sink) and data (Host-to-Device) relationships prior to the normal USB enumeration process.

For the purpose of defining how the CC pins are used to establish the initial power relationship, the following port power behavior modes are defined.

1. Source-only – for this mode, the port exclusively behaves as a Source.
2. Sink-only – for this mode, the port exclusively behaves as a Sink.
3. Dual-Role-Power (DRP) – for this mode, the port can behave either as a Source or Sink.

Additionally, when a port supports USB data operation, a port's data behavior modes are defined.

1. DFP-only – for this mode, the port exclusively behaves as a DFP.
2. UFP-only – for this mode, the port exclusively behaves as a UFP.
3. Dual-Role-Data (DRD) – for this mode, the port can behave either as a DFP or UFP.

The DFP-only and UFP-only ports behaviorally map to traditional USB host ports and USB device ports, respectively but **may not** necessarily do USB data communication. When a host-only port is attached to a device-only port, the behavior from the user's perspective follows the traditional USB host-to-device port model. However, the USB Type-C connector solution does not physically prevent host-to-host or device-to-device connections. In this case, the resulting host-to-host or device-to-device connection results in a safe but non-functional situation.

Once initially established, the Source supplies VBUS and behaves as a DFP, and the Sink consumes VBUS and behaves as a UFP. **USB PD**, when supported by both ports, **may** then be used to independently swap both the power and data roles of the ports.

A port that supports dual-role operation by being able to shift to the appropriate connected mode when attached to either a Source-only or Sink-only port is a DRP. In the special case of a DRP being attached to another DRP, an initialization protocol across the CC pins is used to establish the initial host-to-device relationship. Given no role-swapping intervention, the determination of which is DFP or UFP is random from the user's perspective.

Two independent sets of mechanisms are defined to allow a USB Type-C DRP to functionally swap power and data roles. When **USB PD** is supported, power and data role swapping is performed as a subsequent step following the initial connection process. For non-PD implementations, power/data role swapping can **optionally** be dealt with as part of the initial connection process. To improve the user's experience when connecting devices that are of categorically different types, products **may** be implemented to strongly prefer being a DFP or a UFP, such that the DFP/UFP determination becomes predictable when connecting two DRPs of differing categories. See Section 4.5.1.4 for more on available swapping mechanisms.

As an alternative to role swapping, a USB Type-C DRP **may** provide useful functionality by when operating as a host, exposing a CDC/network (preferably TCP/IP) stack or when operating as a device, exposing a CDC/network interface.

USB hubs have two types of ports, a UFP that is connected to a DFP (host or another hub) that initially functions as a Sink, and one or more DFPs for connecting other devices that initially function as Sources.

2.3.4 USB Type-C VBUS Current Detection and Usage

With the USB Type-C connector solution, a Source (host or downstream hub port) **may** implement higher source current over VBUS to enable faster charging of mobile devices or higher-power devices. All USB host and hub ports advertise via the CC pins the level of current that is presently available. The USB device port is required to manage its load to stay within the current level offered by the host or hub, including dynamically scaling back the load if the host or hub port changes its advertisement to a lower level as indicated over the CC pins.

Three current level advertisements at 5V VBUS are defined by **USB Type-C Current**:

- **USB Type-C Current** at Default is the as-configured for high-power operation current value as defined by a USB Specification (500 mA for **USB 2.0** ports; 900 mA or 1,500 mA for **USB 3.2** ports operating in single-lane or dual-lane, respectively),
- **USB Type-C Current** at 1.5 A, and
- **USB Type-C Current** at 3.0 A.

There is a clear functional distinction between advertising **USB Type-C Current** at Default versus the **USB Type-C Current** at either 1.5 A or 3.0 A.

- **USB Type-C Current** at Default is intended for host operation in providing bus power to a connected device where the host manages the device's current consumption for the low-power, high-power and suspend states as defined in the USB base specifications.
- **USB Type-C Current** at either 1.5 A or 3.0 A is primarily intended for charging applications. The Sink can vary its current draw up to the advertised limit. Offering **USB Type-C Current** at either 1.5 A or 3.0 A is allowed for a host providing bus power to a device. The host needs to assume that the device will continuously draw up to the offered limit.

The higher **USB Type-C Current** levels that can be advertised allows hosts and devices that do not implement **USB PD** to take advantage of higher charging current.

2.3.5 USB PD Communications

USB Power Delivery is a feature on products (hosts, hubs, and devices). **USB PD** communications is used to:

- establish power contracts that allow voltage and current beyond existing USB data bus specifications,
- change the port sourcing VBUS,
- change the port sourcing VCONN,
- swap DFP and UFP roles, and
- communicate with cables.

The **USB PD** Bi-phase Mark Coded (BMC) communications are carried on the CC wire of the USB Type-C cable.

2.3.6 Functional Extensions

Functional extensions, including **USB4** and **Alternate Modes** (see Appendix E) are enabled via a communications channel across CC using methods defined by the **USB Power Delivery** specification.

2.4 VBUS

VBUS provides a path to deliver power between a host and a device, and between a USB power charger and a host/device. A simplified high-current supply capability is defined for hosts and chargers that **optionally** support current levels beyond the **USB 2.0**, **USB 3.2**, and **USB4** specifications. The **USB Power Delivery** specification is supported.

Table 2-1 summarizes the power supply options available from the perspective of a device with the USB Type-C connector. Not all options will be available to the device from all host or hub ports – only the first two listed options are mandated by the base USB specifications and form the basis of **USB Type-C Current** at the Default USB Power level.

Table 2-1 Summary of power supply options

Mode of Operation	Voltage	Current	Notes
USB 2.0	5 V	See USB 2.0	
USB 3.2	5 V	See USB 3.2	
USB4	5 V	1.5 A	See Section 5.3.
USB BC 1.2	5 V	1.5 A ¹	Legacy charging
USB Type-C Current @ 1.5 A	5 V	1.5 A	Supports higher power devices
USB Type-C Current @ 3.0 A	5 V	3 A	Supports higher power devices
USB PD	Configurable up to 48 V	Configurable up to 5 A	Directional control and power level management

Note 1: Whereas **USB BC 1.2** specification permits a power provider to be designed to support a level of power between 0.5 A and 1.5 A, the USB Type-C specification requires that a Source port that supports **USB BC 1.2** be at a minimum capable of supplying 1.5 A and advertise **USB Type-C Current** @ 1.5 A in addition to supporting the **USB BC 1.2** power provider termination.

The USB Type-C receptacle is specified for current capability of 5 A whereas standard USB Type-C cable assemblies are rated for 3 A. The higher rating of the receptacle enables systems to deliver more power over directly attached docking solutions or using appropriately designed chargers with captive cables when implementing **USB PD**. Also, USB Type-C cable assemblies designed for **USB PD** and appropriately identified via electronic marking are allowed to support up to 5 A.

USB Type-C cable assemblies designed for **USB PD** Extended Power Range (EPR) operation are required to have an electronic marking indicating EPR compatibility. These cables are required to be electronically marked for 5 A and 50 V and include the EPR Mode Capable bit set.

2.5 VCONN

Once the connection between host and device is established, the CC pin (CC1 or CC2) in the receptacle that is not connected via the CC wire through the standard cable is repurposed to source VCONN to power circuits in the plug needed to implement Electronically Marked Cables (see Section 4.9), **VCONN-Powered Accessories** and **VCONN-Powered USB Devices**. Initially, the source supplies VCONN and the source of VCONN **may** be swapped using **USB PD VCONN_Swap**.

Once VCONN is available, all electronically marked cables use it as the only power source. If VCONN is applied after VBUS, then until VCONN is available, the cable *may* remain unpowered or *may* draw power from VBUS.

VCONN functionally differs from VBUS in that it is isolated from the other end of the cable. VCONN is independent of VBUS and, unlike VBUS which can use **USB PD** to support higher voltages, VCONN voltage stays within the range of 3.0 to 5.5 V.

2.6 Hubs

USB hubs implemented with USB Type-C receptacles are required to clearly identify the upstream facing port. This requirement is needed because a user can no longer know which port on a hub is the upstream facing port and which ports are the downstream facing ports by the type of receptacles that are exposed, i.e., USB Type-B is the upstream facing port and USB Type-A is a downstream facing port.

3 Mechanical

3.1 Overview

This chapter defines the USB Type-C connectors and wired cable assemblies. Cables which include active elements in the data path are defined in Chapter 6 (Active Cables).

3.1.1 Compliant Connectors

The USB Type-C specification defines the following standard connectors:

- USB Full-Featured Type-C receptacle,
- **USB 2.0** Type-C receptacle,
- USB Type-C Power-Only receptacle,
- USB Full-Featured Type-C plug,
- **USB 2.0** Type-C plug, and
- USB Type-C Power-Only plug.

3.1.2 Compliant Cable Assemblies

Table 3-1 summarizes the USB Type-C standard cable assemblies along with the primary differentiating characteristics. All USB Full-Featured Type-C cables **shall** support simultaneous, independent signal transmission on both **USB 3.2** and **USB4** (TX and RX pairs) data buses. The cable lengths listed in the table are **informative** and represent the practical length based on cable performance requirements. For **USB Power Delivery**, each cable assembly is identified as being either only usable for **USB PD** Standard Power Range (SPR) operation or usable for both **USB PD** SPR and **USB PD** Extended Power Range (EPR) operation. Existing **USB PD** SPR 5 A cables have been deprecated and replaced by **USB PD** EPR cables. All cables that are either full-featured and/or are rated at more than 3 A current are Electronically Marked Cables.

Table 3-1 USB Type-C Standard Cable Assemblies

Cable Ref	Plug 1	Plug 2	USB Version	Cable Length ²	Current Rating	USB Power Delivery	USB Type-C Electronically Marked
CC2-3	C	C	USB 2.0	≤ 4 m	3 A	Supported (SPR Only)	Optional
CC2-5 ¹					5 A		Required
CC2-5E					5 A	Supported (SPR & EPR)	Required
CC3G1-3	C	C	USB 3.2 Gen1 and USB4 Gen2	≤ 2 m	3 A	Supported (SPR Only)	Required
CC3G1-5 ¹					5 A		
CC3G1-5E					5 A	Supported (SPR & EPR)	Required
CC3G2-3	C	C	USB 3.2 Gen2 and USB4 Gen2	≤ 1 m	3 A	Supported (SPR Only)	Required
CC3G2-5 ¹					5 A		
CC3G2-5E					5 A	Supported (SPR & EPR)	Required
CC4G3-3	C	C	USB4 Gen3 and USB4 Gen4 ³	≤ 0.8 m	3 A	Supported (SPR Only)	Required
CC4G3-5 ¹					5 A		
CC4G3-5E					5 A	Supported (SPR & EPR)	Required
Notes: 1. These cables are deprecated in favor of having all 5 A cables be EPR-capable versions. 2. The cable lengths listed in the table are <i>informative</i> and represent the practical lengths based on cable performance requirements. 3. <i>USB4</i> Gen4 passive cables are electrically equivalent to <i>USB4</i> Gen3 passive cables. For <i>USB4</i> Gen4 active cables, refer to Chapter 6 for the <i>USB4</i> Gen3 and Gen4 electrical requirements.							

USB Type-C products are also allowed to have a captive cable. See Section 3.4.3.

3.1.3 Compliant USB Type-C to Legacy Cable Assemblies

Table 3-2 summarizes the USB Type-C legacy cable assemblies along with the primary differentiating characteristics. The cable lengths listed in the table are *informative* and represent the practical length based on cable performance requirements. All USB Type-C to legacy cable assemblies are only defined specific to **USB 2.0** and **USB 3.2** as there are no USB Type-C to legacy cables that are uniquely applicable to **USB4**.

For USB Type-C legacy cable assemblies that incorporate *R_p* termination, the value of this termination is required to be specified to the Default setting of **USB Type-C Current** even though the cable assemblies are rated for 3 A. The *R_p* termination in these cables is intended to represent the maximum current of a compliant legacy USB host port, not the current rating of the cable itself. The cable current rating is intentionally set to a higher level given that there are numerous non-standard power chargers that offer more than the Default levels established by the **USB 2.0** and **USB 3.1** specifications.

Table 3-2 USB Type-C Legacy Cable Assemblies

Cable Ref	Plug 1 ³	Plug 2 ³	USB Version	Cable Length	Current Rating
AC2-3	USB 2.0 Standard-A	USB 2.0 Type-C ¹	USB 2.0	≤ 4 m	3 A
AC3G1-3	USB 3.1 Standard-A	USB Full-Featured Type-C ¹	USB 3.1 Gen1	≤ 2 m	3 A
AC3G2-3	USB 3.1 Standard-A	USB Full-Featured Type-C ¹	USB 3.1 Gen2	≤ 1 m	3 A
CB2-3	USB 2.0 Type-C ²	USB 2.0 Standard-B	USB 2.0	≤ 4 m	3 A
CB3G2-3	USB Full-Featured Type-C ²	USB 3.1 Standard-B	USB 3.1 Gen2	≤ 1 m	3 A
CmB2	USB 2.0 Type-C ²	USB 2.0 Mini-B	USB 2.0	≤ 4 m	500 mA
CμB2-3	USB 2.0 Type-C ²	USB 2.0 Micro-B	USB 2.0	≤ 2 m	3 A
CμB3G1-3	USB Full-Featured Type-C ²	USB 3.1 Micro-B	USB 3.1 Gen1	≤ 2 m	3 A
CμB3G2-3	USB Full-Featured Type-C ²	USB 3.1 Micro-B	USB 3.1 Gen2	≤ 1 m	3 A
Notes:					
<ol style="list-style-type: none"> 1. USB Type-C plugs associated with the “B” end of a legacy adapter cable are required to have Rp (56 kΩ ± 5%) termination incorporated into the plug assembly – see Section 4.5.3.2.2. 2. USB Type-C plugs associated with the “A” end of a legacy adapter cable are required to have Rd (5.1 kΩ ± 20%) termination incorporated into the plug assembly – see Section 4.5.3.2.1. 3. Refer to Section 3.7.5.3 for the mated resistance and temperature rise required for the legacy plugs. 					

3.1.4 Compliant USB Type-C to Legacy Adapter Assemblies

Table 3-3 summarizes the USB Type-C legacy adapter assemblies along with the primary differentiating characteristics. The cable lengths listed in the table are **informative** and represent the practical length based on cable performance requirements. All USB Type-C to legacy adapter assemblies are only defined specific to **USB 2.0** and **USB 3.2** as there are no USB Type-C to legacy adapters that are uniquely applicable to **USB4**.

Table 3-3 USB Type-C Legacy Adapter Assemblies

Cable Ref	Plug	Receptacle ³	USB Version	Cable Length	Current Rating
CμBR2-3	USB 2.0 Type-C ¹	USB 2.0 Micro-B	USB 2.0	≤ 0.15 m	3 A
CAR3G1-3	USB Full-Featured Type-C ²	USB 3.1 Standard-A	USB 3.1 Gen1	≤ 0.15 m	3 A
Notes:					
<ol style="list-style-type: none"> 1. USB Type-C plugs associated with the “B” end of a legacy adapter are required to have Rp (56 kΩ ± 5%) termination incorporated into the plug assembly – see Section 4.5.3.2.2. 2. USB Type-C plugs associated with the “A” end of a legacy adapter are required to have Rd (5.1 kΩ ± 20%) termination incorporated into the plug assembly – see Section 4.5.3.2.1. 3. Refer to Section 3.7.5.3 for the mated resistance and temperature rise required for the legacy receptacles. 					

3.2 USB Type-C Connector Mating Interfaces

This section defines the connector mating interfaces, including the connector interface drawings, pin assignments, and descriptions. All dimensions in figures are in millimeters.

3.2.1 Interface Definition

Figure 3-1 and Figure 3-3 show, respectively, the USB Type-C receptacle and USB Full-Featured Type-C plug interface dimensions.

Figure 3-12 shows the **USB 2.0** Type-C plug interface dimensions. The dimensions that govern the mating interoperability are specified. All the REF dimensions are provided for reference only, not hard requirements.

Key features, configuration options, and design areas that need attention:

1. Figure 3-1 shows a vertical-mount receptacle. Other PCB mounting types such as right-angle mount and mid-mount are allowed.
2. A mid-plate is required between the top and bottom signals inside the receptacle tongue to manage crosstalk in full-featured applications. The mid-plate **shall** be connected to the PCB ground with at least two grounding points. The mid-plate **shall** be designed such that plug pins A4, A5, A6, A7, A8, A9, and B4, B5, B6, B7, B8, B9 do not short to ground during the connector mating process with an effective 6.2 mm receptacle shell implementation. If the receptacle connector has a short shell or no shell, the connector manufacturer **shall** provide an effective length shell fixture for compliance testing. A reference design of the mid-plate is provided in Section 3.2.2.1.
3. Retention of the cable assembly in the receptacle is achieved by the side-latches in the plug and features on the sides of the receptacle tongue. Side latches are required for all plugs except plugs used for docking with no cable attached. Side latches **shall** be connected to ground inside the plug. A reference design of the side latches is provided in Section 3.2.2.2 along with its grounding scheme. Docking applications **may not** have side latches, requiring special consideration regarding EMC (Electromagnetic Compatibility).
4. EMC shielding springs are required inside the cable plug. The shielding spring **shall** be connected to the plug shell. No EMC shielding spring finger tip of the USB Full-Featured Type-C plug or **USB 2.0** Type-C plug **shall** be exposed in the plug housing opening of the unmated USB Type-C plug (see Figure 3-13). Section 3.2.2.3 shows reference designs of the EMC spring.
5. Shorting of any signal or power contact spring to the plug metal shell is not allowed. The spring in the deflected state **should not** touch the plug shell. An isolation layer (e.g., Kapton tape placed on the plug shell) is recommended to prevent accidental shorting due to plug shell deformation.
6. The USB Type-C receptacle **shall** provide an EMC ground return path through one of the following options:
 - a system of specific points of contact on the receptacle outer shell (e.g., spring fingers or spring fingers and formed solid bumps),
 - internal EMC pads, or
 - a combination of both points of contact on the receptacle outer shell and internal EMC pads.

If points of contacts are used on the receptacle, then the receptacle points of contact **shall** make connection with the mated plug within the contact zones defined in Figure 3-2. A minimum of four separate points of contact are required. Additional points of contact are allowed. See Section 3.2.2.4 for a reference design of receptacle outer shell. The reference design includes four spring fingers as points of contact. Alternate configurations **may** include spring fingers on the A contact side or B contact side and formed solid bumps (e.g., dimples) on the B contact side or A contact side, respectively. Spring fingers are required on a minimum of one side to provide a pressure fit on opposing sides of the plug shell. Additional bumps **may** be used, but if bumps are on opposing sides of the receptacle shell, the minimum distance between the bumps **shall** be greater than the maximum plug shell defined dimension.

If internal EMC pads are present in the receptacle, then they **shall** comply with the requirements defined in Figure 3-1. The shielding pads **shall** be connected to the receptacle shell. If no receptacle shell is present, then the receptacle **shall** provide a means to connect the shielding pad to ground. See Section 3.2.2.3 for a reference design of the shielding pad and ground connection.

7. This specification defines the USB Type-C receptacle shell length of 6.20 mm as a reference dimension. The USB Type-C receptacle is designed to have shell length of 6.20 ± 0.20 mm to provide proper mechanical and electrical mating of the plug to the receptacle (e.g., full seating of the plug in the receptacle and protection of the receptacle tongue during insertion/withdrawal). The USB Type-C receptacle at the system level **should** be implemented such that the USB Type-C receptacle connector mounted in the associated system hardware has an effective shell length equal to 6.20 ± 0.20 mm.

8. The USB Type-C connector mating interface is defined so that the electrical connection **may** be established without the receptacle shell. To prevent excessive misalignment of the plug when it enters or exits the receptacle, the enclosure **should** have features to guide the plug for insertion and withdrawal when a modified receptacle shell is present. If the USB Type-C receptacle shell is modified from the specified dimension, then the recommended lead in from the receptacle tongue to the plug point of entry is 1.5 mm minimum when mounted in the system.

This specification allows receptacle configurations with a conductive shell, a non-conductive shell, or no shell. The following requirements apply to the receptacle contact dimensions shown in SECTION A-A and ALTERNATE SECTION A-A shown in Figure 3-1:

- If the receptacle shell is conductive, then the receptacle contact dimensions of SECTION A-A or ALTERNATE SECTION A-A shown in Figure 3-1 **shall** be used.
- If the receptacle shell is non-conductive, then the receptacle contact dimensions of ALTERNATE SECTION A-A shown in Figure 3-1 **shall** be used. The contact dimensions of SECTION A-A are not allowed.
- If there is no receptacle shell, then the receptacle contact dimensions of either SECTION A-A or ALTERNATE SECTION A-A shown in Figure 3-1 **shall** be used. If there is no receptacle shell and the receptacle is used in an implementation that does not effectively provide a conductive shell, then a receptacle with the contact dimensions of ALTERNATE SECTION A-A shown in Figure 3-1 **should** be used.

Note: If the product that incorporates a USB Type-C receptacle supports **USB PD** Extended Power Range (EPR) operation, consideration **should** be given to the choice between the contact dimensions shown in SECTION A-A versus ALTERNATE SECTION A-A. **USB PD** EPR-compatible Sources and Sinks have to be designed to withstand potential electrical arcing during unplug events when power is being supplied across the connector and having a larger difference in length between the CC and VBUS pins **may** be beneficial when implementing detection circuitry intended to help mitigate the damage due to potential arcing. See Section H.3.2 for more information.

9. A paddle card (e.g., PCB) **may** be used in the USB Type-C plug to manage wire termination and electrical performance. Section 3.2.2.5 includes the guidelines and a design example for a paddle card.
10. This specification does not define standard footprints. Figure 3-5 shows an example SMT (surface mount) footprint for the vertical receptacle shown in Figure 3-1. Additional reference footprints and mounting configurations are shown in Figure 3-6, Figure 3-7, Figure 3-8, Figure 3-9, Figure 3-10 and Figure 3-11.
11. The receptacle shell **shall** be connected to the PCB ground plane.
12. All VBUS pins **shall** be connected together in the USB Type-C plug.
13. All Ground return pins **shall** be connected together in the USB Type-C plug.
14. All VBUS pins **shall** be connected together at the USB Type-C receptacle when it is in its mounted condition (e.g., all VBUS pins bussed together in the PCB).
15. All Ground return pins **shall** be connected together at the USB Type-C receptacle when it is in its mounted condition (e.g., all Ground return pins bussed together in the PCB).
16. The USB Type-C Power-Only plug is a depopulated version of the USB Full-Featured Type-C plug or the **USB 2.0** Type-C plug. The interface dimensions **shall** conform to Figure 3-3 or Figure 3-12. Contacts for CC, VBUS, and GND (i.e., A1, A4, A5, A9, A12, B1, B4, B9, and B12) **shall** be present. Physical presence of contacts in the other 15 contact locations is **optional**. The USB Type-C Power-Only plug **shall** only be used on a non-charger captive cable application. Implementation of **Rd** or CC communication on pin A5 is required in the application.
17. The USB Type-C Power-Only receptacle is a depopulated version of the USB Full-Featured Type-C receptacle. The interface dimensions shall conform to Figure 3-1. Contacts for CC, D+/D-, VBUS, and

GND (i.e., A1, A4, A5, A6, A7, A9, A12, B1, B4, B5, B6, B7, B9, and B12) shall be present. The physical presence of SBU contacts (i.e., A8 and B8) are optional. The other 8 contact locations shall not be present. The D+ and D- contacts (i.e., A6, A7, B6, and B7) shall be shorted together inside the receptacle and not terminated to the board. The USB Type-C Power-Only receptacle EMC provisions highlighted in previous notes are optional normative (i.e., grounding points on the shell or EMC pads). The design of the USB Type-C Power-Only receptacle shall be compliant with maximum power specified by USB PD and USB 2.0. The USB Type-C Power-Only receptacle may be used on power source or sink applications but when used on a sink, it will not support implementing USB BC 1.2.

18. USB Type-C plugs with a right-angle overmold shall comply with the dimensions in Figure 3-4.

Figure 3-1 USB Type-C Receptacle Interface Dimensions

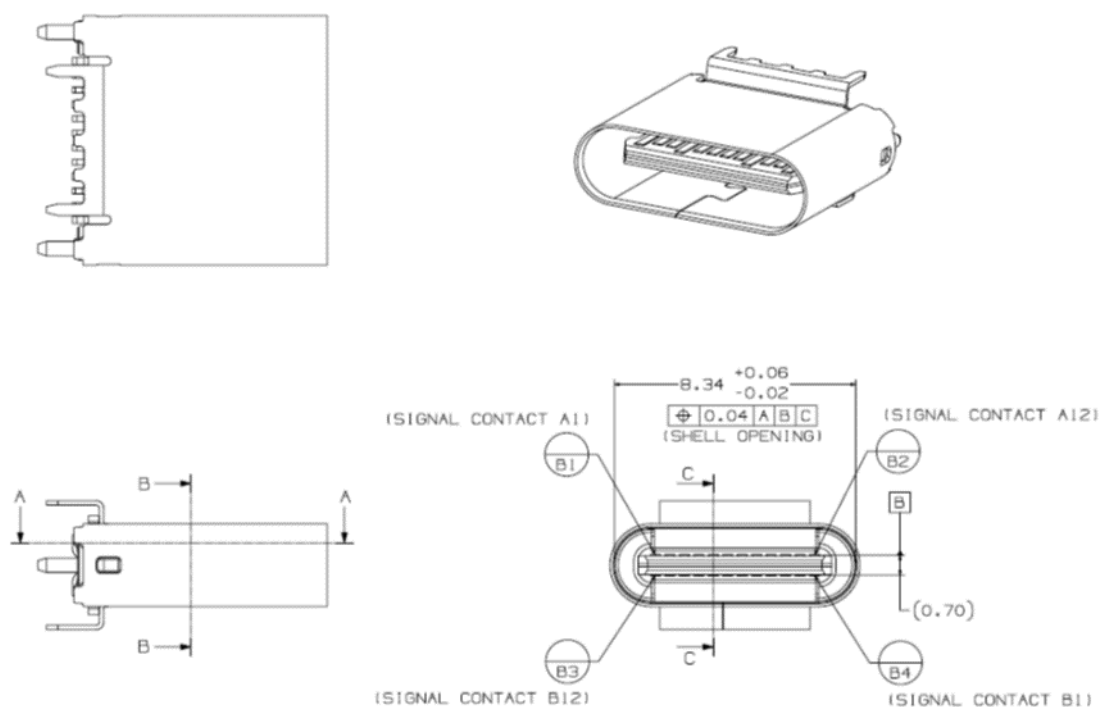
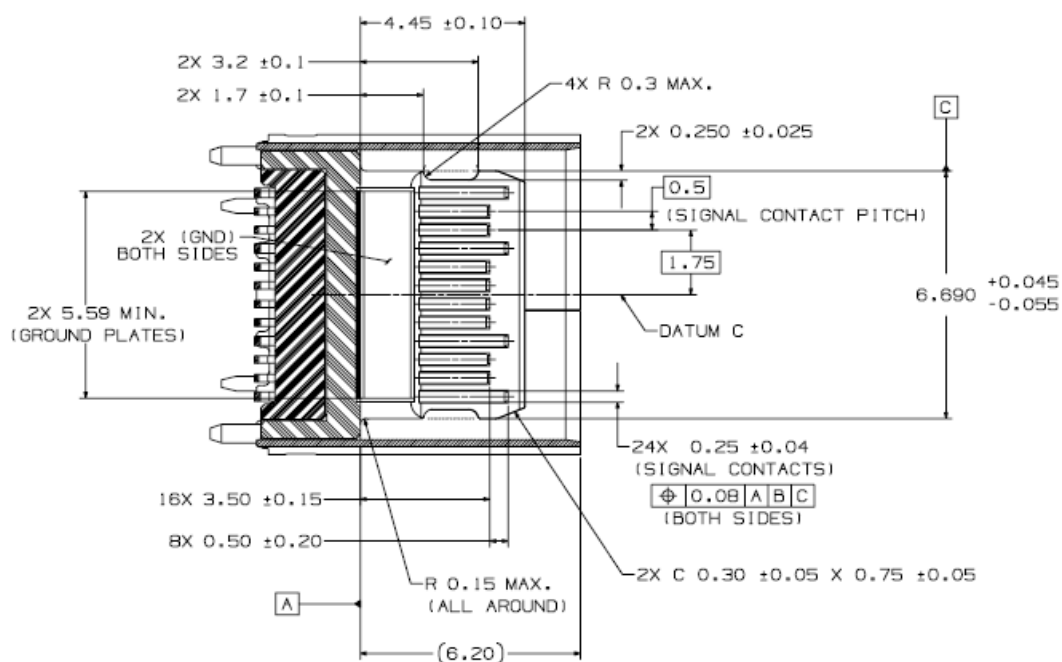
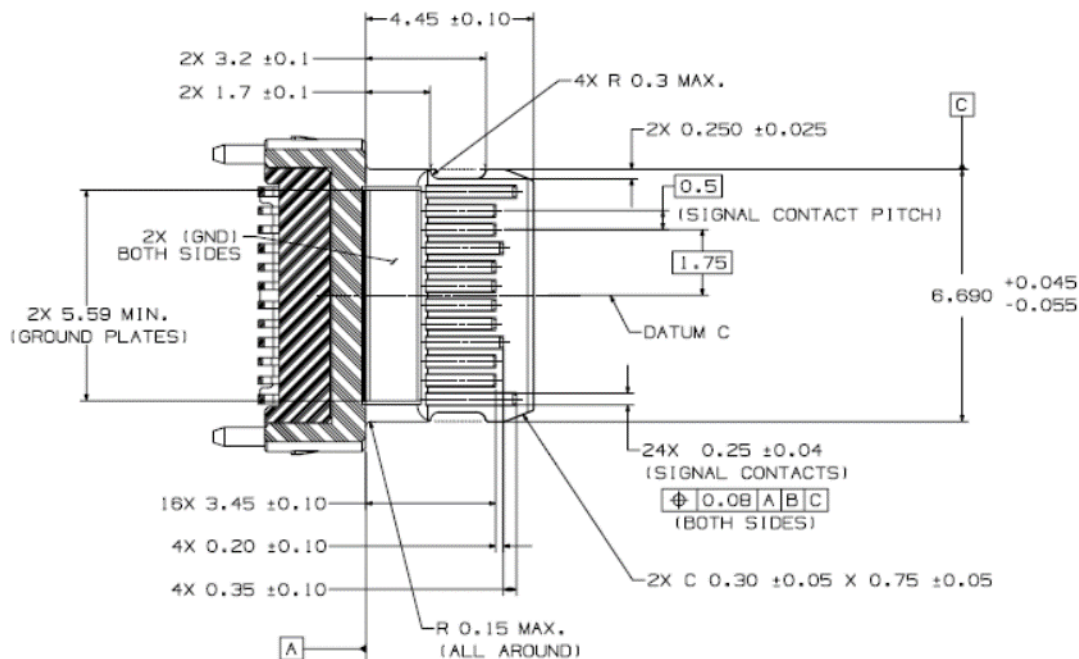


Figure 3-1 USB Type-C Receptacle Interface Dimensions, cont.



SECTION A-A



ALTERNATE SECTION A-A

Figure 3-1 USB Type-C Receptacle Interface Dimensions, cont.

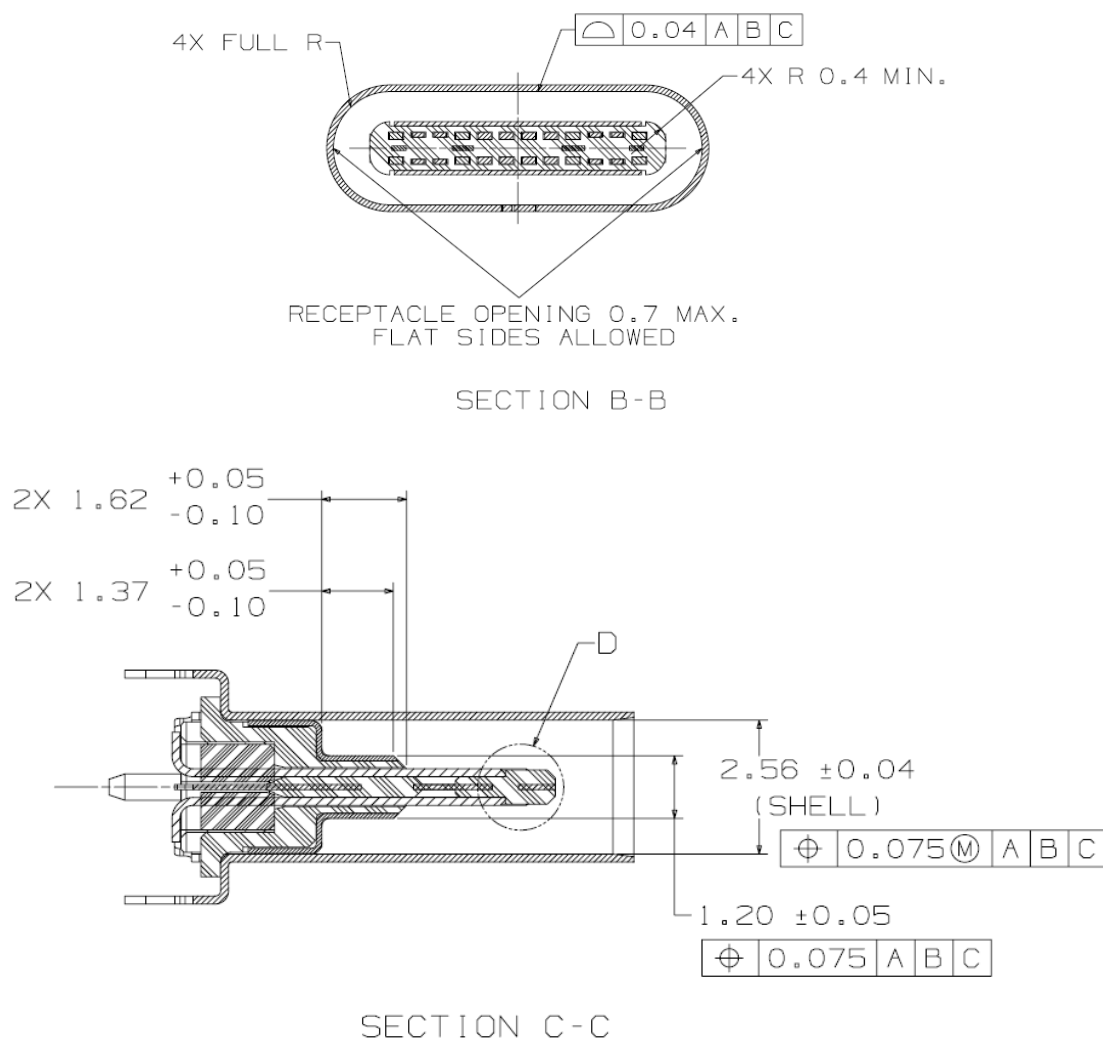


Figure 3-1 USB Type-C Receptacle Interface Dimensions, cont.

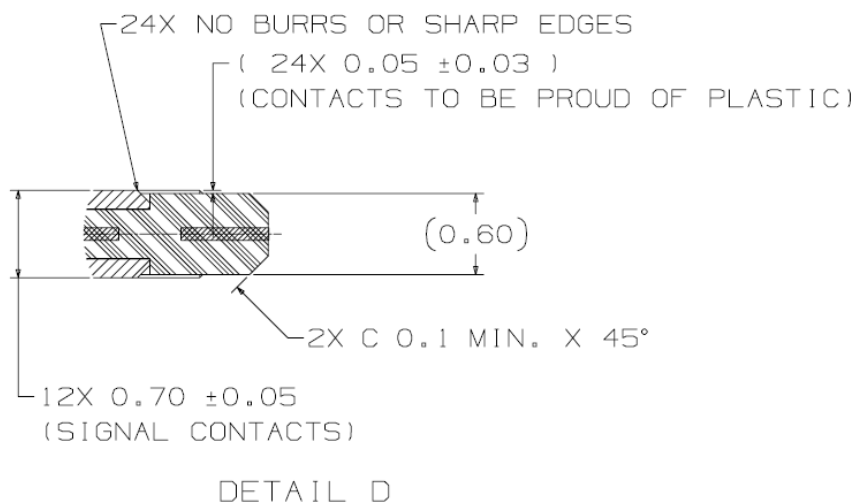


Figure 3-2 Reference Design USB Type-C Plug External EMC Spring Contact Zones

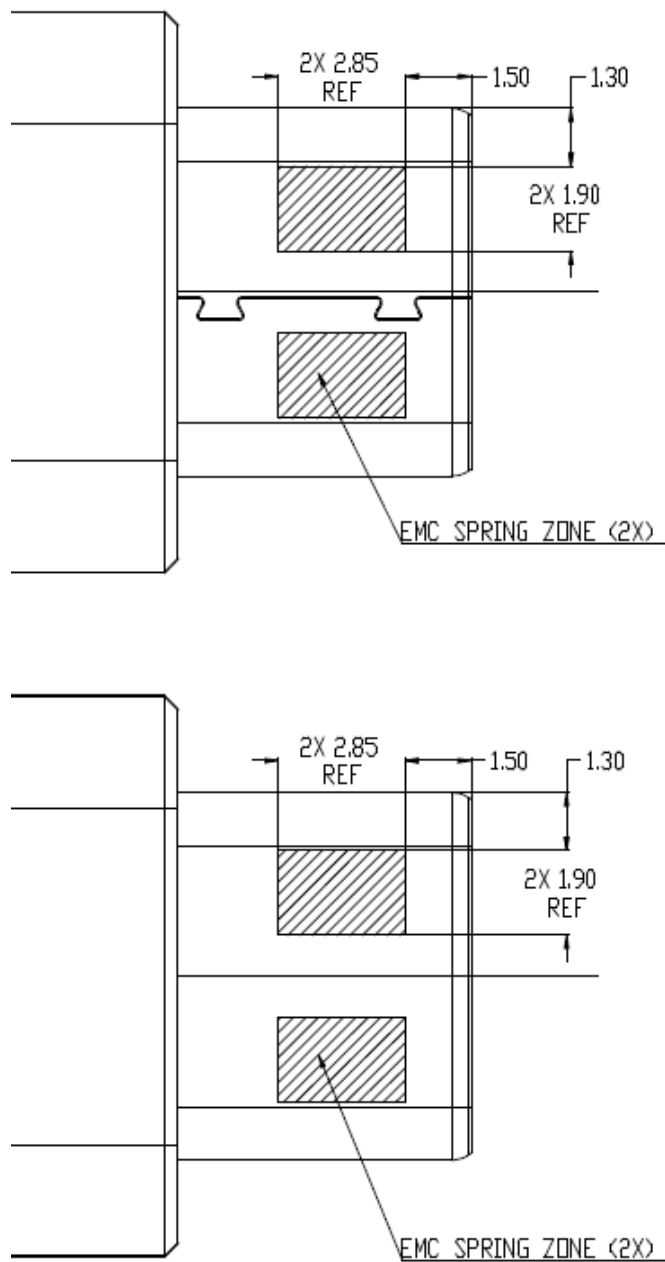


Figure 3-3 USB Full-Featured Type-C Plug Interface Dimensions

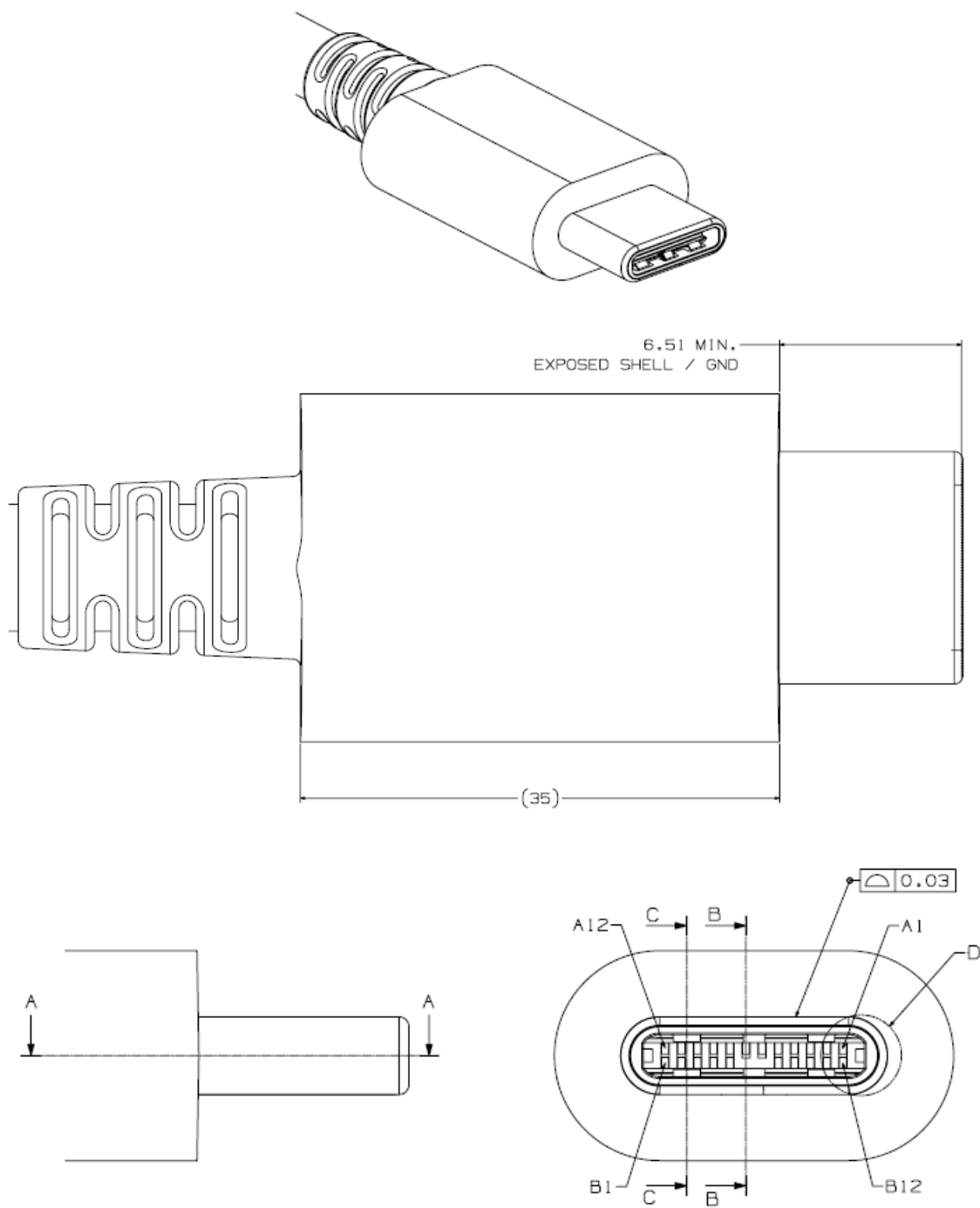
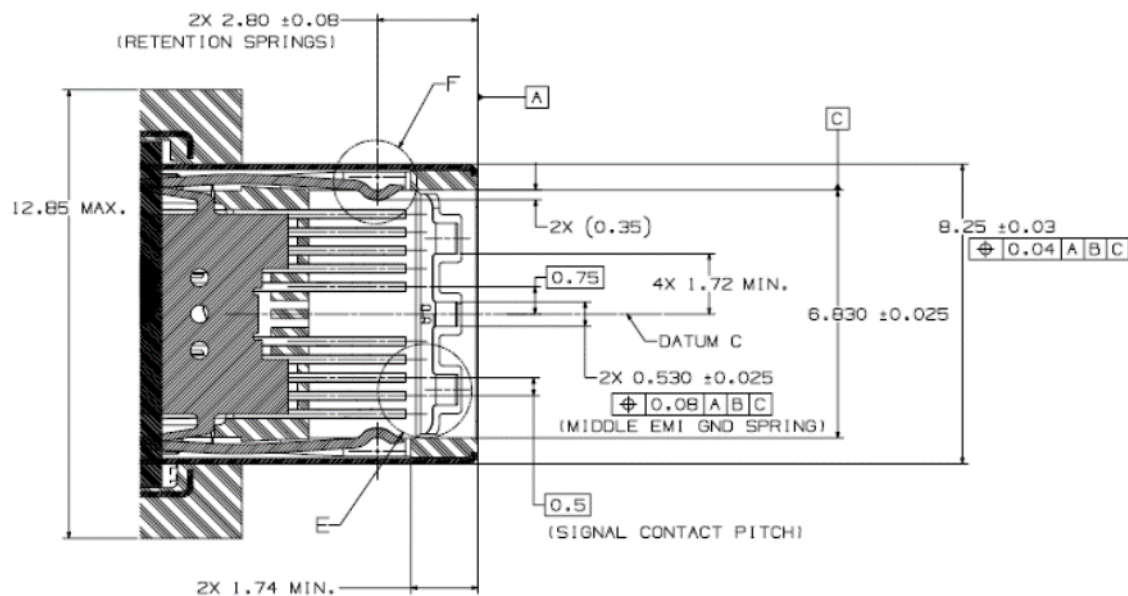
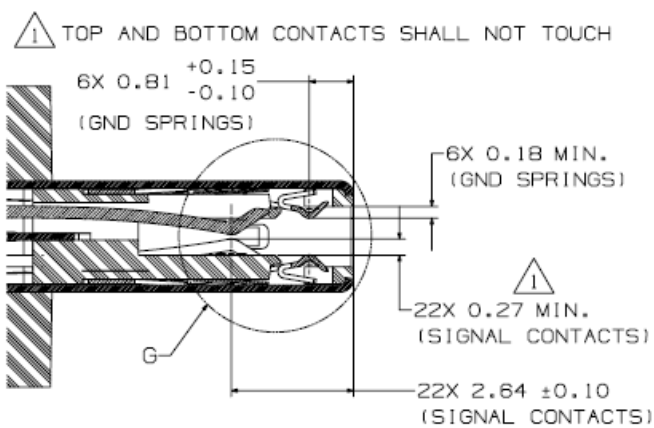


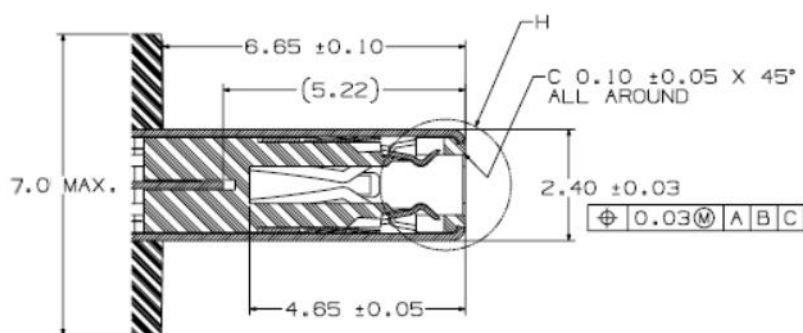
Figure 3-3 USB Full-Featured Type-C Plug Interface Dimensions, cont.



SECTION A-A



SECTION B-B



SECTION C-C

Figure 3-3 USB Full-Featured Type-C Plug Interface Dimensions, cont.

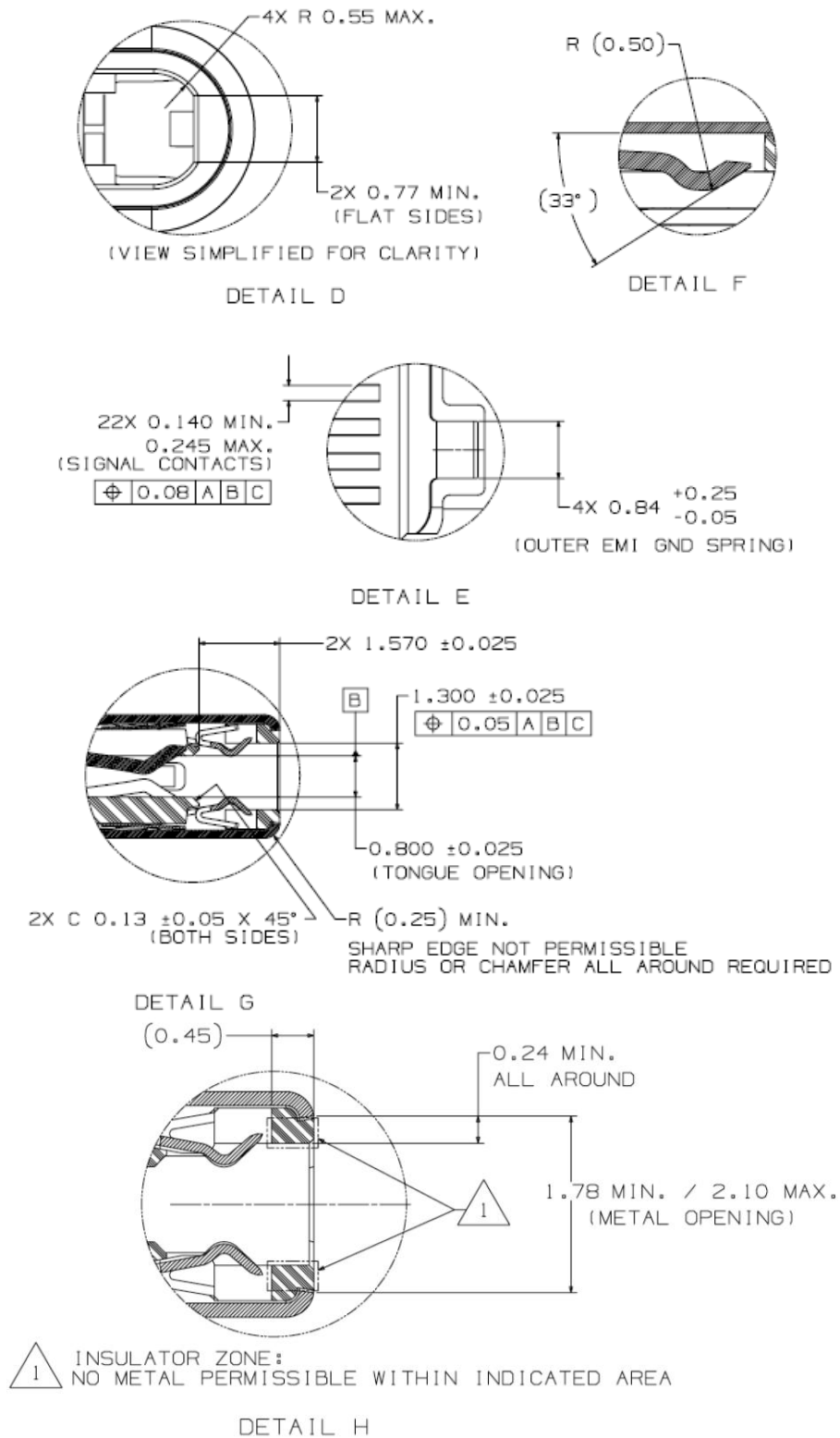
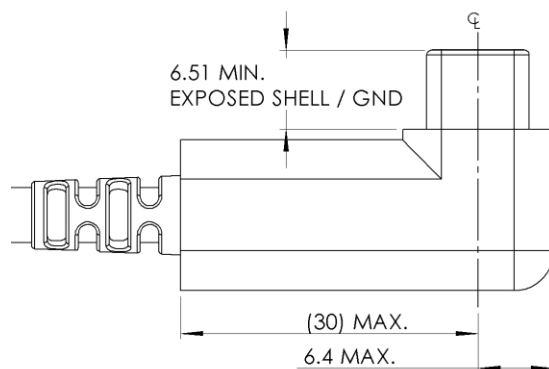
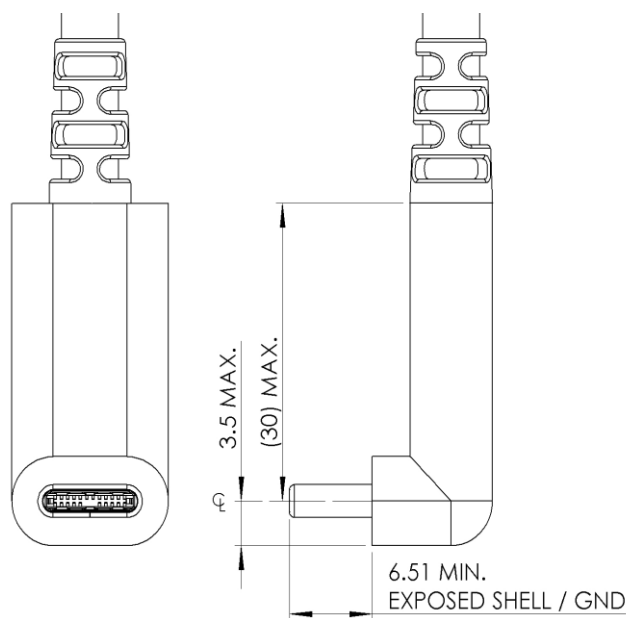


Figure 3-4 Right-Angle USB Type-C Plug Dimensional Requirements



HORIZONTAL RIGHT ANGLE PLUG



VERTICAL RIGHT ANGLE PLUG

Figure 3-5 Reference Footprint for a USB Type-C Vertical Mount Receptacle (*Informative*)

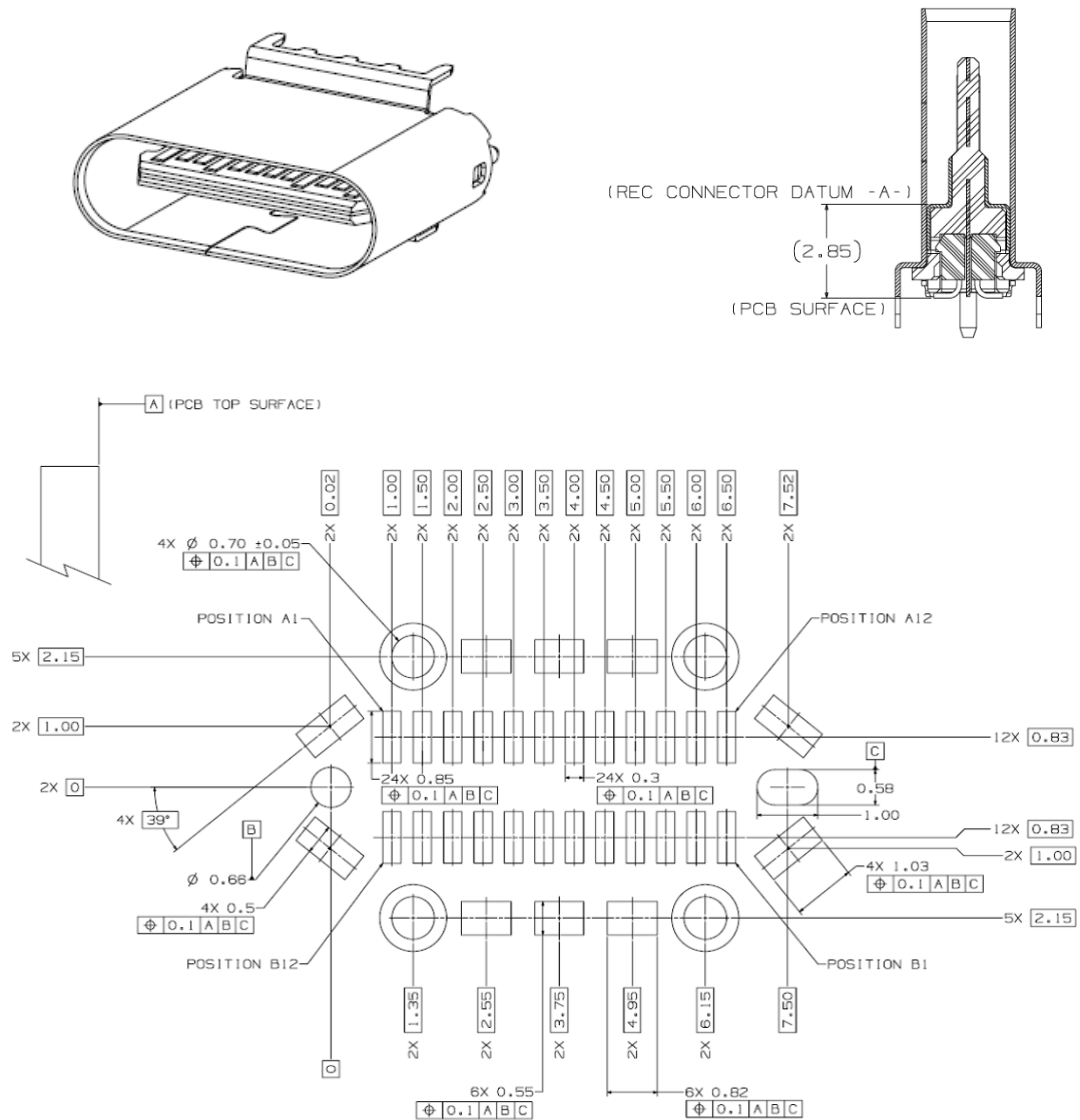
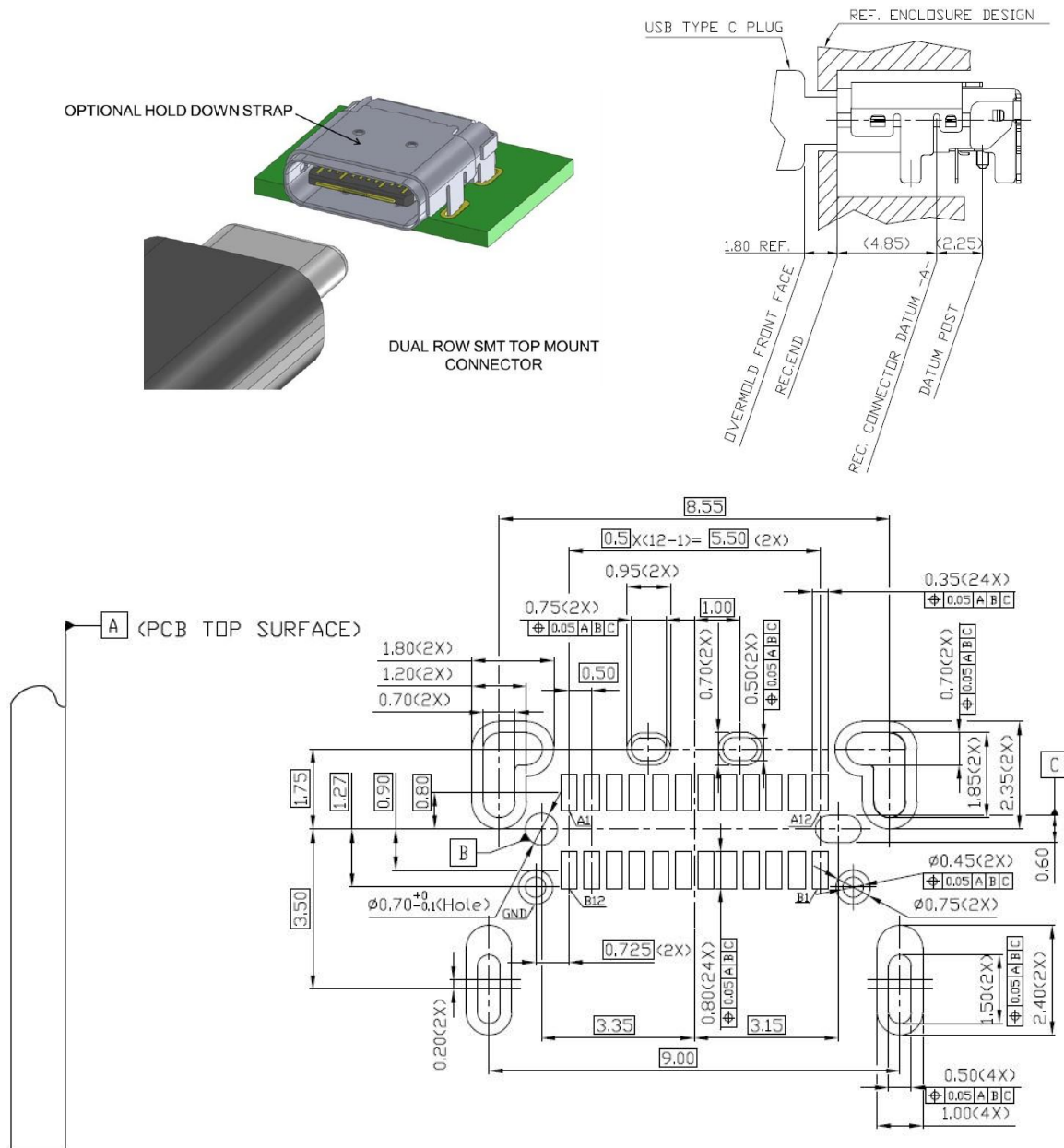


Figure 3-6 Reference Footprint for a USB Type-C Dual-Row SMT Right-Angle Receptacle (Informative)



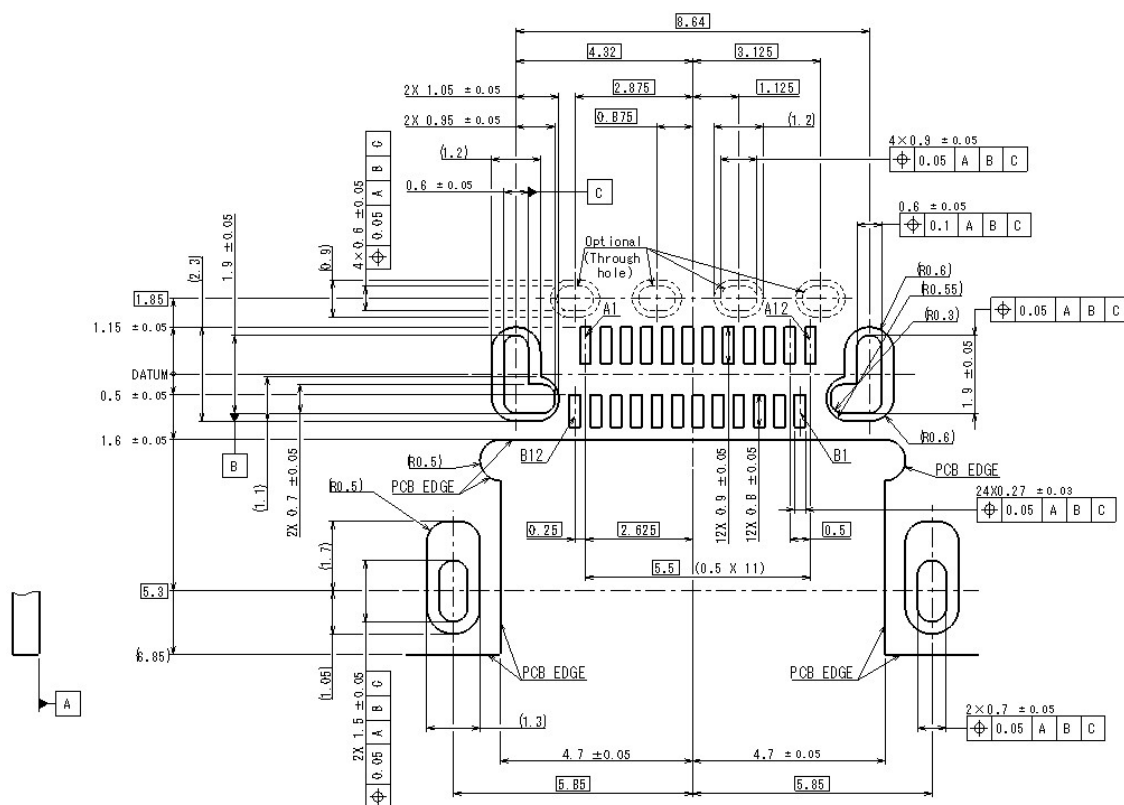
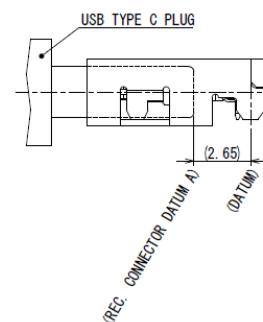
Receptacle Connector Datum A

Datum Post

(3.00)

0.30±0.03
12 PLACES
0.05 A B C
0.50 TYP.
0.85
0.23
0.70±0.03
12 PLACES
1.55
0.66
0.00
0.65
0.65±0.025
B
0.53
1.35
0.40±0.05
0.05 A B C
12 PLACES
0.80
1.20
2.00
2.80
3.20
4.00
5.20
6.00
7.20
7.73
1.25
0.65±0.025
C
0.95±0.05
0.30±0.03
4 PLACES
0.05 A B C
0.30±0.03
4 PLACES
0.05 A B C
0.95±0.05
1.25
0.65±0.025
C
0.89
0.50±0.05
0.05 A B C
4 PLACES
1.10±0.05
0.10 A B C
+ PLACES
7.20
0.10 MIN. ANNULAR RING TYP.
8.09

A 3D CAD model of a custom PCB connector assembly. The assembly consists of a grey plastic housing with a black cylindrical cap on the right side. Inside the housing, a black PCB is visible, featuring a row of gold-plated pins. The assembly is mounted on a green PCB, with yellow circular pads indicating the mounting points. The model is shown from a perspective view, highlighting the internal components and the mounting structure.



[illegible]

Figure 3-10 Reference Footprint for a USB 2.0 Type-C Through Hole Right Angle Receptacle (Informative)

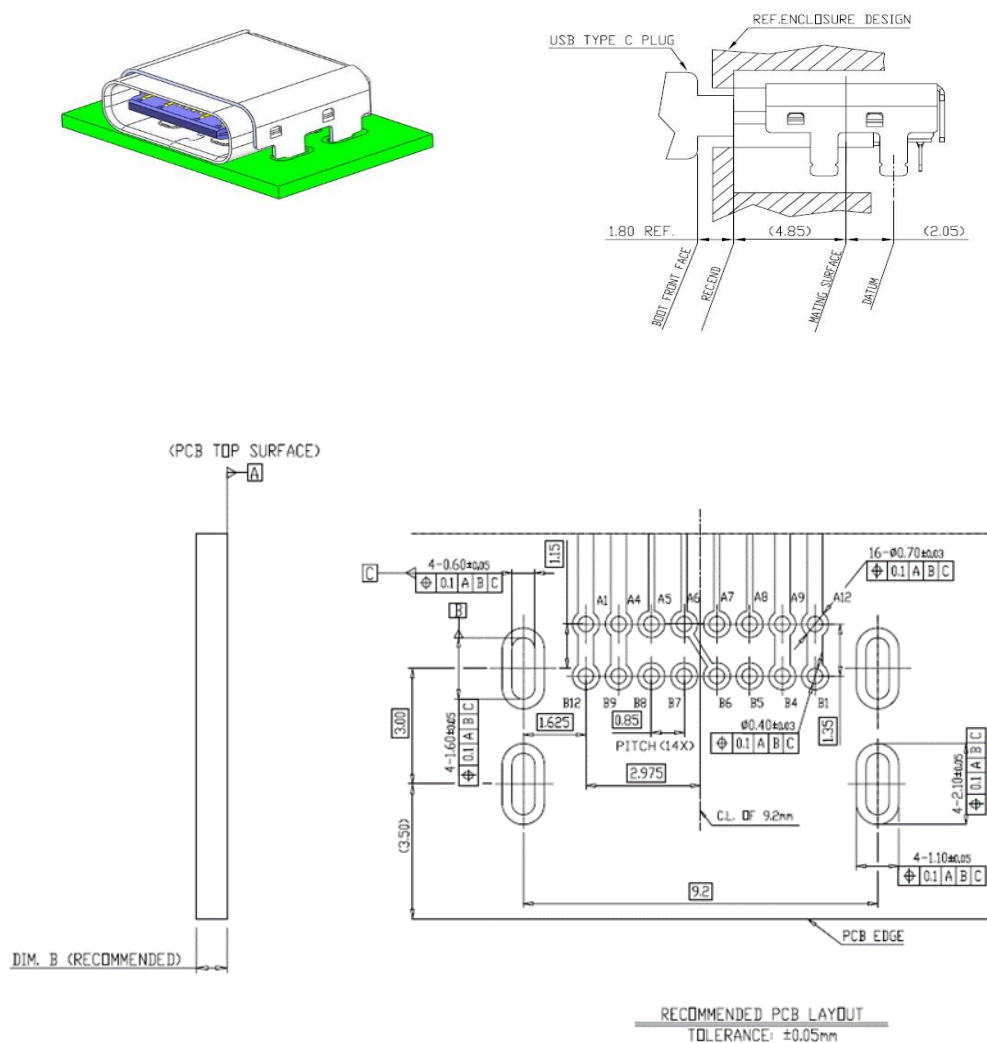
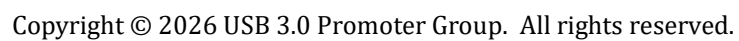


Figure 1 shows a 3D model of a plug and receptacle assembly on the left. The plug is a reddish-brown component with two blue pins, mounted on a green base. On the right is a cross-sectional view of the assembly. The plug is labeled 'PLUG' and the receptacle is labeled 'RECEPTACLE'. The cross-section shows the internal components of the receptacle, including two pins and a central contact. Dimensions are provided in inches: (6.65) for the total width, (6.20) for the width of the receptacle body, (0.45) for the plug overmold, (1.20) for the panel front face, (0.78) for the receptacle face, and (1.850) for the receptacle connector datum A. The height of the receptacle is (3.25).



This specification requires that all contacts be present in the mating interface of the USB Full-Featured Type-C receptacle connector, all contacts *except* the **USB 3.2** or **USB4** signals (i.e., A2, A3, A10, A11, B2, B3, B10 and B11) be present in the mating interface of the **USB 2.0** Type-C receptacle connector, and all contacts *except* the **USB 3.2** or **USB4** signals (i.e., A2, A3, A10, A11, B2, B3, B10 and B11) and SBU contacts (i.e., A8, and B8) be present in the mating interface of the USB Type-C Power-Only receptacle connector, but allows the plug to include only the contacts required for **USB PD** and **USB 2.0** functionality for applications that only support **USB 2.0**.

The **USB 2.0** Type-C plug is shown in Figure 3-12. The following design simplifications *may* be made when only **USB 2.0** is supported:

- Only the contacts necessary to support **USB PD** and **USB 2.0** are required in the plug. All other pin locations *may* be unpopulated. See Table 3-5.
- Unlike the USB Full-Featured Type-C plug, the internal EMC springs *may* be formed from the same strip as the signal, power, and ground contacts. The internal EMC springs contact the inner surface of the plug shell and mate with the receptacle EMC pads when the plug is seated in the receptacle. Alternately, the **USB 2.0** Type-C plug *may* use the same EMC spring configuration as defined for the USB Full-Featured Type-C plug. The **USB 2.0** Type-C plug four EMC spring locations are defined in Figure 3-12. The alternate configuration using the six spring locations is defined in Figure 3-1. Also refer to the reference designs in 3.2.2.3 for further clarification.
- A paddle card inside the plug *may not* be necessary if wires are directly attached to the contact pins.

Figure 3-12 USB 2.0 Type-C Plug Interface Dimensions

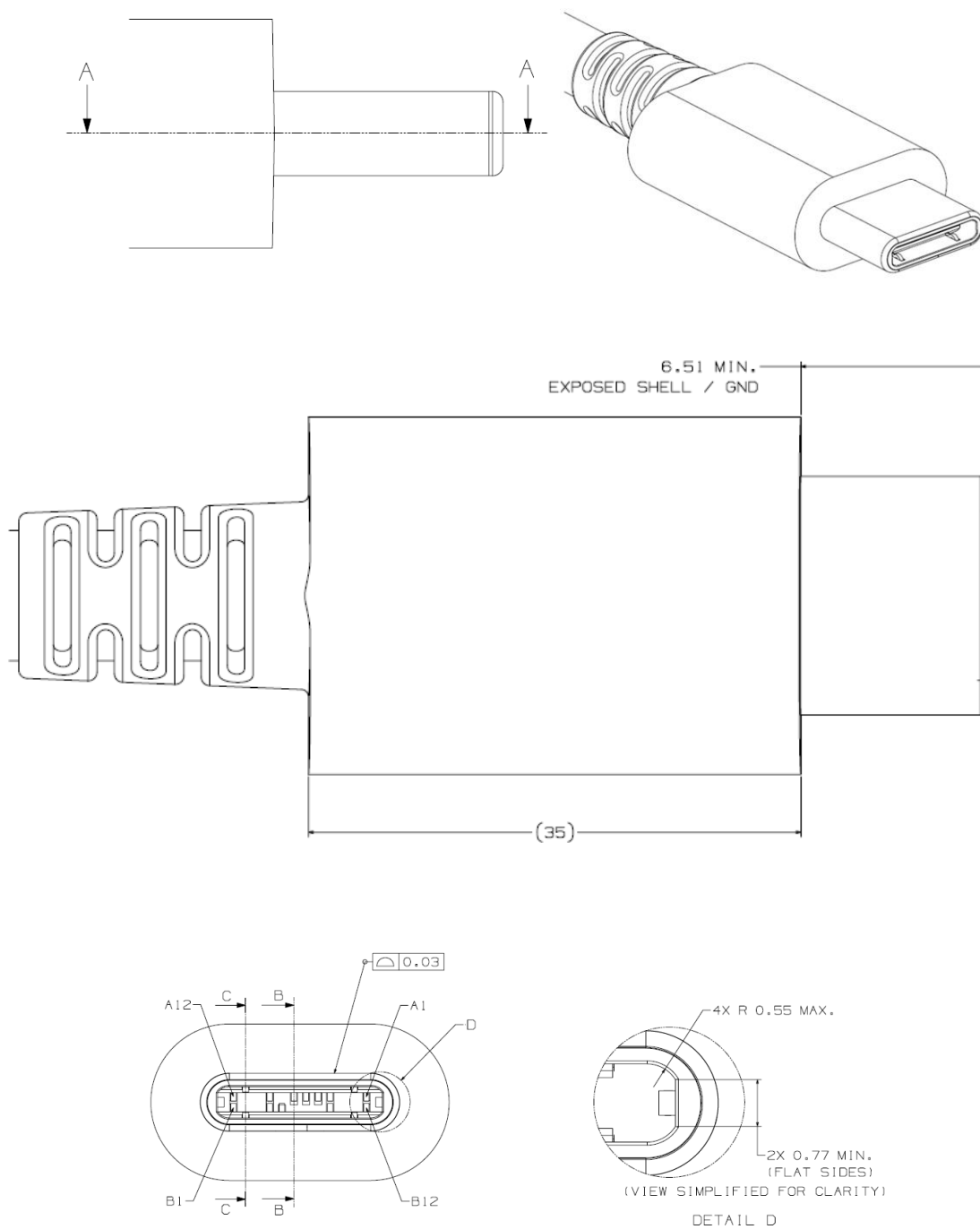


Figure 3-12 USB 2.0 Type-C Plug Interface Dimensions, cont.

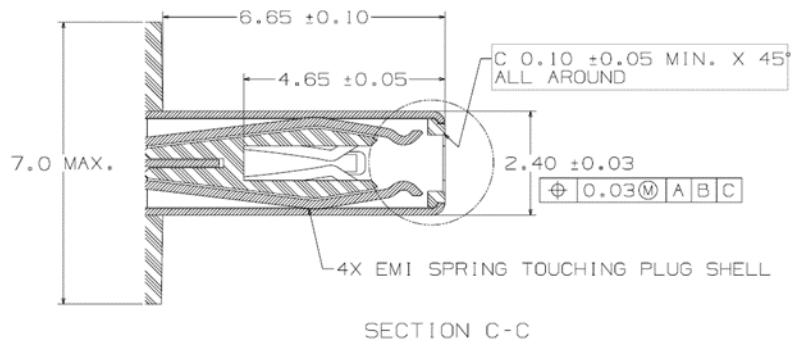
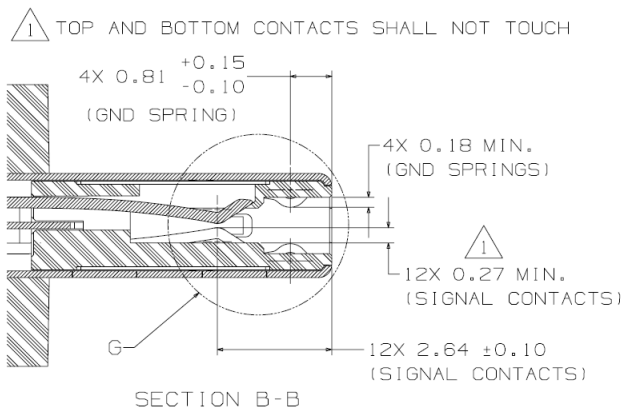
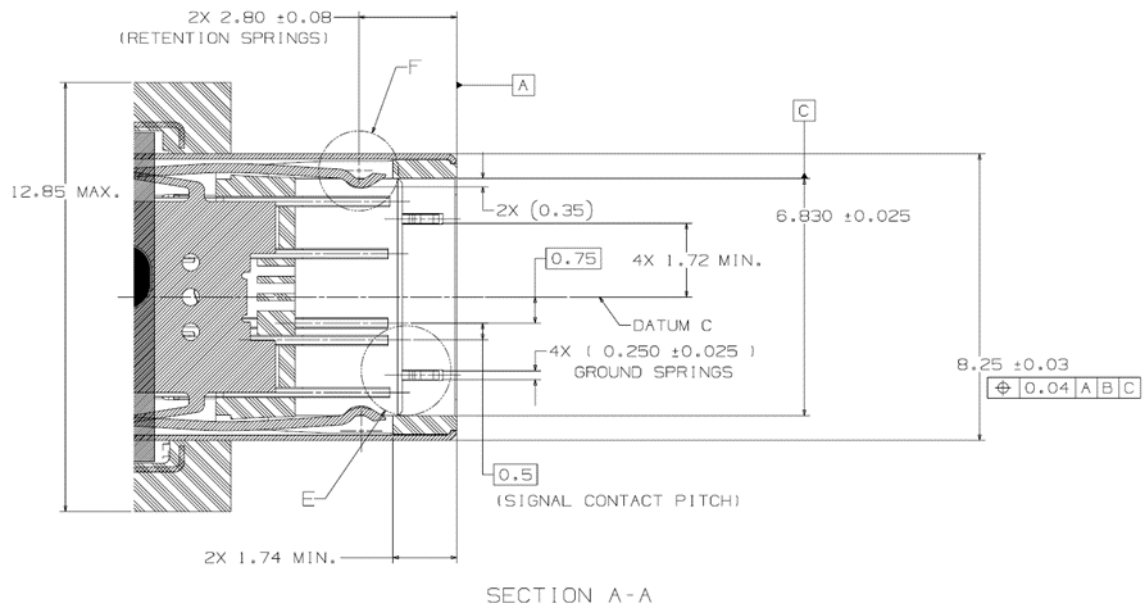


Figure 3-12 USB 2.0 Type-C Plug Interface Dimensions, cont.

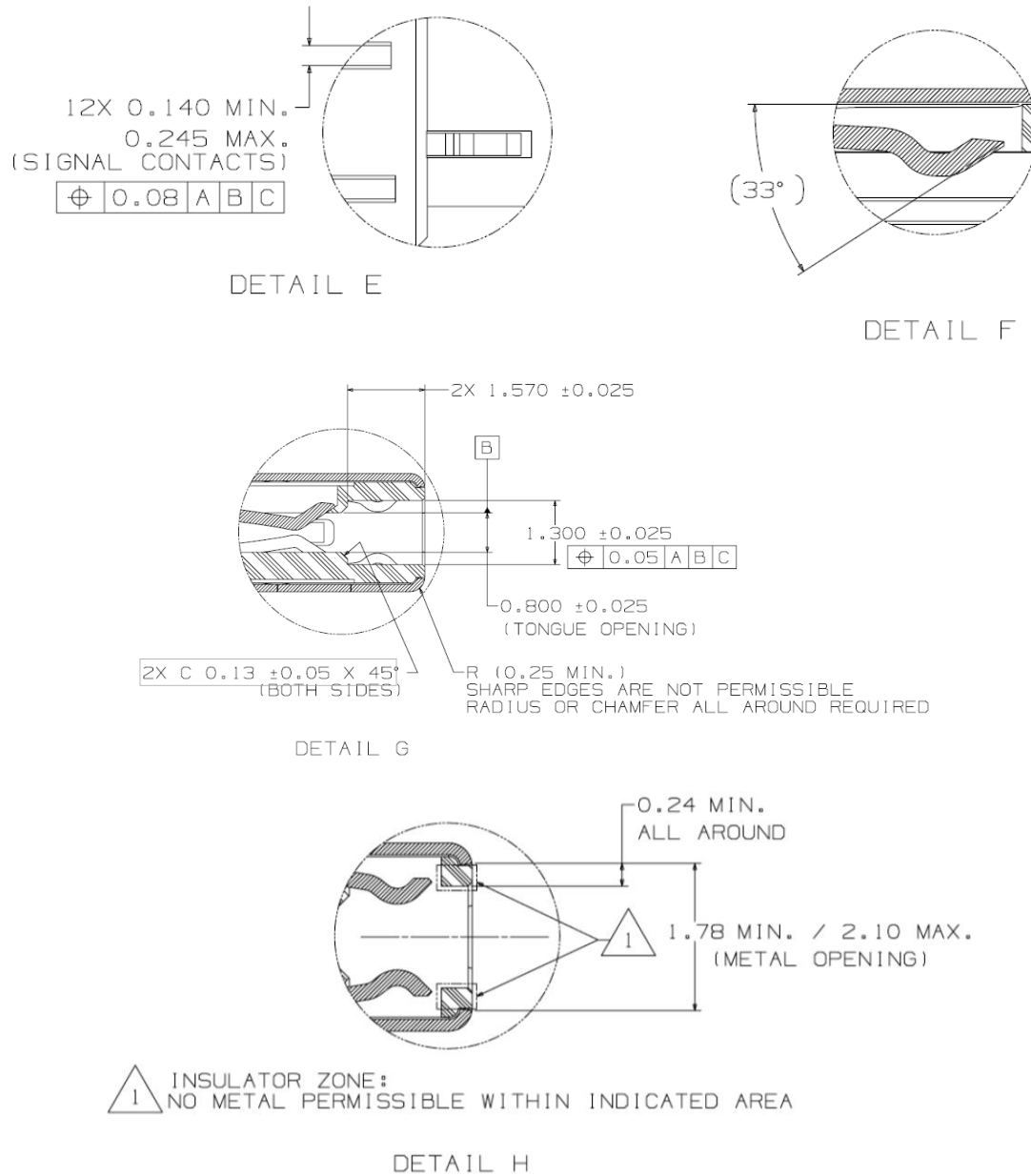
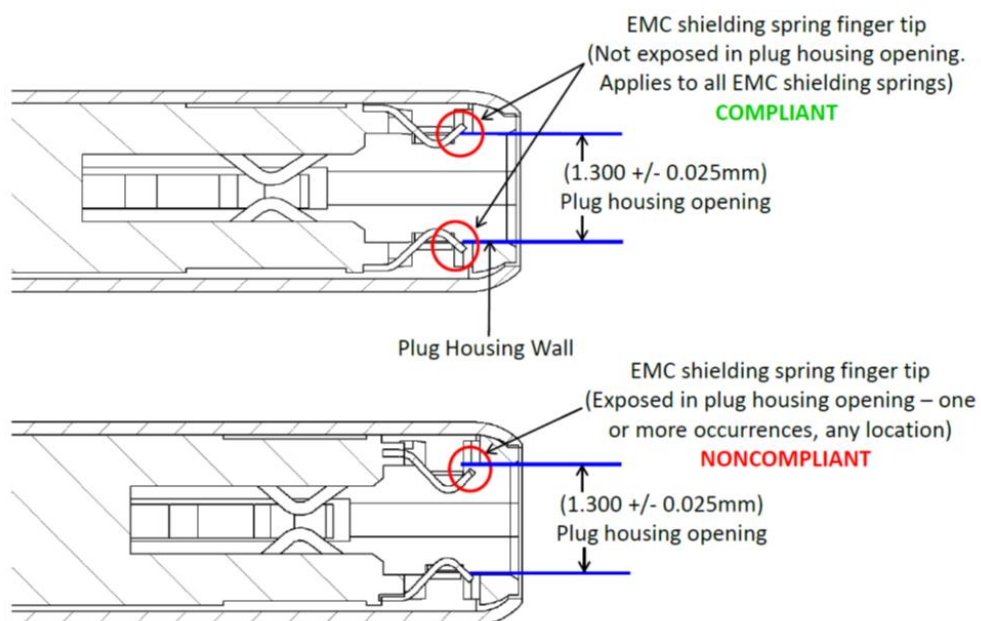


Figure 3-13 USB Type-C Plug EMC Shielding Spring Tip Requirements



3.2.2 Reference Designs

This section provides reference designs for a few key features of the USB Type-C connector. The reference designs are provided as acceptable design examples. They are not **normative**.

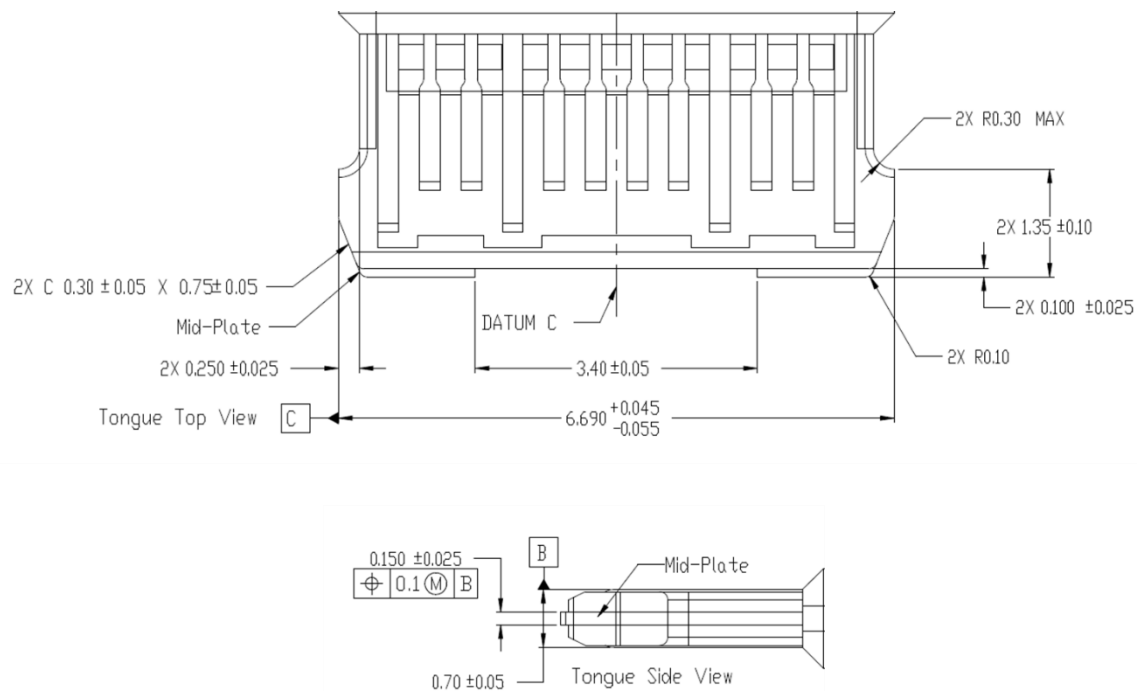
3.2.2.1 Receptacle Mid-Plate (Informative)

The signals between the top and bottom of the receptacle tongue are isolated by a mid-plate inside the tongue.

Figure 3-14 shows a reference design of the mid-plate. It is important to pay attention to the following features of the middle plate:

- The distance between the signal contacts and the mid-plate **should** be accurately controlled since the variation of this distance **may** significantly impact impedance of the connector.
- The mid-plate in this particular design protrudes slightly beyond the front surface of the tongue. This is to protect the tongue front surface from damage caused by miss-insertion of small objects into the receptacle.
- The mid-plate is required to be directly connected to the PCB ground with at least two grounding points.
- The sides of the mid-plate mate with the plug side latches, making ground connections to reduce EMC. Proper surface finishes are necessary in the areas where the side latches and mid-plate connections occur.

Figure 3-14 Reference Design of Receptacle Mid-Plate



3.2.2.2 Side Latch (Informative)

The side latches (retention latches) are located in the plug. Figure 3-15 shows a reference design of a blanked side latch. The plug side latches **should** contact the receptacle mid-plate to provide an additional ground return path.

Figure 3-15 Reference Design of Retention Latch

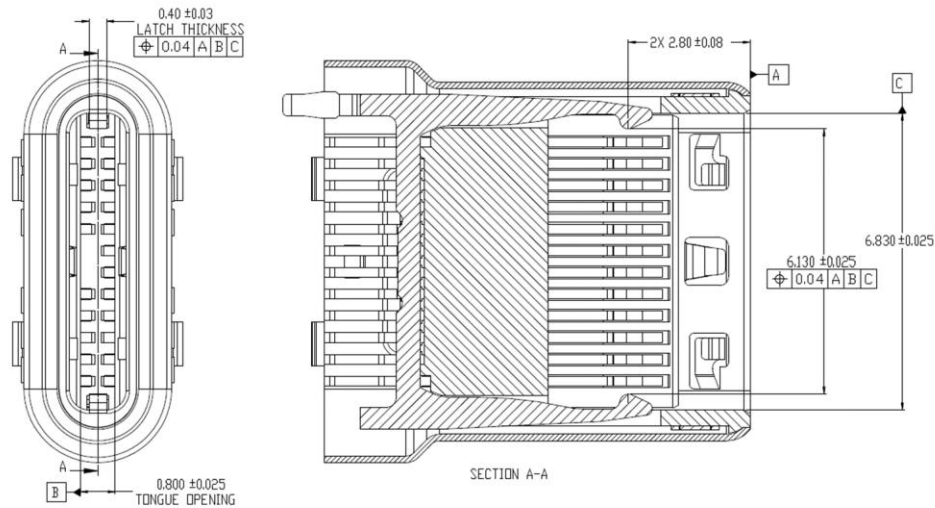
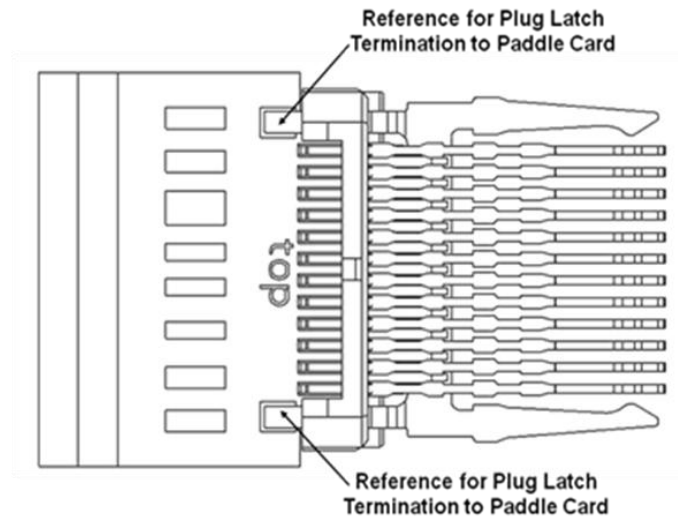


Figure 3-16 Illustration of the Latch Soldered to the Paddle Card Ground



3.2.2.3 Internal EMC Springs and Pads (Informative)

Figure 3-17 is a reference design of the internal EMC spring located inside the USB Full-Featured Type-C plug.
Figure 3-18 is a reference design of the internal EMC spring located inside the **USB 2.0** Type-C plug.

Figure 3-17 Reference Design of the USB Full-Featured Type-C Plug Internal EMC Spring

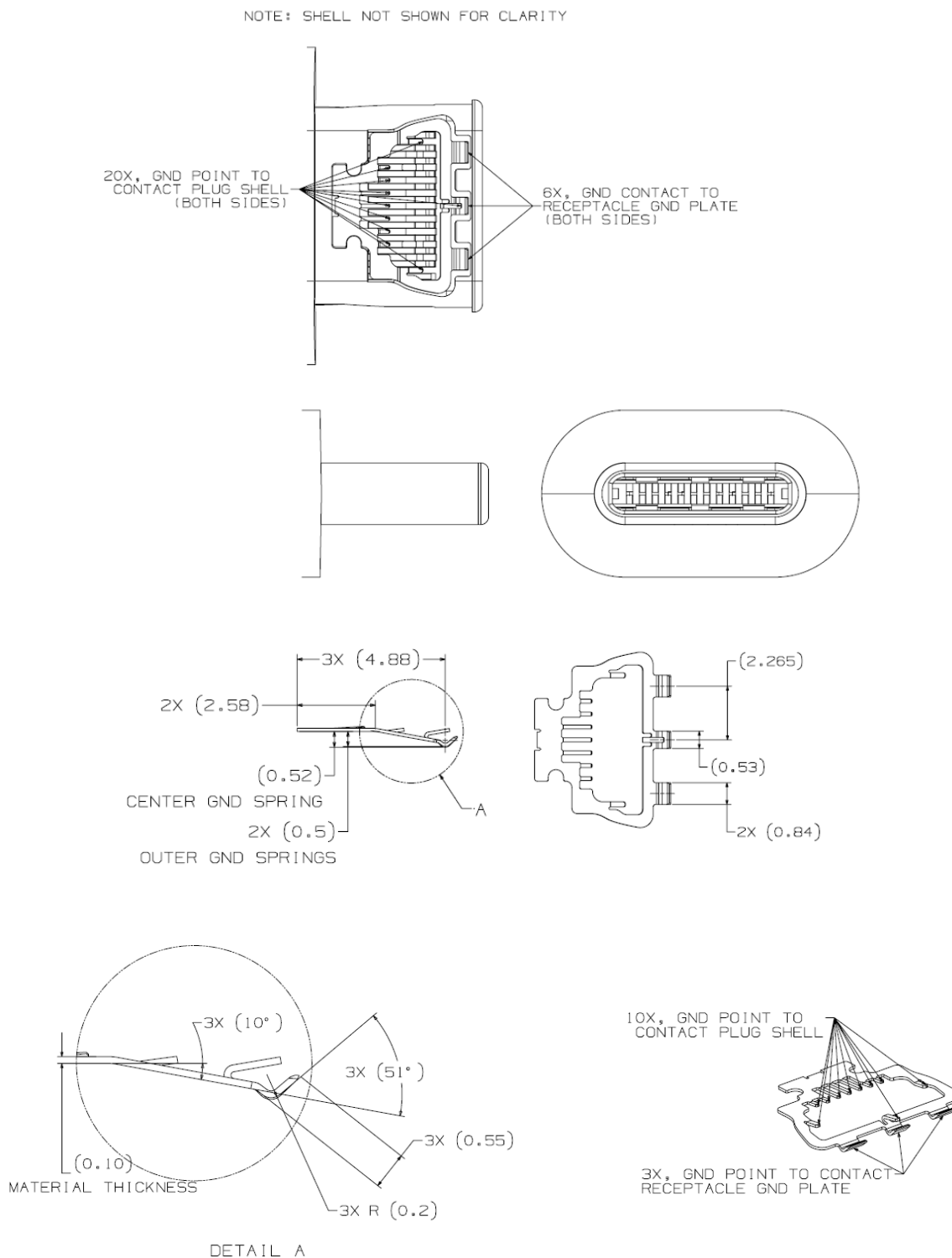
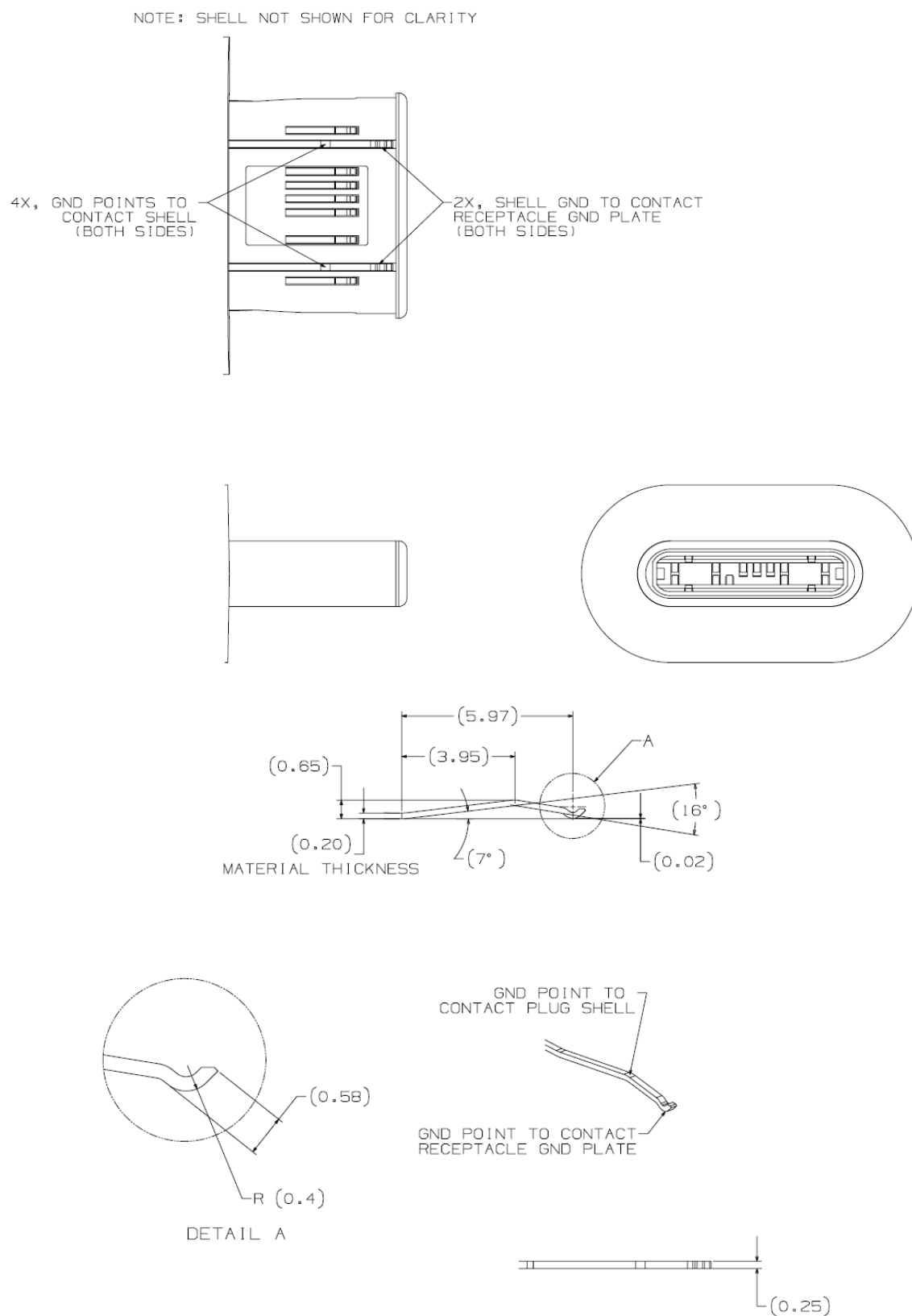


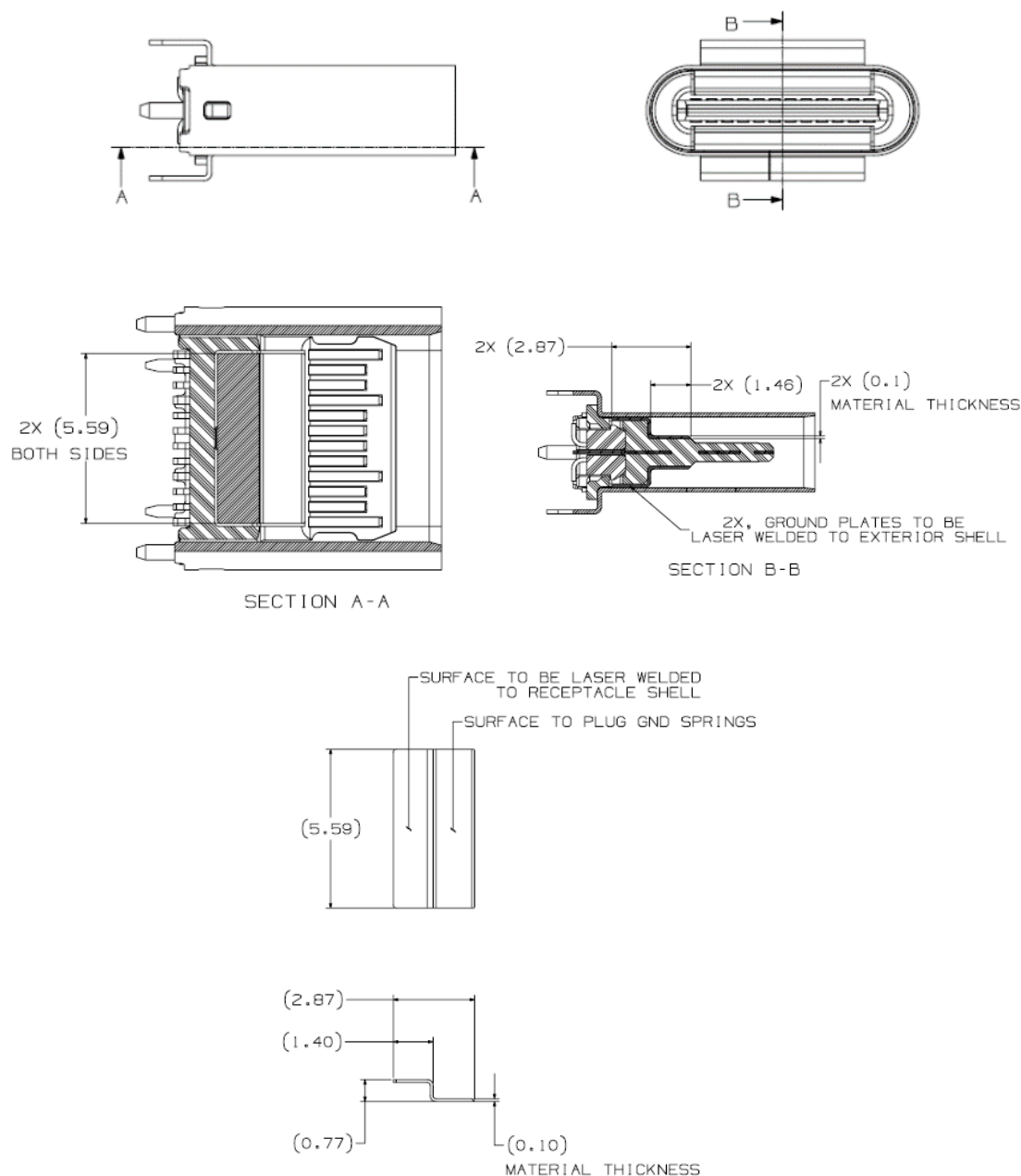
Figure 3-18 Reference Design of the *USB 2.0* Type-C Plug Internal EMC Spring



It is critical that the internal EMC spring contacts the plug shell as close to the EMC spring mating interface as possible to minimize the length of the return path.

The internal EMC pad (i.e., ground plate) shown in Figure 3-19 is inside the receptacle. It mates with the EMC spring in the plug. To provide an effective ground return, the EMC pads **should** have multiple connections with the receptacle shell.

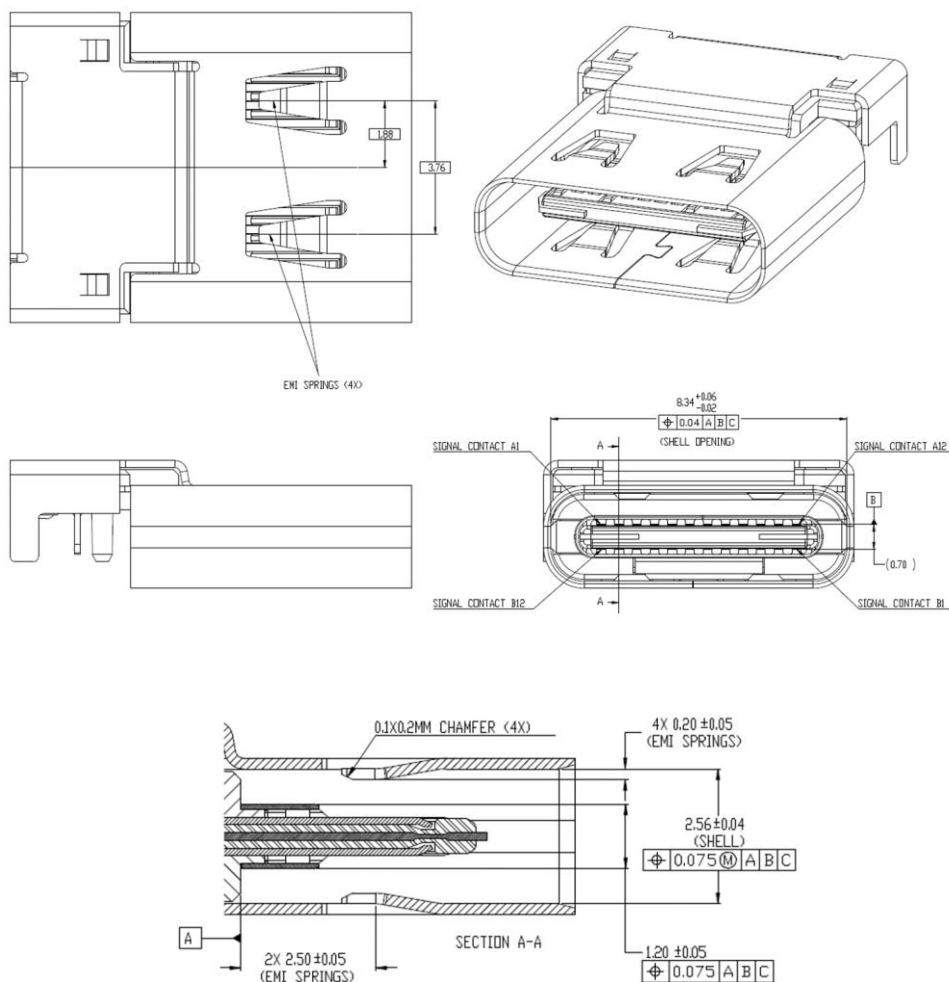
Figure 3-19 Reference Design of Internal EMC Pad



3.2.2.4 Optional External Receptacle EMC Springs (*Informative*)

Some applications *may* use receptacles with EMC springs that contact the outside of the plug shell. Figure 3-20 shows a reference receptacle design with external EMC springs. The EMC spring contact landing zones for the fully mated condition are *normative* and defined in Section 3.2.1.

Figure 3-20 Reference Design of a USB Type-C Receptacle with External EMC Springs



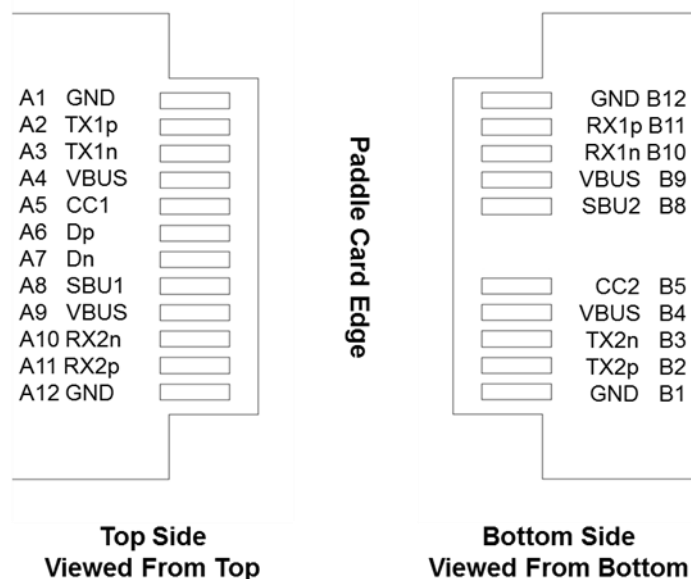
3.2.2.5 USB Full-Featured Type-C Plug Paddle Card (Informative)

The use of a paddle card is expected in the USB Full-Featured Type-C Plug.

Figure 3-21 illustrates the paddle card pin assignment and contact spring connection location for a USB Full-Featured Type-C plug. The following guidelines are provided for the paddle card design:

- The paddle card **should** use high performance substrate material. The recommended paddle card thickness **should** have a tolerance less than or equal to $\pm 10\%$.
- The USB TX/RX traces **should** be as short as possible and have a nominal differential characteristic impedance of $85\ \Omega$.
- The wire attach **should** have two high speed differential pairs on one side and two other high-speed differential pairs on the other side, separated as far as practically allowed.
- It is recommended that a grounded coplanar waveguide (CPWG) system be selected as a transmission line method.
- Use of vias **should** be minimized.
- VBUS pins **should** be bussed together on the paddle card.
- GND pins **should** be bussed together on the paddle card.

Figure 3-21 Reference Design of a USB Full-Featured Type-C Plug Paddle Card



3.2.3 Pin Assignments and Descriptions

The usage and assignments of the 24 pins for the USB Type-C receptacle interface are defined in Table 3-4.

Table 3-4 USB Type-C Receptacle Interface Pin Assignments

Pin	Signal Name	Description	Mating Sequence	Pin	Signal Name	Description	Mating Sequence
A1	GND	Ground return	First	B12	GND	Ground return	First
A2	TXp1	Positive half of first TX differential pair	Second	B11	RXp1	Positive half of first RX differential pair	Second
A3	TXn1	Negative half of first TX differential pair	Second	B10	RXn1	Negative half of first RX differential pair	Second
A4	VBUS	Bus Power	First	B9	VBUS	Bus Power	First
A5	CC1	Configuration Channel	Second	B8	SBU2	Sideband Use (SBU)	Second
A6	Dp1	Positive half of the USB 2.0 differential pair – Position 1	Second	B7	Dn2	Negative half of the USB 2.0 differential pair – Position 2	Second
A7	Dn1	Negative half of the USB 2.0 differential pair – Position 1	Second	B6	Dp2	Positive half of the USB 2.0 differential pair – Position 2	Second
A8	SBU1	Sideband Use (SBU)	Second	B5	CC2	Configuration Channel	Second
A9	VBUS	Bus Power	First	B4	VBUS	Bus Power	First
A10	RXn2	Negative half of second RX differential pair	Second	B3	TXn2	Negative half of second TX differential pair	Second
A11	RXp2	Positive half of second RX differential pair	Second	B2	TXp2	Positive half of second TX differential pair	Second
A12	GND	Ground return	First	B1	GND	Ground return	First
Notes: <ol style="list-style-type: none"> 1. Contacts B6 and B7 should not be present in the USB Type-C plug. The receptacle side shall support the USB 2.0 differential pair present on Dp1/Dn1 or Dp2/Dn2. The plug orientation determines which pair is active. In one implementation, Dp1 and Dp2 may be shorted on the host/device as close to the receptacle as possible to minimize stub length; Dn1 and Dn2 may also be shorted. The maximum shorting trace length should not exceed 3.5 mm. 2. All VBUS pins shall be connected together within the USB Type-C plug and shall be connected together at the USB Type-C receptacle connector when the receptacle is in its mounted condition (e.g., all VBUS pins bussed together on the PCB). 3. All Ground return pins shall be connected together within the USB Type-C plug and shall be connected together at the USB Type-C receptacle connector when the receptacle is in its mounted condition (e.g., all ground return pins bussed together on the PCB). 4. If the contact dimensions shown in Figure 3 1 ALTERNATE SECTION A-A are used, then the VBUS contacts (A4, A9, B4 and B9) mate second, and signal contacts (A2, A3, A5, A6, A7, A8, A10, A11, B2, B3, B5, B6, B7, B8, B10 and B11) mate third. 							

The usage and assignments of the signals necessary for the support of only **USB 2.0** with the USB Type-C mating interface are defined in Table 3-5.

Table 3-5 USB Type-C Receptacle Interface Pin Assignments for USB 2.0-only Support

Pin	Signal Name	Description	Mating Sequence	Pin	Signal Name	Description	Mating Sequence
A1	GND	Ground return	First	B12	GND	Ground return	First
A2				B11			
A3				B10			
A4	VBUS	Bus Power	First	B9	VBUS	Bus Power	First
A5	CC1	Configuration Channel	Second	B8	SBU2	Sideband Use (SBU)	Second
A6	Dp1	Positive half of the USB 2.0 differential pair – Position 1	Second	B7	Dn2	Negative half of the USB 2.0 differential pair – Position 2	Second
A7	Dn1	Negative half of the USB 2.0 differential pair – Position 1	Second	B6	Dp2	Positive half of the USB 2.0 differential pair – Position 2	Second
A8	SBU1	Sideband Use (SBU)	Second	B5	CC2	Configuration Channel	Second
A9	VBUS	Bus Power	First	B4	VBUS	Bus Power	First
A10				B3			
A11				B2			
A12	GND	Ground return	First	B1	GND	Ground return	First
Notes: 1. Unused contact locations shall be electrically isolated from power, ground or signaling (i.e., not connected). 2. Contacts B6 and B7 should not be present in the USB Type-C plug. The receptacle side shall support the USB 2.0 differential pair present on Dp1/Dn1 or Dp2/Dn2. The plug orientation determines which pair is active. In one implementation, Dp1 and Dp2 may be shorted on the host/device as close to the receptacle as possible to minimize stub length; Dn1 and Dn2 may also be shorted. The maximum shorting trace length should not exceed 3.5 mm. 3. Contacts A8 and B8 (SBU1 and SBU2) shall not be connected unless required for a specified purpose. 4. All VBUS pins shall be connected together within the USB Type-C plug and shall be connected together at the USB Type-C receptacle connector when the receptacle is in its mounted condition (e.g., all VBUS pins bussed together on the PCB). 5. All Ground return pins shall be connected together within the USB Type-C plug and shall be connected together at the USB Type-C receptacle connector when the receptacle is in its mounted condition (e.g., all ground return pins bussed together on the PCB). 6. If the contact dimensions shown in Figure 3 1 ALTERNATE SECTION A-A are used then the VBUS contacts (A4, A9, B4 and B9) mate second, and signal contacts (A5, A6, A7, A8, B5, B6, B7 and B8) mate third.							

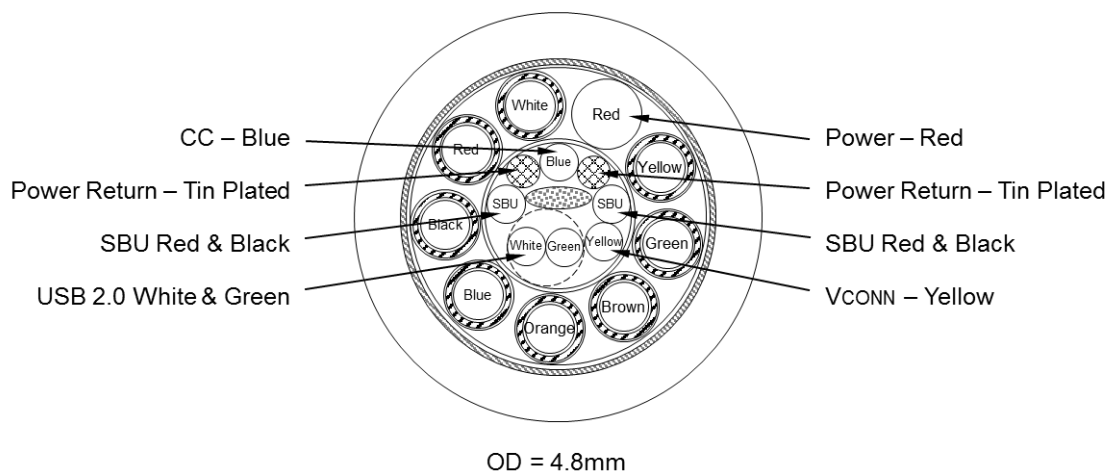
3.3 Cable Construction and Wire Assignments

This section discusses the USB Type-C cables, including cable construction, wire assignments, and wire gauges.

3.3.1 Cable Construction (Informative)

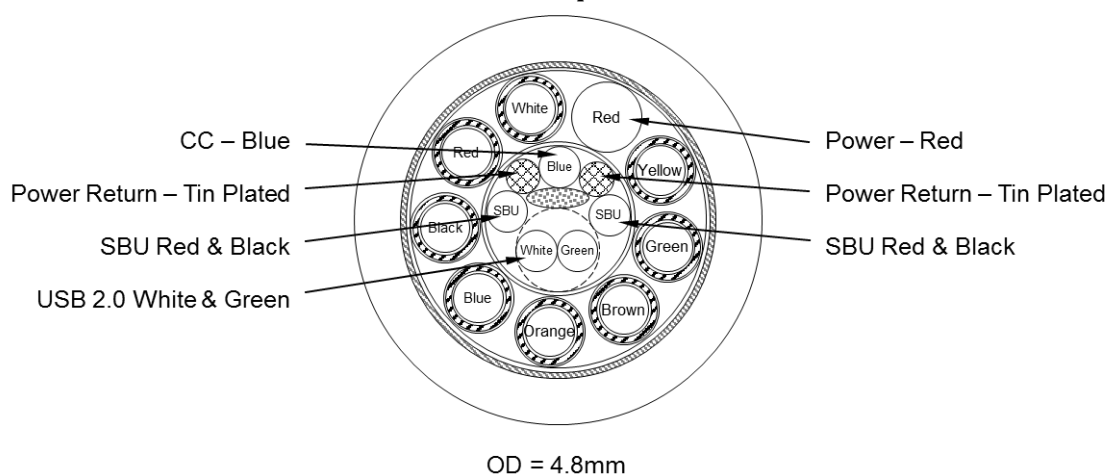
Figure 3-22 illustrates an example of USB Full-Featured Type-C cable cross-section, using micro-coaxial wires for TX/RX pairs. There are four groups of wires: USB D+/D– (typically unshielded twisted pairs (UTP)), TX/RX signal pairs (coaxial wires, twin-axial or shielded twisted pairs), sideband signal wires, and power and ground wires. In this example, the **optional** VCONN wire is shown whereas in Figure 3-23 the example is shown with the VCONN wire removed – the inclusion of VCONN or not relates to the implementation approach chosen for Electronically Marked Cables (See Section 4.9).

**Figure 3-22 Illustration of a USB Full-Featured Type-C Cable Cross Section,
a Coaxial Wire Example with VCONN**



Coax are TX/RX pairs – specific pairs not defined in cable

**Figure 3-23 Illustration of a USB Full-Featured Type-C Cable Cross Section,
a Coaxial Wire Example without VCONN**



Coax are TX/RX pairs – specific pairs not defined in cable

The USB D+/D– signal pair is intended to transmit the **USB 2.0** Low-Speed, Full-Speed and High-Speed signaling while the TX/RX signal pairs are used for either **USB 3.2** or **USB4** signaling. Shielding is needed for the TX/RX differential pairs for signal integrity and EMC performance.

3.3.2 Wire Assignments

Table 3-6 defines the full set of possible wires needed to produce all standard USB Type-C cables assemblies. For some cable assemblies, not all of these wires are used. For example, a USB Type-C cable that only provides **USB 2.0** functionality will not include wires 6–15.

Table 3-6 USB Type-C Standard Cable Wire Assignments

Wire Number	Signal Name	Description
1	GND_PWRrt1	Ground for power return
2	PWR_VBUS1	VBUS power
3	CC	Configuration Channel
4	UTP_Dp	Unshielded twist pair, positive
5	UTP_Dn	Unshielded twist pair, negative
6	SDPp1	Shielded differential pair #1, positive
7	SDPn1	Shielded differential pair #1, negative
8	SDPp2	Shielded differential pair #2, positive
9	SDPn2	Shielded differential pair #2, negative
10	SDPp3	Shielded differential pair #3, positive
11	SDPn3	Shielded differential pair #3, negative
12	SDPp4	Shielded differential pair #4, positive
13	SDPn4	Shielded differential pair #4, negative
14	SBU_A	Sideband Use
15	SBU_B	Sideband Use
16	GND_PWRrt2	Ground for power return (<i>optional</i>)
17	PWR_VBUS2	VBUS power (<i>optional</i>)
18	PWR_VCONN	VCONN power (<i>optional</i> , see Section 4.9)
Braid	Shield	Cable external braid
Note: 1. This table assumes that coaxial wire construction is used for all SDP's and there are no drain wires. The signal ground return is through the shields of the coaxial wires. If shielded twisted or twin-axial pairs are used, then drain wires are needed.		

Table 3-7 defines the full set of possible wires needed to produce USB Type-C to legacy cable assemblies. For some cable assemblies, not all of these wires are needed. For example, a USB Type-C to **USB 2.0** Standard-B cable will not include wires 5–10.

Table 3-7 USB Type-C Cable Wire Assignments for Legacy Cables/Adapters

Wire Number	Signal Name	Description
1	GND_PWRrt1	Ground for power return
2	PWR_VBUS1	VBUS power
3	UTP_Dp	Unshielded twist pair, positive
4	UTP_Dn	Unshielded twist pair, negative
5	SDPp1	Shielded differential pair #1, positive
6	SDPn1	Shielded differential pair #1, negative
7	SDP1_Drain	Drain wire for SDPp1 and SDPn1
8	SDPp2	Shielded differential pair #2, positive
9	SDPn2	Shielded differential pair #2, negative
10	SDP2_Drain	Drain wire for SDPp2 and SDPn2
Braid	Shield	Cable external braid
Note: 1. This table assumes that shielded twisted pair is used for all SDP's and there are drain wires. If coaxial wire construction is used, then no drain wires are needed, and the signal ground return is through the shields of the coaxial wires .		

3.3.3 Wire Gauges and Cable Diameters (*Informative*)

This specification does not specify wire gauge. Table 3-8 and Table 3-9 list typical wire gauges for reference purposes only. A large gauge wire incurs less loss, but at the cost of cable diameter and flexibility. Multiple wires *may* be used for a single wire such as for VBUS or Ground. It is recommended to use the smallest possible wire gauges that meet the cable assembly electrical and mechanical requirements.

To maximize cable flexibility, all wires *should* be stranded, and the cable outer diameter *should* be minimized as much as possible. A typical USB Full-Featured Type-C cable outer diameter *may* range from 4 mm to 6 mm while a typical **USB 2.0** Type-C cable outer diameter *may* range from 2 mm to 4 mm. A typical USB Type-C to **USB 3.1** legacy cable outer diameter *may* range from 3 mm to 5 mm.

Table 3-8 Reference Wire Gauges for standard USB Type-C Cable Assemblies

Wire Number	Signal Name	Wire Gauge (AWG)
1	GND_PWRrt1	20-28
2	PWR_VBUS1	20-28
3	CC	32-34
4	UTP_Dp	28-34
5	UTP_Dn	28-34
6	SDPp1	26-34
7	SDPn1	26-34
8	SDPp2	26-34
9	SDPn2	26-34
10	SDPp3	26-34
11	SDPn3	26-34
12	SDPp4	26-34
13	SDPn4	26-34
14	SBU_A	32-34
15	SBU_B	32-34
16	GND_PWRrt2	20-28
17	PWR_VBUS2	20-28
18	PWR_VCONN	32-34

Table 3-9 Reference Wire Gauges for standard USB Type-C to Legacy Cable Assemblies

Wire Number	Signal Name	Wire Gauge (AWG)
1	GND_PWRrt1	20-28
2	PWR_VBUS1	20-28
3	UTP_Dp	28-34
4	UTP_Dn	28-34
5	SDPp1	26-34
6	SDPn1	26-34
7	SDP1_Drain	28-34
8	SDPp2	26-34
9	SDPn2	26-34
10	SDP2_Drain	28-34

3.4 Standard USB Type-C Cable Assemblies

Two standard USB Type-C cable assemblies are defined and allowed by this specification. In addition, captive cables are allowed (see Section 3.4.3). Shielding (braid) is required to enclose all the wires in the USB Type-C cable. The shield **shall** be terminated to the plug metal shells. The shield **should** be physically connected to the plug metal shell as close to 360° as possible, to control EMC.

Note: Up through Release 1.4 of this specification, the TX and RX signals used in this specification were named SSTX and SSRX. With the introduction of **USB4**, these signals were renamed such that they generically can apply to both SuperSpeed USB and **USB4** signaling. It is intended that the TX and RX signal names are synonymous with the original SSTX and SSRX names for implementations prior to Release 2.0 of this specification.

3.4.1 USB Full-Featured Type-C Cable Assembly

Figure 3-24 shows a USB Full-Featured Type-C standard cable assembly.

Figure 3-24 USB Full-Featured Type-C Standard Cable Assembly

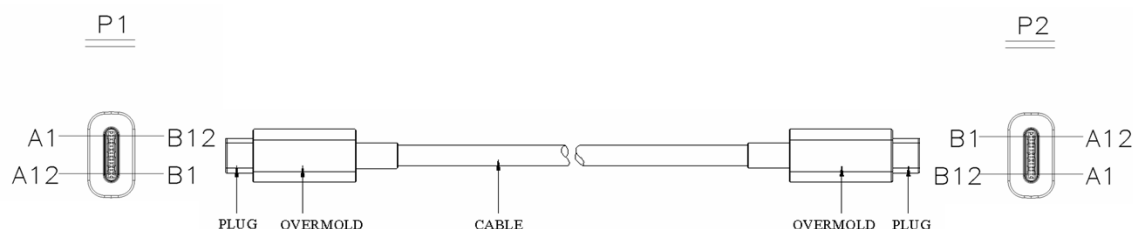


Table 3-10 defines the wire connections for the USB Full-Featured Type-C standard cable assembly.

Table 3-10 USB Full-Featured Type-C Standard Cable Assembly Wiring

USB Type-C Plug #1		Wire		USB Type-C Plug #2	
Pin	Signal Name	Wire Number	Signal Name	Pin	Signal Name
A1, B1, A12, B12	GND	1 [16]	GND_PWRrt1 [GND_PWRrt2]	A1, B1, A12, B12	GND
A4, B4, A9, B9	VBUS	2 [17]	PWR_VBUS1 [PWR_VBUS2]	A4, B4, A9, B9	VBUS
A5	CC	3	CC	A5	CC
B5	VCONN	18	PWR_VCONN (See 4.9)	B5	VCONN
A6	Dp1	4	UTP_Dp	A6	Dp1
A7	Dn1	5	UTP_Dn	A7	Dn1
A2	TXp1	6	SDPp1	B11	RXp1
A3	TXn1	7	SDPn1	B10	RXn1
B11	RXp1	8	SDPp2	A2	TXp1
B10	RXn1	9	SDPn2	A3	TXn1
B2	TXp2	10	SDPp3	A11	RXp2
B3	TXn2	11	SDPn3	A10	RXn2
A11	RXp2	12	SDPp4	B2	TXp2
A10	RXn2	13	SDPn4	B3	TXn2
A8	SBU1	14	SBU_A	B8	SBU2
B8	SBU2	15	SBU_B	A8	SBU1
Shell	Shield	Outer shield	Shield	Shell	Shield
Notes: 1. This table assumes that coaxial wire construction is used for all SDP's and there are no drain wires. The shields of the coaxial wires are connected to the ground pins. If shielded twisted pair is used, then drain wires are needed and shall be connected to the GND pins. 2. Pin B5 (VCONN) of the USB Type-C plug shall be used in electronically marked versions of this cable. See Section 4.9. 3. Contacts B6 and B7 should not be present in the USB Type-C plug. 4. All VBUS pins shall be connected together within the USB Type-C plug. A 10 nF bypass capacitor (minimum recommended voltage rating of 30 V, 63 V if EPR-capable) is required for the VBUS pin in the full-featured cable at each end of the cable. The bypass capacitor should be placed as close as possible to the power supply pad. 5. All GND pins shall be connected together within the USB Type-C plug 6. Shield and GND shall be connected within the USB Type-C plug on both ends of the cable assembly.					

3.4.1.1 Cable Certification and Identification

All USB Full-Featured Type-C cable assemblies **shall** be Electronically Marked and include information in the eMarker as defined by the **USB PD** specification.

All USB Full-Featured Type-C cable assemblies **should** be certified and visibly identified with certified cable identification icons as defined by the **USB-IF**. This enables end users to be able to confirm visually the functional capabilities of the cable in terms of the cable's power rating, either 60W or 240W, and the cable's data performance, e.g., 5 Gbps, 20 Gbps or higher.

3.4.2 USB 2.0 Type-C Cable Assembly

A **USB 2.0** Type-C standard cable assembly has the same form factor shown in Figure 3-24.

Table 3-11 defines the wire connections for the **USB 2.0** Type-C standard cable assembly.

Table 3-11 USB 2.0 Type-C Standard Cable Assembly Wiring

USB Type-C Plug #1		Wire		USB Type-C Plug #2	
Pin	Signal Name	Wire Number	Signal Name	Pin	Signal Name
A1, B1, A12, B12	GND	1 [16]	GND_PWRrt1 [GND_PWRrt2]	A1, B1, A12, B12	GND
A4, B4, A9, B9	VBUS	2 [17]	PWR_VBUS1 [PWR_VBUS2]	A4, B4, A9, B9	VBUS
A5	CC	3	CC	A5	CC
B5	VCONN	18	PWR_VCONN (See Section 4.9)	B5	VCONN
A6	Dp1	4	UTP_Dp	A6	Dp1
A7	Dn1	5	UTP_Dn	A7	Dn1
Shell	Shield	Outer shield	Shield	Shell	Shield
Notes: 1. Pin B5 (VCONN) of the USB Type-C plug shall be used in electronically marked versions of this cable. See Section 4.9. 2. Contacts B6 and B7 should not be present in the USB Type-C plug. 3. All VBUS pins shall be connected together within the USB Type-C plug. A bypass capacitor is not required for the VBUS pin in the USB 2.0 Type-C cable. 4. All GND pins shall be connected together within the USB Type-C plug. 5. All USB Type-C plug pins that are not listed in this table shall be open (not connected). 6. Shield and GND grounds shall be connected within the USB Type-C plug on both ends of the cable assembly.					

3.4.2.1 Cable Certification and Identification

All **USB 2.0** Type-C cable assemblies that are rated for higher than 3A **shall** be Electronically Marked and include information in the eMarker as defined by the **USB PD** specification.

All **USB 2.0** Type-C cable assemblies **should** be certified and visibly identified with certified cable identification icons as defined by the **USB-IF**. This enables end users to be able to confirm visually the functional capabilities of the cable in terms of the cable's power rating, either 60W or 240W.

3.4.3 USB Type-C Captive Cable Assembly

A captive cable assembly is a cable assembly that is terminated on one end with a USB Type-C plug and has a vendor-specific connect means (hardwired or custom detachable) on the opposite end. The cable assembly that is hardwired is not detachable from the device.

The assembly wiring for captive USB Type-C cables follow the same wiring assignments as the standard cable assemblies (see Table 3-10 and Table 3-11) with the exception that the hardwired attachment on the device side substitutes for the USB Type-C Plug #2 end.

The CC wire in a captive cable **shall** be terminated and behave as appropriate to the function of the product to which it is captive (e.g., host or device).

A device (Sink, UFP or DRP) with a captive cable assembly **shall** respond to SOP' cable identity inquiries when the device either sinks higher than 3A current or supports **USB4** operation. The physical location of the eMarker can be either within the captive cable or the device with the cable.

This specification does not define how the hardware attachment is physically done on the device side.

3.4.4 USB Type-C Thumb Drive Assembly

A thumb drive assembly is an assembly that incorporates a USB Type-C plug as its primary USB interface. This assembly does not functionally include a cable assembly.

A thumb drive device (Sink, UFP or DRP) **shall** respond to SOP' cable identity inquiries when it either sinks higher than 3A current or supports **USB4** operation.

3.5 Legacy Cable Assemblies

To enable interoperability between USB Type-C-based products and legacy USB products, the following standard legacy cable assemblies are defined. Only the cables defined within this specification are allowed.

Legacy cable assemblies that source power to a USB Type-C connector (e.g., a USB Type-C to USB Standard-A plug cable assembly and a USB Type-C plug to USB Micro-B receptacle adapter assembly) are required to use the Default **USB Type-C Current R_p** resistor (56 k Ω). The value of **R_p** is used to inform the Sink how much current the Source can provide. Since the legacy cable assembly does not comprehend the capability of the Source it is connected to, it is only allowed to advertise Default **USB Type-C Current** as defined by the **USB 2.0**, **USB 3.1**, and **USB BC 1.2** specifications. No other **R_p** values are permitted because these **may** cause a USB Type-C Sink to overload a legacy power supply.

3.5.1 USB Type-C to USB 3.1 Standard-A Cable Assembly

Figure 3-25 shows a USB Type-C to **USB 3.1** Standard-A cable assembly.

Figure 3-25 USB Type-C to USB 3.1 Standard-A Cable Assembly



Table 3-12 defines the wire connections for the USB Type-C to **USB 3.1** Standard-A cable assembly.

Table 3-12 USB Type-C to USB 3.1 Standard-A Cable Assembly Wiring

USB Type-C Plug		Wire		USB 3.1 Standard-A Plug	
Pin	Signal Name	Wire Number	Signal Name	Pin	Signal Name
A1, B1, A12, B12	GND	1 7, 10	GND_PWRrt1 SDP1_Drain, SDP2_Drain	4 7	GND GND_DRAIN
A4, B4, A9, B9	VBUS	2	PWR_VBUS1	1	VBUS
A5	CC	See Note 2			
B5	VCONN				
A6	Dp1	4	UTP_Dp	3	D+
A7	Dn1	5	UTP_Dn	2	D-
A2	TXp1	6	SDPp1	6	StdA_SSRX+
A3	TXn1	7	SDPn1	5	StdA_SSRX-
B11	RXp1	8	SDPp2	9	StdA_SSTX+
B10	RXn1	9	SDPn2	8	StdA_SSTX-
Shell	Shield	Outer shield	Shield	Shell	Shield

Notes:

1. This table assumes that shielded twisted pair is used for all SDP's and there are drain wires. If coaxial wire construction is used, then no drain wires are present, and the shields of the coaxial wires are connected to the ground pins.
2. Pin A5 (CC) of the USB Type-C plug **shall** be connected to VBUS through a resistor **Rp** (56 kΩ ± 5%). See Section 4.5.3.2.2 and Table 4-27 for the functional description and value of **Rp**.
3. Contacts B6 and B7 **should not** be present in the USB Type-C plug.
4. All VBUS pins **shall** be connected together within the USB Type-C plug. A bypass capacitor is required between the VBUS and ground pins in the USB Type-C plug side of the cable. The bypass capacitor **shall** be 10nF ± 20% in cables which incorporate a USB Standard-A plug. The bypass capacitor **shall** be placed as close as possible to the power supply pad.
5. All Ground return pins **shall** be connected together within the USB Type-C plug.
6. Shield and GND grounds **shall** be connected within the USB Type-C and **USB 3.1** Standard-A plugs on both ends of the cable assembly.
7. All USB Type-C plug pins that are not listed in this table **shall** be open (not connected).

3.5.2 USB Type-C to USB 2.0 Standard-A Cable Assembly

Figure 3-26 shows a USB Type-C to **USB 2.0** Standard-A cable assembly.

Figure 3-26 USB Type-C to USB 2.0 Standard-A Cable Assembly

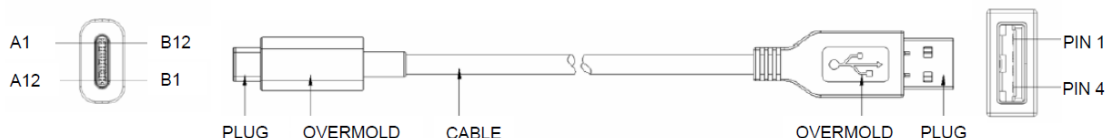


Table 3-13 defines the wire connections for the USB Type-C to **USB 2.0** Standard-A cable assembly.

Table 3-13 USB Type-C to USB 2.0 Standard-A Cable Assembly Wiring

USB Type-C Plug		Wire		USB 2.0 Standard-A Plug	
Pin	Signal Name	Wire Number	Signal Name	Pin	Signal Name
A1, B1, A12, B12	GND	1	GND_PWRrt1	4	GND
A4, B4, A9, B9	VBUS	2	PWR_VBUS1	1	VBUS
A5	CC	See Note 1			
B5	VCONN				
A6	Dp1	3	UTP_Dp	3	D+
A7	Dn1	4	UTP_Dn	2	D-
Shell	Shield	Outer shield	Shield	Shell	Shield
Notes: 1. Pin A5 (CC) of the USB Type-C plug shall be connected to VBUS through a resistor <i>R_p</i> (56 kΩ ± 5%). See Section 4.5.3.2.2 and Table 4-27 for the functional description and value of <i>R_p</i> . 2. Contacts B6 and B7 should not be present in the USB Type-C plug. 3. All VBUS pins shall be connected together within the USB Type-C plug. Bypass capacitors are not required for the VBUS pins in this cable. 4. All Ground return pins shall be connected together within the USB Type-C plug. 5. Shield and GND grounds shall be connected within the USB Type-C and USB 2.0 Standard-A plugs on both ends of the cable assembly. 6. All USB Type-C plug pins that are not listed in this table shall be open (not connected).					

3.5.3 USB Type-C to USB 3.1 Standard-B Cable Assembly

Figure 3-27 shows a USB Type-C to **USB 3.1** Standard-B cable assembly.

Figure 3-27 USB Type-C to USB 3.1 Standard-B Cable Assembly

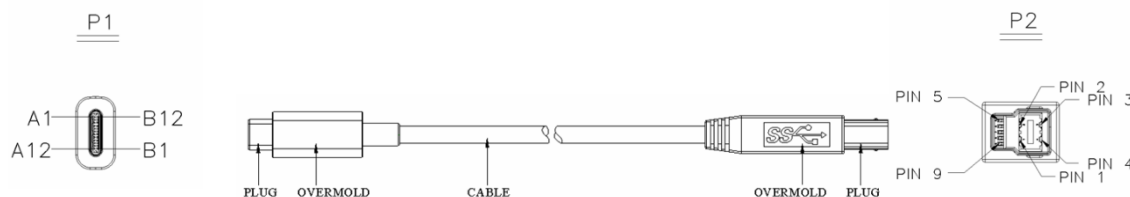


Table 3-14 defines the wire connections for the USB Type-C to **USB 3.1** Standard-B cable assembly.

Table 3-14 USB Type-C to USB 3.1 Standard-B Cable Assembly Wiring

USB Type-C Plug		Wire		USB 3.1 Standard-B Plug	
Pin	Signal Name	Wire Number	Signal Name	Pin	Signal Name
A1, B1, A12, B12	GND	1 7, 10	GND_PWRrt1 SDP1_Drain, SDP2_Drain	4 7	GND GND_DRAIN
A4, B4, A9, B9	VBUS	2	PWR_VBUS1	1	VBUS
A5	CC	See Note 1			
B5	VCONN				
A6	Dp1	4	UTP_Dp	3	D+
A7	Dn1	5	UTP_Dn	2	D-
A2	TXP1	6	SDPp1	9	StdB_SSRX+
A3	TXn1	7	SDPn1	8	StdB_SSRX-
B11	RXP1	8	SDPp2	6	StdB_SSTX+
B10	RXn1	9	SDPn2	5	StdB_SSTX-
Shell	Shield	Outer shield	Shield	Shell	Shield

Notes:

- Pin A5 (CC) of the USB Type-C plug **shall** be connected to GND through a resistor **Rd** (5.1 kΩ ± 20%). See Section 4.5.3.2.1 and Table 4-28 for the functional description and value of **Rd**.
- This table assumes that shielded twisted pair is used for all SDP's and there are drain wires. If coaxial wire construction is used, then no drain wires are present, and the shields of the coaxial wires are connected to the ground pins.
- Contacts B6 and B7 **should not** be present in the USB Type-C plug.
- All VBUS pins **shall** be connected together within the USB Type-C plug. A bypass capacitor is required between the VBUS and ground pins in the USB Type-C plug side of the cable. The bypass capacitor **shall** be 10nF ± 20% in cables which incorporate a USB Standard-B plug. The bypass capacitor **shall** be placed as close as possible to the power supply pad.
- All Ground return pins **shall** be connected together within the USB Type-C plug.
- Shield and GND grounds **shall** be connected within the USB Type-C and **USB 3.1** Standard-B plugs on both ends of the cable assembly.
- All USB Type-C plug pins that are not listed in this table **shall** be open (not connected).

3.5.4 USB Type-C to USB 2.0 Standard-B Cable Assembly

Figure 3-28 shows a USB Type-C to **USB 2.0** Standard-B cable assembly.

Figure 3-28 USB Type-C to USB 2.0 Standard-B Cable Assembly

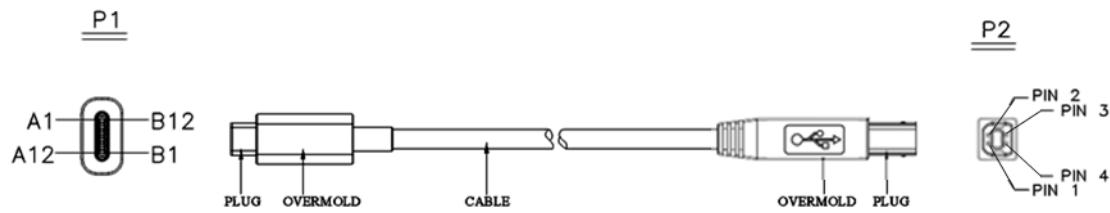


Table 3-15 defines the wire connections for the USB Type-C to **USB 2.0** Standard-B cable assembly.

Table 3-15 USB Type-C to USB 2.0 Standard-B Cable Assembly Wiring

USB Type-C Plug		Wire		USB 2.0 Standard-B Plug	
Pin	Signal Name	Wire Number	Signal Name	Pin	Signal Name
A1, B1, A12, B12	GND	1	GND_PWRrt1	4	GND
A4, B4, A9, B9	VBUS	2	PWR_VBUS1	1	VBUS
A5	CC	See Note 1			
B5	VCONN				
A6	Dp1	3	UTP_Dp	3	D+
A7	Dn1	4	UTP_Dn	2	D-
Shell	Shield	Outer shield	Shield	Shell	Shield
Notes:					
1. Pin A5 (CC) of the USB Type-C plug shall be connected to GND through a resistor Rd (5.1 kΩ ± 20%). See Section 4.5.3.2.1 and Table 4-28 for the functional description and value of Rd .					
2. Contacts B6 and B7 should not be present in the USB Type-C plug.					
3. All VBUS pins shall be connected together within the USB Type-C plug. Bypass capacitors are not required for the VBUS pins in this cable.					
4. All Ground return pins shall be connected together within the USB Type-C plug.					
5. Shield and GND grounds shall be connected within the USB Type-C and USB 2.0 Standard-B plugs on both ends of the cable assembly.					
6. All USB Type-C plug pins that are not listed in this table shall be open (not connected).					

3.5.5 USB Type-C to USB 2.0 Mini-B Cable Assembly

Figure 3-29 shows a USB Type-C to **USB 2.0** Mini-B cable assembly.

Figure 3-29 USB Type-C to USB 2.0 Mini-B Cable Assembly

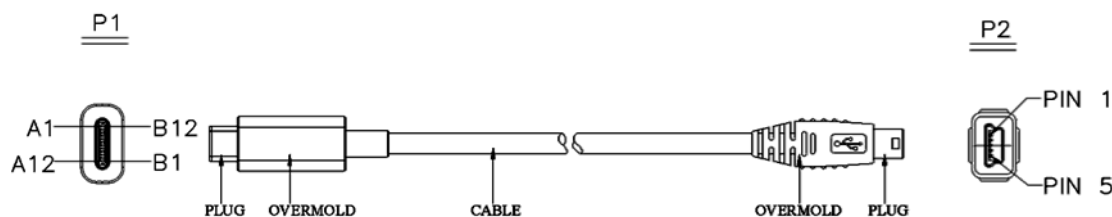


Table 3-16 defines the wire connections for the USB Type-C to **USB 2.0** Mini-B cable assembly.

Table 3-16 USB Type-C to USB 2.0 Mini-B Cable Assembly Wiring

USB Type-C Plug		Wire		USB 2.0 Mini-B Plug	
Pin	Signal Name	Wire Number	Signal Name	Pin	Signal Name
A1, B1, A12, B12	GND	1	GND_PWRrt1	5	GND
A4, B4, A9, B9	VBUS	2	PWR_VBUS1	1	VBUS
A5	CC	See Note 1			
B5	VCONN				
A6	Dp1	3	UTP_Dp	3	D+
A7	Dn1	4	UTP_Dn	2	D-
				4	ID
Shell	Shield	Outer shield	Shield	Shell	Shield

Notes:

1.

Pin A5 (CC) of the USB Type-C plug **shall** be connected to GND through a resistor **Rd** (5.1 kΩ ± 20%). See Section 4.5.3.2.1 and Table 4-28 for the functional description and value of **Rd**.

2.

Contacts B6 and B7 **should not** be present in the USB Type-C plug.

3.

All VBUS pins **shall** be connected together within the USB Type-C plug. Bypass capacitors are not required for the VBUS pins in this cable.

4.

All Ground return pins **shall** be connected together within the USB Type-C plug.

5.

Pin 4 (ID) of the **USB 2.0** Mini-B plug **shall** be terminated as defined in the applicable specification for the cable type.

6.

Shield and GND grounds **shall** be connected within the USB Type-C and **USB 2.0** Mini-B plugs on both ends of the cable assembly.

7.

All USB Type-C plug pins that are not listed in this table **shall** be open (not connected).

3.5.6 USB Type-C to *USB 3.1* Micro-B Cable Assembly

Figure 3-30 shows a USB Type-C to *USB 3.1* Micro-B cable assembly.

Figure 3-30 USB Type-C to *USB 3.1* Micro-B Cable Assembly

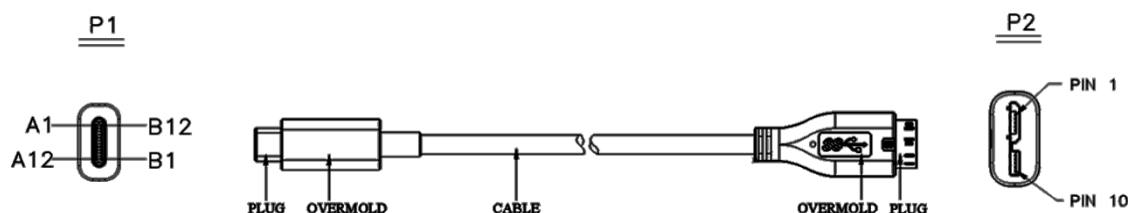


Table 3-17 defines the wire connections for the USB Type-C to *USB 3.1* Micro-B cable assembly.

Table 3-17 USB Type-C to *USB 3.1* Micro-B Cable Assembly Wiring

USB Type-C Plug		Wire		USB 3.1 Micro-B Plug	
Pin	Signal Name	Wire Number	Signal Name	Pin	Signal Name
A1, B1, A12, B12	GND	1 7, 10	GND_PWRrt1 SDP1_Drain, SDP2_Drain	5 8	GND GND_DRAIN
A4, B4, A9, B9	VBUS	2	PWR_VBUS1	1	VBUS
A5	CC	See Note 1			
B5	VCONN				
A6	Dp1	4	UTP_Dp	3	D+
A7	Dn1	5	UTP_Dn	2	D-
A2	TXp1	6	SDPp1	10	MicB_SSRX+
A3	TXn1	7	SDPn1	9	MicB_SSRX-
B11	RXp1	8	SDPp2	7	MicB_SSTX+
B10	RXn1	9	SDPn2	6	MicB_SSTX-
				4	ID
Shell	Shield	Outer shield	Shield	Shell	Shield

Notes:

- Pin A5 (CC) of the USB Type-C plug **shall** be connected to GND through a resistor *Rd* (5.1 kΩ ± 20%). See Section 4.5.3.2.1 and Table 4-28 for the functional description and value of *Rd*.
- This table assumes that shielded twisted pair is used for all SDP's and there are drain wires. If coaxial wire construction is used, then no drain wires are present, and the shields of the coaxial wires are connected to the ground pins.
- Contacts B6 and B7 **should not** be present in the USB Type-C plug.
- All VBUS pins **shall** be connected together within the USB Type-C plug. A bypass capacitor is required between the VBUS and ground pins in the USB Type-C plug side of the cable. The bypass capacitor **shall** be 10nF ± 20% in cables which incorporate a USB Micro-B plug. The bypass capacitor **shall** be placed as close as possible to the power supply pad.
- All Ground return pins **shall** be connected together within the USB Type-C plug.
- Pin 4 (ID) of the *USB 3.1* Micro-B plug **shall** be terminated as defined in the applicable specification for the cable type.
- Shield and GND grounds **shall** be connected within the USB Type-C and *USB 3.1* Micro-B plugs on both ends of the cable assembly.
- All USB Type-C plug pins that are not listed in this table **shall** be open (not connected).

3.5.7 USB Type-C to USB 2.0 Micro-B Cable Assembly

Figure 3-31 shows a USB Type-C to **USB 2.0** Micro-B cable assembly.

Figure 3-31 USB Type-C to USB 2.0 Micro-B Cable Assembly

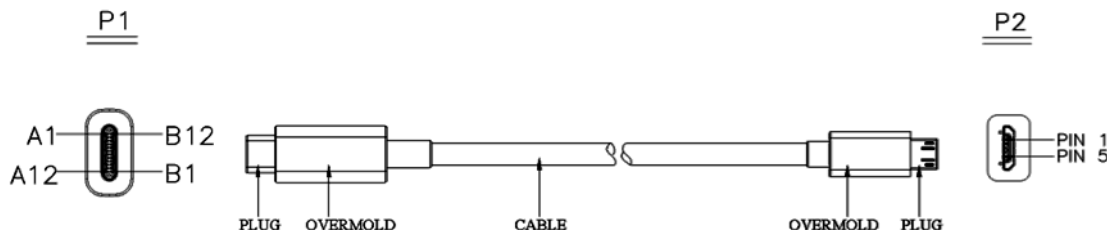


Table 3-18 defines the wire connections for the USB Type-C to **USB 2.0** Micro-B cable assembly.

Table 3-18 USB Type-C to USB 2.0 Micro-B Cable Assembly Wiring

USB Type-C Plug		Wire		USB 2.0 Micro-B Plug	
Pin	Signal Name	Wire Number	Signal Name	Pin	Signal Name
A1, B1, A12, B12	GND	1	GND_PWRrt1	5	GND
A4, B4, A9, B9	VBUS	2	PWR_VBUS1	1	VBUS
A5	CC	See Note 1			
B5	VCONN				
A6	Dp1	3	UTP_Dp	3	D+
A7	Dn1	4	UTP_Dn	2	D-
				4	ID
Shell	Shield	Outer shield	Shield	Shell	Shield

Notes:

- Pin A5 (CC) of the USB Type-C plug **shall** be connected to GND through a resistor **Rd** (5.1 kΩ ± 20%). See Section 4.5.3.2.1 and Table 4-28 for the functional description and value of **Rd**.
- Contacts B6 and B7 **should not** be present in the USB Type-C plug.
- All Vbus pins **shall** be connected together within the USB Type-C plug. Bypass capacitors are not required for the VBUS pins in this cable.
- All Ground return pins **shall** be connected together within the USB Type-C plug.
- Pin 4 (ID) of the **USB 2.0** Micro-B plug **shall** be terminated as defined in the applicable specification for the cable type.
- Shield and GND grounds **shall** be connected within the USB Type-C and **USB 2.0** Micro-B plugs on both ends of the cable assembly.
- All USB Type-C plug pins that are not listed in this table **shall** be open (not connected).

3.6 Legacy Adapter Assemblies

To enable interoperability between USB Type-C-based products and legacy USB products, the following standard legacy adapter assemblies are defined. Only the adapter assemblies defined in this specification are allowed.

3.6.1 USB Type-C to USB 3.1 Standard-A Receptacle Adapter Assembly

Figure 3-32 shows a USB Type-C to **USB 3.1** Standard-A receptacle adapter assembly. This cable assembly is defined for direct connect to a USB device (e.g., a thumb drive). System functionality of using this adaptor assembly together with another USB cable assembly is not guaranteed.

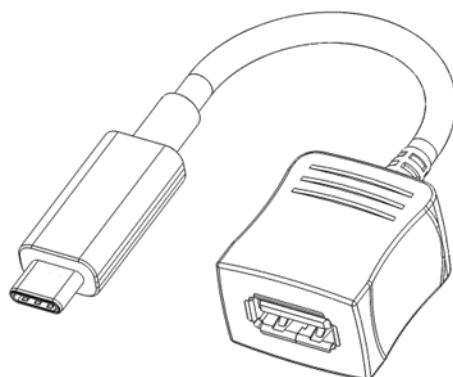
Figure 3-32 USB Type-C to *USB 3.1* Standard-A Receptacle Adapter Assembly

Table 3-19 defines the wire connections for the USB Type-C to ***USB 3.1*** Standard-A receptacle adapter assembly.

Table 3-19 USB Type-C to *USB 3.1* Standard-A Adapter Assembly Wiring

USB Type-C Plug		Wire (Optional – See Note 7)		<i>USB 3.1</i> Standard-A Receptacle	
Pin	Signal Name	Wire Number	Signal Name	Pin	Signal Name
A1, B1, A12, B12	GND	1 7, 10	GND_PWRrt1 SDP1_Drain, SDP2_Drain	4 7	GND GND_DRAIN
A4, B4, A9, B9	VBUS	2	PWR_VBUS1	1	VBUS
A5	CC	See Note 1			
B5	VCONN				
A6	Dp1	4	UTP_Dp	3	D+
A7	Dn1	5	UTP_Dn	2	D–
A2	TXp1	6	SDPp1	9	StdB_SSTX+
A3	TXn1	7	SDPn1	8	StdB_SSTX–
B11	RXp1	8	SDPp2	6	StdB_SSRX+
B10	RXn1	9	SDPn2	5	StdB_SSRX–
Shell	Shield	Outer shield	Shield	Shell	Shield

Notes:

- Pin A5 (CC) of the USB Type-C plug **shall** be connected to GND through a resistor ***Rd*** (5.1 kΩ ± 20%). See Section 4.5.3.2.1 and Table 4-28 for the functional description and value of ***Rd***.
- This table assumes that shielded twisted pair is used for all SDP's and there are drain wires. If coaxial wire construction is used, then no drain wires are present, and the shields of the coaxial wires are connected to the ground pins.
- Contacts B6 and B7 **should not** be present in the USB Type-C plug.
- All VBUS pins **shall** be connected together within the USB Type-C plug. A 10 nF bypass capacitor is required for the VBUS pin in the USB Type-C plug end of the cable. The bypass capacitor **should** be placed as close as possible to the power supply pad. A bypass capacitor is not required for the VBUS pin in the Standard-A receptacle.
- Shield and GND grounds **shall** be connected within the USB Type-C plug and ***USB 3.1*** Standard-A receptacle on both ends of the adapter assembly.
- All USB Type-C plug pins that are not listed in this table **shall** be open (not connected).
- This adapter assembly **may** or **may not** include a cable between the USB Type-C plug and ***USB 3.1*** Standard-A receptacle. Wire information provided here is when a cable is part of the assembly.

3.6.2 USB Type-C to USB 2.0 Micro-B Receptacle Adapter Assembly

Figure 3-33 shows a USB Type-C to **USB 2.0** Micro-B receptacle adapter assembly.

Figure 3-33 USB Type-C to USB 2.0 Micro-B Receptacle Adapter Assembly

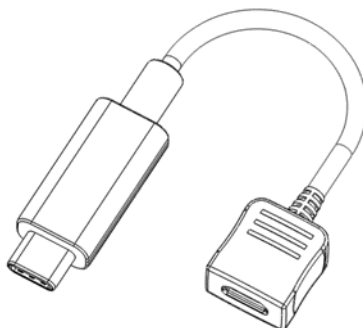


Table 3-20 defines the wire connections for the USB Type-C to **USB 2.0** Micro-B receptacle adapter assembly.

Table 3-20 USB Type-C to USB 2.0 Micro-B Receptacle Adapter Assembly Wiring

USB Type-C Plug		Wire (Optional – See Note 6)		USB 2.0 Micro-B Receptacle	
Pin	Signal Name	Wire Number	Signal Name	Pin	Signal Name
A1, B1, A12, B12	GND	1	GND_PWRrt1	5	GND
A4, B4, A9, B9	VBUS	2	PWR_VBUS1	1	VBUS
A5	CC	See Note 1			
B5	VCONN				
A6	Dp1	3	UTP_Dp	3	D+
A7	Dn1	4	UTP_Dn	2	D–
				4	ID
Shell	Shield	Outer shield	Shield	Shell	Shield
Notes: <ol style="list-style-type: none">Pin A5 (CC) of the USB Type-C plug shall be connected to VBUS through a resistor <i>R_p</i> (56 kΩ ± 5%). See Section 4.5.3.2.2 and Table 4-27 for the functional description and value of <i>R_p</i>.Contacts B6 and B7 should not be present in the USB Type-C plug.All VBUS pins shall be connected together within the USB Type-C plug. Bypass capacitors are not required for the VBUS pins at the Micro-B receptacle end of this cable.Shield and GND grounds shall be connected within the USB Type-C plug and USB 2.0 Micro-B receptacle on both ends of the adapter assembly.All USB Type-C plug pins that are not listed in this table shall be open (not connected).This adapter assembly may or may not include a cable between the USB Type-C plug and USB 2.0 Micro-B receptacle. Wire information provided here is when a cable is part of the assembly.					

3.7 Electrical Characteristics

This section defines the USB Type-C raw cable, connector, and cable assembly electrical requirements, including signal integrity, shielding effectiveness, and DC requirements. Section 3.11.1 defines additional requirements regarding functional signal definition, host/device discovery and configuration, and power delivery.

Unless otherwise specified, all measurements are made at a temperature of 15° to 35° C, a relative humidity of 25% to 85%, and an atmospheric pressure of 86 to 106 kPa and all S-parameters are normalized with an 85 Ω differential impedance.

3.7.1 Raw Cable (*Informative*)

Informative raw cable electrical performance targets are provided to help cable assembly manufacturers manage the procurement of raw cable. These targets are not part of the USB Type-C compliance requirements. The **normative** requirement is that the cable assembly meets the performance characteristics specified in Sections 3.7.2 and 3.7.5.3.

The differential characteristic impedance for shielded differential pairs is recommended to be 90 $\Omega \pm 5 \Omega$. The single-ended characteristic impedance of coaxial wires is recommended to be 45 $\Omega \pm 3 \Omega$. The impedance **should** be evaluated using a 200 ps (10%-90%) rise time; a faster rise time is not necessary for raw cable since it will make cable test fixture discontinuities more prominent.

3.7.1.1 Intra-Pair Skew (*Informative*)

The intra-pair skew for a differential pair is recommended to be less than 10 ps/m. It **should** be measured with a Time Domain Transmission (TDT) in a differential mode using a 200 ps (10%-90%) rise time with a crossing at 50% of the input voltage.

3.7.1.2 Differential Insertion Loss (*Informative*)

Cable loss depends on wire gauges, plating, and dielectric materials. Table 3-21 and Table 3-22 show examples of differential insertion losses.

Table 3-21 Differential Insertion Loss Examples for USB TX/RX with Twisted Pair Construction

Frequency	34AWG	32AWG	30AWG	28AWG
0.625 GHz	-1.8 dB/m	-1.4 dB/m	-1.2 dB/m	-1.0 dB/m
1.25 GHz	-2.5 dB/m	-2.0 dB/m	-1.7 dB/m	-1.4 dB/m
2.50 GHz	-3.7 dB/m	-2.9 dB/m	-2.5 dB/m	-2.1 dB/m
5.00 GHz	-5.5 dB/m	-4.5 dB/m	-3.9 dB/m	-3.1 dB/m
7.50 GHz	-7.0 dB/m	-5.9 dB/m	-5.0 dB/m	-4.1 dB/m
10.00 GHz	-8.4 dB/m	-7.2 dB/m	-6.1 dB/m	-4.8 dB/m
12.50 GHz	-9.5 dB/m	-8.2 dB/m	-7.3 dB/m	-5.5 dB/m
15.00 GHz	-11.0 dB/m	-9.5 dB/m	-8.7 dB/m	-6.5 dB/m

Table 3-22 Differential Insertion Loss Examples for USB TX/RX with Coaxial Construction

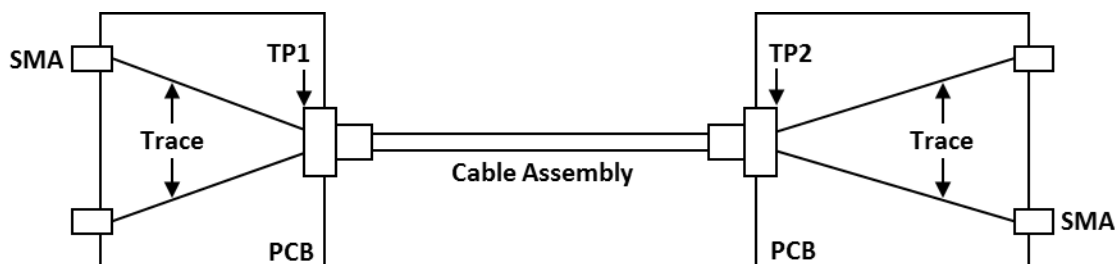
Frequency	34AWG	32AWG	30AWG	28AWG
0.625 GHz	-1.8 dB/m	-1.5 dB/m	-1.2 dB/m	-1.0 dB/m
1.25 GHz	-2.8 dB/m	-2.2 dB/m	-1.8 dB/m	-1.3 dB/m
2.50 GHz	-4.2 dB/m	-3.4 dB/m	-2.7 dB/m	-1.9 dB/m
5.00 GHz	-6.1 dB/m	-4.9 dB/m	-4.0 dB/m	-3.1 dB/m
7.50 GHz	-7.6 dB/m	-6.5 dB/m	-5.2 dB/m	-4.2 dB/m
10.0 GHz	-8.8 dB/m	-7.6 dB/m	-6.1 dB/m	-4.9 dB/m
12.5 GHz	-9.9 dB/m	-8.6 dB/m	-7.1 dB/m	-5.7 dB/m
15.0 GHz	-12.1 dB/m	-10.9 dB/m	-9.0 dB/m	-6.5 dB/m

3.7.2 USB Type-C to USB Type-C Passive Cable Assemblies

A USB Type-C to Type-C cable assembly **shall** be tested using a test fixture with the receptacle tongue fabricated in the test fixture. This is illustrated in Figure 3-34. The USB Type-C receptacles are not present in the test fixture. Hosts and devices **should** account for the additional signal degradation the receptacle introduces.

The requirements are for the entire signal path of the cable assembly mated with the fixture PCB tongues, not including lead-in PCB traces. As illustrated in Figure 3-34, the measurement is between TP1 (test point 1) and TP2 (test point 2). Refer to documentation located at Cables and Connectors page on the **USB-IF** website for a detailed description of a standardized test fixture.

Figure 3-34 Illustration of Test Points for a Mated Cable Assembly



The cable assembly requirements are divided into **informative** and **normative** requirements. The **informative** requirements are provided as design targets for cable assembly manufacturers. The **normative** requirements are the pass/failure criteria for cable assembly compliance.

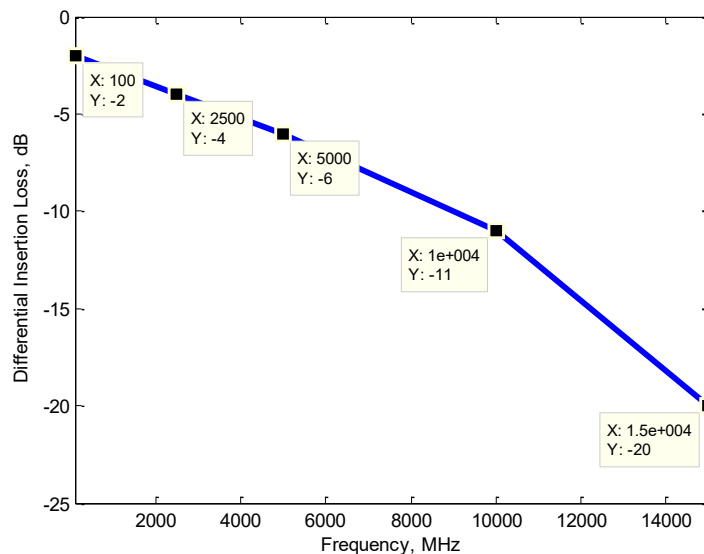
3.7.2.1 Recommended TX/RX Passive Cable Assembly Characteristics (USB 3.2 Gen2 and USB4 Gen2)

The recommended electrical characteristics defined in this section are **informative** design guidelines. Cable assemblies that do not meet these recommended electrical characteristics **may** still pass USB certification testing. Similarly, cable assemblies that meet these recommended electrical characteristics **may** or **may not** pass USB certification testing.

3.7.2.1.1 Differential Insertion Loss (Informative – USB 3.2 Gen2 and USB4 Gen2)

Figure 3-35 shows the differential insertion loss limit for a **USB 3.2 Gen2** or a **USB4 Gen2** Type-C cable assembly, which is defined by the following vertices: (100 MHz, -2 dB), (2.5 GHz, -4 dB), (5.0 GHz, -6 dB), (10 GHz, -11 dB) and (15 GHz, -20 dB).

**Figure 3-35 Recommended Differential Insertion Loss Requirement
(USB 3.2 Gen2 and USB4 Gen2)**

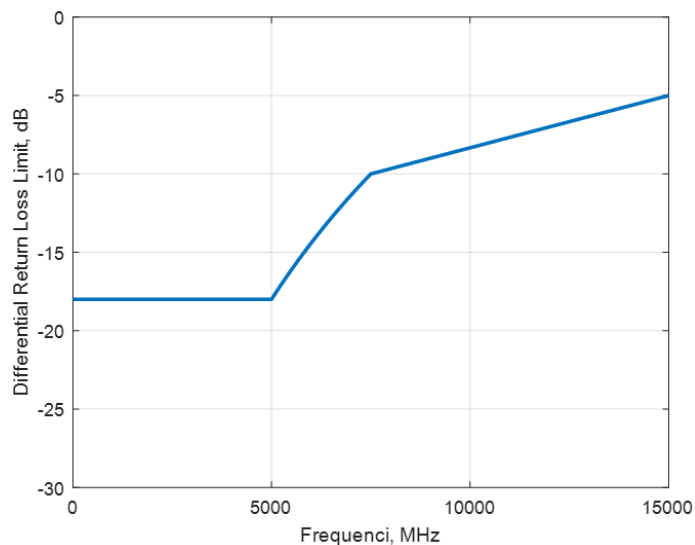


3.7.2.1.2 Differential Return Loss (*Informative – USB 3.2 Gen2 and USB4 Gen2*)

Figure 3-36 shows the differential return loss limit, which is defined by the following equation:

$$RL\ mask = \begin{cases} -18\ dB, & f < 5\ GHz \\ -18 + 45.43 \times \log_{10}\left(\frac{f}{5}\right)\ dB, & 5\ GHz < f \leq 7.5\ GHz \\ -15 + \frac{2}{3} \times f\ dB, & 7.5\ GHz < f \leq 15\ GHz \end{cases}$$

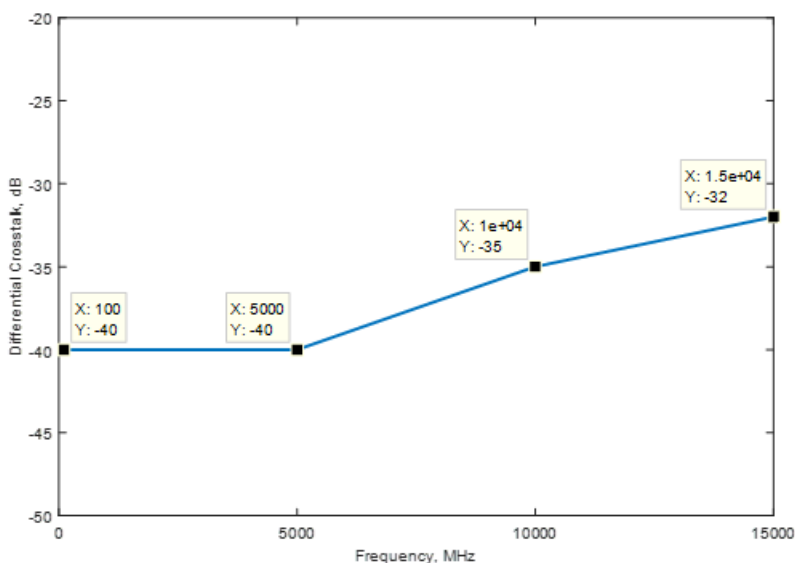
Figure 3-36 Recommended Differential Return Loss Requirement



3.7.2.1.3 Differential Near-End and Far-End Crosstalk between TX/RX Pairs (*Informative – USB 3.2 Gen2 and USB4 Gen2*)

Both the near-end crosstalk (DDNEXT) and far-end crosstalk (DDFEXT) are specified, as shown in Figure 3-37. The DDNEXT/DDFEXT limits are defined by the following vertices: (100 MHz, -40 dB), (5 GHz, -40 dB), (10 GHz, -35 dB), and (15 GHz, -32 dB).

Figure 3-37 Recommended Differential Crosstalk Requirement

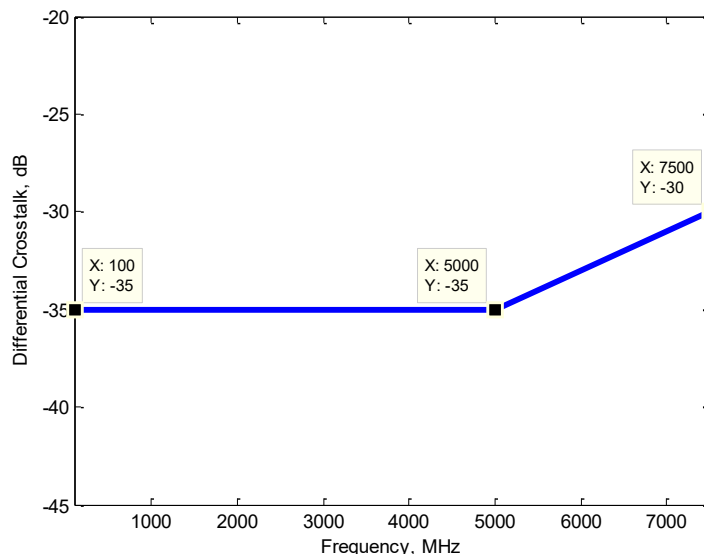


3.7.2.1.4 Differential Crosstalk between USB D+/D- and TX/RX Pairs (*Informative – USB 3.2 Gen2 and USB4 Gen2*)

The differential near-end and far-end crosstalk between the USB D+/D- pair and the TX/RX pairs **should** be managed not to exceed the limits shown in Figure 3-38. The USB D+/D- pair and the TX/RX pairs **should** be considered in the context of both an aggressor and a victim. It **should** also be considered that the D+/D- pair maximum frequency for similar tests is

1.2 GHz (see Table 3-31), but in this case the crosstalk on the D+/D- pair is extended to 7.5 GHz. The limits are defined by the following points: (100 MHz, -35 dB), (5 GHz, -35 dB), and (7.5 GHz, -30 dB).

Figure 3-38 Recommended Differential Near-End and Far-End Crosstalk Requirement between USB D+/D- Pair and TX/RX Pair



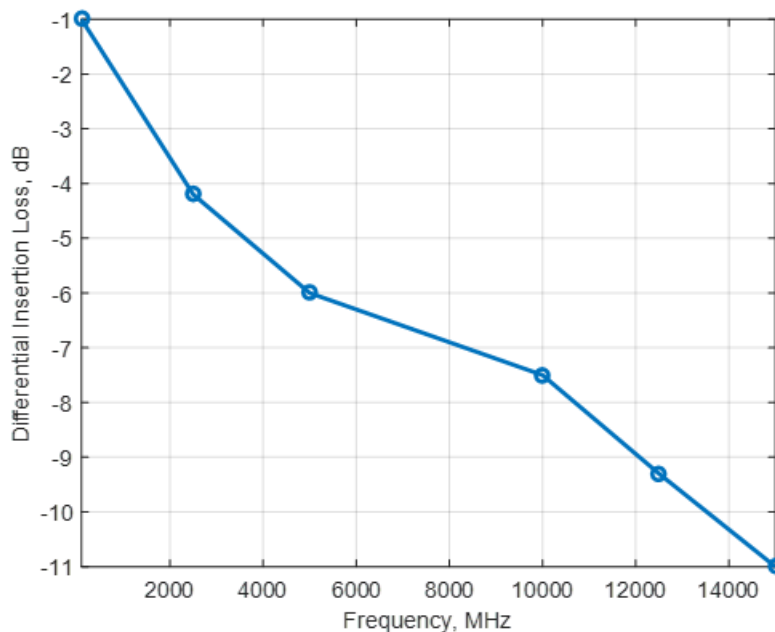
3.7.2.2 Recommended TX/RX Passive Cable Assembly Characteristics (*USB4* Gen3/Gen4)

Note: While the recommended characteristics in this section are defined in terms of *USB4* Gen3, these characteristics apply also to *USB4* Gen4 passive cable assemblies as these are electrically equivalent to *USB4* Gen3 passive cable assemblies.

3.7.2.2.1 Differential Insertion Loss (*Informative* – *USB4* Gen3/Gen4)

Figure 3-39 shows the recommended differential insertion loss limit for a *USB4* Gen3 Type-C cable assembly, which is defined by the following vertices: (100 MHz, -1 dB), (2.5 GHz, -4.2 dB), (5.0 GHz, -6 dB), (10 GHz, -7.5 dB), (12 GHz, -9.3 dB), and (15 GHz, -11 dB).

Figure 3-39 Recommended Differential Insertion Loss Requirement (*USB4* Gen3/Gen4)



3.7.2.2.2 Differential Return Loss (*Informative – USB4 Gen3/Gen4*)

The *informative* differential return loss mask is identical in Section 3.7.2.1.2.

3.7.2.2.3 Differential Near-End and Far-End Crosstalk between TX/RX Pairs (*Informative – USB4 Gen3/Gen4*)

The recommended near-end crosstalk (DDNEXT) and far-end crosstalk (DDFEXT) are defined in Section 3.7.2.1.3. To minimize crosstalk, it is important to optimize the paddle card and wire termination designs inside the cable plug.

3.7.2.2.4 Differential Crosstalk between USB D+/D– and TX/RX Pairs (*Informative – USB4 Gen3/Gen4*)

The *informative* near-end and far-end crosstalk between the USB D+/D– pair and the TX/RX pairs are the same as in Section 3.7.2.1.4.

3.7.2.3 Normative TX/RX Passive Cable Assembly Requirements for USB 3.2 Gen2 and USB4 Gen2

The integrated parameters are used for cable assembly compliance (except for insertion loss and differential-to-common-mode conversion) to avoid potential rejection of a functioning cable assembly that *may* fail the traditional S-parameters spec at a few frequencies.

3.7.2.3.1 Insertion Loss Fit at Nyquist Frequencies (*Normative – USB 3.2 Gen2 and USB4 Gen2*)

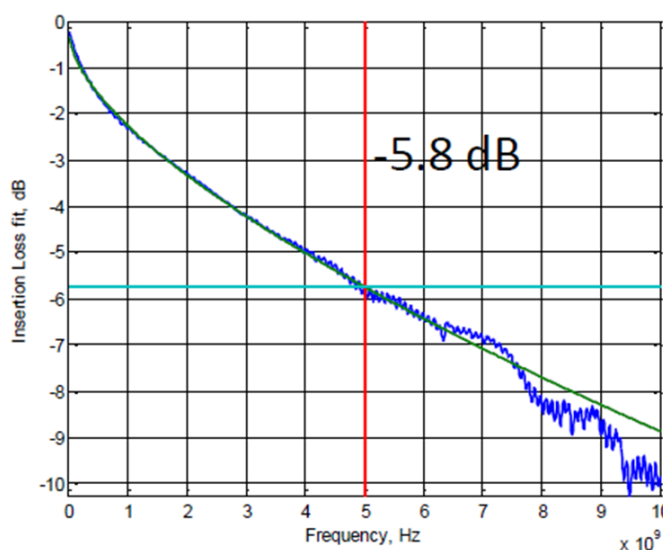
The insertion loss fit at Nyquist frequency measures the attenuation of the cable assembly. To obtain the insertion loss fit at Nyquist frequency, the measured cable assembly differential insertion loss is fitted with a smooth function. A standard fitting algorithm and tool *shall* be used to extract the insertion loss fit at Nyquist frequencies. The fitting equation is defined by the following equation:

$$IL_{fit} = a + b * \sqrt{f} + c * \sqrt{f^2} + d * \sqrt{f^3}$$

where f is the frequency and a , b , c , and d are the fitting coefficients.

Figure 3-40 illustrates an example of a measured cable assembly insertion loss fitted with a smooth function; the insertion loss fit at the Nyquist frequency of SuperSpeed USB Gen2 (5.0 GHz) is –5.8 dB.

Figure 3-40 Illustration of Insertion Loss Fit at Nyquist Frequency



The insertion loss fit at Nyquist frequency (ILfitatNq) **shall** meet the following requirements:

- ≥ -4 dB at 2.5 GHz,
- ≥ -6 dB at 5 GHz, and
- ≥ -11 dB at 10 GHz.

2.5 GHz, 5.0 GHz and 10 GHz are the Nyquist frequencies for SuperSpeed USB Gen1, SuperSpeed USB Gen2, and **USB4** Gen3 data rate, respectively.

3.7.2.3.2 Integrated Multi-Reflection (**Normative – USB 3.2 Gen2 and USB4 Gen2**)

The insertion loss deviation, ILD, is defined as:

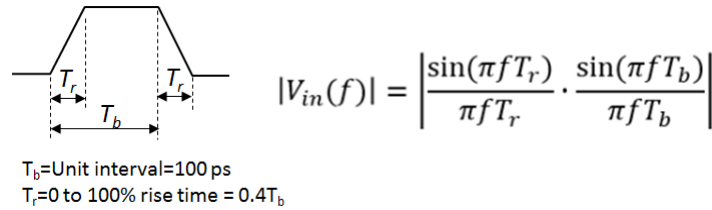
$$ILD(f) = IL(f) - ILfit(f)$$

It measures the ripple of the insertion loss, caused by multiple reflections inside the cable assembly (mated with the fixture). The integration of $ILD(f)$ is called the integrated multi-reflection (IMR):

$$IMR = dB \left(\sqrt{\frac{\int_0^{f_{max}} |ILD(f)|^2 |Vin(f)|^2 df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

where $f_{max} = 12.5$ GHz and $Vin(f)$ is the input trapezoidal pulse spectrum, defined in Figure 3-41.

Figure 3-41 Input Pulse Spectrum

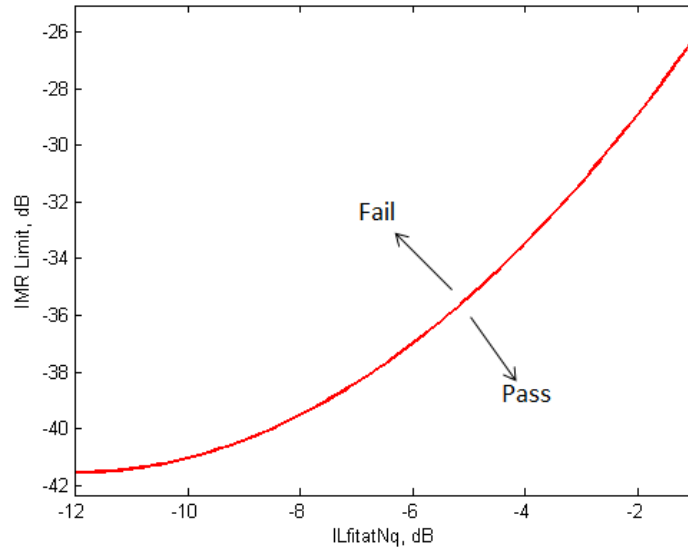


IMR has dependency on ILfitatNq. More IMR **may** be tolerated when ILfitatNq decreases. The IMR limit is specified as a function of ILfitatNq:

$$IMR \leq 0.126 \cdot ILfitatNq^2 + 3.024 \cdot ILfitatNq - 23.392$$

This is plotted in Figure 3-42.

Figure 3-42 IMR Limit as Function of ILfitatNq



3.7.2.3.3 Integrated Crosstalk between TX/RX Pairs (*Normative – USB 3.2 Gen2 and USB4 Gen2*)

The integrated crosstalk between all TX/RX pairs is calculated with the following equations:

$$INEXT = dB \left(\sqrt{\frac{\int_0^{f_{max}} (|Vin(f)|^2 (|NEXT(f)|^2 + 0.125^2 \cdot |C2D(f)|^2) + |Vdd(f)|^2 |NEXTd(f)|^2) df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

$$IFEXT = dB \left(\sqrt{\frac{\int_0^{f_{max}} (|Vin(f)|^2 (|FEXT(f)|^2 + 0.125^2 \cdot |C2D(f)|^2) + |Vdd(f)|^2 |FEXTd(f)|^2) df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

where $NEXT(f)$, $FEXT(f)$, and $C2D(f)$ are the measured near-end and far-end crosstalk between TX/RX pairs, and the common-mode-to-differential conversion, respectively. The factor of 0.1252 accounts for the assumption that the common mode amplitude is 12.5% of the differential amplitude. $NEXTd(f)$ and $FEXTd(f)$ are, respectively, the near-end and far-end crosstalk from the D+/D- pair to TX/RX pairs. $Vdd(f)$ is the input pulse spectrum evaluated using the equation in Figure 3-41 with $T_b=2.08$ ns.

The integration **shall** be done for each NEXT and FEXT between all differential pairs. The largest values of INEXT and IFEXT **shall** meet the following requirements:

- $INEXT \leq -40$ dB to 12.5GHz, for TX1 to RX1, TX2 to RX2, TX1 to RX2, TX2 to RX1, TX1 to TX2, and RX1 to RX2,
- $IFEXT \leq -40$ dB to 12.5GHz, for TX1 to RX1, TX2 to RX2, TX1 to RX2, TX2 to RX1, TX1 to TX2, and RX1 to RX2.

The port-to-port crosstalk (TX1 to RX2, TX2 to RX1, TX1 to TX2, and RX1 to RX2) is specified to support the usages in which all the four pairs transmit or receive signals simultaneously, for example in SuperSpeed USB dual-lane operation.

3.7.2.3.4 Integrated Crosstalk from TX/RX Pairs to USB D+/D- (Normative – USB 3.2 Gen2 and USB4 Gen2)

Crosstalk from the TX/RX pairs to **USB 2.0** D+/D- **shall** be controlled to ensure the robustness of the **USB 2.0** link. Since USB Type-C to Type-C Full-Featured cable assemblies **may** support the usage of **USB 3.2**, **USB4** or an **Alternate Mode** (e.g., **DisplayPort**), the crosstalk from the four high speed differential pairs to D+/D- **may** be from near-end crosstalk, far-end crosstalk, or a combination of the two. The integrated crosstalk to D+/D- is calculated with the following equations:

$$IDDXT_1NEXT + FEXT = dB \left(\sqrt{\frac{\int_0^{f_{max}} |Vin(f)|^2 (|NEXT1(f)|^2 + |FEXT(f)|^2) df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

where:

NEXT = Near-end crosstalk from TX pair to D+/D-,

FEXT = Far-end crosstalk from RX pair to D+/D-, and

$f_{max} = 1.2$ GHz.

$$IDDXT_2NEXT = dB \left(\sqrt{\frac{\int_0^{f_{max}} |Vin(f)|^2 (|NEXT1(f)|^2 + |NEXT2(f)|^2) df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

where:

NEXT1 = Near-end crosstalk from TX pair to D+/D-,

NEXT2 = Near-end crosstalk from RX (the RX functioning in TX mode) pair to D+/D-, and

$f_{max} = 1.2$ GHz.

The integration **shall** be done for *NEXT* + *FEXT* and *2NEXT* on D+/D- from the two differential pairs located at A2, A3, B10 and B11 (see Figure 2-2) and for *NEXT* + *FEXT* and *2NEXT* on D+/D- from the two differential pairs located at B2, B3 A10 and A11 (see Figure 2-2). Measurements are made in two sets to minimize the number of ports required for each measurement. The integrated differential crosstalk on D+/D- **shall** meet the following requirements:

- $IDDXT_1NEXT + FEXT \leq -34.5$ dB, and
- $IDDXT_2NEXT \leq -33$ dB.

3.7.2.3.5 Integrated Return Loss (Normative – USB 3.2 Gen2 and USB4 Gen2)

The integrated return loss (IRL) manages the reflection between the cable assembly and the rest of the system (host and device). It is defined as:

$$IRL = dB \left(\sqrt{\frac{\int_0^{f_{max}} |Vin(f)|^2 |SDD21(f)|^2 (|SDD11(f)|^2 + |SDD22(f)|^2) df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

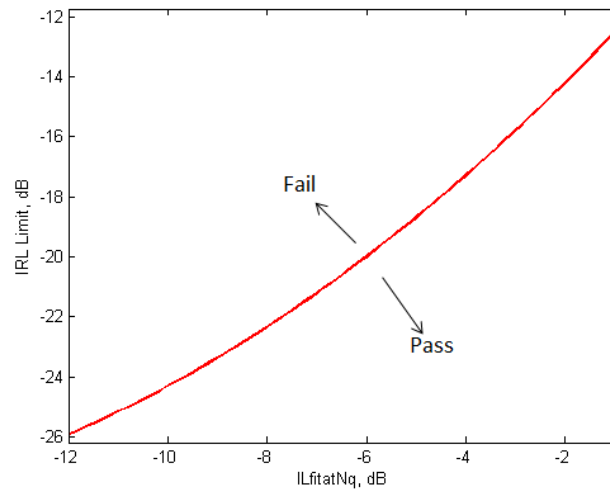
where *SDD21(f)* is the measured cable assembly differential insertion loss, *SDD11(f)* and *SDD22(f)* are the measured cable assembly return losses on the left and right sides, respectively, of a differential pair.

The IRL also has a strong dependency on *ILfitatNq*, and its limit is specified as a function of *ILfitatNq*:

$$IRL \leq 0.046 \cdot ILfitatNq^2 + 1.812 \cdot ILfitatNq - 10.784.$$

It is shown in Figure 3-43.

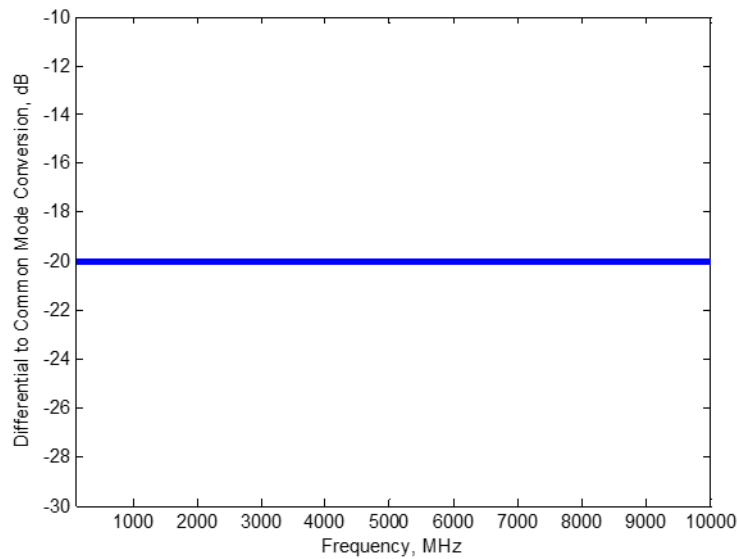
Figure 3-43 IRL Limit as Function of ILfitatNq



3.7.2.3.6 Differential-to-Common Mode Conversion (*Normative – USB 3.2 Gen2 and USB4 Gen2*)

The differential-to-common-mode conversion is specified to control the injection of common mode noise from the cable assembly into the host or device. Figure 3-44 illustrates the differential-to-common mode conversion (SCD12/SCD21) requirement. A mated cable assembly passes if its SCD12/SCD21 is less than or equal to -20 dB from 100 MHz to 10 GHz.

Figure 3-44 Differential-to-Common Mode Conversion Requirement



3.7.2.4 Normative TX/RX Passive Cable Assembly Requirements for **USB 3.2 Gen1** and **USB4 Gen2**

3.7.2.4.1 Insertion Loss Fit at Nyquist Frequencies (**Normative – USB 3.2 Gen1 and USB4 Gen2**)

The integrated S-parameter requirements for **USB 3.2 Gen1** and **USB4 Gen2** follow the same methodology as defined in Section 3.7.2.3. There are parameter adjustments made to suit the **USB4 Gen2** data rate. Unless otherwise specified, the following parameters **shall** be used to calculate insertion loss fit and integrated parameters:

- T_b , the unit interval, is set to 100 ps, reflecting the **USB4 Gen2** data rate.
- T_r , the rise time, remains at $0.4 * T_b$.
- f_{max} , the maximum frequency over which the integration or fitting is performed is increased to 12.5 GHz.
- The fitting equation is defined by the following equation:

$$IL_{fit} = a + b * \sqrt{f} + c * \sqrt{f^2} + d * \sqrt{f^3}$$

where f is the frequency and a , b , c , and d are the fitting coefficients.

The insertion loss fit at Nyquist frequency ($IL_{fitatNq}$) **shall** meet the following requirements:

- ≥ -7.0 dB at 2.5 GHz, and
- > -11.5 dB at 5 GHz.

3.7.2.4.2 Integrated Multi-Reflection (**Normative – USB 3.2 Gen1 and USB4 Gen2**)

The insertion loss deviation, ILD, is defined as:

$$ILD(f) = IL(f) - IL_{fit}(f)$$

It measures the ripple of the insertion loss, caused by multiple reflections inside the cable assembly (mated with the fixture). The integration of $ILD(f)$ is called the integrated multi- reflection (IMR):

$$IMR = dB \left(\sqrt{\frac{\int_0^{f_{max}} |ILD(f)|^2 |Vin(f)|^2 df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

where $f_{max} = 12.5$ GHz and $Vin(f)$ is the input trapezoidal pulse spectrum.

For **USB 3.2 Gen1** and **USB4 Gen2** cable assemblies, IMR limit is specified as:

$$IMR \leq 0.126 \cdot IL_{fitatNq}^2 + 3.024 \cdot IL_{fitatNq} - 24.792$$

3.7.2.4.3 Integrated Crosstalk between TX/RX Pairs (**Normative – USB 3.2 Gen1 and USB4 Gen2**)

The integrated crosstalk between all TX/RX pairs is calculated with the following equations:

$$INEXT = dB \left(\sqrt{\frac{\int_0^{f_{max}} (|Vin(f)|^2 (|NEXT(f)|^2 + 0.125^2 \cdot |C2D(f)|^2) + |Vdd(f)|^2 |NEXTd(f)|^2) df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

$$IFEXT = dB \left(\sqrt{\frac{\int_0^{f_{max}} (|Vin(f)|^2 (|FEXT(f)|^2 + 0.125^2 \cdot |C2D(f)|^2) + |Vdd(f)|^2 |FEXTd(f)|^2) df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

where $NEXT(f)$, $FEXT(f)$, and $C2D(f)$ are the measured near-end and far-end crosstalk between TX/RX pairs, and the common-mode-to-differential conversion, respectively. The factor of 0.1252 accounts for the assumption that the common mode amplitude is 12.5% of the differential amplitude. $NEXTd(f)$ and $FEXTd(f)$ are, respectively, the near-end and far-end crosstalk from the D+/D- pair to TX/RX pairs. $Vdd(f)$ is the input pulse spectrum with $T_b=2.08$ ns.

The largest values of INEXT and IFEXT **shall** meet the following requirements:

- $INEXT \leq -40$ dB to 12.5GHz, for TX1 to RX1, TX2 to RX2, TX1 to RX2, TX2 to RX1, TX1 to TX2, and RX1 to RX2,
- $IFEXT \leq -40$ dB to 12.5GHz, for TX1 to RX1, TX2 to RX2, TX1 to RX2, TX2 to RX1, TX1 to TX2, and RX1 to RX2.

The port-to-port crosstalk (TX1 to RX2, TX2 to RX1, TX1 to TX2, and RX1 to RX2) is specified to support the usages in which all the four high speed pairs transmit or receive signals simultaneously (e.g., USB dual-lane operation).

3.7.2.4.4 Integrated Crosstalk from TX/RX Pairs to USB D+/D- (Normative – USB 3.2 Gen1 and USB4 Gen2)

Crosstalk from the TX/RX pairs to **USB 2.0** D+/D- **shall** be controlled to ensure the robustness of the **USB 2.0** link. Since USB Type-C to Type-C Full-Featured cable assemblies **may** support the usage of **USB 3.2**, **USB4** or an **Alternate Mode** (e.g., **DisplayPort**), the crosstalk from the four high speed differential pairs to D+/D- **may** be from near-end crosstalk, far-end crosstalk, or a combination of the two. The integrated crosstalk to D+/D- is calculated with the following equations:

$$IDDXT_1NEXT + FEXT = dB \left(\sqrt{\frac{\int_0^{f_{max}} |Vin(f)|^2 (|NEXT1(f)|^2 + |FEXT(f)|^2) df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

where:

$NEXT$ = Near-end crosstalk from TX pair to D+/D-,

$FEXT$ = Far-end crosstalk from RX pair to D+/D-, and

$f_{max} = 1.2$ GHz.

$$IDDXT_{2NEXT} = dB \left(\sqrt{\frac{\int_0^{f_{max}} |Vin(f)|^2 (|NEXT1(f)|^2 + |NEXT2(f)|^2) df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

where:

$NEXT1$ = Near-end crosstalk from TX pair to D+/D-,

$NEXT2$ = Near-end crosstalk from RX (the RX functioning in TX mode) pair to D+/D-, and

$f_{max} = 1.2$ GHz.

The integration **shall** be done for $NEXT + FEXT$ and $2NEXT$ on D+/D- from the two differential pairs located at A2, A3, B10 and B11 (see Figure 2-2) and for $NEXT + FEXT$ and $2NEXT$ on D+/D- from the two differential pairs located at B2, B3 A10 and A11 (see Figure 2-2). Measurements are made in two sets to minimize the number of ports required for each measurement.

The integrated differential crosstalk on D+/D- **shall** meet the following requirements:

- $IDDXT_1NEXT + FEXT \leq -34.5$ dB,

- $IDDXT_2NEXT \leq -33$ dB.

3.7.2.4.5 Integrated Return Loss (*Normative – USB 3.2 Gen1 and USB4 Gen2*)

The integrated return loss (IRL) manages the reflection between the cable assembly and the rest of the system (host and device). It is defined as:

$$IRL = dB \left(\sqrt{\frac{\int_0^{f_{max}} |Vin(f)|^2 |SDD21(f)|^2 (|SDD11(f)|^2 + |SDD22(f)|^2) df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

where $SDD21(f)$ is the measured cable assembly differential insertion loss, $SDD11(f)$ and $SDD22(f)$ are the measured cable assembly return losses on the left and right sides, respectively, of a differential pair.

For **USB 3.2** Gen 1 and **USB4** Gen 2 cable assemblies, IRL limit is specified as:

$$IRL \leq 0.046 \cdot IL_{fitatNq}^2 + 1.812 \cdot IL_{fitatNq} - 9.784$$

3.7.2.4.6 Differential-to-Common-Mode Conversion (*Normative – USB 3.2 Gen1 and USB4 Gen2*)

The differential-to-common-mode conversion is specified to control the injection of common mode noise from the cable assembly into the host or device. A mated cable assembly passes if its SCD12/SCD21 is less than or equal to -17 dB from 100 MHz to 10 GHz.

3.7.2.5 Normative TX/RX Passive Cable Assembly Requirements for **USB4** Gen3/Gen4

Note: While the *normative* requirements in this section are defined in terms of **USB4** Gen3, these requirements apply also to **USB4** Gen4 passive cable assemblies as these are electrically equivalent to **USB4** Gen3 passive cable assemblies.

The integrated S-parameter requirements for **USB4** Gen3 follow the same methodology as defined in Section 3.7.2.3. There are parameter adjustments made to suit the **USB4** Gen3 data rate. Unless otherwise specified, the following parameters *shall* be used to calculate insertion loss fit and integrated parameters:

- T_b , the unit interval, is set to 50 ps, reflecting the **USB4** Gen3 data rate.
- T_r , the rise time, remains at $0.4 \cdot T_b$.
- f_{max} , the maximum frequency over which the integration or fitting is performed is increased to 20 GHz.
- An f-square term is added to the insertion loss fit equation to improve fitting quality:

$$IL_{fit} = a + b \cdot \sqrt{f} + c \cdot \sqrt{f^2} + d \cdot \sqrt{f^3} + e \cdot \sqrt{f^4}$$

USB4 Gen3 introduces a system-level COM (Channel Operating Margin) specification for the cable assembly. The details are defined in Section 3.7.2.5.7.

3.7.2.5.1 Insertion Loss Fit at Nyquist Frequencies (*Normative – USB4 Gen3/Gen4*)

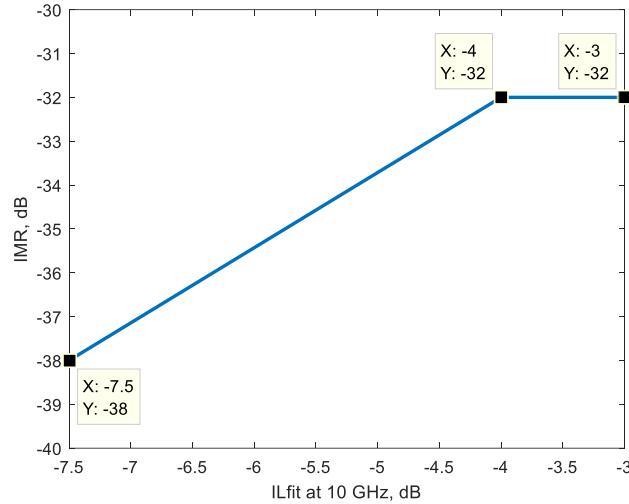
The insertion loss fit at Nyquist frequency ($IL_{fitatNq}$) *shall* meet the following requirements:

- ≥ -1 dB at 100 MHz,
- ≥ -4.2 dB at 2.5 GHz,
- ≥ -6 dB at 5 GHz,
- ≥ -7.5 dB at 10 GHz,
- ≥ -9.3 dB at 12.5 GHz,
- ≥ -9.5 dB at 12.8 GHz, and
- ≥ -11 dB at 15 GHz.

3.7.2.5.2 Integrated Multi-Reflection (*Informative* – **USB4** Gen3/Gen4)

The IMR limit is plotted in Figure 3-45.

Figure 3-45 IMR Limit as Function of ILfit at 10 GHz (USB4** Gen3/Gen4)**



3.7.2.5.3 Integrated Crosstalk between TX/RX Pairs (*Normative* – **USB4** Gen3/Gen4)

The integrated crosstalk within a port for TX1 to RX1 and TX2 to RX2 is recommended to meet the following *informative* requirements:

- $INEXT \leq -43$ dB and
- $IFEXT \leq -43$ dB.

The recommended *informative* requirement for the integrated port-to-port crosstalk for TX1 to RX2, TX2 to RX1, TX1 to TX2, and RX1 to RX2) are defined as:

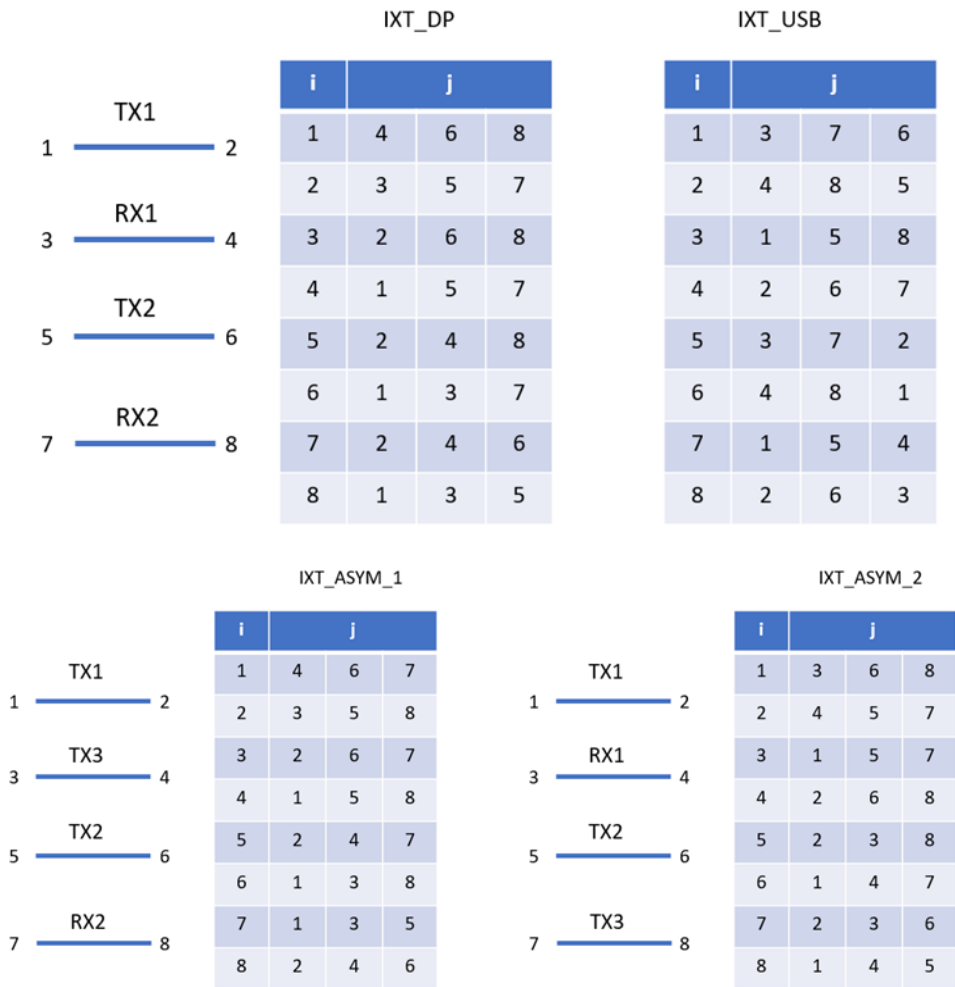
- $INEXT_{p2p} \leq -50$ dB and
- $IFEXT_{p2p} \leq -50$ dB.

The total crosstalk is defined for the **DP Alternate Mode**, **USB4** symmetric, and **USB4** asymmetric operation. In **DP Alternate Mode**, all crosstalk is FEXT, while in **USB4** symmetric and asymmetric operation both FEXT and NEXT exist. The total crosstalk is defined in the equation below:

$$IXTi_{DP} \text{ or } IXTi_{USB} \text{ or } IXTi_{ASYM} = dB \left(\sqrt{\frac{\int_0^{f_{max}} |Vin(f)|^2 \sum_j |SDDij|^2 df}{\int_0^{f_{max}} |Vin(f)|^2 df}} \right)$$

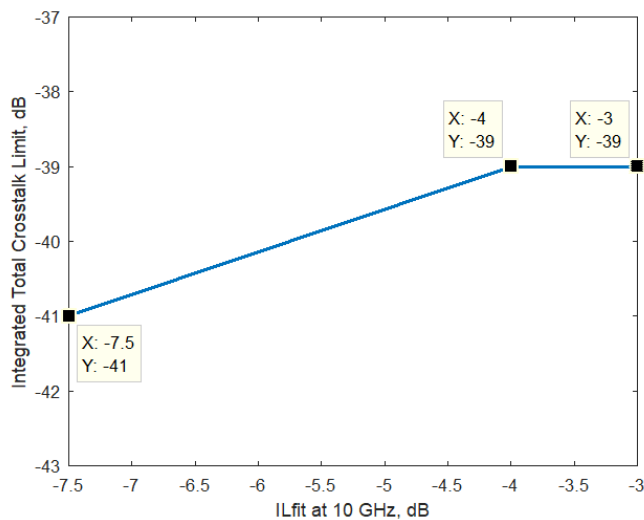
where the victims $i = 1$ to 8 and the aggressors j are defined in Figure 3-46.

Figure 3-46 Definition of Port, Victim, and Aggressor



The total crosstalk for the **DP Alternate Mode**, **USB4** symmetric, and **USB4** asymmetric operation **shall** be controlled. Its **normative** limit is defined in Figure 3-47.

Figure 3-47 IXT_DP and IXT_USB Limit as Function of ILfit at 10 GHz (USB4 Gen3/Gen4)



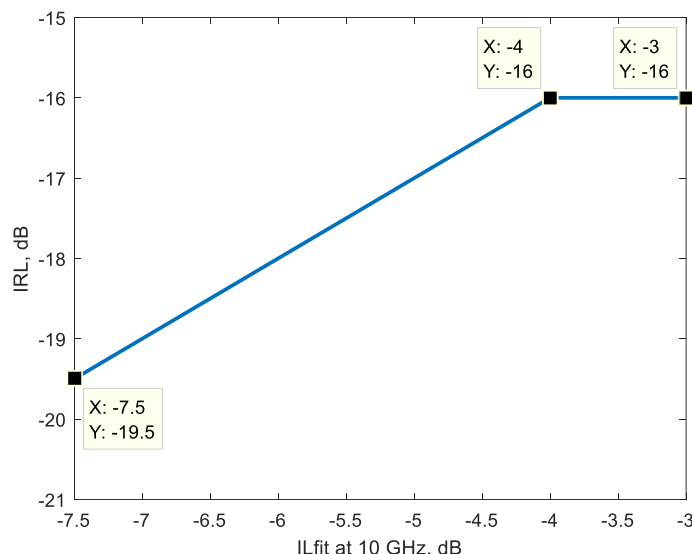
3.7.2.5.4 Integrated Crosstalk from TX/RX Pairs to USB D+/D– (Normative – USB4 Gen3/Gen4)

The requirements for the integrated crosstalk from the TX/RX pairs to **USB 2.0** D+/D– are defined in Section 3.7.2.3.4.

3.7.2.5.5 Integrated Return Loss (Normative – USB4 Gen3/Gen4)

The IRL limit is shown in Figure 3-48.

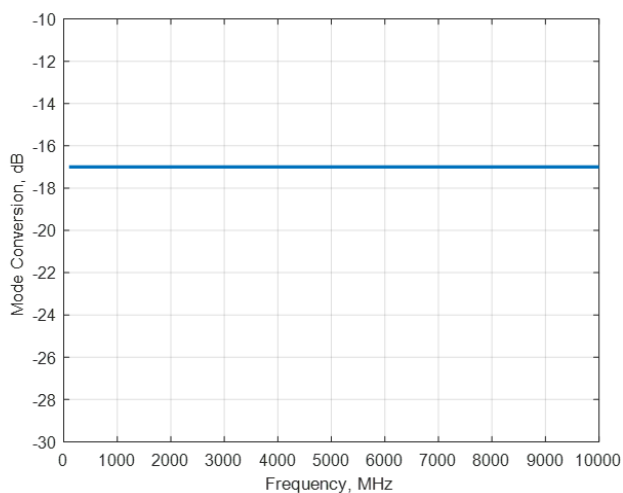
Figure 3-48 IRL Limit as Function of ILfit at 10 GHz (USB4 Gen3/Gen4)



3.7.2.5.6 Differential-to-Common Mode Conversion (Normative – USB4 Gen3/Gen4)

Figure 3-49 illustrates the differential-to-common mode conversion (SCD12/SCD21) requirement. A mated cable assembly passes if its SCD12/SCD21 is less than or equal to -17 dB from 100 MHz to 10 GHz. Note that -17 dB is the worst-case limit; no **USB4** Gen3/Gen4 Type-C cable is allowed to exceed it.

Figure 3-49 Differential-to-Common Mode Conversion Requirement (USB4 Gen3/Gen4)



3.7.2.5.7 COM Requirement (*Normative – USB4 Gen3/Gen4*)

Channel Operating Margin (COM) is a figure of merit to measure the channel electrical quality. The technical detail of COM may be found in IEEE Std 802.3bj™-2014 Clause 93a.

COM is essentially the channel signal-to-noise ratio:

$$COM = 20\log_{10}\left(\frac{A}{N}\right)$$

where A is the signal amplitude and N is the combined noise at BER (bit-error-ratio), which includes the noise sources from ISI (inter-symbol-interference), crosstalk, transmitter jitter, etc.

To calculate COM, reference hosts/devices, which represent the worst-case hosts/devices, **shall** be defined and the reference TX and RX **shall** be used. As illustrated in Figure 3-50, the measured cable assembly S-parameters are cascaded with the reference host and reference device models to form the complete channel; the TX and RX die-loading and equalizers are then applied to the channel to calculate COM.

Figure 3-50 Cable Assembly in System

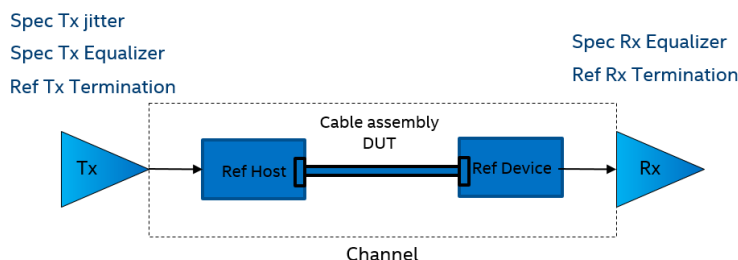


Table 3-23 defines the key parameters in the COM configuration file. It uses the standard COM notations. Note that all the TX and RX equalization settings follow the **USB4** specification.

Table 3-23 Key Parameters in COM Configuration File

Parameter	Setting	Unit	Information
f_b	20	GBd	USB4 Gen 3 data rate
C_d	[0 0]	nF	TX and RX capacitive loading. It is set to zeros as the die-loading is treated as part of the channel
R_d	[42.5 42.5]	Ohm	TX and RX termination resistance
ffe_preset	Table 3-4 of USB4 Specification		TX equalization presets
g_DC	[-9:1:0]	dB	CTLE DC gain
f_p1	5	GHz	CTLE pole 1
f_p2	10	GHz	CTLE pole 2
f_z	3.55	GHz	CTLE zero
A_v	0.4	V	Signal swing
A_fe	0.4	V	FEXT aggressor swing
A_ne	0.6	V	NEXT aggressor swing
N_b	1		Number of DFE tap
b_max(1)	0.7		DFE bound, ratio to cursor
Sigma_RJ	0.01	UI	TX random jitter, rms.
A_DD	0.085	UI	TX deterministic jitter, mean-to-peak
DER_0	1e-12		Target raw bit-error-rate
eta_0	3.3e-8	V ² /GHz	One sided noise spectral density
SNR_TX	40	dB	TX signal to noise ratio
COM Threshold	3	dB	Pass/fail criterion

To support the calculation of the cable assembly COM, the following collaterals is provided and may be obtained from **USB-IF** website:

- Reference host/device S-parameter models,
- Reference TX and RX die-loading S-parameter models,
- COM configuration file, and
- Tool to compute COM.

3.7.2.6 Low-Speed Signal Requirements (*Normative*)

This section specifies the electrical requirements for CC and SBU wires and the coupling among CC, USB D+/D-, VBUS and SBU.

The CC and SBU wires **may** be unshielded or shielded and **shall** have the properties specified in Table 3-24.

Table 3-24 Electrical Requirements for CC and SBU Wires

Name	Description	Min	Max	Units
zCable_CC	Cable characteristic impedance on the CC wire	32	93	Ω
rCable_CC	Cable DC resistance on the CC wire		15	Ω
tCableDelay_CC	Cable propagation delay on the CC wire		26	ns
cCablePlug_CC	Capacitance for each cable plug on the CC wire		25	pF
zCable_SBU	Cable characteristic impedance on the SBU wires	32	53	Ω
tCableDelay_SBU	Cable propagation delay on the SBU wires		26	ns
rCable_SBU	DC resistance of SBU wires in the cable		5	Ω
SBU SE Insertion Loss	Cable SBU single-ended insertion loss		2.0 @ 0.5 MHz 4.0 @ 1 MHz 9.0 @ 10 MHz 10.7 @ 25 MHz 11.9 @ 50 MHz 13.0 @ 100 MHz	dB

Coupling or crosstalk, both near-end and far-end, among the low speed signals **shall** be controlled. Table 3-25 shows the matrix of couplings specified.

Table 3-25 Coupling Matrix for Low Speed Signals

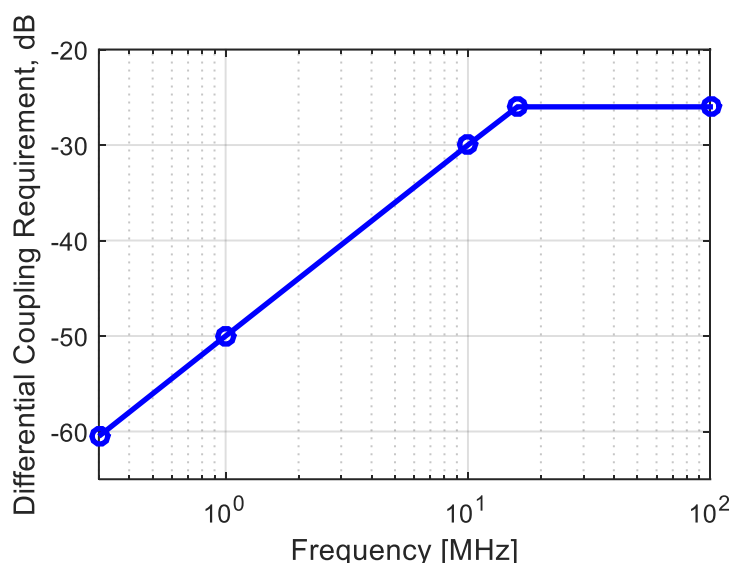
Coupling Matrix	D- (SE)	D+/D- (DF)	V _{BUS}	SBU_B/SBU_A (SE)
CC	FF, CT	FF, CT	FF, CT, CTVPD	FF
D+/D- (DF)	N/A	N/A	FF, CT	FF
SBU_A/SBU_B	N/A	FF	FF	FF

DF: Differential; FF: Full-featured cable; CT: Charge-through cable (including **USB 2.0** function); **CTVPD**: Charge-Through **V_{CONN}**-Powered **USB Device**.

3.7.2.6.1 Coupling between CC to USB D+/D- (Normative)

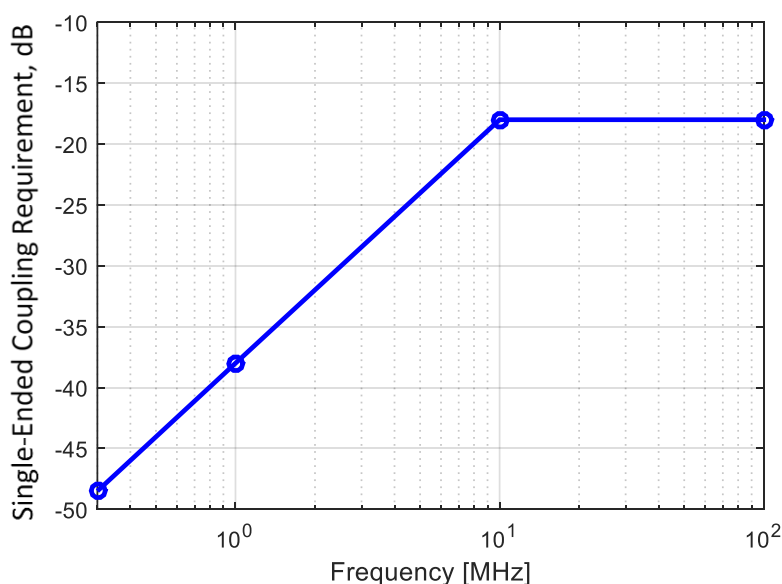
The differential coupling between the CC and D+/D- **shall** be below the limit shown in Figure 3-51. The limit is defined with the vertices of (0.3 MHz, -60.5 dB), (1 MHz, -50 dB), (10 MHz, -30 dB), (16 MHz, -26 dB) and (100 MHz, -26 dB).

Figure 3-51 Requirement for Differential Coupling between CC and D+/D-



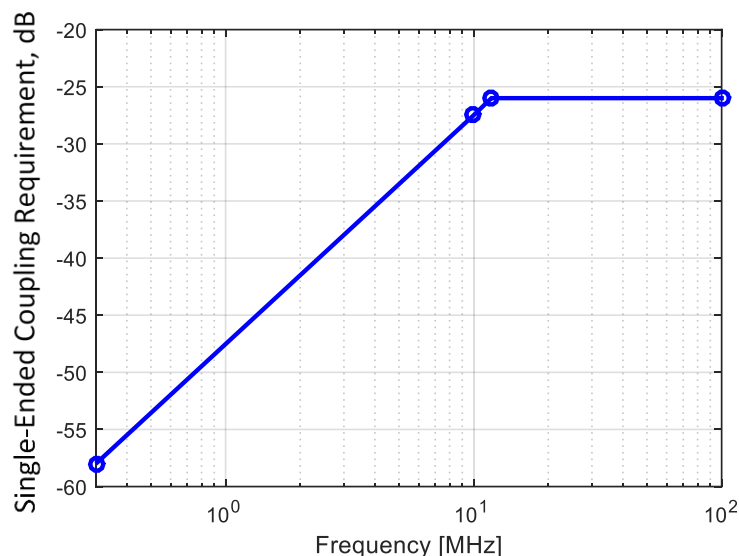
For **USB 2.0** Type-C cables, the singled-ended coupling between the CC and D- **shall** be below the limit shown in Figure 3-52. The limit is defined with the vertices of (0.3 MHz, -48.5 dB), (1 MHz, -38 dB), (10 MHz, -18 dB) and (100 MHz, -18 dB).

Figure 3-52 Requirement for Single-Ended Coupling between CC and D- in USB 2.0 Type-C Cables



For USB Full-Featured Type-C cables, the singled-ended coupling between the CC and D- **shall** be below the limit shown in Figure 3-53. The limit is defined with the vertices of (0.3 MHz, -58 dB), (10 MHz, -27.5 dB), (11.8 MHz, -26 dB) and (100 MHz, -26 dB).

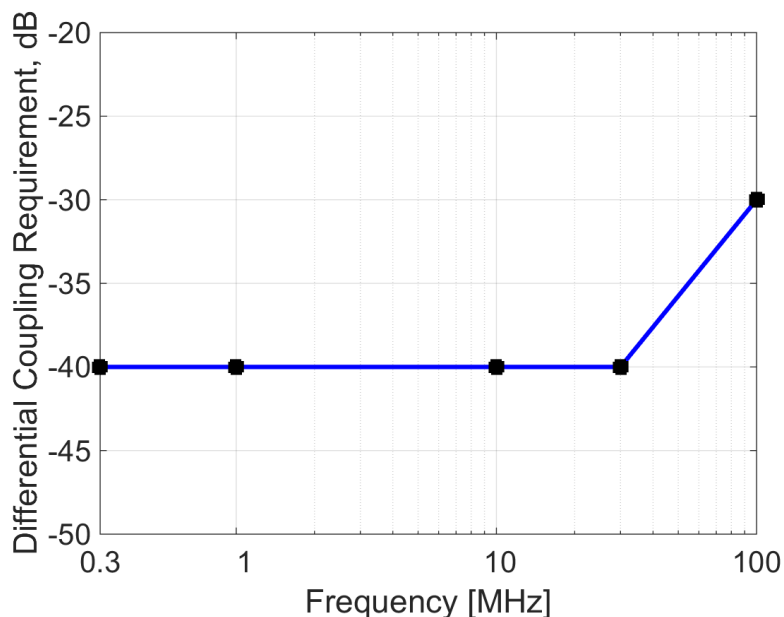
Figure 3-53 Requirement for Single-Ended Coupling between CC and D- in USB Full-Featured Type-C Cables



3.7.2.6.2 VBUS Coupling to SBU_A/SBU_B, CC, and USB D+/D- (Normative)

The differential coupling between VBUS and USB D+/D- **shall** be less than the limit shown in Figure 3-54. The limit is defined by the following vertices: (0.3 MHz, -40 dB), (1 MHz, -40 dB), (30 MHz, -40 dB), and (100 MHz, -30 dB).

Figure 3-54 Requirement for Differential Coupling between VBUS and D+/D-



The maximum VBUS loop inductance **shall** be 900 nH and the maximum mutual inductance (M) between VBUS and low speed signal lines (CC, SBU_A, SBU_B, D+, D-) **shall** be as specified in Table 3-26 to limit VBUS inductive noise coupling on low speed signal lines. For full-featured cables, the range of VBUS bypass capacitance **shall** be 8nF up to 500nF as any of the values in the range is equally effective for high-speed return-path bypassing.

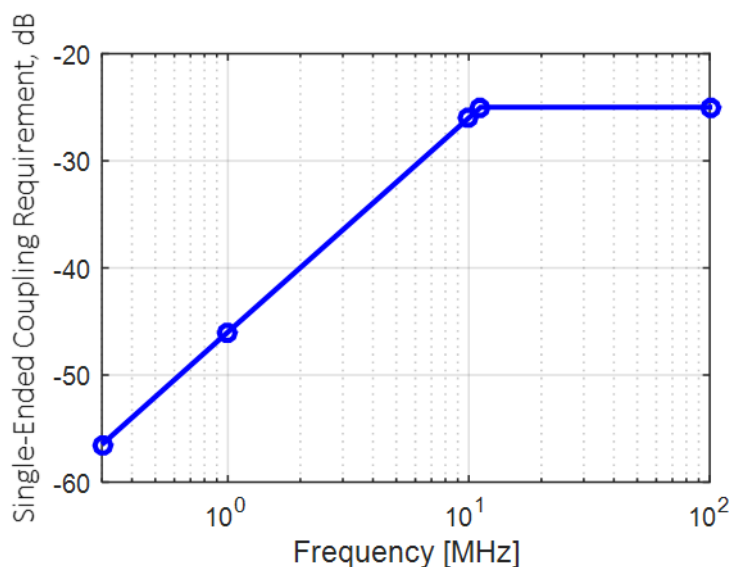
Table 3-26 Maximum Mutual Inductance (M) between VBUS and Low Speed Signal Lines

Low Speed Wire	Maximum Mutual Inductance (nH)
CC	350
SBU_A/SBU_B	330
D+/D-	330

3.7.2.6.3 Coupling between SBU_A and SBU_B (Normative)

The single-ended coupling between SBU_A and SBU_B *shall* be less than the limit shown in Figure 3-55. The limit is defined with the vertices of (0.3 MHz, -56.5 dB), (1 MHz, -46 dB), (10 MHz, -26 dB), (11.2 MHz, -25 dB), and (100 MHz, -25 dB).

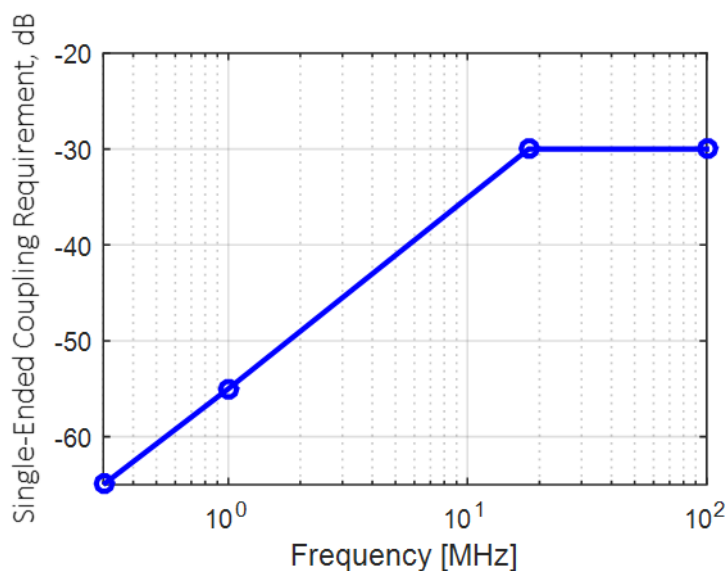
Figure 3-55 Requirement for Single-Ended Coupling between SBU_A and SBU_B



3.7.2.6.4 Coupling between SBU_A/SBU_B and CC (Normative)

The single-ended coupling between SBU_A and CC, and between SBU_B and CC *shall* be less than the limit shown in Figure 3-56. The limit is defined with the vertices of (0.3 MHz, -65 dB), (1 MHz, -55 dB), (18 MHz, -30 dB), and (100 MHz, -30 dB).

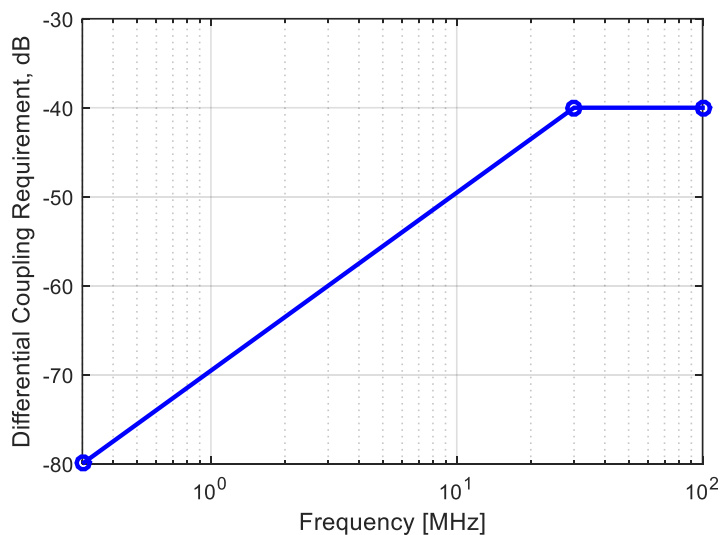
Figure 3-56 Requirement for Single-Ended Coupling between SBU_A/SBU_B and CC



3.7.2.6.5 Coupling between SBU_A/SBU_B and USB D+/D- (Normative)

The coupling between SBU_A and differential D+/D-, and between SBU_B and differential D+/D- **shall** be less than the limit shown in Figure 3-57. The limit is defined with the vertices of (0.3 MHz, -80 dB), (30 MHz, -40 dB), and (100 MHz, -40 dB).

Figure 3-57 Requirement for Coupling between SBU_A and differential D+/D-, and SBU_B and differential D+/D-



3.7.2.7 USB D+/D- Signal Requirements (Normative)

The USB D+/D- lines of the USB Type-C to USB Type-C passive cable assembly **shall** meet the requirements defined in Table 3-27.

**Table 3-27 USB D+/D- Signal Integrity Requirements for
USB Type-C to USB Type-C Passive Cable Assemblies**

Items	Descriptions and Procedures	Requirements
Differential Impedance	EIA 364-108 This test ensures that the D+/D- lines of the cable assembly have the proper impedance. For the entire cable assembly.	75 ohms min and 105 ohms max. 400 ps rise time (20%-80%).
Propagation Delay	EIA 364-103 The purpose of the test is to verify the end-to-end propagation of the D+/D- lines of the cable assembly.	26 ns max. 400 ps rise time (20%-80%).
Intra-pair Skew	EIA 364 - 103 This test ensures that the signal on both the D+ and D- lines of cable assembly arrive at the receiver at the same time.	100 ps max. 400 ps rise time (20%-80%).
D+/D- Pair Attenuation	EIA 364 - 101 This test ensures the D+/D- pair of a cable assembly is able to provide adequate signal strength to the receiver in order to maintain a low error rate.	≥ -1.52 dB @ 50 MHz ≥ -2.03 dB @ 100 MHz ≥ -2.91 dB @ 200 MHz ≥ -4.35 dB @ 400 MHz
D+ or D- DC Resistance	This test ensures the D+/D- has the proper DC resistance range in order to predict the EOP level and set the USB 2.0 disconnect level.	3.5 ohms max.

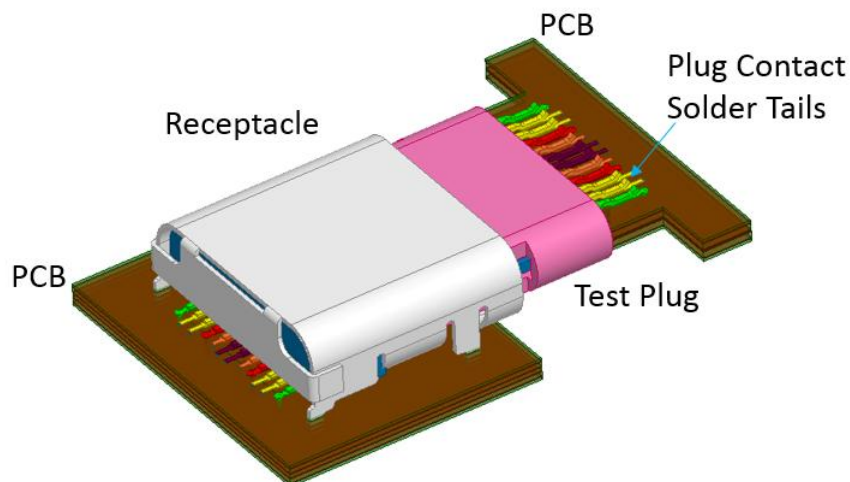
3.7.2.8 VBUS DC Voltage Tolerance (*Normative*)

A USB Type-C to USB Type-C cable assembly **shall** tolerate a VBUS voltage of 21 V DC at the cable rated current (i.e., 3 A or 5 A) applied for one hour as a pre-condition of the testing of the electrical aspects of the cable assembly.

3.7.3 Mated Connector (*Informative – USB 3.2 Gen2 and USB4 Gen2*)

The mated connector as defined in this specification for USB Type-C consists of a receptacle mounted on a PCB, representing how the receptacle is used in a product, and a test plug also mounted on a PCB (paddle card) without cable. This is illustrated in Figure 3-58. Note that the test plug is used in host/device TX/RX testing also.

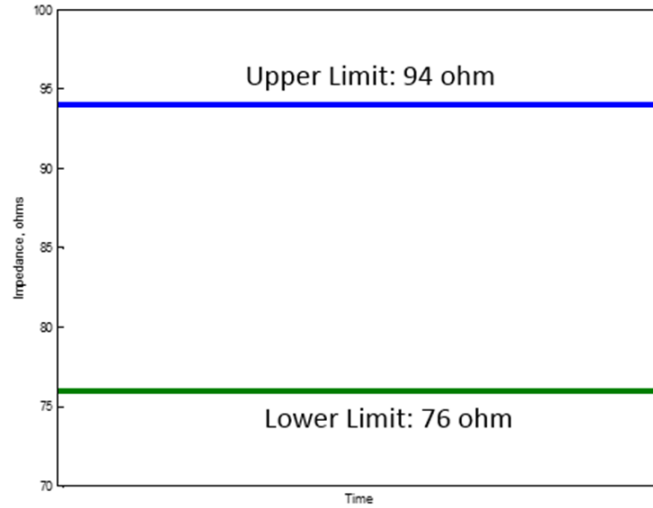
Figure 3-58 Illustration of USB Type-C Mated Connector



3.7.3.1 Differential Impedance (*Informative*)

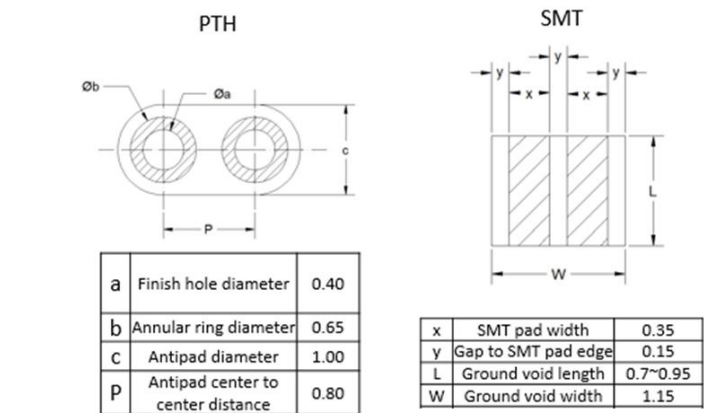
The mated connector impedance target is specified to minimize reflection from the connector. The differential impedance of a mated connector **should** be within $85\ \Omega \pm 9\ \Omega$, as seen from a 40 ps (20% – 80%) rise time. The impedance profile of a mated connector **should** fall within the limits shown in Figure 3-59.

Figure 3-59 Recommended Impedance Limits of a USB Type-C Mated Connector

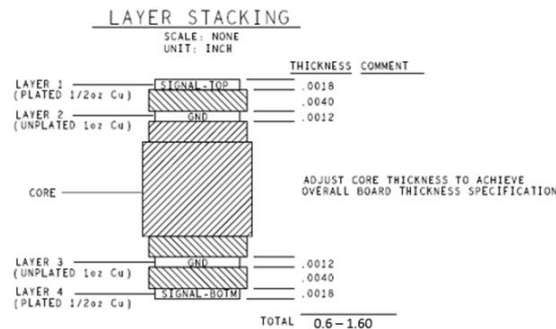


The PCB stack up, lead geometry, and solder pad geometry **should** be modeled in 3D field-solver to optimize electrical performance. Example ground voids under signal pads are shown in Figure 3-60 based on pad geometry, mounting type, and PCB stack-up shown.

Figure 3-60 Recommended Ground Void Dimensions for USB Type-C Receptacle



Unit: mm



3.7.3.2 Mated Connector Recommended Differential S-Parameter and Signal Integrity Characteristics

The recommended signal integrity characteristics of USB Type-C mated connector pair are listed in Table 3-28.

Table 3-28 USB Type-C Mated Connector Recommended Signal Integrity Characteristics (Informative)

Items	Descriptions and Procedures	Requirements
Differential Insertion Loss Fit at Nyquist Frequencies (ILfitatNq)	ILfitatNq is evaluated at the TX/RX Gen1, Gen2 and Gen3 generation Nyquist frequencies.	≥ -0.6 dB @ 2.5 GHz ≥ -0.8 dB @ 5.0 GHz ≥ -1.0 dB @ 10 GHz
Integrated Differential Multi-reflection (IMR)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} ILD(f) ^2 Vin(f) ^2 df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$	≤ -40 dB ≤ -39 dB (with $f_{max} = 20$ GHz and $Vin(f)$ defined with $Tb(UI) = 50$ ps; USB4 Gen3)
Integrated Differential Near-end Crosstalk on TX/RX (INEXT)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} Vin(f) ^2 (NEXT(f) ^2 + 0.125^2 \cdot C2D(f) ^2) df + \int_0^{f_{max}} Vdd(f) ^2 NEXTd(f) ^2 df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$ <p>where: $NEXT$ = NEXT between TX/RX pairs $NEXTd$ = NEXT between D+/D- and TX/RX pairs</p>	≤ -44 dB
Integrated Differential Far-end Crosstalk on TX/RX (IFEXT)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} Vin(f) ^2 (FEXT(f) ^2 + 0.125^2 \cdot C2D(f) ^2) df + \int_0^{f_{max}} Vdd(f) ^2 FEXTd(f) ^2 df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$ <p>where: $FEXT$ = FEXT between TX/RX pairs $FEXTd$ = FEXT between D+/D- and TX/RX pairs</p>	≤ -44 dB
Differential Crosstalk of TX/RX on D+/D-	The differential near-end and far-end crosstalk of the TX/RX pairs on the D+/D- pair in mated connectors.	See Figure 3-61
Differential Crosstalk of D+/D- on TX/RX	The differential near-end and far-end crosstalk of the D+/D- pair on the TX/RX pairs in mated connectors.	See Figure 3-61
Integrated Return Loss (IRL)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} Vin(f) ^2 (SDD21(f) ^2 + SDD11(f) ^2 + SDD22(f) ^2) df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$	≤ -18 dB

Differential to Common Mode Conversion (SCD12 and SCD21)	The differential to common mode conversion is specified to control the injection of common mode noise from the cable assembly into the host or device. Frequency range: 100 MHz ~ 10.0 GHz	See Figure 3-62
<p>Note: f_{max} = 12.5 GHz (unless otherwise specified); $V_{in}(f)$ is defined in Figure 3-41 with T_b (UI) = 100 ps (unless otherwise specified); $V_{dd}(f)$ is also defined in Figure 3-41 with T_b (UI) = 2.08 ns. $C2D(f)$ = measured near-end and far-end crosstalk between TX/RX pairs, and the common-mode-to-differential conversion, respectively. The factor of 0.125^2 accounts for the assumption that the common mode amplitude is 12.5% of the differential amplitude.</p>		

Figure 3-61 Recommended Differential Near-End and Far-End Crosstalk Limits between D+/D- Pair and TX/RX Pairs

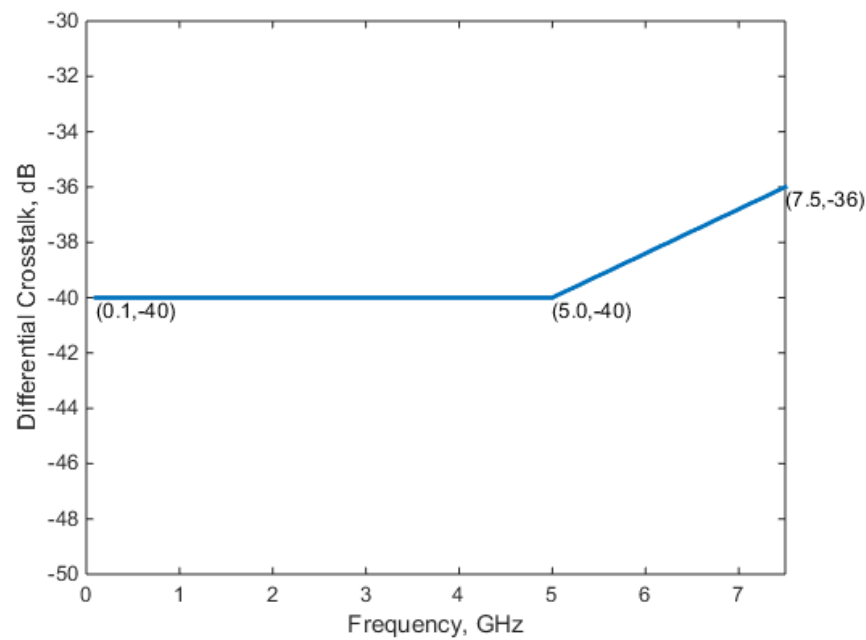
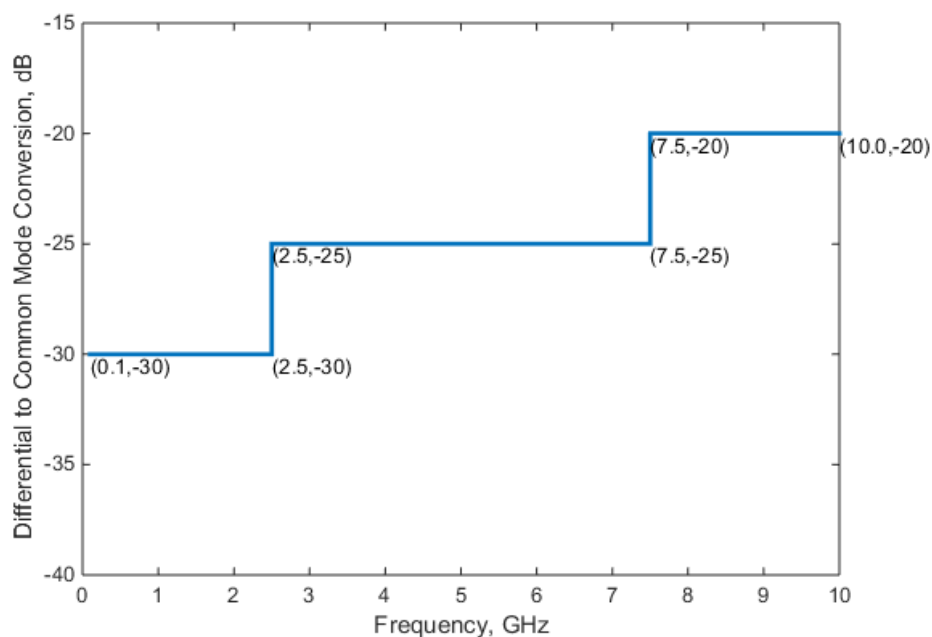


Figure 3-62 Recommended Limits for Differential-to-Common-Mode Conversion



3.7.4 Receptacle Connector SI Requirements and Testing (*Normative – USB4 Gen3/Gen4*)

Note: While the **normative** requirements in this section are defined in terms of **USB4** Gen3, these requirements apply also to **USB4** Gen4 passive cable assemblies as these are electrically equivalent to **USB4** Gen3 passive cable assemblies.

The USB Type-C receptacle connector requirements for **USB4** Gen3 are **normative** and listed in Table 3-29. Unless otherwise specified, the items to be specified are identical to what is defined in Section 3.7.3 and the parameters used to calculate the integrated parameters are the same as defined in Section 3.7.2.4 for the **USB4** Gen3 cable assembly.

**Table 3-29 USB Type-C Receptacle Connector Signal Integrity Characteristics for USB4 Gen3
(Normative)**

Items	Descriptions and Procedures	Requirements
Differential Insertion Loss Fit at Nyquist Frequencies (ILfitatNq)	ILfitatNq is evaluated at a few different frequencies.	≥ -0.6 dB @ 2.5 GHz ≥ -0.8 dB @ 5.0 GHz ≥ -1.0 dB @ 10 GHz ≥ -1.25 dB @ 12.5 GHz ≥ -1.5 dB @ 15 GHz
Integrated Differential Near-end Crosstalk on TX/RX (INEXT)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} Vin(f) ^2 (NEXT(f) ^2 df)}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$ <p>where: <i>NEXT</i> = NEXT between TX1 and RX1, and TX2 and RX2.</p>	≤ -43 dB
Integrated Differential Far-end Crosstalk on TX/RX (IFEXT)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} Vin(f) ^2 (FEXT(f) ^2 df)}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$ <p>where: <i>FEXT</i> = FEXT between TX1 and RX1, and TX2 and RX2.</p>	≤ -43 dB
Differential Crosstalk of TX/RX on D+/D-	The differential near-end and far-end crosstalk of the TX/RX pairs on the D+/D- pair in mated connectors, and the differential near-end and far-end crosstalk of the D+/D- pair on the TX/RX pairs in mated connectors.	≤ -50 dB
Differential Crosstalk of D+/D- on TX/RX	$dB \left(\sqrt{\frac{\int_0^{f_{max}} Vin(f) ^2 (FEXT(f) ^2 + NEXT(f) ^2) df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$ <p>where: <i>FEXT</i> = Far-end crosstalk between TX/RX and D+/D- pairs <i>NEXT</i> = Near-end crosstalk between TX/RX and D+/D- pairs <i>fmax</i> = 1.2 GHz</p>	
Integrated Return Loss (IRL)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} Vin(f) ^2 (SDD21(f) ^2 (SDD11(f) ^2 + SDD22(f) ^2) df)}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$	≤ -15 dB
Differential to Common Mode Conversion (SCD12 and SCD21)	<p>The differential to common mode conversion is specified to control the injection of common mode noise from the cable assembly into the host or device.</p> <p>Frequency range: 100 MHz ~ 10.0 GHz</p>	≤ -20 dB
<p>Note: <i>fmax</i> = 20 GHz (unless otherwise specified); <i>Vin(f)</i> is defined with Tb (UI) = 50 ps.</p>		

The requirements defined in this section do not apply to the USB Type-C plug connector, as the limits Table 3-29 factor in the electrical characteristics of test fixture that includes a USB Type-C plug connector specifically selected for testing of the USB Type-C receptacle. The USB Type-C plug connector does not have a set of requirements defined at the mated connector level as there are tradeoffs allowed to achieve acceptable performance at the finished cable assembly level.

3.7.5 USB Type-C to USB Legacy Cable Assemblies (*Normative*)

The USB Type-C to legacy cable assemblies *may* support **USB 2.0** only or **USB 3.1**.

3.7.5.1 **USB 2.0-only Cable Assemblies (*Normative*)**

The **USB 2.0**-only Type-C to legacy USB cable assemblies include:

- USB Type-C plug to **USB 2.0** Standard-A plug
- USB Type-C plug to **USB 2.0** Standard-B plug
- USB Type-C plug to **USB 2.0** Micro-B plug
- USB Type-C plug to **USB 2.0** Mini-B plug

The USB D+/D- signal integrity requirements are specified in Table 3-30.

Table 3-30 USB D+/D- Signal Integrity Requirements for USB Type-C to Legacy USB Cable Assemblies (*Normative*)

Items	Descriptions and Procedures	Requirements
Differential Impedance	EIA 364-108 This test ensures that the D+/D- lines of the cable assembly have the proper impedance. For the entire cable assembly.	75 ohms min and 105 ohms max. 400 ps rise time (20% – 80%).
Propagation Delay	EIA 364-103 The purpose of the test is to verify the end-to-end propagation of the D+/D- lines of the cable assembly.	10 ns max for USB Type-C to Micro-B cable assembly; 20 ns max for all other USB Type-C to legacy USB cable assemblies. 400 ps rise time (20% – 80%).
Intra-pair Skew	EIA 364 – 103 This test ensures that the signal on both the D+ and D- lines of cable assembly arrive at the receiver at the same time.	100 ps max. 400 ps rise time (20% – 80%).
D+/D- Pair Attenuation	EIA 364 – 101 This test ensures the D+/D- pair of a cable assembly is able to provide adequate signal strength to the receiver in order to maintain a low error rate.	≥ –1.52 dB @ 50 MHz ≥ –2.03 dB @ 100 MHz ≥ –2.91 dB @ 200 MHz ≥ –4.35 dB @ 400 MHz
D+ or D- DC Resistance	This test ensures the D+/D- has the proper DC resistance range in order to predict the EOP level and set the USB 2.0 disconnect level.	3.5 ohms max.

3.7.5.2 **USB 3.1 Gen2 Cable Assemblies (*Normative*)**

The USB Type-C to **USB 3.1** Gen2 legacy cable assemblies include:

- USB Type-C plug to **USB 3.1** Standard-A plug
- USB Type-C plug to **USB 3.1** Standard-B plug
- USB Type-C plug to **USB 3.1** Micro-B plug

The *informative* design targets for these cables are provided in Table 3-31.

Table 3-31 Design Targets for USB Type-C to *USB 3.1* Gen2 Legacy Cable Assemblies (*Informative*)

Items	Design Targets
Differential Impedance	76 ohms min and 96 ohms max. 40 ps rise time (20%-80%).
Differential Insertion Loss	≥ −2 dB @ 100 MHz ≥ −4 dB @ 2.5 GHz, except for the USB Type-C plug to USB 3.1 Standard-A plug cable assembly which is ≥ −3.5 dB @ 2.5 GHz −6.0 dB max @ 5.0 GHz
Differential NEXT between SuperSpeed Pairs	≤ −34 dB to 5 GHz
Differential NEXT and FEXT between D+/D- and SuperSpeed Pairs	≤ −30 dB to 5 GHz

The **normative** requirements include the USB D+/D− signaling as specified in Table 3-30, and the SuperSpeed USB parameters specified in Table 3-32.

**Table 3-32 USB Type-C to USB 3.1 Gen2 Legacy Cable Assembly Signal Integrity Requirements
(Normative)**

Items	Descriptions and Procedures	Requirements
Differential Insertion Loss Fit at Nyquist Frequencies (ILfitatNq)	ILfitatNq is evaluated at both the SuperSpeed Gen1 and Gen2 Nyquist frequencies.	≥ -4 dB @ 2.5 GHz, except for the USB Type-C plug to USB 3.1 Standard-A plug cable assembly which is ≥ -3.5 dB @ 2.5 GHz ≥ -6.0 dB at 5.0 GHz
Integrated Differential Multi-reflection (IMR)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} ILD(f) ^2 Vin(f) ^2 df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$	$\leq 0.126 \cdot ILfitatNq^2 + 3.024 \cdot ILfitatNq - 21.392$ See Figure 3-63.
Integrated Differential Crosstalk on SuperSpeed (ISSXT)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} (Vin(f) ^2 NEXTs(f) ^2 + Vdd(f) ^2 NEXTd(f) ^2) df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$ <p>where: NEXTs = NEXT between SuperSpeed pairs NEXTd = NEXT between D+/D- and SuperSpeed pairs Vdd(f) = Input pulse spectrum on D+/D- pair, evaluated using equation shown in Figure 3-41 with Tb (UI) = 2.08 ns.</p>	≤ -38 dB
Integrated Differential Crosstalk on D+/D- (IDDXT)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} Vin(f) ^2 (NEXT(f) ^2 + FEXT(f) ^2) df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$ <p>where: NEXT = Near-end crosstalk from SuperSpeed to D+/D- FEXT = Far-end crosstalk from SuperSpeed to D+/D- fmax = 1.2 GHz</p>	≤ -28.5 dB
Integrated Return Loss (IRL)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} Vin(f) ^2 SDD21(f) ^2 (SDD11(f) ^2 + SDD22(f) ^2) df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$	$\leq 0.046 \cdot ILfitatNq^2 + 1.812 \cdot ILfitatNq - 9.784$ See Figure 3-64.
Differential to Common Mode Conversion (SCD12 and SCD21)	<p>The differential to common mode conversion is specified to control the injection of common mode noise from the cable assembly into the host or device.</p> <p>Frequency range: 100 MHz ~ 10.0 GHz</p>	≤ -20 dB

Note: fmax = 10 GHz (unless otherwise specified); Vin(f) is defined in Figure 3-41 with Tb (UI) = 100 ps; and Vdd(f) is also defined in Figure 3-41 with Tb (UI) = 2.08 ns.

Figure 3-63 IMR Limit as Function of ILfitatNq for USB Type-C to Legacy Cable Assembly

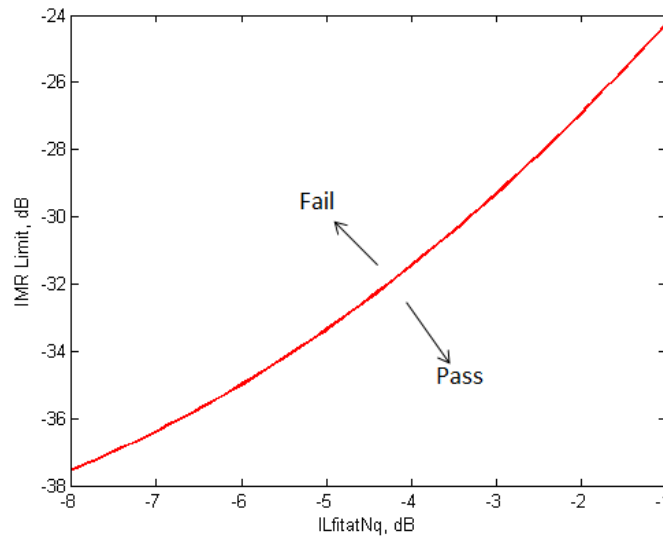
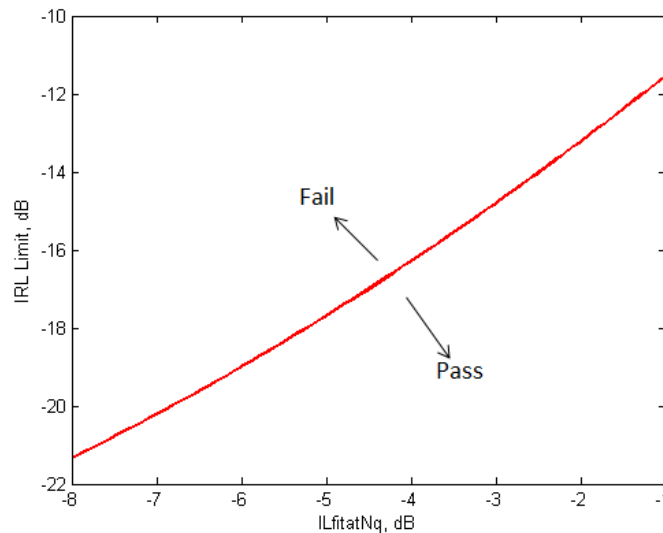


Figure 3-64 IRL Limit as Function of ILfitatNq for USB Type-C to Legacy Cable Assembly



3.7.5.3 Compliant USB Legacy Plugs used in USB Type-C to USB Legacy Cable Assemblies

The following requirements are incremental to the existing requirements for legacy connectors when used in compliant USB Type-C to legacy cable assemblies.

3.7.5.3.1 Contact Material Requirements for USB Type-C to USB Micro-B Assemblies

For USB Type-C to USB Micro-B assemblies, change the contact material in the USB Micro-B connector to achieve the following Low-Level Contact Resistance (EIA 364-23B):

- 20 milliohms (Max) initial for VBUS and GND contacts, and
- Maximum change (delta) of +10 milliohms after environmental stresses.

3.7.5.3.2 Contact Current Ratings for USB Standard-A, USB Standard-B and USB Micro-B Connector Mated Pairs (EIA 364-70, Method 2)

When a current of 3 A is applied to the VBUS pin and its corresponding GND pin (i.e., pins 1 and 4 in a USB Standard-A or USB Standard-B connector or pins 1 and 5 in a USB Micro-B connector), the delta temperature **shall not** exceed +30° C at any point on the connectors under test, when measured at an ambient temperature of 25° C.

3.7.6 USB Type-C to USB Legacy Adapter Assemblies (Normative)

Only the following standard legacy adapter assemblies are defined:

- **USB 2.0** Type-C plug to **USB 2.0** Micro-B receptacle, and
- USB Full-Featured Type-C plug to **USB 3.1** Standard-A receptacle.

3.7.6.1 USB 2.0 Type-C Plug to USB 2.0 Micro-B Receptacle Adapter Assembly

This adapter assembly supports only the **USB 2.0** signaling. It **shall not** exceed 150 mm in total length, measured from end to end. Table 3-33 defines the electrical requirements.

Table 3-33 USB D+/D- Signal Integrity Requirements for USB Type-C to Legacy USB Adapter Assemblies (Normative)

Items	Descriptions and Procedures	Requirements
Differential Impedance	EIA 364-108 This test ensures that the D+/D- lines of the adapter assembly have the proper impedance. For the entire adapter assembly.	75 ohms min and 105 ohms max. 400 ps rise time (20% – 80%).
Intra-pair Skew	EIA 364 – 103 This test ensures that the signal on both the D+ and D- lines of adapter assembly arrive at the receiver at the same time.	20 ps max. 400 ps rise time (20% – 80%).
Differential Insertion Loss	EIA 364 – 101 This test ensures the D+/D- pair of an adapter assembly can provide adequate signal strength to the receiver.	–0.7 dB max @ 400 MHz
D+ or D- DC Resistance	This test ensures the D+/D- has the proper DC resistance range in order to predict the EOP level and set the USB 2.0 disconnect level.	2.5 ohms max.

3.7.6.2 USB Full-Featured Type-C Plug to USB 3.1 Standard-A Receptacle Adapter Assembly

The USB Full-Featured Type-C plug to **USB 3.1** Standard-A receptacle adapter assembly is intended to be used with a direct-attach device (e.g., USB thumb drive). A system is not guaranteed to function when using an adapter assembly together with a Standard USB cable assembly.

To minimize the impact of the adapter assembly to system signal integrity, the adapter assembly **should** meet the **informative** design targets in Table 3-34.

Table 3-34 Design Targets for USB Type-C to USB 3.1 Standard-A Adapter Assemblies (Informative)

Items	Design Targets
Differential Return Loss	≤ -15 dB to 5 GHz Normalized with 85 ohms.
Differential Insertion Loss	≥ -2.4 dB to 2.5 GHz, ≥ -3.5 dB to 5 GHz
Differential NEXT between SuperSpeed Pairs	≤ -40 dB to 2.5 GHz ≤ -34 dB at 5 GHz
Differential NEXT and FEXT between D+/D- and SuperSpeed Pairs	≤ -30 dB to 2.5 GHz

The **normative** requirements for the adapter assembly are defined in Table 3-33 and Table 3-35. The adapter assembly total length is limited to 150 mm max.

Table 3-35 USB Type-C to USB 3.1 Standard-A Receptacle Adapter Assembly Signal Integrity Requirements (Normative)

Items	Descriptions and Procedures	Requirements
Differential Insertion Loss Fit at Nyquist Frequency (ILfitatNq)	ILfitatNq is evaluated at the SuperSpeed Gen1 Nyquist frequency.	≥ -2.4 dB at 2.5 GHz ≥ -3.5 dB at 5 GHz
Integrated Differential Multi-reflection (IMR)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} ILD(f) ^2 Vin(f) ^2 df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$	≤ -38 dB, Tb = 200 ps ≤ -27 dB, Tb = 100 ps
Integrated Differential Crosstalk on SuperSpeed (ISSXT)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} (Vin(f) ^2 NEXTs(f) ^2 + Vdd(f) ^2 NEXTd(f) ^2) df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$ <p>where: NEXTs = NEXT between SuperSpeed pairs NEXTd = NEXT between D+/D- and SuperSpeed pairs Vdd(f) = Input pulse spectrum on D+/D- pair, evaluated using equation shown in Figure 3-41 with Tb (UI) = 2.08 ns.</p>	≤ -37 dB
Integrated Differential Crosstalk on D+/D- (IDDXT)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} Vin(f) ^2 (NEXT(f) ^2 + FEXT(f) ^2) df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$ <p>where: NEXT = Near-end crosstalk from SuperSpeed to D+/D- FEXT = Far-end crosstalk from SuperSpeed to D+/D- fmax = 1.2 GHz</p>	≤ -30 dB
Integrated Return Loss (IRL)	$dB \left(\sqrt{\frac{\int_0^{f_{max}} Vin(f) ^2 SDD21(f) ^2 (SDD11(f) ^2 + SDD22(f) ^2) df}{\int_0^{f_{max}} Vin(f) ^2 df}} \right)$	≤ -14.5 dB, Tb = 200 ps ≤ -12.0 dB, Tb = 100 ps
Diff to Comm mode	Differential to Common Mode conversion (SCD12, SCD21)	≤ -15 dB
Note: fmax = 7.5 GHz; Vin(f) is defined in Figure 3 40 with Tb (UI) = 200 ps; and Vdd(f) is also specified in Figure 3-41 with Tb (UI) = 2.08 ns.		

3.7.6.3 Compliant USB Legacy Receptacles used in USB Type-C to USB Legacy Adapter Assemblies**3.7.6.3.1 Contact Material Requirements**

Refer to Section 3.7.5.3.1 for contact material requirements as these apply to legacy USB Standard-A and USB Micro-B receptacles used in USB Type-C to Legacy Adapter Assemblies.

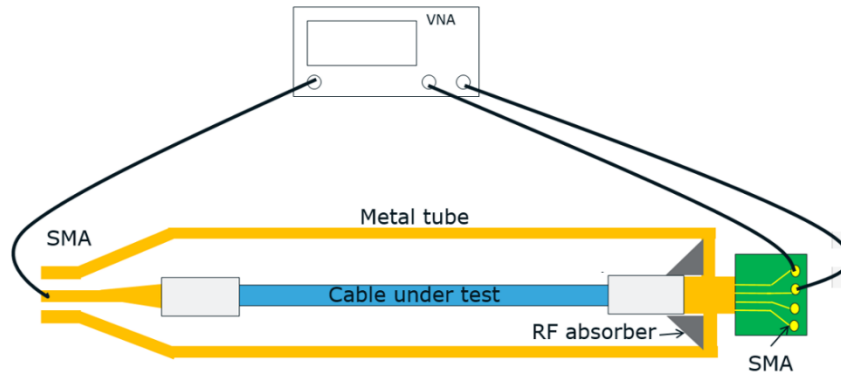
3.7.6.3.2 Contact Current Ratings

Refer to Section 3.7.5.3.2 for contact current rating requirements as these apply to legacy USB Standard-A and USB Micro-B receptacles used in USB Type-C to Legacy Adapter Assemblies.

3.7.7 Shielding Effectiveness Requirements (*Normative*)

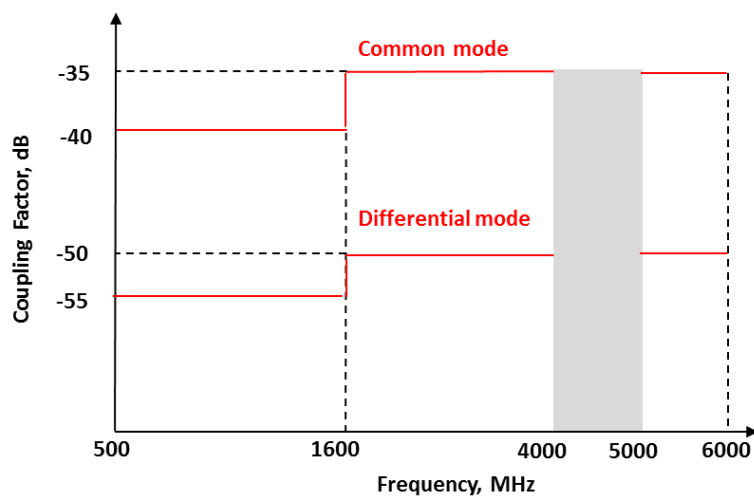
The cable assembly shielding effectiveness (SE) test measures the EMI and RFI levels from the cable assembly. To perform the measurement, the cable assembly **shall** be installed in the cable SE test fixture as shown in Figure 3-65. The coupling factors from the cable to the fixture are characterized with a VNA.

Figure 3-65 Cable Assembly Shielding Effectiveness Testing

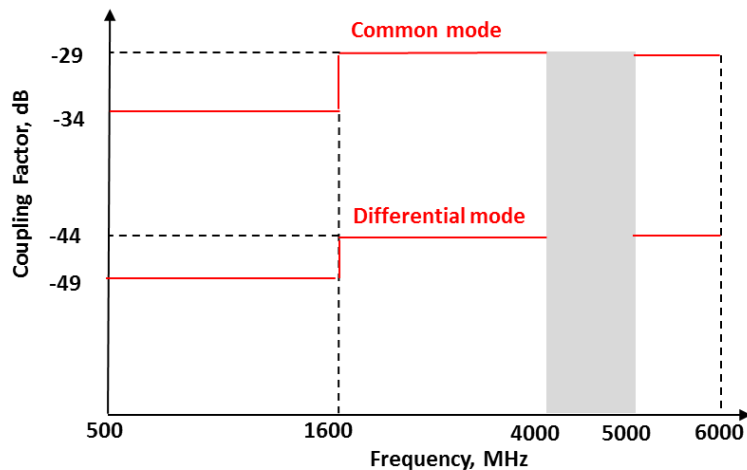


All USB Type-C cable assemblies **shall** pass the shielding effectiveness test for compliance. Figure 3-66 shows the pass/fail criteria for (a) USB Type-C to USB Type-C cable assemblies, (b) USB Type-C to legacy USB cable assemblies, and (c) the USB Type-C to **USB 3.1** Standard-A Receptacle Adapter assembly. Note that the shielding effectiveness for the frequency band from 4 GHz to 5 GHz is not specified since there is no antenna operating in this frequency range.

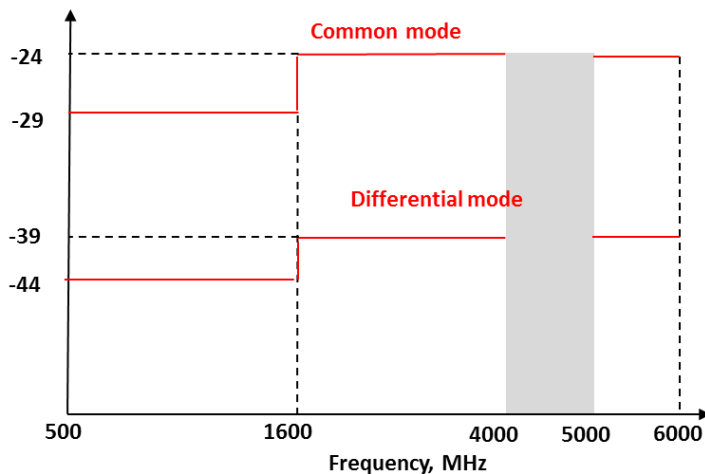
Figure 3-66 Shielding Effectiveness Pass/Fail Criteria



(a) For USB Type-C to USB Type-C Cable Assemblies



(b) For USB Type-C to legacy USB Cable Assemblies



(c) For USB Type-C to USB3.1 Standard-A Receptacle Adapter Assembly

3.7.8 DC Electrical Requirements (*Normative*)

Unless otherwise stated, the tests in this section are performed on mated connector pairs.

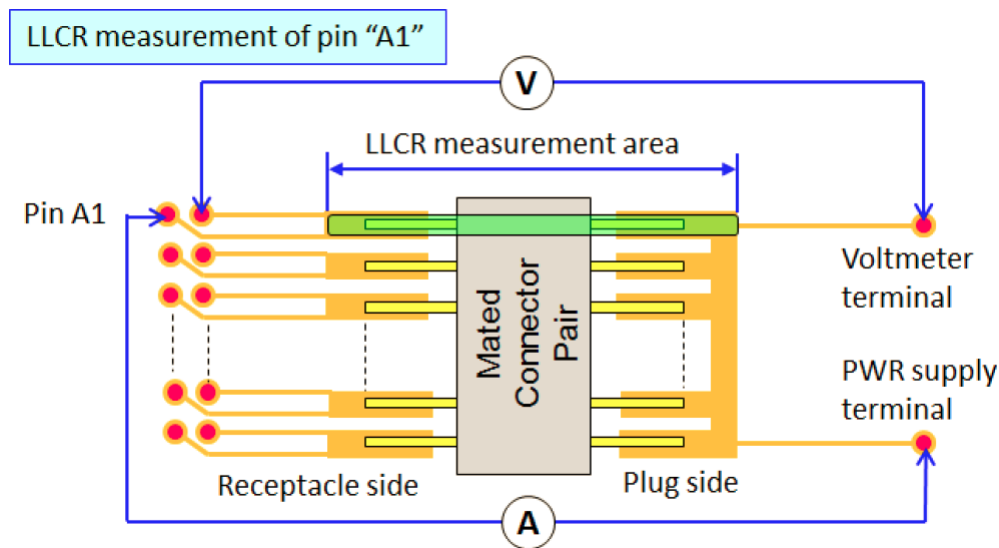
3.7.8.1 Low Level Contact Resistance (EIA 364-23B)

The low-level contact resistance (LLCR) measurement is made across the plug and receptacle mated contacts and does not include any internal paddle cards or substrates of the plug or receptacle. See Figure 3-67. The following apply to the power and signal contacts:

- 40 mΩ (Max) initial for VBUS, GND and all other contacts.
- 50 mΩ (Max) after environmental stresses.
- Measure at 20 mV (Max) open circuit at 100 mA.

Refer to Section 3.8 for environmental requirements and test sequences.

Figure 3-67 LLCR Measurement Diagram



3.7.8.2 Dielectric Strength (EIA 364-20)

No breakdown **shall** occur when 100 Volts AC (RMS) is applied between adjacent contacts of unmated and mated connectors.

3.7.8.3 Insulation Resistance (EIA 364-21)

A minimum of 100 MΩ insulation resistance is required between adjacent contacts of unmated and mated connectors.

3.7.8.4 Contact Current Rating

The contact current rating for the USB Type-C connector (plug and receptacle) contacts used for VBUS (A4, A9, B4, and B9) **shall** be a minimum of 1.25 A per contact. The contact current rating for VCONN (A5 and B5) **shall** be a minimum of 0.5 A per contact. The contact current rating for GND (A1, A12, B1, and B12) contacts **shall** be a minimum of 1.375 A per contact. The contact current rating for the remaining contacts **shall** be a minimum of 0.25 A per contact.

To assure the safe interoperability of USB Type-C connectors from different manufacturers used in Sources, Sinks, and cables; USB Type-C connectors **shall** be only used for supplying up to 5 A on VBUS as defined by the power rules of **USB PD**.

The current rating testing for the USB Type-C connector (plug and receptacle) **shall** be conducted per the following set up and procedures:

- A current of 5 A **shall** be applied collectively to VBUS pins (i.e., pins A4, A9, B4, and B9) and 0.5 A **shall** be applied to the VCONN pin (i.e., B5) as applicable, terminated through the corresponding GND pins (i.e., pins A1, A12, B1, and B12). A minimum current of 0.25 A **shall** also be applied individually to all the other contacts, as applicable. When current is applied to the contacts, the temperature of the connector pair **shall** be allowed to stabilize. The temperature rise of the outside shell surface of the mated pair above the VBUS and GND contacts **shall not** exceed 30 °C above the ambient temperature. Figure 3-68 provides an illustration of the measurement location.
- The measurement **shall** be done in still air.
- The connectors **shall** be oriented such that the accessible outer shell surface is on top and horizontal to the ground.
- The plug and receptacle **may** require modification to access solder tails or cable attachment points.
- Either thermocouple or thermo-imaging (preferred) method **may** be used for temperature measurement
- For certification, the connector manufacturer **shall** provide the receptacle and plug samples under test mounted on a current rating test PCB with no copper planes. A cable plug **may** use short wires to attach the cable attachment points together rather than using a current rating test PCB.

The current rating test PCBs **shall** be of a 2-layer construction. If 2-layer construction is not possible due to the solder tail configuration, VBUS and ground traces **shall** be located on the outer layers with the inner layers reserved for signal traces, as required; VCONN traces **may** be routed either on internal or external layers. Table 3-36 defines the requirements for the test PCB thickness and traces. The trace length applies to each PCB (receptacle PCB and plug PCB) and is from the contact terminal to the current source tie point.

- Figure 3-69 provides an **informative** partial trace illustration of the current rating test PCB.
- If short wires are used instead of a current rating test PCB, the wire length **shall not** exceed 70 mm, measured from the plug contact solder point to the other end of the wire. There **shall** be no paddle card or overmold included in the test set-up. Each plug solder tail **shall** be attached with a wire with the wire gauge of AWG 36 for signals, AWG 32 for power (VBUS and VCONN), and AWG 30 for ground.

Figure 3-68 Temperature Measurement Point

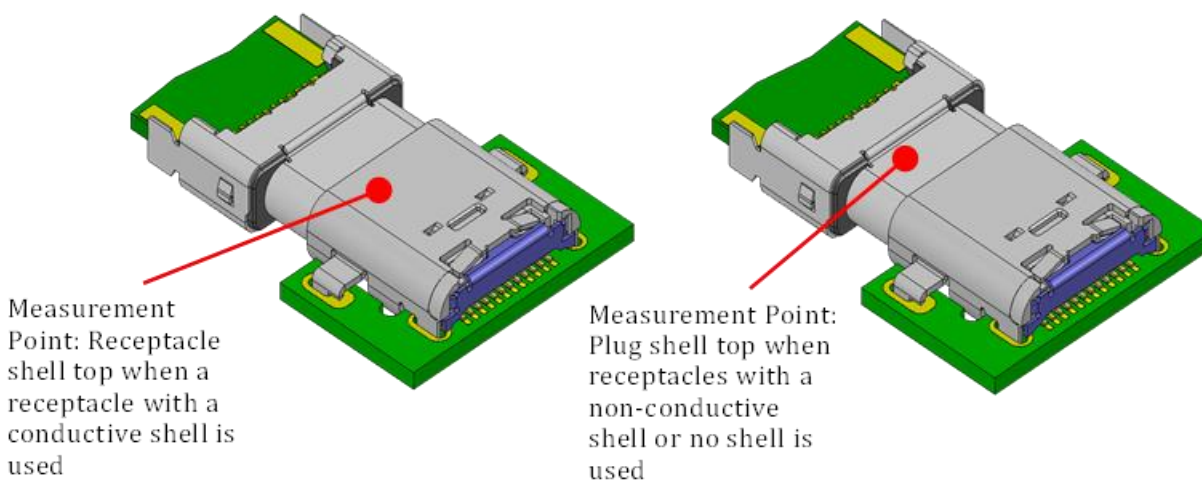
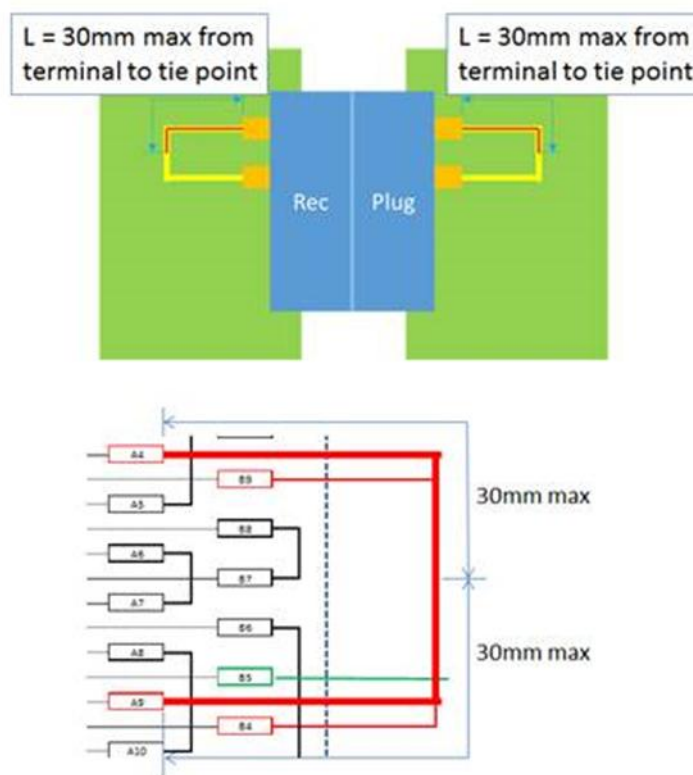


Table 3-36 Current Rating Test PCB

Items	Trace Width (mm)	Trace Length (mm) on each PCB	Thickness
Signal trace	0.25 max.	13 max.	35 μm (1 oz. copper)
Ground trace	1.57 max.	38 max.	35 μm (1 oz. copper)
V _{BUS} and V _{CONN}	1.25 max.	30 max.	35 μm (1 oz. copper)
PCB	N/A	N/A	0.80 – 1.20 mm

Figure 3-69 Example Current Rating Test Fixture Trace Configuration



3.7.8.5 DC Resistance of D+ and D–

The DC Resistance of the D+ and D– in **USB 2.0** High-Speed capable USB Type-C devices and **USB 2.0** High-Speed capable USB Type-C Captive devices **shall** be equal or less than the maximum value specified in Table 3-37. The D+ and D– DC Resistance is the series combination of any resistance in switches, multiplexers, and the USB PHY.

Table 3-37 Maximum DC Resistance Requirement (Normative)

	Maximum DC Resistance
USB Type-C Hosts, Hubs, or Dual-Role Host/Device (USB 2.0 High-speed capable)	13 Ω
USB Type-C Device only (USB 2.0 High-speed capable)	17 Ω
USB Type-C Captive Device (USB 2.0 High-speed capable)	23 Ω

A USB Type-C Host operating in **USB 2.0** High-Speed mode **shall** implement a disconnect threshold voltage (V_{HSDSC}) level as defined in the **USB 2.0** DCR ECN.

3.8 Mechanical and Environmental Requirements (*Normative*)

The requirements in this section apply to all USB Type-C connectors and/or cable assemblies unless otherwise specified. For USB Type-C plug connectors and cable assemblies, the test methods assume that the cable exits the overmold in line with mating direction to a USB Type-C receptacle (i.e., straight out the back of the overmold). For USB Type-C plug connectors and cable assemblies with the cable exiting the overmold in a different direction than straight out the back (e.g., right angle to the mating direction), test fixtures and procedures **shall** be modified as required to accomplish the measurement.

3.8.1 Mechanical Requirements

3.8.1.1 Insertion Force (EIA 364-13)

The initial connector insertion force **shall** be within the range from 5 N to 20 N at a maximum rate of 12.5 mm (0.492") per minute. This requirement does not apply when the connectors are used in a docking application.

It is recommended to use a non-silicone-based lubricant on the latching mechanism to reduce wear. The effects of lubricants **should** be restricted to insertion and extraction characteristics and **should not** increase the resistance of the mated connection.

3.8.1.2 Extraction Force (EIA 364-13)

The initial connector extraction force **shall** be within the range of 8 N to 20 N, measured after a preconditioning of five insertion/extraction cycles (i.e., the sixth extraction). After an additional twenty-five insertion/extraction cycles, the extraction force **shall** be measured again (i.e., the thirty-second extraction) and the extraction force **shall** be:

- a. within 33% of the initial reading, and
- b. within the range of 8 N to 20 N.

The extraction force **shall** be within the range of 6 N to 20 N after 10,000 insertion/ extraction cycles. The extraction force measurement **shall** be performed at a maximum speed of 12.5 mm (0.492") per minute. The extraction force requirement does not apply when the connectors are used in a mechanical docking application.

It is recommended to use a non-silicone-based lubricant on the latching mechanism to reduce wear. The effects of lubricants **should** be restricted to insertion and extraction characteristics and **should not** increase the resistance of the mated connection.

3.8.1.3 Durability or Insertion/Extraction Cycles (EIA 364-09)

The durability rating **shall** be 10,000 cycles minimum for the USB Type-C connector family. The durability test **shall** be done at a rate of 500 ± 50 cycles per hour and no physical damage to any part of the connector and cable assembly **shall** occur.

3.8.1.4 Cable Flexing (EIA 364-41, Condition 1)

No physical damage or discontinuity over 1 ms during flexing **shall** occur to the cable assembly with Dimension X = 22 mm and 500 cycles in each of two planes.

3.8.1.5 Cable Pull-Out (EIA 364-38, Method A)

No physical damage to the cable assembly **shall** occur when it is subjected to a 40 N axial load for a minimum of 1 minute while clamping one end of the cable plug.

3.8.1.6 4-Axis Continuity Test

The USB Type-C connector family *shall* be tested for continuity under stress using a test fixture shown in Figure 3-70 or equivalent.

Figure 3-70 Example of 4-Axis Continuity Test Fixture

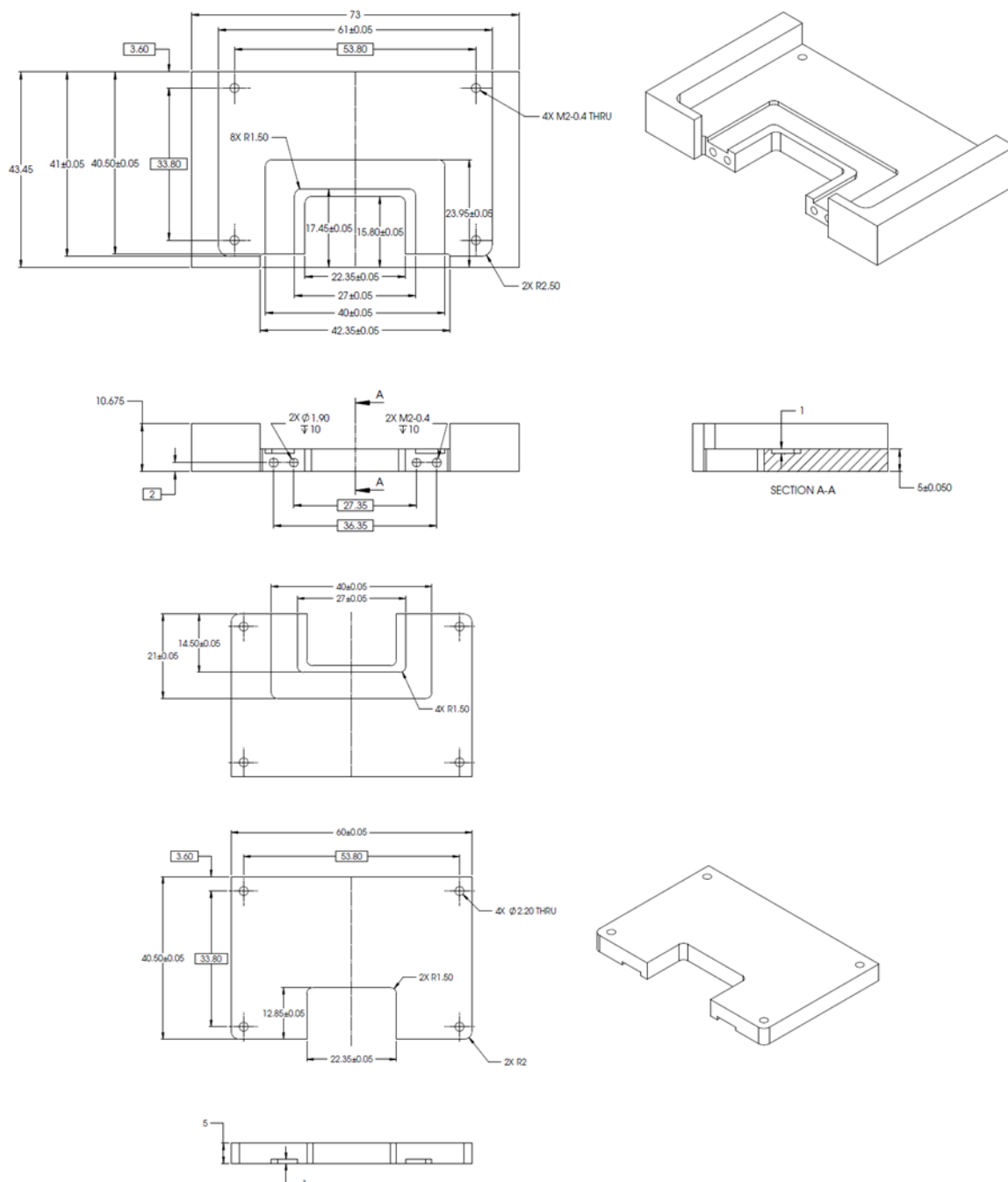
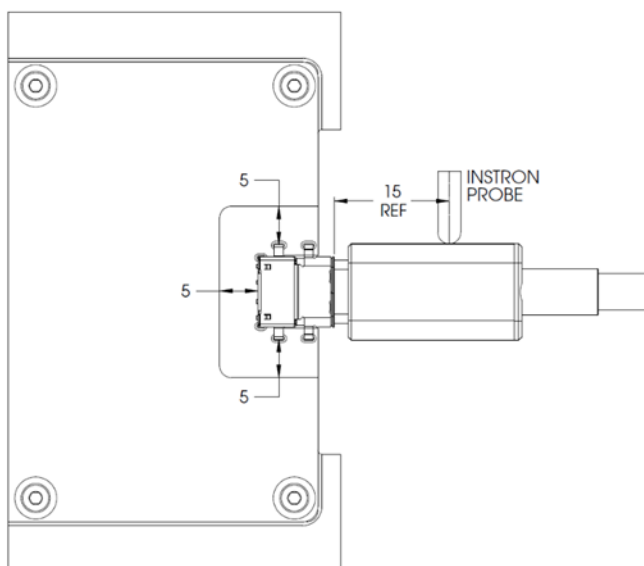


Figure 3-70 Example of 4-Axis Continuity Test Fixture, cont.



Plugs **shall** be supplied with a representative overmold or mounted on a 2-layer printed circuit board (PCB) between 0.8 mm and 1.0 mm thickness as applicable. A USB Type-C receptacle **shall** be mounted on a 2-layer PCB between 0.8 mm and 1.0 mm thickness. The PCB **shall** be clamped on three sides of the receptacle no further than 5 mm away from the receptacle outline. The receptacle PCB **shall** initially be placed in a horizontal plane, and a perpendicular moment **shall** be applied to the plug with a 5 mm ball tipped probe for a period of at least 10 seconds at a distance of 15 mm from the mating edge of the receptacle shell in a downward direction, perpendicular to the axis of insertion. See Table 3-38 for the force and moment to be applied. Any configuration of non-conductive shell receptacles **shall** be tested at the values specified for the vertical receptacle configuration.

Table 3-38 Force and Moment Requirements

Receptacle configuration with respect to mounting surface	Force at 15 mm from receptacle shell mating edge (N)	Moment with respect to receptacle shell mating edge (Nm)
Right angle	20	0.30
Vertical ¹	8	0.12
Notes:		
1. Any configuration of non-conductive shell receptacles shall be tested at the values specified for the vertical receptacle configuration.		

The continuity across each contact **shall** be measured throughout the application of the tensile force. Each non-ground contact **shall** also be tested to confirm that it does not short to the shell during the stresses. The PCB **shall** then be rotated 90 degrees such that the cable is still inserted horizontally and the tensile force in Table 3-38 **shall** be applied again in the downward direction and continuity measured as before. This test is repeated for 180 degree and 270 degree rotations. Passing parts **shall not** exhibit any discontinuities or shorting to the shell greater than 1 μ s duration in any of the four orientations.

One method for measuring the continuity through the contacts is to short all the wires at the end of the cable pigtail and apply a voltage through a pull-up to each of VBUS, USB D+, USB D-, SBU, CC, and TX/RX pins, with the GND pins connected to ground.

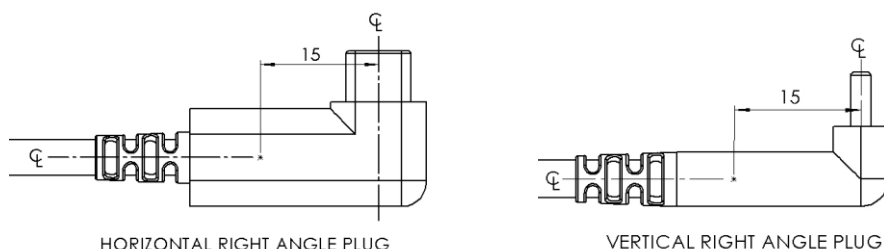
Alternate methods are allowed to verify continuity through all pins.

3.8.1.7 Wrenching Strength

USB Type-C plugs on cable assemblies and fixture plugs without overmold (including PCB-mount USB Type-C plugs) **shall** be tested using the mechanical wrenching test fixture defined in the Universal Serial Bus Type-C Connectors and Cable Assemblies Compliance Document. For plug without overmold, the supplier **shall** provide a plug test fixture that conforms to the specified plug overmold dimensions for the USB Type-C plug (see Figure 3-72). The fixture **may** be metal or other suitable material. Perpendicular moments are applied to the plug with a 5 mm ball tipped probe for a period of at least 10 seconds when inserted in the test fixture to achieve the defined moments in four directions of up or down (i.e., perpendicular to the long axis of the plug opening) and left or right (i.e., in the plane of the plug opening). Compliant connectors **shall** meet the following force thresholds:

- A moment of 0-0.75 Nm (e.g., 50 N at 15 mm from the edge of the receptacle) is applied to a plug inserted in the test fixture in each of the four directions.
- USB Type-C plugs with a right-angle overmold shall apply torque forces of 50 N, 15mm from the centerline of the plug perpendicular to the direction of the cable exit, along the centerline of the cable, (see Figure 3-71). This force will be applied in all four directions for Horizontal right angle plugs and in three direction for Vertical right angle plugs, (force in opposition to the insertion direction causes Vertical right angle plugs to disconnect from the test fixture). If a suitable surface for applying the force is not available, the force will be scaled to provide the equivalent force at the plug interface or the overmold shall use a clamp fixture to provide a suitable surface for applying the torque and moment forces.

Figure 3-71 Torque Force Application Distance for Right-Angle Plugs



- A single plug **shall** be used for this test. Some mechanical deformation **may** occur. The plug **shall** be mated with the continuity test fixture after the test forces have been applied to verify no damage has occurred that causes discontinuity or shorting. The continuity test fixture **shall** provide a planar surface on the mating side located 6.20 ± 0.20 mm from the receptacle Datum A, perpendicular to the direction of insertion. No moment forces are applied to the plug during this continuity test. Figure 3-73 illustrates an example continuity test fixture to perform the continuity test. The Dielectric Withstanding Voltage test **shall** be conducted after the continuity test to verify plug compliance.
- Note that in order to test horizontal right-angle plugs, a receptacle fixture with a vertically mounted receptacle may be required to access the inside surface of the overmold for applying the test force.

Figure 3-72 Example Wrenching Strength Test Fixture for Plugs without Overmold

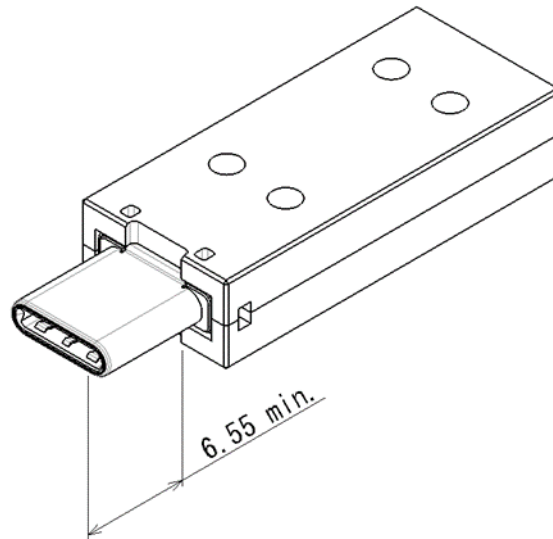
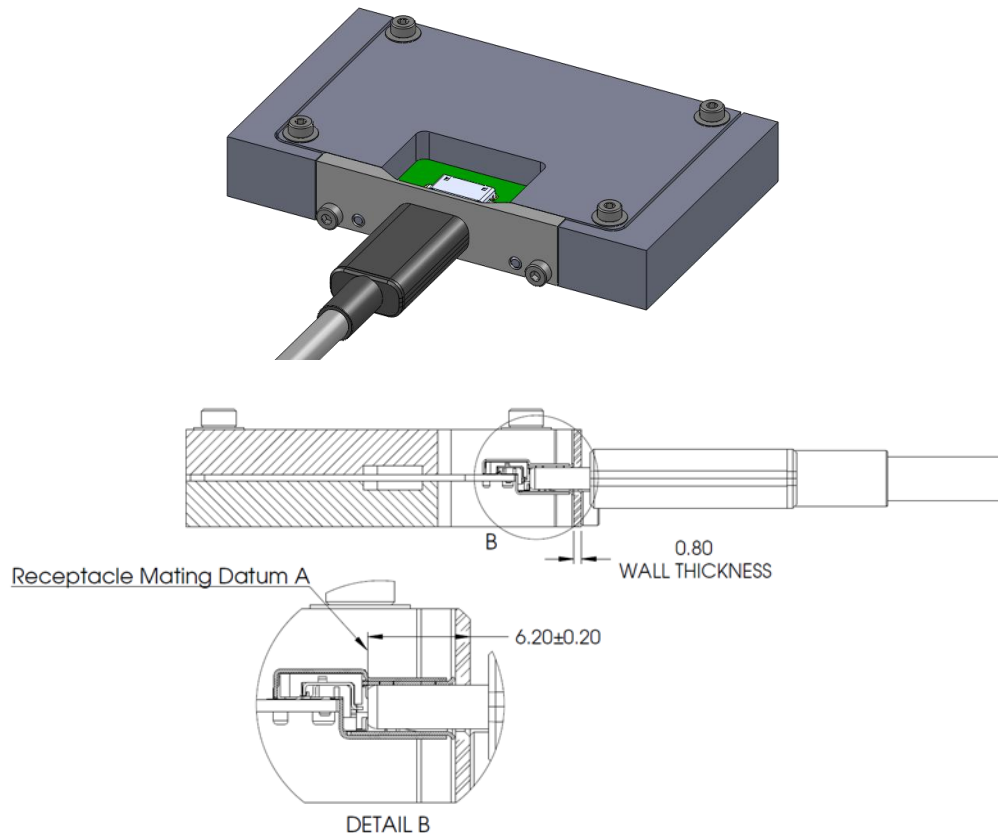


Figure 3-73 Reference Wrenching Strength Continuity Test Fixture



- The plug **shall** disengage from the test fixture or demonstrate mechanical failure (i.e., the force applied during the test procedure peaks and drops off) when a moment of 2.0 Nm is applied to the plug in the up and down directions, the wide side for a right-angle overmold, and a moment 3.5 Nm is applied to the plug in the left and right directions, the narrow side for a right-angle overmold. A new

plug is required for each of the four test directions. An example of the mechanical failure point and an illustration of the wrenching test fixture are shown in Figure 3-74 and Figure 3-75, respectively.

Figure 3-74 Example of Wrenching Strength Test Mechanical Failure Point

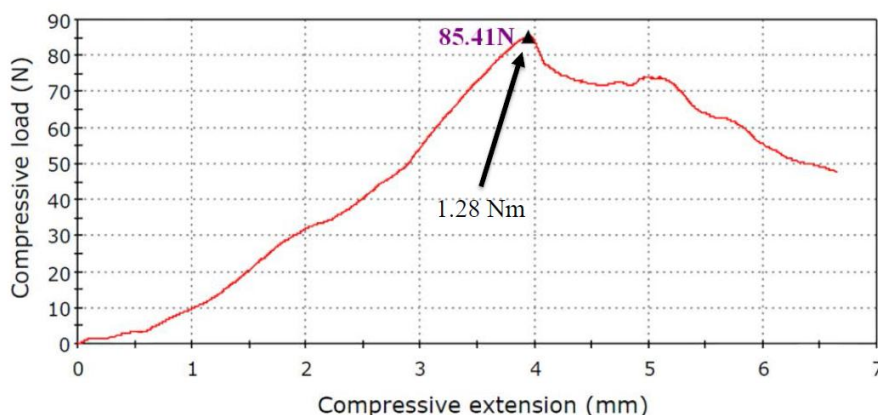
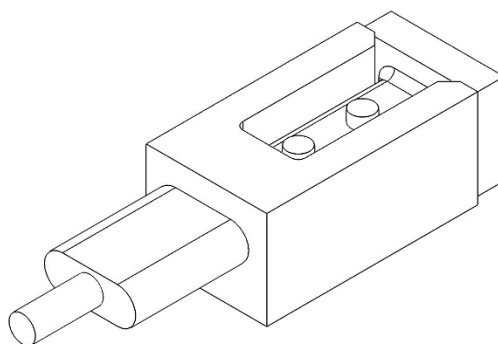


Figure 3-75 Wrenching Strength Test with Cable in Fixture



3.8.1.8 Restriction of Hazardous Substances

It is recommended that components be RoHS compliant.

3.8.2 Environmental Requirements

The connector interface environmental tests *shall* follow EIA 364-1000.01, Environmental Test Methodology for Assessing the Performance of Electrical Connectors and Sockets Used in Business Office Applications.

Since the connector defined has more than 0.127 mm wipe length, Test Group 6 in EIA 364-1000.01 is not required. The temperature life test duration and the mixed flowing gas test duration values are derived from EIA 364-1000.01 based on the field temperature per the following.

Table 3-39 Environmental Test Conditions

Temperature Life test temperature and duration	105 °C for 120 hours
Temperature Life test temperature and duration for preconditioning	105 °C for 72 hours
Mixed flowing gas test duration	7 days

The pass/fail criterion for the low-level contact resistance (LLCR) is as defined in Section 3.7.8.1. The durability ratings are defined in Section 3.8.1.3.

3.8.2.1 Reference Materials (*Informative*)

This specification does not specify materials for connectors and cables. Connector and cable manufacturers **should** select appropriate materials based on performance requirements. The information below is provided for reference only.

Note: Connector and cable manufacturers **should** comply with contact plating requirements per the following options:

Option I

Receptacle: Contact area: (Min) 0.05 μm Au + (Min) 0.75 μm Ni-Pd on top of (Min) 2.0 μm Ni

Plug: Contact area: (Min) 0.05 μm Au + (Min) 0.75 μm Ni-Pd on top of (Min) 2.0 μm Ni

Option II

Receptacle: Contact area: (Min) 0.75 μm Au on top of (Min) 2.0 μm Ni

Plug: Contact area: (Min) 0.75 μm Au on top of (Min) 2.0 μm Ni

Other reference materials that connector and cable manufacturers select based on performance parameters listed in Table 3-40 are for reference only.

Table 3-40 Reference Materials

Component	Material
Cable	Conductor: copper with tin or silver plating
	SDP Shield: AL foil or AL/mylar foil
	Coaxial shield: copper strand
	Braid: Tin plated copper or aluminum
	Jacket: PVC or halogen free substitute material
Cable Overmold	Thermoset or thermoplastic
Connector Shells	Stainless steel or phosphor bronze
Plug Side Latches	Stainless steel
Receptacle Mid-Plate	Stainless steel
Plug Internal EMC Spring	Stainless steel or high yield strength copper alloy
Receptacle EMC Pad	Stainless steel or phosphor bronze
Receptacle Shell	Stainless steel or phosphor bronze
Receptacle Tongue	Glass-filled nylon
Housing	Thermoplastics capable of withstanding lead-free soldering temperature
Note: Halogen-free materials should be considered for all plastics	

3.9 Docking Applications (*Informative*)

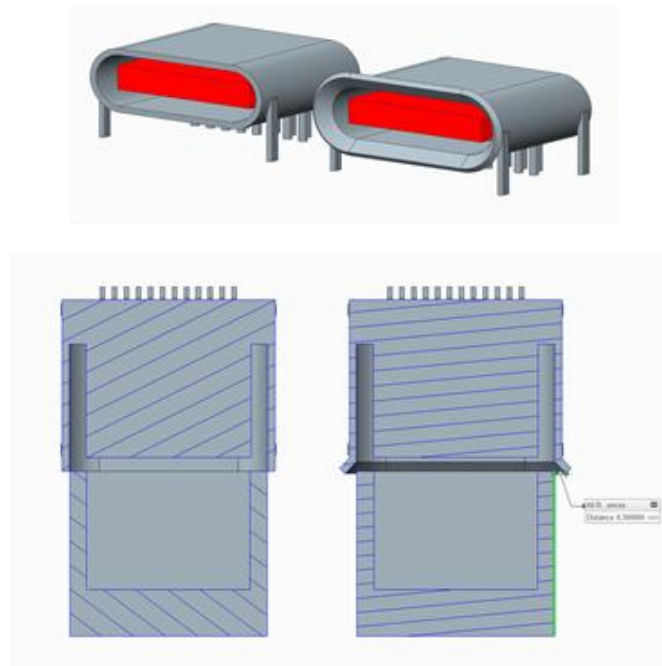
In this specification, docking refers to plugging a device directly into a dock without using a cable assembly. The USB Type-C connector is defined to support such applications.

The connector is only part of a docking solution. A complete docking solution at the system level **may** also include retention or locking mechanisms, alignment mechanisms, docking plug mounting solutions, and protocols supported through the connector. This specification does not attempt to standardize system docking solutions, therefore there is no interoperability requirement for docking solutions.

The following list includes the requirements and guidelines when using the USB Type-C connector for docking:

1. The USB Type-C plug used for docking **shall** work with compliant USB Type-C receptacle. It **shall** comply with all dimensional, electrical, and mechanical requirements.
2. If the plug on the dock does not include the side latches, then the dock **should** provide a retention or locking mechanism to secure the device to the plug. The retention latches also serve as one of the ground return paths for EMC. The docking design **should** ensure adequate EMC performance without the side latches if they are not present.
3. The internal EMC fingers are not required for the docking plug as long as the receptacle and plug shells have adequate electrical connection.
4. Alignment is critical for docking. Depending on system design, standard USB Type-C connectors alone **may not** provide adequate alignment for mating. System level alignment is highly recommended. Alignment solutions are implementation-specific.
5. Fine alignment is provided by the connector. The receptacle front face **may** have lead-in features for fine alignment. Figure 3-76 shows an example of a USB Type-C receptacle with a lead-in flange compared to a receptacle without the flange.

Figure 3-76 USB Type-C Cable Receptacle Flange Example



3.10 Implementation Notes and Design Guides

This section discusses a few implementation notes and design guides to help users design and use the USB Type-C connectors and cables.

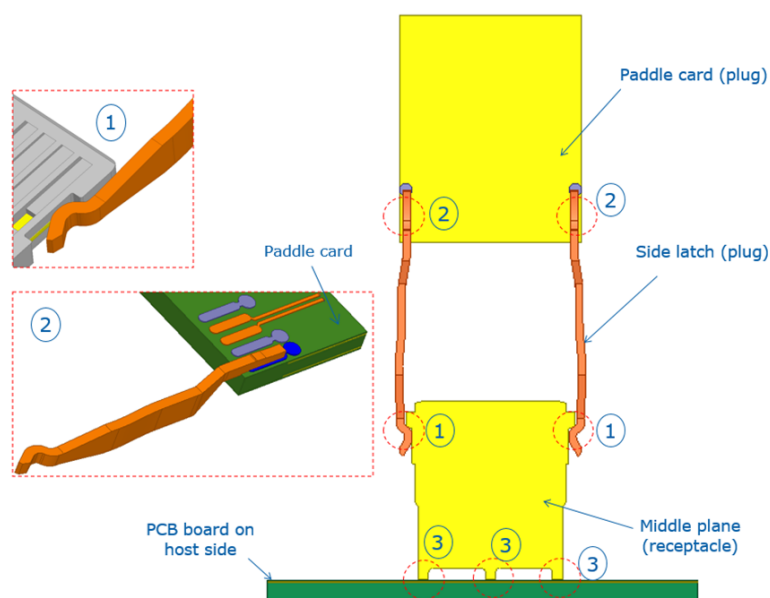
3.10.1 EMC Management (*Informative*)

Connector and cable assembly designers, as well as system implementers **should** pay attention to receptacle and cable assembly shielding to ensure a low-impedance grounding path. The following are guidelines for EMC management:

- The quality of raw cables **should** be ensured. The intra-pair skew or the differential to common mode conversion of the TX/RX pairs has a significant impact on cable EMC and **should** be controlled within the limits of this specification.

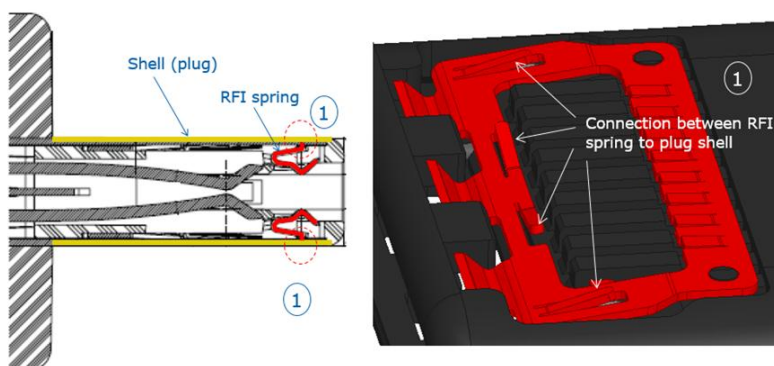
- The cable external braid **should** be physically connected to the plug metal shell as close to 360° as possible to control EMC. Without appropriate shielding termination, even a perfect cable with zero intra-pair skew **may** not meet EMC requirements. Copper tape **may** be needed to shield off the braid termination area.
- The wire termination contributes to common-mode noise. The breakout distance for the wire termination **should** be kept as small as possible to optimize EMC and signal integrity performance. If possible, symmetry **should** be maintained for the two lines within a differential pair.
- Besides the mechanical function, the side latches on the plug and the mid-plate in the receptacle also play a role for EMC. This is illustrated in Figure 3-77:
 1. The side latch **should** have electrical connection to the receptacle mid-plate (a docking plug **may not** have side latches).
 2. The side latches **should** be terminated to the paddle card GND plane inside the plug.
 3. The mid-plate **should** be directly connected to system PCB GND plane with three or more solder leads/tails.

Figure 3-77 EMC Guidelines for Side Latch and Mid-Plate



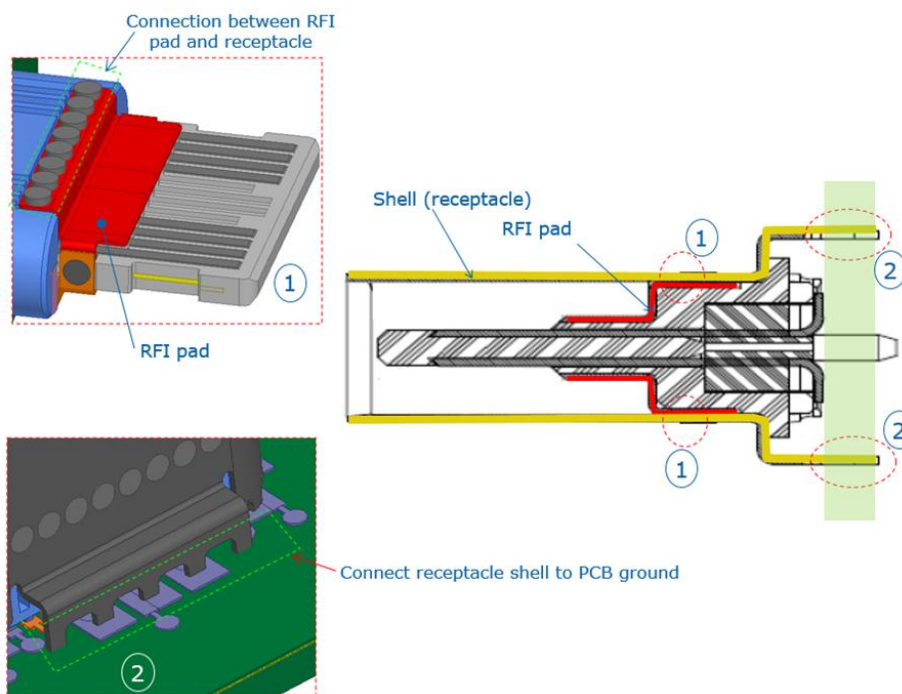
- The internal RFI finger inside the plug **should** have adequate connection points to the inner surface of the plug shell. Four or more connection points are recommended as illustrated in Figure 3-78.

Figure 3-78 EMC Finger Connections to Plug Shell



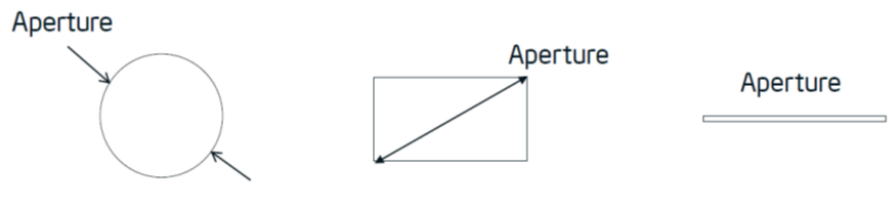
- The EMC fingers inside the plug mates with the EMC pad in the receptacle. It is important for the EMC pad to have adequate connections to the receptacle shell. As illustrated in Figure 3-79, there are multiple laser welding points between the EMC pads and the receptacle shell, top and bottom.
- The receptacle shell **should** have sufficient connection points to the system PCB GND plane with apertures as small as possible. Figure 3-79 illustrates an example with multiple solder tails to connect the receptacle shell to system PCB GND.

Figure 3-79 EMC Pad Connections to Receptacle Shell



- Apertures in the receptacle and plug shells **should** be minimized. If apertures are unavoidable, a maximum aperture size of 1.5 mm is recommended. See Figure 3-80 for aperture illustrations. Copper tape **may** be applied to seal the apertures inside the cable plug.

Figure 3-80 Examples of Connector Apertures

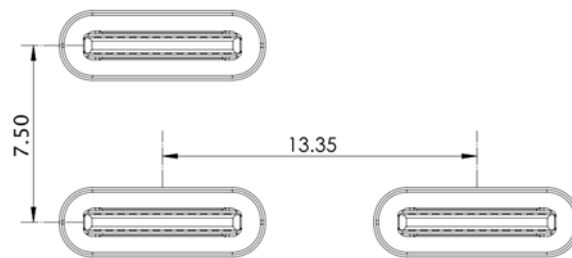


- The receptacle connectors **should** be connected to metal chassis or enclosures through grounding fingers, screws, or any other way to manage EMC.

3.10.2 Stacked and Side-by-Side Connector Physical Spacing (*Informative*)

Stacked and side-by-side USB connectors are commonly used in PC systems. Figure 3-81 illustrates the recommended spacing between connectors for stacked and side-by-side configurations.

Figure 3-81 Recommended Minimum Spacing between Connectors



3.10.3 Cable Mating Considerations (*Informative*)

The receptacle mounting location, exterior product surfaces, cable overmold, and plug mating length need to be considered to ensure the USB Type-C plug is allowed to fully engage the USB Type-C receptacle. Figure 3-82 illustrates the recommended minimum plug overmold clearance to avoid interference between the plug overmold and enclosure opening, allowing the cable plug to fully seat in the product receptacle.

Figure 3-82 Recommended Minimum Plug Overmold Clearance

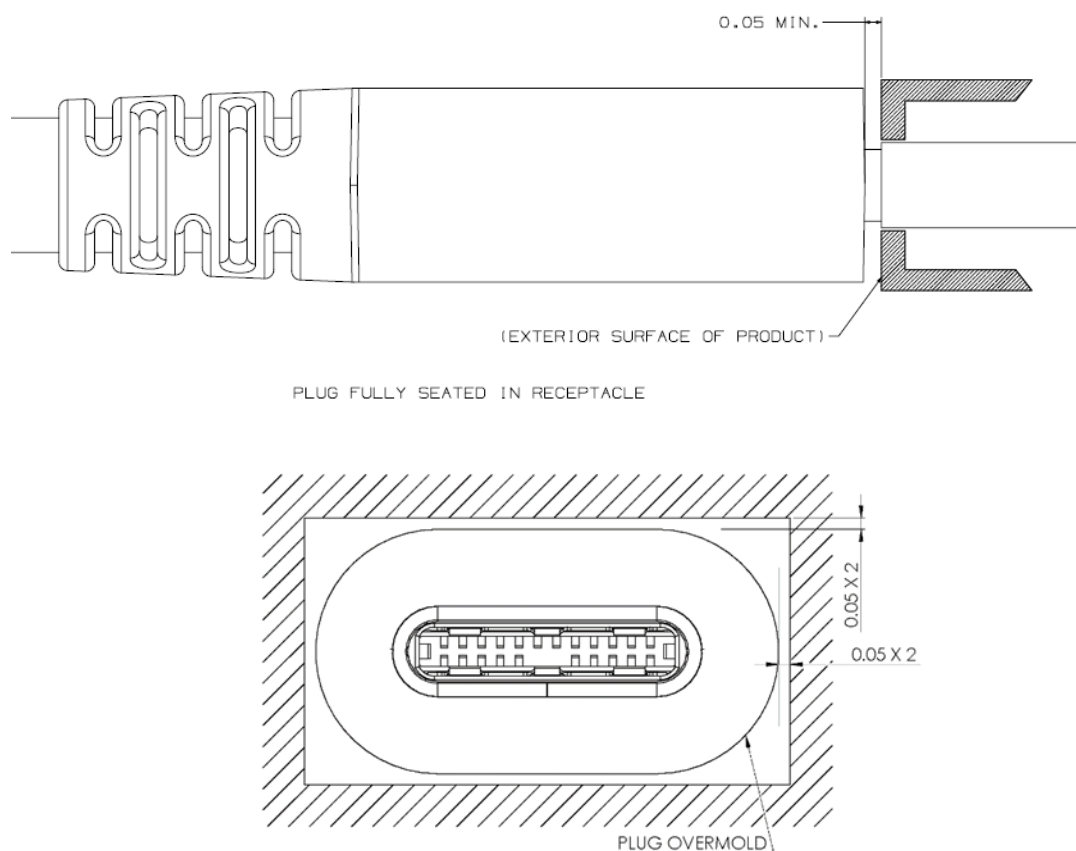
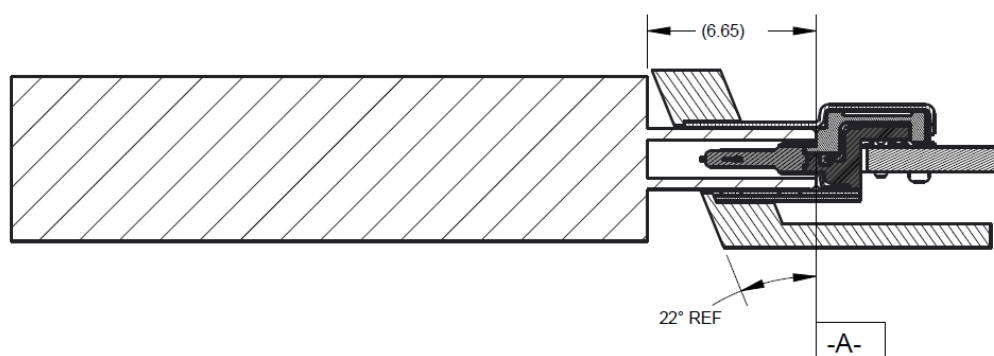


Figure 3-83 illustrates special considerations required when external walls are angled. For such applications, the USB Type-C receptacle shell **may not** provide as much mechanical alignment protection to the receptacle tongue as in the full shell design. Design options to allow the receptacle to pass mechanical test requirements include relief in the exterior wall surface to allow use of a full shell receptacle or use of a receptacle specifically designed for the application.

Figure 3-83 Cable Plug Overmold and an Angled Surface



3.11 Extended Power Range (EPR) Cables

3.11.1 Electrical Requirements

Extended Power Range cables have additional requirements to assure that these cables can deliver the full defined voltage and current range for **USB PD** EPR operation.

EPR cables **shall** functionally support a reported 50 V and 5 A operation. The minimum functional voltage that a cable **shall** support is 50.9 V. The electrical components potentially in the path of VBUS in an EPR cable, e.g., bypass capacitors, **should** be minimally rated for 63 V.

See Appendix H for information about **USB PD** high-voltage design and mitigating arcing damage during cable withdrawal.

3.11.2 EPR Cable Identification Requirements

All EPR cables **shall** be Electronically Marked and include EPR-specific information in the eMarker as defined by the **USB PD** specification. As defined in the **USB PD** specification, EPR cables are marked as 50 V and 5 A capable.

Refer to Sections 3.4.1.1 and 3.4.2.1 regarding cable certification and identification.

4 Functional

This chapter covers the functional requirements for the signaling across the USB Type-C® cables and connectors. This includes functional signal definition, discovery and configuration processes, and power delivery.

4.1 Signal Summary

Table 4-1 summarizes the list of signals used on the USB Type-C connectors.

Table 4-1 USB Type-C List of Signals

Signal Group	Signal	Description
USB 3.2 / USB4	TXp1, TXn1 RXp1, RXn1 TXp2, TXn2 RXp2, RXn2	Both the USB 3.2 SuperSpeed USB and USB4 serial data interfaces defines 1 differential transmit pair and 1 differential receive pair per lane. On a USB Type-C receptacle, two sets of signal pins are defined to support dual-lane operation and enable plug flipping feature.
USB 2.0	Dp1, Dn1 Dp2, Dn2	USB 2.0 serial data interface defines a differential pair. On a USB Type-C receptacle, two set of USB 2.0 signal pins are defined to enable plug flipping feature.
Configuration	CC1, CC2 (receptacle) CC (plug)	CC channel in the plug used for connection detect, interface configuration and VCONN.
Auxiliary signals	SBU1, SBU2	Sideband Use. For USB4 , these signals are used for SBTX and SBRX.
Power	VBUS	USB cable bus power
	VCONN (plug)	USB plug power
	GND	USB cable return current path

4.2 Signal Pin Descriptions

4.2.1 USB 3.2/USB4 Pins

TXp1, TXn1 TXp2, TXn2	<p>These pins are required to implement the system's transmit path of either a USB 3.2 SuperSpeed or USB4 TX/RX interface. The transmitter differential pair in a port are routed to the receiver differential pair in the port at the opposite end of the path. Depending on the established connection, the USB 3.2 Specification or USB4 Specification defines all electrical characteristics, enumeration, protocol, and management features for this interface.</p> <p>Two pairs of pins are defined to enable dual-lane operation – see Section 4.5.1.1 for further definition.</p>
RXp1, RXn1 RXp2, RXn2	<p>These pins are required to implement the system's receive path of a USB 3.2 SuperSpeed or USB4 TX/RX interface. The receiver differential pair in a port are routed to the transmitter differential pair in the port at the opposite end of the path. Depending on the established connection, the USB 3.2 Specification or USB4 Specification defines all electrical characteristics, enumeration, protocol, and management features for this interface.</p> <p>Two pairs of pins are defined to enable dual-lane operation – see Section 4.5.1.1 for further definition.</p>

4.2.2 USB 2.0 Pins

Dp1, Dn1
(Dp2, Dn2)

These pins are required to implement **USB 2.0** functionality. **USB 2.0** in all three modes (LS, FS, and HS) is supported. The **USB 2.0** Specification defines all electrical characteristics, enumeration, bus protocol and bus management features for this interface.

Two pairs of pins are defined to enable the plug flipping feature – see Section 4.5.1.1 for further definition.

4.2.3 Auxiliary Signal Pins

SBU1, SBU2

These pins are assigned to sideband use. For **USB4**, these signals are used for SBTX and SBRX. Refer to Section 4.3 for the functional requirements.

4.2.4 Power and Ground Pins

VBUS

These pins are for USB cable bus power as defined by the USB specifications. VBUS is only present when a Source-to-Sink connection across the CC channel is present – see Section 4.5.1.2.1. Refer to Section 4.4.2 for the functional requirements for VBUS.

VCONN

VCONN is applied to the unused CC pin to supply power to the local plug. Refer to Section 4.4.3 for the functional requirements for VCONN.

GND

Return current path.

4.2.5 Configuration Pins

CC1, CC2, CC

These pins are used to detect connections and configure the interface across the USB Type-C cables and connectors. Refer to Section 4.5 for the functional definition. Once a connection is established, CC1 or CC2 will be reassigned for providing power over the VCONN pin of the plug – see Section 4.5.1.2.1.

4.3 Sideband Use (SBU)

The Sideband Use pins (SBU1 and SBU2) are limited to the uses as defined by this specification and additional functionality defined in the **USB4** Specification, liquid detection, and **Alternate Modes**. See Appendix E for use of the SBU pins in **Alternate Modes**.

The SBU pins on a port that *does not* implement either **USB4** or **Alternate Modes** *shall* either be open circuit or have a weak pull-down to ground no stronger than **zSBU Termination**.

The SBU pins on a port that implements either **USB4** or an **Alternate Mode** that utilize the SBU pins *shall* either be open circuit or have a weak pull-down to ground no stronger than **zSBU Termination** prior to completing **USB4** or **Alternate Mode** entry.

The SBU pins *shall* meet the USB Safe State electrical requirements (see Table E-1) when transitioning to or from liquid detection, open circuit or **zSBU Termination**, **USB4** operation, or **Alternate Modes**.

Note: Using SBU pins for liquid detection may only be practical in an unattached state because an attached cable or port partner may introduce loading that interferes with the liquid detection method.

These pins are pre-wired in the standard USB Full-Featured Type-C cable as individual single-ended wires (SBU_A and SBU_B). Note that SBU1 and SBU2 are cross-connected in the cable.

When operating in **USB4**, these pins are used as the **USB4** Sideband Channel with SBU1 mapping to SBTX and SBU2 mapping to SBRX. SBTX and SBRX functional requirements are as defined in the **USB4** specification. When a port determines that the locally-inserted plug is flipped (i.e., CC1 is open, CC2 is terminated), the **USB4** specification (reference Sideband Channel Lane Reversal) dictates that the port flip the SBTX and SBRX mappings to SBU1 and SBU2 in order to assure proper sideband transmit-to-receive end-to-end operation.

4.4 Power and Ground

4.4.1 IR Drop

The maximum allowable cable IR drop for ground (including ground on a captive cable) **shall** be 250 mV and for VBUS **shall** be 500 mV through the cable to the cable's maximum rated VBUS current capacity. When VCONN is being sourced, the IR drop for the ground **shall** still be met considering any additional VCONN return current.

Figure 4-1 illustrates what parameters contribute to the IR drop and where it **shall** be measured. The IR drop includes the contact resistance of the mated plug and receptacles at each end.

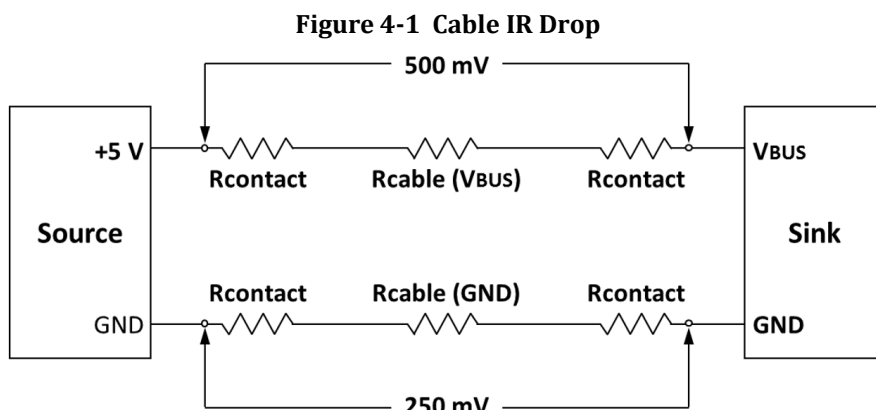
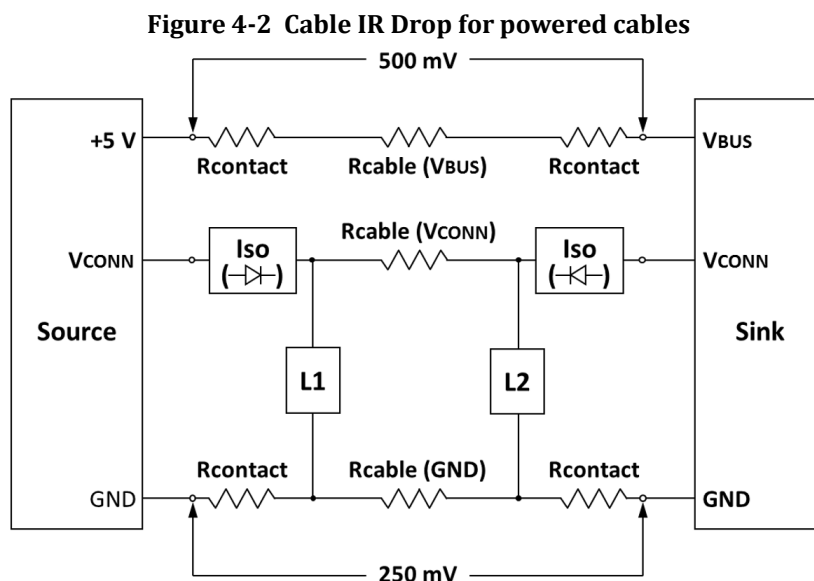


Figure 4-2 illustrates what parameters contribute to the IR drop for a powered cable and where it **shall** be measured. Note that the powered cable includes isolation elements (Iso) and loads (L1 and L2) for the functions in the powered cable such as **USB PD** controllers. The IR drop **shall** remain below 250 mV in all cases.



4.4.2 VBUS

The allowable default range for VBUS as measured at the Source receptacle **shall** be as defined by the **USB 2.0** and **USB 3.2** specifications. For **USB4**, the **USB 3.2** specification is used for this requirement. NOTE that due to higher currents allowed, legacy devices **may** experience a higher voltage (up to 5.5V maximum) at light loads.

The Source's USB Type-C receptacle VBUS pin **shall** remain unpowered and **shall** limit the capacitance between VBUS and GND as specified in Table 4-2 until a Sink is attached. The VBUS pin **shall** return to the unpowered state when the Sink is detached. See Table 4-32 for VBUS timing values. Legacy hosts/chargers that by default source VBUS when connected using any legacy USB connector (Standard-A, Micro-B, etc.) to USB Type-C cable or adapter are exempted from these two requirements.

A DRP or Source (or device with Accessory Support) implementing an **Rp** pull-up as its method of connection detection **shall** provide an impedance between VBUS and GND on its receptacle pins as specified in Table 4-2 when not sourcing power on VBUS (i.e., when in states **Unattached.SRC** or **Unattached.Accessory**).

Table 4-2 VBUS Source Characteristics

	Minimum	Maximum	Notes
VBUS Leakage Impedance	72.4 k Ω		Leakage between VBUS pins and GND pins on receptacle when VBUS is not being sourced.
VBUS Capacitance		3000 μ F	Capacitance for source-only ports between VBUS and GND pins on receptacle when VBUS is not being sourced.
		10 μ F	Capacitance for DRP ports between VBUS and GND pins on receptacle when VBUS is not being sourced.

Table 4-3 specifies VBUS Sink characteristics with regard to disconnect behavior based on monitoring VBUS. Sinks **may** monitor the CC pin for the removal of **Rp** by the Source as an additional indication of disconnect.

Table 4-3 VBUS Sink Characteristics

	Minimum	Maximum	Notes
<i>tSinkDisconnect</i>		40 ms	Time limit for transition from <i>Attached.SNK</i> to <i>Unattached.SNK</i>
<i>vSinkDisconnect</i> ¹	0.8 V	3.67 V	Threshold used for transition from <i>Attached.SNK</i> to <i>Unattached.SNK</i> when VBUS is 5 V. This also applies for USB PD contracts at 5 V. For USB PD contracts at 5 V, the Sink <i>shall</i> take IR drop and margin into account when selecting this threshold.
		PPS_APDO_Min_Voltage * 0.95	Threshold used for transition from <i>Attached.SNK</i> to <i>Unattached.SNK</i> . This applies for USB PD PPS contracts for RDO Output Voltage less than or equal to 5 V. This also applies for USB PD PPS contracts operating in the Current Limit mode. The Sink <i>shall</i> take IR drop and margin into account when selecting this threshold.
<i>vSinkPD_min</i> ¹		vNew – 750 mV + vValid	The minimum valid VBUS voltage seen by sink when negotiated through USB PD . vNew = vSrcNew (min) or vPpsNew (min) or vAvsNew (min) as defined in USB PD . 750 mV = 500 mV + 250 mV (maximum IR drop) vValid = vSrcValid (min) or vPpsValid (min) or vAvsNew (min) as defined in USB PD .
<i>vSinkDisconnectPD</i> ¹	90% of <i>vSinkPD_min</i>	<i>vSinkPD_min</i>	VBUS disconnect threshold when VBUS voltage was negotiated through USB PD to a value above 5 V. This applies for USB PD PPS contracts for RDO Output Voltage above 5 V. This also applies for USB PD PPS contracts operating in the Constant Voltage mode.
VBUS Capacitance		10 µF	Capacitance between VBUS and GND pins on receptacle when not in <i>Attached.SNK</i> .

Note 1: See Section 4.5.2.2.5.2 with regard to applicability of this requirement.

4.4.3 VCONN

VCONN is provided by the Source to power cables with electronics in the plug. VCONN is provided over the CC pin that is determined not to be connected to the CC wire of the cable.

Initially, VCONN is sourced on Source USB Type-C receptacles based on the *Attached.SRC* requirements in Section 4.5.2.2.9.1. Subsequently, if VCONN is not explicitly required by the cable or device as indicated in its eMarker, VCONN *may* be removed as described in Table 4-4. **USB PD VCONN_Swap** command also provides the Source a means to request that the attached Sink source VCONN.

Table 4-4 USB Type-C Source Port's VCONN Requirements Summary

D+/D-	TX/RX, VPD	> 3 A	VCONN Requirements
No	No	No	Not required to source VCONN
Yes	No	No	Not required to source VCONN
Yes	Yes	No	Required to source 1 W for x1 implementations and 1.5 W for x2 implementations. If after reading the cable's eMarker, the Source has determined that the cable is not a VPD and the Cable Termination Type field is 00b, it may remove VCONN power.
No	No	Yes	Required to source 100 mW. VCONN power may be removed after the source has read the cable's eMarker and has determined the cable's current carrying capacity.
Yes	No	Yes	Required to source 100 mW. VCONN power may be removed after the source has read the cable's eMarker and has determined the cable's current carrying capacity.
Yes	Yes	Yes	Required to source 1 W for x1 implementations and 1.5 W for x2 implementations. If after reading the cable's eMarker, the Source has determined the cable's current carrying capacity and the cable is not a VPD and the Cable Termination Type field is 00b, it may remove VCONN power.

Table 4-5 provides the voltage and power requirements that **shall** be met for VCONN. See Section 4.9 for more details about Electronically Marked Cables. See Appendix E regarding **optional** support for an increased VCONN power range in **Alternate Modes**.

Table 4-5 VCONN Source Characteristics

	Minimum	Maximum	Notes
vVonValid	3.0 V	5.5 V	The voltage range over which VCONN is considered valid.
Power for Sources with TX/RX Signals	x1 1 W		Source may latch-off VCONN if excessive power is drawn beyond the specified inrush and mode wattage.
	x2 1.5 W		Source may disable VCONN per Table 4-4. Alternate modes may require higher power.
Power for Sources with VPD support	1 W		Source may latch-off VCONN if excessive power is drawn beyond the specified inrush and mode wattage.
Power for Sources in USB Suspend or without TX/RX Signals	100 mW		Minimum power Source must provide in USB Suspend or without TX/RX signals. ¹ Source may disable VCONN per Table 4-4.
Rdch	30 Ω	6120 Ω	Discharge resistance applied in UnattachedWait.SRC between the CC pin being discharged and GND.
Note 1: During transition from U3 to U0, VCONN Source shall provide up to max U0 power.			

To aid in reducing the power associated with supplying VCONN, a Source is allowed to either not source VCONN or turn off VCONN under any of the following conditions:

- **Ra** is not detected on the CC pin after **tCCDebounce** when the other CC pin is in the **SRC.Rd** state, or
- if there is no GoodCRC response to **USB PD Discover Identity** messages sent to SOP.

If the power source used to supply VCONN power is a shared power source for other USB VCONN and VBUS outputs, it must be bypassed with capacitance identical to the VBUS capacitance requirements of **USB 3.2**

Section 11.4.4 – Dynamic Attach and Detach. Any VCONN power source bypass capacitance must be isolated from the CC pins when VCONN is not being provided.

Table 4-6 provides the requirements that **shall** be met for cables that consume VCONN power.

Table 4-6 Cable VCONN Sink Characteristics

	Minimum	Maximum	Notes
Voltage	3.0 V	5.5 V	Voltage range (vVCONNValid) when VCONN is valid.
Inrush Capacitance		10 µF	A cable shall not present more than the equivalent inrush capacitance to the VCONN source. The active cable is responsible for discharging its capacitance.
Power for Electronically Marked Passive Cables		20 mW	See Section 4.9. Measured with no USB PD traffic at least tRaWeaken after VCONN applied. Note: 75mW max allowed for the first tRaWeaken after VCONN applied.
USB 3.2 Power for Active Cables in U-states		See Table 6-6	U0, U1, U2, U3, Rx.Detect, and eSS.Disabled.
tVCONNDischarge		230 ms	Time from cable disconnect to vVCONNDischarge met.
vVCONNDischarge		800 mV	VCONN voltage after tVCONNDischarge
vVCONNDisconnect	1 V	2.4 V	Threshold used to detect VCONN disconnect.
iRaDetect		10 mA	The maximum current drawn from VCONN when the voltage is below vVCONNValid. Note: This current is below the 75 mW allowance for the first 500 ms at 5.5 V.
tRaReconnect		1 ms	Time from VCONN falling below vVCONNDisconnect at the cable plug until the cable has re-applied Ra .
tRaWeaken	21 ms	1.2 s	Time from VCONN exceeding vVCONNDisconnect until the cable removes or weakens Ra .
tVCONNSwitch		teMarkerReady	Cables that optionally use VBUS shall power from VCONN within teMarkerReady .

Any cable (or direct connect device) that uses VCONN to power circuitry shall implement cable **Ra** management as defined in Section 4.5.2.5. The cable **shall** remove or weaken **Ra** according to the state diagram behavior in Section 4.5.2.5. The cable **shall** reapply **Ra** according to the state diagram behavior in Section 4.5.2.5. The cable **shall** discharge VCONN to below **vVCONNDischarge** on a cable disconnect. The cable **shall** control **Ra** at each of its ends independently based on the VCONN on that end.

Implementation Note: Increasing **Ra** to 20KΩ will meet both the power dissipation for electronically marked passive cables and discharge 10µF to less than **vVCONNDischarge** in **tVCONNDischarge**.

Table 4-7 VCONN-Powered Accessory (VPA) Sink Characteristics

	Minimum	Maximum	Notes
Voltage	3.0 V	5.5V	Voltage range ($vV_{CONNValid}$) when V_{CONN} is valid.
Inrush Capacitance		10 μ F	An accessory shall not present more than the equivalent inrush capacitance to the V_{CONN} source. The accessory is responsible for discharging its capacitance.
Power before Alternate Mode Entry		35 mW	Maximum power in USB suspend. Note: Power shall be reduced 5 seconds after V_{CONN} is applied if no Alternate Mode Entry has occurred. A V_{CONN} power cycle may be required to re-enable USB PD communication. Note: 75 mW max allowed for the first 500 ms after V_{CONN} is applied.
$tV_{CONNDischarge}$		230 ms	Time from VPA disconnect to $vV_{CONNDischarge}$ met.
$vV_{CONNDischarge}$		800 mV	V_{CONN} voltage after $tV_{CONNDischarge}$.
$iRaDetect$		10 mA	The maximum current drawn from V_{CONN} when the voltage is below $vV_{CONNValid}$. Note: This current is below the 75 mW allowance for the first 500 ms at 5.5 V.
$vRaReconnect$	800 mV		Voltage at which the VPA shall reapply Ra on the falling edge of V_{CONN} .
$tRaWeaken$		1.2 s	Time from V_{CONN} exceeding $vV_{CONNDisconnect}$ until the VPA removes or weakens Ra .
$vV_{CONNDisconnect}$	1 V	2.4 V	Threshold used to detect V_{CONN} disconnect.

The **VPA shall** remove or weaken **Ra** within $tRaWeaken$ (as defined in Table 4-7) after V_{CONN} enters the valid voltage range ($vV_{CONNValid}$).

The **VPA shall** reapply **Ra** when V_{CONN} falls below $vRaReconnect$ as defined in Table 4-7. The **VPA shall** discharge V_{CONN} to below $vV_{CONNDischarge}$ within $tV_{CONNDischarge}$ on a cable disconnect. The **VPA shall** consider the V_{CONN} capacitance present in the accessory when discharging V_{CONN} .

The maximum power consumption while in an **Alternate Mode** is defined by the specification specific to the **Alternate Mode** being used.

Table 4-8 VCONN-Powered Device (VPD) Sink Characteristics

	Minimum	Maximum	Notes
Voltage	3.0 V	5.5V	Voltage range (vVCONNValid) when VCONN is valid.
Inrush Capacitance		10 µF	A device shall not present more than the equivalent inrush capacitance to the VCONN source. The device is responsible for discharging its capacitance.
Power before USB enumeration		35 mW	Maximum power in USB suspend. Note: 75 mW max allowed for the first 500 ms after VCONN is applied.
Power when active		500 mW (USB 2.0) 750 mW (USB 3.2)	A VPD shall only expose a low-power interface over USB.
tVCONNDischarge		230 ms	Time from VPD disconnect to vVCONNDischarge met.
vVCONNDischarge		800 mV	VCONN voltage after tVCONNDischarge .
iRaDetect		10 mA	The maximum current drawn from VCONN when the voltage is below vVCONNValid. Note: This current is below the 75 mW allowance for the first 500 ms at 5.5 V.
vRaReconnect	800 mV		Voltage at which the VPD shall reapply Ra on the falling edge of VCONN.
tRaWeaken		1.2 s	Time from VCONN exceeding vVCONNDisconnect until the VPD removes or weakens Ra .
vVCONNDisconnect	1 V	2.4 V	Threshold used to detect VCONN disconnect.

The **VPD shall** remove or weaken **Ra** within **tRaWeaken** (as defined in Table 4-8) after VCONN enters the valid voltage range (**vVCONNValid**).

The **VPD shall** reapply **Ra** when VCONN falls below **vRaReconnect** as defined in Table 4-8. The **VPD shall** discharge VCONN to below **vVCONNDischarge** within **tVCONNDischarge** on a cable disconnect. The **VPD shall** consider the VCONN capacitance present in the device when discharging VCONN.

4.5 Configuration Channel (CC)

4.5.1 Architectural Overview

For the USB Type-C solution, two pins on the connector, CC1 and CC2, are used to establish and manage the Source-to-Sink connection. When the device is connected through a hub, the connection between a Sink (UFP) on the hub and the Source (host port) and the connection between the Sink (device port) and a Source (DFP on the hub), are treated as separate connections. Functionally, the configuration channel is used to serve the following purposes:

- Detect attach of USB ports, e.g., a Source to a Sink,
- Resolve cable orientation and twist connections to establish USB data bus routing,
- Establish data roles between two attached ports,
- Discover and configure VBUS: **USB Type-C Current** modes or **USB Power Delivery**,
- Configure VCONN, and
- Discover and configure **optional** Alternate and Accessory modes.

4.5.1.1 USB Data Bus Interface and USB Type-C Plug Flip-ability

Since the USB Type-C plug can be inserted in either right-side-up or upside-down position, the hosts and devices that support USB data bus functionality must operate on the signal pins that are actually connected end-to-end. In the case of **USB 2.0**, this is done by shorting together the two D+ signal pins and the two D- signal pins in the host and device receptacles. In the case of **USB 3.2** SuperSpeed USB or **USB4** TX/RX signals in a single-lane implementation, it requires the functional equivalent of a switch in both the host and device to appropriately route the TX and RX signal pairs to the connected path through the cable. For a **USB 3.2** SuperSpeed USB or **USB4** dual-lane implementation, the host and/or device resolves the lane ordering.

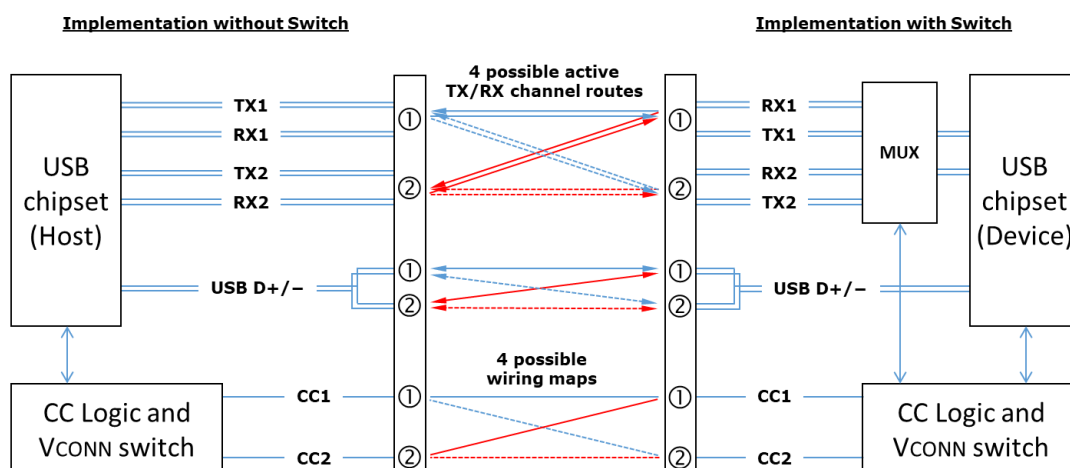
Figure 4-3 illustrates the logical data bus model for a USB Type-C-based Host connected to a USB Type-C-based Device that is only capable of SuperSpeed USB single-lane operation. The USB cable that sits between a host and device can be in one of four possible connected states when viewed by the host:

- Un-flipped straight through – Position ① ⇔ Position ①
- Un-flipped twisted through – Position ① ⇔ Position ②
- Flipped straight through – Position ② ⇔ Position ②
- Flipped twisted through – Position ② ⇔ Position ①

To establish the proper routing of the active USB data bus from host to device, the standard USB Type-C cable is wired such that a single CC wire is position aligned with the first TX/RX signal pairs (TXp1/TXn1 and RXp1/RXn1) – in this way, the CC wire and TX/RX data bus wires that are used for single-lane operational signaling within the cable track with regard to the orientation and twist of the cable. By being able to detect which of the CC pins (CC1 or CC2) at the receptacle is terminated by the device, the host is able to determine which TX/RX signals are to be used for the single-lane connection and the host can use this to control the functional switch for routing the TX/RX signal pairs. Similarly in the device, detecting which of the CC pins at the receptacle is terminated by the host allows the device to control the functional switch that routes its TX/RX signal pairs.

For a dual-lane implementation, the TX/RX signal pairs in the cable/plug aligned with the CC wire/pin is Lane 0 and in reference to **USB 3.2**, **shall** be identified as the Configuration Lane. The second TX/RX signal pairs (TXp2/TXn2 and RXp2/RXn2) in the cable/plug is Lane 1 of a dual-lane configuration.

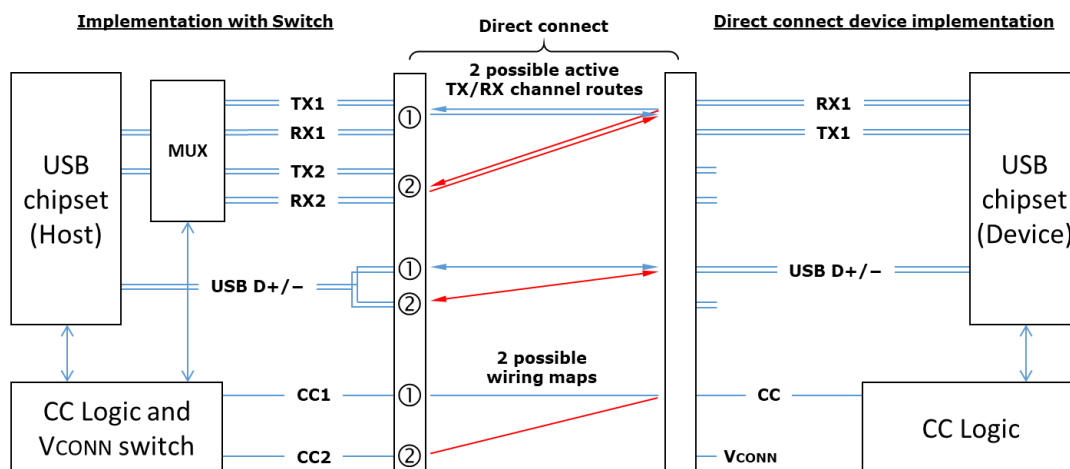
Figure 4-3 Logical Model for Single-Lane Data Bus Routing across USB Type-C-based Ports



While Figure 4-3 illustrates the functional model as a host connected to a device, this model equally applies to a USB hub's downstream ports as well.

Figure 4-4 illustrates the logical data bus model for a single-lane USB Type-C-based Device (implemented with a USB Type-C plug either physically incorporated into the device or permanently attached as a captive cable) connected directly to a USB Type-C-based Host. For the device, the location of the TX/RX data bus, **USB 2.0** data bus, CC and VCONN pins are fixed by design. Given that the device pin locations are fixed, only two possible connected states exist when viewed by the host.

Figure 4-4 Logical Model for USB Type-C-based Ports for a Single-Lane Direct Connect Device



The functional requirements for implementing TX/RX data bus routing for the USB Type-C receptacle are not included in the scope of this specification. There are multiple host, device and hub architectures that can be used to accomplish this which could include either discrete or integrated switching and could include merging this functionality with other **USB 3.2** or **USB4** design elements, e.g., a bus repeater.

The functional requirements for addressing SBU1 and SBU2 routing are not included in the scope of this specification. For **USB4**, where SBTX and SBRX are mapped to SBU1 and SBU2, the adjustment to the mapping of these signals based on the connection state (flipped and/or twisted) of the cable is defined by the **USB4** Specification (reference Sideband Channel Lane Reversal).

4.5.1.2 Connecting Sources and Sinks

Given that the USB Type-C receptacle and plug no longer differentiate host and device roles based on connector shape, e.g., as was the case with USB Type-A and Type-B connectors, any two ports that have USB Type-C receptacles can be connected together with a standard USB Type-C cable. Table 4-9 summarizes the expected results when interconnecting Source, Sink and DRP ports.

Table 4-9 USB Type-C Port Interoperability

	Source-Only	Sink-Only	DRP (Dual-Role-Power)
Source-Only	Non-functional	Functional	Functional
Sink-Only	Functional	Non-functional	Functional
DRP (Dual-Role-Power)	Functional	Functional	Functional ¹
Note 1: Resolution of roles <i>may</i> be automatic or manually driven.			

In the cases where no function results, neither port **shall** be harmed by this connection. The user has to independently realize the invalid combination and take appropriate action to resolve. While these two invalid combinations mimic traditional USB where host-to-host and device-to-device connections are not intended to work, the non-keyed USB Type-C solution does not prevent the user from attempting such interconnects. VBUS and VCONN **shall not** be applied by a Source (host) in these cases.

The typical flow for the configuration of the interface in the general USB case of a Source (Host) to a Sink (Device) is as follows:

1. Detect a valid connection between the ports (including determining cable orientation, Source/Sink and DFP/UEP relationship).
2. **Optionally** discover the cable's capabilities.
3. **Optionally** establish alternatives to traditional USB power (See Section 4.6.2).
 - a. **USB PD** communication over CC for advanced power delivery negotiation.
 - b. **USB Type-C Current** modes.
 - c. **USB BC 1.2**.
4. USB Device Enumeration.

For cases of Dual-Role-Power (DRP) ports connecting to either Source-only, Sink-only or another DRP, the process is essentially the same except that during the detecting a valid connection step, the DRP alternates between operating as a Source for detecting an attached Sink and presenting as a Sink to be detected by an attached Source. Ultimately this results in a Source-to-Sink connection.

4.5.1.2.1 Detecting a Valid Source-to-Sink Connection

The general concept for setting up a valid connection between a Source and Sink is based on being able to detect terminations residing in the product being attached.

To aid in defining the functional behavior of CC, a pull-up (**R_p**) and pull-down (**R_d**) termination model is used – actual implementation in hosts and devices *may* vary, for example, the pull-up termination could be replaced by a current source. Figure 4-5 and Figure 4-6 illustrates two models, the first based on a pull-up resistor in the Source and the second replacing this with a current source.

Figure 4-5 Pull-Up/Pull-Down CC Model

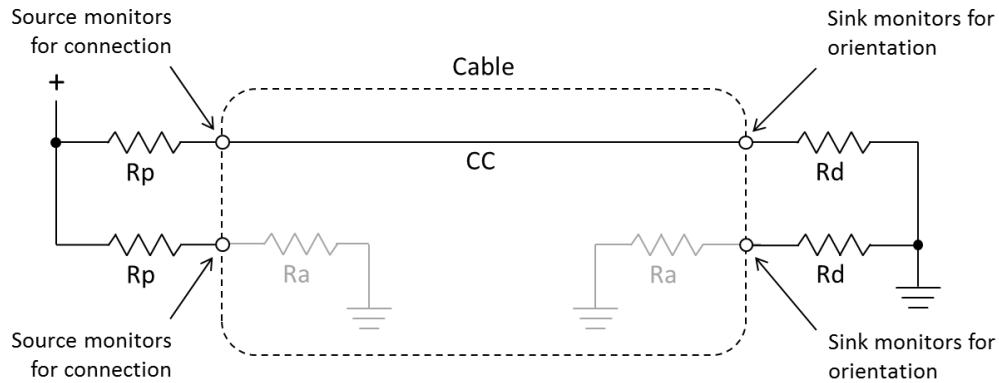
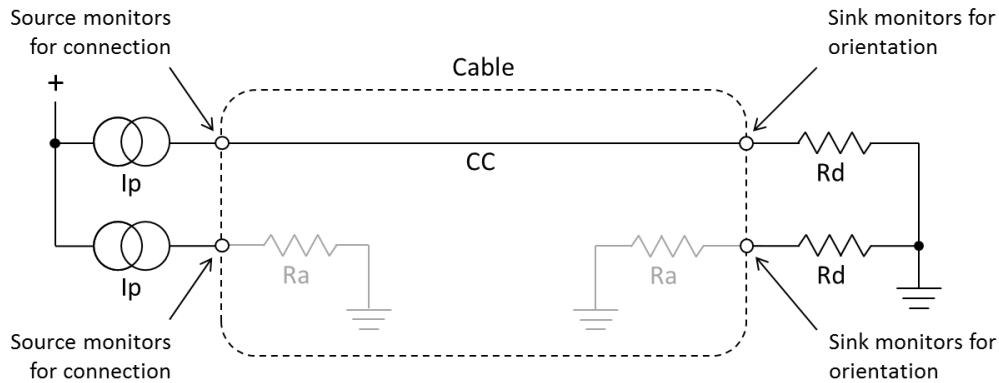


Figure 4-6 Current Source/Pull-Down CC Model



Initially, a Source exposes independent R_p terminations on its CC1 and CC2 pins, and a Sink exposes independent R_d terminations on its CC1 and CC2 pins, the Source-to-Sink combination of this circuit configuration represents a valid connection. To detect this, the Source monitors CC1 and CC2 for a voltage lower than its unterminated voltage – the choice of R_p is a function of the pull-up termination voltage and the Source's detection circuit. This indicates that either a Sink, a powered cable, or a Sink connected via a powered cable has been attached.

Prior to application of V_{CONN} , a powered cable exposes R_a on its V_{CONN} pin. R_a represents the load on V_{CONN} plus any resistive elements to ground. In some cable plugs it might be purely resistive and in others it *may* be simply the load.

The Source has to be able to differentiate between the presence of R_d and R_a to know whether there is a Sink attached and where to apply V_{CONN} . The Source is not required to source V_{CONN} unless R_a is detected.

Two special termination combinations on the CC pins as seen by a Source are defined for specific modes: R_a/R_a for *Liquid Corrosion Mitigation Mode* (Appendix A) and R_d/R_d for *Debug Accessory Mode* (Appendix B).

The Source uses de-bounce timers to reliably detect states on the CC pins to de-bounce the connection ($t_{CCDebounce}$) and hide *USB PD* BMC communications ($t_{PDDebounce}$).

Table 4-10 summarizes the port state from the Source's perspective.

Table 4-10 Source Perspective

CC1	CC2	State	Position
Open	Open	Nothing attached	N/A
Rd	Open	Sink attached	①
Open	Rd		②
Open	Ra	Powered cable without Sink attached or for <i>Liquid Corrosion Mitigation Mode</i> attached (Appendix A)	①
Ra	Open		②
Rd	Ra	Powered cable with Sink, <i>VCONN-Powered Accessory (VPA)</i> , or <i>VCONN-Powered USB Device (VPD)</i> attached	①
Ra	Rd		②
Rd	Rd	<i>Debug Accessory Mode</i> attached (Appendix B)	N/A
Ra	Ra	<i>Liquid Corrosion Mitigation Mode</i> attached (Appendix A)	N/A

Once the Sink is powered, the Sink monitors CC1 and CC2 for a voltage greater than its local ground. The CC pin that is at a higher voltage (i.e., pulled up by *Rp* in the Source) indicates the orientation of the plug.

Table 4-11 summarizes the typical behaviors for simple Sources (Hosts) and Sinks (Devices) for each state in Table 4-10.

Table 4-11 Source (Host) and Sink (Device) Behaviors by State

State	Source Behavior	Sink Behavior
Nothing attached	<ul style="list-style-type: none"> • Sense CC pins for attach • Do not apply VBUS or VCONN 	<ul style="list-style-type: none"> • Sense VBUS for attach
Sink attached	<ul style="list-style-type: none"> • Sense CC for orientation • Sense CC for detach • Apply VBUS and VCONN 	<ul style="list-style-type: none"> • Sense CC pins for orientation • Sense loss of VBUS for detach
Powered cable without Sink attached	<ul style="list-style-type: none"> • Sense CC pins for attach • Do not apply VBUS or VCONN 	<ul style="list-style-type: none"> • Sense VBUS for attach
Powered cable with Sink, <i>VCONN-Powered Accessory (VPA)</i> , or <i>VCONN-Powered USB Device (VPD)</i> attached	<ul style="list-style-type: none"> • Sense CC for orientation • Sense CC for detach • Apply VBUS and VCONN • Detect <i>VPD</i> and remove VBUS 	<ul style="list-style-type: none"> • If accessories or <i>VPDs</i> are supported, see Source Behavior with exception that VBUS is not applied, otherwise, N/A.
<i>Debug Accessory Mode</i> attached	<ul style="list-style-type: none"> • Sense CC pins for detach • Reconfigure for debug 	<ul style="list-style-type: none"> • Sense VBUS for detach • Reconfigure for debug
<i>Liquid Corrosion Mitigation Mode</i> attached	<ul style="list-style-type: none"> • Sense CC pins for detach • Reconfigure for analog audio 	<ul style="list-style-type: none"> • If accessories are supported, see Source Behavior, otherwise, N/A

Figure 4-3 shows how the inserted plug orientation is detected at the Source's receptacle by noting on which of the two CC pins in the receptacle an *Rd* termination is sensed. Now that the Source (Host) has recognized

that a Sink (Device) is attached and the plug orientation is determined, it configures the TX/RX data bus routing to the receptacle.

The Source (Host) then turns on VBUS. For the CC pin that does not connect Source-to-Sink through the cable, the Source supplies VCONN and **may** remove the termination. With the Sink (Device) now powered, it configures the USB data path. This completes the Host-to-Device connection.

The Source monitors the CC wire for the loss of pull-down termination to detect detach. If the Sink is removed, the Source port removes any voltage applied to VBUS and VCONN, resets its interface configuration and resumes looking for a new Sink attach.

Once a valid Source-to-Sink connection is established, alternatives to traditional USB power (VBUS as defined by either **USB 2.0** or **USB 3.2** specifications) **may** be available depending on the capabilities of the host and device. These include **USB Type-C Current**, **USB Battery Charging 1.2**, and **USB Power Delivery**.

In the case where **USB PD PR_Swap** is used to swap the Source and Sink of VBUS, the supplier of VCONN remains unchanged during and after the VBUS power swap. The new Source monitors the CC wire and the new Sink monitors VBUS to detect detach. When a detach event is detected, any voltages applied to VBUS and VCONN are removed, each port resets its interface configuration and resumes looking for an attach event.

In the case where **USB PD DR_Swap** is used to swap the data roles (DFP and UFP), the source of VBUS and VCONN do not change after the data role swap.

In the case where **USB PD VCONN_Swap** is used to swap the VCONN source, the VBUS Source/Sink and DFP/UFP roles are maintained during and after the VCONN swap.

The last step in the normal USB Type-C connect process is for the USB device to be attached and enumerated per standard **USB 2.0** and **USB 3.2** processes.

4.5.1.3 Configuration Channel Functional Models

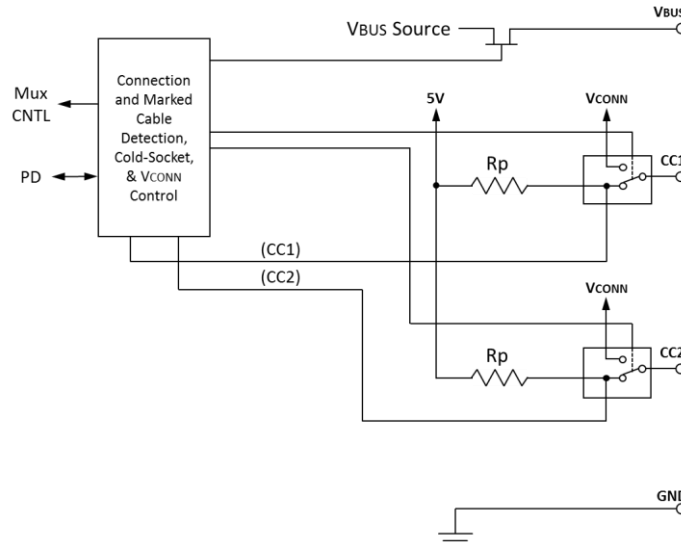
The functional models for the configuration channel behavior based on the CC1 and CC2 pins are described in this section for each port type: Source, Sink and Dual-Role-Power (DRP).

The figures in the following sections illustrate the CC1 and CC2 routing after the CC detection process is complete. In these figures, VBUS and VCONN **may** or **may not** actually be available.

4.5.1.3.1 Source Configuration Channel Functional Model

Figure 4-7 illustrates the functional model for CC1 and CC2 for a Source port prior to attach. This illustration includes consideration for **USB PD**.

Figure 4-7 Source Functional Model for CC1 and CC2

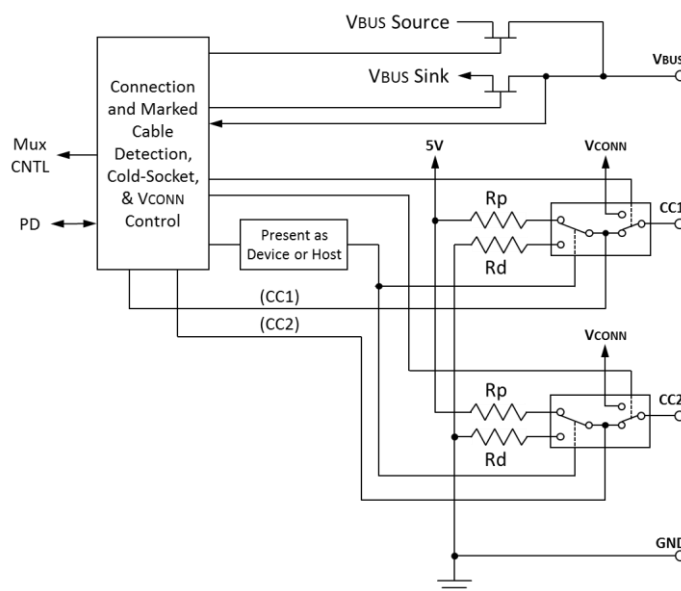


Referring to Figure 4-7, a port that behaves as a Source has the following functional characteristics:

1. The Source uses a FET to enable/disable power delivery across VBUS and initially the Source has VBUS disabled.
2. The Source supplies pull-up resistors (**R_p**) on CC1 and CC2 and monitors both to detect a Sink. The presence of an **R_d** pull-down resistor on either pin indicates that a Sink is being attached. The value of **R_p** indicates the initial **USB Type-C Current** level supported by the host.
3. The Source can **optionally** clamp the voltage on either of its CC pins. The minimum clamping voltage **shall** be **vCC-Clamp**. The clamp is intended to protect the Source circuitry associated with CC functionality.
4. The Source uses the CC pin pull-down characteristic to detect and establish the correct routing for the USB TX/RX data path and determine which CC pin is intended for supplying VCONN.
5. Once a Sink is detected, the Source enables VBUS and VCONN.
6. The Source can dynamically adjust the value of **R_p** to indicate a change in available **USB Type-C Current** to a Sink.
7. The Source monitors the continued presence of **R_d** to detect Sink detach. When a detach event is detected, the Source removes, if supplied, VBUS and VCONN, and returns to step 2.
8. If the Source supports advanced functions (e.g., **USB Power Delivery**, **USB4**, and **Alternate Modes**), **USB PD** communication is required.

Figure 4-8 illustrates the functional model for CC1 and CC2 for a Source that supports **USB PD PR_Swap**.

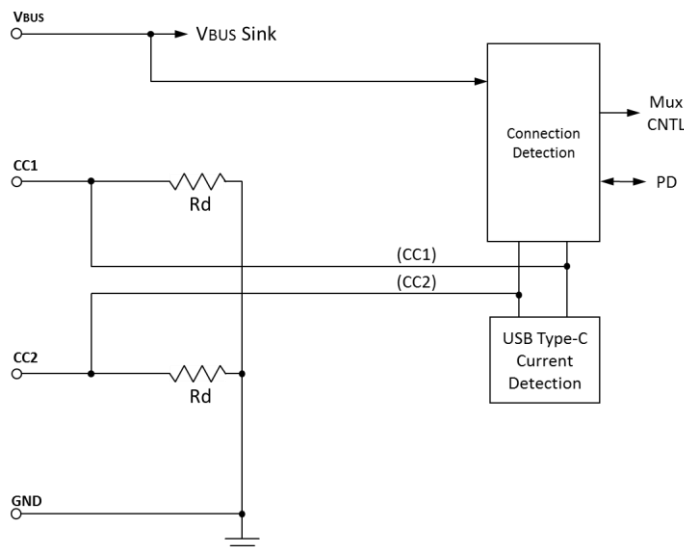
Figure 4-8 Source Functional Model Supporting USB PD PR_Swap



4.5.1.3.2 Sink Configuration Channel Functional Model

Figure 4-9 illustrates the functional model for CC1 and CC2 for a Sink. This illustration includes consideration for both **USB Type-C Current** and **USB PD**.

Figure 4-9 Sink Functional Model for CC1 and CC2



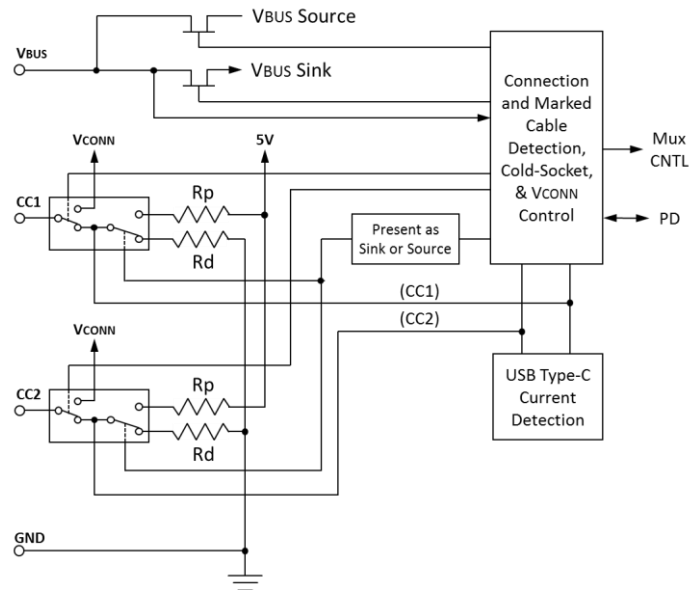
Referring to Figure 4-9, a port that behaves as a Sink has the following functional characteristics:

1. The Sink terminates both CC1 and CC2 to GND using pull-down resistors.
2. The Sink determines that a Source is attached by the presence of power on VBUS.
3. The Sink uses the CC pin pull-up characteristic to detect and establish the correct routing for the USB TX/RX data path.

4. The Sink can **optionally** monitor CC to detect an available higher **USB Type-C Current** from the Source. The Sink **shall** manage its load to stay within the detected Source current limit.
5. The Sink can **optionally** clamp the voltage on either of its CC pins. The minimum clamping voltage **shall** be **vCC-Clamp**. The clamp is intended to protect the Sink circuitry associated with CC functionality.
6. If the Sink supports advanced functions (e.g., **USB Power Delivery**, **USB4**, and **Alternate Modes**), **USB PD** communication is required.

Figure 4-10 illustrates the functional model for CC1 and CC2 for a Sink that supports **USB PD PR_Swap** and supports **USB PD VCONN_Swap** prior to attach.

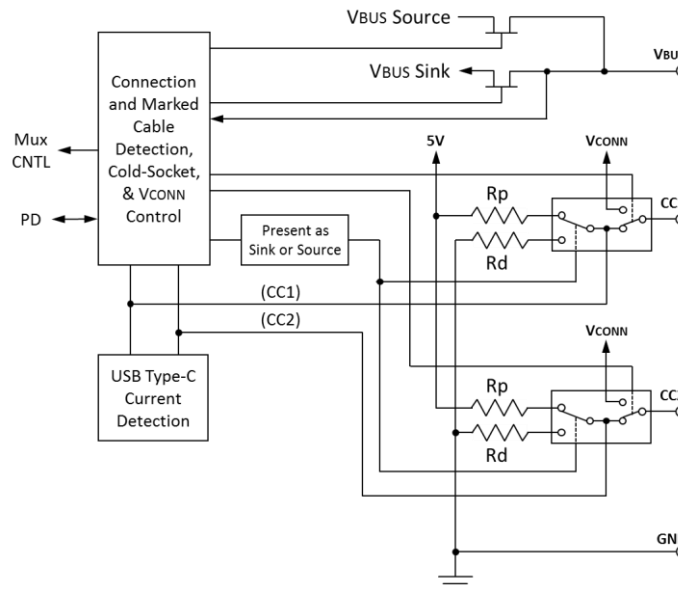
Figure 4-10 Sink Functional Model Supporting USB PD PR_Swap and VCONN_Swap



4.5.1.3.3 Dual-Role-Power (DRP) Configuration Channel Functional Model

Figure 4-11 illustrates the functional model for CC1 and CC2 for a DRP presenting as a Source prior to attach. This illustration includes consideration for both the **USB Type-C Current** and the **USB PD** features.

Figure 4-11 DRP Functional Model for CC1 and CC2



Referring to Figure 4-11, a port that can alternate between DFP and UFP behaviors has the following functional characteristics:

1. The DRP uses a FET to enable/disable power delivery across VBUS and initially when in Source mode has VBUS disabled.
2. The DRP uses switches for presenting as a Source or Sink.
3. The DRP has logic used during initial attach to toggle between Source and Sink operation:
 - a. Until a specific stable state is established, the DRP alternates between exposing itself as a Source and Sink. The timing of this process is dictated by a period (***tDRP***), percentage of time that a DRP exposes ***Rp*** (***dcSRC.DRP***) and role transition time (***tDRPTransition***).
 - b. When the DRP is presenting as a Source, it follows Source operation to detect an attached Sink – if a Sink is detected, it applies VBUS, VCONN, and continues to operate as a Source (e.g., cease alternating).
 - c. When the DRP is presenting as a Sink, it monitors VBUS to detect that it is attached to a Source – if a Source is detected, it continues to operate as a Sink (cease alternating).
4. If the DRP supports advanced functions (e.g., **USB Power Delivery**, **USB4**, and **Alternate Modes**), **USB PD** communication is required.

4.5.1.4 USB Type-C Port Power Roles and Role Swapping Mechanisms

USB Type-C ports on products (USB hosts, USB devices, USB chargers, etc.) can be generally characterized as implementing one of seven power role behavioral models:

- Source-only
- Source (Default) – strong preference toward being a Source but subsequently capable of becoming a Sink using **USB PD** swap mechanisms.
- Sink-only

- Sink (Default) – strong preference toward being a Sink but subsequently capable of becoming a Source using **USB PD** swap mechanisms.
- DRP: Toggling (Source/Sink)
- DRP: Sourcing Device
- DRP: Sinking Host

Two independent sets of swapping mechanisms are defined for USB Type-C port implementations, one based on role swapping within the initial state machine connection process and the other based on subsequent use of **USB PD**-based swapping mechanisms.

4.5.1.4.1 USB Type-C State-Machine-Based Role Swapping

During the initial USB Type-C state machine connection process, the products being connected end up in one of the two following roles associated with the termination of its port:

- **Rp** → VBUS and VCONN source and behaving as a downstream facing port (USB Host)
- **Rd** → VBUS sink and behaving as an upstream facing port (USB Device)

A USB Type-C DRP-based product *may* incorporate either or both the **Try.SRC** and **Try.SNK** swap mechanisms to affect the resulting role. **Try.SRC** allows a DRP that has a policy-based preference to be a Source when connecting to another DRP to affect a transition from a destined Sink role to the Source role. Alternately, **Try.SNK** allows a DRP that has a policy-based preference to be a Sink when connecting to another DRP to affect a transition from a destined Source role to the Sink role. Connection timing and other factors are involved in this process as defined in the USB Type-C state machine operation (see Section 4.5.2). It is important to note that these mechanisms, **Try.SRC** and **Try.SNK**, can only be used once as part of the initial connection process. If both sides support **USB PD**, the appropriate roles *may* then be further refined or swapped as per the **USB PD** specification.

A USB Type-C DRP-based product that does not support **USB PD** *should* implement either **Try.SRC** or **Try.SNK** depending on its preference to ensure predictable power and data roles. For example, a small mobile device that prefers being a Sink or UFP *should* implement **Try.SNK**, so that when attached to a DRP system such as a laptop, the mobile device will always be the power sink. If the mobile device is connected to a Sink-only port partner, the **Try.SNK** method will fail, and the mobile device will end up in the Source role. If the mobile device is connected to a port partner that also implements **Try.SNK**, the mobile device will randomly end up in either a Source or Sink role. Similarly, a DRP host such as a desktop PC that doesn't consume power over the port *should* implement **Try.SRC** to ensure it always ends up being a DFP when attached to DRPs.

A USB Type-C DRP-based product, independent of if that product supports **USB PD**, *may* also implement either **Try.SRC** or **Try.SNK** in order to ensure a preferred data role when connecting to another DRP that doesn't support **USB PD**. If that product only supports one specific data role, DFP or UFP, it *should* implement either **Try.SRC** or **Try.SNK** as appropriate in order to ensure a usable data role when connecting to another DRP that doesn't support **USB PD**. In this latter case, the use of the Try mechanism to correctly align a product's data role *should* take precedence over the use of the Try mechanism to align on a preferred power role as the Try mechanism might be the only opportunity for the product to get into a useable functional role.

Table 4-12 summarizes the recommended implementation of **Try.SRC** and **Try.SNK** by USB Type-C dual-role ports with preferred power or data roles. For dual-role ports where only one data role is relevant to its functional purpose, that data role *should* be its preferred data role, e.g., a USB storage device would have a preferred data role of UFP on its port.

**Table 4-12 Recommended Implementation of Try.SRC and Try.SNK for
Dual-Role Ports with Preferred Roles**

Current Preferred Roles of the Dual-Role Port:	Recommendation
Source / No DR preference	Enable Try.SRC (to become Source)
Source / DFP	Enable Try.SRC
Source / UFP	Enable Try.SNK (to become UFP)
Sink / No DR preference	Enable Try.SNK (to become Sink)
Sink / DFP	Enable Try.SRC (to become DFP)
Sink / UFP	Enable Try.SNK
No PR preference / DFP	Enable Try.SRC (to become DFP)
No PR preference / UFP	Enable Try.SNK (to become UFP)

4.5.1.4.2 USB PD-based Power Role, Data Role and VCONN Swapping

Following the completion of the initial USB Type-C state machine connection process, products **may** use **USB PD**-based swapping mechanisms to command a change in power roles, data roles and which end of the cable will supply VCONN. These mechanisms are:

- **USB PD PR_Swap** : swaps Source (**Rp**) and Sink (**Rd**)
- **USB PD DR_Swap** : swaps DFP (host data) and UFP (device data) roles
- **USB PD VCONN_Swap** : swaps which port supplies VCONN

Table 4-13 summarizes the behaviors of a port in response to the three **USB PD** swap commands.

Table 4-13 USB PD Swapping Port Behavior Summary

	DFP/UFP Data Roles	Rp/Rd	VBUS Source/Sink	VCONN Source
PR_Swap	Unchanged	Swapped	Swapped	Unchanged
DR_Swap	Swapped	Unchanged	Unchanged	Unchanged
VCONN_Swap	Unchanged	Unchanged	Unchanged	Swapped ¹
Note 1: Swapping of VCONN Source Port				

A USB Type-C dual-role product that supports **USB PD** **should** use **USB PD PR_Swap** and **DR_Swap** to ensure that it gets into its preferred power and/or data roles. For example, a hub's UFP or a monitor's UFP that prefers being in the Source role **should** initiate **PR_Swap** if needed to switch from being in the Sink role to the preferred Source role. Similarly, a small mobile device that prefers the Sink role **should** initiate **PR_Swap** to switch from being in the Source role to the preferred Sink role.

Table 4-14 summarizes the recommended use of power or data role swaps by USB Type-C dual-role ports that have preferred power or data roles. For dual-role ports where only one data role is relevant to its functional purpose, that data role **should** be its preferred data role, e.g., the physical upstream port of a hub would have a preferred data role of UFP on that port.

Table 4-14 Use of Power and Data Role Swaps for Dual-Role Ports with Preferred Roles

Preferred Roles of the Dual-Role Port:	When initially connected to a:	
	Source / DFP	Sink / UFP
Source / No DR preference	Issue PR_Swap	
Source / DFP	Issue PR_Swap and DR_Swap	
Source / UFP	Issue PR_Swap	Issue DR_Swap
Sink / No DR preference		Issue PR_Swap
Sink / DFP	Issue DR_Swap	Issue PR_Swap
Sink / UFP		Issue PR_Swap and DR_Swap
No PR preference / DFP	Issue DR_Swap	
No PR preference / UFP		Issue DR_Swap

4.5.1.4.3 Power Role Behavioral Model Summary

Table 4-15 provides a summary of the defining characteristics of the seven fundamental power roles.

Table 4-15 Power Role Behavioral Model Summary

Power Role		Toggles	PR_Swap	USB Host	USB Device	DFP	UFP	DR_Swap	Try.SRC/ Try.SNK	Connects with
Source-Only		No	NA	Opt.	Opt. ¹	Req.	Opt.	Opt.	NA	Sink or DRP
Source (Default)		No	Req.	Opt.	Opt. ¹	Req.	Opt.	Opt.	NA	Sink or DRP
Sink-Only		No	NA	Opt. ¹	Opt.	Opt.	Req.	Opt.	NA	Source or DRP
Sink (Default)		No	Req.	Opt. ¹	Opt.	Opt.	Req.	Opt.	NA	Source or DRP
DRP	Toggling (Source/ Sink)	Req.	Req.	Opt.	Opt.	Req.	Req.	Req.	Opt.	Source, Sink or DRP
	Sourcing Device	Req.	Req.	NA	Req.	Req.	Req.	Req.	Opt.	
	Sinking Host	Req.	Req.	Req.	NA	Req.	Req.	Req.	Opt.	
Note 1: Requires use of <i>DR_Swap</i> .										

4.5.2 CC Functional and Behavioral Requirements

This section provides the functional and behavioral requirements for implementing CC. The first sub-section provides connection state diagrams that are the basis for the remaining sub-sections.

The terms Source (SRC) and Sink (SNK) used in this section refer to the port's power role while the terms DFP and UFP refer to the port's data role. A DRP (Dual-Role-Power) port is capable of acting as either a Source or Sink. Typically, Sources are found on hosts and supply VBUS while a Sink is found on a device and consumes power from VBUS. When a connection is initially made, the port's initial power state and data role are established. **USB PD** introduces three swap commands that **may** alter a port's power or data role:

- The **PR_Swap** command changes the port's power state as reflected in the following state machines. **PR_Swap** does not change the port sourcing VCONN.

- The **DR_Swap** command has no effect on the following state machines or VCONN as it only changes the port's data role.
- **VCONN_Swap** command changes the port sourcing VCONN. The **PR_Swap** command and **DR_Swap** command have no effect on the port sourcing VCONN.

Note: A **VCONN-Powered USB Device** that supports the **optional** Charge-Through capability, once detected via **USB PD** messaging, will also change the Host-side port's power state without changing the port sourcing VCONN.

Note: **USB PD** defines another **optional** swapping mechanism (**FR_Swap**) that is used in a special case where a user interaction could inadvertently trigger a need to change the source of VBUS. A variant of **PR_Swap**, **FR_Swap** similarly swaps Source (**Rp**) and Sink (**Rd**) between two connected ports. For purposes of this specification, only **PR_Swap** is explicitly considered in the behavior requirements and implementations that support **FR_Swap should**, where applicable, apply **PR_Swap**-related behaviors to **FR_Swap**. See the **USB PD** specification for further details regarding **FR_Swap**.

The connection state diagrams and CC behavior descriptions in this section describe the behavior of receptacle-based ports. The plug on a direct connect device or a device with a captive cable **shall** behave as a plug on a cable that is attached at its other end in normal orientation to a receptacle. These devices **shall** apply and sense CC voltage levels on pin A5 only and pin B5 **shall** have an impedance above **zOPEN**, unless it is a **VCONN-Powered Accessory**, in which case B5 **shall** have an impedance **Ra**.

4.5.2.1 Connection State Diagrams

This section provides reference connection state diagrams for CC-based behaviors.

Refer to Section 4.5.2.2 for the specific state transition requirements related to each state shown in the diagrams.

Refer to Section 4.5.2.6 for a description of which states are mandatory for each port type, and a list of states where **USB PD** communication is permitted.

Figure 4-12 illustrates a connection state diagram for a Source (Host/Hub DFP).

Figure 4-12 Connection State Diagram: Source

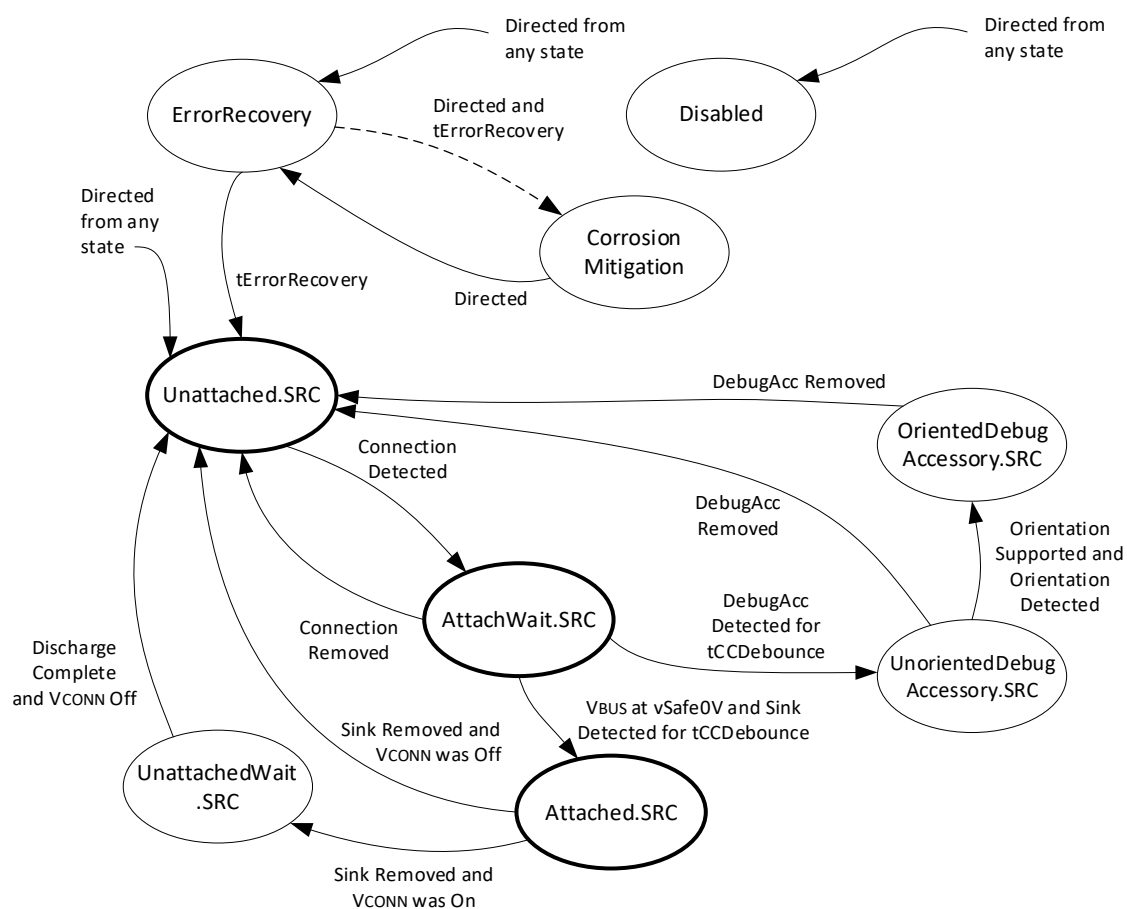


Figure 4-13 illustrates a connection state diagram for a simple Sink (Device/Hub UFP).

Figure 4-13 Connection State Diagram: Sink

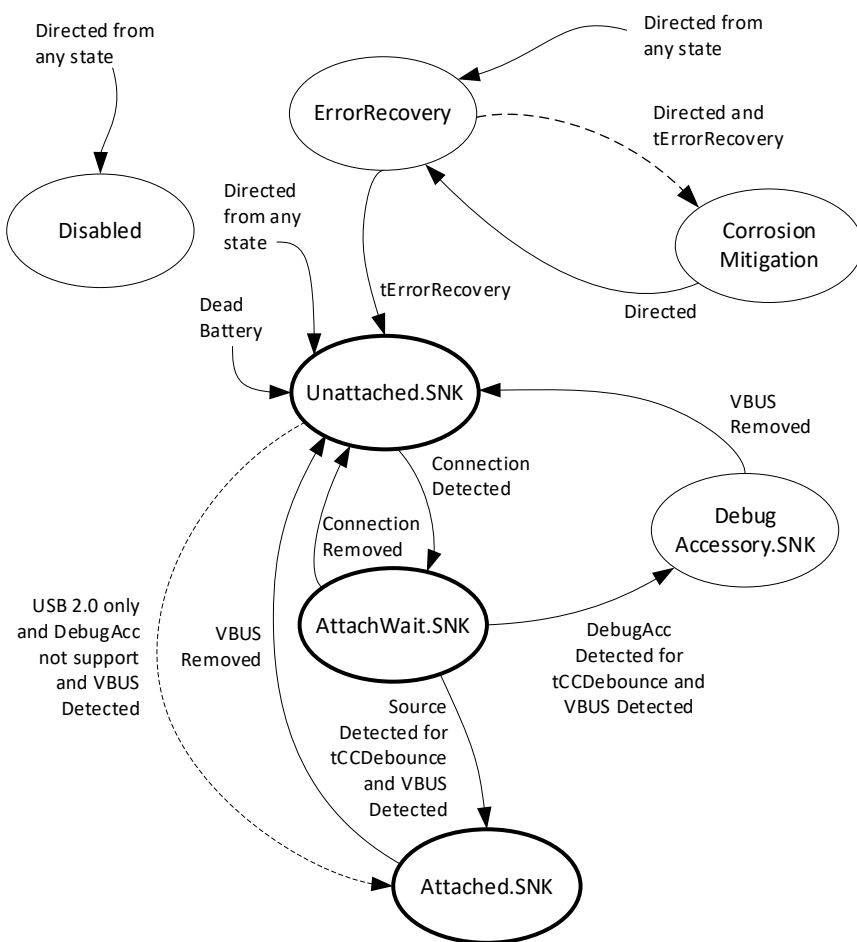


Figure 4-14 Connection State Diagram: Sink with Accessory Support

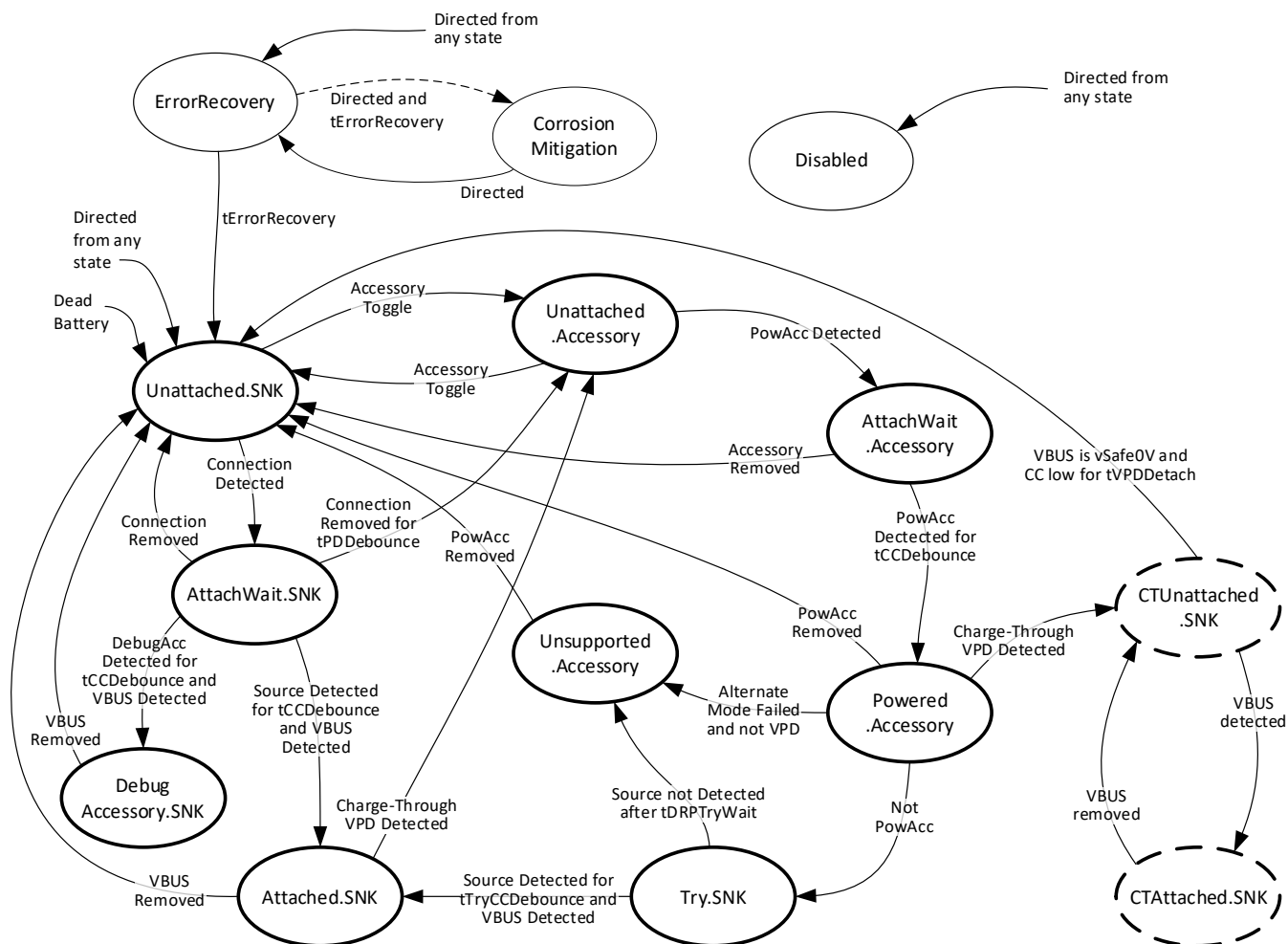


Figure 4-15 illustrates a connection state diagram for a simple DRP (Dual-Role-Power) port.

Figure 4-15 Connection State Diagram: DRP

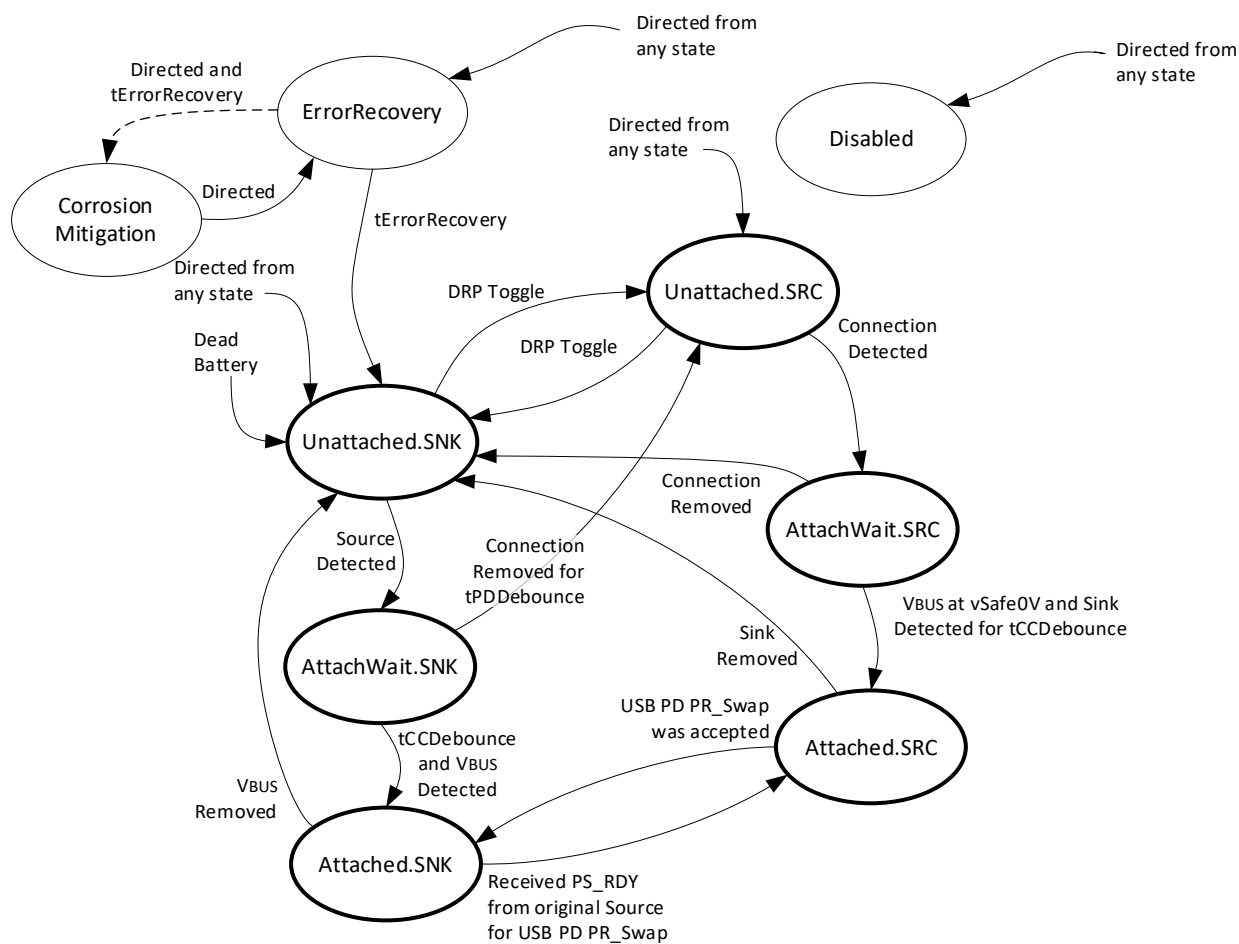


Figure 4-16 illustrates a connection state diagram for a DRP that supports **Try.SRC** and Accessory Modes.

Figure 4-16 Connection State Diagram: DRP with Accessory and Try.SRC Support

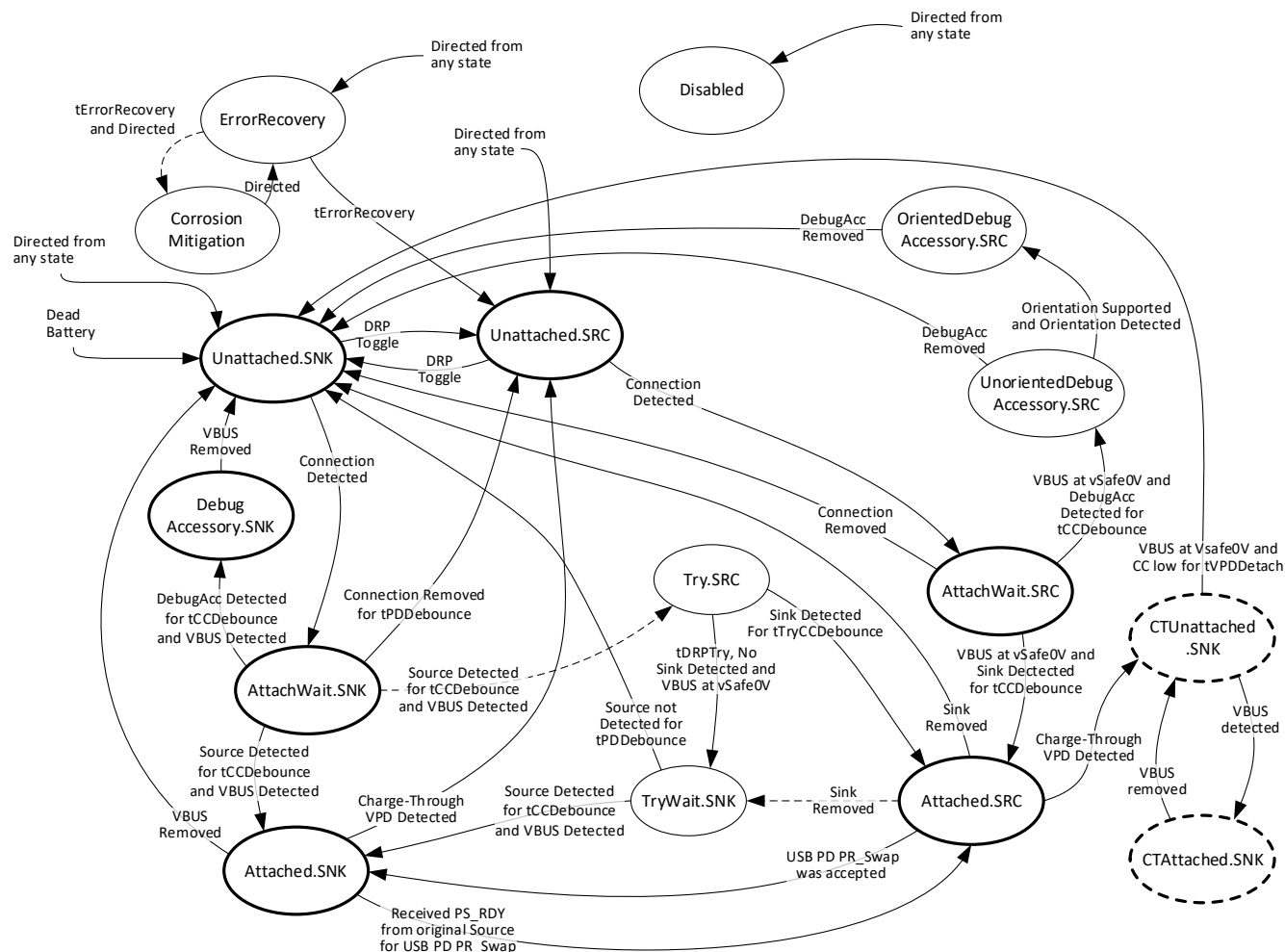


Figure 4-17 Connection State Diagram: DRP with Accessory and Try.SNK Support

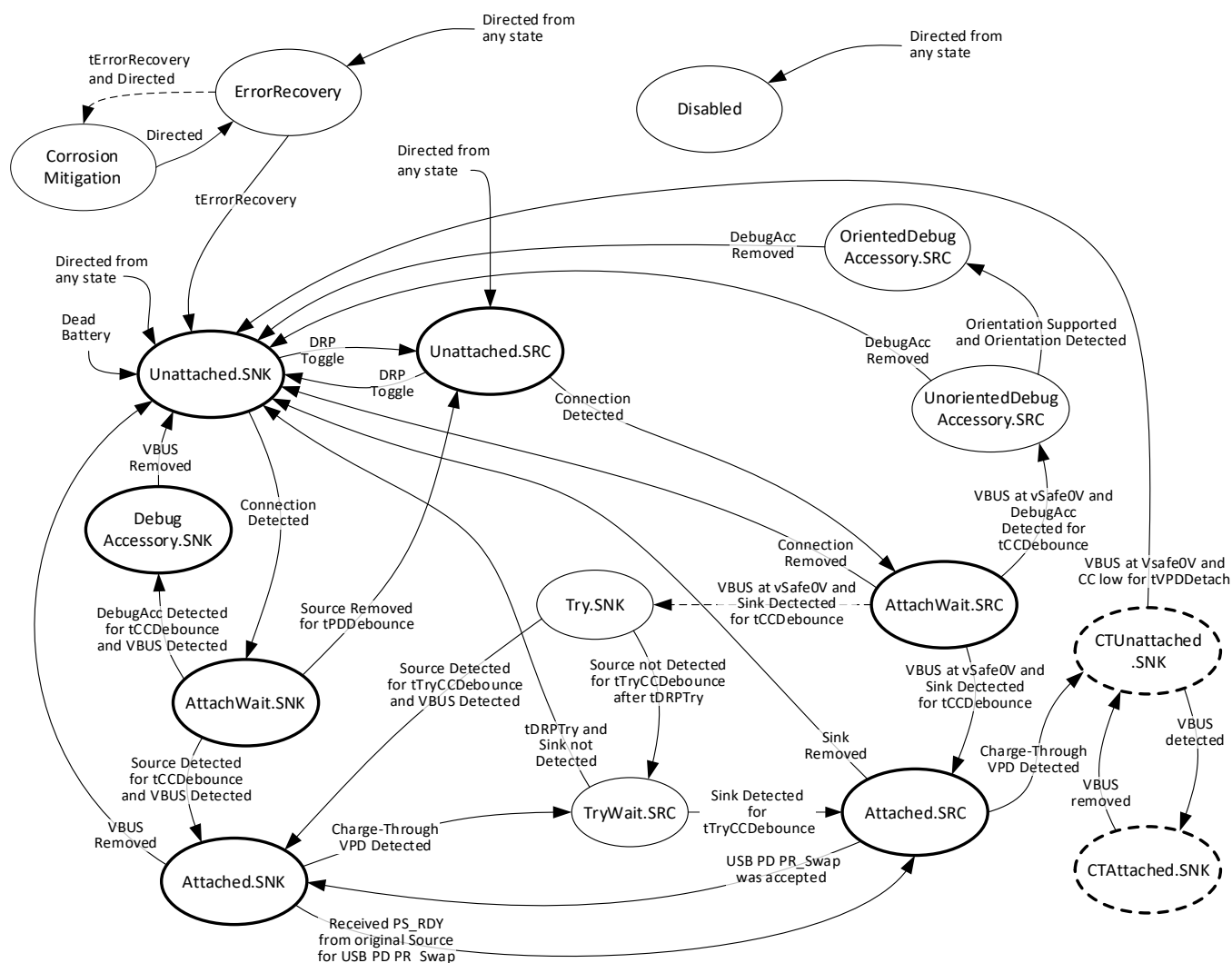
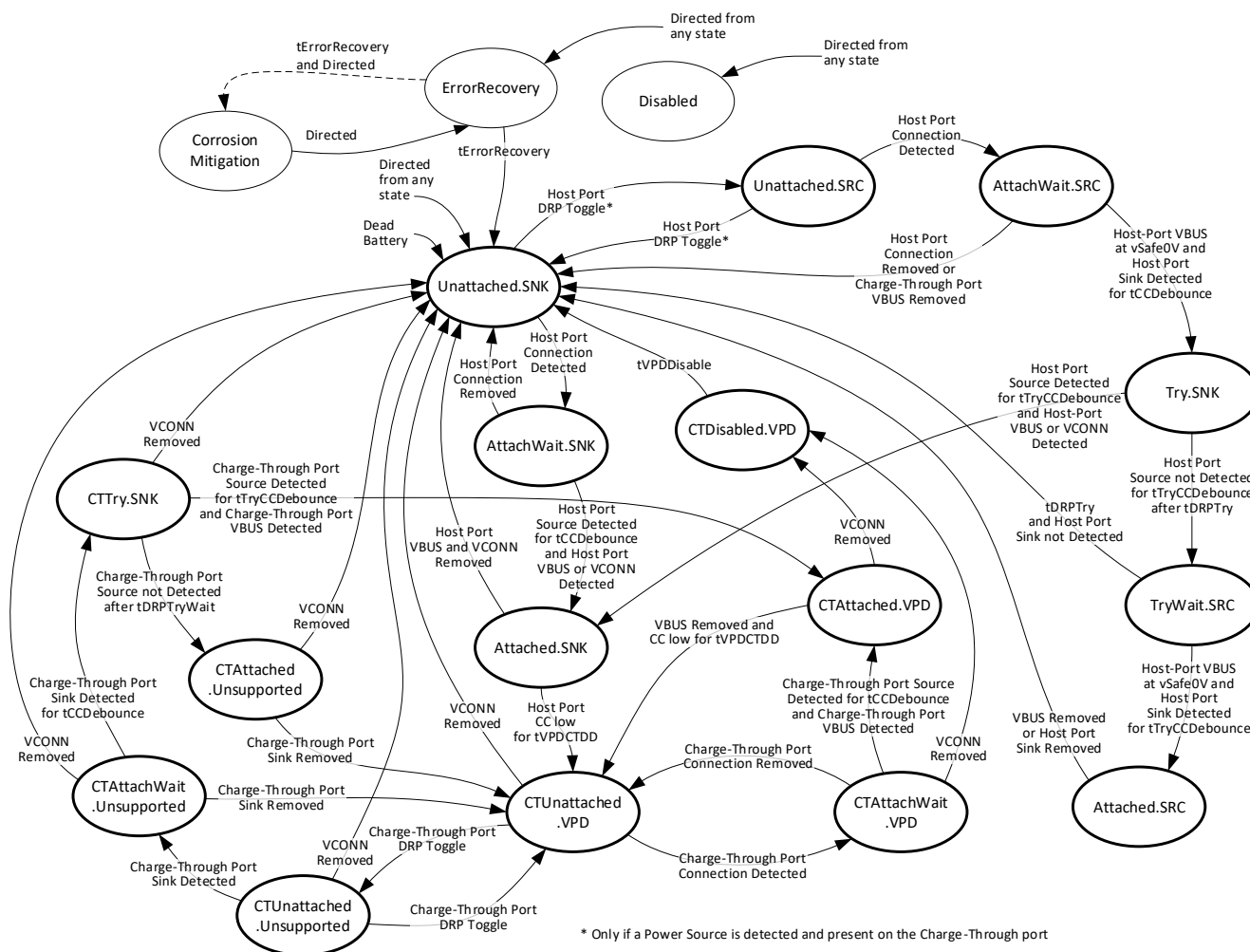


Figure 4-18 illustrates a connection state diagram for a Charge-Through *VCONN-Powered USB Device*.

Figure 4-18 Connection State Diagram: Charge-Through VPD



4.5.2.2 Connection State Machine Requirements

Entry into any unattached state when “directed from any state” **shall not** be used to override *tDRP* toggle.

A *DRP* or a Sink **may** consume default power from *VBUS* in any state where it is not required to provide *VBUS*.

The following two tables define the electrical states for a *CC* pin in both a Source and a Sink. Every port has *CC1* and *CC2* pins, each with its own individual *CC* pin state. The combination of a port’s *CC1* and *CC2* pin states are used to define the conditions under which a port transitions from one state to another.

Table 4-16 Source Port CC Pin State

CC Pin State	Port Partner CC Termination	Voltage Detected on CC when port asserts Rp
<i>SRC.Open</i>	Open, Rp	Above maximum <i>vRd</i>
<i>SRC.Rd</i>	Rd	Within the <i>vRd</i> range (i.e., between minimum <i>vRd</i> and maximum <i>vRd</i>)
<i>SRC.Ra</i>	Ra	Below minimum <i>vRd</i>
Note: See Section 4.11.3.1.2 for more on CC voltage detection thresholds for a Source port.		

Table 4-17 Sink Port CC Pin State

CC Pin State	Port Partner CC Termination	Voltage Detected on CC when port asserts Rd
<i>SNK.Rp</i>	Rp	Above minimum <i>vRd-Connected</i>
<i>SNK.Open</i>	Open, Ra, Rd	Below minimum <i>vRd-Connected</i>
Note: See Section 4.11.3.1.1 for more on CC voltage detection thresholds for a Sink port.		

4.5.2.2.1 Disabled State

This state appears in Figure 4-12, Figure 4-13, Figure 4-14, Figure 4-15, Figure 4-16, Figure 4-17 and Figure 4-18.

The *Disabled* state is where the port prevents connection from occurring by removing all terminations from the CC pins.

The port *should* transition to the *Disabled* state from any other state when directed. When the port transitions to the *Disabled* state from *Attached.SNK*, it *shall* keep all terminations on the CC pins removed for a minimum of *tErrorRecovery*.

A port *may* choose not to support the *Disabled* state. If the *Disabled* state is not supported, the port *shall* be directed to either the *Unattached.SNK* or *Unattached.SRC* states after power-on.

4.5.2.2.1.1 Disabled State Requirements

The port *shall not* drive VBUS or VCONN and *shall* present a high-impedance to ground (above *zOPEN*) on its CC1 and CC2 pins.

4.5.2.2.1.2 Exiting from Disabled State

A Sink *shall* transition to *Unattached.SNK* when directed.

A Source *shall* transition to *Unattached.SRC* when directed.

A DRP *shall* transition to either *Unattached.SNK* or *Unattached.SRC* when directed.

4.5.2.2.2 ErrorRecovery State

This state appears in Figure 4-12, Figure 4-13, Figure 4-14, Figure 4-15, Figure 4-16, Figure 4-17 and Figure 4-18.

The **ErrorRecovery** state is where the port removes the terminations from the CC1 and CC2 pins for **tErrorRecovery** followed by transitioning to the appropriate **Unattached.SNK** or **Unattached.SRC** state based on port type. This is the equivalent of forcing a detach event and looking for a new attach. The port may alternately transition from **ErrorRecovery** to **CorrosionMitigation** if directed.

Ports that support **USB Power Delivery** **shall** support the **ErrorRecovery** state.

Ports that support the **ErrorRecovery** state **shall** transition to the **ErrorRecovery** state from any other state when directed.

A port that does not support **USB Power Delivery** **may** choose not to support the **ErrorRecovery** state. If the **ErrorRecovery** state is not supported, the port **shall** be directed to the **Disabled** state if supported. If the **Disabled** state is not supported, the port **shall** be directed to either the **Unattached.SNK** or **Unattached.SRC** states.

4.5.2.2.2.1 ErrorRecovery State Requirements

The port **shall not** drive VBUS or VCONN AND **shall** present a high-impedance to ground (above **zOPEN**) on its CC1 and CC2 pins.

4.5.2.2.2.2 Exiting from ErrorRecovery State

A Sink **shall** transition to **Unattached.SNK** after **tErrorRecovery** or **shall** transition to **CorrosionMitigation** after **tErrorRecovery** if directed.

A Source **shall** transition to **Unattached.SRC** after **tErrorRecovery** or **shall** transition to **CorrosionMitigation** after **tErrorRecovery** if directed.

A DRP (Figure 4-15) and a DRP with Accessory and **Try.SNK** Support (Figure 4-17) **shall** transition to **Unattached.SNK** after **tErrorRecovery** or **shall** transition to **CorrosionMitigation** after **tErrorRecovery** if directed.

A DRP with Accessory and **Try.SRC** Support (Figure 4-16) **shall** transition to **Unattached.SRC** after **tErrorRecovery** or **shall** transition to **CorrosionMitigation** after **tErrorRecovery** if directed.

A Charge-Through **VCONN-Powered USB Device** (Figure 4-18) **shall** transition to **Unattached.SNK** after **tErrorRecovery** or **shall** transition to **CorrosionMitigation** after **tErrorRecovery** if directed.

4.5.2.2.3 Unattached.SNK State

This state appears in Figure 4-13, Figure 4-14, Figure 4-15, Figure 4-16, Figure 4-17 and Figure 4-18.

When in the **Unattached.SNK** state, the port is waiting to detect the presence of a Source.

A port with a dead battery **shall** enter this state while unpowered.

4.5.2.2.3.1 Unattached.SNK State Requirements

The port **shall not** drive VBUS or VCONN.

Both CC1 and CC2 pins **shall** be independently terminated to ground through **Rd**.

A Charge-Through **VCONN-Powered USB Device** **shall** isolate its Host-side port from its Charge-Through port, including CCs and VBUS, and independently terminate its Charge-Through port's CC1 and CC2 pins and Host-side port's CC pin to ground through **Rd**.

4.5.2.2.3.2 Exiting from Unattached.SNK State

If the port supports **USB PD** or accessories, the port **shall** transition to **AttachWait.SNK** when the **SNK.Rp** state is present on at least one of its CC pins.

The maximum times that a Port **shall** take to transition to **AttachWait.SNK** are the following:

- **tNoToggleConnect** when neither Port Partner is toggling
- **tOnePortToggleConnect** when one Port Partner only is toggling

When both Port Partners are toggling, a Port **should** transition to **AttachWait.SNK** within **tTwoPortToggleConnect**. When both Port Partners are DRPs it is indeterminate whether the local port will transition to **AttachWait.SRC** or **AttachWait.SNK**.

Note: The times **tOnePortToggleConnect** and **tTwoPortToggleConnect** relate to how long toggling ports **may** take to sync and detect a connection.

A **USB 2.0** only Sink that doesn't support accessories and is self-powered or requires only default power and does not support **USB PD** **may** transition directly to **Attached.SNK** when VBUS is detected.

A DRP **shall** transition to **Unattached.SRC** within **tDRPTransition** after the state of both CC pins is **SNK.Open** for $tDRP - dcSRC.DRP \cdot tDRP$, or if directed.

A Sink with Accessory support **shall** transition to **Unattached.Accessory** within **tDRPTransition** after the state of both the CC1 and CC2 pins is **SNK.Open** for $tDRP - dcSRC.DRP \cdot tDRP$, or if directed.

A Charge-Through **VCONN-Powered USB Device** **shall** transition to **Unattached.SRC** within **tDRPTransition** after the state of the Host-side port's CC pin is **SNK.Open** for $tDRP - dcSRC.DRP \cdot tDRP$ and both of the following is detected on the Charge-Through port.

- **SNK.Rp** state is detected on exactly one of the CC1 or CC2 pins for at least **tCCDebounce**
- VBUS is detected

A Charge-Through **VCONN-Powered USB Device** **shall** transition to **Attached.SNK** when a Source connection is detected, as indicated by the **SNK.Rp** state on its Host-side port's CC pin.

4.5.2.2.4 AttachWait.SNK State

This state appears in Figure 4-13, Figure 4-14, Figure 4-15, Figure 4-16, Figure 4-17 and Figure 4-18.

When in the **AttachWait.SNK** state, the port has detected the **SNK.Rp** state on at least one of its CC pins and is waiting for VBUS.

When in the **AttachWait.SNK** state, the Charge-Through **VCONN-Powered USB Device** has detected the **SNK.Rp** state on its Host-side port's CC pin and is waiting for host-side VBUS.

4.5.2.2.4.1 AttachWait.SNK State Requirements

The port **shall not** drive VBUS or VCONN.

Both the CC1 and CC2 pins **shall** be independently terminated to ground through **Rd**.

A Charge-Through **VCONN-Powered USB Device** **shall** isolate its Host-side port from its Charge-Through port, including CCs and VBUS, and independently terminate its Charge-Through port's CC1 and CC2 pins and Host-side port's CC pin to ground through **Rd**.

It is strongly recommended that a **USB 3.2** SuperSpeed device holds off VBUS detection to the device controller until the **Attached.SNK** state or the **DebugAccessory.SNK** state is reached, i.e., at least one CC pin is in the **SNK.Rp** state. Otherwise, it **may** connect as **USB 2.0** when attached to a legacy host or hub's DFP.

4.5.2.2.4.2 Exiting from AttachWait.SNK State

A Sink **shall** transition to **Unattached.SNK** when the state of both the CC1 and CC2 pins is **SNK.Open** for at least **tPDDebounce**.

A Sink with Accessory Support **shall** transition to **Unattached.Accessory** when the state of both the CC1 and CC2 pins is **SNK.Open** for at least **tPDDebounce**.

A DRP **shall** transition to **Unattached.SRC** when the state of both the CC1 and CC2 pins is **SNK.Open** for at least **tPDDebounce**.

The port **shall** transition to **Attached.SNK** after the state of only one of the CC1 or CC2 pins is **SNK.Rp** for at least **tCCDebounce** and VBUS is detected. Note the Source **may** initiate **USB PD** communications which will cause brief periods of the **SNK.Open** state on one of the CC pins with the state of the other CC pin remaining **SNK.Open**, but this event will not exceed **tPDDebounce**.

If the port is a **VCONN-Powered Accessory** or a **VCONN-Powered USB Device**, the port **shall** transition to **Attached.SNK** when either VCONN or VBUS is detected. The port **may** transition without waiting **tCCDebounce** on CC.

If the port supports **Debug Accessory Mode**, the port **shall** transition to **DebugAccessory.SNK** if the state of both the CC1 and CC2 pins is **SNK.Rp** for at least **tCCDebounce** and VBUS is detected. Note the DAM Source **may** initiate **USB PD** communications which will cause brief periods of the **SNK.Open** state on one of the CC pins with the state of the other CC pin remaining **SNK.Rp**, but this event will not exceed **tPDDebounce**.

A Charge-Through **VCONN-Powered USB Device** **shall** transition to **Attached.SNK** after the state of the Host-side port's CC pin is **SNK.Rp** for at least **tCCDebounce** and either host-side VCONN or VBUS is detected.

A DRP that strongly prefers the Source role **may optionally** transition to **Try.SRC** instead of **Attached.SNK** when the state of only one CC pin has been **SNK.Rp** for at least **tCCDebounce** and VBUS is detected.

4.5.2.2.5 Attached.SNK State

This state appears in Figure 4-13, Figure 4-14, Figure 4-15, Figure 4-16, Figure 4-17 and Figure 4-18.

When in the **Attached.SNK** state, the port is attached and operating as a Sink. When the port initially enters this state it is also operating as a UFP. The power and data roles can be changed using **USB PD** commands.

A port that entered this state directly from **Unattached.SNK** due to detecting VBUS **shall not** determine orientation or availability of higher than Default USB Power and **shall not** use **USB PD**.

4.5.2.2.5.1 Attached.SNK State Requirements

If the port needs to determine the orientation of the connector, it **shall** do so only upon entry to this state by detecting which of the CC1 or CC2 pins is connected through the cable (i.e., the CC pin that is in the **SNK.Rp** state).

If the port supports signaling on USB TX/RX pairs, it **shall** functionally connect the USB TX/RX pairs and maintain the connection during and after a **USB PD PR_Swap**.

If the port has entered the **Attached.SNK** state from the **AttachWait.SNK** or **TryWait.SNK** states, only one the CC1 or CC2 pins will be in the **SNK.Rp** state. The port **shall** continue to terminate this CC pin to ground through **Rd**.

If the port has entered the **Attached.SNK** state from the **Attached.SRC** state following a **USB PD PR_Swap**, the port **shall** terminate the connected CC pin to ground through **Rd**.

The port **shall** meet the **Sink Power Sub-State** requirements specified in Section 4.5.2.3.

If the port is a **VCONN-Powered USB Device**, it **shall** respond to **USB PD** cable identity queries on SOP'. It **shall not** send or respond to messages on SOP. It **shall** ensure there is sufficient capacitance on CC to meet cReceiver as defined in **USB PD**.

A Charge-Through **VCONN-Powered USB Device** **shall** isolate its Host-side port from its Charge-Through port, including CCs and VBUS, present a high-impedance to ground (above **zOPEN**) on its Charge-Through port's CC1 and CC2 pins and terminate its Host-side port's CC pin to ground through **Rd**.

A Charge-Through **VCONN-Powered USB Device** **shall** start a Charge-Through Support Timer when it enters the **Attached.SNK** state. If a Charge-Through **VCONN-Powered USB Device** fails to exit the **Attached.SNK** state before the Charge-Through Support Timer exceeds **tAMETimeout**, it **shall** present a **USB Billboard Device Class** interface indicating that it does not support Charge-Through.

A Charge-Through **VCONN-Powered USB Device** **shall** reset the Charge-Through Support Timer when it first receives any **USB PD** Structured VDM Command it supports. If a Charge-Through **VCONN-Powered USB Device** receives a Structured VDM Command multiple times, it **shall** only reset the Charge-Through Support Timer once. This ensures a Charge-Through **VCONN-Powered USB Device** will present a **USB Billboard Device Class** interface if it fails to exit **Attached.SNK** while receiving repeated or continuous Structured VDM Commands (e.g., **Discover Identity**).

A Charge-Through **VCONN-Powered USB Device** **shall** reset the Charge-Through Support Timer when it receives any Data Message it supports. A Charge-Through **VCONN-Powered USB Device** **shall** hold the Charge-Through Support Timer in reset while it is in any **USB PD** BIST mode.

Except for a **VCONN-Powered USB Device** or Charge-Through **VCONN-Powered USB Device**, the port **may** negotiate a **USB PD PR_Swap**, **DR_Swap** or **VCONN_Swap**.

If the port supports Charge-Through **VCONN-Powered USB Device**, and an explicit **USB PD** contract has failed to be negotiated, the port **shall** query the identity of the cable via **USB PD** on SOP'.

By default, upon entry from **AttachWait.SNK** or **Unattached.SNK**, VCONN **shall not** be supplied in the **Attached.SNK** state. If **Attached.SNK** is entered from **Attached.SRC** as a result of a **USB PD PR_Swap**, it **shall** maintain VCONN supply state, whether on or off, and its data role/connections. A **USB PD DR_Swap** has no effect on which port sources VCONN.

The port **may** negotiate a **USB PD VCONN_Swap**. When the port successfully executes **USB PD VCONN_Swap** operation and is not sourcing VCONN, it **shall** start sourcing VCONN within **tVCONNON**. The port **shall** execute the **VCONN_Swap** in a make-before-break sequence in order to keep active USB Type-C to USB Type-C cables powered. When the port successfully executes **USB PD VCONN_Swap** operation and was sourcing VCONN, it **shall** stop sourcing VCONN within **tVCONNOFF**.

4.5.2.2.5.2 Exiting from Attached.SNK State

A port that is not a **VCONN-Powered USB Device** and is not in the process of a **USB PD PR_Swap** or a **USB PD** Hard Reset or a **USB PD FR_Swap** **shall** transition to **Unattached.SNK** within **tSinkDisconnect** when VBUS falls below **vSinkDisconnect** for VBUS operating at or below 5 V or below **vSinkDisconnectPD** when negotiated by **USB PD** to operate above 5 V.

A **VCONN-Powered USB Device** **shall** return to **Unattached.SNK** when VBUS has fallen below **vSinkDisconnect** and VCONN has fallen below **vVCONNDisconnect**.

A port that has entered into **USB PD** communications with the Source and has seen the CC voltage exceed **vRd-USB** *may* monitor the CC pin to detect cable disconnect in addition to monitoring VBUS.

A port that is monitoring the CC voltage for disconnect (but is not in the process of a **USB PD PR_Swap** or **USB PD FR_Swap**) *shall* transition to **Unattached.SNK** within **tSinkDisconnect** after the CC voltage remains below **vRd-USB** for **tPDDebounce**.

If supplying VCONN, the port *shall* cease to supply it within **tVCONNOff** of exiting **Attached.SNK**.

A Charge-Through **VCONN-Powered USB Device** *shall* transition to **CTUnattached.VPD** if VCONN is present and the state of its Host-side port's CC pin is **SNK.Open** for **tVPDCTDD**.

A port that via SOP' has detected an attached Charge-Through **VCONN-Powered USB Device** *shall* transition to **TryWait.SRC** if implemented, or transition to **Unattached.SRC** or **Unattached.Accessory** if **TryWait.SRC** is not supported. This transition *may* be delayed until the device has sufficient battery charge needed to remain powered until it reaches the **CTAttached.SNK** state.

After receiving a **USB PD PS_RDY** from the original Source during a **USB PD PR_Swap**, the port *shall* transition directly to the **Attached.SRC** state (i.e., remove **Rd** from CC, assert **Rp** on CC and supply VBUS), but *shall* maintain its VCONN supply state, whether off or on, and its data role/connections.

4.5.2.2.6 UnattachedWait.SRC State

This state appears in Figure 4-12.

When in the **UnattachedWait.SRC** state, the port is discharging the CC pin that was providing VCONN in the previous **Attached.SRC** state.

4.5.2.2.6.1 UnattachedWait.SRC State Requirements

The port *shall not* enable VBUS or VCONN.

The port *shall* complete the VCONN turn off initiated when leaving the previous **Attached.SRC** state.

The port *shall* continue to provide an **Rp** termination, as specified in Table 4-27, on the CC pin not being discharged.

The port *shall not* provide an **Rp** termination on the CC pin being discharged.

The port *shall* provide an **Rdch** termination on the CC pin being discharged.

The port *shall* discharge the CC pin being discharged below **vVCONNDischarge**.

4.5.2.2.6.2 Exiting from UnattachedWait.SRC State

The port *shall* transition to **Unattached.SRC** when VCONN is below **vVCONNDischarge**. The port *may* delay this transition to allow the cable plug more time to reapply **Ra**.

4.5.2.2.7 Unattached.SRC State

This state appears in Figure 4-12, Figure 4-15, Figure 4-16, Figure 4-17 and Figure 4-18.

When in the **Unattached.SRC** state, the port is waiting to detect the presence of a Sink or an Accessory.

When in the **Unattached.SRC** state, the Charge-Through **VCONN-Powered USB Device** has detected a Source on its Charge-Through port and is independently monitoring its Host-side port to detect the presence of a Sink.

4.5.2.2.7.1 Unattached.SRC State Requirements

The port *shall not* drive VBUS or VCONN.

The port **shall** source current on both the CC1 and CC2 pins independently.

The port **shall** provide a separate **Rp** termination on the CC1 and CC2 pins as specified in Table 4-27. Note: A Source with a captive cable or just a plug presents a single **Rp** termination on its CC pin (A5).

The Charge-Through **VCONN-Powered USB Device** **shall** isolate its Host-side port from its Charge-Through port, including CCs and VBUS. The Charge-Through **VCONN-Powered USB Device** **shall** ensure that it is powered by VBUS from the Charge-Through port.

Upon entry into this state, the Charge-Through **VCONN-Powered USB Device** **shall** remove its **Rd** termination to ground on the Host-side port CC and provide an **Rp** termination instead advertising Default USB Power, as specified in Table 4-27, and continue to independently terminate its Charge-Through port's CC1 and CC2 pins to ground through **Rd**.

4.5.2.2.7.2 Exiting from Unattached.SRC State

The port **shall** transition to **AttachWait.SRC** when:

- The **SRC.Rd** state is present on either the CC1 or CC2 pin or
- The **SRC.Ra** state is present on both the CC1 and CC2 pins.

The maximum times that a Port **shall** take to transition to **AttachWait.SRC** are the following:

- **tNoToggleConnect** when neither Port Partner is toggling
- **tOnePortToggleConnect** when one Port Partner only is toggling

When both Port Partners are toggling, a Port **should** transition to **AttachWait.SRC** within **tTwoPortToggleConnect**. Note that when both Port Partners are DRPs it is indeterminate whether the local port will transition to **AttachWait.SRC** or **AttachWait.SNK**.

Note: The times **tOnePortToggleConnect** and **tTwoPortToggleConnect** relate to how long toggling ports **may** take to sync and detect a connection.

Note: A cable without an attached device can be detected, when the **SRC.Ra** state is detected on one of the CC1 or CC2 pins and the other CC pin is **SRC.Open**. However, in this case, the port **shall not** transition to **AttachWait.SRC**.

The Charge-Through **VCONN-Powered USB Device** **shall** transition to **AttachWait.SRC** when host-side VBUS is **vSafe0V** and **SRC.Rd** state is detected on the Host-side port's CC pin.

A DRP **shall** transition to **Unattached.SNK** within **tDRPTransition** after **dcSRC.DRP · tDRP**, or if directed.

A Charge-Through **VCONN-Powered USB Device** **shall** transition to **Unattached.SNK** within **tDRPTransition** after **dcSRC.DRP · tDRP**, or if Charge-Through VBUS is removed.

4.5.2.2.8 AttachWait.SRC State

This state appears in Figure 4-12, Figure 4-15, Figure 4-16, Figure 4-17 and Figure 4-18.

The **AttachWait.SRC** state is used to ensure that the state of both CC1 and CC2 pins are stable after a Sink is connected.

When in the **AttachWait.SRC** state, the Charge-Through **VCONN-Powered USB Device** ensures that the state of Host-side port's CC pin is stable after a Sink is connected.

4.5.2.2.8.1 AttachWait.SRC State Requirements

The port **shall not** drive VBUS or VCONN.

The port **shall** apply a load to VBUS to ensure it reaches **vSafe0V** within **tVsafe0V**. (Note: The connected Sink port might have charge left in its capacitors at up to **vSafe5V** that will prevent the Source port from transitioning to **Attached.SRC**.)

The port **shall** source current on both the CC1 and CC2 pins independently.

The port **shall** provide a separate **Rp** termination on the CC1 and CC2 pins as specified in Table 4-27. Note: A Source with a captive cable or just a plug presents a single **Rp** termination on its CC pin (A5).

The Charge-Through **VCONN-Powered USB Device** **shall** isolate its Host-side port from its Charge-Through port, including CCs and VBUS. The Charge-Through **VCONN-Powered USB Device** **shall** ensure that it is powered by VBUS from the Charge-Through port.

Upon entry into this state, the Charge-Through **VCONN-Powered USB Device** **shall** remove its **Rd** termination to ground on the Host-side port CC and provide an **Rp** termination instead advertising Default USB Power, as specified in Table 4-27, and continue to independently terminate its Charge-Through port's CC1 and CC2 pins to ground through **Rd**.

4.5.2.2.8.2 Exiting from AttachWait.SRC State

The port **shall** transition to **Attached.SRC** when VBUS is at **vSafe0V** and the **SRC.Rd** state is detected on exactly one of the CC1 or CC2 pins for at least **tCCDebounce** and VBULK is ready to supply VBUS at **vSafe5V**. The port **shall** have VBULK ready to supply VBUS within **tBulkDischarge** of exiting previous **Attached.SRC**.

The Charge-Through **VCONN-Powered USB Device** **shall** transition to **Try.SNK** when the host-side VBUS is at **vSafe0V** and the **SRC.Rd** state is on the Host-side port's CC pin for at least **tCCDebounce**.

If the port supports **Debug Accessory Mode**, it **shall** transition to **UnorientedDebugAccessory.SRC** when VBUS is at **vSafe0V** and the **SRC.Rd** state is detected on both the CC1 and CC2 pins for at least **tCCDebounce**.

A Source **shall** transition to **Unattached.SRC** and a DRP to **Unattached.SNK** when the **SRC.Open** state is detected on both the CC1 and CC2 pins. The Source **shall** detect the **SRC.Open** state within **tSRCDisconnect** but **should** detect it as quickly as possible.

A Source **shall** transition to **Unattached.SRC** and a DRP to **Unattached.SNK** when the **SRC.Open** state is detected on either the CC1 or CC2 pin and the other CC pin is **SRC.Ra**. The Source **shall** detect the **SRC.Open** state within **tSRCDisconnect** but **should** detect it as quickly as possible.

The port **may** exit to **ErrorRecovery** or **Disabled** if the VBUS voltage has not reached **vSafe0V** within **tVBUSOFF** (Maximum) after entering the **AttachWait.SRC** state.

A Charge-Through **VCONN-Powered USB Device** **shall** transition to **Unattached.SNK** when the **SRC.Open** state is detected on the Host-side port's CC or if Charge-Through VBUS falls below **vSinkDisconnect**. The Charge-Through **VCONN-Powered USB Device** **shall** detect the **SRC.Open** state within **tSRCDisconnect** but **should** detect it as quickly as possible.

A DRP that strongly prefers the Sink role **may optionally** transition to **Try.SNK** instead of **Attached.SRC** when VBUS is at **vSafe0V** and the **SRC.Rd** state is detected on exactly one of the CC1 or CC2 pins for at least **tCCDebounce**.

4.5.2.2.9 Attached.SRC State

This state appears in Figure 4-12, Figure 4-15, Figure 4-16, Figure 4-17 and Figure 4-18.

When in the **Attached.SRC** state, the port is attached and operating as a Source. When the port initially enters this state, it is also operating as a DFP. Subsequently, the initial power and data roles can be changed using **USB PD** commands.

When in the **Attached.SRC** state, the Charge-Through **VCONN-Powered USB Device** has detected a Sink on its Host-side port and has connected the Charge-Through port VBUS to the Host-side port VBUS.

4.5.2.2.9.1 Attached.SRC State Requirements

If the port needs to determine the orientation of the connector, it **shall** do so only upon entry to the **Attached.SRC** state by detecting which of the CC1 or CC2 pins is connected through the cable, i.e., which CC pin is in the **SRC.Rd** state.

If the port has entered this state from the **AttachWait.SRC** state or the **Try.SRC** state, the **SRC.Rd** state will be on only one of the CC1 or CC2 pins. The port **shall** source current on this CC pin and monitor its state.

If the port has entered this state from the **Attached.SNK** state as the result of a **USB PD PR_Swap**, the port **shall** source current on the connected CC pin and monitor its state.

The port **shall** provide an **Rp** as specified in Table 4-27.

The port **shall** supply VBUS current at the level it advertises on **Rp**.

Note: When the port is in a **USB PD** contract, the supply level of VBUS current is no longer associated with **Rp**, and the behavior of **Rp** may differ as defined in the **USB PD** specification.

The port **shall** supply VBUS within **tVBUSON** of entering this state, and ~~for the port shall continue to supply VBUS as long as it remains in the **Attached.SRC** state. It is only permissible to remove VBUS in **Attached.SRC** state when the port partner is a **VCONN-Powered USB Device**, as described below.~~

Note: Sources in the **Attached.SRC** is operating as a power source state **shall not** remove VBUS based on a minimal level of current drawn by the Sink as its basic functionality may depend on VBUS presence.

The port **shall not** initiate any **USB PD** communications until VBUS reaches **vSafe5V**.

If the port supports signaling on USB TX/RX pairs, it **shall**:

- Functionally connect the USB TX/RX pairs
- For VCONN, do one of two things:
 - Supply VCONN unconditionally to the CC pin not in the **SRC.Rd** state, or
 - Supply VCONN to the CC pin in the **SRC.Ra** state.

A port that does not support signaling on USB TX/RX pairs **may** supply VCONN in the same manner described above.

The port **may** negotiate a **USB PD PR_Swap**, **DR_Swap** or **VCONN_Swap**.

If the port supplies VCONN, it **shall** do so within **tVCONNON**.

The port **may** query the identity of the cable via **USB PD** on SOP'. If it detects that it is connected to a **VCONN-Powered USB Device**, the port **may** remove VBUS and discharge it to **vSafe0V**, while continuing to remain in this state with VCONN applied. The port **may** also initiate other SOP' communication.

The port **shall not** supply VCONN if it has entered this state as a result of a **USB PD PR_Swap** and was not previously supplying VCONN. A **USB PD DR_Swap** has no effect on which port sources VCONN.

The port **may** negotiate a **USB PD VCONN_Swap**. When the port successfully executes **USB PD VCONN_Swap** operation and was sourcing VCONN, it **shall** stop sourcing VCONN within **tVCONNOFF**. The port **shall** execute the **VCONN_Swap** in a make-before-break sequence in order to keep active USB Type-C to USB Type-C cables powered. When the port successfully executes **USB PD VCONN_Swap** operation and was not sourcing VCONN, it **shall** start sourcing VCONN within **tVCONNON**.

The Charge-Through **VCONN-Powered USB Device** *shall* continue to isolate its Host-side port's CC pin from its Charge-Through CC pins.

The Charge-Through **VCONN-Powered USB Device** *shall* maintain its **R_p** termination advertising Default USB Power on the Host-side port's CC pin, and continue to independently terminate its Charge-Through port's CC1 and CC2 pins to ground through **R_d**.

The Charge-Through **VCONN-Powered USB Device** *shall* immediately connect the Charge-Through port's VBUS through to the Host-side port's VBUS.

The Charge-Through **VCONN-Powered USB Device** *shall* ensure that it is powered entirely by VBUS.

The Charge-Through **VCONN-Powered USB Device** *shall* only respond to **USB PD Discover Identity** queries on SOP' on its Host-side port and complete any active queries prior to exiting this state. It *shall* ensure there is sufficient capacitance on the Host-side port CC to meet cReceiver as defined in **USB PD**.

4.5.2.2.9.2 Exiting from Attached.SRC State

A Source that is supplying VCONN or has yielded VCONN source responsibility to the Sink through **USB PD VCONN_Swap** messaging *shall* transition to **UnattachedWait.SRC** when the **SRC.Open** state is detected on the monitored CC pin. The Source *shall* detect the **SRC.Open** state within **tSRCDisconnect** but *should* detect it as quickly as possible.

A Source that is not supplying VCONN and has not yielded VCONN responsibility to the Sink through **USB PD VCONN_Swap** messaging *shall* transition to **Unattached.SRC** when the **SRC.Open** state is detected on the monitored CC pin. The Source *shall* detect the **SRC.Open** state within **tSRCDisconnect** but *should* detect it as quickly as possible.

When the **SRC.Open** state is detected on the monitored CC pin, a DRP *shall* transition to **Unattached.SNK** unless it strongly prefers the Source role. In that case, it *shall* transition to **TryWait.SNK**. This transition to **TryWait.SNK** is needed so that two devices that both prefer the Source role do not loop endlessly between Source and Sink. In other words, a DRP that would enter **Try.SRC** from **AttachWait.SNK** *shall* enter **TryWait.SNK** for a Sink detach from **Attached.SRC**.

A DRP that supports Charge-Through **VCONN-Powered USB Device** *shall* transition to **CTUnattached.SNK** if the connected device identifies itself as a Charge-Through **VCONN-Powered USB Device** in its **Discover Identity** Command response. The DRP *may* delay this transition to perform further SOP' communication.

A port *shall* cease to supply VBUS within **tVBUSOFF** of exiting **Attached.SRC**.

A port that is supplying VCONN *shall* cease to supply it within **tVCONNOff** of exiting **Attached.SRC**, unless it is exiting as a result of a **USB PD PR_Swap** or is transitioning into the **CTUnattached.SNK** state.

After a **USB PD PR_Swap** is accepted (i.e., either an **Accept** message is received or acknowledged), a DRP *shall* transition directly to the **Attached.SNK** state (i.e., remove **R_p** from CC, assert **R_d** on CC and stop supplying VBUS) and maintain its current data role, connection and VCONN supply state.

A Charge-Through **VCONN-Powered USB Device** *shall* transition to **Unattached.SNK** when VBUS falls below **vSinkDisconnect** or the Host-side port's CC pin is **SRC.Open**. The Charge-Through **VCONN-Powered USB Device** *shall* detect the **SRC.Open** state within **tSRCDisconnect** but *should* detect it as quickly as possible.

4.5.2.2.10 Try.SRC State

This state appears in Figure 4-16.

When in the **Try.SRC** state, the port is querying to determine if the port partner supports the Sink role.

Note: if both **Try.SRC** and **Try.SNK** mechanisms are implemented, only one **shall** be enabled by the port at any given time. Deciding which of these two mechanisms is enabled is product design-specific.

4.5.2.2.10.1 Try.SRC State Requirements

The port **shall not** drive VBUS or VCONN.

The port **shall** source current on both the CC1 and CC2 pins independently.

The port **shall** provide an **Rp** as specified in Table 4-27.

4.5.2.2.10.2 Exiting from Try.SRC State

The port **shall** transition to **Attached.SRC** when the **SRC.Rd** state is detected on exactly one of the CC1 or CC2 pins for at least **tTryCCDebounce**.

The port **shall** transition to **TryWait.SNK** after **tDRPTry** and the **SRC.Rd** state has not been detected and VBUS is within **vSafe0V**, or after **tTryTimeout** and the **SRC.Rd** state has not been detected.

4.5.2.2.11 TryWait.SNK State

This state appears in Figure 4-16.

When in the **TryWait.SNK** state, the port has failed to become a Source and is waiting to attach as a Sink. Alternatively, the port is responding to the Sink being removed while in the **Attached.SRC** state.

4.5.2.2.11.1 TryWait.SNK State Requirements

The port **shall not** drive VBUS or VCONN.

Both the CC1 and CC2 pins **shall** be independently terminated to ground through **Rd**.

4.5.2.2.11.2 Exiting from TryWait.SNK State

The port **shall** transition to **Attached.SNK** after **tCCDebounce** if or when VBUS is detected. Note the Source **may** initiate **USB PD** communications which will cause brief periods of the **SNK.Open** state on both the CC1 and CC2 pins, but this event will not exceed **tPDDebounce**.

The port **shall** transition to **Unattached.SNK** when the state of both CC1 and CC2 pins is **SNK.Open** for at least **tPDDebounce**.

4.5.2.2.12 Try.SNK State

This state appears in Figure 4-14, Figure 4-17 and Figure 4-18.

When in the **Try.SNK** state, the port is querying to determine if the port partner supports the Source role.

When in the **Try.SNK** state, the Charge-Through **VCONN-Powered USB Device** is querying to determine if the port partner on the Host-side port supports the Source role.

Note: if both **Try.SRC** and **Try.SNK** mechanisms are implemented, only one **shall** be enabled by the port at any given time. Deciding which of these two mechanisms is enabled is product design-specific.

4.5.2.2.12.1 Try.SNK State Requirements

The port **shall not** drive VBUS or VCONN.

Both the CC1 and CC2 pins **shall** be independently terminated to ground through **Rd**.

The Charge-Through **VCONN-Powered USB Device** *shall* isolate its Host-side port from its Charge-Through port, including CCs and VBUS. The Charge-Through **VCONN-Powered USB Device** *shall* ensure that it is powered by VBUS from the Charge-Through port.

The Charge-Through **VCONN-Powered USB Device** *shall* remove its **Rp** termination (Default USB Power advertisement) on the Host-side port CC and provide an **Rd** termination to ground instead, as specified in Table 4-27 and remain to independently terminate its Charge-Through port's CC1 and CC2 pins to ground through **Rd**.

4.5.2.2.12.2 Exiting from Try.SNK State

The port *shall* wait for **tDRPTry** and only then begin monitoring the CC1 and CC2 pins for the **SNK.Rp** state.

The port *shall* then transition to **Attached.SNK** when the **SNK.Rp** state is detected on exactly one of the CC1 or CC2 pins for at least **tTryCCDebounce** and VBUS is detected.

Alternatively, the port *shall* transition to **TryWait.SRC** if **SNK.Rp** state is not detected for **tTryCCDebounce**.

The Charge-Through **VCONN-Powered USB Device** *shall* wait for **tDRPTry** and only then begin monitoring the Host-side port's CC pin for the **SNK.Rp** state.

The Charge-Through **VCONN-Powered USB Device** *shall* then transition to **Attached.SNK** when the **SNK.Rp** state is detected on the Host-side port's CC pin for at least **tTryCCDebounce** and VBUS or VCONN is detected on Host-side port.

Alternatively, the Charge-Through **VCONN-Powered USB Device** *shall* transition to **TryWait.SRC** if Host-side **SNK.Rp** state is not detected for **tTryCCDebounce**.

A Sink with Accessory Support *shall* transition to **Unsupported.Accessory** if **SNK.Rp** state is not detected for **tDRPTryWait**.

Note: The Source *may* initiate **USB PD** communications which will cause brief periods of the **SNK.Open** state on both the CC1 and CC2 pins, but this event will not exceed **tTryCCDebounce**.

4.5.2.2.13 TryWait.SRC State

This state appears in Figure 4-17 and Figure 4-18.

When in the **TryWait.SRC** state, the port has failed to become a Sink and is waiting to attach as a Source.

When in the **TryWait.SRC** state, the Charge-Through **VCONN-Powered USB Device** has failed to become a Sink on its Host-side port and is waiting to attach as a Source on its Host-side port.

4.5.2.2.13.1 TryWait.SRC State Requirements

The requirements for this state are identical to **Unattached.SRC**.

4.5.2.2.13.2 Exiting from TryWait.SRC State

The port *shall* transition to **Attached.SRC** when VBUS is at **vSafe0V** and the **SRC.Rd** state is detected on exactly one of the CC pins for at least **tTryCCDebounce**.

The Charge-Through **VCONN-Powered USB Device** *shall* transition to **Attached.SRC** when host-side VBUS is at **vSafe0V** and the **SRC.Rd** state is detected on the Host-side port's CC pin for at least **tTryCCDebounce**.

The port *shall* transition to **Unattached.SNK** after **tDRPTry** if neither of the CC1 or CC2 pins are in the **SRC.Rd** state.

The Charge-Through *VCONN-Powered USB Device* **shall** transition to *Unattached.SNK* after *tDRPTry* if the Host-side port's CC pin is not in the *SRC.Rd* state.

4.5.2.2.14 Unattached.Accessory State

This state appears in Figure 4-14.

The *Unattached.Accessory* state allows accessory-supporting Sinks to connect to *VCONN-Powered Accessories*.

This state is functionally equivalent to the *Unattached.SRC* state in a DRP, except that *Attached.SRC* is not supported.

4.5.2.2.14.1 Unattached.Accessory State Requirements

The port **shall not** drive VBUS or VCONN.

The port **shall** source current on both the CC1 and CC2 pins independently.

The port **shall** provide an *Rp* as specified in Table 4-27.

4.5.2.2.14.2 Exiting from Unattached.Accessory State

A port that supports *VCONN-Powered Accessories* also **shall** transition to *AttachWait.Accessory* when the state of either CC1 or CC2 pin is *SRC.Ra* and the other CC pin is *SRC.Rd*.

The maximum time the local port **shall** take to transition from *Unattached.Accessory* to the *AttachWait.Accessory* state when a *VCONN-Powered Accessory* is present is *tOnePortToggleConnect*.

Otherwise, the port **shall** transition to *Unattached.SNK* within *tDRPTransition* after $dcSRC.DRP \cdot tDRP$, or if directed.

4.5.2.2.15 AttachWait.Accessory State

This state appears in Figure 4-14.

The *AttachWait.Accessory* state is used to ensure that the state of both CC1 and CC2 pins is stable after a cable is plugged in.

4.5.2.2.15.1 AttachWait.Accessory State Requirements

The requirements for this state are identical to *Unattached.Accessory*.

4.5.2.2.15.2 Exiting from AttachWait.Accessory State

The port **shall** transition to *Unattached.SNK* when the state of either the CC1 or CC2 pin is *SRC.Open* for at least *tCCDebounce*.

If the port supports *VCONN-Powered Accessories*, it **shall** transition to *Powered.Accessory* state if the state of either the CC1 or CC2 pin is *SRC.Rd* and the other CC pin is *SRC.Ra* concurrently for at least *tCCDebounce*.

4.5.2.2.16 CorrosionMitigation State

This state appears in Figure 4-12, Figure 4-14, Figure 4-16, Figure 4-17 and Figure 4-18.

The *CorrosionMitigation* state is used for the *Liquid Corrosion Mitigation Mode* specified in Appendix A.

4.5.2.2.16.1 CorrosionMitigation State Requirements

The port **shall** reconfigure its pins as detailed in Appendix A.

The port **shall not** drive VBUS or VCONN.

The port **shall** do one of the following implementations:

1. The port **shall** pull low with a resistance on both CC1 and CC2 less than or equal to the Maximum Impedance of **Ra** as specified in Table 4-29. The port should pull low with a resistance stronger than **Ra**.
2. The port **shall** pull low with a resistance on only one of the CC1 or CC2 pins and have the other pin open (present an impedance to ground above **zOPEN**). When using this single pull low implementation, the Sink **shall** monitor the voltage on the open CC pin and **shall** swap the resistance and open from one CC pin to the other when **SNK.Rp** is detected.

4.5.2.2.16.2 Exiting from CorrosionMitigation State

The port shall transition to **ErrorRecovery** when directed.

4.5.2.2.17 UnorientedDebugAccessory.SRC State

This state appears in Figure 4-12, Figure 4-16 and Figure 4-17.

The **UnorientedDebugAccessory.SRC** state is used for the **Debug Accessory Mode** specified in Appendix B.

4.5.2.2.17.1 UnorientedDebugAccessory.SRC State Requirements

This mode is for debug only and **shall not** be used for communicating with commercial products.

The port **shall** provide an **Rp** as specified in Table 4-27 on both the CC1 and CC2 pins and monitor to detect when the state of either is **SRC.Open**.

The port **shall** supply VBUS current at the level it advertises on **Rp**. The port **shall not** drive VCONN.

The port **may** connect any non-orientation specific debug signals for **Debug Accessory Mode** operation only after entry to this state.

4.5.2.2.17.2 Exiting from UnorientedDebugAccessory.SRC State

If the port is a Source, the port **shall** transition to **Unattached.SRC** when the **SRC.Open** state is detected on either the CC1 or CC2 pin.

If the port is a DRP, the port **shall** transition to **Unattached.SNK** when the **SRC.Open** state is detected on either the CC1 or CC2 pin.

The port **shall** transition to **OrientedDebugAccessory.SRC** state if orientation is required and detected as described in Section B.2.6.1.2.

4.5.2.2.18 OrientedDebugAccessory.SRC State

This state appears in Figure 4-12, Figure 4-16 and Figure 4-17.

The **OrientedDebugAccessory.SRC** state is used for the **Debug Accessory Mode** specified in Appendix B.

4.5.2.2.18.1 OrientedDebugAccessory.SRC State Requirements

This mode is for debug only and **shall not** be used for communicating with commercial products.

The port **shall** provide an **Rp** as specified in Table 4-27 on both the CC1 and CC2 pins and monitor to detect when the state of either is **SRC.Open**.

The port **shall** supply VBUS current at the level it advertises on **Rp**. The port **shall not** drive VCONN.

The port **shall** connect any orientation specific debug signals for **Debug Accessory Mode** operation only after entry to this state. Any non-orientation specific debug signals for **Debug Accessory Mode** operation **shall** be connected or remain connected in this state.

If the port needs to establish **USB PD** communications, it **shall** do so only after entry to this state. The port **shall not** initiate any **USB PD** communications until VBUS reaches **vSafe5V**. In this state, the port takes on the initial **USB PD** role of DFP/Source.

4.5.2.2.18.2 Exiting from OrientedDebugAccessory.SRC State

If the port is a Source, the port **shall** transition to **Unattached.SRC** when the **SRC.Open** state is detected on either the CC1 or CC2 pin.

If the port is a DRP, the port **shall** transition to **Unattached.SNK** when the **SRC.Open** state is detected on either the CC1 or CC2 pin.

4.5.2.2.19 DebugAccessory.SNK State

This state appears in Figure 4-13, Figure 4-14, Figure 4-16 and Figure 4-17.

The **DebugAccessory.SNK** state is used for the **Debug Accessory Mode** specified in Appendix B.

4.5.2.2.19.1 DebugAccessory.SNK State Requirements

This mode is for debug only and **shall not** be used for communicating with commercial products.

The port **shall not** drive VBUS or VCONN.

The port **shall** provide an **Rd** as specified in Table 4-28 on both the CC1 and CC2 pins.

If supported, orientation is determined as outlined in Section B.2.6.1.1. The port **shall** connect any debug signals for **Debug Accessory Mode** operation only after entry to this state.

4.5.2.2.19.2 Exiting from DebugAccessory.SNK State

The port **shall** transition to **Unattached.SNK** when VBUS is no longer present.

4.5.2.2.20 ~~PoweredAccessory~~Powered.Accessory State

This state appears in Figure 4-14.

When in the **Powered.Accessory** state, the port is powering a **VCONN-Powered Accessory** or **VCONN-Powered USB Device**.

4.5.2.2.20.1 ~~PoweredAccessory~~Powered.Accessory State Requirements

If the port needs to determine the orientation of the connector, it **shall** do so only upon entry to the **Powered.Accessory** state by detecting which of the CC1 or CC2 pins is connected through the cable (i.e., which CC pin is in the **SRC.Rd** state).

The **SRC.Rd** state is detected on only one of the CC1 or CC2 pins. The port **shall** advertise either 1.5 A or 3.0 A (see Table 4-27) on this CC pin and monitor its state.

The port **shall** supply VCONN on the unused CC pin within **tVCONN-PA** of entering the ~~**PoweredAccessory**~~**Powered.Accessory** state.

The port **shall not** drive VBUS.

When the port initially enters the **Powered.Accessory** state it **shall** operate as a **USB Power Delivery** Source with a DFP data role. In addition, the port **shall** support at least one of the following:

- Use **USB PD** to establish an explicit contract and then use Structured Vendor Defined Messages (Structured VDMs) to identify a **VCONN-Powered Accessory** and enter an **Alternate Mode**.
- Use **USB PD** to query the identity of a **VCONN-Powered USB Device** (that operates as a cable plug responding to SOP').

4.5.2.2.20.2 Exiting from ~~PoweredAccessory~~**Powered.Accessory** State

The port **shall** transition to **Unattached.SNK** when the **SRC.Open** state is detected on the monitored CC pin.

The port **shall** transition to **Try.SNK** if the attached device is not a **VCONN-Powered Accessory** or **VCONN-Powered USB Device**. For example, the attached device does not support **USB PD** or does not respond to **USB PD** commands required for a **VCONN-Powered Accessory** (e.g., Discover SVIDs, Discover Modes, etc.) or is a Sink or DRP attached through a Powered Cable.

The port **shall** transition to **Unsupported.Accessory** if the attached device is a **VCONN-Powered Accessory** but the port has not successfully entered an **Alternate Mode** within **tAMTimeout** (see Appendix E).

A port that supports Charge-Through **VCONN-Powered USB Device** **shall** transition to **CTUnattached.SNK** if the connected device identifies itself as a Charge-Through **VCONN-Powered USB Device** in its **Discover Identity** Command response. The port **may** delay this transition in order to perform further SOP' communication.

The port **shall** cease to supply VCONN within **tVCONNOFF** of exiting the **Powered.Accessory** state unless it is transitioning into the **CTUnattached.SNK** state.

4.5.2.2.21 Unsupported.Accessory State

This state appears in Figure 4-14.

If a **VCONN-Powered Accessory** does not enter an **Alternate Mode**, the **Unsupported.Accessory** state is used to wait until the accessory is unplugged before continuing.

4.5.2.2.21.1 Unsupported.Accessory State Requirements

Only one of the CC1 or CC2 pins **shall** be in the **SRC.Rd** state. The port **shall** advertise Default USB Power (see Table 4-27) on this CC pin and monitor its voltage.

The port **shall not** drive VBUS or VCONN.

A Sink with either **VCONN-Powered Accessory** or **VCONN-Powered USB Device** support **shall** provide user notification that it does not recognize or support the attached accessory or device.

4.5.2.2.21.2 Exiting from Unsupported.Accessory State

The port **shall** transition to **Unattached.SNK** when the **SRC.Open** state is detected on the monitored CC pin.

4.5.2.2.22 CTUnattached.VPD State

This state appears in Figure 4-18.

When in the **CTUnattached.VPD** state, the Charge-Through **VCONN-Powered USB Device** has detected **SNK.Open** on its host port for **tVPDCTDD**, indicating that it is connected to a Charge-Through capable Source, and is independently monitoring its Charge-Through port for the presence of a pass-through Power Source.

This state **may** also have been entered through detach of a Power Source on the Charge-Through port or detach of a sink from the **CTVPD**'s Charge-through port.

4.5.2.2.2.1 CTUnattached.VPD State Requirements

The Charge-Through **VCONN-Powered USB Device** *shall* isolate its Host-side port from its Charge-Through port, including CCs and VBUS. The Charge-Through **VCONN-Powered USB Device** *shall* ensure that it is powered by VCONN, does not consume more than ICCS (**USB 3.2**) / ICCSH (**USB 2.0**) from VBUS for monitoring, and is sufficiently isolated from VBUS to tolerate high voltages during Charge-Through operation.

Upon entry into this state, the device *shall* remove its **Rd** termination to ground (if present) on the Host-side port CC and provide an **Rp** termination advertising 3.0 A instead, as specified in Table 4-27. Note that because VBUS is not provided, the **Rp** termination signals continued connection to the port partner but does not carry with it any current advertisement.

The Charge-Through **VCONN-Powered USB Device** *shall* only respond to **USB PD Discover Identity** queries on SOP' on its Host-side port. It *shall* ensure there is sufficient capacitance on the Host-side port CC to meet cReceiver as defined in **USB PD**.

The Charge-Through **VCONN-Powered USB Device** *shall* independently terminate both the Charge-Through port's CC1 and CC2 pins to ground through **Rd**.

The Charge-Through **VCONN-Powered USB Device** *shall* provide a bypass capacitance of CCTB on the Charge-Through Port's VBUS pins.

4.5.2.2.2.2 Exiting from CTUnattached.VPD State

The Charge-Through **VCONN-Powered USB Device** *shall* transition to **CTAttachWait.VPD** when a Source connection is detected on the Charge-Through port, as indicated by the **SNK.Rp** state on exactly one of the Charge-Through port's CC pins.

Debug accessories are not supported on the Charge-Through port.

The Charge-Through **VCONN-Powered USB Device** *shall* transition to **Unattached.SNK** if VCONN falls below **vVCONNDisconnect**.

The Charge-Through **VCONN-Powered USB Device** *shall* transition to **CTUnattached.Unsupported** within **tDRPTransition** after the state of both the Charge-Through port's CC1 and CC2 pins is **SNK.Open** for **tDRP – dcSRC.DRP · tDRP**, or if directed.

4.5.2.2.2.3 CTAttachWait.VPD State

This state appears in Figure 4-18.

When in the **CTAttachWait.VPD** state, the device has detected the **SNK.Rp** state on exactly one of its Charge-Through port's CC pins and is waiting for VBUS on the Charge-Through port.

4.5.2.2.2.3.1 CTAttachWait.VPD State Requirements

The Charge-Through **VCONN-Powered USB Device** *shall* isolate its Host-side port from its Charge-Through port, including CCs and VBUS. The Charge-Through **VCONN-Powered USB Device** *shall* ensure that it is powered by VCONN, does not consume more than ICCS (**USB 3.2**) / ICCSH (**USB 2.0**) from VBUS for monitoring, and is sufficiently isolated from VBUS to tolerate high voltages during Charge-Through operation.

The Charge-Through **VCONN-Powered USB Device** *shall* maintain its **Rp** termination advertising 3.0 A on the Host-side port's CC pin, as well as the independent terminations to ground through **Rd** on the Charge-Through port's CC1 and CC2 pins.

The Charge-Through **VCONN-Powered USB Device** *shall* only respond to **USB PD Discover Identity** queries on SOP' on its Host-side port and complete any active queries prior to exiting this state. It *shall* ensure there is sufficient capacitance on the Host-side port CC to meet cReceiver as defined in **USB PD**.

4.5.2.2.23.2 Exiting from CTAttachWait.VPD State

The Charge-Through **VCONN-Powered USB Device** *shall* transition to **CTUnattached.VPD** when the state of both the Charge-Through port's CC1 and CC2 pins are **SNK.Open** for at least **tPDDebounce**.

The Charge-Through **VCONN-Powered USB Device** *shall* transition to **CTAttached.VPD** after the state of only one of the Charge-Through port's CC1 or CC2 pins is **SNK.Rp** for at least **tCCDebounce** and VBUS on the Charge-Through port is detected.

Note the Charge-Through Source *may* initiate **USB PD** communications which will cause brief periods of the **SNK.Open** state on one of the Charge-Through port's CC pins with the state of the Charge-Through port's other CC pin remaining **SNK.Open**, but this event will not exceed **tPDDebounce**.

The Charge-Through **VCONN-Powered USB Device** *shall* transition to **CTDisabled.VPD** if VCONN falls below **vVCONNDisconnect**.

4.5.2.2.24 CTAttached.VPD State

This state appears in Figure 4-18.

When in the **CTAttached.VPD** state, the Charge-Through **VCONN-Powered USB Device** has detected a Power Source on its Charge-Through port and has connected the Charge-Through port's CC and VBUS pins directly to the Host-side port's CC and VBUS pins. Hence all power delivery, negotiation and **USB PD** communication are performed directly between the unit on Host-side port and the Power Source connected to the Charge-Through port.

4.5.2.2.24.1 CTAttached.VPD State Requirements

Upon entry to this state, the Charge-Through **VCONN-Powered USB Device** *shall* detect which of the Charge-Through port's CC1 or CC2 pins is connected through the cable (i.e., the CC pin that is in the **SNK.Rp** state). The device *shall* then immediately, in the following order:

1. Remove or reduce any additional capacitance on the Host-side CC port that was introduced in order to meet cReceiver as defined in **USB PD** to present on CC a value equal to or less than two times the maximum value for cCablePlug_CC.
2. Disable the **Rp** termination advertising 3.0 A on the host port's CC pin.
3. Passively multiplex the detected Charge-Through port's CC pin through to the host port's CC pin with an impedance of less than RccCON.
4. Disable the **Rd** on the Charge-Through port's CC1 and CC2 pins.
5. Connect the Charge-Through port's VBUS through to the host port's VBUS.

These steps *shall* be completed within **tVPDDetach** minimum of entering this state.

The Charge-Through **VCONN-Powered USB Device** *shall* ensure that it is powered by VCONN, does not consume more than ICCS (**USB 3.2**) / ICCSH (**USB 2.0**) from VBUS for monitoring, and is sufficiently isolated from VBUS to tolerate high voltages during Charge-Through operation.

The Charge-Through **VCONN-Powered USB Device** *shall not* respond to any **USB PD** communication on any CC pin in this state. Any active queries on SOP' *shall* have been completed prior to entering this state.

4.5.2.2.24.2 Exiting from CTAttached.VPD State

The Charge-Through **VCONN-Powered USB Device** *shall* transition to **CTUnattached.VPD** when VBUS falls below **vSinkDisconnect** and the state of the passed-through CC pin is **SNK.Open** for **tVPDCTDD**.

The Charge-Through **VCONN-Powered USB Device** *shall* transition to **CTDisabled.VPD** if VCONN falls below **vVCONNDisconnect**.

4.5.2.2.25 CTDIsabled.VPD State

This state appears in Figure 4-18.

When in the **CTDisabled.VPD** state, the Charge-Through **VCONN-Powered USB Device** has detected the detach on its Host-side port but **may** still potentially be connected to a Power Source on the Charge-Through port, and is thus ensuring that the VBUS from the Power Source is removed.

4.5.2.2.25.1 CTDIsabled.VPD State Requirements

The Charge-Through **VCONN-Powered USB Device** **shall** isolate its Host-side port from its Charge-Through port, including CCs and VBUS.

The device **shall** present a high-impedance to ground (above **zOPEN**) on the Host-side port's CC pin and on the Charge-Through port CC1 and CC2 pins.

The Charge-Through **VCONN-Powered USB Device** **shall** ensure that it is powered entirely by VBUS.

4.5.2.2.25.2 Exiting from CTDIsabled.VPD State

The Charge-Through **VCONN-Powered USB Device** **shall** transition to **Unattached.SNK** after **tVPDDisable**.

4.5.2.2.26 CTUnattached.SNK State

This state appears in Figure 4-14, Figure 4-16 and Figure 4-17.

When in the **CTUnattached.SNK** state, the port has detected that it is attached to a Charge-Through **VCONN-Powered USB Device** and is ready if a Power Source is attached to the Charge-Through **VCONN-Powered USB Device**.

This state **may** also have been entered through detach of a Charge-Through Power Source.

4.5.2.2.26.1 CTUnattached.SNK State Requirements

Upon entry to this state, the port **shall** remove its **Rp** termination (if present) and terminate CC to ground through **Rd**.

The port **shall** continue to supply VCONN.

The port **shall** stop sourcing or sinking VBUS and discharge it.

In **USB PD** Version 2.0, the port **shall** initiate PD messages.

The port **may** query the state of the attached **VCONN-Powered USB Device** by sending SOP' messages on **USB PD** to read the **VPD**'s eMarker.

4.5.2.2.26.2 Exiting from CTUnattached.SNK State

The port **shall** transition to **CTAttached.SNK** when VBUS is detected. Note that by this point, the **VCONN-Powered USB Device** has already de-bounced the passed-through CC pin.

The port **shall** transition to **Unattached.SNK** if the state of the CC pin is **SNK.Open** for **tVPDDetach** after VBUS is **vSafe0V**.

4.5.2.2.27 CTAttached.SNK State

This state appears in Figure 4-14, Figure 4-16 and Figure 4-17.

When in the **CTAttached.SNK** state, the port is connected to a Charge-Through **VCONN-Powered USB Device**, which in turn is passing through the connection to a Power Source.

4.5.2.2.27.1 CTAttached.SNK State Requirements

The port **shall** continue to terminate CC to ground through **Rd**. Since there is now a Power Source connected through to VBUS and CC, the port **shall** operate in one of the **Sink Power Sub-States** shown in Figure 4-19, and remain within the **Sink Power Sub-States**, until either VBUS is removed, or a **USB PD** contract is established with the source.

The port **shall not** negotiate a voltage on VBUS higher than the maximum voltage specified in the Charge-Through **VCONN-Powered USB Device's Discover Identity** Command response.

The port **shall** continue to supply VCONN.

The port **shall** reject a VCONN swap request.

The port **shall not** perform **USB BC 1.2** primary detection, as that will interfere with **VPD** functionality.

In **USB PD** Version 2.0, the port **shall not** initiate **USB PD** messages, although it remains a DFP for USB data.

The port **shall** neither initiate nor respond to any SOP' communication.

The port **shall** meet the **Sink Power Sub-State** requirements specified in Section 4.5.2.3.

The port **shall** meet the additional maximum current constraints described in Section 4.6.2.5.

The port **shall** follow the restrictions on **USB PD** messages described in Section 4.10.2.

The port **shall** alter its advertised capabilities to UFP role/sink only role as described in Section 4.10.2.

4.5.2.2.27.2 Exiting from CTAttached.SNK State

A port that is not in the process of a **USB PD** Hard Reset **shall** transition to **CTUnattached.SNK** within **tSinkDisconnect** when VBUS falls below **vSinkDisconnect** for VBUS operating at or below 5 V or below **vSinkDisconnectPD** when negotiated by **USB PD** to operate above 5 V.

A port that has entered into **USB PD** communications with the Source and has seen the CC voltage exceed **vRd-USB** **may** monitor the CC pin to detect cable disconnect in addition to monitoring VBUS.

A port that is monitoring the CC voltage for disconnect **shall** transition to **CTUnattached.SNK** within **tSinkDisconnect** after the CC voltage remains below **vRd-USB** for **tPDDebounce**.

4.5.2.2.28 CTUnattached.Unsupported State

This state appears in Figure 4-18.

When in the **CTUnattached.Unsupported** state, the Charge-Through **VCONN-Powered USB Device** has previously detected **SNK.Open** on its host port for **tVPDCTDD**, indicating that it is connected to a Charge-Through Capable Source, and is now monitoring its Charge-Through port for the presence of an unsupported sink.

A Charge-Through **VCONN-Powered USB Device** does not support Sinks or **Debug Accessory Mode**.

4.5.2.2.28.1 CTUnattached.Unsupported State Requirements

The Charge-Through **VCONN-Powered USB Device** **shall** isolate its Host-side port from its Charge-Through port, including CCs and VBUS. The Charge-Through **VCONN-Powered USB Device** **shall** ensure that it is powered by VCONN, does not consume more than ICCS (**USB 3.2**) / ICCSH (**USB 2.0**) from VBUS for monitoring, and is sufficiently isolated from VBUS to tolerate high voltages during Charge-Through operation.

Upon entry into this state, the Charge-Through **VCONN-Powered USB Device** shall maintain its **Rp** termination advertising 3.0 A on the Host-side port's CC pin, remove its **Rd** terminations to ground on the Charge-Through port's CC1 and CC2 pins, and provide a **Rp** termination advertising Default USB Power instead.

The Charge-Through **VCONN-Powered USB Device** shall only respond to **USB PD Discover Identity** queries on SOP' on its Host-side port. It shall ensure there is sufficient capacitance on the Host-side port CC to meet cReceiver as defined in **USB PD**.

4.5.2.2.28.2 Exiting from CTUnattached.Unsupported State

The Charge-Through **VCONN-Powered USB Device** shall transition to **CTAttachWait.Unsupported** when a Sink connection is detected on the Charge-Through port, as indicated by the **SRC.Rd** state on at least one of the Charge-Through port's CC pins or **SRC.Ra** state on both the CC1 and CC2 pins.

The Charge-Through **VCONN-Powered USB Device** shall transition to **Unattached.SNK** if VCONN falls below **vVCONNDisconnect**.

Otherwise, a Charge-Through **VCONN-Powered USB Device** shall transition to **CTUnattached.VPD** within **tDRPTransition** after **dcSRC.DRP · tDRP**, or if directed.

4.5.2.2.29 CTAttachWait.Unsupported State

This state appears in Figure 4-18.

The **CTAttachWait.Unsupported** state is used to ensure that the state of both the Charge-Through Port's CC1 and CC2 pins are stable for at least **tCCDebounce**.

4.5.2.2.29.1 CTAttachWait.Unsupported State Requirements

The requirements for this state are identical to **CTUnattached.Unsupported** state.

4.5.2.2.29.2 Exiting from CTAttachWait.Unsupported State

The Charge-Through **VCONN-Powered USB Device** shall transition to **CTTry.SNK** if the state of at least one of the Charge-Through port's CC pins is **SRC.Rd**, or if the state of both the CC1 and CC2 pins is **SRC.Ra** for at least **tCCDebounce**.

The Charge-Through **VCONN-Powered USB Device** shall transition to **CTUnattached.VPD** when the state of either the Charge-Through Port's CC1 or CC2 pin is **SRC.Open** for at least **tCCDebounce**.

The Charge-Through **VCONN-Powered USB Device** shall transition to **Unattached.SNK** if VCONN falls below **vVCONNDisconnect**.

4.5.2.2.30 CTTry.SNK State

This state appears in Figure 4-18.

When in the **CTTry.SNK** state, the Charge-Through **VCONN-Powered USB Device** is querying to determine if the port partner on the Charge-Through port supports the source role.

4.5.2.2.30.1 CTTry.SNK State Requirements

The requirements for this state are identical to **CTUnattached.VPD** state.

4.5.2.2.30.2 Exiting from CTTry.SNK State

The Charge-Through **VCONN-Powered USB Device** shall wait for **tDRPTry** and only then begin monitoring the Charge-Through port's CC pins for the **SNK.Rp** state.

The Charge-Through **VCONN-Powered USB Device** *shall* then transition to **CTAttached.VPD** when the **SNK.Rp** state is detected on the Charge-Through port's CC pins for at least **tTryCCDebounce** and VBUS is detected on Charge-Through port.

A Charge-Through **VCONN-Powered USB Device** *shall* transition to **CTAttached.Unsupported** if **SNK.Rp** state is not detected for **tDRPTryWait**.

Note: The Source *may* initiate **USB PD** communications which will cause brief periods of the **SNK.Open** state on both the CC1 and CC2 pins, but this event will not exceed **tTryCCDebounce**.

The Charge-Through **VCONN-Powered USB Device** *shall* transition to **Unattached.SNK** if VCONN falls below **vVCONNDisconnect**.

4.5.2.2.31 CTAttached.Unsupported State

This state appears in Figure 4-18.

If the port partner to the Charge-Through **VCONN-Powered USB Device**'s Charge-Through port either does not support the source power role, or failed to negotiate the source role, the **CTAttached.Unsupported** state is used to wait until that device is unplugged before continuing.

4.5.2.2.31.1 CTAttached.Unsupported State Requirements

The Charge-Through **VCONN-Powered USB Device** *shall* isolate its Host-side port from its Charge-Through port, including CCs and VBUS. The Charge-Through **VCONN-Powered USB Device** *shall* ensure that it is powered by VCONN, does not consume more than ICCS (**USB 3.2**) / ICCSH (**USB 2.0**) from VBUS for monitoring, and is sufficiently isolated from VBUS to tolerate high voltages during Charge-Through operation.

Upon entry into this state, the Charge-Through **VCONN-Powered USB Device** *shall* maintain its **Rp** termination advertising 3.0 A on the Host-side port's CC pin, remove its **Rd** terminations to ground on the Charge-Through port's CC1 and CC2 pins, and provide a **Rp** termination advertising Default USB Power instead.

At least one of the CC1 or CC2 pins will be in the **SRC.Rd** state or both will be in the **SRC.Ra** state. The Charge-Through port *shall* advertise Default USB Power (see Table 4-27) on its CC pins and monitor their voltage.

The Charge-Through **VCONN-Powered USB Device** *shall* present a **USB Billboard Device Class** interface indicating that it does not recognize or support the attached accessory or device.

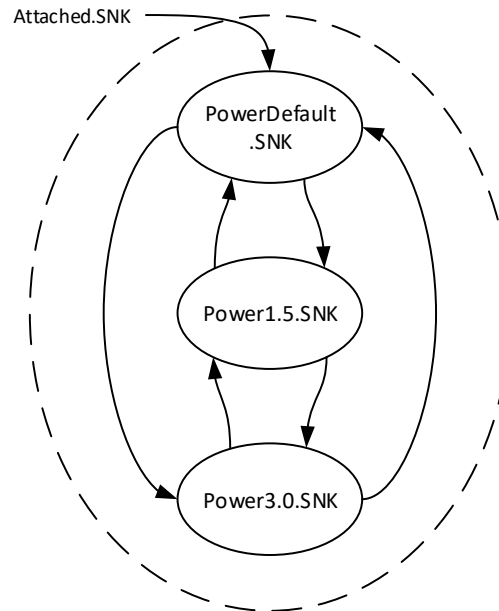
4.5.2.2.31.2 Exiting from CTAttached.Unsupported State

The Charge-Through **VCONN-Powered USB Device** *shall* transition to **CTUnattached.VPD** when **SRC.Open** state is detected on both the Charge-Through port's CC pins or the **SRC.Open** state is detected on one CC pin and **SRC.Ra** is detected on the other CC pin.

4.5.2.3 Sink Power Sub-State Requirements

When in the **Attached.SNK** or **CTAttached.SNK** states and the Source is supplying default VBUS, the port *shall* operate in one of the sub-states shown in Figure 4-19. The initial **Sink Power Sub-State** is **PowerDefault.SNK**. Subsequently, the **Sink Power Sub-State** is determined by Source's USB Type-C current advertisement. The port in **Attached.SNK** *shall* remain within the **Sink Power Sub-States** until either VBUS is removed, or a **USB PD** contract is established with the Source.

Figure 4-19 Sink Power Sub-States



The Sink is only required to implement *Sink Power Sub-State* transitions if the Sink wants to consume more than default USB current.

Note that for the *CTAttached.SNK* state, there are further limitations on maximum current (see Section 4.6.2.5).

4.5.2.3.1 PowerDefault.SNK Sub-State

This sub-state supports Sinks consuming current within the lowest range (default) of Source-supplied current.

4.5.2.3.1.1 PowerDefault.SNK Sub-State Requirements

The port **shall** draw no more than the default USB power from VBUS. See Section 4.6.2.1.

If the port wants to consume more than the default USB power, it **shall** monitor *vRd* to determine if more current is available from the Source.

4.5.2.3.1.2 Exiting from PowerDefault.SNK Sub-State

For any change in *vRd* indicating a change in allowable power, the port **shall not** transition until the new *vRd* has been stable for at least *tRpValueChange*.

For a *vRd* in the *vRd-1.5* range, the port **shall** transition to the *Power1.5.SNK* Sub-State.

For a *vRd* in the *vRd-3.0* range, the port **shall** transition to the *Power3.0.SNK* Sub-State.

4.5.2.3.2 Power1.5.SNK Sub-State

This sub-state supports Sinks consuming current within the two lower ranges (default and 1.5 A) of Source-supplied current.

4.5.2.3.2.1 Power1.5.SNK Sub-State Requirements

The port **shall** draw no more than 1.5 A from VBUS.

The port **shall** monitor **vRd** while it is in this sub-state.

4.5.2.3.2.2 Exiting from Power1.5.SNK Sub-State

For any change in **vRd** indicating a change in allowable power, the port **shall not** transition until the new **vRd** has been stable for at least **tRpValueChange**.

For a **vRd** in the **vRd-USB** range, the port **shall** transition to the **PowerDefault.SNK** Sub-State and reduce its power consumption to the new range within **tSinkAdj**.

For a **vRd** in the **vRd-3.0** range, the port **shall** transition to the **Power3.0.SNK** Sub-State.

4.5.2.3.3 Power3.0.SNK Sub-State

This sub-state supports Sinks consuming current within all three ranges (default, 1.5 A and 3.0 A) of Source-supplied current.

4.5.2.3.3.1 Power3.0.SNK Sub-State Requirements

The port **shall** draw no more than 3.0 A from VBUS.

The port **shall** monitor **vRd** while it is in this sub-state.

4.5.2.3.3.2 Exiting from Power3.0.SNK Sub-State

For any change in **vRd** indicating a change in allowable power, the port **shall not** transition until the new **vRd** has been stable for at least **tRpValueChange**.

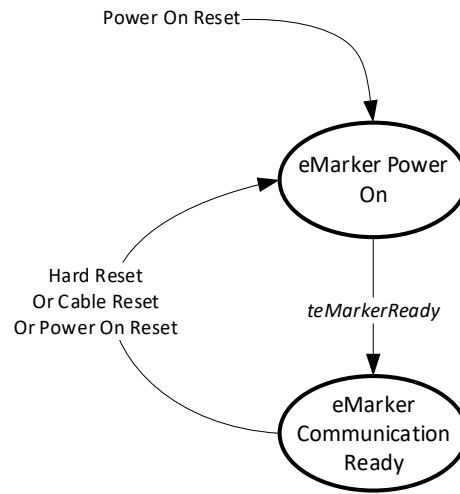
For a **vRd** in the **vRd-USB** range, the port **shall** transition to the **PowerDefault.SNK** Sub-State and reduce its power consumption to the new range within **tSinkAdj**.

For a **vRd** in the **vRd-1.5** range, the port **shall** transition to the **Power1.5.SNK** Sub-State and reduce its power consumption to the new range within **tSinkAdj**.

4.5.2.4 Cable eMarker State Machine Requirements

Figure 4-20 ~~and illustrate~~ illustrates the cable eMarker connection state ~~diagrams for passive and active cables, respectively~~ diagram. The requirements in this section also apply to an eMarker that is in a direct connect device.

Figure 4-20 ~~Passive~~ Cable eMarker State Diagram



~~Figure 4-20 Active Cable eMarker State Diagram~~

4.5.2.4.1 ~~Cable~~ Power On State

This state appears in Figure 4-20. This is the initial power on state for each eMarker ~~in the cable~~ when VCONN or VBUS is applied.

CableNote: For USB Type-C detachable cables, VCONN **shall** be used for powering its eMarker(s). For a direct connect device that includes an eMarker, its eMarker **may** be powered by either VCONN, VBUS, or another source of local power.

4.5.2.4.1.1 ~~eMarker~~ Power On State Requirements

Each eMarker ~~in the cable~~ **shall** power on.

The cable **shall not** respond to SOP' and SOP'' commands in this state. **initialize** when power is applied.

4.5.2.4.1.2 ~~Exiting from Cable~~ eMarker Power On State

Each eMarker ~~in a passive or active cable~~ **shall** transition to ~~Assign Cable SOP*~~the eMarker Communication Ready State when it has completed its boot process. Each eMarker **shall** transition to ~~Assign Cable SOP*~~the eMarker Communication Ready State within ~~teVconnStable~~teMarkerReady.

Note: For a direct connect device that includes an eMarker, if the local power used for the eMarker is not VCONN or VBUS, the eMarker **shall** still meet the **teMarkerReady** Respond to SOP'/SOP'' timing requirement.

4.5.2.4.2 ~~eMarker~~ Communications Ready State

This state appears in Figure 4-20 ~~and~~.

A passive cable has only one eMarker powered at a time. This cable eMarker in a passive cable **shall** respond to SOP' in this state.

Each cable eMarker in an active cable **shall** respond to a pre-set SOP' or SOP''. If only one eMarker exists in the cable, it **shall** only respond to SOP'.

Cable designers **shall** ensure that the eMarker works correctly in the presence of ~~ground and VCONN~~ the maximum allowable cable IR drop for ground (see Section 4.4.1.).

4.5.2.4.2.1 **Respond to SOP'/SOP''eMarker Communications Ready State Requirements**

Each eMarker ~~in the passive or active cable~~ **shall** be able to respond as appropriate to ~~any USB PD communication~~ communications sent to ~~its pre-set SOP' or SOP''~~ it.

For a passive cable, only one eMarker **should** be powered at a time and **shall** respond to SOP' only. If two eMarkers exist in a ~~passive or active~~ cable and are powered at the same time, then only one **shall** respond to SOP' and the other **shall** respond to SOP''. The assignment of SOP' and SOP'' is fixed for each eMarker in a cable and **shall not** be dynamically set when power is applied to VCONN.

~~Exiting from A direct connect device~~ **shall** contain at most one eMarker. ~~Any eMarker in a direct connect device~~ **shall only** respond to SOP'/SOP''.

4.5.2.4.2.2 **Exiting from eMarker Communications Ready State**

Each eMarker in the cable **shall** transition to ~~Cable~~the eMarker Power On ~~upon~~State upon a Power On Reset event. ~~A Power On Reset event in a cable~~ **shall** be triggered by sensing VCONN less than ~~vVCONNDisconnect~~ vVCONNDisconnect.

~~The eMarker in a direct connect device~~ **shall** transition to the eMarker Power On State upon a Power On Reset event. ~~A Power On Reset event in a direct connect device~~ **shall** be triggered by either:

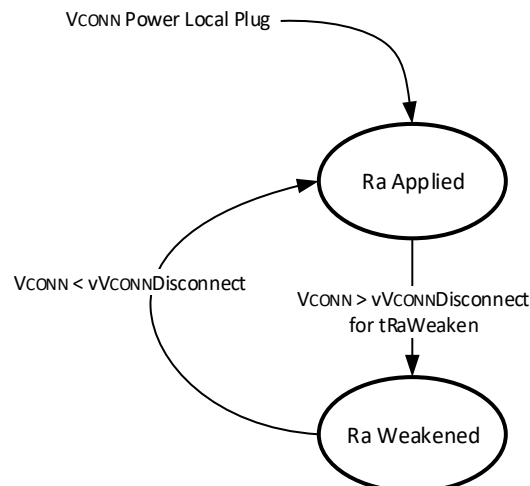
- sensing VCONN less than vVCONNDisconnect, or
- transitioning from any attached state to any unattached state.

Each eMarker ~~in the cable~~ **shall** transition to ~~Cable~~the eMarker Power On State upon sensing a Hard Reset or Cable Reset.

4.5.2.5 **Cable Ra Management State Machine Requirements**

Figure 4-21 illustrates the cable eMarker state diagram for applying and weakening or removing **Ra**. This state machine runs independently from the Cable eMarker state machine described in Section 4.5.2.4.

Figure 4-21 Cable Ra Management State Diagram



4.5.2.5.1 Ra Applied State

This state appears in Figure 4-21. This is the initial state at power on for each eMarker in the cable.

4.5.2.5.1.1 Ra Applied State Requirements

Each eMarker in the cable **shall** apply **Ra** to VCONN within **tRaReconnect**.

4.5.2.5.1.2 Exiting from Ra Applied State

Each eMarker in a passive or active cable **shall** transition to the **Ra** Weakened state when VCONN is greater than **vVCONNDisconnect** for **tRaWeaken**.

4.5.2.5.2 Ra Weakened State

This state appears in Figure 4-21.

4.5.2.5.2.1 Ra Weakened State Requirements

The eMarker in the cable **shall** remove or weaken **Ra**.

Passive cables **shall** meet the Power for electronically marked passive cables defined in Table 4-6. Active cables **shall** meet the Power for Active Cables in Table 4 6.

4.5.2.5.2.2 Exiting from Ra Weakened State

Each eMarker in a passive or active cable **shall** transition to the **Ra** Applied state when VCONN is below **vVCONNDisconnect**.

4.5.2.6 Connection States Summary

Table 4-18 defines the mandatory and **optional** states for each type of port.

Table 4-18 Mandatory and Optional States

	SOURCE	SINK	DRP	USB PD Communications
Disabled	Optional	Optional	Optional	Not Permitted
ErrorRecovery	Mandatory ¹⁰	Mandatory ¹⁰	Mandatory ¹⁰	Not Permitted
Unattached.SNK	N/A	Mandatory	Mandatory	Not Permitted
AttachWait.SNK	N/A	Mandatory ¹	Mandatory	Not Permitted
Attached.SNK	N/A	Mandatory	Mandatory	Permitted
UnattachedWait.SRC	Mandatory or N/A ⁷	N/A	N/A	Not Permitted
Unattached.SRC	Mandatory	N/A	Mandatory	Not Permitted
AttachWait.SRC	Mandatory	N/A	Mandatory	Not Permitted
Attached.SRC	Mandatory	N/A	Mandatory	Permitted
Try.SRC⁴	N/A	N/A	Optional	Not Permitted
TryWait.SNK²	N/A	N/A	Optional	Not Permitted
Try.SNK^{4, 8}	N/A	N/A	Optional	Not Permitted
TryWait.SRC^{5, 8}	N/A	N/A	Optional	Not Permitted

	SOURCE	SINK	DRP	USB PD Communications
<i>CorrosionMitigation</i>	<i>Optional</i>	<i>Optional</i>	<i>Optional</i>	Not Permitted
<i>UnorientedDebugAccessory.SRC</i>	Optional ⁶	N/A	Optional ⁶	Not Permitted
<i>OrientedDebugAccessory.SRC</i>	Optional ⁶	N/A	Optional ⁶	Permitted
<i>DebugAccessory.SNK</i>	N/A	<i>Optional</i>	<i>Optional</i>	Permitted
<i>Unattached.Accessory</i>	N/A	<i>Optional</i>	N/A	Not Permitted
<i>AttachWait.Accessory</i>	N/A	<i>Optional</i>	N/A	Not Permitted
<i>Powered.Accessory</i>	N/A	<i>Optional</i>	N/A	Permitted
<i>Unsupported.Accessory</i> ³	N/A	<i>Optional</i>	N/A	Not Permitted
<i>CTUnattached.VPD</i>	N/A	N/A	<i>Optional</i>	SOP' Permitted
<i>CTAttachWait.VPD</i> ⁸	N/A	N/A	<i>Optional</i>	SOP' Permitted
<i>CTAttached.VPD</i> ⁸	N/A	N/A	<i>Optional</i>	Not Permitted
<i>CTDisabled.VPD</i> ⁸	N/A	N/A	<i>Optional</i>	Not Permitted
<i>CTUnattached.SNK</i>	N/A	N/A	<i>Optional</i>	SOP' Permitted
<i>CTAttached.SNK</i> ⁹	N/A	N/A	<i>Optional</i>	Permitted
<i>CTUnattached.Unsupported</i> ⁸	N/A	N/A	<i>Optional</i>	SOP' Permitted
<i>CTAttachWait.Unsupported</i> ⁸	N/A	N/A	<i>Optional</i>	SOP' Permitted
<i>CTTry.SNK</i> ⁸	N/A	N/A	<i>Optional</i>	SOP' Permitted
<i>CTAttached.Unsupported</i> ⁸	N/A	N/A	<i>Optional</i>	SOP' Permitted
<i>PowerDefault.SNK</i>	N/A	Mandatory	Mandatory	Permitted
<i>Power1.5.SNK</i>	N/A	<i>Optional</i>	<i>Optional</i>	Permitted
<i>Power3.0.SNK</i>	N/A	<i>Optional</i>	<i>Optional</i>	Permitted

Notes:

1. *Optional* for UFP applications that are **USB 2.0**-only, consume USB Default Power and do not support **USB PD** or accessories.
2. *TryWait.SNK* is mandatory when *Try.SRC* is supported.
3. *Unsupported.Accessory* is mandatory when *Powered.Accessory* is supported.
4. *Try.SRC* and *Try.SNK* **shall not** be supported at the same time, although an unattached device **may** dynamically choose between *Try.SRC* and *Try.SNK* state machines based on external factors.
5. *TryWait.SRC* is mandatory when *Try.SNK* is supported.
6. *UnorientedDebugAccessory.SRC* is required for any Source or DRP that supports *Debug Accessory Mode*. *OrientedDebugAccessory.SRC* is only required if orientation detection is necessary in *Debug Accessory Mode*.
7. Mandatory for a DFP that was providing VCONN in the previous *Attached.SRC* state. N/A for a DFP that was not providing VCONN in the previous *Attached.SRC* state.
8. *CTAttachWait.VPD*, *CTAttached.VPD*, *CTDisabled.VPD*, *Try.SNK*, *TryWait.SRC*, *CTUnattached.Unsupported*, *CTAttachWait.Unsupported*, *CTAttached.Unsupported*, and *CTTry.SNK* are mandatory when *CTUnattached.VPD* is supported.
9. *CTAttached.SNK* is mandatory when *CTUnattached.SNK* is supported.
10. *Optional* for non-**USB4** and non-**USB PD** implementations

4.5.3 USB Port Interoperability Behavior

This section describes interoperability behavior between USB Type-C to USB Type-C ports and between USB Type-C to legacy USB ports.

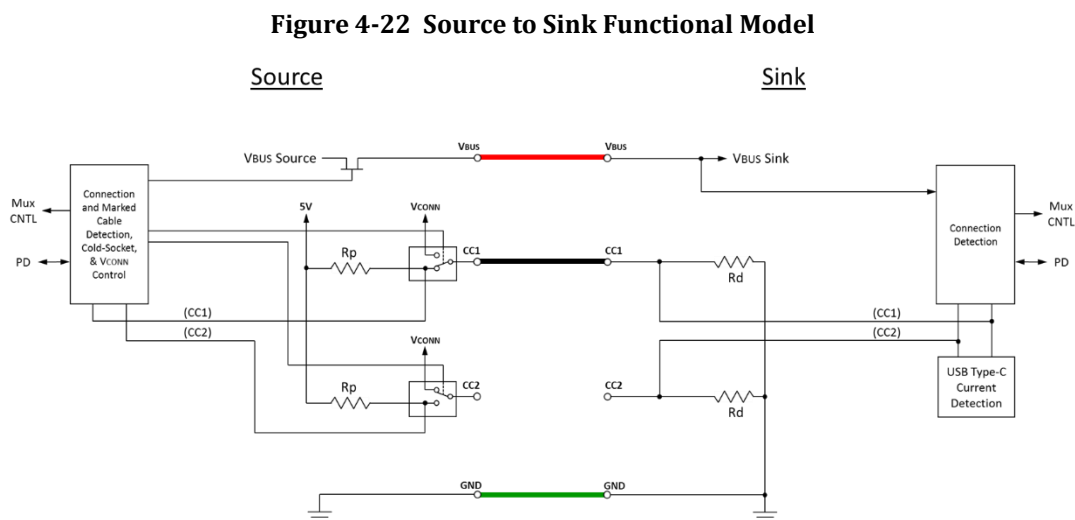
4.5.3.1 USB Type-C Port to USB Type-C Port Interoperability Behaviors

The following sub-sections describe typical port-to-port interoperability behaviors for the various combinations of USB Type-C Source, Sink and DRPs as presented in Table 4-9. In all the described behaviors, the impact of **USB PD**-based swaps (**PR_Swap**, **DR_Swap** or **VCONN_Swap**) are not considered.

The figures in the following sections illustrate the CC1 and CC2 routing after the CC detection process is complete.

4.5.3.1.1 Source to Sink Behavior

Figure 4-22 illustrates the functional model for a Source connected to a Sink. The single CC wire that is in a standard cable is only shown in one of the four possible connection routes, CC1 to CC1.

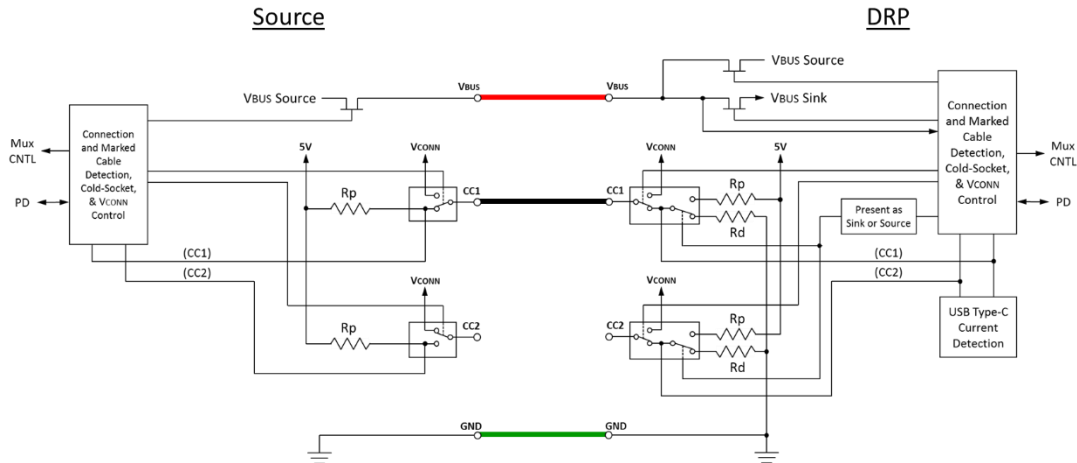


The following describes the behavior when a Source is connected to a Sink.

1. Source and Sink in the unattached state.
2. Source transitions from **Unattached.SRC** to **Attached.SRC** through **AttachWait.SRC**.
 - Source detects the Sink's pull-down on CC and enters **Attached.SRC** through **AttachWait.SRC**.
 - Source turns on VBUS and VCONN.
3. Sink transitions from **Unattached.SNK** to **Attached.SNK** through **AttachWait.SNK**. Sink **may** skip **AttachWait.SNK** if it is **USB 2.0** only and does not support accessories.
 - Sink detects VBUS and enters **Attached.SNK** through **AttachWait.SNK**.
4. While the Source and Sink are in the attached state:
 - Source adjusts **Rp** as needed to limit the current the Sink **may** draw.
 - Sink detects and monitors **vRd** for available current on VBUS.
 - Source monitors CC for detach and when detected, enters **Unattached.SRC**.
 - Sink monitors VBUS for detach and when detected, enters **Unattached.SNK**.

Figure 4-23 illustrates the functional model for a Source connected to a DRP. The single CC wire that is in a standard cable is only shown in one of the four possible connection routes, CC1 to CC1.

Figure 4-23 Source to DRP Functional Model



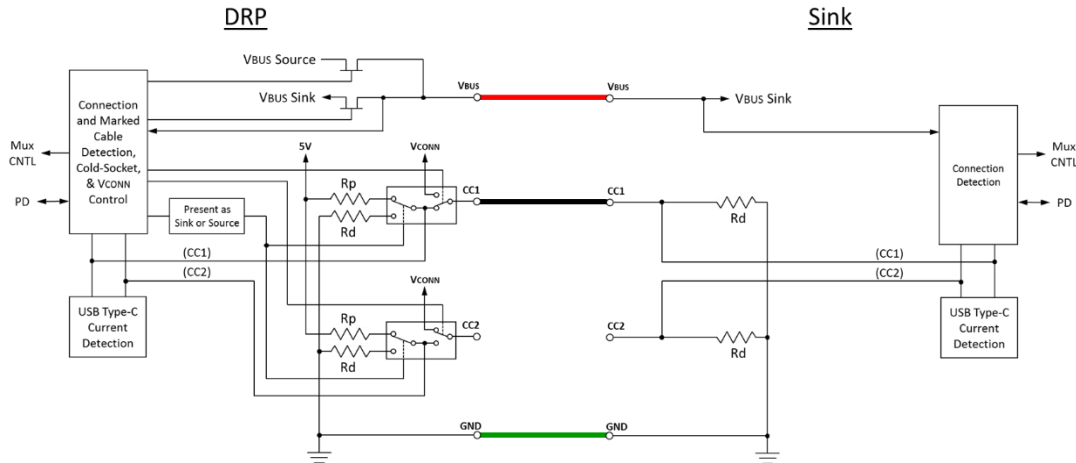
The following describes the behavior when a Source is connected to a DRP.

1. Source and DRP in the unattached state.
 - DRP alternates between *Unattached.SRC* and *Unattached.SNK*.
2. Source transitions from *Unattached.SRC* to *Attached.SRC* through *AttachWait.SRC*.
 - Source detects the DRP's pull-down on CC and enters *AttachWait.SRC*. After *tCCDebounce* it then enters *Attached.SRC*.
 - Source turns on VBUS and VCONN.
3. DRP transitions from *Unattached.SNK* to *Attached.SNK* through *AttachWait.SNK*.
 - DRP in *Unattached.SNK* detects pull up on CC and enters *AttachWait.SNK*. After that state persists for *tCCDebounce* and it detects VBUS, it enters *Attached.SNK*.
4. While the Source and DRP are in their respective attached states:
 - Source adjusts *Rp* as needed to limit the current the DRP (as Sink) **may** draw.
 - DRP (as Sink) detects and monitors *vRd* for available current on VBUS.
 - Source monitors CC for detach and when detected, enters *Unattached.SRC*.
 - DRP (as Sink) monitors VBUS for detach and when detected, enters *Unattached.SNK* (and resumes toggling between *Unattached.SNK* and *Unattached.SRC*).

4.5.3.1.3 DRP to Sink Behavior

Figure 4-24 illustrates the functional model for a DRP connected to a Sink. The single CC wire that is in a standard cable is only shown in one of the four possible connection routes, CC1 to CC1.

Figure 4-24 DRP to Sink Functional Model



The following describes the behavior when a DRP is connected to a Sink.

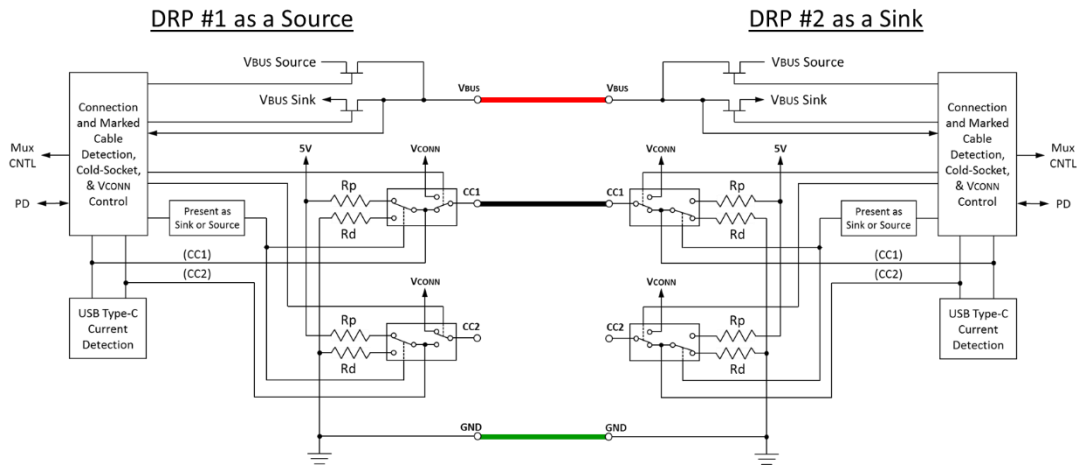
1. DRP and Sink in the unattached state.
 - DRP alternates between *Unattached.SRC* and *Unattached.SNK*.
2. DRP transitions from *Unattached.SRC* to *AttachWait.SRC* to *Attached.SRC*.
 - DRP in *Unattached.SRC* detects one of the CC pull-downs of Sink which is in *Unattached.SNK* and DRP enters *AttachWait.SRC*.
 - DRP in *AttachWait.SRC* detects that pull down on CC persists for *tCCDebounce*. It then enters *Attached.SRC* and turns on VBUS and VCONN.
3. Sink transitions from *Unattached.SNK* to *Attached.SNK* through *AttachWait.SNK* if required.
 - Sink detects VBUS and enters *Attached.SNK*.
4. While the DRP and Sink are in their respective attached states:
 - DRP (as Source) adjusts *Rp* as needed to limit the current the Sink *may* draw.
 - Sink detects and monitors *vRd* for available current on VBUS.
 - DRP (as Source) monitors CC for detach and when detected, enters *Unattached.SNK* (and resumes toggling between *Unattached.SNK* and *Unattached.SRC*).
 - Sink monitors VBUS for detach and when detected, enters *Unattached.SNK*.

4.5.3.1.4 DRP to DRP Behavior

Two behavior descriptions based on the connection state diagrams are provided below. In the first case, the two DRPs accept the resulting Source-to-Sink relationship achieved randomly whereas in the second case the DRP #2 chooses to drive the random result to the opposite result using the *Try.SRC* mechanism.

Figure 4-25 illustrates the functional model for a DRP connected to a DRP in the first case described. The single CC wire that is in a standard cable is only shown in one of the four possible connection routes, CC1 to CC1. Port numbers have been arbitrarily assigned in the diagram to assist the reader to understand the process description.

Figure 4-25 DRP to DRP Functional Model – CASE 1

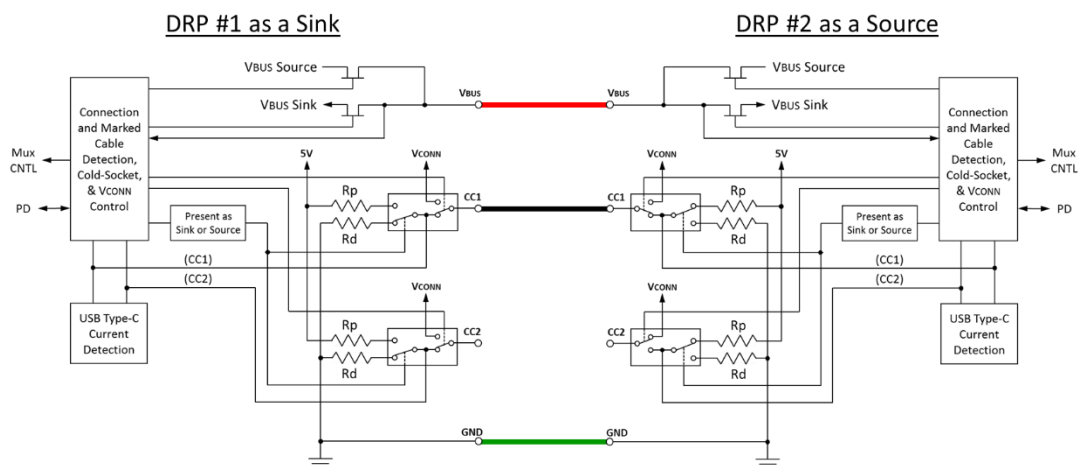


CASE 1: The following describes the behavior when a DRP is connected to another DRP. In this flow, the two DRPs accept the resulting Source-to-Sink relationship achieved randomly.

1. Both DRPs in the unattached state.
 - DRP #1 and DRP #2 alternate between *Unattached.SRC* and *Unattached.SNK*.
2. DRP #1 transitions from *Unattached.SRC* to *AttachWait.SRC*.
 - DRP #1 in *Unattached.SRC* detects a CC pull down of DRP #2 in *Unattached.SNK* and enters *AttachWait.SRC*.
3. DRP #2 transitions from *Unattached.SNK* to *AttachWait.SNK*.
 - DRP #2 in *Unattached.SNK* detects pull up on a CC and enters *AttachWait.SNK*.
4. DRP #1 transitions from *AttachWait.SRC* to *Attached.SRC*.
 - DRP #1 in *AttachWait.SRC* continues to see CC pull down of DRP #2 for *tCCDebounce*, enters *Attached.SRC* and turns on VBUS and VCONN.
5. DRP #2 transitions from *AttachWait.SNK* to *Attached.SNK*.
 - DRP #2 after having been in *AttachWait.SNK* for *tCCDebounce* and having detected VBUS, enters *Attached.SNK*.
6. While the DRPs are in their respective attached states:
 - DRP #1 (as Source) adjusts *Rp* as needed to limit the current DRP #2 (as Sink) **may** draw.
 - DRP #2 (as Sink) detects and monitors *vRd* for available current on VBUS.
 - DRP #1 (as Source) monitors CC for detach and when detected, enters *Unattached.SNK* (and resumes toggling between *Unattached.SNK* and *Unattached.SRC*).
 - DRP #2 (as Sink) monitors VBUS for detach and when detected, enters *Unattached.SNK* (and resumes toggling between *Unattached.SNK* and *Unattached.SRC*).

Figure 4-26 illustrates the functional model for a DRP connected to a DRP in the second case described.

Figure 4-26 DRP to DRP Functional Model – CASES 2 & 3



CASE 2: The following describes the behavior when a DRP is connected to another DRP. In this flow, the DRP #2 chooses to drive the random result to the opposite result using the *Try.SRC* mechanism.

1. Both DRPs in the unattached state.
 - DRP #1 and DRP #2 alternate between *Unattached.SRC* and *Unattached.SNK*.
2. DRP #1 transitions from *Unattached.SRC* to *AttachWait.SRC*.
 - DRP #1 in *Unattached.SRC* detects a CC pull down of DRP #2 in *Unattached.SNK* and enters *AttachWait.SRC*.
3. DRP #2 transitions from *Unattached.SNK* to *AttachWait.SNK*.
 - DRP #2 in *Unattached.SNK* detects pull up on a CC and enters *AttachWait.SNK*.
4. DRP #1 transitions from *AttachWait.SRC* to *Attached.SRC*.
 - DRP #1 in *AttachWait.SRC* continues to see CC pull down of DRP #2 for *tCCDebounce*, enters *Attached.SRC* and turns on VBUS and VCONN.
5. DRP #2 transitions from *AttachWait.SNK* to *Try.SRC*.
 - DRP #2 in *AttachWait.SNK* has been in this state for *tCCDebounce* and detects VBUS but strongly prefers the Source role, so transitions to *Try.SRC*.
 - DRP #2 in *Try.SRC* asserts a pull-up on CC and waits.
6. DRP #1 transitions from *Attached.SRC* to *Unattached.SNK* to *AttachWait.SNK*.
 - DRP #1 in *Attached.SRC* no longer detects DRP #2's pull-down on CC and transitions to *Unattached.SNK*.
 - DRP #1 in *Unattached.SNK* turns off VBUS and VCONN and applies a pull-down on CC.
 - DRP #1 in *Unattached.SNK* detects pull up on a CC and enters *AttachWait.SNK*.
7. DRP #2 transitions from *Try.SRC* to *Attached.SRC*.
 - DRP #2 in *Try.SRC* detects the DRP #1 in *Unattached.SNK*'s pull-down on CC and enters *Attached.SRC*.
 - DRP #2 in *Attached.SRC* turns on VBUS and VCONN.

8. DRP #1 transitions from *AttachWait.SNK* to *Attached.SNK*.
 - DRP #1 in *AttachWait.SNK* after *tCCDebounce* and detecting VBUS, enters *Attached.SNK*.
9. While the DRPs are in their respective attached states:
 - DRP #2 (as Source) adjusts *Rp* as needed to limit the current DRP #1 (as Sink) *may* draw.
 - DRP #1 (as Sink) detects and monitors *vRd* for available current on VBUS.
 - DRP #2 (as Source) monitors CC for detach and when detected, enters *Unattached.SNK* (and resumes toggling between *Unattached.SNK* and *Unattached.SRC*).
 - DRP #1 (as Sink) monitors VBUS for detach and when detected, enters *Unattached.SNK* (and resumes toggling between *Unattached.SNK* and *Unattached.SRC*).

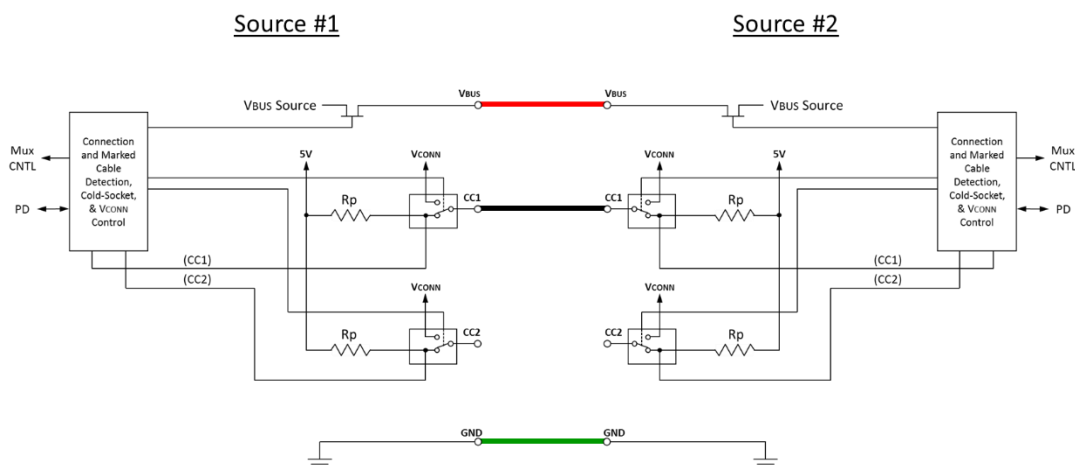
CASE 3: The following describes the behavior when a DRP is connected to another DRP. In this flow, the DRP #1 chooses to drive the random result to the opposite result using the *Try.SNK* mechanism.

1. Both DRPs in the unattached state.
 - DRP #1 and DRP #2 alternate between *Unattached.SRC* and *Unattached.SNK*.
2. DRP #1 transitions from *Unattached.SRC* to *AttachWait.SRC*.
 - DRP #1 in *Unattached.SRC* detects a CC pull down of DRP #2 in *Unattached.SNK* and enters *AttachWait.SRC*.
3. DRP #2 transitions from *Unattached.SNK* to *AttachWait.SNK*.
 - DRP #2 in *Unattached.SNK* detects pull up on a CC and enters *AttachWait.SNK*.
4. DRP #1 transitions from *AttachWait.SRC* to *Try.SNK*.
 - DRP #1 in *AttachWait.SRC* has been in this state for *tCCDebounce* and detects DRP #2's pull-down on CC but strongly prefers the Sink role, so transitions to *Try.SNK*.
 - DRP #1 in *Try.SNK* asserts a pull down on CC and waits.
5. DRP #2 transitions from *AttachWait.SNK* to *Unattached.SRC* to *AttachWait.SRC*.
 - DRP #2 in *AttachWait.SNK* no longer detects DRP #1's pull up on CC and transitions to *Unattached.SRC*.
 - DRP #2 in *Unattached.SRC* applies a pull up on CC.
 - DRP #2 in *Unattached.SRC* detects a pull down on a CC pin and enters *AttachWait.SRC*.
 - DRP #1 detects DRP #2's pull up on CC and remains in *Try.SNK*.
6. DRP #2 transitions from *AttachWait.SRC* to *Attached.SRC*.
 - DRP #2 in *AttachWait.SRC* times out (*tCCDebounce*) and transitions to *Attached.SRC*.
 - DRP #2 in *Attached.SRC* turns on VBUS and VCONN.
7. DRP #1 transitions from *Try.SNK* to *Attached.SNK*.
 - DRP #1 in *Try.SNK* after detecting VBUS, enters *Attached.SNK*.
8. While the DRPs are in their respective attached states:
 - DRP #2 (as Source) adjusts *Rp* as needed to limit the current DRP #1 (as Sink) *may* draw.
 - DRP #1 (as Sink) detects and monitors *vRd* for available current on VBUS.
 - DRP #2 (as Source) monitors CC for detach and when detected, enters *Unattached.SNK* (and resumes toggling between *Unattached.SNK* and *Unattached.SRC*).
 - DRP #1 (as Sink) monitors VBUS for detach and when detected, enters *Unattached.SNK* (and resumes toggling between *Unattached.SNK* and *Unattached.SRC*).

4.5.3.1.5 Source to Source Behavior

Figure 4-27 illustrates the functional model for a Source connected to a Source. The single CC wire that is in a standard cable is only shown in one of the four possible connection routes, CC1 to CC1. Port numbers have been arbitrarily assigned in the diagram to assist the reader to understand the process description.

Figure 4-27 Source to Source Functional Model



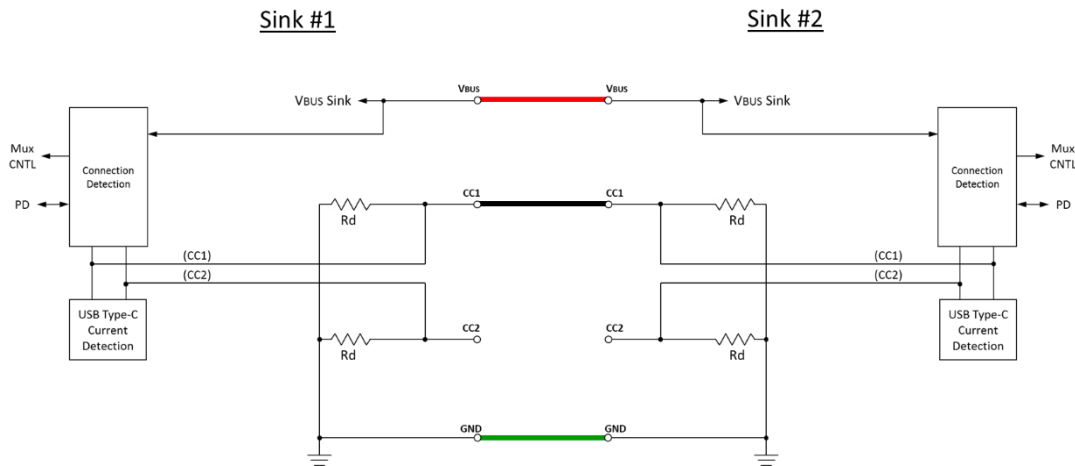
The following describes the behavior when a Source is connected to another Source.

1. Both Sources in the unattached state.
 - Source #1 fails to detect a Sink's pull-down on CC and remains in *Unattached.SRC*.
 - Source #2 fails to detect a Sink's pull-down on CC and remains in *Unattached.SRC*.

4.5.3.1.6 Sink to Sink Behavior

Figure 4-28 illustrates the functional model for a Sink connected to a Sink. The single CC wire that is in a standard cable is only shown in one of the four possible connection routes, CC1 to CC1. Port numbers have been arbitrarily assigned in the diagram to assist the reader to understand the process description.

Figure 4-28 Sink to Sink Functional Model



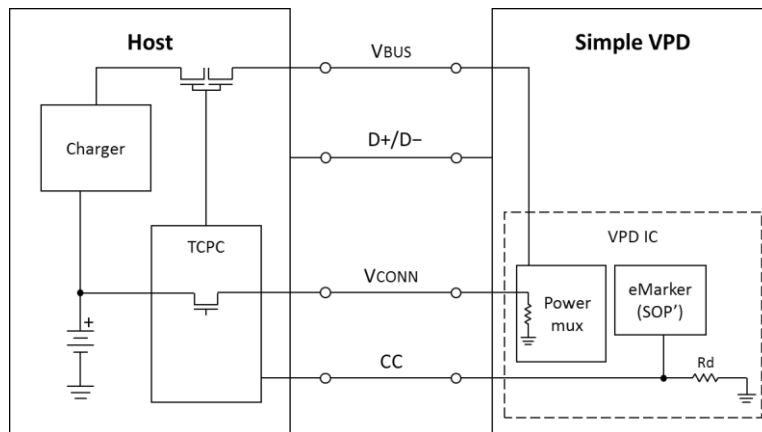
The following describes the behavior when a Sink is connected to another Sink.

1. Both Sinks in the unattached state.
 - Sink #1 fails to detect pull up on CC or VBUS supplied by a Source and remains in *Unattached.SNK*.
 - Sink #2 fails to detect pull up on CC or VBUS supplied by a Source and remains in *Unattached.SNK*.

4.5.3.1.7 DRP to VCONN-Powered USB Device (VPD) Behavior

Figure 4-29 illustrates the functional model for a DRP connected to a *VCONN-Powered USB Device* that does not feature charge-through functionality.

Figure 4-29 DRP to VPD Model



The following describes the behavior when a DRP that supports *VPDs* is connected to a *VPD*.

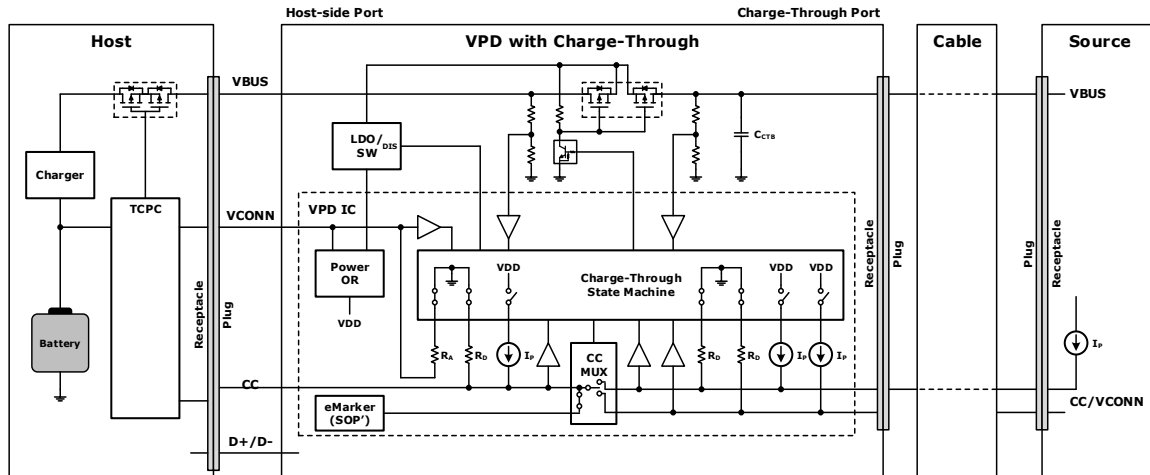
1. DRP and *VPD* in the unattached state.
 - DRP alternates between *Unattached.SRC* and *Unattached.SNK*.
2. DRP transitions from *Unattached.SRC* to *Attached.SRC* through *AttachWait.SRC*.
 - DRP in *Unattached.SRC* detects the CC pull-down of *VPD* which is in *Unattached.SNK* and DRP enters *AttachWait.SRC*.
 - DRP in *AttachWait.SRC* detects that pull-down on CC persists for *tCCDebounce*. It then enters *Attached.SRC* and turns on VBUS and VCONN.
3. *VPD* transitions from *Unattached.SNK* to *Attached.SNK* through *AttachWait.SNK*.
 - *VPD* detects VCONN and enters *Attached.SNK*.
4. While DRP and *VPD* are in their respective attached states, DRP discovers the *VPD* and removes VBUS.
 - DRP (as Source) queries the cable identity via *USB PD* on SOP'.
 - *VPD* responds on SOP', advertising that it is a *VCONN-Powered USB Device* that does not support charge-through.
 - DRP (as Source) removes VBUS.
 - DRP (as Source) maintains its *Rp*.
5. DRP and *VPD* monitor for detach.
 - DRP (as Source) monitors CC for detach and when detected, enters *Unattached.SNK* (and resumes toggling between *Unattached.SNK* and *Unattached.SRC*).

- **VPD** monitors VCONN for detach and when detected, enters **Unattached.SNK**.

4.5.3.1.8 DRP to Charge-Through VCONN-Powered Device (CTVPD) Behavior

Figure 4-30 illustrates the functional model for a DRP connected to a Charge-Through **VCONN-Powered USB Device**, with a Source attached to the Charge-Through port on the **VCONN-Powered USB Device**.

Figure 4-30 Example DRP to Charge-Through VPD Model



CASE 1: The following describes the behavior when a DRP is connected to a Charge-Through **VCONN-Powered USB Device** (abbreviated **CTVPD**), with no Power Source attached to the Charge-Through port on the **CTVPD**.

1. DRP and **CTVPD** are both in the unattached state.
 - a. DRP alternates between **Unattached.SRC** and **Unattached.SNK**.
 - b. **CTVPD** has applied **Rd** on its Charge-Through port's CC1 and CC2 pins and **Rd** on the Host-side port's CC pin.
2. DRP transitions from **Unattached.SRC** to **Attached.SRC** through **AttachWait.SRC**.
 - a. DRP in **Unattached.SRC** detects a CC pull down of **CTVPD** which is in **Unattached.SNK** and DRP enters **AttachWait.SRC**.
 - b. DRP in **AttachWait.SRC** detects that pull down on CC persists for **tCCDebounce**, enters **Attached.SRC** and turns on VBUS and VCONN.
3. **CTVPD** transitions from **Unattached.SNK** to **Attached.SNK** through **AttachWait.SNK**.
 - a. **CTVPD** detects the host-side CC pull-up of the DRP and **CTVPD** enters **AttachWait.SNK**.
 - b. **CTVPD** in **AttachWait.SNK** detects that pull up on the Host-side port's CC persists for **tCCDebounce**, VCONN present and enters **Attached.SNK**.
 - c. **CTVPD** presents a high-impedance to ground (above **zOPEN**) on its Charge-Through port's CC1 and CC2 pins.
4. While DRP and **CTVPD** are in their respective attached states, DRP discovers the **CTVPD** and transitions to **CTUnattached.SNK**.
 - a. DRP (as Source) queries the device identity via **USB PD** (Device Identity Command) on SOP'.
 - b. **CTVPD** responds on SOP', advertising that it is a Charge-Through **VCONN-Powered USB Device**.
 - c. DRP (as Source) removes VBUS.
 - d. DRP (as Source) changes its **Rp** to a **Rd**.

- e. DRP (as Sink) continues to provide VCONN and enters *CTUnattached.SNK*.
5. *CTVPD* transitions to *CTUnattached.VPD*.
 - a. *CTVPD* detects VBUS removal, VCONN presence, the low Host-side CC pin and enters *CTUnattached.VPD*.
 - b. *CTVPD* changes its host-side *Rd* to a *Rp* advertising 3.0 A.
 - c. *CTVPD* isolates itself from VBUS.
 - d. *CTVPD* apply *Rd* on its Charge-Through port's CC1 and CC2 pins.
6. While the *CTVPD* in *CTUnattached.VPD* state and the DRP in *CTUnattached.SNK* state:
 - a. *CTVPD* monitors Charge-Through CC pins for a source or sink; when a Power Source attach is detected, enters *CTAttachWait.VPD*; when a sink is detected, enters *CTAttachWait.Unsupported*.
 - b. *CTVPD* monitors VCONN for Host detach and when detected, enters *Unattached.SNK*.
 - c. DRP monitors VBUS and CC for *CTVPD* detach for *tVPDDetach* and when detected, enters *Unattached.SNK*.
 - d. DRP monitors VBUS for Power Source attach and when detected, enters *CTAttached.SNK*.

CASE 2: The following describes the behavior when a Power Source is connected to a Charge-Through *VCONN-Powered USB Device* (abbreviated *CTVPD*), with a Host already attached to the Host-side port on the *CTVPD*.

1. DRP is in *CTUnattached.SNK* state, *CTVPD* in *CTUnattached.VPD*, and Power Source in the unattached state.
 - a. *CTVPD* has applied *Rd* on the Charge-Through port's CC1 and CC2 pins and *Rp* termination advertising 3.0 A on the Host-side port's CC pin.
2. Power Source transitions from *Unattached.SRC* to *Attached.SRC* through *AttachWait.SRC*.
 - a. Power Source detects the CC pull-down of the *CTVPD* and enters *AttachWait.SRC*.
 - b. Power Source in *AttachWait.SRC* detects that pull down on CC persists for *tCCDebounce*, enters *Attached.SRC* and turns on VBUS.
3. *CTVPD* transitions from *CTUnattached.VPD* through *CTAttachWait.VPD* to *CTAttached.VPD*.
 - a. *CTVPD* detects the Source's *Rp* on one of its Charge-Through CC pins, and transitions to *CTAttachWait.VPD*.
 - b. *CTVPD* finishes any active *USB PD* communication on SOP' and ceases to respond to SOP' queries.
 - c. *CTVPD* in *CTAttachWait.VPD* detects that the pull up on Charge-Through CC pin persists for *tCCDebounce*, detects VBUS and enters *CTAttached.VPD*.
 - d. *CTVPD* connects the active Charge-Through CC pin to the Host-side port's CC pin.
 - e. *CTVPD* disables its *Rp* termination advertising 3.0 A on the Host-side port's CC pin.
 - f. *CTVPD* disables its *Rd* on the Charge-Through CC pins.
 - g. *CTVPD* connects VBUS from the Charge-Through side to the Host side.
4. DRP (as Sink) transitions to *CTAttached.SNK*.
 - a. DRP (as Sink) detects VBUS, monitors *vRd* for available current and enter *CTAttached.SNK*.
5. While the devices are all in their respective attached states:
 - a. *CTVPD* monitors VCONN for DRP detach and when detected, enters *CTDisabled.VPD*.
 - b. *CTVPD* monitors VBUS and CC for Power Source detach and when detected, enters *CTUnattached.VPD* within *tVPDCTDD*.

- c. DRP (as Sink) monitors VBUS for Charge-Through Power Source detach and when detected, enters *CTUnattached.SNK*.
- d. DRP (as Sink) monitors VBUS and CC for *CTVPD* detach and when detected, enters *Unattached.SNK* (and resumes toggling between *Unattached.SNK* and *Unattached.SRC*).
- e. Power Source monitors CC for *CTVPD* detach and when detected, enters *Unattached.SRC*.

CASE 3: The following describes the behavior when a Power Source is connected to a Charge-Through *VCONN-Powered USB Device* (abbreviated *CTVPD*), with no Host attached to the Host-side port on the *CTVPD*.

1. *CTVPD* and Power Source are both in the unattached state.
 - a. *CTVPD* has applied *Rd* on the Charge-Through port's CC1 and CC2 pins and *Rd* on the Host-side port's CC pin.
2. Power Source transitions from *Unattached.SRC* to *Attached.SRC* through *AttachWait.SRC*.
 - a. Power Source detects the CC pull-down of the *CTVPD* and enters *AttachWait.SRC*.
 - b. Power Source in *AttachWait.SRC* detects that pull down on CC persists for *tCCDebounce*, enters *Attached.SRC* and turns on VBUS.
3. *CTVPD* alternates between *Unattached.SNK* and *Unattached.SRC*.
 - a. *CTVPD* detects the Source's *Rp* on one of its Charge-Through CC pins, detects VBUS for *tCCDebounce* and starts alternating between *Unattached.SRC* and *Unattached.SNK*.
4. While the *CTVPD* alternates between *Unattached.SRC* and *Unattached.SNK* state and the Power Source in *Attached.SRC* state:
 - a. *CTVPD* monitors the Host-side port's CC pin for device attach and when detected, enters *AttachWait.SRC*.
 - b. *CTVPD* monitors VBUS for Power Source detach and when detected, enters *Unattached.SNK*.
 - c. Power Source monitors CC for *CTVPD* detach and when detected, enters *Unattached.SRC*.

CASE 4: The following describes the behavior when a DRP is connected to a Charge-Through *VCONN-Powered USB Device* (abbreviated *CTVPD*), with a Power Source already attached to the Charge-Through side on the *CTVPD*.

1. DRP and *CTVPD* are in unattached state and Power Source in *Attached.SRC* state.
 - a. DRP alternates between *Unattached.SRC* and *Unattached.SNK*.
 - b. *CTVPD* alternates between *Unattached.SRC* and *Unattached.SNK*.
 - c. *CTVPD* has applied *Rd* on its Charge-Through port's CC1 and CC2 pins.
 - d. Power Source has applied VBUS.
2. DRP transitions from *Unattached.SNK* to *AttachWait.SNK*.
 - a. DRP in *Unattached.SNK* detects the CC pull-up of *CTVPD* which is in *Unattached.SRC* and DRP enters *AttachWait.SNK*.
3. *CTVPD* transitions from *Unattached.SRC* to *Try.SNK* through *AttachWait.SRC*.
 - a. *CTVPD* in *Unattached.SRC* detects the CC pull-down of DRP which is in *Unattached.SNK* and *CTVPD* enters *AttachWait.SRC*.
 - b. *CTVPD* in *AttachWait.SRC* detects that pull down on CC persists for *tCCDebounce* and enters *Try.SNK*.
 - c. *CTVPD* disables *Rp* termination advertising Default USB Power on the Host-side port's CC pin.
 - d. *CTVPD* enables *Rd* on the Host-side port's CC pin.

4. DRP transitions from *AttachWait.SNK* to *Attached.SRC* through *Unattached.SRC* and *AttachWait.SRC*.
 - a. DRP in *AttachWait.SNK* detects the CC pull-up removal of *CTVPD* which is in *Try.SNK* and DRP enters *Unattached.SRC*.
 - b. DRP in *Unattached.SRC* detects the CC pull-down of *CTVPD* which is in *Try.SNK* and DRP enters *AttachWait.SRC*.
 - c. DRP in *AttachWait.SRC* detects that pull down on CC persists for *tCCDebounce*. It then enters *Attached.SRC* and enables VBUS and VCONN.
5. *CTVPD* transitions from *Try.SNK* to *Attached.SNK*.
 - a. *CTVPD* detects the CC pull-up of the DRP persists for *tTryCCDebounce*.
 - b. *CTVPD* detects VBUS on the Host-side port and enters *Attached.SNK*.
6. While DRP and *CTVPD* are in their respective attached states, DRP discovers the Charge-Through *CTVPD* and transitions to *CTUnattached.SNK*.
 - a. DRP (as Source) queries the device identity via **USB PD (Discover Identity Command)** on SOP'.
 - b. *CTVPD* responds on SOP', advertising that it is a Charge-Through **VCONN-Powered USB Device**.
 - c. DRP (as Source) removes VBUS.
 - d. DRP (as Source) changes its *Rp* into an *Rd*.
 - e. DRP (as Sink) continues to provide VCONN and enters *CTUnattached.SNK*.
7. *CTVPD* transitions to *CTUnattached.VPD*.
 - a. *CTVPD* detects VBUS removal, VCONN presence, and the low CC pin on its host port and enters *CTUnattached.VPD*.
 - b. *CTVPD* changes its host-side *Rd* into an *Rp* termination advertising 3.0 A.
 - c. *CTVPD* isolates itself from VBUS.
8. *CTVPD* transitions from *CTUnattached.VPD* through *CTAttachWait.VPD* to *CTAttached.VPD*.
 - a. *CTVPD* detects the Source's *Rp* on one of its Charge-Through CC pins, and transitions to *CTAttachWait.VPD*.
 - b. *CTVPD* in *CTAttachWait.VPD* detects that the pull up on Charge-Through CC pin persists for *tCCDebounce*, detects VBUS and enters *CTAttached.VPD*.
 - c. *CTVPD* finishes any active **USB PD** communication on SOP' and ceases to respond to SOP' queries.
 - d. *CTVPD* connects the active Charge-Through CC pin to the Host-side port's CC pin.
 - e. *CTVPD* disables its *Rp* termination advertising 3.0 A on the Host-side port's CC pin.
 - f. *CTVPD* disables its *Rd* on the Charge-Through CC pins.
 - g. *CTVPD* connects VBUS from the Charge-Through side to the Host side.
9. DRP (as Sink) transitions to *CTAttached.SNK*.
 - a. DRP (as Sink) detects VBUS and monitors *vRd* for available current and enter *CTAttached.SNK*.
10. While the devices are all in their respective attached states:
 - a. *CTVPD* monitors VCONN for DRP detach and when detected, enters *CTDisabled.VPD*.

- b. **CTVPD** monitors VBUS and CC for Power Source detach and when detected, enters **CTUnattached.VPD** within **tVPDCTDD**.
- c. DRP (as Sink) monitors VBUS for Charge-Through Power Source detach and when detected, enters **CTUnattached.SNK**.
- d. DRP (as Sink) monitors VBUS and CC for **CTVPD** detach and when detected, enters **Unattached.SNK** (and resumes toggling between **Unattached.SNK** and **Unattached.SRC**).
- e. Power Source monitors CC for **CTVPD** detach and when detected, enters **Unattached.SRC**.

CASE 5: The following describes the behavior when a Power Source is connected to a Charge-Through **VCONN-Powered USB Device** (abbreviated **CTVPD**), with a DRP (with dead battery) attached to the Host-side port on the **CTVPD**.

1. DRP, **CTVPD** and Power Source are all in the unattached state.
 - a. DRP apply dead battery **Rd**.
 - b. **CTVPD** apply **Rd** on the Charge-Through port's CC1 and CC2 pins and **Rd** on the Host-side port's CC pin.
2. Power Source transitions from **Unattached.SRC** to **Attached.SRC** through **AttachWait.SRC**.
 - a. Power Source detects the CC pull-down of the **CTVPD** and enters **AttachWait.SRC**.
 - b. Power Source in **AttachWait.SRC** detects that pull down on CC persists for **tCCDebounce**, enters **Attached.SRC** and turns on VBUS.
3. **CTVPD** alternates between **Unattached.SNK** and **Unattached.SRC**.
 - a. **CTVPD** detects the Source's **Rp** on one of its Charge-Through CC pins, detects VBUS for **tCCDebounce** and starts alternating between **Unattached.SRC** and **Unattached.SNK**.
4. **CTVPD** transitions from **Unattached.SRC** to **Try.SNK** through **AttachWait.SRC**.
 - a. **CTVPD** in **Unattached.SRC** detects the CC pull-down of DRP which is in **Unattached.SNK** and **CTVPD** enters **AttachWait.SRC**.
 - b. **CTVPD** in **AttachWait.SRC** detects that pull down on CC persists for **tCCDebounce** and enters **Try.SNK**.
 - c. **CTVPD** disables **Rp** termination advertising Default USB Power on the Host-side port's CC pin.
 - d. **CTVPD** enables **Rd** on the Host-side port's CC pin.
5. DRP in dead battery condition remains in **Unattached.SNK**.
6. **CTVPD** transitions from **Try.SNK** to **Attached.SRC** through **TryWait.SRC**.
 - a. **CTVPD** didn't detect the CC pull-up of the DRP for **tTryCCDebounce** after **tDRPTry** and enters **TryWait.SRC**.
 - b. **CTVPD** disables **Rp** on the Host-side port's CC pin.
 - c. **CTVPD** enables **Rp** termination advertising Default USB Power on the Host-side port's CC pin.
 - d. **CTVPD** detects the CC pull-down of the DRP for **tTryCCDebounce** and enters **Attached.SRC**.
 - e. **CTVPD** connects VBUS from the Charge-Through side to the Host side.
7. DRP transitions from **Unattached.SNK** to **Attached.SNK** through **AttachWait.SNK**.
 - a. DRP in **Unattached.SNK** detects the CC pull-up of **CTVPD** which is in **Attached.SRC** and DRP enters **AttachWait.SNK**.
 - b. DRP in **AttachWait.SNK** detects that pull up on CC persists for **tCCDebounce**, VBUS present and enters **Attached.SNK**.

8. While the devices are all in their respective attached states:
 - a. **CTVPD** monitors the Host-side port's CC pin for device attach and when detected, enters **Unattached.SNK**.
 - b. **CTVPD** monitors VBUS for Power Source detach and when detected, enters **Unattached.SNK**.
 - c. Power Source monitors CC for **CTVPD** detach and when detected, enters **Unattached.SRC**.
 - d. DRP monitors VBUS for **CTVPD** detach and when detected, enters **Unattached.SNK**.
 - e. Additionally, the DRP *may* query the identity of the cable via **USB PD** on SOP' when it has sufficient battery power and when a Charge-Through VPD is identified enters **TryWait.SRC** if implemented or enters **Unattached.SRC** if **TryWait.SRC** is not supported.

CASE 6: The following describes the behavior when a DRP is connected to a Charge-Through **VCONN-Powered USB Device** (abbreviated **CTVPD**) and a Sink is attached to the Charge-Through port on the **CTVPD**.

1. DRP, **CTVPD** and Sink are all in the unattached state.
 - a. DRP alternates between **Unattached.SRC** and **Unattached.SNK**.
 - b. **CTVPD** has applied **Rd** on its Charge-Through port's CC1 and CC2 pins and **Rd** on the Host-side port's CC pin.
2. DRP transitions from **Unattached.SRC** to **Attached.SRC** through **AttachWait.SRC**.
 - a. DRP in **Unattached.SRC** detects the CC pull-down of **CTVPD** which is in **Unattached.SNK** and DRP enters **AttachWait.SRC**.
 - b. DRP in **AttachWait.SRC** detects that pull down on CC persists for **tCCDebounce**. It then enters **Attached.SRC** and enables VBUS and VCONN.
3. **CTVPD** transitions from **Unattached.SNK** to **Attached.SNK** through **AttachWait.SNK**.
 - a. **CTVPD** detects the host-side CC pull-up of the DRP and **CTVPD** enters **AttachWait.SNK**.
 - b. **CTVPD** in **AttachWait.SNK** detects that pull up on the Host-side port's CC persists for **tCCDebounce**, VCONN present and enters **Attached.SNK**.
 - c. **CTVPD** presents a high-impedance to ground (above **zOPEN**) on its Charge-Through port's CC1 and CC2 pins.
4. While DRP and **CTVPD** are in their respective attached states, DRP discovers the Charge-Through **CTVPD** and transitions to **CTUnattached.SNK**.
 - a. DRP (as Source) queries the device identity via **USB PD** (**Discover Identity** Command) on SOP'.
 - b. **CTVPD** responds on SOP', advertising that it is a Charge-Through **VCONN-Powered USB Device**.
 - c. DRP (as Source) removes VBUS.
 - d. DRP (as Source) changes its **Rp** into an **Rd**.
 - e. DRP (as Sink) continues to provide VCONN and enters **CTUnattached.SNK**.
5. **CTVPD** transitions to **CTUnattached.VPD**.
 - a. **CTVPD** detects VBUS removal, VCONN presence, and the low CC pin on its host port and enters **CTUnattached.VPD**.
 - b. **CTVPD** changes its host-side **Rd** into an **Rp** termination advertising 3.0 A.
 - c. **CTVPD** isolates itself from VBUS.
 - d. **CTVPD** apply **Rd** on its Charge-Through port's CC1 and CC2 pins.
6. **CTVPD** alternates between **CTUnattached.VPD** and **CTUnattached.Unsupported**.

- a. **CTVPD** detects **SRC.Open** on its Charge-Through CC pins and starts alternating between **CTUnattached.VPD** and **CTUnattached.Unsupported**.
7. **CTVPD** transitions from **CTUnattached.Unsupported** to **CTTry.SNK** through **CTAttachWait.Unsupported**.
 - a. **CTVPD** in **CTUnattached.Unsupported** detects the CC pull-down of the Sink which is in **Unattached.SNK** and **CTVPD** enters **CTAttachWait.Unsupported**.
 - b. **CTVPD** in **CTAttachWait.Unsupported** detects that pull down on CC persists for **tCCDebounce** and enters **CTTry.SNK**.
 - c. **CTVPD** disables **Rp** termination advertising Default USB Power on the Charge-Through port's CC pins.
 - d. **CTVPD** enables **Rd** on the Charge-Through port's CC pins.
8. **CTVPD** transitions from **CTTry.SNK** to **CTAttached.Unsupported**.
 - a. **CTVPD** didn't detect the CC pull-up of the potential Source for **tDRPTryWait** after **tDRPTry** and enters **CTAttached.Unsupported**.
9. While the **CTVPD** in **CTAttached.Unsupported** state, the DRP in **CTUnattached.SNK** state and the Sink in **Unattached.SNK** state:
 - a. **CTVPD** disables the **Rd** termination on the Charge-Through port's CC pins and applies **Rp** termination advertising Default USB Power.
 - b. **CTVPD** exposes a **USB Billboard Device Class** to the DRP indicating that it is connected to an unsupported device on its Charge Through port.
 - c. **CTVPD** monitors Charge-Through CC pins for Sink detach and when detected, enters **CTUnattached.VPD**.
 - d. **CTVPD** monitors VCONN for Host detach and when detected, enters **Unattached.SNK**
 - e. DRP monitors CC for **CTVPD** detach for **tVPDDetach** and when detected, enters **Unattached.SNK**.
 - f. DRP monitors VBUS for **CTVPD** Charge-Through source attach and, when detected, enters **CTAttached.SNK**.

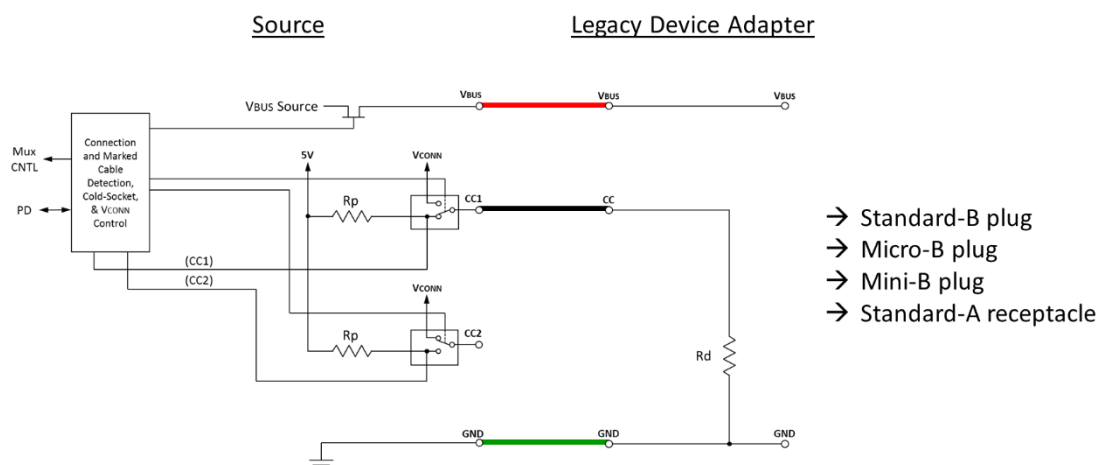
4.5.3.2 USB Type-C Port to USB Legacy Port Interoperability Behaviors

The following sub-sections describe port-to-port interoperability behaviors for the various combinations of USB Type-C Source, Sink and DRPs and legacy USB ports.

4.5.3.2.1 Source to Legacy Device Port Behavior

Figure 4-31 illustrates the functional model for a Source connected to a legacy device port. This model is based on having an adapter present as a Sink to the Source. This adapter has a USB Type-C plug on one end plugged into the Source and either a USB Standard-B plug, USB Micro-B plug, USB Mini-B plug, or a USB Standard-A receptacle on the other end.

Figure 4-31 Source to Legacy Device Port Functional Model



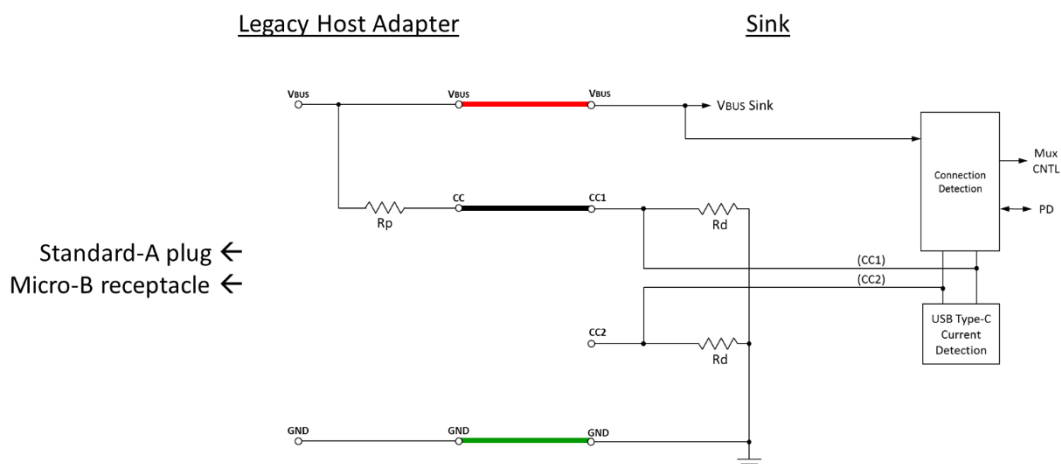
The following describes the behavior when a Source is connected to a legacy device adapter that has an *Rd* to ground to mimic the behavior of a Sink.

1. Source in the unattached state.
2. Source transitions from *Unattached.SRC* to *Attached.SRC* through *AttachWait.SRC*.
 - Source detects the Sink's pull-down on CC and enters *AttachWait.SRC*. After *tCCDebounce*, it enters *Attached.SRC*.
 - Source turns on VBUS and VCONN.
3. While the Source is in the attached state:
 - Source monitors CC for detach and when detected, enters *Unattached.SRC*.

4.5.3.2.2 Legacy Host Port to Sink Behavior

Figure 4-32 illustrates the functional model for a legacy host port connected to a Sink. This model is based on having an adapter that presents itself as a Source to the Sink. This adapter is either a USB Standard-A legacy plug or a USB Micro-B legacy receptacle on one end and the USB Type-C plug on the other end plugged into a Sink.

Figure 4-32 Legacy Host Port to Sink Functional Model



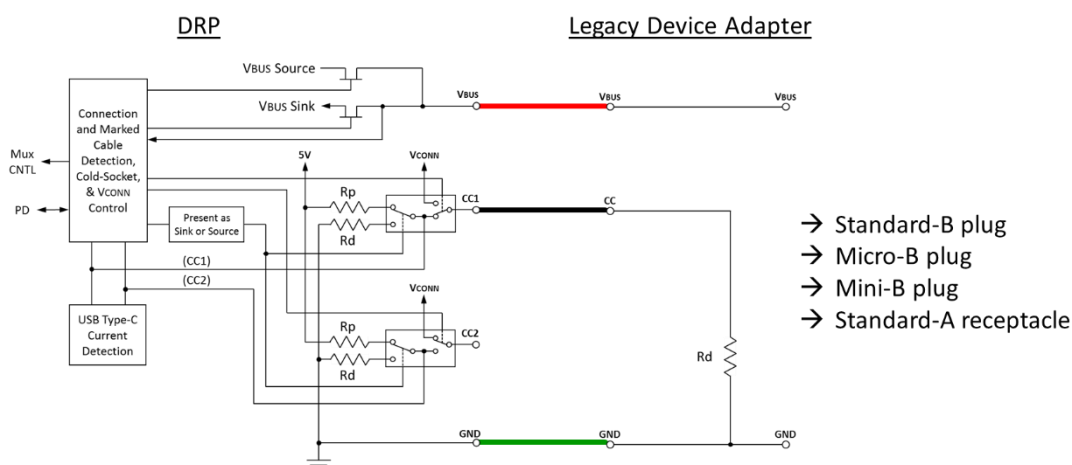
The following describes the behavior when a legacy host adapter that has an ***R_p*** to VBUS to mimic the behavior of a Source that is connected to a Sink. The value of ***R_p*** *shall* indicate an advertisement of Default USB Power (See Table 4-27), even though the cable itself can carry 3 A. This is because the cable has no knowledge of the capabilities of the power source, and any higher current is negotiated via **USB BC 1.2**.

1. Sink in the unattached state.
2. Sink transitions from ***Unattached.SNK*** to ***Attached.SNK*** through ***AttachWait.SNK*** if needed.
 - While in ***Unattached.SNK***, if device is not **USB 2.0** only, supports accessories or requires more than default power, it enters ***AttachWait.SNK*** when it detects a pull up on CC and ignores VBUS. Otherwise, it *may* enter ***Attached.SNK*** directly when VBUS is detected.
 - Sink detects VBUS and enters ***Attached.SNK***.
3. While the Sink is in the attached state:
 - Sink monitors VBUS for detach and when detected, enters ***Unattached.SNK***.

4.5.3.2.3 DRP to Legacy Device Port Behavior

Figure 4-33 illustrates the functional model for a DRP connected to a legacy device port. This model is based on having an adapter present as a Sink (Device) to the DRP. This adapter has a USB Type-C plug on one end plugged into a DRP and either a USB Standard-B plug, USB Micro-B plug, USB Mini-B plug, or a USB Standard-A receptacle on the other end.

Figure 4-33 DRP to Legacy Device Port Functional Model



The following describes the behavior when a DRP is connected to a legacy device adapter that has an ***R_d*** to ground to mimic the behavior of a Sink.

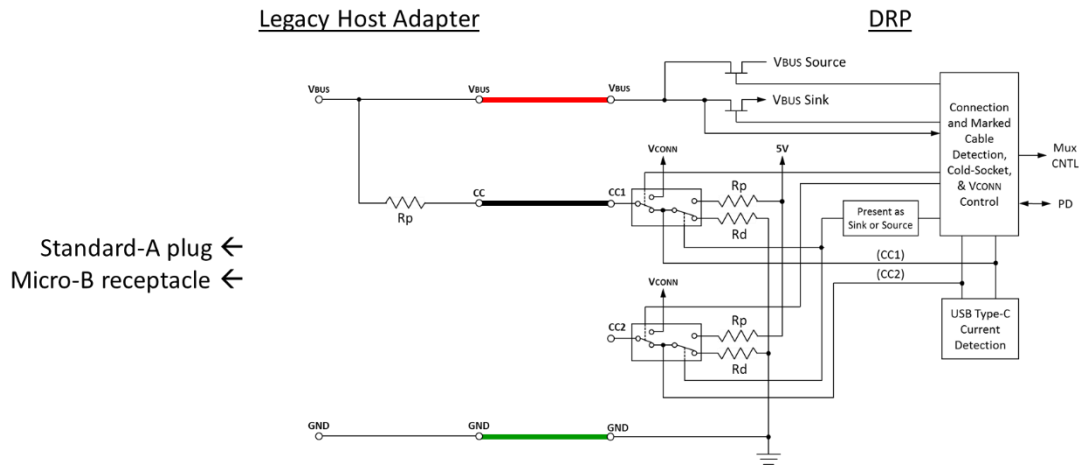
1. DRP in the unattached state.
 - DRP alternates between ***Unattached.SRC*** and ***Unattached.SNK***.
2. DRP transitions from ***Unattached.SRC*** to ***Attached.SRC***.
 - DRP in ***Unattached.SRC*** detects the adapter's pull-down on CC and enters ***AttachWait.SRC***.
 - DRP in ***AttachWait.SRC*** times out (***tCCDebounce***) and transitions to ***Attached.SRC***.
 - DRP in ***Attached.SRC*** turns on VBUS and VCONN.
 - DRP in ***AttachWait.SRC*** *may* support ***Try.SNK*** and if so, *may* transition through ***Try.SNK*** and ***TryWait.SRC*** prior to entering ***Attached.SRC***.
3. While the DRP is in the attached state:

- DRP monitors CC for detach and when detected, enters **Unattached.SNK** (and resumes toggling between **Unattached.SNK** and **Unattached.SRC**).

4.5.3.2.4 Legacy Host Port to DRP Behavior

Figure 4-34 illustrates the functional model for a legacy host port connected to a DRP operating as a Sink. This model is based on having an adapter that presents itself as a Source (Host) to the DRP operating as a Sink, this adapter is either a USB Standard-A legacy plug or a USB Micro-B legacy receptacle on one end and the USB Type-C plug on the other end plugged into a DRP.

Figure 4-34 Legacy Host Port to DRP Functional Model



The following describes the behavior when a legacy host adapter that has an **Rp** to VBUS to mimic the behavior of a Source is connected to a DRP. The value of **Rp** *shall* indicate an advertisement of Default USB Power (See Table 4-27), even though the cable itself can carry 3 A. This is because the cable has no knowledge of the capabilities of the power source, and any higher current is negotiated via **USB BC 1.2**.

1. DRP in the unattached state.
 - DRP alternates between **Unattached.SRC** and **Unattached.SNK**.
2. DRP transitions from **Unattached.SNK** to **AttachWait.SNK** to **Attached.SNK**.
 - DRP in **Unattached.SNK** detects pull up on CC and enters **AttachWait.SNK**.
 - DRP in **AttachWait.SNK** detects VBUS and enters **Attached.SNK**.
 - DRP in **AttachWait.SNK** *may* support **Try.SRC** and if so, *may* transition through **Try.SRC** and **TryWait.SNK** prior to entering **Attached.SNK**.
3. While the DRP is in the attached state:
 - DRP monitors VBUS for detach and when detected, enters **Unattached.SNK** (and resumes toggling between **Unattached.SNK** and **Unattached.SRC**).

4.6 Power

Power delivery over the USB Type-C connector takes advantage of the existing USB methods as defined by the **USB 2.0** and **USB 3.2** specifications, the **USB BC 1.2** specification, and the **USB Power Delivery** specification. Power for **USB4** operation requires a **USB PD** explicit contract as defined in Section 5.3 and the **USB Power Delivery** specification. Prior to entering a **USB PD** explicit contract, a **USB4** port operates as a **USB 3.2** port regarding power.

The **USB Type-C Current** mechanism allows the Source to offer more current than defined by the **USB BC 1.2** specification. A USB power source *shall not* provide more than 48 V nominal on VBUS. **USB PD** power sources

that deliver power over a USB Type-C connector **shall** follow the power rules as defined in ~~Section 10~~ of the **USB Power Delivery** specification.

All USB Type-C-based devices **shall** support **USB Type-C Current** and **may** support other USB-defined methods for power. The following order of precedence of power negotiation **shall** be followed: **USB BC 1.2** supersedes the **USB 2.0** and **USB 3.2** specifications, **USB Type-C Current** at 1.5 A and 3.0 A supersedes **USB BC 1.2**, and **USB Power Delivery** supersedes **USB Type-C Current**. Table 4-19 summarizes this order of precedence of power source usage.

Table 4-19 Precedence of Power Source Usage

Precedence	Mode of Operation		Nominal Voltage	Maximum Current
Highest ↓ Lowest	USB PD		Configurable up to 48 V	5 A
	USB Type-C Current @ 3.0 A		5 V	3.0 A
	USB Type-C Current @ 1.5 A		5 V	1.5 A
	USB BC 1.2		5 V	Up to 1.5 A ¹
	Default USB Power	USB 3.2	5 V	See USB 3.2
		USB 2.0	5 V	See USB 2.0

Note 1: **USB BC 1.2** permits a power provider to be designed to support a level of power between 0.5 A and 1.5 A. If the **USB BC 1.2** power provider does not support 1.5 A, then it is required to follow power droop requirements. A **USB BC 1.2** power consumer **may** consume up to 1.5 A provided that the voltage does not drop below 2 V, which **may** occur at any level of power above 0.5 A.

For example, once the **USB PD** mode (e.g., a power contract has been negotiated) has been entered, the device **shall** abide by that power contract ignoring any other previously made or offered by the **USB Type-C Current**, **USB BC 1.2** or **USB 2.0** and **USB 3.2** specifications. When the **USB PD** mode is exited, the device **shall** fallback, in precedence order, to the **USB Type-C Current**, **USB BC 1.2**, or **USB 2.0** and **USB 3.2** specification power levels.

All USB Type-C ports **shall** tolerate being connected to USB power source supplying default USB power, e.g., a host being connected to a legacy USB charger that always supplies VBUS.

4.6.1 Power Requirements during USB Suspend

USB Type-C implementations with **USB Type-C Current**, **USB PD** and VCONN, along with active cables, require the need to expand the traditional USB suspend definition.

4.6.1.1 Vbus Requirements during USB Suspend

The **USB 2.0** and **USB 3.2** specifications define the amount of current a Sink is allowed to consume during suspend.

USB suspend power rules **shall** apply when the **USB Type-C Current** is at the Default USB Power level or when **USB PD** is being used and the Suspend bit is set appropriately.

When **USB Type-C Current** is set at 1.5 A or 3.0 A, the Sink is allowed to continue to draw current from VBUS during USB suspend. During USB suspend, the Sink's requirement to track and meet the **USB Type-C Current** advertisement remains in force (See Section 4.5.2.3).

USB PD provides a method for the Source to communicate to the Sink whether the Sink must follow the USB power rules for suspend.

4.6.1.2 VCONN Requirements during USB Suspend

If the Source supplies VBUS power during USB suspend, it **shall** also supply VCONN and meet the requirements defined in Table 4-5.

Electronically Marked Cables **shall** meet the requirements in Table 4-6 during USB suspend.

VCONN powered accessories **shall** meet the requirements defined in Table 4-7 during USB suspend.

4.6.2 VBUS Power Provided Over a USB Type-C Cable

The minimum requirement for VBUS power supplied over the USB Type-C cable assembly matches the existing requirement for VBUS supplied over existing legacy USB cable assemblies.

USB Power Delivery in Standard Power Range (SPR) operation is intended to work over un-modified USB Type-C to USB Type-C cables, therefore any USB Type-C cable assembly that incorporates electrical components or electronics **shall** ensure that it tolerate, or be protected from, a VBUS voltage of 21 V.

USB Power Delivery in Extended Power Range (EPR) operation requires EPR-compatible USB Type-C to USB Type-C cables. Any USB Type-C cable assembly that incorporates electrical components or electronics that **may** be powered by VBUS **shall** ensure that it can functionally tolerate, or be protected from, a VBUS voltage of up to 50.9 V (48 V + 5% + 500 mV).

4.6.2.1 USB Type-C Current

Default USB voltage and current are defined by the **USB 2.0** and **USB 3.2** specifications. All **USB Type-C Current** advertisements are at the USB VBUS voltage defined by these specifications.

The **USB Type-C Current** feature provides the following extensions:

- Higher current than defined by the **USB 2.0**, the **USB 3.2**, or the BC 1.2 specifications.
- Allows the power source to manage the current it provides.

The USB Type-C connector uses CC pins for configuration including an ability for a Source to advertise to its port partner (Sink) the amount of current it **shall** supply:

- Default is the as-configured for high-power operation current value as defined by the USB Specification (500 mA for **USB 2.0** ports; 900 mA or 1,500 mA for **USB 3.2** ports in single-lane or dual-lane operation, respectively).
- 1.5 A.
- 3.0 A.

When a Source is advertising USB Type-C Default current, the Sink behavior is defined as follows:

- It connects as a **USB 2.0** or **USB 3.2** device, after which the Sink **shall** follow the appropriate USB specification.
- It enters a **USB PD** contract, after which the Sink **shall** follow the **USB PD** specification to determine the current (e.g., **Rp** will no longer be Default as it is superseded by the **USB PD** contract).
- It detects a **USB BC 1.2** charging port, after which the Sink **shall** follow the **USB BC 1.2** specification.
- It attaches as a USB Type-C Power Sinking Device (PSD), after which the Sink **may** draw up to 500 mA and **shall** meet the inrush requirement for **USB 2.0**.

A PSD **shall** fully support **USB Type-C Current** operation, **should** support **USB PD**, and **may** support **USB BC 1.2**. A PSD **may** be a Sink or a DRP operating in Sink mode. A PSD **shall not** have a USB or USB Type-C **Alternate Mode** communications function.

The relationship of **USB Type-C Current** and the equivalent **USB PD** Power (PDP) value is shown in Table 4-20.

Table 4-20 USB Type-C Current Advertisement and PDP Equivalent

USB Type-C Current		PDP Equivalent
Default	500 mA (USB 2.0)	2.5 W
	900 mA (USB 3.2 single-lane)	4.5 W
	1,500 mA (USB 3.2 dual-lane)	7.5 W
1.5 A		7.5 W
3.0 A		15 W

A Sink that takes advantage of the additional current offered (e.g., 1.5 A or 3.0 A) **shall** monitor the CC pins and **shall** adjust its current consumption within **tSinkAdj** to remain within the value advertised by the Source. While a **USB PD** contract is in place, a Sink is not required to monitor USB Type-C current advertisements and **shall not** respond to USB Type-C current advertisements.

The Source **shall** supply VBUS to the Sink within **tVBUSON**. VBUS **shall** be in the specified voltage range at the advertised current.

A Source (port supplying VBUS) **shall** protect itself from a Sink that draws current more than the port's **USB Type-C Current** advertisement.

The Source adjusts **Rp** (or current source) to advertise which of the three current levels it supports. See Table 4-27 for the termination requirements for the Source to advertise currents.

The value of **Rp** establishes a voltage (**vRd**) on CC that is used by the Sink to determine the maximum current it **may** draw.

Table 4-38 defines the CC voltage range observed by the Sink that only support default USB current.

If the Sink wants to consume more than the default USB current, it **shall** track **vRd** to determine the maximum current it **may** draw. See Table 4-39.

Figure 4-35 and **Figure 4-36** illustrate where the Sink monitors CC for **vRd** to detect if the host advertises more than the default USB current.

Figure 4-35 Sink Monitoring for Current in Pull-Up/Pull-Down CC Model

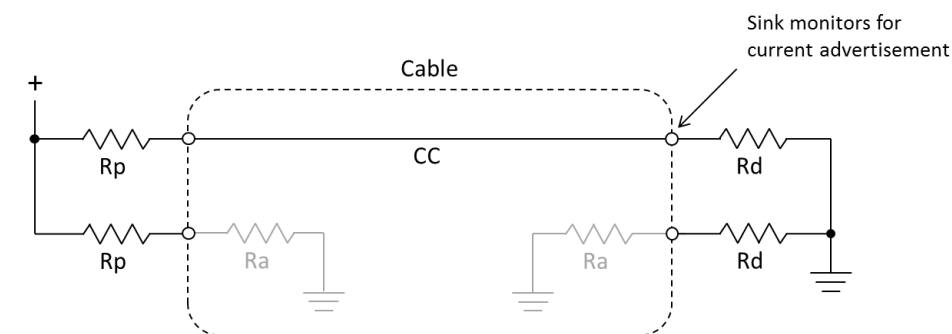


Figure 4-36 Sink Monitoring for Current in Pull-Up/Pull-Down CC Model

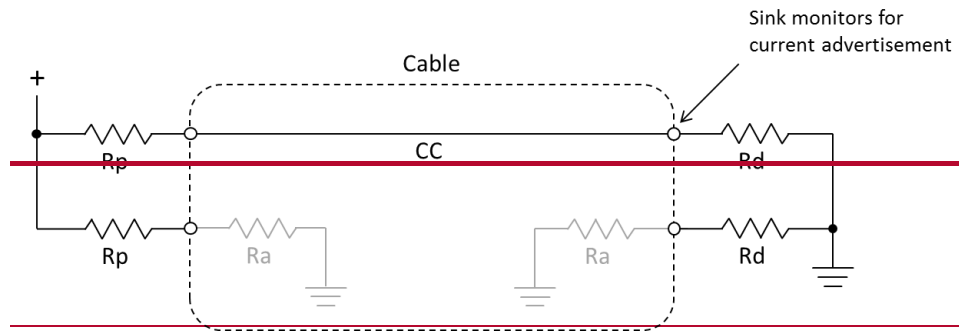
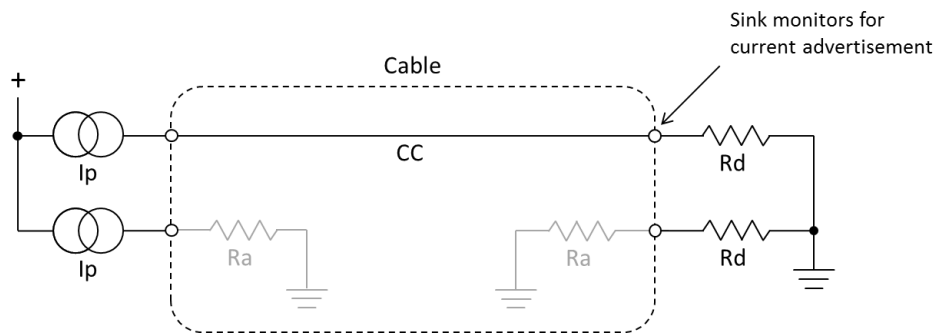


Figure -- Sink Monitoring for Current in Current Source/Pull-Down CC Model



4.6.2.2 USB Battery Charging 1.2

USB Battery Charging Specification, Revision 1.2 defines a method that uses the **USB 2.0** D+ and D- pins to advertise VBUS can supply up to 1.5 A. Support for **USB BC 1.2** charging is *optional*.

A USB Type-C port that implements **USB BC 1.2** that is capable of supplying at least 1.5 A *shall* advertise **USB Type-C Current** at the 1.5 A level within *tVBUSON* of entering the **Attached.SRC** state, otherwise the port *shall* advertise **USB Type-C Current** at the Default USB Power level. A USB Type-C port that implements **USB BC 1.2** that also supports **USB Type-C Current** at 3.0 A *may* advertise **USB Type-C Current** at 3.0 A.

If a Sink that supports **USB BC 1.2** detection, detects *Rp* at the Default USB Power level and does not discover a **USB BC 1.2**-compliant Source, then it *shall* limit its maximum current consumption to the standard USB levels based on Table 4-19. This will ensure maximum current limits are not exceeded when connected to a Source which does not support **USB BC 1.2**.

A Sink that supports **USB BC 1.2** detection and has a maximum current draw greater than Default **USB Type-C Current** *shall* monitor *vRd* on the CC pins to detect the Source's *Rp* and *shall* implement **Sink Power Sub-State** transitions (Figure 4-19). If a Sink that supports **USB BC 1.2** detection and has a maximum current draw greater than Default **USB Type-C Current** detects *Rp* at **USB Type-C Current** of 1.5 A or 3.0 A levels but does not detect a **USB BC 1.2** source, it *shall* limit its maximum current consumption to the appropriate **USB Type-C Current** level advertised, and *shall* adjust its current consumption to remain within the value advertised by the Source on Sub-State transitions. For Sub-State transitions starting from a higher power level and ending at a lower power level, the Sink *shall* reduce power consumption within *tSinkAdj*. See Sections 4.5.2.3.2.2 and 4.5.2.3.3.2.

While in a **USB Power Delivery** Mode, a device acting as a Sink *shall not* initiate a **USB BC 1.2** detection until the port pair is detached or there is an Error Recovery or Hard Reset.

4.6.2.3 Proprietary Power Source

This section has been deprecated. Devices with USB Type-C connectors **shall** only employ signaling methods defined in USB specifications to negotiate power.

4.6.2.4 USB Power Delivery

USB Power Delivery is a feature on the USB Type-C connector. When **USB PD** is implemented, **USB PD** Bi-phase Mark Coded (BMC) carried on the CC wire **shall** be used for **USB PD** communications between USB Type-C ports.

At attach, VBUS **shall** be operationally stable prior to initiating **USB PD** communications.

Figure 4-37 illustrates how the **USB PD** BMC signaling is carried over the USB Type-C cable's CC wire.

Figure 4-37 USB PD over CC Pins

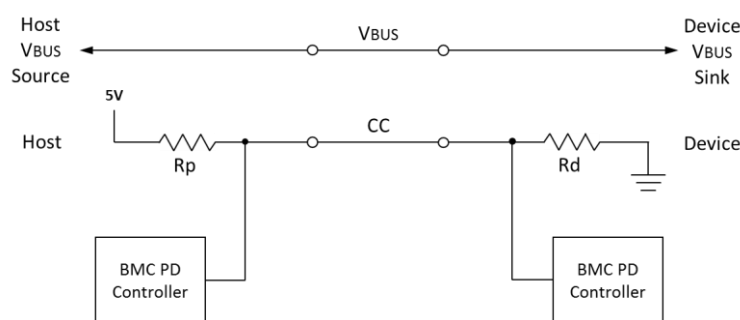
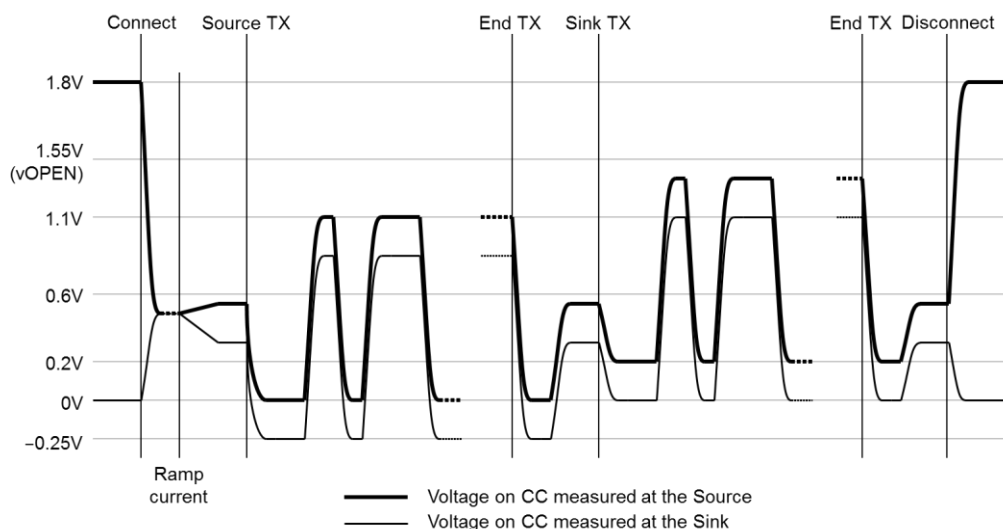


Figure 4-38 illustrates **USB PD** BMC signaling as seen on CC from both the perspective of the Source and Sink. The breaks in the signaling are intended to represent the passage of time.

Figure 4-38 USB PD BMC Signaling over CC



When not in an Explicit Contract, **USB PD** Sources that are, based on their PDP, (i.e., Guaranteed PDP as defined by USB PD), capable of supplying:

- 5 V at 3 A 15 W or greater **shall** advertise **USB Type-C Current** at the 3.0 A level,
- 7.5 V at 1.5 A W or greater but less than 3 A 15 W **shall** advertise **USB Type-C Current** at the 1.5 A level, or

- ~~5 V at 1.5 A~~ ~~7.5 W~~ **shall** advertise **USB Type-C Current** at the Default USB Power level within **tVBUSON** of entering the **Attached.SRC** state. For ~~Multi-Port~~ Shared Capacity ~~Chargers~~ **Ports**, a **USB PD** Source capable of supplying ~~5 V at 3 A~~ **15 W** or greater **may** initially offer **USB Type-C Current** at the 1.5 A level and subsequently increase the offer after attach (see Section 4.8.6.2). During USB Suspend a **USB PD** Source **may** set its **Rp** value to default to indicate that the Sink **shall** only draw USB suspend current as defined in Section 4.6.1.1.

While a **USB PD** Explicit Contract is in place, a Source compliant with **USB PD** Revision 2 **shall** advertise a **USB Type-C Current** of either 1.5 A or 3.0 A. The **USB PD** Revision 2 Source upon entry into an Explicit Contract **shall** advertise an **Rp** value of 1.5 A or 3.0 A after it receives the GoodCRC in response to the first PS_RDY Message.

While a **USB PD** Explicit Contract is in place, a Source compliant with **USB PD** Revision 3 **shall** set the **Rp** value according to the collision avoidance scheme defined in Section 5.7 of the **USB PD** Revision 3 specification. The **USB PD** Revision 3 Source upon entry into an Explicit Contract **shall** advertise an **Rp** value consistent with the **USB PD** Revision 3 collision avoidance scheme.

Refer to ~~Section 1.6 of~~ the **USB Power Delivery** specification for a definition of an Explicit Contract.

4.6.2.5 Charge-Through VCONN-Powered USB Device (CTVPD) Current Limitations

Since Charge-Through **VCONN-Powered USB Devices** implement charging by passively connecting the Source's CC and VBUS to the Host, the **VCONN-Powered USB Device** is effectively increasing the impedance on VBUS, GND, and CC between the Power Source and the Host, resulting in impedances that can exceed the maximum allowed for cables. To avoid communication issues and false disconnects from the increased GND and VBUS drops, the following **shall** occur:

1. The Charge-Through **VCONN-Powered USB Device** **shall** report its worst-case GND and VBUS impedance (including the extra mated connector pair and FETs) in its **USB PD Discover Identity** Command response on SOP'.
2. The Host that supports Charge-Through **VCONN-Powered USB Device** **shall** use this information, along with inferred information about the cable, to limit its maximum current in the case where the Power Source advertises a current greater than what the Charge-Through **VCONN-Powered USB Device** would allow.

The Host has no way to query the cable, as its VCONN source is consumed by the **VCONN-Powered USB Device**. Instead, the Host **may** assume the cable is 5 A for the purposes of calculating the Charge-Through current limit only if it receives a **USB PD** Source Capability PDO of greater than 3 A (even if the Host ultimately does not **Request** that PDO, or if the host requests a current of 3 A or less).

The Host **shall** further limit its maximum current beyond that advertised by the Power Source, based on the reported GND impedance and the inferred cable capability. GND impedance is reported in the **VPD Discover Identity** Command Response in 1-milliohm steps and is used in the following formulas:

- GND-limited current with a 3A cable inferred = $0.25 \text{ V} / (0.25 \text{ V} / 3 \text{ A} + \text{VPD_GND_DCR})$, and
- GND-limited current with a 5A cable inferred = $0.25 \text{ V} / (0.25 \text{ V} / 5 \text{ A} + \text{VPD_GND_DCR})$.

Some examples are in Table 4-21.

Table 4-21 Sink Maximum Current Limit When Attached to CTVPD

Reported GND Impedance	3A Cable Inferred ¹	5A Cable Inferred ²
0.010 Ω	2.679 A	4.167 A
0.015 Ω	2.542 A	3.846 A
0.020 Ω	2.419 A	3.571 A

0.025 Ω	2.308 A	3.333 A
0.030 Ω	2.206 A	3.125 A
0.035 Ω	2.113 A	2.941 A
0.040 Ω	2.027 A	2.778 A
Notes:		
1. As calculated by $0.25 \text{ V} / (0.25 \text{ V} / 3 \text{ A} + \text{VPD_GND_DCR})$. 2. As calculated by $0.25 \text{ V} / (0.25 \text{ V} / 5 \text{ A} + \text{VPD_GND_DCR})$.		

In addition, the increased VBUS impedance could result in a greater than 1 V VBUS drop as measured at the input to the Host. Based on the VBUS impedance reported in the **VPD Discover Identity** Command Response in 2-milliohm steps and the inferred cable capability, the Host **shall** either lower its VBUS detach threshold or further limit its maximum current based on the following formulas:

- VBUS and GND-limited current with a 3A cable inferred = $0.75 \text{ V} / (0.75 \text{ V} / 3 \text{ A} + \text{VPD_VBUS_DCR} + \text{VPD_GND_DCR})$, and
- VBUS and GND-limited current with a 5A cable inferred = $0.75 \text{ V} / (0.75 \text{ V} / 5 \text{ A} + \text{VPD_VBUS_DCR} + \text{VPD_GND_DCR})$.

Table 4-22 Example Charge-Through VPD Sink Maximum Currents based on VBUS Impedance and GND Impedance

Reported VBUS Impedance ⁴	Reported GND Impedance ⁴	3A Cable Inferred ¹	5A Cable Inferred ²
0.020 Ω	0.010 Ω	2.679 A	4.167 A
0.030 Ω	0.015 Ω	2.542 A	3.846 A
0.040 Ω	0.020 Ω	2.419 A	3.571 A
0.050 Ω	0.025 Ω	2.308 A	3.333 A
0.060 Ω	0.030 Ω	2.206 A	3.125 A
0.070 Ω	0.035 Ω	2.113 A	2.941 A
0.080 Ω	0.040 Ω	2.027 A	2.778 A
Notes:			
1. As calculated by $0.75 \text{ V} / (0.75 \text{ V} / 3 \text{ A} + \text{VPD_VBUS_DCR} + \text{VPD_GND_DCR})$. 2. As calculated by $0.75 \text{ V} / (0.75 \text{ V} / 5 \text{ A} + \text{VPD_VBUS_DCR} + \text{VPD_GND_DCR})$. 3. The table does not show all allowable combinations, only a subset provided for illustration. 4. The ratio of the reported VBUS impedance to the reported GND impedance is 2:1.			

4.6.2.6 USB Type-C Sink Requirements for High Voltage Operation

This section sets electrical requirements for USB Type-C Sinks that support high-voltage operation. See Section 3.11 for EPR requirements for USB Type-C cables that support EPR.

The Sink **shall** keep the voltage on its VBUS contact to within 12 volts of the voltage on the Source VBUS contact at the time of the cable plug withdrawal for a minimum of 250 μs from the time the VBUS contacts separate (see Figure 3-1). Refer to Appendix H for more information related to high-voltage contact arcing and mitigation guidelines.

4.7 USB Hubs

USB 2.0, **USB 3.2**, and **USB4** hubs are defined by the **USB 2.0**, **USB 3.2**, and **USB4** specifications, respectively. USB hubs implemented with one or more USB Type-C connectors **shall** comply with these specifications as relevant to a USB Type-C implementation. All the downstream facing USB Type-C ports on a USB hub **should** support the same functionality or **shall** be clearly marked as to the functionality supported.

USB hubs **shall** have an upstream facing port (to connect to a host or hub higher in the USB tree) that **may** be a Sourcing Device (See Section 4.8.4). The hub **shall** clearly identify to the user its upstream facing port. This **may** be accomplished by physical isolation, labeling or a combination of both.

USB hub's downstream facing ports **shall not** have Dual-Role-Data (DRD) capabilities. However, these ports **may** have Dual-Role-Power (DRP) capabilities.

CC pins are used for port-to-port connections and **shall** be supported on all USB Type-C connections on the hub.

For **USB 2.0** and **USB 3.2** hubs, downstream-facing ports **shall not** implement or pass-through Alternate or Accessory Modes and SBU pins **shall not** be connected (**zSBU Termination**) on any USB hub port. For **USB4** hubs, see Section 5.2.3 regarding support for **Alternate Modes**.

The USB hub's DFPs **shall** support power source requirements for a Source. See Section 4.8.1.

Additional requirements for **USB4** hubs are in Chapter 5.

4.8 Power Sourcing and Charging

This section defines requirements and recommendations related to using USB Type-C ports for delivering power. Any USB Type-C port that offers more than Default Current and/or supports **USB Power Delivery** **shall** meet the requirements as if it is a charger.

The following lists the most applicable subsections by USB Type-C ports on:

- Host systems: 4.8.1 and 4.8.5. Note: 4.8.6 is not intended for host systems.
- Devices that can supply power: 4.8.4.
- Hubs:
 - Traditional hubs – Refer to **USB 2.0/USB 3.2** base specifications and 4.8.1 as applicable if **USB BC 1.2** is supported.
 - Hubs that can supply power beyond the base specs – 4.8.1, 4.8.4, 4.8.5 and 4.8.6.
- Dedicated chargers:
 - Single-port chargers – 4.8.1.
 - Multi-port chargers – 4.8.1 and 4.8.6.

4.8.1 DFP as a Power Source

Sources (e.g., battery chargers, hub downstream ports and hosts) **may** all be used for battery charging. When a charger is implemented with a USB Type-C receptacle or a USB Type-C captive cable, it **shall** follow all the applicable requirements.

- A Source **shall** expose its power capabilities using the **USB Type-C Current** method and it **may** additionally support other USB-standard methods (**USB BC 1.2** or **USB PD**).
- A Source advertising its current capability using **USB BC 1.2** **shall** meet the requirements in Section 4.6.2.2 regarding **USB Type-C Current** advertisement.

- A Source that has negotiated a **USB PD** contract **shall** meet the requirements in Section 4.6.2.4 regarding **USB Type-C Current** advertisement.
- If a Source is capable of supplying a voltage greater than default V_{BUS}, it **shall** fully conform to the **USB PD** specification and **shall** negotiate its power contracts using only **USB PD**.
- If a Source is capable of reversing source and sink power roles, it **shall** fully conform to the **USB PD** specification and **shall** negotiate its power contracts using only **USB PD**.
- If a Source is capable of supplying a current greater than 3.0 A, it **shall** use the **USB PD Discover Identity** to determine the current carrying capacity of the cable.

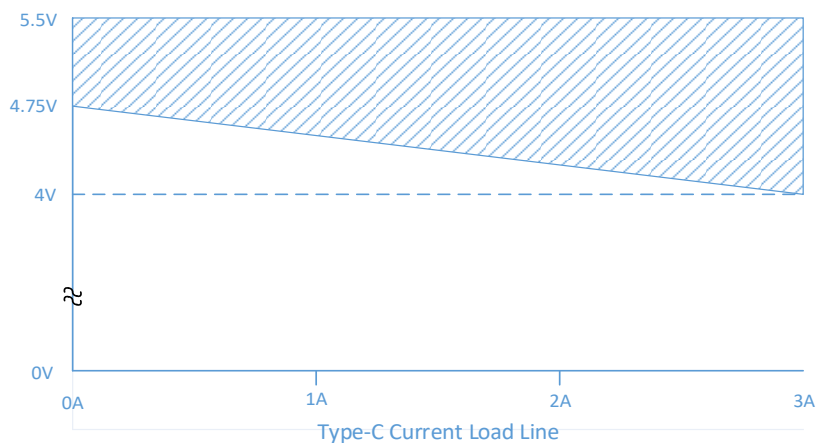
4.8.1.1 USB-based Chargers with USB Type-C Receptacles

- A USB-based charger with a USB Type-C receptacle (Source) **shall** only apply power to V_{BUS} when it detects a Sink is attached and **shall** remove power from V_{BUS} when it detects the Sink is detached (above the CC disconnect threshold).
- A USB-based charger with a USB Type-C receptacle **shall not** advertise current exceeding 3.0 A except when it uses the **USB PD Discover Identity** mechanism to determine the cable's actual current carrying capability and then it **shall** limit the advertised current accordingly.
- A USB-based charger with a USB Type-C receptacle (Source) which is not capable of data communication **shall** advertise **USB Type-C Current** of at least 1.5 A within **t_{VBUSON}** of entering the **Attached.SRC** state and **shall** short D+ and D- together with a resistance less than 200 ohms. This will ensure backwards compatibility with legacy sinks which **may** use **USB BC 1.2** for charger detection.

4.8.1.2 USB-based Chargers with USB Type-C Captive Cable

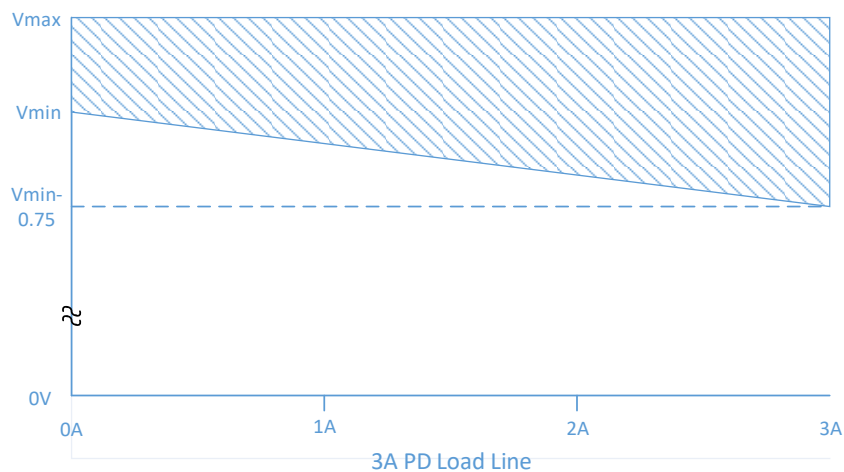
- A USB-based charger with a USB Type-C captive cable **shall** only apply power to V_{BUS} when it detects a Sink is attached and **shall** remove power from V_{BUS} when it detects the Sink is detached (above the CC disconnect threshold).
- A USB-based charger with a USB Type-C captive cable **shall** limit its current advertisement so as not to exceed the current capability of the cable (up to 5 A).
- A USB-based charger with a USB Type-C captive cable which is not capable of data communication **shall** advertise **USB Type-C Current** of at least 1.5 A. It **shall** short D+ and D- together with a resistance less than 200 ohms. This will ensure backwards compatibility with legacy sinks which **may** use **USB BC 1.2** for charging detection.
- The voltage as measured at the plug of a USB-based charger with a USB Type-C captive cable **may** be up to $0.75 \times I / 3 \text{ V}$ ($0 < I \leq 3 \text{ A}$), or $0.75 \times I / 5 \text{ V}$ ($0 < I \leq 5 \text{ A}$) lower than the standard tolerance range for the chosen voltage, where I is the actual current being drawn.
 - A USB-based charger that advertises **USB Type-C Current** **shall** output a voltage in the range of 4.75 V – 5.5 V when no current is being drawn and between 4.0 V – 5.5 V at 3 A. The output voltage as a function of load up to the advertised **USB Type-C Current** (default, 1.5 A and 3 A) **shall** remain within the cross-hatched area shown in Figure 4-39.

Figure 4-39 USB Type-C Cable's Output as a Function of Load for Non-PD-based USB Type-C Charging



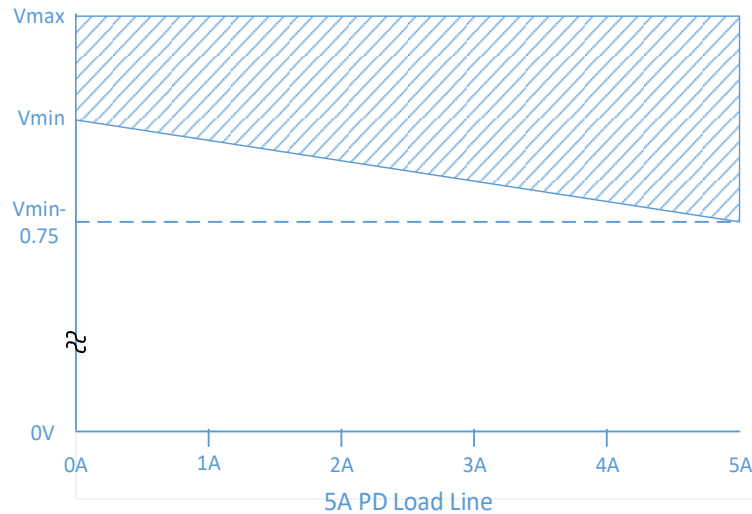
- A **USB PD**-based charger that has negotiated a voltage V at $\leq 3\text{ A}$ **shall** output a voltage in the range of V_{max} ($V + 5\%$) and V_{min} ($V - 5\%$) when no current is being drawn and V_{max} and $V_{\text{min}} - 0.75\text{ V}$ at 3 A . Under all loads, the output voltage **shall** remain within the cross-hatched area shown in Figure 4-40.

Figure 4-40 0 – 3 A USB PD-based Charger USB Type-C Cable's Output as a Function of Load



- A **USB PD**-based charger that has negotiated a voltage V at between 3 A and 5 A **shall** output a voltage in the range of V_{max} ($V + 5\%$) and V_{min} ($V - 5\%$) when no current is being drawn and V_{max} and $V_{\text{min}} - 0.75\text{ V}$ at 5 A . Under all loads, the output voltage **shall** remain within the cross hatched area shown in Figure 4-41.

Figure 4-41 3 – 5 A USB PD-based Charger USB Type-C Cable's Output as a Function of Load



- Note: The maximum allowable cable IR drop for ground is 250 mV (see Section 4.4.1). This is to ensure the signal integrity of the CC wire when used for connection detection and **USB PD** BMC signaling.

4.8.2 Non-USB Charging Methods

A product (Source and/or Sink) with a USB Type-C connector **shall** only employ signaling methods defined in USB specifications to negotiate power over its USB Type-C connector(s).

4.8.3 Sinking Host

A Sinking Host is a special sub-class of a DRP that is capable of consuming power but is not capable of acting as a USB device. For example, a hub's DFP or a notebook's DFP that operates as a host but not as a device.

The Sinking Host **shall** follow the rules for a DRP (See Section 4.5.1.4 and Figure 4-15). The Sinking Host **shall** support **USB PD** and **shall** support the **DR_Swap** command. If the Sinking Host ends up in the UFP data role after entering **Attached.SNK** and establishing an Explicit **USB PD** contract, it **shall** accept or send a **DR_Swap** command in order to get itself into the DFP data role.

4.8.4 Sourcing Device

A Sourcing Device is a special sub-class of a DRP that is capable of supplying power but is not capable of acting as a USB host. For example, a hub's UFP or a monitor's UFP that operates as a device but not as a host.

The Sourcing Device **shall** follow the rules for a DRP (See Section 4.5.1.4 and Figure 4-15). It **shall** also follow the requirements for the Source as Power Source (See Section 4.8.1). The Sourcing Device **shall** support **USB PD** and **shall** support the **DR_Swap** command. If the Sourcing Device ends up in the DFP data role after entering **Attached.SRC** and establishing an Explicit **USB PD** contract, it **shall** accept or send a **DR_Swap** command in order to get itself into the UFP data role.

4.8.5 Charging a System with a Dead Battery

A system that supports being charged by USB whose battery is dead **shall** apply **Rd** to both CC1 and CC2 and follow all Sink rules. When it is connected to a Source, DRP or Sourcing Device, the system will receive the default VBUS. It **may** use any allowed method to increase the amount of power it can use to charge its battery.

The circuitry to present **Rd** in a dead battery case only needs to guarantee the voltage on CC is pulled within the same range as the voltage clamp implementation of **Rd** in order for a Source to recognize the Sink and provide VBUS. For example, a 20% resistor of value **Rd** in series with a FET with $V_{GTH(max)} < V_{CLAMP(max)}$ with the gate weakly pulled to CC would guarantee detection and be removable upon power up.

When the system with a dead battery has sufficient charge, it **may** use the **USB PD DR_Swap** message to become the DFP.

4.8.6 USB Type-C Multi-Port Chargers

A USB Type-C multi-port charger is a product that exposes multiple USB Type-C Source ports for the purpose of charging multiple connected devices. A compliant USB Type-C charger **may** offer on each of its ports a mix of power options as defined in Section 4.6.

Each port on multi-port chargers will fall into one of the following two categories.

1. **Assured Capacity Ports:** An Assured Capacity port has a fixed allocated amount of power and is either a Guaranteed Capability port or Managed Capability port as defined in the **USB PD** specification. Assured Capacity ports are independent ports that are not grouped.
2. **Shared Capacity Ports:** A Shared Capacity port does not have a fixed allocated amount of power and is a Managed Capability port as defined in the **USB PD** specification. Shared Capacity ports only exist as a group of two or more ports where the sum of the maximum capabilities of all of the exposed ports, as indicated to the user, is greater than the total power delivery capacity of the group of ports.

A multi-port charger **may** offer in a single product separate visually identifiable Assured Capacity ports and groups of Shared Capacity ports.

The remainder of this section defines the requirements and provides guidelines for the operation and behavior of a USB Type-C multi-port charger.

4.8.6.1 General Requirements

Individual source ports **shall** always comply with power negotiation and rules set forth by the USB Type-C and **USB Power Delivery** specifications, adjusted as needed when available resources change as other ports take more or less power.

The minimum capability of all individual USB Type-C ports of a USB Type-C multi-port charger **shall** be 5 V @ 1.5 A independent of how many of the other ports are in use.

For Shared Capacity Ports, the following are defined:

- **Total Shared Capacity** is the total power available for a group of Shared Capacity ports. This is the overall power available when none of the ports in the group are connected to Sinks
- **Remaining Power Available** = Total Shared Capacity minus the sum of the power allocated to the attached ports. When doing this calculation, for ports in a **USB PD** power contract, the maximum power offered in the **USB PD** source capabilities is used, not the power requested in the RDO.
- **Shared Port Power Available** is either:
 - a) the port's PDP when it is less than or equal to the Remaining Power Available,
 - b) the Remaining Power Available when it is greater than 7.5 W and less than the port's PDP, or
 - c) 7.5 W.

When a Shared Capacity port in a group reclaims power from one or more already connected ports to meet the 7.5 W minimum requirement for a newly attached device, the charger **shall** use standard **USB Type-C Current** and/or **USB PD** renegotiation methods and **shall not** use Hard Resets, **Error Recovery**, or the **Disabled** state.

When a USB Type-C charger includes charging ports that are based on USB Standard-A receptacles, the following requirements **shall** be met.

- The USB Standard-A ports **shall** be implemented as an independent group, i.e., USB Standard-A ports **shall not** be included in a group of USB Type-C ports behaving as a Shared Capacity group. Any load change on a USB Type-A port **shall not** result in a voltage change on any of the USB Type-C ports and vice-versa.
- The minimum capability of each USB Standard-A port **shall** be 5 V @ 500 mA independent of how many of the other ports are in use.

Implementation Note: When designing a Shared Capacity group of ports, user experience should be an important factor in deciding the Total Shared Capacity to support. There are tradeoffs to be made regarding port PDP, the number of ports, and the total power available to be shared that will notably impact what a user can expect when charging multiple devices.

Table 4-23 gives some comparative examples of a 4-port Shared Capacity group to illustrate how the PDP and Total Shared Capacity impacts user experience as more devices are connected to the charger. Two simplistic power sharing policies are illustrated: a Highest-to-Lowest policy that emphasizes supplying the highest power to ports based on a first-come/first-serve priority, and a Balanced policy that always equally balances power across all connected ports. These example policies assume that the charger has no knowledge of the device's specific needs other than what the device requests when selecting from the charger's advertised capabilities, and that the order of connecting devices is from highest to lowest power consumer.

Table 4-23 Examples of 4-port Shared Capacity Group and Power Sharing Policies

	Port PDP Rating	Total Shared Capacity	Highest-to-Lowest				Balanced				Notes
			1	2	3	4	1	2	3	4	
1	7.5 W	30 W	7.5	-	-	-	7.5	-	-	-	The minimal implementation is the functional equivalent to a group of Assured Ports where the Total Shared Capacity = PDP · <i>n</i> with <i>n</i> being the number of ports in the group.
			7.5	7.5	-	-	7.5	7.5	-	-	
			7.5	7.5	7.5	-	7.5	7.5	7.5	-	
			7.5	7.5	7.5	7.5	7.5	7.5	7.5	7.5	
2	30 W	30 W	30	-	-	-	30	-	-	-	These implementation examples illustrate that as Total Shared Capacity increases while port count and PDP remain constant, the user experience when charging multiple devices improves. Also, as the ratio of Total Shared Capacity to Port PDP Rating increases, the sharing policy of the charger relies less on reclaiming power from already connected ports.
			22.5	7.5	-	-	15	15	-	-	
			15	7.5	7.5	-	10	10	10	-	
			7.5	7.5	7.5	7.5	7.5	7.5	7.5	7.5	
3	30 W	60 W	30	-	-	-	30	-	-	-	
			30	30	-	-	30	30	-	-	
			30	22.5	7.5	-	20	20	20	-	
			30	15	7.5	7.5	15	15	15	15	
4	30 W	100 W	30	-	-	-	30	-	-	-	
			30	30	-	-	30	30	-	-	
			30	30	30	-	30	30	30	-	

Port PDP Rating	Total Shared Capacity	Highest-to-Lowest				Balanced				Notes
		1	2	3	4	1	2	3	4	
		30	30	30	10	25	25	25	25	

More sophisticated policies could be implemented for an even better user experience, these likely considering capabilities information retrieved from the device. For example, when multiple devices are attached, the needs of each device can be determined, and power allocated to give preference to devices that require higher power. As each new device is attached, power can be rebalanced after considering the additional needs of the new device.

4.8.6.2 Multi-Port Charger Behaviors

Each Assured Capacity port **shall**, by design, behave independently and be unaffected by the status and loading of the other ports. An exception to this behavior is allowed if the charger has to take any action necessary to meet an overall product operational safety requirement due to unexpected behavior on any port.

For each group of Shared Capacity ports, the following behavioral rules **shall** apply:

- Each of the exposed ports **shall** have the same power capabilities and can draw from the shared power equally.
 - When the power available to the port is not limited, a port's Port Present PDP will be to equal its Port Maximum PDP, and the port shall have the ability to supply power up to the port's Port Maximum PDP.
 - When the power available to the port is limited, a port's Port Present PDP will be less than its Port Maximum PDP, and the port shall have the ability to supply power up to the port's Port Present PDP.
 - Ports **may** offer less than its Port Present PDP but **shall** increase the offer up to its Port Present PDP when the Sink sets the Capabilities Mismatch bit in its response. This **may** be done in multiple steps, but all ports in the Shared Capacity group **shall** reach the maximum power or satisfy the Sink's needs within three seconds.
 - Whenever a power contract is made or changed on any port, the Remaining Power Available and the Shared Port Power Available **shall** be re-computed, and the Source **shall** send updated Source Capability messages as needed.
 - Ports when not in a **USB PD** contract **shall** follow the rules for a shared **USB Type-C Current** source.
- All exposed **USB Type-C Current** ports **shall** have the ability to offer the same power capabilities.
 - All ports **shall** initially offer at least 1.5 A, e.g., **shall not** offer Default.
 - The total of offers across all the ports **shall** never exceed the capacity of the shared supply.
 - Ports that initially offer 1.5 A **shall** increase to 3 A after attach if they have sufficient available shared capacity within one second.

As Source ports are connected and begin providing power, the remaining Source ports will each have the same power capabilities. The maximum capability **may** be less than the previously connected ports due to less unused capacity of the total power delivery capacity of the charger. For example, if the total power delivery capacity of a USB Type-C two-port charger is 60 W with a port PDP of 35 W and the first connected Source port has established a 35 W power contract with its connected Sink, then the second Source port will only be able to offer a PDP of 25 W.

Each port **should** start by offering the minimum capability for the port and increase the offering to the Sink upon a connection. For example, if the maximum capability of a USB Type-C only Source port is 3 A, then all

the exposed Source ports will be able to offer 3 A. Each port **should** start by offering less than the max (such as 1.5 A) and then increase the offering to 3 A after an attach. This would happen for each port as it is connected until the unused shared capacity is exhausted, at which point no other ports would increase to 3 A offering. A sink, in this example, would see a starting advertisement of **USB Type-C Current** @ 1.5 A at attach and would then see the **USB Type-C Current** advertisement increase to 3 A. As another example, if the maximum capability of a USB Type-C Source port is to offer **USB PD** with a PDP of 35 W, then all the exposed Source ports would also support **USB PD** 35 W. Each port would start by offering something less on initial connection, like 15 W, and then increase the offering with new Source Capabilities when it determines the Sink would like more power. If the Sink is not offered the power it requires, it will send a request with the Capability Mismatch bit set to indicate to the source it wants more power. This will happen for each port as it is connected until the unused shared capacity is exhausted, at which point no other ports would increase the power offering.

When establishing the remaining available capacity, a charger that supports policy-based power rebalancing **may** include the power that can be reclaimed from ports already in use:

1. by adjusting advertised source capabilities equivalent with a reduced PDP to one or more ports that are already in use; or
2. by issuing a **USB PD** GotoMin command to one or more ports already in use.

Policy-based power rebalancing **should** consider providing good user experience and preserving nominal USB functionality on impacted devices. Fixed rebalancing algorithms that do not factor in overall USB system policy **may not** be appropriate for power rebalancing implementations.

4.8.6.3 Multi-Port Charger Port Labeling

Multi-port chargers **shall** have OEM-designed port labeling consistent with the following rules.

- For Assured Capacity ports, each exposed Source port **shall** be labeled to indicate the PDP of the port. In this case, the user will be able to expect that each of the labeled ports will be able to meet power contracts consistent with the labeling independent of how many of the Source ports are in use.
- For Shared Capacity port groups, each Source port **shall** be labeled to indicate the same PDP. Additionally, the charger **shall** have a label that, with a minimum of equal visual prominence, indicates the Total Shared Capacity (see Section 4.8.6.1) shared across all the ports identified as a group.

For a multi-port charger that offers a mix of Assured Capacity and Shared Capacity ports, each port and port group **shall** be clearly identified, and each **shall** be individually labeled consistent with an Assured Capacity port or a Shared Capacity port group as defined above.

Refer to the USB Implementers Forum (**USB-IF**) for USB Type-C Chargers certification along with further labeling guidelines.

4.8.6.4 Multi-Port Charger that includes USB Data Hub Functionality

Multi-port chargers that also incorporate USB data hub capabilities **shall** meet the same requirements as standalone chargers. These charging-capable hubs **shall** be self-powered and **shall** fully operate as a charger independent of the state of the USB data bus connections.

For hub-based multi-port chargers that offer power to the upstream-facing port (to the host), this port **may** either behave as an Assured Capacity port (e.g., be a dedicated charging port) or as a Shared Capacity port (e.g., sharing capacity with downstream-facing ports). In either case, it **should** be clearly labeled consistent with its designed behavior, including identifying it as part of a group if it is sharing capacity with other ports.

When the upstream-facing port is sharing capacity with the downstream-facing ports, the PDP of the upstream-facing port can differ from the downstream-facing ports.

4.9 Electronically Marked Cables

All USB Full-Featured Type-C cables **shall** be electronically marked. **USB 2.0** Type-C cables **may** be electronically marked. An eMarker is an element in an Electronically Marked Cable that returns information about the cable in response to a **USB PD Discover Identity** command.

Electronically marked cables **shall** support **USB Power Delivery** Structured VDM **Discover Identity** command directed to SOP' (the eMarker). This provides a method to determine the characteristics of the cable, e.g., its current carrying capability, its performance, vendor identification, etc. This **may** be referred to as the USB Type-C Cable ID function.

Prior to an explicit **USB PD** contract, a Sourcing Device is allowed to use SOP' to discover the cable's identity. After an explicit **USB PD** contract has been negotiated, only the VCONN Source **shall** communicate with SOP' and SOP'' (see also Section 6.1.1).

Passive cables that include an eMarker **shall** follow the Cable State Machine defined in Section 4.5.2.4 and Figure 4-20.

Any cable (or direct connect device) that uses VCONN to power circuitry is required to implement cable **Ra** management as defined in Section 4.5.2.5.

Once VCONN is available, all electronically marked cables **shall** use it as the only power source. If VCONN is applied after VBUS then until VCONN is available, the cable **may** remain unpowered or **may** draw power from VBUS. Within **tVCONNSwitch**, the cable **shall** switch from VBUS to VCONN. Cables that include an eMarker **shall** meet the maximum power defined in Table 4-6. The only exception is an Optically Isolated Active Cable (see Section 6.3), which can draw from both VCONN and VBUS.

Refer to Table 4-5 for the requirements of a Source to supply VCONN. When VCONN is not present, a powered cable **shall not** interfere with normal CC operation including Sink detection, current advertisement, and **USB PD** operation.

Figure 4-42 illustrates a typical electronically marked cable. The isolation elements (Iso) **shall** prevent VCONN from traversing end-to-end through the cable. **Ra** is required in the cable to allow the Source to determine that VCONN is needed.

Figure 4-42 Electronically Marked Cable with VCONN connected through the cable

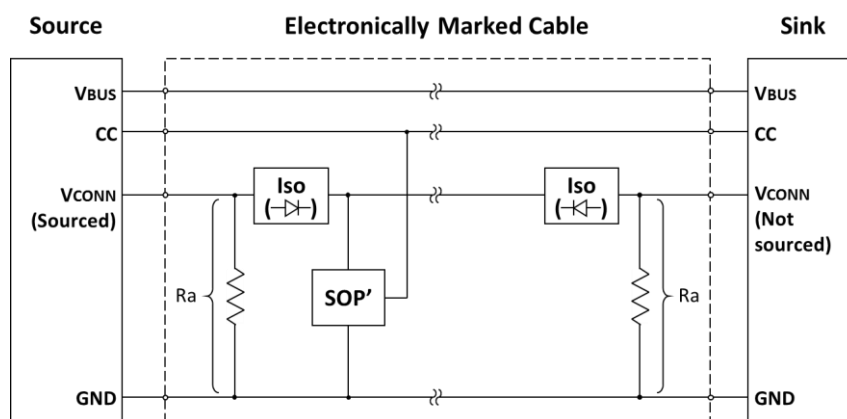
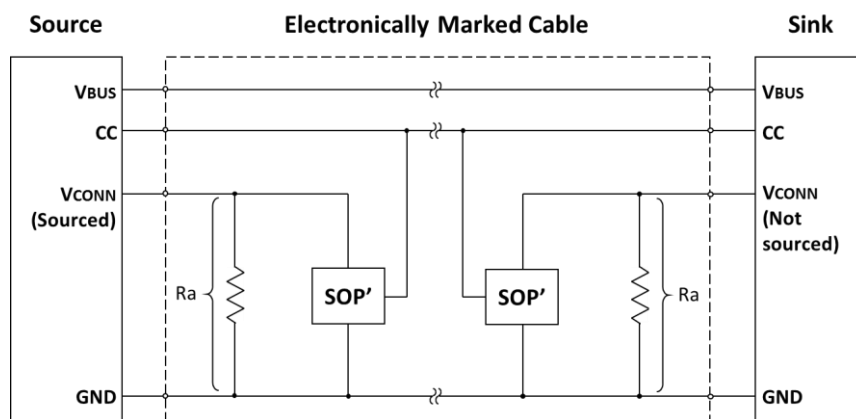


Figure 4-43 illustrates an electronically marked cable where the VCONN wire does not extend through the cable, therefore an SOP' (eMarker) element is required at each end of the cable. In this case, no isolation elements are needed.

Figure 4-43 Electronically Marked Cable with SOP' at both ends



For cables that only respond to SOP', the location of the responder is not relevant.

Figure 4-44 illustrates an electronically marked direct connect device with a captive cable where the SOP' is powered by VCONN.

Figure 4-44 Electronically Marked Direct Connect Device with a Captive Cable SOP' Powered by VCONN

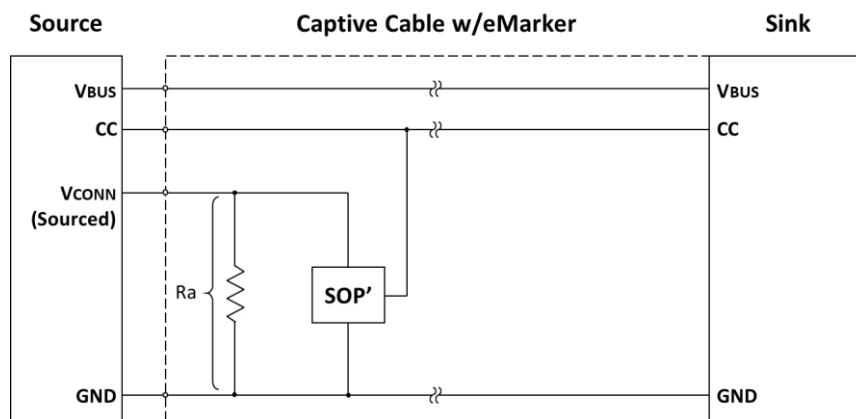


Figure 4-45 illustrates an electronically marked direct connect device with a captive cable where the SOP' is not powered by VCONN. While the SOP' doesn't rely on VCONN for power, Ra shall still be implemented but is not required to be weakened or adhere to the Ra Management State Machine defined in Section 4.5.2.5. Note: Even if the SOP' is integrated into the SOP, the timing requirements in Section 4.5.2.4 still apply.

Figure 4-45. Electronically Marked Direct Connect Device with a Captive Cable SOP' Powered by VBUS or Local Power

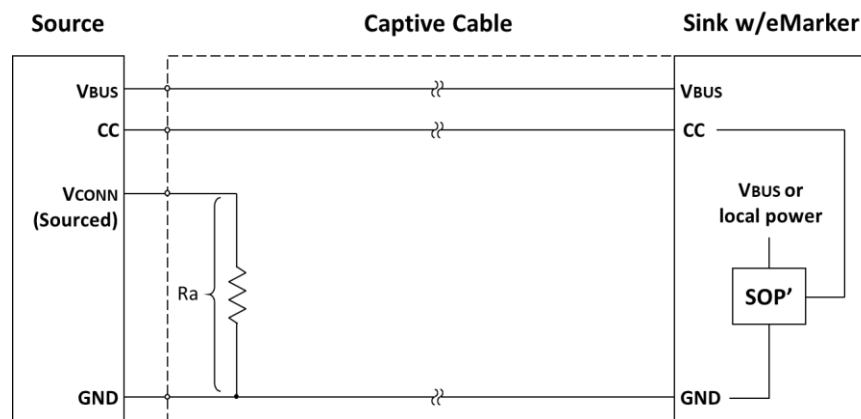


Figure 4-46 illustrates an electronically marked direct connect device with no cable (e.g., thumb drive) where the SOP' is powered by VCONN.

Figure 4-46. Electronically Marked Direct Connect Device with No Cable with SOP' Powered by VCONN

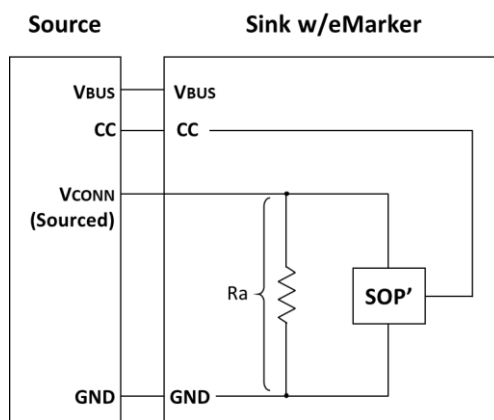
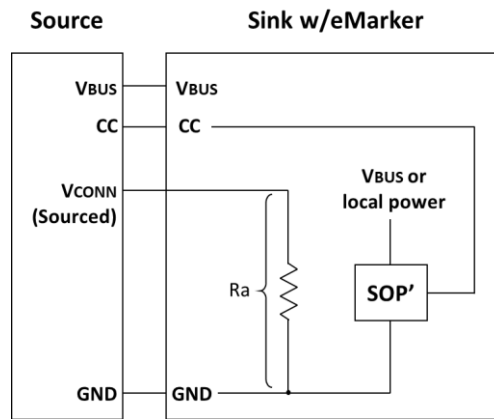


Figure 4-47 illustrates an electronically marked thumb drive where the SOP' is not powered by VCONN. While the SOP' doesn't rely on VCONN for power, R_a shall still be implemented but is not required to be weakened or adhere to the R_a Management State Machine defined in Section 4.5.2.5. Note: Even if the SOP' is integrated into the SOP, the timing requirements in Section 4.5.2.4 still apply.

Figure 4-47 Electronically Marked Direct Connect Device with No Cable with SOP' Powered by VBUS or Local Power



4.9.1 Parameter Values

Table 4-24 provides the power on timing requirements for ~~the an~~ eMarker ~~SOP' and SOP''~~ to be ready to communicate.

Table 4-24 ~~SOP' and SOP''~~eMarker Timing

	Maximum	Description
V_{CONN} Stable to eMarker Ready	50 ms	The time between the application of V_{CONN} or VBUS, whichever is later, until SOP' and SOP'' an eMarker shall be ready for communication.

4.9.2 Active Cables

An active cable is an electronically marked cable that incorporates data bus signal conditioning circuits, for example to allow for implementing longer cables. Active cables with data bus signal conditioning in both plugs **shall** implement SOP' and **may** implement SOP''. Active cables **shall** meet the power requirements defined in Table 4-6.

Active cables **may** support either one TX/RX pair or two TX/RX pairs. The eMarker in the cable **shall** identify the number of TX/RX lanes supported. Active cables **may** or **may not** require configuration management. Active cable configuration management is defined in Section 6.

4.10 VCONN-Powered Accessories (VPAs) and VCONN-Powered USB Devices (VPDs)

VCONN-Powered Accessories and **VCONN-Powered USB Devices** are both direct-attach Sinks that can operate with just VCONN.

Both expose a maximum impedance to ground of R_a on the VCONN pin and R_d on the CC pin.

The removal of VCONN when VBUS is not present **shall** be treated as a detach event.

4.10.1 VCONN-Powered Accessories (VPAs)

A **VCONN-Powered Accessory** implements an **Alternate Mode** (See Appendix E).

VCONN-Powered Accessories **shall** comply with Table 4-7.

When operating in the Sink role and when VBUS is not present, **VCONN-Powered Accessories** *shall* treat the application of VCONN as an attach signal and *shall* respond to **USB Power Delivery** messages.

When powered by only VCONN, a **VCONN-Powered Accessory** *shall* negotiate an **Alternate Mode**. If it fails to negotiate an **Alternate Mode** within **tAMTimeout**, its port partner removes VCONN.

When VBUS is supplied, a **VCONN-Powered Accessory** is subject to all the requirements for **Alternate Modes**, including presenting a **USB Billboard Device Class** interface if negotiation for an **Alternate Mode** fails.

Should a **VCONN-Powered Accessory** wish to provide charge-through functionality, it must do so by negotiating voltage and current independently on both the Host and charge-through ports, and possibly re-regulating the voltage from the Source before passing it through to the Sink. The Sink can take the full current that is advertised to it by the **VCONN-Powered Accessory**.

4.10.2 VCONN-Powered Devices (VPDs)

A **VCONN-Powered USB Device** *shall* implement a USB UFP endpoint.

VCONN-Powered USB Devices *shall* comply with Table 4-8.

When VBUS is not present, **VCONN-Powered USB Devices** *shall* treat the application of VCONN as an attach signal.

A **VCONN-Powered USB Device** *shall* respond to **USB PD** messaging on SOP' and *shall not* respond to other **USB PD** messaging. A **VCONN-Powered USB Device** *shall* respond to **USB PD** Hard Reset and Cable Reset signaling.

A Charge-Through **VCONN-Powered USB Device** *shall* discard all **USB PD** messages while a connection is enabled between the host port CC and Charge-Through port CC.

When VBUS is supplied by the Host, the **VCONN-Powered USB Device** *shall* behave like a normal UFP Sink, but still only respond to **USB PD** messaging on SOP'. If VBUS is subsequently removed while VCONN remains applied, the **VCONN-Powered USB Device** *shall* remain connected, and use VCONN as the sole detach signal.

Since **VCONN-Powered USB Devices** do not respond to **USB PD** on SOP, they cannot enter **Alternate Modes**.

A **VCONN-Powered USB Device** *may* provide Charge-Through functionality via **VPD** Charge-Through. **VCONN-Powered USB Devices** *shall not* provide any data pass-through to the Charge-Through port other than the CC wire.

Since the power and CC negotiation is passed through directly, the Sink *shall* limit its maximum current based on the additional impedance introduced by the **VCONN-Powered USB Device**.

Additionally, since power can only flow from the Charge-Through port to the Host, VCONN must be provided by the host, and there is no data connection beyond the CC wire passed through to the connected source, there are limitations on what the Host can advertise and support via **USB PD**:

- The Host *shall not* negotiate or accept a **PR_Swap** or **VCONN_Swap**.
- The Host *shall not* enable **FR_Swap**.
- The Host *may* only negotiate a **DR_Swap** when using **USB PD** Revision 2.0, and only for the purpose of switching which side initiates PD communications. The Host will always remain a DFP for USB data.
- The Host *shall not* advertise dual-role data or dual-role power in its Source Capability or Sink Capability messages – Host changes its advertised capabilities to UFP role/sink only role.
- The Host *shall not* negotiate any **Alternate Modes** that change the function of pins on the connector.

- The Host **shall** represent itself to the Charge-Through Source using **USB PD** as if it were a Sink-only, data-less device.

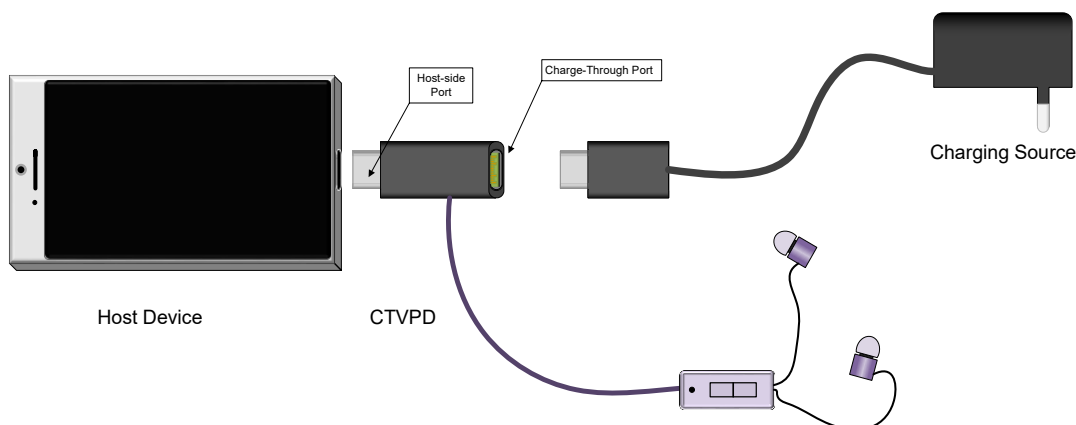
Table 4-25 Charge-Through VPD CC Impedance (RccCON) Requirements

	Minimum	Maximum	Description
RccCON		15 Ω	Impedance in the Charge-Through VPD while a connection is enabled between the host port CC and Charge-Through port CC.
	zOPEN		Impedance between the host port CC and Charge-Through CC when a connection is disabled.

Table 4-26 CTVPD Charge-Through Port VBUS Bypass Requirements

	Minimum	Maximum	Description
CCTB	1 μ F	10 μ F	Bypass capacitance on Charge-Through port VBUS connection to support ADP max C _{ADP_THR} .

Figure 4-48 Example Charge-Through VCONN-Power USB Device Use Case



4.11 Parameter Values

4.11.1 Termination Parameters

Table 4-27 provides the values that **shall** be used for the Source's **Rp** or current source. Other pull-up voltages **shall** be allowed if they remain less than 5.5 V and fall within the correct voltage ranges on the Sink side. Note: when two Sources are connected together, they **may** use different termination methods which could result in unexpected current flow.

Table 4-27 Source CC Termination (Rp) Requirements

Source Advertisement	Current Source to 1.7 – 5.5 V	Resistor pull-up to 4.75 – 5.5 V	Resistor pull-up to 3.3 V ± 5%
Default USB Power	80 μ A ± 20%	56 k Ω ± 20% (Note 1)	36 k Ω ± 20%
1.5 A @ 5 V	180 μ A ± 8%	22 k Ω ± 5%	12 k Ω ± 5%
3.0 A @ 5 V	330 μ A ± 8%	10 k Ω ± 5%	4.7 k Ω ± 5%
Note 1: For Rp when implemented in the USB Type-C plug on a USB Type-C to USB 3.1 Standard-A Cable Assembly, a USB Type-C to USB 2.0 Standard-A Cable Assembly, a USB Type-C to USB 2.0 Micro-B Receptacle Adapter Assembly or a USB Type-C captive cable connected to a USB host, a value of 56 k Ω ± 5% shall be used, in order to provide tolerance to IR drop on VBUS and GND in the cable assembly. Note 2: Each value above is the total current or resistance into the Source CC pin including all leakage and parallel resistance.			

The Sink **may** find it convenient to implement **Rd** in multiple ways simultaneously (a wide range **Rd** when unpowered and a trimmed **Rd** when powered). Transitions between **Rd** implementations that do not exceed **tCCDebounce** **shall not** be interpreted as exceeding the wider **Rd** range. Transitions between **Rd** implementations **shall not** allow the voltage on CC to go outside the voltage band that defines a connection. Table 4-28 provides the methods and values that **shall** be used for the Sink's **Rd** implementation.

Table 4-28 Sink CC Termination (Rd) Requirements

Rd Implementation	Nominal Value	Can detect power capability?	Max voltage on pin
± 20% voltage clamp ¹	1.1 V	No	1.32 V
± 20% resistor to GND ²	5.1 k Ω	No	2.18 V
± 10% resistor to GND ²	5.1 k Ω	Yes	2.04 V
Note 1: The clamp implementation inhibits USB PD communication although the system can start with the clamp and transition to the resistor once it is able to do USB PD . Note 2: This is the total equivalent resistance into the Sink CC pin including all internal resistances.			

Table 4-29 provides the impedance value to ground on VCONN in powered cables.

Table 4-29 Powered Cable Termination Requirements

	Minimum Impedance	Maximum Impedance
Ra	800 Ω ¹	1.2 k Ω

Notes:

1. The minimum impedance **may** be less when VCONN is not applied. The current consumed from VCONN **shall** be as specified in Table 4-6, Table 4-7, and Table 4-8 when the voltage is less than vVCONNValid.
2. The voltage across **Ra** when connected to **Rp** is determined by equations defined in Section 4.11.3.1.2.

Table 4-30 provides the minimum impedance value to ground on CC for a device (Sink or Source) to be undetected by a Source. This **shall** apply for ports in the **Disabled** state or **ErrorRecovery** state. This **shall** also apply for Sources when unpowered (for example a power brick unplugged from AC mains).

**Table 4-30 CC Termination Requirements for Disabled state,
ErrorRecovery state, and Unpowered Source**

	Minimum Impedance to GND	Notes
<i>zOPEN</i>	126 kΩ	<i>zOPEN shall</i> only be evaluated below <i>vCC-Clamp</i> .

Table 4-31 provides the impedance value for an SBU to appear open.

Table 4-31 SBU Termination Requirements

	Termination	Notes
<i>zSBUTermination</i>	≥ 950 kΩ	Functional equivalent to an open circuit.

4.11.2 Timing Parameters

Table 4-32 provides the timing values that **shall** be met for delivering power over VBUS and VCONN.

Table 4-32 VBUS and VCONN Timing Parameters

	Minimum	Maximum	Description
<i>tVBUSON</i>	0 ms	275 ms	From entry to <i>Attached.SRC</i> until VBUS reaches the minimum <i>vSafe5V</i> threshold as measured at the source's receptacle.
<i>tVBUSOFF</i>	0 ms	650 ms	From the time the Sink is detached until the Source removes VBUS and reaches <i>vSafe0V</i> (See <i>USB PD</i>).
<i>tVsafe0V</i>	0 ms	650 ms	The time the Source is allowed to use for discharging the Sink capacitors to reach <i>vSafe0V</i> in <i>AttachWait.SRC</i> state in order to progress to <i>Attached.SRC</i> .
<i>tBulkDischarge</i>	0 ms	275 ms	From the time the Source exits the <i>Attached.SRC</i> state with $V_{BUS} \leq 21$ V until the Source is prepared to re-enable VBUS.
		700 ms	From the time the Source exits the <i>Attached.SRC</i> state with $V_{BUS} > 21$ V until the Source is prepared to re-enable VBUS.
<i>tVCONNON</i>	Note 1	2 ms	From the time the Source supplied VBUS in the <i>Attached.SRC</i> state. Measured from <i>vSafe5V</i> to the minimum VCONN voltage (see Table 4-5)
<i>tVCONNON-PA</i>	0 ms	100 ms	From the time a Sink with accessory support enters the <i>Powered.Accessory</i> state until the Sink sources minimum VCONN voltage (see Table 4-5)
<i>tVCONNOFF</i>	0 ms	35 ms	From the time that a Sink is detached or as directed until the VCONN supply is disconnected.
<i>tSinkAdj</i>	<i>tRpValueChange</i> (Min)	60 ms	Response time for a Sink to adjust its current consumption to be in the specified range due to a change in <i>USB Type-C Current</i> advertisement.

Note 1: VCONN **may** be applied prior to the application of VBUS.

Figure 4-49 shows an example Source and defines VBULK for illustrating the *tBulkDischarge* parameter in conjunction with Figure 4-50 and Figure 4-51.

Figure 4-49 Example Source Implementation

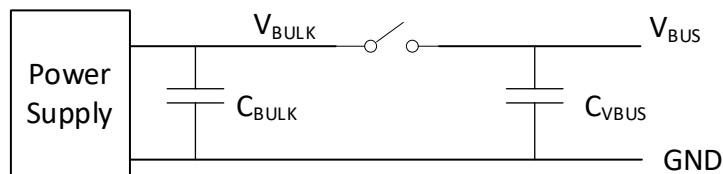


Figure 4-50 shows a scenario following a contract with VBUS above 21 volts. In this example, the *tCCDebounce* timer completes before the Source is ready to supply VBUS at *vSafe5V* since $V_{BULK} > vSafe5V$. Note that many implementations will discharge VBUS below *vSafe0V* much faster than *tVBUSOFF* (Maximum).

Figure 4-50 Exiting from *Attached.SRC* with slow discharging VBULK

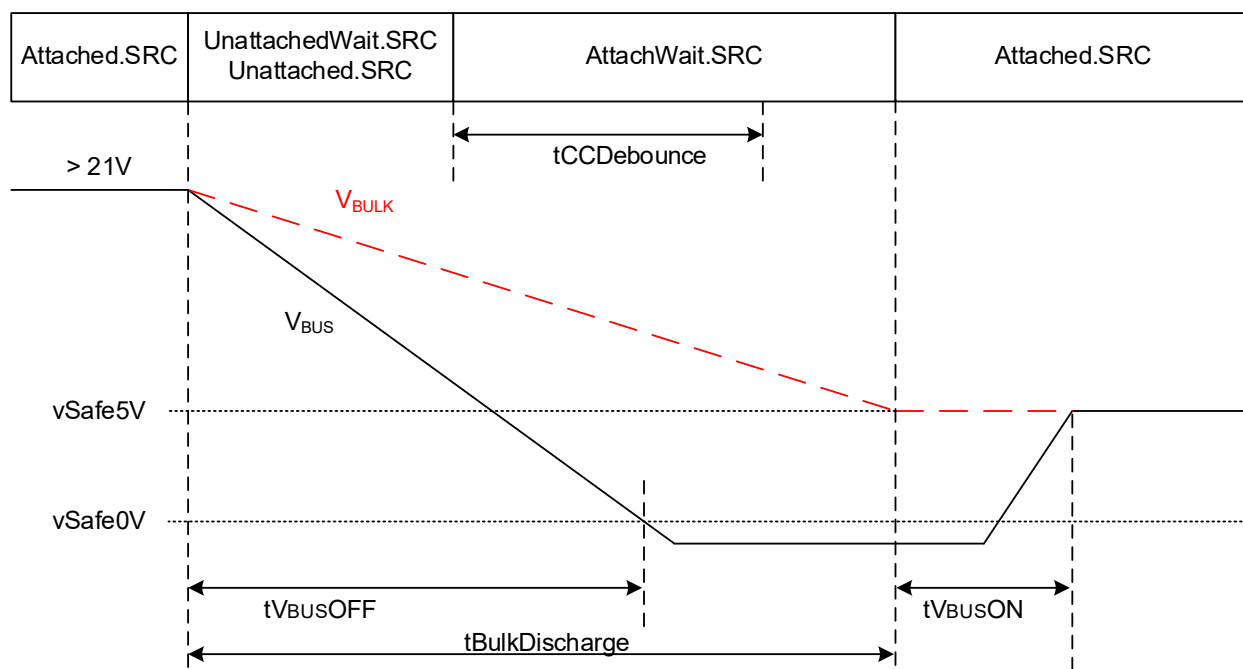


Figure 4-51 shows a scenario following a contract with VBUS at 21 volts or lower. In this example, VBULK is discharged to *vSafe5V* before the *tCCDebounce* timer completes. Note that many implementations will discharge VBUS below *vSafe0V* much faster than *tVBUSOFF* (Maximum).

Figure 4-51 Exiting from *Attached.SRC* with fast discharging VBULK

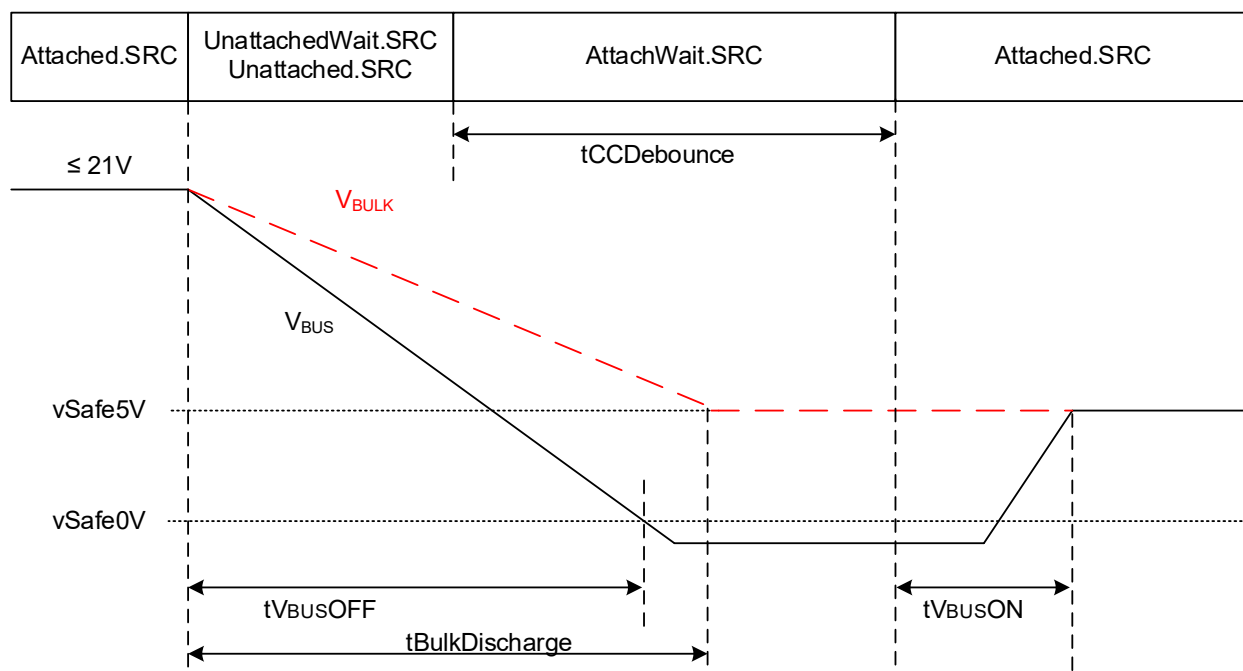


Figure 4-52 illustrates the timing parameters associated with the DRP toggling process. The ***tDRP*** parameter represents the overall period for a single cycle during which the port is exposed as both a Source and a Sink. The portion of the period where the DRP is exposed as a Source is established by ***dcSRC.DRP*** and the maximum transition time between the exposed states is dictated by ***tDRPTransition***.

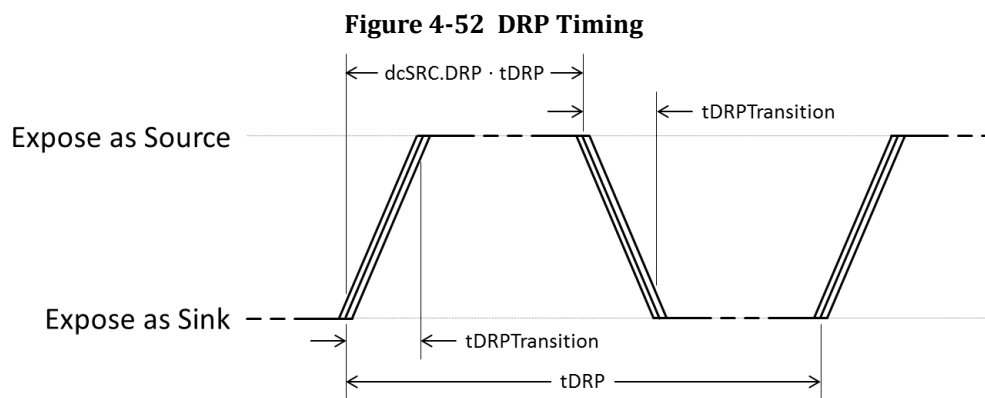


Table 4-33 provides the timing values that ***shall*** be met for DRPs. The clock used to control DRP swap ***should not*** be derived from a precision timing source such as a crystal, ceramic resonator, etc. to help minimize the probability of two DRP devices indefinitely failing to resolve into a Source-to-Sink relationship. Similarly, the percentage of time that a DRP spends advertising Source ***should not*** be derived from a precision timing source.

Table 4-33 DRP Timing Parameters

	Minimum	Maximum	Description
<i>tDRP</i>	50 ms	100 ms	The period a DRP <i>shall</i> complete a Source to Sink and back advertisement.
<i>dcSRC.DRP</i>	30%	70%	The percent of time that a DRP <i>shall</i> advertise Source during <i>tDRP</i> .
<i>tDRPTransition</i>	0 ms	1 ms	The time a DRP <i>shall</i> complete transitions between Source and Sink roles during role resolution.
<i>tDRPTry</i>	75 ms	150 ms	Wait time associated with the <i>Try.SRC</i> state.
<i>tDRPTryWait</i>	400 ms	800 ms	Wait time associated with the <i>Try.SNK</i> state.
<i>tTryTimeout</i>	550 ms	1100 ms	Timeout for transition from <i>Try.SRC</i> to <i>TryWait.SNK</i> .
<i>tVPDDetach</i>	10 ms	20 ms	Time for a DRP to detect that the connected Charge-Through VCONN-Powered USB Device has been detached, after VBUS has been removed.

Table 4-34 provides the timing requirement for CC connection behaviors.

Table 4-34 CC Timing Parameters

	Minimum	Maximum	Description
<i>tCCDebounce</i>	100 ms	200 ms	Time a port shall wait before it can determine it is attached.
<i>tPDDebounce</i>	10 ms	20 ms	Time a Sink port shall wait before it can determine it is detached due to the potential for USB PD signaling on CC as described in the state definitions.
<i>tTryCCDebounce</i>	10 ms	20 ms	Time a port shall wait before it can determine it is re-attached during the try-wait process.
<i>tErrorRecovery</i>	25 ms		Time a self-powered port shall remain in the ErrorRecovery state.
	240 ms		Time a source shall remain in the ErrorRecovery state if it was sourcing VCONN in the previous state.
<i>tRpValueChange</i>	10 ms	20 ms	Time a Sink port shall wait before it can determine there has been a change in Rp where CC is not BMC Idle, or the port is unable to detect BMC Idle.
	0 ms	5 ms	Time a Sink port shall wait before it can determine that there has been a change in Rp when USB PD signaling can be detected by the port and CC line is BMC Idle.
<i>tSRCDisconnect</i>	0 ms	20 ms	Time a Source shall detect the SRC.Open state. The Source should detect the SRC.Open state as quickly as practical.
<i>tNoToggleConnect</i>	0 ms	5 ms	Time to detect connection when neither Port Partner is toggling.
<i>tOnePortToggleConnect</i>	0 ms	80 ms	Time to detect connection when one Port Partner is toggling (0ms ... $dcSRC.DRP_{max} \cdot tDRP_{max} + 2 \cdot tNoToggleConnect$).
<i>tTwoPortToggleConnect</i>	0 ms	510 ms	Time to detect connection when both Port Partners are toggling (0ms ... $5 \cdot tDRP_{max} + 2 \cdot tNoToggleConnect$).
<i>tVPDCTDD</i>	30 μs	5 ms	Time for a Charge-Through VCONN -Powered USB Device to detect that the Charge-Through source has disconnected from CC after VBUS has been removed, transition to CTUnattached.VPD , and re-apply its Rp termination advertising 3.0 A on the host port CC.
<i>tVPDDisable</i>	25 ms		Time for a Charge-Through VCONN -Powered USB Device shall remain in CTDisabled.VPD state.

4.11.3 Voltage Parameters

Note: Starting with Release 2.4 of this specification, this section has been completely restructured, transitioning from a set of operational CC voltage range and threshold requirement tables in Release 2.3 to an approach that documents a design methodology for implementing CC threshold detection followed by Sink and Source CC voltage range and threshold examples.

This section provides voltage calculations to be used by a Source to detect an attach and a Sink to detect the **USB Type-C Current** advertisement (Default USB, 1.5 A @ 5 V, or 3.0 A @ 5 V) with some numeric examples. This allows calculations for thresholds of specific implementations to be based on the parameter variances of the implementation.

4.11.3.1 CC Detection Voltage Threshold Calculations

For calculating threshold settings to detect a Sink attach by a Source and the Source **USB Type-C Current** advertisement by a Sink, the equations in this section are used. The cable CC wire resistance is considered negligible compared to the Source R_p and Sink R_d and is not included. Each equation calculates the CC pin voltage as detected by the Sink or the Source. Figure 4-53 and Figure 4-54 illustrate the two implementation models considered for these equations, the first based on a pull-up resistor in the Source and the second replacing this with a current source.

Figure 4-53 Pull-Up/Pull-Down Voltage Detection Model

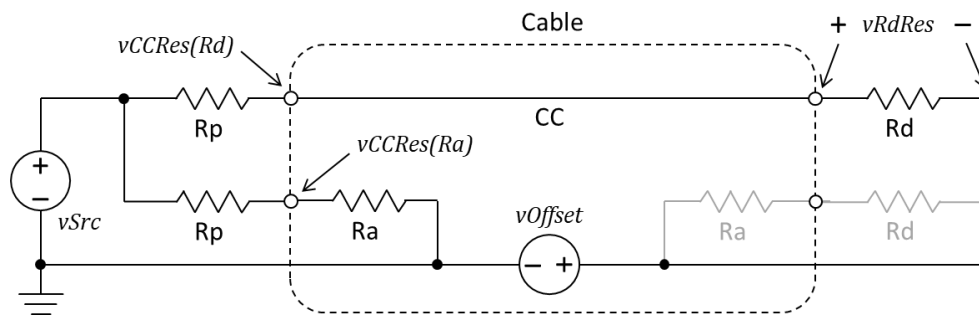
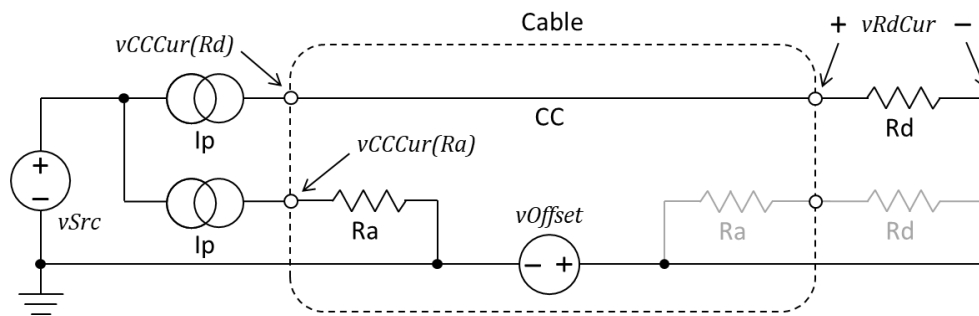


Figure 4-54 Current Source Voltage Detection Model



4.11.3.1.1 CC Detection Voltage on Sink CC Pins

The CC pin voltage detected by the Sink is the voltage across the Sink **Rd** pull-down resistor.

For the following calculations:

v_{Src} = voltage supply of the Source pull-up resistor,

R_p = Source pull-up resistor,

I_p = Source pull-up current,

R_d = Sink pull-down resistor,

v_{Offset} = the ground offset due to the cable IR drop when current is drawn by the Sink,

v_{RdRes} = voltage across the Sink R_d for a Source pull-up to v_{Src} through the resistor R_p , and

v_{RdCur} = voltage across the Sink R_d for a Source pull-up current of I_p .

$$v_{RdRes}(min) = (v_{Src}(min) - v_{Offset}(max)) \frac{R_d(min)}{R_d(min) + R_p(max)}$$

$$v_{RdRes}(max) = v_{Src}(max) \frac{R_d(max)}{R_d(max) + R_p(min)}$$

$$v_{RdCur}(min) = I_p(min) \cdot R_d(min)$$

$$v_{RdCur}(max) = I_p(max) \cdot R_d(max)$$

$$v_{Rd}(min) = \min(v_{RdRes}(min), v_{RdCur}(min))$$

$$v_{Rd}(max) = \max(v_{RdRes}(max), v_{RdCur}(max))$$

A Sink is not aware of the Source advertising method with either a current source or a resistive pull-up. The v_{Rd} for both a Source advertisement with a current source and a resistance must be calculated and the minimum and maximum taken. From the Sink perspective, the maximum variance for each parameter in the Source is the maximum for each source implementation, from Table 4-27. The $v_{Offset}(max)$ can depend on the maximum current drawn by the Sink, whether the connection is a direct attach, or assume the maximum 250 mV for a cable. The variance of **Rd** can be smaller than the maximum in Table 4-28. Thus, the Sink can use these equations to determine the optimum minimum and maximum thresholds for detecting the Source current advertisements.

4.11.3.1.2 CC Detection Voltage on Source CC Pins

The Source equations are used to determine when either an attach or detach occurs with a Sink (**Rd**) or a cable/accessory (**Ra**).

For the following calculations:

$vSrc$ = voltage supply of the source pull-up resistor,

Rp = Source pull-up resistor,

Ip = Source pull-up current,

Rd = Sink pull-down resistor,

$Rd|a$ = either a device pull-down resistor **Rd** or a cable/accessory pull-down resistor **Ra**,

$vOffset$ = the ground offset due to the cable IR drop when current is drawn by a sink,

$vCCRes$ = Source CC pin voltage for a Source pull-up to $vSrc$ through the resistor **Rp**, and

$vCCCur$ = Source CC pin voltage for a Source pull-up current of Ip .

The CC pin voltage seen by the source is given by the following equations:

$$vCCRes = (vSrc - vOffset) \frac{Rd|a}{Rd|a + Rp} + vOffset$$

$$vCCCur = Ip \cdot Rd|a + vOffset$$

To determine the minimum and maximum values, these equations become:

$$vCCRes(min) = vSrc(min) \frac{Rd(min)}{Rd(min) + Rp(max)}$$

$$vCCRes(max) = (vSrc(max) - vOffset(max)) \frac{Rd|a(max)}{Rd|a(max) + Rp(min)} + vOffset(max)$$

$$vCCCur(min) = Min(Ip(min) \cdot Rd(min), vRdclamp(min))$$

$$vCCCur(max) = Max(ip(max) \cdot Rd|a(max) + vOffset(max) , vRdclamp(max))$$

$vOffset$ is 0V at initial connect as the current drawn is negligible (no VBUS), therefore $vOffset$ is included in determining the disconnect threshold only. Note, the connect and disconnect thresholds can be the same in a design and in this case the higher value including the $vOffset(max)$ is used.

A Source is aware of whether it is advertising with a current source or a resistive pull-up and will use only the equations associated with its specific method. From the Source perspective, the sink voltage calculation must assume the maximum 20% variance of **Rd** or the clamp voltage as shown in Table 4-28. The $vOffset(max)$ may depend on the max current the Source is capable of offering or the maximum 250 mV may be assumed. The variance of **Rp** and Ip can be smaller than the maximum in Table 4-27.

4.11.3.2 CC Voltage Threshold Examples

The following provides the calculations using the equations for determining threshold settings for example designs.

4.11.3.2.1 Example of Voltage on Sink CC Pins – Among all Source advertisement ($R_d \pm 20\%$)

Table 4-35 shows the range of the Sink CC pin voltage (v_{Rd}) for a R_d with a variance of $\pm 20\%$ and clamp voltage variation of $\pm 20\%$. ~~If a Sink implementation used the CC pin voltage as connect detection, then a threshold above the maximum from this table would be required.~~

Table 4-35 Sink CC Pin Voltages for Connect Detection for R_d and Clamp Voltage $\pm 20\%$

	Minimum Voltage (V)	Maximum Voltage (V)
$v_{Rd-\text{connect}}$	0.249	2.181
Threshold for detecting disconnect		0.248

4.11.3.2.2 Example of Voltage on Sink CC Pins – Multiple Source Current Advertisements ($R_d \pm 10\%$)

Table 4-36 shows the range of the Sink CC pin voltage and resulting thresholds for detecting a connect/disconnect and/or the *USB Type-C Current* advertisement from the Source.

Table 4-36 Sink CC Pin Voltages for Connect and Current Advertisement Detection for $R_d \pm 10\%$

	Minimum Voltage (V)	Maximum Voltage (V)
v_{Rd-USB}	0.277	0.612
$v_{Rd-1.5}$	0.746	1.164
$v_{Rd-3.0}$	1.369	2.042
$v_{Rd-\text{connected}}$	0.277	2.042
Threshold between v_{Rd-USB} and $v_{Rd-1.5}$	0.613	0.745
Threshold between $v_{Rd-1.5}$ and $v_{Rd-3.0}$	1.165	1.368
Disconnect/Connect Threshold for detecting disconnect	2.043	0.276

4.11.3.2.3 Example of Voltage on Source CC Pins – Current Source

Table 4-37 shows the range of the Source CC pin voltage for the three different current advertisements when using a pull-up current for the advertisement. These voltages are based on the variance found in Table 4-27.

Table 4-37 Source CC Pin Voltages for All Current Advertisements using a Pull-Up current

	Minimum Voltage (V)	Maximum Voltage (V)
Connect Threshold $v_{CCCur-Rd-USB}$	0.261	1.32 ⁽²⁾
Disconnect Threshold² – v_{Rd-USB}	1.321	
Connect Threshold $v_{CCCur-Rd-1.5}$	0.676	1.32 ⁽²⁾
Disconnect Threshold¹ – $v_{Rd-1.5}$	1.441	
Connect Threshold $v_{CCCur-Rd-3.0}$	0.88 ⁽²⁾	2.181
Disconnect Threshold¹ – $v_{Rd-3.0}$	2.432	
$v_{CCCur-Ra-USB}$	0 ⁽³⁾	0.115
$v_{CCCur-Ra-1.5}$	0 ⁽³⁾	0.233
$v_{CCCur-Ra-3.0}$	0 ⁽³⁾	0.428
Threshold between $v_{CCCur-Rd-USB}$ and $v_{CCCur-Ra-USB}$	0.116	0.260
Threshold between $v_{CCCur-Rd-1.5}$ and $v_{CCCur-Ra-1.5}$	0.234	0.675
Threshold between $v_{CCCur-Rd-3.0}$ and $v_{CCCur-Ra-3.0}$	0.429	0.879
Note 1: Disconnect threshold should take v_{Offset} into account. $v_{Offset} = 0.25$ V is assumed here.		
Note 2: Result is from the voltage clamp variance.		
Note 3: Per Note 1 of Table 4-29, R_a minimum impedance may be less than $R_a(\min)$ when V_{CONN} is <i>not</i> applied.		

4.11.3.2.4 Example of Voltage on Source CC Pins – Rp @ 5 V

Table 4-38 shows the range of the Source CC pin voltage for a pull-up resistance to 5 V using the variance found in Table 4-27.

Table 4-38 Source CC Pin Voltages for Pull-Up Resistance to 5 V

	Minimum Voltage (V)	Maximum Voltage (V)
Connect Threshold <i>vCCRes-Rd-USB</i>	0.272	1.32 ⁽²⁾
Disconnect Threshold² – <i>vRd-USB</i>	1.321	2.93 ⁽⁴⁾
Connect Threshold <i>vCCRes-Rd-1.5</i>	0.713	1.32 ⁽²⁾
Disconnect Threshold¹ – <i>vRd-1.5</i>	1.440	2.44 ⁽⁴⁾
Connect Threshold <i>vCCRes-Rd-3.0</i>	0.88 ⁽²⁾	2.155
Disconnect Threshold¹ – <i>vRd-3.0</i>	2.308	2.66 ⁽⁴⁾
<i>vCCRes-Ra-USB</i>	0 ⁽³⁾	0.143
<i>vCCRes-Ra-1.5</i>	0 ⁽³⁾	0.299
<i>vCCRes-Ra-3.0</i>	0 ⁽³⁾	0.617
Threshold between <i>vCCRes-Rd-USB</i> and <i>vCCRes-Ra-USB</i>	0.144	0.271
Threshold between <i>vCCRes-Rd-1.5</i> and <i>vCCRes-Ra-1.5</i>	0.300	0.712
Threshold between <i>vCCRes-Rd-3.0</i> and <i>vCCRes-Ra-3.0</i>	0.618	0.879
Note 1: Disconnect threshold should take <i>vOffset</i> into account. <i>vOffset</i> = 0.25 V is assumed here. Note 2: Result is from the voltage clamp variance. Note 3: Per Note 1 of Table 4-29, <i>Ra</i> minimum impedance may be less than <i>Ra</i> (min) when <i>VCONN</i> is <i>not</i> applied. Note 4: Considers <i>zOPEN</i> minimum.		

4.11.3.2.5 Example of Voltage on Source CC Pins – Rp @ 3.3 V

Table 4-39 shows the range of the Source CC pin voltage for a pull-up resistance to 3.3 V using the variance found in Table 4-27.

Table 4-39 Source CC Pin Voltages for Pull-Up Resistance to 3.3 V

	Minimum Voltage (V)	Maximum Voltage (V)
Connect Threshold <i>vCCRes-Rd-USB</i>	0.271	1.32 ⁽²⁾
Disconnect Threshold ² – <i>vRd-USB</i>	1.321 ⁽²⁾	2.15 ⁽⁴⁾
Connect Threshold <i>vCCRes-Rd-1.5</i>	0.767	1.32 ⁽²⁾
Disconnect Threshold ¹ – <i>vRd-1.5</i>	1.374	2.62 ⁽⁴⁾
Connect Threshold <i>vCCRes-Rd-3.0</i>	0.88 ⁽²⁾	2.003
Disconnect Threshold ¹ – <i>vRd-3.0</i>	2.110	2.77 ⁽⁴⁾
<i>vCCRes-Ra-USB</i>	0 ⁽³⁾	0.139
<i>vCCRes-Ra-1.5</i>	0 ⁽³⁾	0.330
<i>vCCRes-Ra-3.0</i>	0 ⁽³⁾	0.734
Threshold between <i>vCCRes-Rd-USB</i> and <i>vCCRes-Ra-USB</i>	0.140	0.270
Threshold between <i>vCCRes-Rd-1.5</i> and <i>vCCRes-Ra-1.5</i>	0.331	0.766
Threshold between <i>vCCRes-Rd-3.0</i> and <i>vCCRes-Ra-3.0</i>	0.735	0.879
Note 1: Disconnect threshold should take <i>vOffset</i> into account. <i>vOffset</i> = 0.25 V is assumed here. Note 2: Result is from the voltage clamp variance. Note 3: Per Note 1 of Table 4-29, <i>Ra</i> minimum impedance may be less than <i>Ra</i> (min) when <i>VCONN</i> is <i>not</i> applied. Note 4: Considers <i>zOPEN</i> minimum.		

4.11.3.3 CC Pin Clamp Voltage

Table 4-40 provides the clamping voltage that any port (Source, Sink or DRP) **may** clamp its CC pin to protect from damage. The inclusion of clamping **shall not** impact the functionality when the CC pin is functioning as *VCONN* Source or Sink.

Table 4-40 CC Pin Clamping Voltage

	Minimum Voltage
<i>vCC-Clamp</i>	2.9 V

5 **USB4** Discovery and Entry

USB4 discovery and operational entry differs significantly from **USB 2.0** and **USB 3.2**. This chapter defines the process of discovering across a USB Type-C connection that both port partners are **USB4**-capable (or not), having the DFP-side of the link decide regarding to enter **USB4** operation (or not), and how operational entry is accomplished.

5.1 Overview of the Discovery and Entry Process

The following provides an overview of the general process for discovery and entry into **USB4** operation.

1. USB Type-C CC Connection State Machines resolve Source/Sink and the initial data roles (DFP/UFP).
2. Initial VBUS and VCONN power is supplied.
3. **USB Power Delivery** protocol is used to establish a power contract between the port partners.
4. **USB PD Discover Identity** process is used by the DFP to identify port partner (SOP) capabilities.
5. **USB PD Discover Identity** process is used by the DFP to identify cable (SOP') capabilities.
6. If the cable and port partner both support **USB4** operation, the DFP issues **USB PD Enter_USB** Messages to both the cable (if it is an active cable) and port partner to enter **USB4** operation.
7. If both port partners are Dual-Role-Data (DRD) capable, either the DFP or UFP can **optionally** initiate a data-role swap to exchange host and device roles.

The first three steps above are the same as used for all USB connections for establishing port relationships and power between the port partners. Step 5 where the cable is queried for its capabilities **may optionally** occur during Step 3, this would most likely be done before if the Source needs to know if the cable supports supplying current beyond 3 A.

Depending on the resulting power source relationship after the first few steps, the use of **USB PD DR_Swap** **may** be necessary to establish the port partner that is closest to the host as the data role DFP. For example, a hub supplying power to a host and **DR_Swap** is used to correct the data roles between the hub and host.

After the port partner's capabilities are identified by the DFP, it **may** be appropriate based on what is discovered about the port partner to also query the port partner using the **USB PD Alternate Mode** SVID discovery process as an extension to Step 4. There are situations where a port partner supports **Alternate Modes** that **may** also be useable during **USB4** operation and this would be discovered during this additional query.

After the cable capabilities are identified by the DFP, it **may** be appropriate based on what is discovered about the cable to also query the cable using the **USB PD Alternate Mode** SVID discovery process as an extension to Step 5. There are situations where a cable that supports **Thunderbolt 3 Alternate Mode** **may** also be useable for **USB4** operation, and this would be discovered during this additional query.

USB4 operation is entered using a **USB PD Enter_USB** Message. This message will be sent to both the cable (SOP', and SOP'' if present) and the port partner (SOP), each of which will respond with an **Accept** message to confirm and establish when the cable or port partner is functionally ready for **USB4** operation. If the cable to be used will be operating in **Thunderbolt 3 Alternate Mode**, then the cable will be enabled using the **USB PD Enter Mode Command** instead of the **USB PD Enter_USB** Message (See Appendix F).

USB4 functionally enables an ability for connecting two host platforms and establishing a data channel between the hosts, this is dependent on at least one of these host platforms being capable of Dual-Role-Data operation so that a proper USB Type-C DFP-to-UFP data relationship can be established between them. In most cases, both host platforms will be DRD-capable and once **USB4** operation is established, either of these host platforms can choose to initiate a change of its role in the DFP-to-UFP relationship. To accomplish this, the **USB PD DR_Swap** process is used during Step 7 listed above.

5.2 USB4 Functional Requirements

The following functional requirements are for **USB4** hosts and devices.

5.2.1 USB4 Host Functional Requirements

USB4 hosts **shall** meet the following functional requirements:

- **USB4** hosts with dual-role-data (DRD) support **shall** respond to **USB PD Discover Identity** command with both DFP and UFP VDOs.

5.2.2 USB Device Functional Requirements

USB4 devices **shall** meet the following functional requirements:

- **USB4** devices **shall** respond to **USB PD Discover Identity** command with UFP VDOs.
- The **USB4** device **shall** provide a USB interface exposing a **USB Billboard Device Class** when it cannot connect as a **USB4** device and doesn't support and expose equivalent functionality over an **Alternate Mode**, **USB 3.2** and/or **USB 2.0** within **tUSB4Timeout**.
- If the **USB4** device supports **Alternate Modes**, the device **shall** complete the **USB4** discovery and entry process (successful or not) before falling back to **USB 3.2** or **USB 2.0** and, when required, exposing an appropriate **USB Billboard Device Class**.

5.2.3 USB4 Alternate Mode Support

The **USB4** specification enables products to be designed to support **Alternate Modes**, specifically **DisplayPort Alt Mode** and **Thunderbolt 3 Alt Mode**. Unlike **USB 3.2** and **USB 2.0** hubs, this also includes supporting specific **Alternate Modes** on **USB4** hubs.

5.2.3.1 USB4 Alternate Mode Support on Hosts

For **USB4** hosts that implement **DisplayPort Tunneling**, **DP Alt Mode** with Multi-function support (DP_BR 1 channel signaling combined with **USB 3.2** support) as defined by the **DP Alt Mode** specification **shall** be implemented on all its USB Type-C DFPs.

The **USB4** host **shall** support the first connected **DisplayPort** display on any of its USB Type-C ports. Support for subsequently connected **DisplayPort** displays is **optional**.

USB4 hosts **may optionally** implement **TBT3** compatibility support as defined by the **USB4** specification on its USB Type-C DFPs.

5.2.3.2 USB4 Alternate Mode Support on Hubs and USB4-based Docks

USB4 hubs and **USB4**-based docks **shall** implement **DP Alt Mode** with Multi-function support (DP_BR 1 channel signaling combined with **USB 3.2** support) as defined by the **DP Alt Mode** specification on all exposed USB Type-C DFPs.

USB4 hubs **shall** support the first connected **DisplayPort** display on any of its USB Type-C DFPs. Support for subsequently connected **DisplayPort** displays is **optional**.

USB4-based docks **shall** support the first connected display on any of its USB Type-C DFPs or non-USB display connectors (if present, collectively). Support for subsequent connected displays is **optional**.

USB4 hubs and **USB4**-based docks **shall** implement **TBT3** compatibility support as defined by the **USB4** specification on its USB Type-C DFPs. **USB4** hubs and **USB4**-based docks **may** implement **TBT3** compatibility support as defined by the **USB4** specification on its USB Type-C UFP.

For **USB4** hubs, downstream-facing ports **shall not** implement **Alternate Modes** that do not have a **USB-IF** Standard ID (SID) or Accessory Modes.

5.3 USB Power Requirements

USB4 requires that the power connection between the port partners be established and maintained using **USB PD** prior to and through-out **USB4** operation. A **USB4** port, prior to entering **USB4** operation, **shall** operate as a **USB 3.2** port with regarding power (See Section 4.6). **USB4** does not use the USB device protocols defined in **USB 2.0** and **USB 3.2** for managing device power.

5.3.1 Source Power Requirements

For USB Type-C ports that support **USB4** data bus operation, the following requirements **shall** be met.

- **USB4** ports **shall** be minimally capable of supplying at least 7.5 W (i.e., 5 V @ 1.5 A) on VBUS to bus-powered **USB4** devices.
- The minimum capability for powering bus powered devices on data-capable ports **shall** be independently met on each USB Type-C port of a multi-port host or hub.

5.3.2 Sink Power Requirements

For **USB4** devices that rely on bus power to operate (independent of any charging needs), the following requirements **shall** be met.

- **USB4** devices **shall** draw only up to 250 mA on VBUS when the Source advertises Default USB power (see Section 4.11.1) prior to a **USB PD** power contract being made between the device and its port partner.
 - **USB4** devices **may** draw higher levels of power prior to a **USB PD** power contract being made if the Source advertises **USB Type-C Current** at either 1.5 A or 3.0 A.
- **USB4** devices **shall** be minimally capable of operating with a Source that only delivers up to 7.5 W (i.e., 5 V @ 1.5 A).
- **USB4** devices **shall not** enter **USB4** data bus operation until after a **USB PD** power contract has been established, and while in **USB4** operation, the device **shall** adhere to **USB PD** power behavioral requirements at all times including appropriately responding to changes in Source capabilities.

In cases where the full functional capabilities or the highest performance of the **USB4** device requires more than the power being offered by the host, the device **shall** be minimally capable of providing the user with basic functionality as expected for the type and listed functions of the device. This allows for making available a higher level of operation or performance when a higher level of power is supplied, e.g., 15 W for full functionality versus 7.5 W for basic functionality. In this case, the device **shall** expose a **USB Billboard Device Class** if the device is unable to provide equivalent functionality limited by the available power.

5.3.3 Device Power Management Requirements

For **USB4**, device power management is enabled using a combination of **USB PD** capabilities that operate in conjunction with the **USB4** link states of the device's UFP connection.

The connection between the device's UFP and its DFP port partner can be put into a suspend state based on the value of the USB Suspend Supported Flag in the Source-Capabilities Message used in the **USB PD** explicit power contract.

When the USB Suspend Supported Flag is set by the Source, the Sink **shall** meet the Suspend power requirement when the **USB4** link is in the CLd state. Prior to the entry of the link into CLd state, it is expected that the host will have placed all of the device's functions into an appropriate suspend state.

Suspend power is defined based on the capabilities of the **USB4** device:

- **USB4** Device that is not capable of remote wake or has remote wake disabled: 25 mW.
- **USB4** Device that supports remote wake and has remote wake enabled: 50 mW.

If the Source clears the USB Suspend Supported Flag, the Sink **shall** follow Explicit Contract power requirements regardless of the **USB4** link state. For **USB4**, the use of **USB PD** zero negotiated current is not a valid Suspend entry method since it is not coordinated with the host operating system and the function device drivers.

5.4 **USB4** Discovery and Entry Flow Requirements

This section provides the detailed requirements for **USB4** discovery and entry. Additional requirements related to **USB4** operation are in the **USB4** specification.

Prior to entering and during **USB4** operation, the functional requirements of Chapter 4 **shall** be met including all functional interface and configuration channel (CC) requirements.

5.4.1 USB Type-C Initial Connection

For a **USB4**-capable port, prior to initiating **USB4** cable and device discovery, a valid Source-to-Sink connection **shall** exist and the USB Type-C connection state machine of the port **shall** either be in the **Attached.SRC** or **Attached.SNK** state.

When two **USB4** dual-role-data (DRD) ports are connected together, e.g., two **USB4** hosts, USB Type-C connection process will establish the initial data roles between the port partners.

Once the initial data roles are established, the **USB4** DFP **may** immediately proceed to train the link for **USB 3.2**. If a UFP is **USB4** capable, it **shall** hold off exposing **USB 2.0** speed identification (Section 7.1.5. of the **USB 2.0** specification) and **USB 3.2** receiver terminations until receipt of an **USB PD Enter_USB** message, completing entry (including any pin reassignment) into an **Alternate Mode**, or the **tUSB4Timeout** has expired.

5.4.2 USB Power Delivery Contract

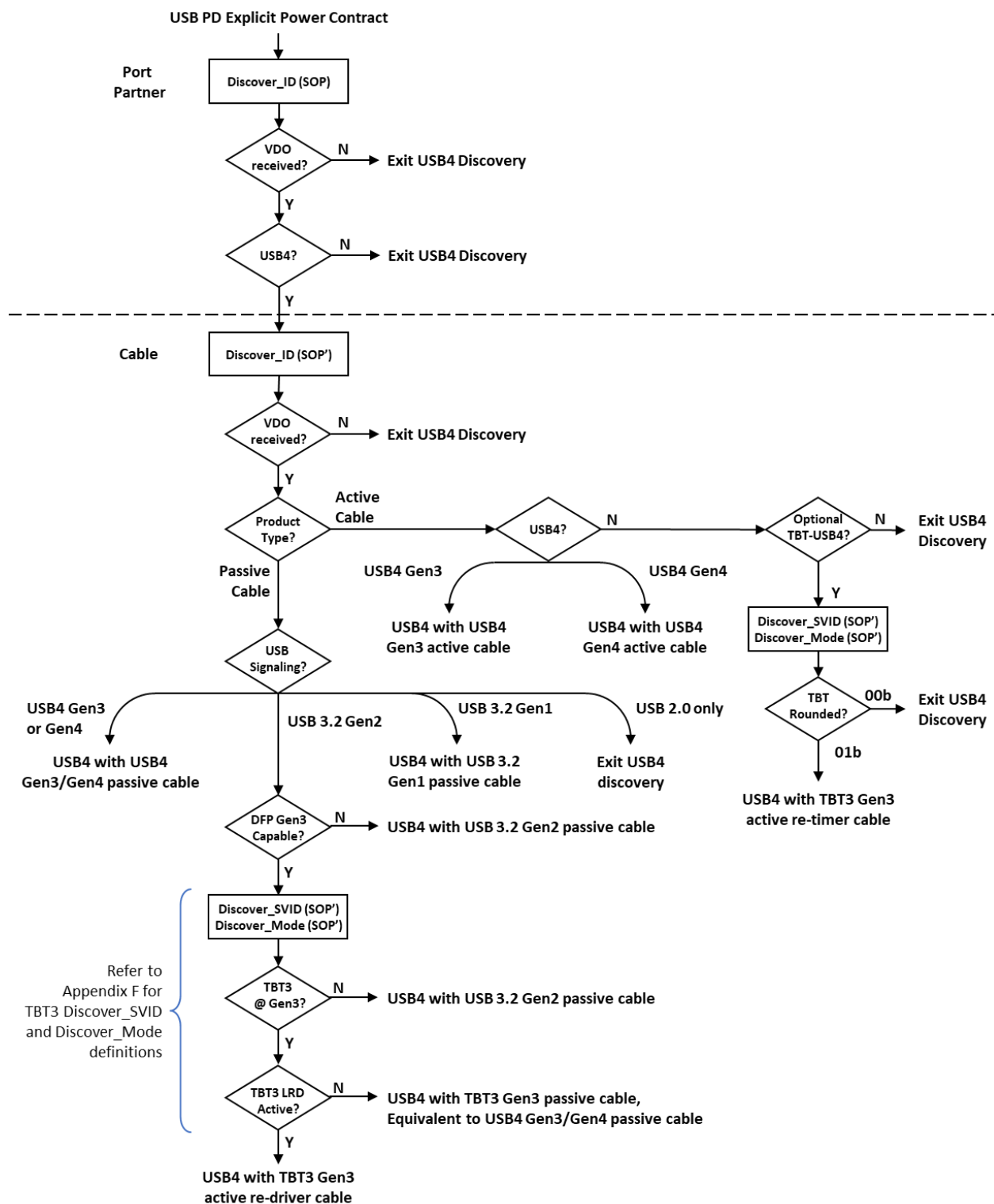
Prior to initiating **USB4** device discovery, the port partners **shall** negotiate a **USB PD** Explicit Contract.

During the process of establishing a stable **USB PD** Explicit Contract, the Source or Sink **may** have initiated power-role and VCONN swaps. Prior to moving on to **USB4** discovery, the functional data role **shall** be properly established (e.g., a self-powered hub upstream facing port is a DRP and comes up as a Source where **DR_Swap** is then required to the correct data role to the hub) and the DFP **shall** be the source of VCONN.

5.4.3 **USB4** Discovery and Entry Flow

Figure 5-1 illustrates the basic flow model for **USB4** discovery and entry.

Figure 5-1 *USB4* Discovery and Entry Flow Model



5.4.3.1 *USB4* Port Partner Discovery (SOP)

USB4 port partner discovery *shall* occur only after having a negotiated *USB PD* Explicit Contract.

USB4 port partner discovery involves the use of the **USB PD** Discover ID process between the DFP and its port partner (SOP).

The **USB4** port partner to be discovered can be any data-capable product presenting itself as the UFP port partner during the discovery process. Depending on the type and capabilities of the port partner, either or both UFP and DFP VDOs **may** be returned in response to the DFP's **USB PD** Discover_ID(SOP) command. As long as at least one of those VDOs are received, then the DFP **should** proceed to assess its port partner's ability to support **USB4** operation based on the content of the VDOs received. For a received UFP VDO, this is based on the Device Capability Field (B27...24) and for a received DFP VDO, this is based on the Host Capability Field (B26...24).

Refer to Appendix I for more information and guidance on the use of UFP and DFP Product Types and VDOs.

5.4.3.2 **USB4 Cable Discovery (SOP')**

The DFP **shall** determine that the attached cable is **USB4**-compatible prior to entering into **USB4** operation. In cases where the **USB4** device is directly connected or has a captive cable, the **USB4** device **shall** respond to **USB4** cable discovery on SOP' as a captive passive cable and indicating the appropriate USB Signaling support.

Table 5-1 summarizes the list of cables that are intended to support **USB4**-compatible operation. Regarding Active Cables, this list does not include Optically-Isolated Active Cables (OIACs) which are to be handled as a special case given that these cables do not support **USB 2.0** and power delivery over the cable (See Chapter 6).

Table 5-1 Certified Cables Where *USB4*-compatible Operation is Expected

	Cable Signaling	<i>USB4</i> Operation	Notes
USB Type-C Full-Featured Cables (Passive)	<i>USB 3.2</i> Gen 1	20 Gbps	This cable will indicate support for <i>USB 3.2</i> Gen1 (001b) in the USB Signaling field of its Passive Cable VDO response. Note: even though this cable isn't explicitly tested, certified or logo'ed for <i>USB 3.2</i> Gen2 operation, <i>USB4</i> Gen2 operation will generally work.
	<i>USB 3.2</i> Gen 2 (<i>USB4</i> Gen2)	20 Gbps	This cable will indicate support for <i>USB 3.2</i> Gen2 (010b) in the USB Signaling field of its Passive Cable VDO response.
	<i>USB4</i> Gen4 (<i>USB4</i> Gen3)	80 Gbps	This cable will indicate support for <i>USB4</i> Gen3 (011b) or <i>USB4</i> Gen4 (100b) in the USB Signaling field of its Passive Cable VDO response.
Thunderbolt™ 3 Cables (Passive)	<i>TBT3</i> Gen2 (<i>USB 3.2</i> Gen 1 or <i>USB4</i> Gen2)	20 Gbps	This cable will indicate support for <i>USB 3.2</i> Gen1 (001b) or <i>USB 3.2</i> Gen2 (010b) in the USB Signaling field of its Passive Cable VDO response.
	<i>TBT3</i> Gen3 (<i>USB4</i> Gen4³)	80 Gbps	In addition to indicating support for <i>USB 3.2</i> Gen2 (010b) in the USB Signaling field of its Passive Cable VDO response, this cable will indicate that it supports <i>TBT3</i> Gen3 in the Cable Speed parameter and Passive Cable in the Active_Passive parameter of the Discover Mode VDO response. ²
Thunderbolt™ 3 Cables (Active Linear Re-driver)	<i>TBT3</i> Gen3 (<i>USB4</i> Gen3³)	40 Gbps	In addition to indicating support for <i>USB 3.2</i> Gen2 (010b) in the USB Signaling field of its Passive Cable VDO response, this cable will indicate that it supports <i>TBT3</i> Gen3 in the Cable Speed parameter and Active Cable in the Active_Passive parameter of the Discover Mode VDO response. ²
USB Type-C Full-Featured Cables (Active)¹	<i>USB4</i> Gen2	20 Gbps	This cable will indicate support for <i>USB4</i> Gen2 (010b) in the USB Signaling field of its Active Cable VDO response.
	<i>USB4</i> Gen3	40 Gbps	This cable will indicate support for <i>USB4</i> Gen3 (011b) in the USB Signaling field of its Active Cable VDO response.
	<i>USB4</i> Gen4	80 Gbps	This cable will indicate support for <i>USB4</i> Gen4 (100b) in the USB Signaling field of its Active Cable VDO response.
Note 1: SuperSpeed USB active cables do not support <i>USB4</i> -compatible operation. Note 2: See Appendix F.2.6 and Table F-11. Note 3: For the Cable Speed and Cable Type fields of the Enter_USB Message, the DFP should use the <i>TBT3</i> Discover Mode VDO response values instead of the values in the Passive Cable VDO response. See Section 5.4.3.3.			

Determining if the cable is ***USB4***-compatible starts the use of the ***USB PD*** Discover ID process between the DFP and the attached cable (SOP'). If no response is received when the DFP issues a ***USB PD*** Discover ID command to the cable, then the ***USB4*** discovery process ***shall*** be exited and the DFP ***shall*** proceed to establish a functional connection to its UFP port partner following traditional ***USB 2.0*** process.

USB4-compatible support is generally determined based on the USB Signaling Support indicated in the cable VDO responses appropriate for the type of cable it is, passive or active, with compatible passive cables being those that indicate USB signaling support for Gen1 or higher and with compatible active cables being those that indicate USB signaling support for Gen2 or higher.

When a passive cable is identified as a **USB 3.2** Gen2 cable and the DFP is Gen3 capable, the DFP needs to check further using **USB PD Alternate Mode** process to determine if the cable is a **Thunderbolt 3** passive cable supporting Gen3.

Some existing **Thunderbolt 3** active cables **may not** support **USB4** operation, discovery and use of this cable is **optional**. Please refer to Appendix F regarding how to discover and support these cables.

5.4.3.2.1 Discovering Passive Cables

The **USB PD** specification defines the Passive Cable VDO responses to the **Discover Identity** Command sent by the DFP to a **USB4**-compatible passive cable.

If the USB Signaling field [B2...0] in the Passive Cable VDO response is 011b (**USB4** Gen3) or 100b (**USB4** Gen4), the **USB4** discovery process is complete and **USB4** operation up to as high as 80 Gbps is supported. In Chapter 3 of this specification, these cable assemblies are those with the following cable references: CC4G3-3, CC4G3-5 and CC4G3-5E indicated in Table 3-1.

If the USB Signaling field [B2...0] in the Passive Cable VDO response is 000b (**USB 2.0** only), the **USB4** discovery process will be exited and the DFP **shall** proceed to establish a functional connection to its UFP port partner following traditional **USB 2.0** process.

If the USB Signaling field [B2...0] in the Passive Cable VDO response is either 010b (**USB 3.2** Gen2) or 001b (**USB 3.2** Gen1), the **USB4** discovery process is complete if the DFP is limited to **USB4** Gen2. In Chapter 3 of this specification, these cable assemblies are those with the following cable references: CC3G2-3, CC3G2-5, CC3G1-3, and CC3G1-3 indicated in Table 3-1. Note that **USB 3.2** Gen1 cables, while not tested and certified to be used for **USB 3.2** Gen2 operation, are expected to work for **USB4** Gen2 operation.

If the USB Signaling field [B2...0] in the Passive Cable VDO response is 010b (**USB 3.2** Gen2) but the DFP is capable of **USB4** Gen3 or higher operation, then the DFP **shall** use the **USB PD Alternate Mode** process to determine if the cable also can be identified as a **TBT3** Gen3 cable. Refer to Section 5.4.3.2.3 for **TBT3** cable discovery process. If the Cable Speed field of the Discover Modes VDO response is set to 011b, then the **USB4** discovery process is complete and **USB4** operation up to as high as Gen4 is supported using the **TBT3** passive cable (see Table F-11).

5.4.3.2.2 Discovering Active Cables

The **USB PD** specification defines the Active Cable VDO responses to the **Discover Identity** Command sent by the DFP to a **USB4**-compatible active cable.

If the USB Signaling Support field [B2...0] in the Active Cable VDO 1 response is 011b (**USB4** Gen3), the **USB4** discovery process is complete and **USB4** operation up to as high as Gen3 is supported. If the USB Signaling Support field [B2...0] in the Active Cable VDO 1 response is 100b (**USB4** Gen4), the **USB4** discovery process is complete and **USB4** operation up to as high as Gen4 is supported.

Optionally, discovery and use of existing **TBT3** active cables that indicate support for rounded data rate operation is allowed if the active cable isn't explicitly identified as **USB4**-compatible. If the USB Highest Speed field [B2...0] in the Active Cable VDO 1 response is 010b (**USB 3.2** Gen2) but the DFP is capable of **USB4** Gen3 or higher operation, then the DFP shall use the **USB PD Alternate Mode** process to determine if the cable also can be identified as a **TBT3** active cable with rounded data rate operation support. Refer to Section 5.4.3.2.3 for **TBT3** cable discovery process. If the TBT_Rounded_Support field of the Discover Modes VDO response is set to 01b, and the Cable Speed field of the Discover Modes VDO response is set to 011b, then the **USB4** discovery process is complete and **USB4** operation up to as high as Gen3 is supported using the **TBT3** active re-timer cable (see Table F-11).

Failure to identify that the attached active cable is **USB4**-compatible will result in exiting the **USB4** discovery process and reverting to following traditional **USB 3.2** and **USB 2.0** process.

5.4.3.2.3 Process for Discovering Thunderbolt 3 Cables

The **USB PD** specification defines the process for discovering **Alternate Mode**-enabled cables. The following summarizes this process specific to discovering **Thunderbolt 3** cables for purposes of determining **USB4**-compatibility. Prior to performing these steps, the **USB PD Discover Identity** process will have already been used to establish if the cable is passive or active.

The following steps are used for discovering **Thunderbolt 3** cables and the cable's capabilities using the **USB PD Alternate Mode** process.

1. DFP issues the Discover SVIDs command to the cable SOP'.
2. If the cable's Discover SVID response indicates 0x8087 (Intel/**TBT3**) as one of its SVIDs, then proceed to next step, otherwise the cable is not a **Thunderbolt 3** cable (see Section F.2.4).
3. DFP issues the Discover Modes command with its SVID set to 0x8087 to the cable's SOP'.
4. If this discovery is part of the **USB4**-compatible passive cable discovery process, from the cable's Discover Modes VDO responses (see Section F.2.6), extract the value in the Cable Speed field to complete the process. Additionally, if this cable is a **TBT3** LRD cable as indicated by the cable's Discover Modes VDO response (Bit 22 = 0 and Bit 25 = 1), then the DFP **may optionally** send a **USB PD Enter_USB** Message or **USB PD Enter Mode Command** to the cable.
5. If this discovery is part of the **USB4**-compatible active cable discovery process, from the cable's Discover Modes VDO responses (see Section F.2.6), extract the value in the TBT_Rounded_Support field and extract the value in the Cable Speed field to complete the process. [Note: discovery and use of **USB4**-compatible **TBT3** active cables is an **optional** feature that also would require use of **USB PD Enter Mode** command to enable the cable for **USB4** operation.].

5.4.3.3 USB4 Operational Entry

USB4 operational entry **shall** occur only after having established that the attached cable, if present, and the port partner are **USB4**-capable.

USB4 operational entry involves the use of the **USB PD Enter_USB** Message process between the DFP and both the attached **USB4**-compatible cable and the **USB4**-capable port partner – sending this message is order specific: SOP' first, SOP" second if present, and SOP third. Sending the **USB PD Enter_USB** Message to SOP' and SOP" is not needed for passive cables.

When using the **USB PD Enter_USB** Message for enabling **USB4** operation, the DFP **shall** indicate 010b (**USB4**) in the USB Mode field of the **USB PD Enter_USB** Data Object.

Cable Speed and Cable Type fields in the **USB PD Enter_USB** Data Object (EUDO) are determined as follows:

- For the Cable Speed field of the EUDO, the value **shall** be the same as returned in Cable Speed field of either a Passive Cable VDO or an Active Cable VDO 1 of USB Type-C Full Featured cable response.
- For the Cable Type field of the EUDO, the type **shall** be based on the USB Type-C Full Featured cable response, either a Passive Cable VDO (Cable Type = Passive) or an Active Cable VDO 2 (Cable Type = Active Re-timer, Active Re-driver or Optically Isolated based on the Physical connection and Active element fields).
- The only exception is for those cables that are identified using **Thunderbolt 3** cable discovery (see Section F.2.6 and the **USB 3.2** Gen 2 branch of Figure 5-1). In these cases, the Cable Speed and Cable Type fields of the EUDO **shall** be based on the Cable Speed, Active_Passive, Re-timer and Cable Type fields as returned by the **TBT3** Discover Mode VDO response.

The remaining fields of the EUDO **shall** be set appropriately by the DFP based on the capabilities of the DFP and attached cable.

5.4.4 USB4 Post-Entry Operation

5.4.4.1 During USB4 Operation

While in **USB4** operation, the following are allowed:

- The **USB PD** explicit power contract *may* be re-negotiated.
- Issue a **USB PD** Data_Reset command to change the mode of operation, e.g., from **USB4** to **USB 3.2** or an **Alternate Mode**.
- Enable **Alternate Modes** that do not reconfigure the port interface and operate in parallel with **USB4**.

5.4.4.2 Exiting USB4 Operation

The **USB PD** Data_Reset process causes USB data connections to be reset and **Alternate Modes** to be exited. This process does not change the existing power contract and data roles between the port partners. The Data_Reset process *shall* include the following steps:

- Issue a **USB PD** Data_Reset command to the SOP port partner to reset the data bus, reset the cable, and exit any **Alternate Modes** while preserving the power on VBUS.
- The **tUSB4Timeout** and **tAMTimeout** timers within the UFP *shall* be reset upon sending or receiving a **USB PD** Data_Reset command.
- Re-enter the **USB4** Discovery and Entry process (Section 5.4.3).

5.5 USB4 Hub Connection Requirements

USB4 hub behavior in managing its DFP connections has **USB4**-specific dependencies on the connection status and capabilities of its single UFP. Additionally, hubs have **USB4**-specific responsibilities for communicating the capabilities of the **USB4** host to downstream-connected **USB4** hubs. This section provides both requirements and guidance for **USB4** hub port connection behavior.

5.5.1 USB4 Hub Port Initial Connection Requirements

The following requirements apply to all hub ports.

1. Run the USB Type-C Connection process,
2. Establish an initial **USB PD** explicit contract,
3. If desired, use **USB PD PR_Swap** to establish the preferred power role, and
4. Use **USB PD DR_Swap** to establish data role to be consistent with the port's position in the USB tree if needed.

5.5.2 USB4 Hub UFP and Host Capabilities Discovery

The **USB4** hub DFPs capabilities are ultimately based on the capabilities seen at its UFP (once it has established a connection to the host). If the **USB4** hub's UFP is connected to an upstream **USB4** hub, then the capabilities over the connection between the two hubs *may not* initially represent the capabilities all the way back to the host.

The following summarizes the general principles regarding how UFP and host capabilities impact DFP connections.

- The downstream connection to a device or hub that is attached to the **USB4** hub's DFP is based on the capabilities of the hub.
- Once the **USB4** hub's UFP has established a connection, the hub's capabilities are limited to the capabilities of that UFP connection.
- When the **USB4** hub's UFP connection indicates that a host is not present, once the hub is notified that a host becomes present, the hub will limit its capabilities as needed to match those of the host.

- Any connections on the **USB4** hub's DFPs that existed prior to the host being present are adjusted to align with changes in capabilities if needed.
- Once a host becomes present in a **USB4** tree, all intermediary hubs will update their connections to align with the host capabilities as the host present status is propagated to the downstream connected **USB4** hubs.

The capabilities seen by the hub's UFP are based on one of the following:

- A **USB PD Enter_USB** message is received which indicates the USB operation (**USB4**, **USB 3.2** or **USB 2.0**) and associated characteristics supported (**USB4** PCIe-supported, **USB4 DP** supported, etc.) by the upstream port partner.
- A **USB PD Enter Mode** command is received to start a supported **Alternate Mode** (**Thunderbolt 3**, **DisplayPort**).
- No **USB PD Enter_USB** message is received within the **tUSB4Timeout**, or **USB PD Enter Mode** message is received within the **tAMETimeout** indicating only **USB 3.2** and **USB 2.0** are available.

If the **USB4** hub's UFP is connected to an upstream **USB4** hub, then the capabilities reported in the received **USB PD Enter_USB** message **shall** only be considered the host's capabilities if the Host Present bit is set. If the Host Present bit is reset, then the hub **shall** wait for a subsequent **USB PD Enter_USB** message to be received with the Host Present bit set. Once the Host Present bit is set, the capabilities as represented in the **USB PD Enter_USB** message can be used as the host's capabilities for the purpose of establishing final DFP connections.

If the **USB4** hub's UFP receives an **USB PD Enter_USB** message which indicates the USB operation as either **USB 3.2** or **USB 2.0**, the **USB4** hub **shall not** wait for the completion of the **tUSB4Timeout** before proceeding to establish its UFP and DFP connections following **USB 3.2** or **USB 2.0** hub requirements, respectively.

5.5.3 Hub DFP Connection Requirements

5.5.3.1 Speculative Connections

When a device is attached to the DFP of a **USB4** hub prior to the hub's UFP being connected and the host capabilities are known, the hub will speculatively connect to the attached device based on the following requirements.

- Use **USB PD Discover ID** and the **USB PD Alternate Mode** process to determine the full capabilities of the attached cable and device.
- Based on the discovered capabilities, establish the most capable connection based on the capabilities of the **USB4** hub's DFP in the following priority order:
 1. **USB4**
 2. **Thunderbolt 3 Alt Mode**
 3. **DP Alt Mode**
 4. **USB 3.2**
 5. **USB 2.0**
- Inhibit port status notifications and data paths upstream from the **USB4** hub's DFPs while waiting for the **USB4** hub's UFP connection to be established.

5.5.3.2 Operational Connections

Once the **USB4** hub's UFP connection is established and the host capabilities are determined (see Section 5.5.2), the hub **shall** evaluate each existing DFP connection based on the capabilities associated with the hub's UFP connection and perform one of the following actions.

- If the DFP connection properly aligns with the capabilities of the UFP connection, enable the status notifications and data path for that port.
- If the DFP connection does not properly align with the capabilities of the UFP connection, the DFP connection **shall** do one of the following:
 - If the UFP remains in **USB4** and the DFP connection is with a downstream **USB4** hub, send a revised **USB PD Enter_USB** message with the Host Present bit set.
 - If the UFP changed to **USB 3.2** or **USB 2.0** and the DFP connection is with a downstream **USB4** hub, the DFP **shall** be reset the connection using **USB PD Data_Reset** followed by sending a revised **USB PD Enter_USB** message indicating **USB 3.2** or **USB 2.0** and the Host Present bit set.
 - Otherwise, the DFP **shall** enter the **ErrorRecovery** state to reset the connection and establish a new connection that aligns with the hub's UFP capabilities.

5.5.4 Hub Ports Connection Behavior Flow Examples

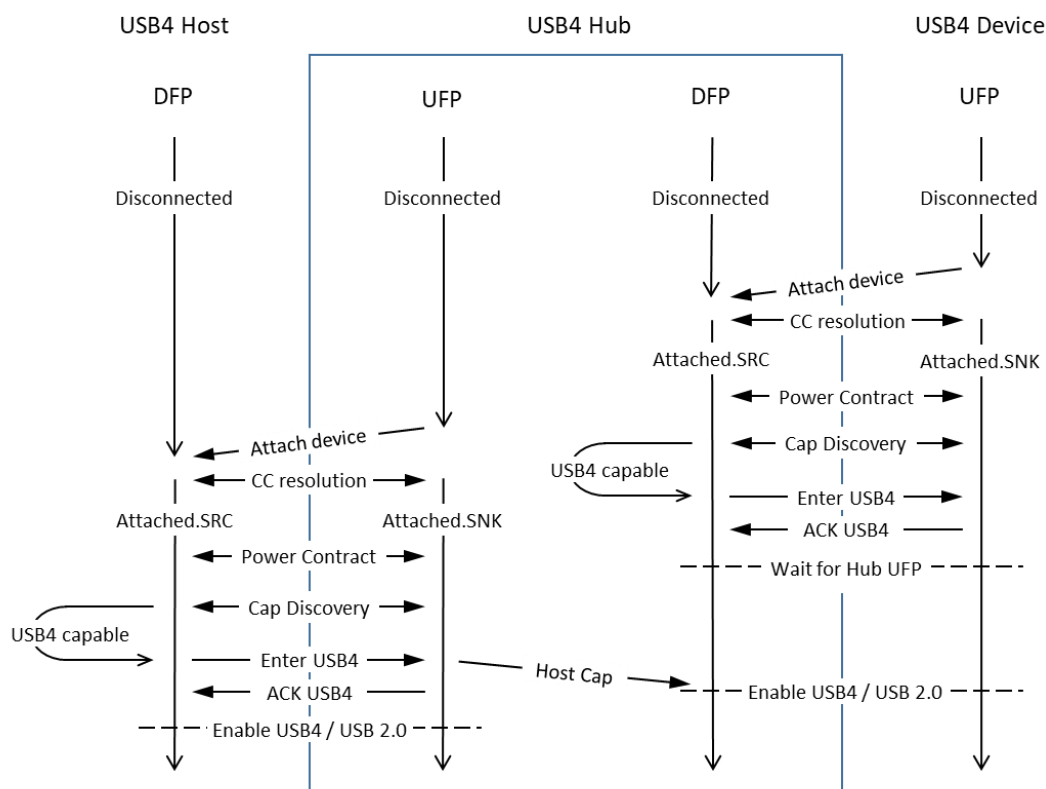
This section illustrates several example connection flows that assume that the hub's DFP connection is established prior to the hub's UFP connection. In these cases, the host capabilities are unknown at the time that the hub's DFP is connecting with the attached device. Given this, the initial connection established by the hub's DFP is speculatively based only on the hub and device's capabilities and **may** have to be revised once the host capabilities are known if there is a functional mismatch. When the hub's UFP connection is fully established prior to devices appearing on the hub's DFP, the connection can be established with full knowledge of the host's capabilities – flows associated with this relationship are not illustrated.

For the example flows in this section, the Source/Sink power roles remain as initially resolved by the CC connection state machine with no **PR_Swap** or **DR_Swap** activity.

All these example flows intend to minimize the total connection time for enabling the functionality of the device connected to the hub's DFP. This is accomplished by establishing the highest functional connection based on mutual capabilities between the hub and the device even as the hub's UFP capabilities are unknown or not ready for operation. If the speculatively established connection turns out to be valid once the hub's UFP capabilities are established, then the DFP's connection will be enabled as is. If the speculatively established connection turns out to be invalid, the DFP connection will be reset and a connection that aligns with the hub's UFP capabilities **shall** be established.

Figure 5-2 Illustrates a connection flow aligned across the combination of a **USB4** host, hub, and device. The expected result is the successful enabling of end-to-end **USB4** operation.

Figure 5-2 USB4 Hub with USB4 Host and Device Connection Flow Alignment



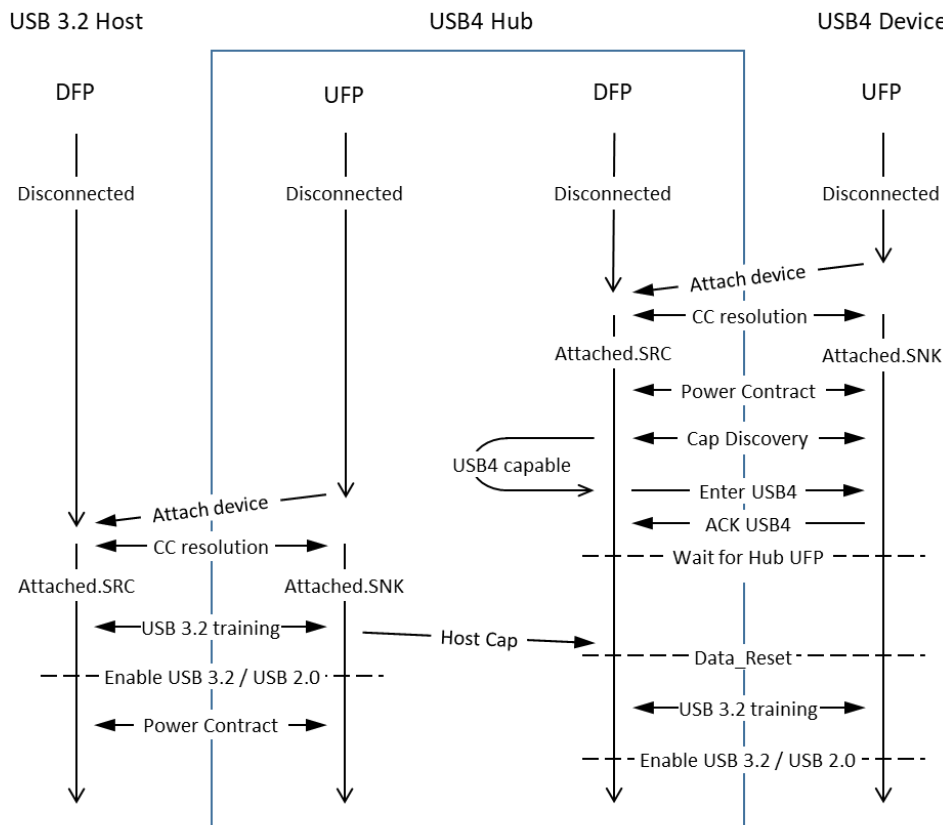
In the illustrated flow, Cap Discovery includes all **USB PD** message exchanges needed between the DFP and its UFP port partner to discover the UFP's USB capabilities along with **TBT3**-compatibility and **DP Alt Mode** capabilities. For the **USB4** hub's DFP, Cap Discovery is done on a speculative basis whenever it does not already know of the capabilities of the host that will eventually be connected via the hub's UFP.

Upon completion of Cap Discovery between the hub's DFP and its UFP port partner, the hub DFP will establish the highest functional connection and then wait for the hub UFP to complete its connection. Once the hub's UFP connection is established, the host capabilities available are used to determine what **should** be done to complete the hub DFP connection to the UFP port partner.

Host Cap is based on the resulting configuration (e.g., data bus protocol and speed) of the **USB4** Hub's UFP and the Host capabilities information received in the **USB PD Enter_USB** Message from its DFP port partner (see Section 5.5.2). The **USB4** hub uses Host Cap to set the available capabilities of the hub's DFPs.

Figure 5-3 illustrates a connection flow aligned across the combination of a **USB 3.2** host, a **USB4** hub and a **USB4** device.

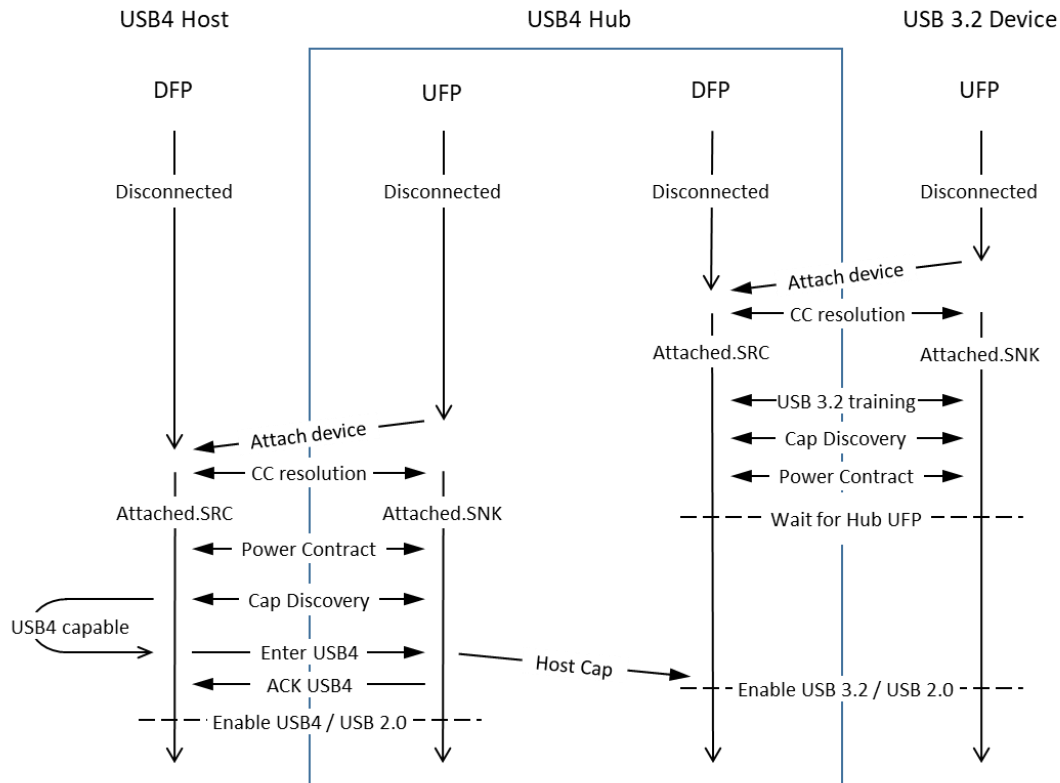
Figure 5-3 USB4 Hub with USB 3.2 Host and USB4 Device Connection Flow Alignment



In the flow above, once connected to a **USB 3.2** host, the Host Cap reflects that the hub can only support **USB 3.2** on its DFPs and the speculatively established **USB4** connection on the DFP is exited with the **USB4** hub now operating as a traditional **USB 3.2** hub.

Figure 5-4 illustrates a connection flow aligned across the combination of a **USB4** hub with a **USB4** host and **USB 3.2** device.

Figure 5-4 USB4 Hub with USB4 Host and USB 3.2 Device Connection Flow Alignment

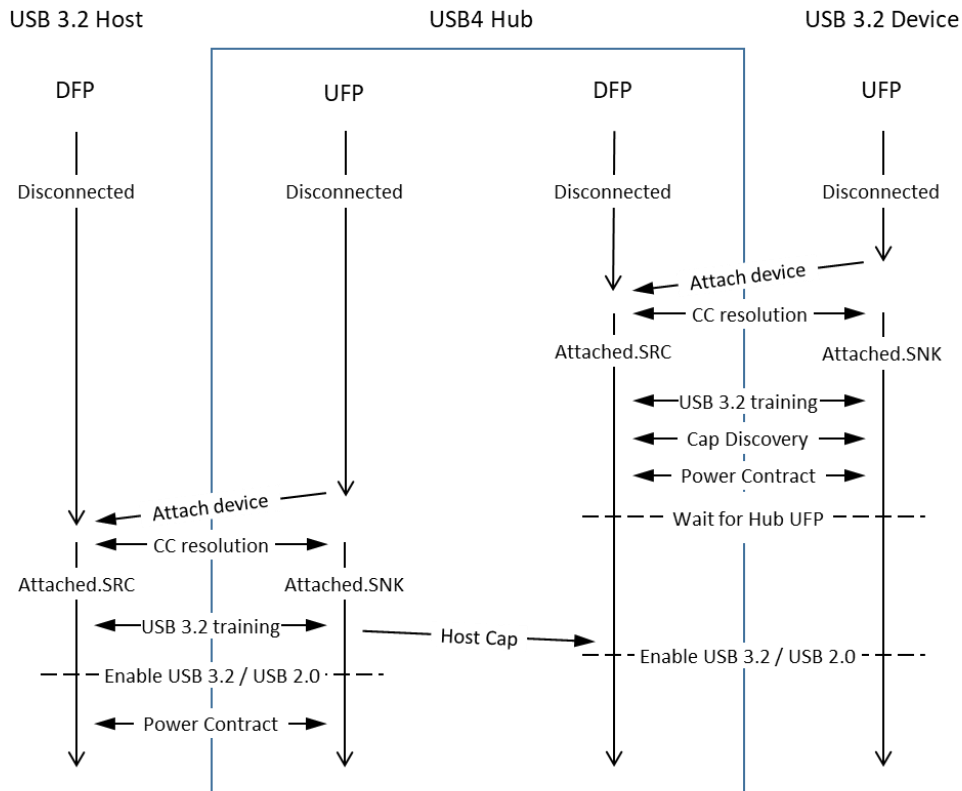


While **USB 3.2** devices won't necessarily respond to the Discover ID (SOP), the **USB4** hub's DFP will attempt to discover the capabilities of the attached device.

In the flow above, after the **USB4** connection of the hub's UFP is established, the DFP connection remains valid with the **USB 3.2** data path of the DFP being serviced by the **USB4** Enhanced SuperSpeed tunnel.

Figure 5-5 illustrates a connection flow aligned across the combination of a **USB4** hub with a **USB 3.2** host and device.

Figure 5-5 USB4 Hub with USB 3.2 Host and Device Connection Flow Alignment

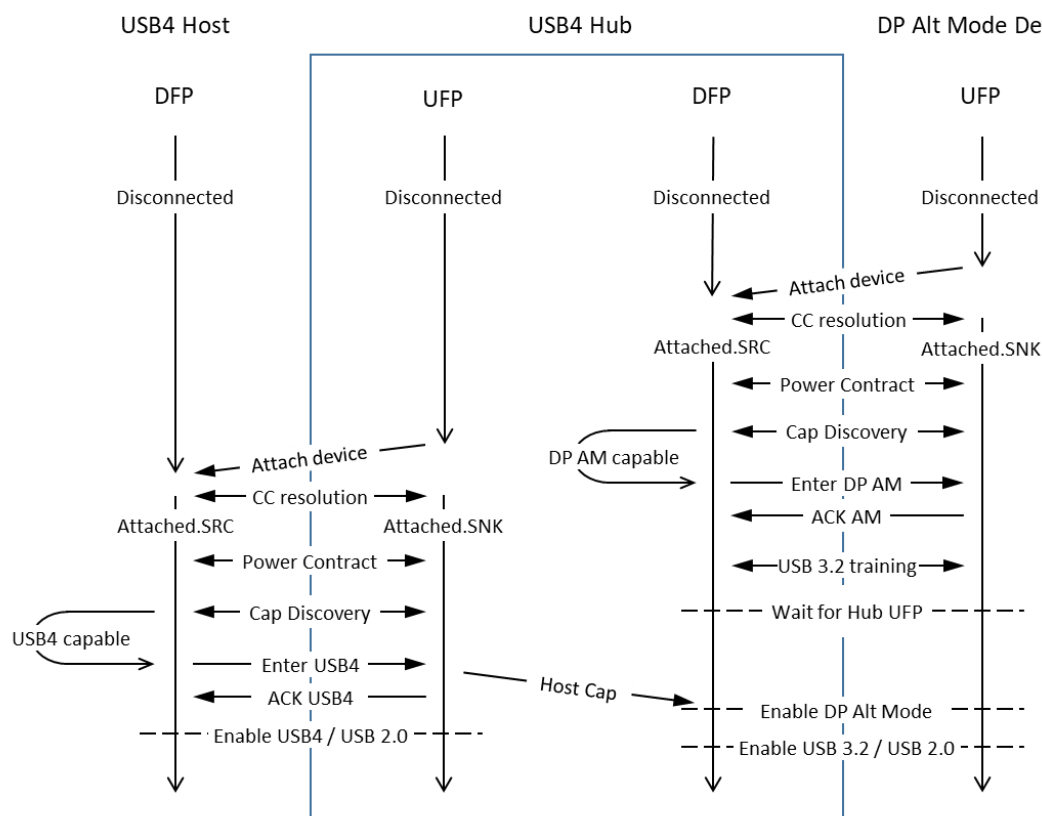


While **USB 3.2** devices won't necessarily respond to the Discover ID (SOP), the **USB4** hub's DFP will attempt to discover the capabilities of the attached device.

In the flow above, after the **USB 3.2** connection of the hub's UFP is established, the DFP connection remains valid with the **USB4** hub now operating as a traditional **USB 3.2** hub.

Figure 5-6 illustrates a connection flow aligned across the combination of a **USB4** host, **USB4** hub and a **DP Alt Mode** device (operating in Multi-function mode). In this case, the expected result is the enabling of the **DP Alt Mode** as bridged from **USB4 DisplayPort** tunneling.

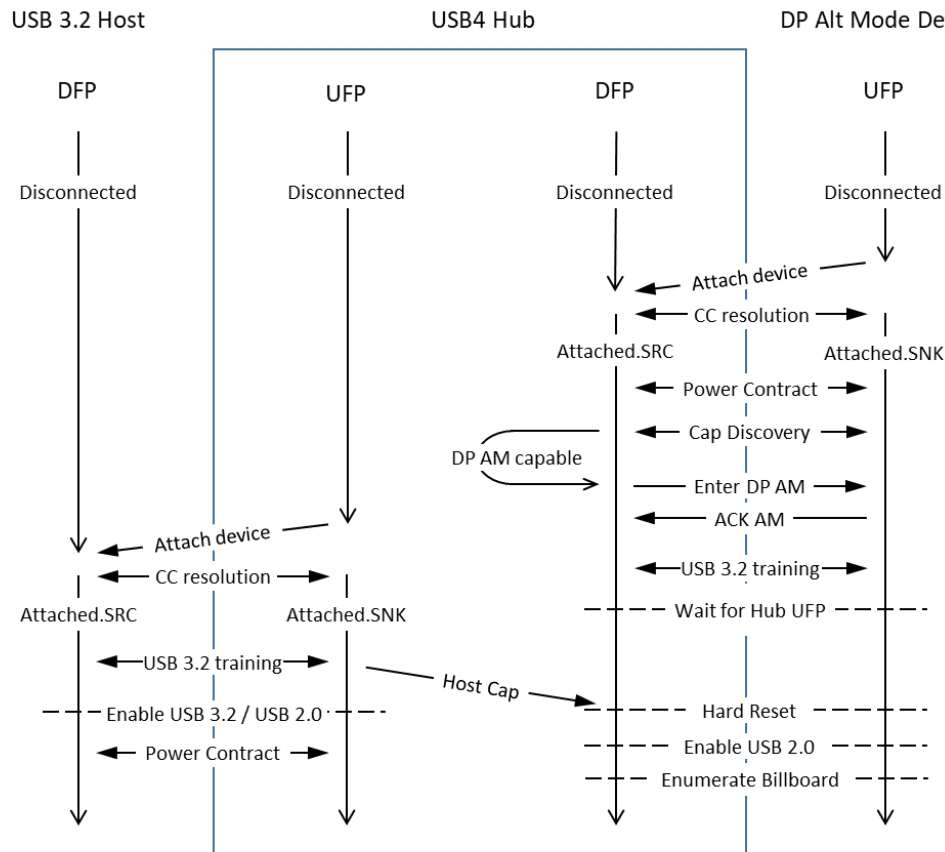
Figure 5-6 USB4 Hub with USB4 Host and DP Alt Mode Device Connection Flow Alignment



In the flow above, after the **USB4** connection of the hub's UFP is established, the DFP connection remains valid with the **DisplayPort** and **USB 3.2** data paths of the DFP being serviced by the **USB4 DisplayPort** and Enhanced SuperSpeed tunnels.

Figure 5-7 illustrates a connection flow aligned across the combination of a **USB 3.2** host, a **USB4** hub and a **DP Alt Mode** device (operating in Multi-function mode). The expected result in this case is that the **DP Alt Mode** will not be enabled, and a **USB Billboard Device Class** exposed by the device since the host doesn't support **USB4**.

Figure 5-7 USB4 Hub with USB 3.2 Host and DP Alt Mode Device Connection Flow Alignment



In the flow above, after the **USB 3.2** connection of the hub's UFP is established, the DFP connection is no longer valid with the **USB4** hub now operating as a traditional **USB 3.2** hub. The hub's DFP would then reset the port which will lead to a **USB 2.0** connection and the exposure of the **USB Billboard Device Class**.

5.5.5 Connecting to Downstream USB4 Hubs

When a **USB4** hub is attached on its DFP to the UFP of another **USB4** hub, the **USB4** hub **shall** use the Host Present bit of the **USB PD Enter_USB** message to inform the downstream hub if the **USB4** capabilities listed in the message reflects the host's capabilities or not. If an initial connection is made with the downstream hub with the Host Present bit reset in the **USB PD Enter_USB** message, the **USB4** hub **shall** subsequently send a revised **USB PD Enter_USB** message with the Host Present bit set after its UFP has been fully established (see Section 5.5.2).

5.5.6 Fallback Functional Requirements for USB4 Hubs

When a **USB4** hub is attached on its UFP to a non-**USB4** DFP, the **USB4** hub **shall** seamlessly fall back to functioning as and meeting the requirements for a **USB 3.2** hub.

5.6 USB4 Device Connection Requirements

5.6.1 Fallback Mapping of USB4 Peripheral Functions of USB Device Class Types

USB4 peripheral devices provide functions based on data transferred over one or more of the protocol tunnels of **USB4: Enhanced SuperSpeed USB, DisplayPort** and **PCI Express**. For all peripheral functions that use the Enhanced SuperSpeed USB protocol tunnel, the mapping of those functions to USB device class specifications is clear but for those based on the other tunneled protocols, this mapping doesn't apply since those functions don't rely on USB device class drivers to operate.

Each function of a **USB4** device **shall** be mapped to an equivalent USB device class when possible. **USB4** devices that contain mapped USB device class functions **shall** support operation at **USB 3.2** or **USB 2.0** when connected to non-**USB4** hosts. This requirement is exempt for those functions that rely on **DisplayPort** or **PCIe** tunnels for **USB4** data transfer that don't reasonably map to an existing USB device class, e.g., a PCIe graphics adapter.

The performance of the function when mapped to a lower speed connection is expected to scale appropriately while still providing the functional equivalent of the primary capabilities of the peripheral function.

Table 5-2 lists USB Device Class types and the mapping requirements for **USB4** device peripheral functions as it relates to fallback when operating over **USB 3.2** or **USB 2.0**.

Table 5-2 Fallback Mapping USB4 Functions to USB Device Class Types

Device Class Category	USB4 Peripheral Function Mapping	Notes
Audio	<i>Required</i>	
Video	<i>Required</i>	
Mass Storage	<i>Required</i>	
Comms/Networking	<i>Required</i>	
Printer	<i>Required</i>	
HID	<i>Required</i>	Only required when an equivalent HID subclass or report usage is defined.
Media Transfer Protocol	<i>Required</i>	
Smart Card	<i>Required</i>	
Still Image Capture	<i>Required</i>	
Monitor Device	<i>Required</i>	Only required in conjunction with providing associated display applications.

For all **USB4** peripheral functions based on **DisplayPort** and **PCIe** protocol tunneling that do not map to USB device class equivalents when operating over **USB 3.2** or **USB 2.0**, an appropriate **USB Billboard Device Class** **shall** be exposed to enable user notifications by the operating system of the host platform.

5.7 Parameter Values

5.7.1 Timing Parameters

Table 5-3 provides the timeout requirement for a device that supports **USB4** to enable a **USB Billboard Device Class** interface when the device cannot connect as a **USB4** device during the discovery and entry process (Section 5.4).

Table 5-3 USB Billboard Device Class Availability Following **USB4** Device Entry Failure

	Maximum	Description
<i>tUSB4Timeout</i>	1000 ms	The time between (1) a Sink attach or (2) the data connection is reestablished in the USB PD Data Reset process until the USB Billboard Device Class interface is exposed when USB4 device entry is not successful.

6 Active Cables

Active cables **shall** minimally support **USB 3.2** Gen 2x2. **USB4** active cables **shall** support **USB4** in addition to all **USB 3.2** rates. Active cables **shall** support **USB PD** eMarkers and **may** support **Alternate Modes** and advertise them as defined in Section 6.1.1.

All **USB4** active cables **shall** be interoperable with **Thunderbolt 3** as defined in the **USB4** Specification (Chapter 13) and this specification (Section 6.1.1 and Appendices E and F).

Active cables supporting lengths up to 5 meters **shall** work in both directions and plug orientations and **should** function like passive cables from the user's perspective.

Hybrid Optical active cables use copper for power and **USB 2.0** data and optical for high-speed data signaling (TX/RX) and are functionally the same as copper-based active cables and **shall** function like passive cables from the user's perspective.

Optically Isolated Active Cables (OIACs) are a special case of active cables with unique requirements and limitations that are specific to OIACs. OIACs can support longer lengths up to 50 meters and provide electrical isolation between the two ends of the cable. OIACs are targeted for Industrial, Machine Vision, Remote Sensor, Pro Video, and Medical applications. OIACs do not 'just work' like normal cables. Further details on the requirements and limitations for the OIAC are in Section 6.3.

6.1 General Specifications for All Active Cables

6.1.1 Discovering Active Cable Characteristics

The **USB PD** Discover_Identity Command is used to discover the characteristics of the active cable. This command **shall** only be sent to SOP'. All active cables **shall** respond to the Discover_Identity Command with Active Cable VDOs that returns information about the cable. Note the active cable **should** respond with **USB PD** Revision 3 but **may** respond with **USB PD** Revision 2 following the **USB PD** Interoperability rules.

Table 6-1 summarizes the **USB4** cables regarding key identity values that will be returned to **USB PD** Revision 3 Discover_Identity commands.

Table 6-1 *USB4* Cable Identity Summary

<i>USB4</i> Cable	Function				SOP' Configuration (<i>USB PD</i> Revision 3)					
					ID Header VDO	Passive Cable VDO	Active Cable VDO 1	Active Cable VDO 2		
	USB2	USB3	<i>TBT3</i>	DP	Cable Plug Passive/Active B29...27	Cable Termination Type B12...11	Cable Termination Type B12...11	Physical connection B10	Active element B9	Optical Isolated Active Cable B2
Passive ¹	Yes	Yes	Yes	Yes	011b	00b/01b	n/a	n/a	n/a	n/a
<i>TBT3</i> Linear Re-driver ¹	Yes	Yes	Yes	Opt.	011b	01b ²	n/a	n/a	n/a	n/a
Linear Re-driver (Gen4) ¹	Yes	Yes	Yes	Opt.	100b ⁴	n/a	10b/11b	0b	0b	0b
					011b ⁴	01b ²	n/a	n/a	n/a	n/a
Re-timer ¹	Yes	Yes	Yes	Opt.	100b	n/a	10b/11b	0b	1b	0b
Hybrid Optical 1, 3	Yes	Yes	Yes	Opt.	100b	n/a	11b	1b	0b/1b	0b
Optically Isolated	No	Yes	Opt.	No	100b	n/a	11b	1b	0b/1b	1b
Notes: 1. <i>USB4</i> cables, except for Optically-Isolated cables, are required to support <i>Thunderbolt 3</i> compatibility. The <i>TBT3</i> -specific identity requirements are defined in Appendix F. 2. The <i>TBT3</i> Linear Re-driver active cable represents as only a Passive Cable that is discovered per Figure 5-1. 3. A Hybrid Optical active cable is defined as using an intermediate optical transmission line for the high-speed signaling path (TX/RX) while retaining a copper-based solution for the rest of the defined interfaces. 4. The <i>USB4</i> Gen4 Linear Re-driver active cable identifies as either an Active Cable or Passive Cable (for backward compatibility) depending on the version number (B14...11) given in the <i>USB PD</i> Structured VDM Discover Identity message as follows: for B14...11 = 0101b, the cable identifies as Active, and for B14...11 = 0100b, the cable identifies as Passive.										

Note: As indicated in Table 6-1, *USB4* passive cables are required to support *Thunderbolt 3* compatibility. The implication of this requirement is that *USB4* Gen3 passive cables will properly respond to *TBT3* Passive Cable Discover Identity commands with the VDOs as defined in Table F-1 and Table F-3 of Section F.2.1 in order that they will be used in a *TBT3* connection at Gen3 speeds. *USB4* Gen2 cables **should not** implement *TBT3* Passive Cable Identity VDOs.

Table 6-2 Active Cable Features

Cable Type	Length	USB PD	V _{BUS}	V _{CONN} Wiring	CC	USB 2.0	USB 3.2 (All required)	USB4	Alternate Modes	SBU
USB 3.2 Active	< 5 m	SOP' Required (SOP" Optional)	3 A or 5 A	Same as passive cable	Same as passive cable	Same as passive cable	Gen 1x1 Gen 2x1 Gen 1x2 Gen 2x2	<i>N/A</i>	Optional	Note 2 Note 3
USB4 Active	< 5 m	SOP' Required (SOP" Optional)	3 A or 5 A	Same as passive cable	Same as passive cable	Same as passive cable	Gen 1x1 Gen 2x1 Gen 1x2 Gen 2x2	Gen2 Gen3 Gen4	TBT3 Note 1	Note 2 Note 4
USB4 Hybrid Active	< 5 m	SOP' Required (SOP" Optional)	3 A or 5 A	Same as passive cable	Same as passive cable	Same as passive cable	Gen 1x1 Gen 2x1 Gen 1x2 Gen 2x2	Gen2 Gen3 Gen4	TBT3 Note 1	Note 2 Note 4
Notes: 1. Thunderbolt 3 Alternate Mode support required as defined in Appendix F. 2. A passive connection in USB 3.2 mode is required. 3. Support for SBU is optional normative in Alternate Modes . 4. SBU support for USB4 can be either passive or active in the case of a linear re-driver-based active cable or active in the case of a re-timer-based active cable.										

All active cables, regardless of length, **shall** be compliant with this specification, the **USB 3.2** including Appendix E, and the **USB 3.2** Active Cable CTS.

6.1.2 Electrical Requirements

6.1.2.1 Shielding Effectiveness Requirement

All active cables **shall** meet the shielding effectiveness requirement defined in Section 3.7.7 and Figure 3-66.

6.1.2.2 Low Speed Signal Requirements

6.1.2.2.1 CC Channel Requirements

Active cables **shall** meet the Low-Speed Signal Requirements in Section 3.7.2.6.

6.1.2.2.2 SBU Requirements

The SBUs are a pair of end-to-end electrical connections in the cable. While **USB 3.2** does not use these connections, they are used by **Alternate Modes** such as **DisplayPort**. **USB4** also uses these connections but has renamed them as SBTX and SBRX. The electrical requirements are the same for both.

Active cable SBU wires **shall** meet the requirements defined in Table 6-3 and **shall** meet the crosstalk requirements both near-end and far-end between the low speed signals as defined in Section 3.7.2.6.

SBUs have no guaranteed performance when V_{CONN} is not provided to the cable. The Host or Device **shall not** provide any signal beyond what is defined in Table 6-3 when V_{CONN} has not been provided.

Table 6-3 Active Cable SBU Requirements

Name	Description	Min	Max	Units
zCable_SBU	Cable characteristic impedance on the SBU wires	32	53	Ω
tCableDelay_SBU	Cable propagation delay on the SBU wire		26	ns
rCable_SBU	DC resistance of SBU wires in the cable in USB		40	Ω
vCable_SBU	Cable voltage swing on SBU wires	-0.3	4.0	V
Insertion Loss ¹	Cable insertion Loss		5 @ 0.5 MHz 7 @ 1 MHz 12 @ 10 MHz 13 @ 25 MHz 15 @ 50 MHz 16 @ 100 MHz	dB
iCableSBU	Maximum end-to-end current	-25	25	mA

Note 1: Measurement referenced to 50 Ohms.

6.1.2.3 USB 2.0

Active cables **shall** meet the **USB 2.0** requirements defined in Section 3.7.2.6 and 3.7.2.7.

Note: Active Cables greater than 5 m report the number of hub hops consumed in the Active Cable VDOs.

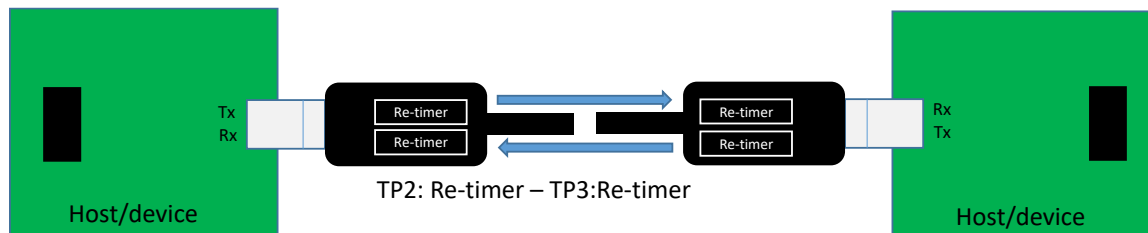
6.1.2.4 USB 3.2

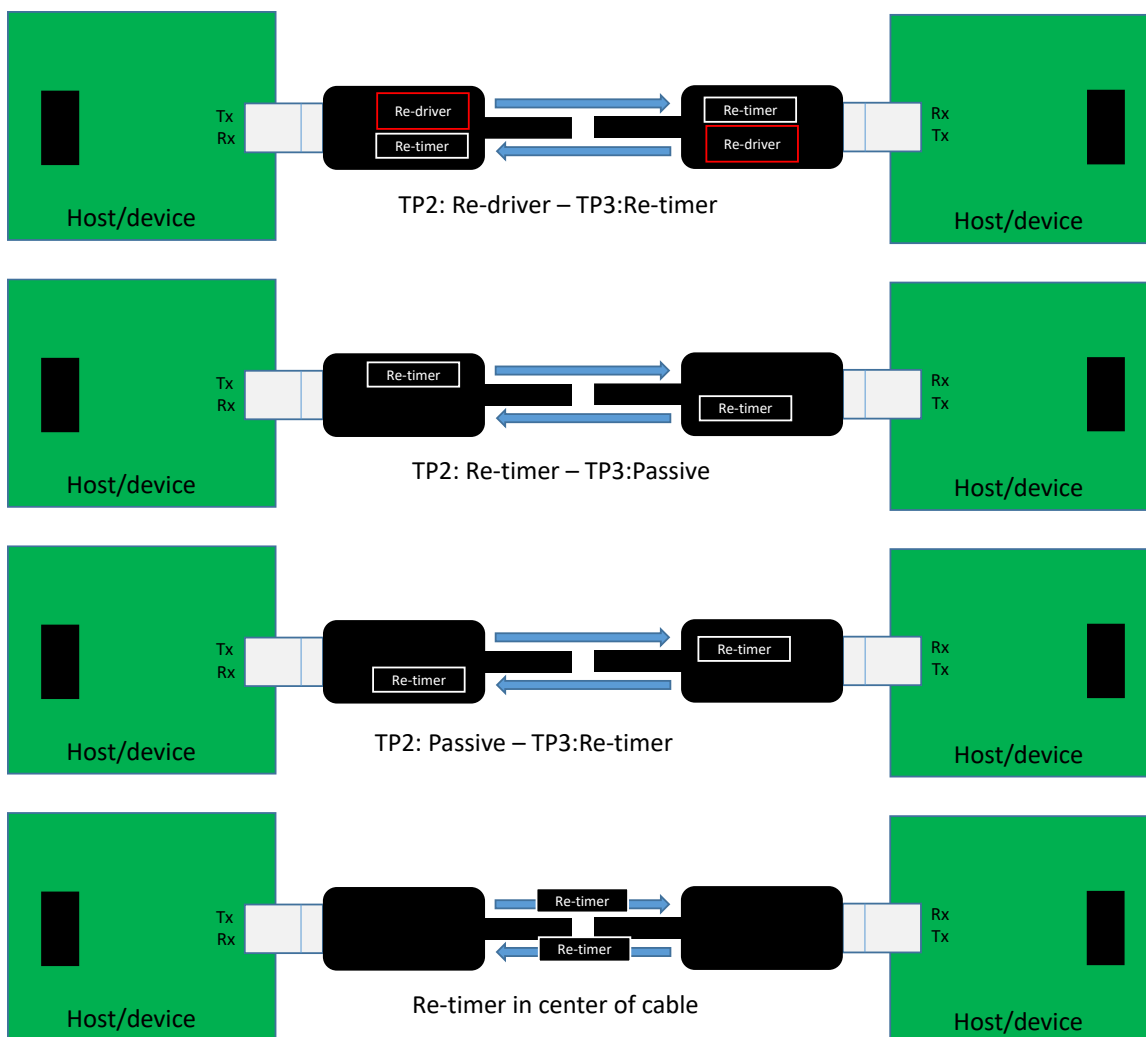
Active cables **shall** incorporate AC-coupling from the plug to repeater on both the **USB 3.2** TX and RX signals. Active cables **shall** provide a discharge path for discharging the AC-coupling capacitors in the cable on unplug per **USB 3.2**.

6.1.2.4.1 USB 3.2 Active Cable Architectures

Active cables **may** have ~~the different~~ combinations of re-timers and re-drivers ~~as~~. ~~Some active cable examples are illustrated in Figure 6-1. Active cables without at least one re-timer are out of scope. Active cables without re-timers connected to TP3 Other design options are out of scope possible.~~ Active cables **shall** support the features defined in Table 6-2.

Figure 6-1 Active Cable Topologies





6.1.2.4.2 USB 3.2 Power-On and Rx.Detect

Active cables **shall** present a high impedance to ground of $Z_{RX-HIGH-IMP-DC-POS}$ when not powered. Active cables **shall** present a high impedance to ground of $Z_{RX-HIGH-IMP-DC-POS}$ at initial power-on. The active cable **shall** perform far-end receiver termination detection on both cable ends upon receiving V_{CONN} . Upon detecting a far-end low-impedance receiver termination (R_{RX-DC}), the active cable **shall** enable its low-impedance receiver termination (R_{RX-DC}) to mirror the presence of the Host/Device. The active cable **shall** perform far-end receiver termination detection for Repeaters per **USB 3.2** including in low power states U2/U3.

An active cable **shall** complete power-on and far-end receiver termination detection through the cable within $t_{FWD_RX_DETECT}$.

Table 6-4 Active Cable Power-on Requirements

Name	Minimum	Maximum	Units
$Z_{RX-HIGH-IMP-DC-POS}$	Per USB 3.2	Per USB 3.2	
R_{RX-DC}	Per USB 3.2	Per USB 3.2	
$t_{FWD_RX_DETECT}$		42 ¹	ms
Note 1: 84 ms – (2 · 12 ms + 18 ms) worst case.			

Active cables including OIACs **shall** reflect the receiver terminations across the cable to replicate the behavior of a passive cable.

6.1.2.4.3 USB 3.2 U0 Delay

All active cables **shall** meet the **USB 3.2** delay defined in Table 6-5.

Table 6-5 Maximum USB 3.2 U0 Delay

USB 3.2 Gen	Cable Maximum U0 Delay	Notes
Gen1	3000 ns	Active cables with USB 3.2 Gen1 latency larger than 125 ns may not function correctly when used in conjunction with host, devices, and hubs that do not support USB 3.2 .
Gen2	3000 ns	Active cables with USB 3.2 Gen2 latency larger than 305 ns may not function correctly when used in conjunction with host, devices, and hubs that do not support USB 3.2 .

6.1.2.4.4 USB 3.2 U-State Power Requirements

Active cables **shall** meet the VCONN power requirements for **USB 3.2** operation in Table 6-6. These requirements are for the entire cable, not just a cable plug.

Table 6-6 USB 3.2 U-State Requirements

State	Maximum VCONN Power Consumption	Power Consumption Notes
U0	1.0 W single-lane 1.5 W dual-lane	Applies to POLLING.LFPS, TRAINING, and RECOVERY states.
U1	≤ U0 power	Forwarding LFPS is required
U2	≤ U1 power	Forwarding LFPS is required
U3	20 mW	Steady state power. eMarker in sleep. ¹
Rx.Detect	20 mW	Rx.Detect period may be lengthened when no USB 3.2 terminations have been detected. eMarker in sleep. ¹
eSS.Disabled	20 mW	USB 3.2 is disabled. eMarker in sleep. ¹

Note 1: Measured after the maximum **trRaWeaken** time after VCONN is applied with no USB PD traffic for at least 500 ms.

6.1.2.4.5 USB 3.2 U-State Exit Latency

Active cables **shall** meet the U-state exit latency defined in **USB 3.2** Appendix E.

6.1.2.4.6 USB 3.2 Signal Swing

An active cable transmitter only must drive 8.5 dB insertion loss at 5 GHz to the Host/Device controller receiver for **USB 3.2** Gen2, if the transmitter is located in the cable plug next to the receiving port.

A Host/Device controller transmitter must drive a total loss of 23 dB at 5 GHz to the far side for **USB 3.2 Gen2**. The difference in loss budget allows the active cable transmitter swing to be reduced. An active cable receiver can assume a larger receiver swing than in the Host/Device for the same reason.

Figure 6-2 defines the SuperSpeed electrical test points and is copied from the **USB 3.2** specification. Figure 6-3 indicates the test points and test equipment connections.

Active cables which include at least one re-timer **shall** meet the requirements in Sections 6.1.2.4.6.1 and 6.1.2.4.6.4.

Linear re-driven active cables (LRDs) **shall** meet the requirements in Table 6-15.

Figure 6-2 SuperSpeed USB Electrical Test Points

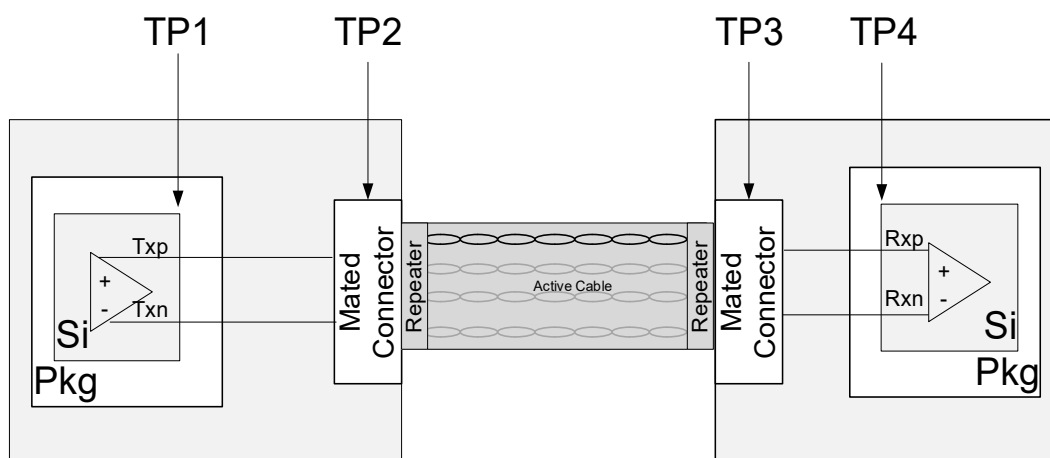
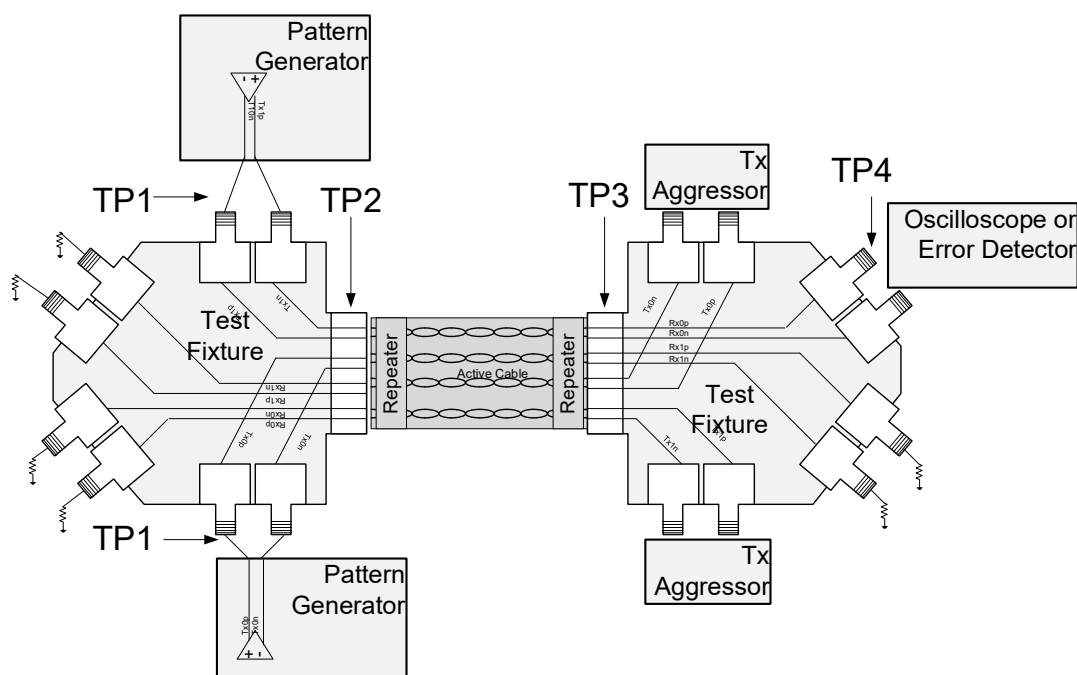


Figure 6-3 SuperSpeed USB Compliance Test Setup



6.1.2.4.6.1 TP1 – Active Cable Input Stressed Source

The active cable input stressed source is generated at TP1 per Table 6-7 for amplitude and per Table 6-8 for jitter. SSC **shall** be present in the stressed signal at TP1. Table 6-7 is a subset of the **USB 3.2** Table 6-18. Table 6-8 is a subset of **USB 3.2** Table 6-28. If any discrepancy exists between this specification and the **USB 3.2** specification, the **USB 3.2** specification **shall** take precedence.

The maximum swing with the maximum de-emphasis and pre-shoot **shall** be tested with the minimum loss compliance test board. The minimum swing with the minimum de-emphasis and pre-shoot **shall** be tested with the maximum loss compliance test board. The input jitter composition is the same for both the minimum and maximum swing stressed sources.

The active cable **shall** function over the range of parameter in Table 6-7 and **USB 3.2** Table 6-17.

Table 6-7 Active Cable USB 3.2 Stressed Source Swing, TP1

Symbol	Parameter	Gen1 (5.0 GT/s)	Gen2 (10 GT/s)	Units	Comments
V _{TX-DIFF-PP}	Differential p-p TX voltage swing	0.8 (min) 1.2 (max)	0.8 (min) 1.2 (max)	V	Nominal is 1 V p-p.
V _{TX-DE-RATIO}	TX de-emphasis	USB 3.2 Table 6-17	-3.1 ± 1.0	dB	Nominal is 3.5 dB for Gen1 operation. Gen2 transmitter equalization requirements are described in USB 3.2 Section 6.7.5.2.
V _{PRESHOOT}	TX Preshoot	USB 3.2 Table 6-17	2.2 ± 1.0	dB	Gen2 transmitter equalization requirements are described in USB 3.2 Section 6.7.5.2.

Table 6-8 Active Cable USB 3.2 Stressed Source Jitter, TP1

Symbol	Parameter	Gen1 (5.0 GT/s)	Gen2 (10 GT/s)	Units	Notes
f1	Tolerance corner	4.9	7.5	MHz	
J _{Rj}	Random Jitter	0.0121	0.0100	UI rms	1
J _{Rj,p-p}	Random Jitter peak-peak at 10 ⁻¹²	0.17	0.14	UI p-p	1,4
J _{Pj_500kHz}	Sinusoidal Jitter	2	4.76	UI p-p	1,2,3
J _{Pj_1MHz}	Sinusoidal Jitter	1	2.03	UI p-p	1,2,3
J _{Pj_2MHz}	Sinusoidal Jitter	0.5	0.87	UI p-p	1,2,3
J _{Pj_4MHz}	Sinusoidal Jitter	N/A	0.37	UI p-p	1,2,3
J _{Pj_f1}	Sinusoidal Jitter	0.2	0.17	UI p-p	1,2,3
J _{Pj_50MHz}	Sinusoidal Jitter	0.2	0.17	UI p-p	1,2,3
J _{Pj_100MHz}	Sinusoidal Jitter	N/A	0.17	UI p-p	1,2,3
Notes: 1. All parameters measured at TP1. The test point is shown in Figure 6-2 and Figure 6-3. 2. Due to time limitations at compliance testing, only a subset of frequencies can be tested. However, the RX is required to tolerate Pj at all frequencies between the compliance test points. 3. During the RX tolerance test, SSC is generated by test equipment and is always present. Each JPj source is then added and tested to the specification limit one at a time. 4. Random jitter is also present during the RX tolerance test. 5. The JTOL specs for Gen2 comprehend jitter peaking with re-timers in the system and has a 25 dB/decade slope.					

6.1.2.4.6.2 TP2 – Active Cable Input (Informative)

The values in Table 6-9 indicate the *informative* input signal swings at TP2 for an active cable. Table 6-9 is included to provide guidance beyond the *normative* requirements of Table 6-7 and Table 6-8. An active cable can be fully compliant with the *normative* requirements of this specification and not meet all the values in Table 6-9. Similarly, an active cable that meets all the values in Table 6-9, is not guaranteed to be in fully compliance with the *normative* part of this specification.

Table 6-9 Active Cable USB 3.2 Input Swing at TP2 (Informative)

Symbol	Parameter	Gen1 (5.0 GT/s)	Gen2 (10 GT/s)	Units	Comments
V _{TX-DIFF-PP}	Differential p-p TX voltage swing	250 (min) 1000 (max)	250 (min) 850 (max)	mV	Nominal is 550 mV p-p.
V _{TX-DE-RATIO}	TX de-emphasis	0 (min) 4.0 (max)	2.1 (min) 4.1 (max)	dB	There is no de-emphasis requirement for Gen1.
V _{PRESHOOT}	TX Preshoot	NA	1.2 (min) 3.2 (max)	dB	Applicable to USB 3.2 Gen2 operation only.

6.1.2.4.6.3 TP3 – Active Cable Output (Informative)

The values in Table 6-10 indicate the *informative* output signal swings at TP3 for an active cable. Table 6-10 is included to provide guidance beyond the *normative* requirements of Table 6-7 and Table 6-8. An active cable can be fully compliant with the *normative* requirements of this specification and not meet all the values

in Table 6-10. Similarly, an active cable that meets all the values in Table 6-10, is not guaranteed to be in full compliance with the **normative** part of this specification.

Table 6-10 Active Cable *USB 3.2* Output Swing at TP3 (Informative)

Symbol	Parameter	Gen1 (5.0 GT/s)	Gen2 (10 GT/s)	Units	Comments
$V_{TX-DIFF-PP}$	Differential RX peak-to-peak voltage	300 (min) 850 (max)	300 (min) 850 (max)	mV	Measured after the RX EQ function. Nominal is 0.5 V p-p.
$V_{TX-DE-RATIO-GEN1}$	TX de-emphasis	0 (min) 4.0 (max)	NA	dB	No pre-shoot allowed.
$V_{TX-DE-RATIO} + V_{PRESHOOT-GEN2}$	TX de-emphasis + TX Preshoot	NA	0 (min) 3.0 (max)	dB	Sum of the de-emphasis and pre-shoot. There is no de-emphasis and pre-shoot requirement.

6.1.2.4.6.4 TP4 – Active Cable Output

The active cable transmitter output is defined at TP4 for both high and low loss channels. The requirements for TP4 are defined in the ***USB 3.2*** specification Table 6-20. The input signal for the test ***shall*** be applied at TP1 as defined in the ***USB 3.2*** specification.

The low loss test board ***shall*** be used to test the maximum output swing. The maximum loss test board ***shall*** be used to test the minimum output swing. Jitter must be met with both test boards.

The active cable bit-error-rate ***shall*** be tested at TP4 and meet or exceed a BER of 10^{-12} . The error detector used ***shall*** have the ability to remove SKP ordered sets.

6.1.2.5 ***USB4***

This section describes the electrical requirements and compliance testing for ***USB4*** active cables. Compliance testing is defined to ensure interoperability in terms of data integrity and electrical specifications enabling the active cable to reliably receive an input signal and output a valid signal at its other end.

The ***USB4*** active cable types are:

1. Re-timer-based active cable (this section covers re-timer based cable for ***USB4*** only, ***USB 3.2*** re-timer cable is defined in Section 6.1.2.4),
2. Linear re-driver-based (LRD-based) active cable (***USB 3.2*** and ***USB4***), or
3. Linear optical active cable (the electrical LRD requirements in Section 6.1.2.5.4 will generally apply to a ***USB4*** OIAC).

6.1.2.5.1 Electrical Requirements Applying to All Active Cables

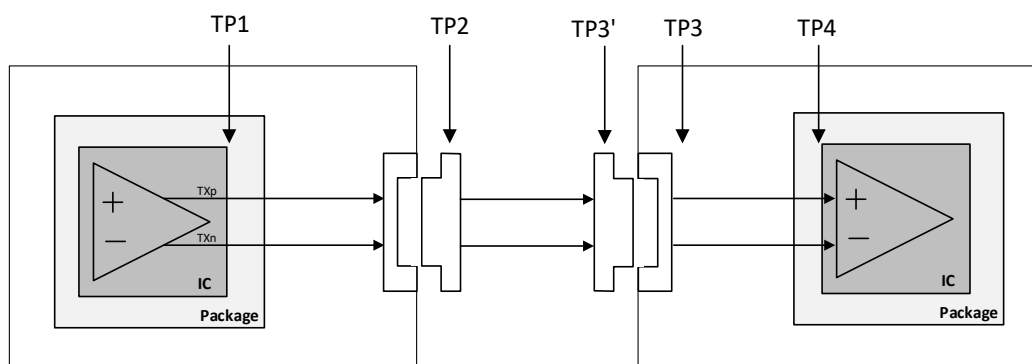
6.1.2.5.1.1 System Compliance Test Point Definition

All measurements ***shall*** be referenced to the following compliance points. Calibration ***shall*** be applied in cases where direct measurement is not feasible.

Table 6-11 Compliance Points Definition

Test Point	Description	Comments
TP1	Transmitter IC output	Not used for electrical testing
TP2	Transmitter port connector output	Measured at the plug side of the connector
TP3	Receiver port connector output	Measured at the receptacle side of the connector. All the measurements at this point shall be done while applying reference equalization function
TP3'	Receiver port connector input	Measured at the plug side of the connector
TP4	Receiver IC input	Not used for electrical testing

Figure 6-4 Compliance Points Definition



6.1.2.5.1.2 Compliance Receptacle Test Boards

The USB Type-C high speed test fixture **shall** be used to enable cable compliance testing. The fixture **shall** be comprised of a high-quality USB Type-C receptacle and a short PCB trace that **may** be connected to coaxial cable with SMA/SMP connector at its end. Because TP2 and TP3' reference points are located on the USB Type-C plug side of the connector, the loss and distortion of the receptacle fixture **shall** be calibrated such that all the measured values correspond to the standard reference points. The reference point TP3 is defined such that the insertion-loss from the connector pads to the compliance point is $0.5 \text{ dB} \pm 0.25 \text{ dB}$ at 5 GHz and $1 \text{ dB} \pm 0.25 \text{ dB}$ at 10 GHz. Extra loss and distortion elements **shall** be compensated by physical and/or mathematical means.

The target impedance of the fixture **shall** be 85Ω . AC coupling capacitors **shall** be placed on the receptacle test fixture following the Router Assembly requirements as specified in **USB4** specification and CTS.

6.1.2.5.1.3 AC Coupling Capacitor

Active cables **shall** include AC-coupling capacitance between 135 nF and 265 nF inside their plugs placed at the output transmit path and between 300 nF and 363 nF at the input receive path. Discharge resistors between $200 \text{ K}\Omega$ and $242 \text{ K}\Omega$ **shall** be placed at the input receive path. See Figure 3-3 in the **USB4** specification for a diagram of the AC-coupling capacitors and discharge resistors.

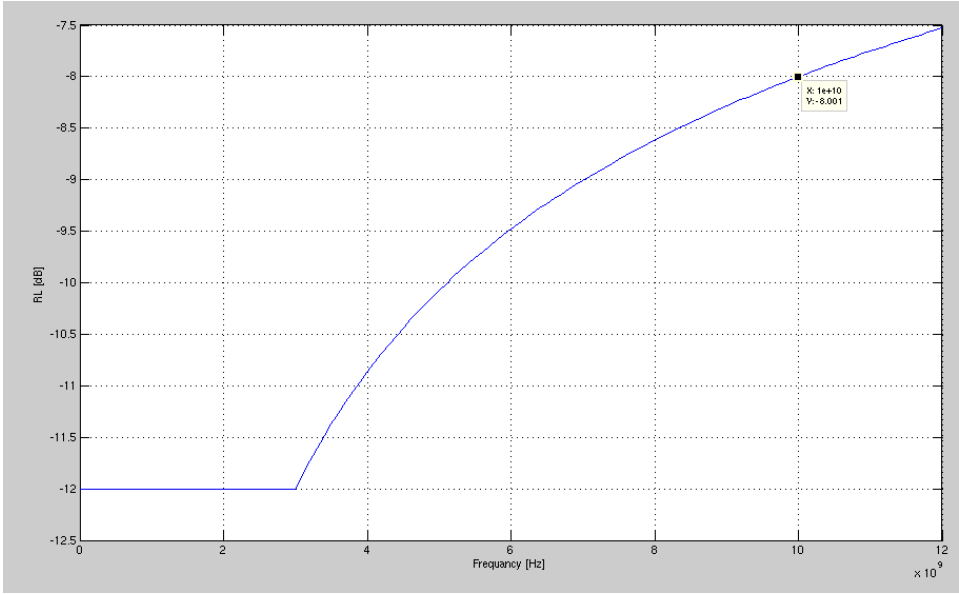
Active cable designs need to consider that a change of current consumption from VBUS as allowed by **USB Power Delivery** can add a considerable amount of common mode offset that **may not** be handled by the AC-coupling in this spec.

6.1.2.5.1.4 Differential Return-Loss Mask (*Informative*)

Re-timer and re-driver cable input and output return-loss measurements *shall* be referenced to a differential impedance of 85 Ω. When measured at TP3' and TP1 (respectively), the differential mode return loss recommended to not exceed the limits given in the following equation:

$$SDD22(f), SDD11(f) = \begin{cases} -12 & 0.05 < f_{GHz} \leq 3 \\ -7.5 + 7.5 \cdot \log_{10}\left(\frac{f_{GHz}}{12}\right) & 3 < f_{GHz} \leq 12 \end{cases}$$

Figure 6-5 RX Differential Return-Loss Mask



6.1.2.5.2 USB4 CL-State Power Requirements for Active Cables

Active cables *shall* support all the CL-States and meet the VCONN power requirements for **USB4** operation in Table 6-12. These requirements are for the entire cable, not just a cable plug.

Table 6-12 USB4 CL-State Requirements for Active Cables

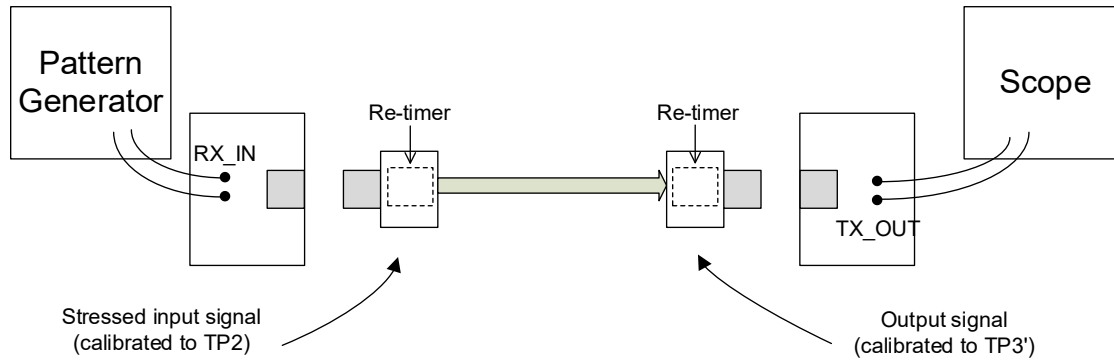
State	Maximum VCONN Power Consumption	Power Consumption Notes
CL0	1.5 W	Applies to training states and CL0.
CL0s	≤ CL0 power	
CL1	≤ CL0 power	
CL2	≤ CL1 power	
CLd	20 mW	Steady state power.
Note: OIAC CL-State Power requirements are in Section 6.3.6.5.2.		

6.1.2.5.3 Re-timer-based Active Cable Electrical Requirements

This section describes the electrical requirements and compliance testing for **USB4** re-timer-based active cables.

6.1.2.5.3.1 Re-timer-based Active Cable Compliance Test Setup

Figure 6-6 Re-timer-based Active Cable Compliance Test Setup



6.1.2.5.3.2 USB4 Gen2/3 Re-timer-based Active Cable

6.1.2.5.3.2.1 Output Equalization

A **USB4 Gen2/3** active cable **shall** implement tunable 3-tap finite-impulse-response (FIR) equalization at its output. The transmit equalization **shall** support 16 preset configurations with different de-emphasis and pre-shoot settings as specified in the **USB4** specification, and **shall** be measured at TP3'.

6.1.2.5.3.2 Re-timer-based Active Cable Compliance Test Setup

Figure -- Re-timer-based Active Cable Compliance Test Setup

6.1.2.5.3.2.2 Cable Compliance Testing

Table 6-13 defines the **USB4 Gen2/3** Active Cable specifications for Gen2 and Gen3 systems at TP3'. These parameters **shall** be measured at the Active Cable's output while applying a stressed signal at the input as specified in Table 6-14.

A **USB4 Gen2/3** active cable **shall** be tested by injecting several different periodic jitter components, one at a time. The test **shall** include sinusoidal jitter frequencies of 1 MHz, 2 MHz, 10 MHz, 50 MHz, and 100 MHz. In all cases, the incoming signal **shall** include SSC modulation on top of the sinusoidal jitter component at the range of 300ppm to -5300ppm. PRBS31 pattern **shall** be used for **USB4 Gen2/3** active cable compliance testing. However, calibration of the stressed signal source **may** be performed with a periodic pattern shorter than PRBS31. AC common-mode noise **shall** be added at the pattern generator output to ensure worst-case transmitter characteristics. The total common-mode noise **shall** be 100 mV peak-to-peak at TP2, where the added noise profile **shall** be sinewave at frequency not smaller than 400 MHz. All the specified jitter values **shall** be calibrated while applying the reference CDR defined in the **USB4 Gen2/3** specification.

A **USB4** active cable receiver **may** configure its Link Partner's TX equalizer during the Link establishment. The pattern generator **shall** support tunable 3-tap FIR at its output, which **may** be adjusted during the test by the receiver under test through out-of-band software channel.

Table 6-13 Re-timer-based *USB4 Gen2 and Gen3* Active Cable Output Specifications Applied for All Speeds (at TP3')

Name	Description	Min	Max	Units	Comments
CABLE_BER	End to End bit error rate		1E-12		See Note 1
AC_CM	Output AC Common Mode Voltage		100	mV pp	
LANE_TO_LANE_SKEW	Cable's Input-to-Output Skew between lanes		18	ns	
NRL	Noise Contributed by Integrated Return Loss				See 6.1.2.5.3.5
JTF_BW	Jitter tracking (forwarding) 3dB bandwidth from cable input to output		0.5	MHz	See Note 2
JTF_PEAKING	Jitter amplification from cable input to output		0.3	dB	Measured from 0 to 0.5 MHz. See Note 2
SSC_VARIATION	SSC output to input down-spread variation	-0.3	+0.3	dB	See Note 2
SSC_SLEW_RATE	SSC frequency slew rate (df/dt) during steady-state		1250	ppm/μs	See Note 3
INIT_FREQ_VARIATION	Non-modulated transmit frequency accuracy during the initial stages of the training period	-300	+300	ppm	See Notes 4 and 5
DELTA_FREQ_200ns	Transmit frequency variation over 200 ns measurement windows following the switching from local to recovered clock		1400	ppm	See Notes 4 and 5
DELTA_FREQ_1000ns	Transmit frequency variation over 1 μs measurement windows following the switching from local to recovered clock		2200	ppm	See Notes 4 and 5
V_OUTPUT_DC_AC_CONN	Instantaneous DC+AC voltages at the cable output side of the AC coupling capacitors	-0.5 (min1) -0.3 (min2)	+1.0	V	See Note 6

Name	Description	Min	Max	Units	Comments
TJ	Total Jitter		0.32	UI pp	See Notes 7 and 9
UDJ	Deterministic jitter that is uncorrelated to the transmitted data		0.13	UI pp	
UDJ_LF	Low Frequency Uncorrelated Deterministic Jitter		0.06	UI pp	See Note 10
DCD	Deterministic Jitter Associated by Duty-Cycle-Distortion		0.03	UI pp	
Y1	Eye inner height at TP3' (one-sided voltage opening of the differential signal)	200		mV	Measured for 1E6 UI. See Notes 8 and 9, and USB4 specification.
Y2	Eye outer height at TP3' (one-sided voltage opening of the differential signal)		650	mV	Measured for 1E6 UI. See Notes 8 and 9, and USB4 specification.

Notes:

1. The cable BER requirement refers to the raw data, without applying forward error correction nor pre -coding.
2. JTF_BW and JTF_PEAKING characterizes the corresponding input-to-output low-pass Jitter Transfer Function bandwidth and peaking. In addition, it is required that the cable will not change the SSC modulation depth by more than specified. For verifying that, the SSC down-spreading depth of the cable input and output **shall** be compared.
3. The SSC slew rate **shall** be extracted from the transmitted signal over measurement intervals of 0.5 μ s. The SSC slew-rate **shall** be extracted from the transmitter phase after applying a 2nd order low-pass filter with 3 dB point at 5 MHz. Steady-state clocking **shall** be applied from the point that SLOS training pattern is forwarded by the transmitter.
4. As shown in Figure 6-7, the initial transmit frequency is not modulated. The transmit frequency variation following the switching from local to recovered clock **shall** be measured over time intervals of 200 ns and 1 μ s.
5. Measurement **shall** be performed over the transmitted signal. The signal phase **shall** be extracted while applying 2nd order low-pass filter with 3 dB point at 5 MHz.
6. The absolute single-ended voltage seen by the receiver. This requirement applies to all link states and during power-on, and power-off. (min1, max) is measured with a 200 K Ω receiver load, and (min2, max) is measured with a 50 Ω receiver load. The ground offset between the cable output and UFP is not included in V_OUTPUT_DC_AC_CONN.
7. TJ is defined as the sum of all "deterministic" components plus 14.7 times the RJ RMS (the transmitter RJ RMS multiplier corresponds to the target BER with some margin on top).
8. The output voltage is differential.
9. Transmit jitter **shall** be measured while applying the reference CDR described in the **USB4** Specification. Note that the measured jitter includes residual SSC jitter passing the reference CDR.
10. UDJ_LF is the uncorrelated deterministic jitter measured after applying 2nd order Low-Pass-Filter with 3 dB cut-off at 0.5 MHz on the measured jitter. This filter needs to be applied on top of the reference CDR rejection function. The measurement **shall** be performed while applying input stress signal with periodic jitter component of 100 MHz.

Figure 6-7 Example of Transmitter Frequency Variation During Clock Switching

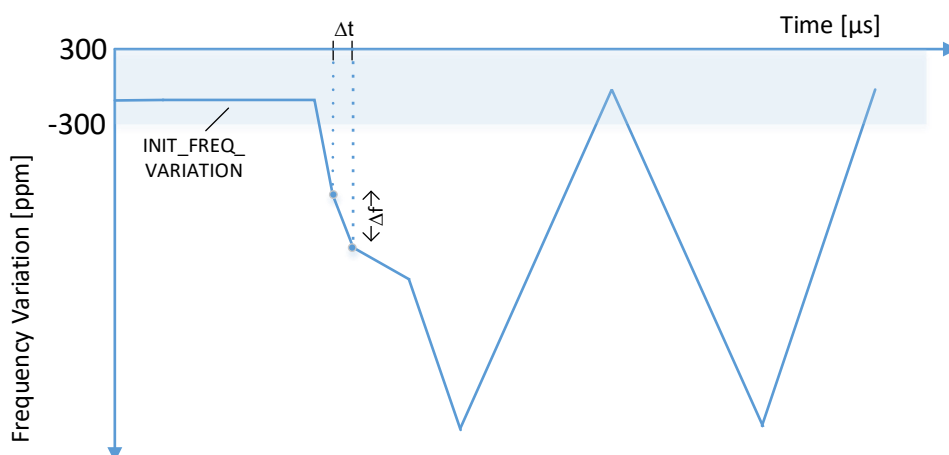
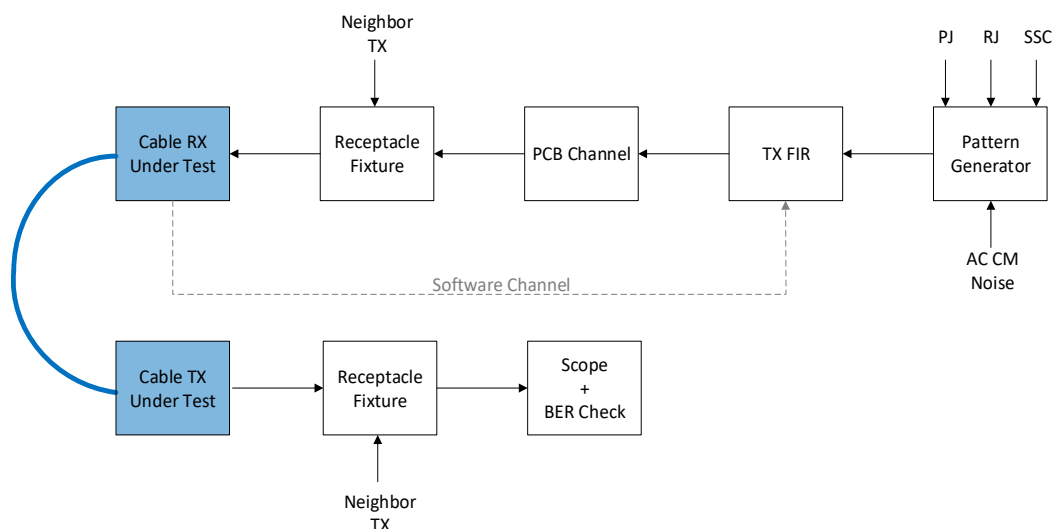


Table 6-14 Stressed Receiver Conditions for *USB4* Gen2 and Gen3 Cable Compliance Testing (at TP2)

<i>USB4</i> Gen	Inner eye Voltage [mV peak]	Data Dependent Jitter (DDJ) [UI peak-to-peak]	Random Jitter [UI peak-to-peak]	Periodic Jitter [UI peak-to-peak]	Total Jitter [UI peak-to-peak]
Gen2	140	0.12	0.14	0.17	0.43
Gen3	120	0.15	0.14	0.17	0.46

Figure 6-8 Active Cable Functional Test Setup



6.1.2.5.3.3 *USB4* Gen4 Re-timer-based Active Cable

This section will be added in a future ECN.

6.1.2.5.3.36.1.2.5.3.4 Cable Error-Bursts Testing

To facilitate proper FEC operation, an active cable ~~receiver~~ **shall** take steps to limit the probability that a burst of errors is restarted immediately after receiving one or more correct bits (see *USB4* specification). ~~The cable~~

~~receiver under test **shall** trigger on bit errors and **shall** capture error events that follow. The probability of Error-Burst Restart Events **should not** exceed 5E-7 (less than one restart event per 2 million bursts on average). The combination of an uncoded BER of 1E-12 and an Error-Burst Restart probability of 5E-7 supports an uncoded BER of 1E-19. In cases where the actual uncoded BER is less than 1E-21 at a certain ratio, the Error Burst Restart probability can be larger than 5E-7 at the same ratio, since the coded BER depends on the product of the two terms.~~

~~The test setup **shall** be initialized with the same configuration used for testing the uncoded BER with periodic jitter component of 100 MHz. As part of this setup, PRBS31 pattern is assumed and neither forward error-correction nor pre-coding are applied. After initialization, the periodic jitter magnitude **shall** be increased to the point where uncoded BER of 1E-8 is observed. The receiver under test **shall** trigger on bit error and **shall** capture error events that follow. An error event is defined as a mismatch between the received data and the reference PRBS31 pattern. At least 32 consecutive bits **shall** be examined for errors starting from the initial trigger. The probability for burst renewal **shall** be 5E-7 or less (i.e., one error burst restart per 2 million error captures).~~

The following is an example analysis:

No burst restart: ~~captured_data[31:0]=0000000000000000000000001111111111~~

Burst restart: ~~captured_data[31:0]=000000000000000000000000111001111111~~

~~where '1' represents a bit error and '0' represents a correct bit, as expected from "exclusive or" (XOR) operation between the received bits and the synchronized reference PRBS31 pattern. Captured_data[0] corresponds to the error event trigger.~~

~~It is recommended that the cable multi error-bursts probability be characterized using the compliance post processing tool (Informative).~~

Note: A burst of errors contains 1 or more consecutive bit errors.

~~6.1.2.5.3.4 Noise Contributed by Integrated Return Loss (NRL)~~

~~This section will be added in a future ECN.~~

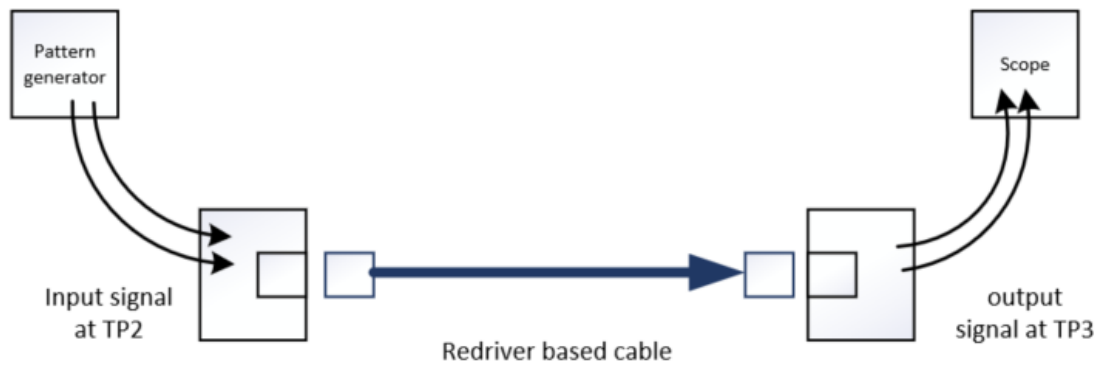
6.1.2.5.4 LRD-based Active Cable Electrical Requirements

Linear re-driver-based (LRD-based) active cables **shall** be tested as a complete component for compliance.

An LRD-based active cable is expected to receive a reference signal (referenced to TP2) defined in this specification and output a signal at the other end with electrical characterization that meets the requirements (referenced to TP3).

As shown in Figure 6-9, a compliant USB Type-C receptacle **shall** be connected to both ends of the active cable for injecting and measuring the signal to the corresponding TP2 and TP3 reference points. Details of the Compliance Receptacle and boards can be found in ~~Section 3.3.6 of the~~ **USB4** Specification.

Figure 6-9 Linear Re-driver-based Active Cable Compliance Setup



Note: The internal placement of the re-driver ICs is purposely not specified to allow full flexibility to the manufacture to develop various re-driver-based solutions.

6.1.2.5.4.1 General Implementation Notes

This specification was developed considering electrical interoperability with legacy systems, as the LRD-based Active Cables were added to the USB ecosystem in a late phase when a lot of devices were already in the field.

The USB Type-C interconnect ecosystem assumes the worst case 1 m/2 m/0.8 m passive cable is the worst-case connection (for **USB 3.2**, **USB4** Gen2 and **USB4** Gen3 respectively).

The intent is to align the LRD-based Active-Cable specifications to the existing passive cable specifications defined in Chapter 3 of this specification, such that the LRD-based active cable characteristics will be equal or better than those of the worst-case passive cable. The worst-case passive cable is defined in the **USB 3.2** and **USB4** Compliance Test Specification (CTS). This specification will define the electrical characteristics of the LRD-based cable that *shall* meet this requirement.

The LRD-based active cable specification assumes no change is needed to the existing TX/RX specification of the endpoint PHYs so that compatibility to existing certified **USB 3.2** and **USB4** devices is maintained.

For the Gen4 update to the **USB4** specification, LRD-based cable technology was considered, and all TX/RX implications are implicitly comprehended in the base specification.

Given this background, the following are assumptions regarding the LRD-based active cable implementation:

1. LRD-based active cable is assumed to have no clock mechanism in its ~~datapath~~data path (such as CDR).
2. LRD-based active cable is assumed to not have a dynamic equalization mechanism (as opposed to fixed equalization) to avoid masking the TxFFE training from the receiver.
- ~~2.3.~~ 3. For **USB 3.2** Gen2 and **USB4** Gen2/3, LRD-based active cable is assumed to not have a dynamic amplitude control (such as AGC)~~to avoid masking the TxFFE training from the receiver.~~
- ~~3.4.~~ 4. For **USB4** Gen4, it is a valid implementation choice to include AGC in the cable, however the time which this AGC can be enabled is limited to *tLRDSelfTune* which defined in this specification in Section 6.1.2.5.4.13.
- ~~4.5.~~ 5. LRD-based active cable is assumed to not use the training patterns to train itself, especially it is assumed to not block the output data during any phase of the training period.

~~5.6.~~ Receiver systems rely on the low-pass-filter nature of the cable and having an over-equalized cable (i.e., weak LPF characteristic) can lead to interoperability issues. Therefore, it is recommended that when developing an LRD-based active cable, the cable **should** be built and tuned in a way that will make it the most passive-cable-like as opposed to most equalized cable.

~~6.7.~~ During the development of the **USB4** Gen4 LRD cable we used a set of design assumptions that can be found in Appendix J.

6.1.2.5.4.2 **USB4** LRD-based Active Cable Compliance Testing

Table 6-15 defines the **USB4** linear re-driver-based active cable specifications for **USB 3.2** and for **USB4** Gen2, Gen3 and Gen4 systems at TP3.

These parameters **shall** be measured at the LRD-based active cable's output while applying a reference signal at the input as specified in Table 6-16.

An LRD-based active cable **shall** be tested by injecting several patterns, calibrated to TP2.

Table 6-15 Linear Re-driver-based Active Cable Output Parameters

Symbol	Description	Min	Max	Units	Conditions
<i>ILfitatDC</i>	Defining the ILfit mask for the cable response. Note: The main intention is to keep the cable with LPF characteristic similar to the passive cable.	<i>ILfitatNq</i> + 1.5 USB4 Gen2: -5	0	dB	See 6.1.2.5.4.4
<i>ILfitatf1</i>		<i>ILfitatNq</i>	<i>ILfitatDC</i>	dB	See 6.1.2.5.4.4
<i>ILfitatNq</i>		USB 3.2 : -6 USB4 Gen2: -12 USB4 Gen3: -7.5	<i>ILfitatDC</i> - 1.5	dB	See 6.1.2.5.4.4
<i>ILfitatf2</i>		<i>ILfitatNq</i> - 3	<i>ILfitatNq</i>	dB	See 6.1.2.5.4.4
<i>ILfitatf3</i>	Note: Not applicable to USB4 Gen4, see 6.1.2.5.4.11.	<i>ILfitatf2</i> - 4	<i>ILfitatf2</i>	dB	See 6.1.2.5.4.4
<i>ILfitatWB</i>	Max gain of the cable in the range of DC to f_N		0	dB	See 6.1.2.5.4.4
OUTPUT_NOISE	Standard deviation of the cable output noise. Combination of all noises beside the non-linearity noise.	See 6.1.2.5.4.5		mV	See 6.1.2.5.4.5
SIGMA_E	Standard deviation of the Non-linearity noise measured in the cable output		USB 3.2, USB4 Gen2/3: 15	mV	See 6.1.2.5.4.6
CABLE_SNR	Signal to non-linearity noise ratio in the cable output	USB4 Gen4: TBD		dB	See 6.1.2.5.4.6
Operating margin	Receiver margin evaluation	USB4 Gen3: 3 USB4 Gen4: 0		dB	See 6.1.2.5.4.9 normative only for USB4 Gen3 and Gen4
Eye mask	Eye mask in the cable output				See 6.1.2.5.4.10

Symbol	Description	Min	Max	Units	Conditions
CM_NOISE	Common mode noise		100	mV pp	See 6.1.2.5.4.12
IRL	Integrated Return Loss				See 6.1.2.5.4.7
IMR	Integrated multi-reflection (integration of ILD (Insertion loss deviation))				See 6.1.2.5.4.8

Table 6-16 Input Signal at TP2 for Compliance Testing

Pattern (defined in <i>USB4</i> Specification 8.3.2.1.1)	Swing (reference to TP2)	Added Jitter	TX Equalization	SSC	Minimum Measurement Time	Usage
PRBS15	800 mV p-p	No	No EQ	No	20 μs	Cable gain, non-linearity noise
PRBS15	<i>USB4</i> : 1300 mV p-p <i>USB 3.2</i> : 1200 mV p-p	No	No EQ	No	20 μs	Non-linearity noise
PRTS7	800 mV p-p	No	No EQ	No	20 μs	Cable gain, non-linearity noise (Gen4)
PRTS7	1000 mV p-p	No	No EQ	No	20 μs	Non-linearity noise (Gen4)
SQ512	300 mV p-p	No	No EQ	No	20 μs	Output noise
PRBS31	As defined in <i>USB4</i> Specification and CTS					Eye mask at TP3

6.1.2.5.4.3 Measurement Methods

The compliance testing of an LRD-based active cable to this specification will be done based on measurements from both time and frequency domains.

For all time domain specification items, the measured LRD-based active cable parameters will be compared to the worst-case passive cable supported in each technology (with nominal cable length of 1 m for *USB 3.2*, 2 m for *USB4* Gen2 and 0.8 m for *USB4* Gen3) measured in the exact same setup to reduce testing complexity. The worst-case passive cable is defined in the active cable CTS.

For *USB4* Gen4, the measurements will be against absolute values and not referenced to the worst-case passive cable.

More details on the measurement methods can be found in the active cable CTS.

6.1.2.5.4.4 Cable ILfit Mask (DC/f₁/Nq/f₂/f₃/WB) for *USB 3.2* and *USB4* Gen2/Gen3

Based on the pulse response extraction from the time domain measurements, ($h(n)$) Fourier Transform is used to extract the impulse frequency response $H_{impulse}(f)$. For the specification parameters calculation, a fitted version of $H_{impulse}(f)$ **shall** be used. The pulse extraction and fitting methods are detailed in the active cable CTS and in Appendix G of this document.

$$ILfitatDC = 20 \cdot \log_{10}(ILfit(DC))$$

$$ILfitatf_1 = 20 \cdot \log_{10}(ILfit(f_1))$$

$$IL_{fitatf_N} = 20 \cdot \log_{10}(IL_{fit}(f_N))$$

$$IL_{fitatf_2} = 20 \cdot \log_{10}(IL_{fit}(f_2))$$

$$IL_{fitatf_3} = 20 \cdot \log_{10}(IL_{fit}(f_3))$$

$$IL_{fitatWB} = \max(20 \cdot \log_{10}(IL_{fit}(f_0))), \text{ where } f_0 \text{ is in the range of DC to } f_N$$

$$DC = 100 \text{ MHz}$$

$$f_1 = f_N \cdot 0.7$$

$$f_2 = f_N \cdot 1.25$$

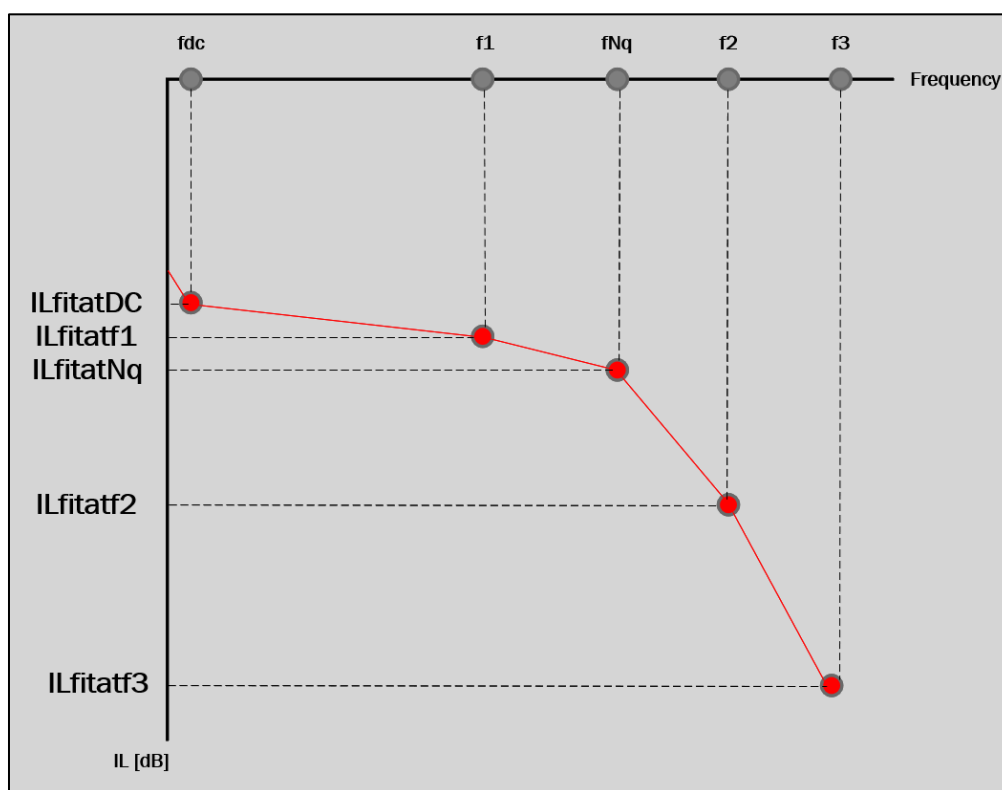
$$f_3 = f_N \cdot 1.5$$

$$f_N \text{ for USB3.2: } 5 \text{ GHz}$$

$$f_N \text{ for USB4 Gen2: } 5 \text{ GHz}$$

$$f_N \text{ for USB4 Gen3: } 10 \text{ GHz}$$

Figure 6-10 Gain Parameters Specified for the Linear Re-driver Active Cable



6.1.2.5.4.5 OUTPUT_NOISE (σ_n)

σ_n is the standard deviation of the uncorrelated additive noise added to the output signal of the LRD-based active cable.

To achieve an accurate measurement, the calculation will be done based on a low frequency signal (SQ512 pattern) applied to the cable input, with 0.3 Vpp amplitude.

Since the noise calculation is referred to the receiver input, a 2nd order Butterworth LPF filter with -2 dB @ Nq **shall** be applied on the captured wave to account for the receiver BW and device side platform.

The limit of OUTPUT_NOISE is defined as function of the IL at Nyquist frequency. This allows a degree of freedom to the cable developer to trade between the cable's gain and noise.

The limit is defined as:

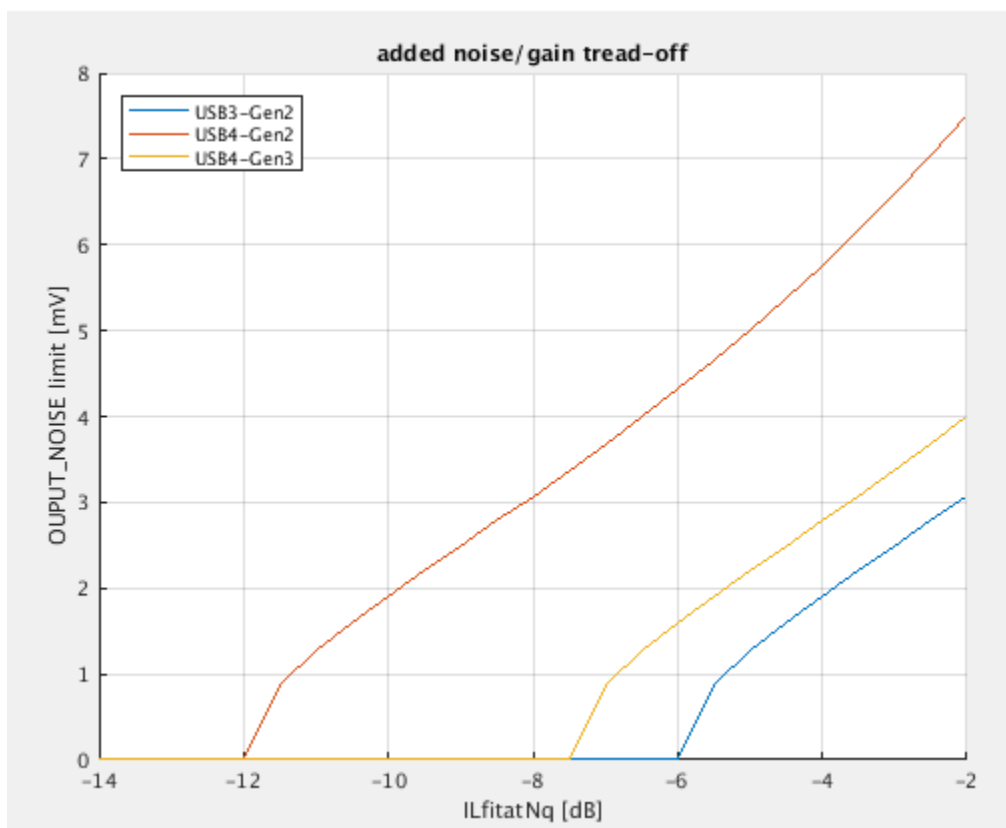
$$\sigma_{cable} \leq \sqrt{10^{\frac{\sigma_1^2}{H(f_N)_{PC} - H(f_N)_{RC}} - \sigma_1^2} \cdot \frac{1}{\alpha}}$$

Using the following parameters:

σ_1	2 mV
α	0.9
$H_{PC}(f_N)$	USB 3.2: -6 dB USB4 Gen2: -12 dB USB4 Gen3: -7.5 dB

Figure 6-11 shows a graph of this function for the given parameters.

Figure 6-11 OUTPUT_NOISE Limit Versus $IL_{fitatNq}$



6.1.2.5.4.6 SIGMA_E (σ_e)

σ_e is the standard deviation of the non-linearity related noise added by the LRD-based active cable.

This measurement **shall** be performed twice: once with minimum input swing and once with maximum input swing.

Then, $\sigma_e = \max(\sigma_{e \text{ max swing}}, \sigma_{e \text{ min swing}})$.

where:

maximum input swing
 = 1300mVpp (USB4 Gen2/3), or 1200mVpp (USB 3.2) or 1000mVpp (USB4 Gen4)
 Minimum input swing = 800mVpp
 (compatible with USB 3.2/USB4 specification max swing definition)

For **USB4** Gen4, CABLE_SNR results **should** be further calculated as:

$$CableSNR = db\left(\frac{PulseResponsePeak}{\sqrt{C_e \cdot \sigma_e^2 + C_n \cdot \sigma_n^2}}\right)$$

The value of C_e and C_n is still TBD and will be added by an ECN when determined.

More details on the calculation of the non-linearity noise can be found in the Active Cable CTS and in Appendix G of this document.

6.1.2.5.4.7 Integrated Return-Loss (IRL)

The IRL term for an LRD-based active cable is calculated similarly to the passive cable, see 3.7.2.3.5.

The IRL limit is different for LRD-based active cables and is given by the following functions:

Mode:	IRL Limit:
USB 3.2 Gen2	$IRL \leq 0.046 \cdot ILfitatNq^2 + 1.812 \cdot ILfitatNq - 8.784$
USB4 Gen2	$IRL \leq 0.046 \cdot ILfitatNq^2 + 1.812 \cdot ILfitatNq - 8.784$
USB4 Gen3	$if(-4dB \leq ILfit@Nq): IRL \leq -13dB$ $if(ILfit@Nq < -4dB): IRL \leq ILfit@Nq - 9$
USB4 Gen4	TBD (Will be added by an ECN when determined)

6.1.2.5.4.8 Integrated Multi-Reflection (IMR)

The IMR term for an LRD-based active cable is calculated similarly to the passive cable, see 3.7.2.3.2.

The IMR limit is different for LRD-based active cables and is given by the following functions:

Mode:	IRL Limit:
USB 3.2 Gen2	$IMR \leq 0.126 \cdot ILfitatNq^2 + 3.024 \cdot ILfitatNq - 20.392$
USB4 Gen2	$IMR \leq 0.126 \cdot ILfitatNq^2 + 3.024 \cdot ILfitatNq - 20.392$
USB4 Gen3	$if(-4dB \leq ILfit@Nq): IMR \leq -29dB$ $if(ILfit@Nq < -4dB): IMR \leq 1.741 \cdot ILfit@Nq - 22.143$
USB4 Gen4	TBD (Will be added by an ECN when determined)

6.1.2.5.4.9 Evaluating Channel Operating Margin (COM)

For **USB4** Gen3, Channel operating margin (COM) **shall** be calculated for LRD-based active cables supporting **USB4** Gen3, for evaluating a reference receiver margin based on the cable's measured S parameters.

The measurements of the cable and the setting of the associated COM tool is defined in the Active Cable CTS.

For **USB4** Gen4, Channel operating margin **shall** be calculated with the eCOM tool instead of the standard COM tool.

6.1.2.5.4.10 Cable Output Eye Mask

An LRD-based active cable **shall** meet eye mask limits at its output.

The test setup **shall** be identical to the **USB 3.2** and **USB4** calibrated receiver test which includes worst case passive cable.

During the test, a reference CTLE, DFE and TXFFE settings **shall** be tuned according to the **USB 3.2 /USB4** spec for obtaining the optimal eye.

1. **USB 3.2** has fixed TXFFE settings according to the **USB 3.2** Gen2 TX specification.
2. **USB4** can tune the TXFFE according to the TXFFE preset table in **USB4** Specification Table 3-5.

After obtaining the optimal eye with the passive cable, repeat the same measurement with the LRD cable under test, and compare the extracted eyes.

The optimal LRD cable eye **shall** meet these criteria:

$$\begin{aligned} &LRD \text{ cable eye area} \geq \text{Passive cable eye area} \\ &\text{AND} \\ &LRD \text{ cable eye width} > 0.9 \cdot \text{Passive cable eye width} \end{aligned}$$

For **USB4** Gen4, the optimal LRD cable eye **shall** meet the limits as defined in the CTS.

Note that the maximum eye height is constrained by the spec item *ILfitatWB* that prevents active amplification over the entire frequency range.

6.1.2.5.4.11 **USB4** Gen4 ILfitMask Limits

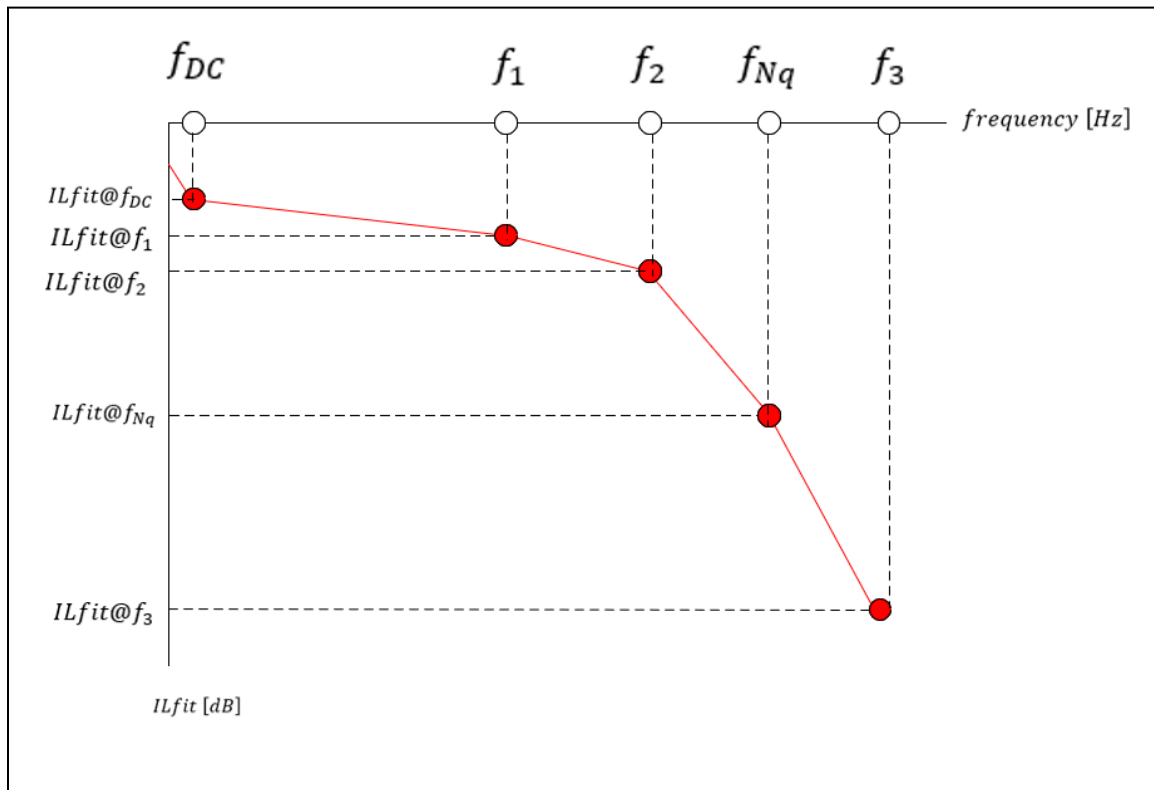
The ILfitMask for an LRD-based active cable supporting **USB4** Gen4 is defined in Table 6-17.

Table 6-17 **USB4 Gen4 LRD-based Active Cable ILfitMask Limits**

Symbol	Frequency	Min	Max	Units
<i>ILfitatDC</i>	100 MHz	-3 [TBD]	0	dB
<i>ILfitatf1</i>	$0.7 \cdot fNq$	<i>ILfitatf2</i>	<i>ILfitatDC</i>	dB
<i>ILfitatf2</i>	$0.85 \cdot fNq$	<i>ILfitatNq</i>	<i>ILfitatf1</i>	dB
<i>ILfitatNq</i>	12.8 GHz	<i>ILfitatDC</i> - 7.5	<i>ILfitatDC</i> - 3	dB
<i>ILfitatf3</i>	$1.2 \cdot fNq$	<i>ILfitatNq</i> - 4 [TBD]	<i>ILfitatNq</i>	dB

This mask **shall** be met for any TXFFE setting in the cable input, and for minimum and maximum swing level in the TX.

Figure 6-12 Gain Parameters for *USB4* Gen4 LRD Active Cables



6.1.2.5.4.12 Cable CM_NOISE

CM_NOISE is defined as the maximum peak value of the signal captured in the cable output with common mode setting on the scope (p+n) and prbs15 data pattern.

The CM_NOISE **shall not** exceed 100mV.

6.1.2.5.4.13 Self-Gain Tuning Capabilities in *USB4* Gen4 LRD Cables

For the **USB4** Gen4 rate, the cable can tune its own gain to the optimal settings depending on the incoming signal. Refer to 6.1.2.5.4.16.3 Phase 4 and Phase 5 for details on when the cable **may** tune its gain.

$t_{LRDSelfTune}$ is defined to be 250 μ s.

After that period the LRD response **shall** be fixed.

Note: this self-tune ability is for gain only, the EQ response **shall** be fixed for all input (see more on the ILfitMask for **USB4** Gen4 in 6.1.2.5.4.4).

Section 6.1.2.5.4.16.3 defines how the cable snoops and adjusts its gain.

6.1.2.5.4.14 *USB4* Gen4 Lane Direction Change

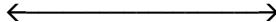

LRD-based active cables that support **USB4** Gen4 **shall** support the asymmetric mode as defined in the **USB4** specification (Section 4.2.2.5) **and the LRD_Tuning_Force_Lane_Reversal** for cable compliance. That means that each high-speed pair of the cable (TX0/TX1/RX0/RX1) **shall** be able to operate in both directions.

The cable **shall** follow 6.1.2.5.4.16.5 and 6.1.2.5.4.16.6 when switching into and out of asymmetric mode.

6.1.2.5.4.15 LRD Active Cable Logical Layer Wiring Diagram

The LRD active cable *shall* follow the wiring diagram as indicated in Table 6-18. The assumption in this table is that there is an LRD in each cable plug.

Table 6-18 USB4 Gen4 LRD-based Active Cable Wiring Diagram for Compliance

USB Type-C Plug #1		Cable Connection	USB Type-C Plug #2			
Pin	Signal Name		Pin	Signal Name		
A1, B1, A12, B12	GND	Fixed Connection	A1, B1, A12, B12	GND		
A4, B4, A9, B9	VBUS	Fixed Connection	A4, B4, A9, B9	VBUS		
A5	CC	Fixed Connection	A5	CC		
B5	VCONN	<i>Ra</i> to GND, diode connection	B5	VCONN		
A6	Dp1		A6	Dp1		
A7	Dn1		A7	Dn1		
A8	SBU1	Fixed Connection	B8	SBU2		
B8	SBU2	Fixed Connection	A8	SBU1		
Shell	Shield	Fixed Connection, outer shield	Shell	Shield		
Normal Operation or Compliance Force Lane Reversal = 0b						
		Symmetric Connection	Enable3TX Plug #1 Enable3RX Plug #2	Enable3RX Plug #1 Enable3TX Plug #2		
A2	TXp1	—— LRD →	—— LRD →	—— LRD →	B11	RXp1
A3	TXn1	—— LRD →	—— LRD →	—— LRD →	B10	RXn1
B11	RXp1	← LRD ——	← LRD ——	← LRD ——	A2	TXp1
B10	RXn1	← LRD ——	← LRD ——	← LRD ——	A3	TXn1
B2	TXp2	—— LRD →	—— LRD →	← LRD ——	A11	RXp2
B3	TXn2	—— LRD →	—— LRD →	← LRD ——	A10	RXn2
A11	RXp2	← LRD ——	—— LRD →	← LRD ——	B2	TXp2
A10	RXn2	← LRD ——	—— LRD →	← LRD ——	B3	TXn2
Compliance Force Lane Reversal = 1b						
		Symmetric Connection	Enable3TX Plug #1 Enable3RX Plug #2	Enable3RX Plug #1 Enable3TX Plug #2		
A2	TXp1	—— LRD →	← LRD ——	← LRD ——	B11	RXp1
A3	TXn1	—— LRD →	← LRD ——	← LRD ——	B10	RXn1
B11	RXp1	← LRD ——	← LRD ——	← LRD ——	A2	TXp1
B10	RXn1	← LRD ——	← LRD ——	← LRD ——	A3	TXn1
B2	TXp2	—— LRD →	← LRD ——	—— LRD →	A11	RXp2
B3	TXn2	—— LRD →	← LRD ——	—— LRD →	A10	RXn2
A11	RXp2	← LRD ——	—— LRD →	← LRD ——	B2	TXp2
A10	RXn2	← LRD ——	—— LRD →	← LRD ——	B3	TXn2

6.1.2.5.4.16 USB4 and TBT3 Logical Layer

The LRD active cable **shall** determine its **USB4** behavior based on snooping the LT, ELT, RT broadcast (RTb) and RT addressed (RTa) transactions.

Table 6-19 defines the SBx transactions the LRD active cable is required to snoop and common **optional** SBx transactions which **may** be snooped. The LRD active cable **may** snoop more SBx transactions if desired. SBx transactions snooped on SBU1 have been transmitted from SBTX of the attached Router or Re-timer and **may** have an assumption of direction depending on the transaction.

Table 6-19 USB4 ~~Gen4~~ Active Cable SBx Transaction Snooping

Transaction Type	Symbol or Transaction Contents	Optional or Required?	USB4/TBT Generation	Action
LT	LT_Fall	Required	<u>TBT3</u>	Transition to CLd.
	LT_Resume	Optional if LOS (loss of signal) is used	<u>TBT3</u>	Transition to Forwarding.
	LT_LROff	Required	<u>All</u>	Transition to CLd. Issued on Lane 0 but affects all both lanes.
	LT_SwitchRx2Tx	Required	<u>Gen4</u>	Transition to/from Asymmetrical when snooped on SBU1.
	LT_SwitchAck	Required	<u>Gen4</u>	Transition to/from Asymmetrical when snooped on SBU1.
	LT_Gen2	Optional if LOS is used	<u>TBT3</u>	Transition to Forwarding. Optional to use specific Gen2 fixed gain and equalization.
	LT_Gen3	Optional if LOS is used	<u>TBT3</u>	Transition to Forwarding. Optional to use specific Gen3 fixed gain and equalization.
	<u>LT_Fall</u>	<u>Required</u>	<u>Transition to CLd.</u>	

Transaction Type	Symbol or Transaction Contents		Optional or Required?	USB4/TBT Generation	Action
RT Broadcast	<i>TBT3</i> -CompatibleSpeed		<i>Optional</i>	<u>TBT3</u>	Modify gain and equalization for TBT rates.
	<i>USB4</i>		Required	<u>All</u>	0b – Use LT transactions for link training. 1b – Use RT transactions for link training.
	SelectedGen		<i>Optional</i>	<u>All</u>	Modify gain and equalization per data rate Gen2, Gen3, Gen4.
	Enable3Tx		Required	<u>Gen4</u>	Asymmetrical Initial link training.
	Enable3Rx		Required	<u>Gen4</u>	Asymmetrical Initial link training.
	Lane1Enabled		<i>Optional</i> if LOS is used	<u>All</u>	0b – CLd lane1. 1b – Enable forwarding lane1.
	Lane0Enabled		<i>Optional</i> if LOS is used	<u>All</u>	0b – CLd lane0. 1b – Enable forwarding lane0.
RT Addressed Index 0 to Gen4TxFFE	Partner_Tx_Status_Byte.TxFFE_Start		<i>Optional</i>	<u>Gen4</u>	Adjustment of AGC is allowed.
	Partner_Tx_Status_Byte.Request_Done		<i>Optional</i>	<u>Gen4</u>	0b – Do not start AGC. 1b – start AGC for tLRDSelfTune.
RT Addressed Index 0 to LRD Tuning	Force Lane Reversal (bit 1)		Required		Required for Compliance Force Lane Reversal. Optional to adjust gain and equalization in a TBD way.
	ELT	ELT_OpDone	<i>Optional</i>	<u>Gen4</u>	No action.

Transaction Type	Symbol or Transaction Contents	Optional or Required?	<u>USB4/TBT Generation</u>	Action
	ELT_Recovery	Required	<u>Gen4</u>	Transition to Forwarding <u>on both lanes.</u>

6.1.2.5.4.16 ~~USB4 and TBT3~~ Logical Layer

The LRD active cable **shall** determine its ~~USB4~~ behavior based on snooping the RT broadcast (RTb) and RT addressed (RTa) transactions, the LT, and the ELT transactions.

6.1.2.5.4.16.1 USB4 and TBT3 Timing Parameters

The LRD active cable **shall** meet the timing requirements in Table 6-20.

Table 6-20 USB4 LRD Active Cable Logical Timing Parameters

Parameter	Description	Min	Max	Units
tLRDForward	The amount of time the LRD takes to transition to Forwarding upon snooping Lane Enable.		1	ms
tLRDSwitch	The amount of time the LRD takes to switch direction after snooping LT_SwitchRX2TX or LT_SwitchAck and receiving no high speed data.		1	ms

6.1.2.5.4.16.2 USB4 and TBT3 USB PD Mode Entry

This section defines the behavior of the LRD cable during the phases of link training.

The **USB PD** controller **shall** use the information from the SOP' and SOP'' (if present) in the **USB PD** Enter Mode command to configure to the LRD to **USB4/TBT3**.

6.1.2.5.4.16.3 USB4 and TBT3 Lane Initialization

The following describes initializing the lanes, including training, across an LRD active cable. See the **USB4** specification for more details regarding the five phases of lane initialization.

Phase 1: Initial Conditions

Receiver and Transmitter terminations are enabled. The common-mode voltage is unchanged. The LRD **shall** assume the Router far side termination is not present in this state. The Router **may** change its common-mode voltage with or without terminations present when transitioning to this state.

The **USB PD** Controller informs the LRD in the active cable of the connection and mode.

Phase 2: Router Detection

No action.

Phase 3: Port Characteristics

No action.

Phase 4: Transmission Start

General Requirements:

Snooping of any SBU transaction other than LT_Fall or LT_LRoff starts this phase. The LRD **may** change its common-mode voltage after snooping a LT or RT transaction. The LRD **shall** meet $V_{dc}+ac_{-}conn$ if changing its common-mode voltage when measured into a 50 Ohm load. The common-mode voltage ramp **shall** complete in tLaneParams minimum defined in **USB4** V2 specification.

Note: Routers and Re-timers present terminations before transmitting LT or RT transactions.

TBT3 Legacy Requirements:

1. Snooped LT transactions on SBU1:
 - a. LT_Gen2 and LT_Gen3: Set fixed gain and equalization. It is **optional** for the gain and equalization to change based on the speed indication. Enable forwarding if not already enabled.
 - b. LT_Resume: Enable forwarding if not already enabled.
2. Snooped Broadcast RT transactions on SBU1:
 - a. **USB4**: If the **USB4** bit is set to 1b, addressed RT transactions are used for link training.
 - b. SelectGen: Set fixed gain and equalization. It is **optional** for the gain and equalization to change based on the speed indication.
 - c. Lane0Enabled, Lane1Enabled: Enable forwarding if not already enabled. **Optional** to use LOS to enable forwarding.

USB4 Requirements:

1. Snooped Broadcast RT transactions on SBU1:
 - a. **USB4**: If the **USB4** bit is set to 1b, addressed RT transactions are used for link training.
 - b. SelectGen: Set fixed gain and equalization. It is **optional** for the gain and equalization to change based on the speed indication.
 - c. Lane0Enabled, Lane1Enabled: Enable forwarding if Enable3Tx or Enable3Rx are not enabled. The LRD **shall** enable forwarding in tLRDForward after snooping the Lane Enable. **Optional** to use LOS to enable forwarding.
 - d. Enable3Tx: Enable three transmitters and one receiver as defined in Table 6-18. Enable forwarding after the LRD has switched direction within tLRDSwitch.
 - e. Enable3Rx: Enable three receivers and one transmitter as defined in Table 6-18. Enable forwarding after the LRD has switched direction within tLRDSwitch.
2. Snooped Addressed RT transaction to SB Register Gen_4_TxFFE with Gen4TxFFE.Tx_Status_Byte.Start_TXFFE:
 - a. If Gen4TxFFE.Tx_Status_Byte.Start_TXFFE (bit 7) =1b, the LRD **may** adjust its AGC in Phase 5.
 - b. If Gen4TxFFE.Tx_Status_Byte.Start_TXFFE (bit 7) =0b, the LRD **may not** adjust its AGC in Phase 5.
 - c. Note that the bytes in Gen4TxFFE.Tx_Status_Byte change if the link is initialized as an asymmetrical link. Refer to the **USB4** V2 specification for details.

~~3. Snooped addressed RT transaction to SB Register LRD_Tuning on SBU1:~~

- ~~a. If LRD_Tuning.Force_Lane_Reversal=0b, use normal Lane1 and Lane2 operation relative to CC without changing the SBU1/2.~~
- ~~b. If LRD_Tuning.Force_Lane_Reversal=1b, swap Lane1 and Lane2 without changing the SBU1/2. The lane reversal **shall** complete within tLRDSelfTune.~~

Phase 5: Link Equalization

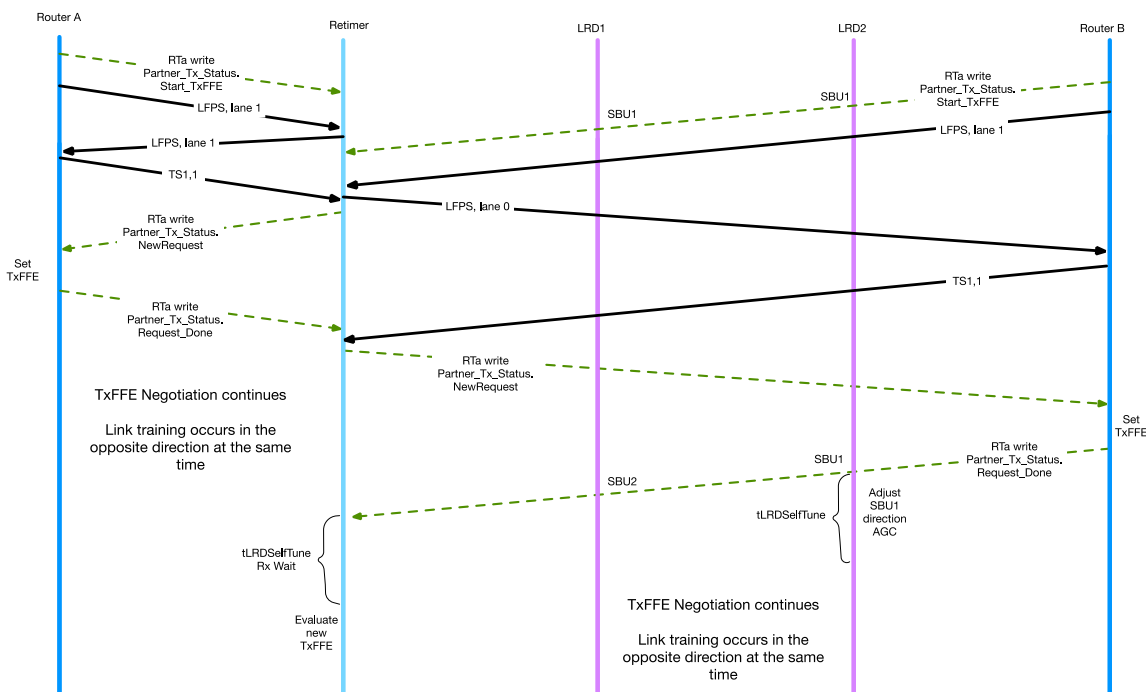
Requirements (only applies to **USB4** Gen4):

1. Snooped addressed RT transaction to SB Register Gen_4_TxFEE:
 - a. Partner_Tx_Status_Byte.Request_Done (bit 6)=1b is the indication to the LRD to start the AGC training in the SBU1 to SBU2 direction.
 - b. The LRD active cable AGC **shall** complete its adjustments within tLRDSelfTune. If there are two LRDs in the cable, each **may** be adjusted independently but tLRDSelfTune **shall** be met for the cable assembly.
- ~~2. Snooped addressed RT transaction to SB Register LRD_Tuning on SBU1:
 - a. **Optionally** save the entire LRD_Tuning register. The LRD **may optionally** adjust its gain and equalization in a TBD way based on the contents of the register. The LRD **shall** complete its gain and equalization adjustment in tLRDSelfTune minimum time (USB4 V2 specification) when it snoops a write to LRD_Tuning.~~

An example of link training with LRD AGC adjustment is indicated in Figure 6-13.

Note: The LRD has no sideband indication of the switch from PAM2 to PAM3 and AGC and equalization training must function on both PAM2 and PAM3 incoming data.

Figure 6-13 USB4 LRD Active Cable Snooping of TxFFE Negotiation



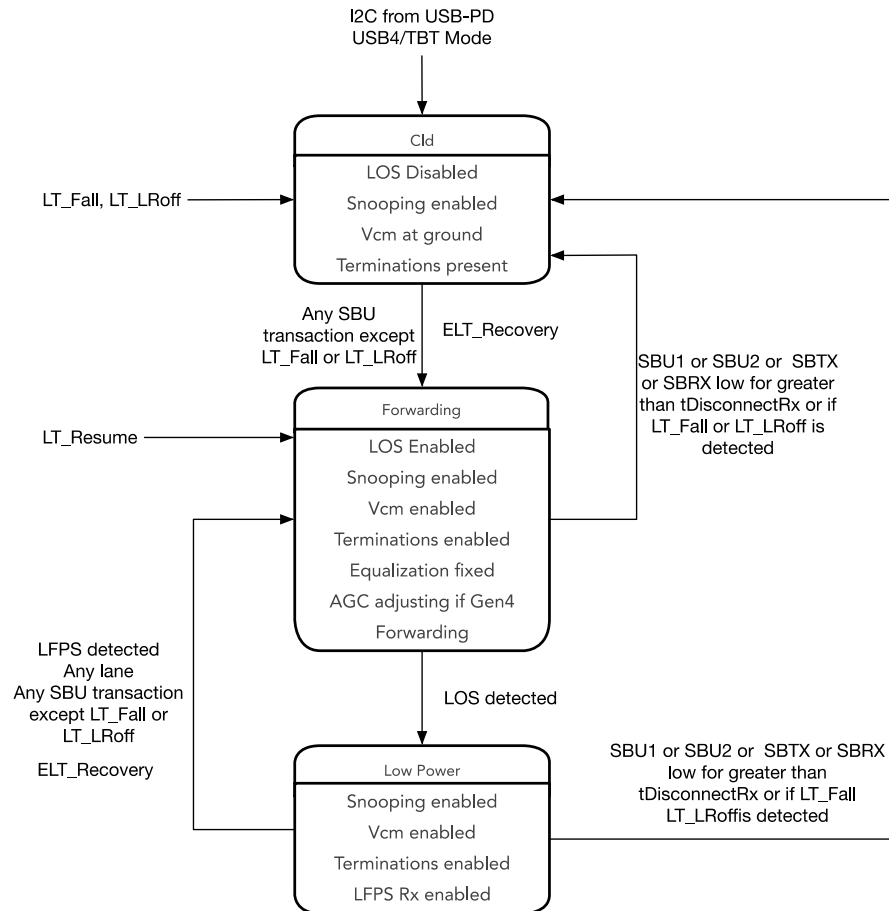
6.1.2.5.4.16.4 USB4 and TBT3 Logical Layer State Machine (Informative)

The state machine in Figure 6-14 describes the behavior of the Logical Layer of the LRD(s) in the active cable. The state machine is per channel (i.e., per direction per lane).

- CLd State – the LRD is disabled but snooping for any traffic on SBU1 (and **optionally** SBU2)
- Forwarding State – the LRD is forwarding. The LRD snoops the link characteristics on the RT broadcast transactions and adjusts its AGC if Gen4 and enabled.

- Low Power State – reduced power state.

Figure 6-14 USB4 State Machine per Channel (Informative)



6.1.2.5.4.16.4.1 Cld State

Entry to State

The LRD **shall** enter this state on the following events:

- Entry to **USB4/TBT3** Alt mode being set from the **USB PD** Controller
- Snooping of LT_Fall or LT_LRoff transaction
- SBU1 or SBU2 low for greater than tDisconnectRx

Behavior in State

- LOS detector disabled.
- SBU1 (and **optionally** SBU2) **shall** be snooped for activity.
- The common mode voltage remains constant.
- The 50 Ohm terminations are present on all TX and RX.

Exit from State

- Transition to Forwarding State upon detection of SBU1 (and **optionally** SBU2) transaction which is not LT_Fall (per lane) or LT_LRoff (lane independent).
- Transition to Forwarding State upon detection of SBU1 (and **optionally** SBU2) ELT_Recovery.

6.1.2.5.4.16.4.2 Forwarding

Entry to State

The LRD **shall** enter this state on the following events:

- Any SBU1 (or **optionally** SBU2) transaction which is not LT_Fall or LT_LRoff
- After exiting Low Power state
- Snooping ELT_Recovery

Behavior in State

- Enable the LOS receiver.
- Snoop SBU1 (and **optionally** SBU2).
- Determine the direction of the lanes if **USB4** Gen4.
- Ramp the common mode voltage if needed as defined in the link training.
- Forward the high-speed data. Use the last known AGC setting if the transition was from Low Power or ELT_Recovery.
- Forward the high-speed data and adjust the AGC per Phase 4 and 5 Link training if Gen4.
- Forward the high-speed data and use the fixed stored AGC ~~and EQ~~ if Gen2 or Gen3.
- If Entry to Forwarding was from ELT Recovery, both lanes **shall** enable forwarding.

Exit from State

- Transition to CLd on detection of low on SBU1 (or **optionally** SBU2) for greater than tDisconnectRx or upon snooping LT_Fall or LT_LRoff.
- Transition to Low Power if LOS is detected.

6.1.2.5.4.16.4.3 Low Power State

The low power state is used to conserve power in electrical idle.

Entry to State

- **USB4** Gen2/3 and TBT3/4: Transition to Low Power when detecting no high-speed signal on the Rx input **per lane**.
- **USB4** Gen4: Transition to Low Power when detecting no high-speed signal on a channel.

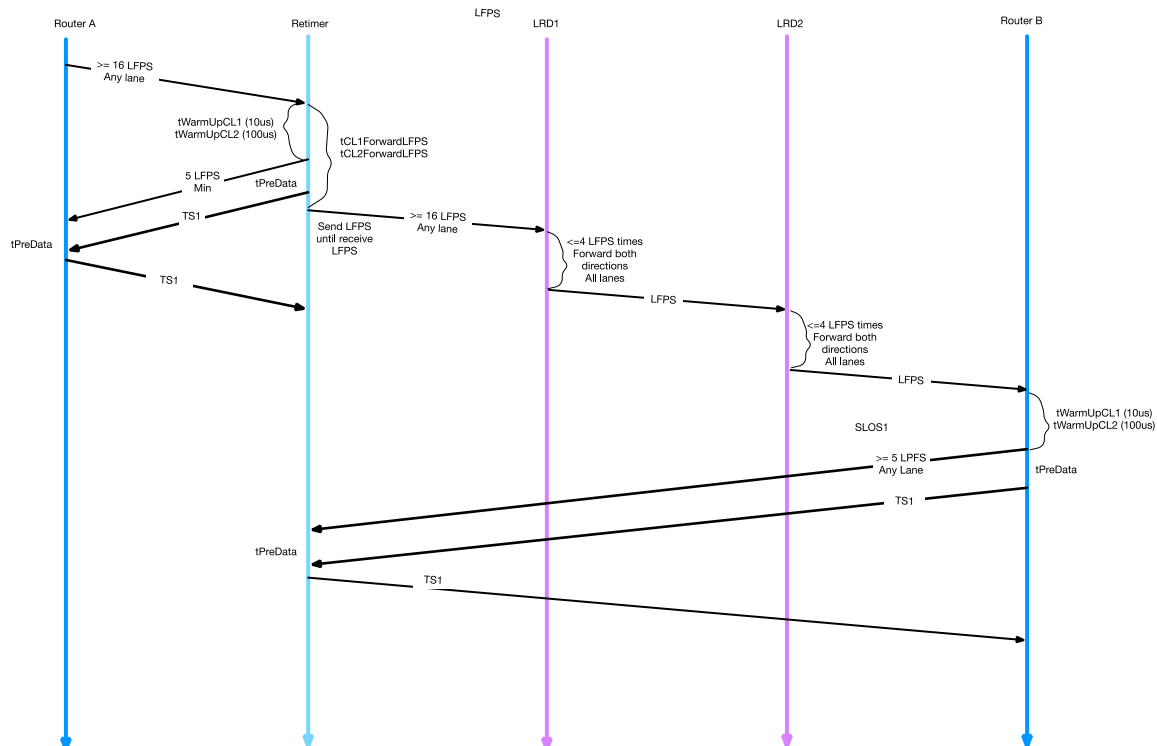
Behavior in State

- Snoop SBU1 (and **optionally** SBU2).
- Common mode voltage **shall** be maintained.
- Terminations **shall** be maintained.
- Consume less power than in Forwarding.

Exit from State

- Transition to Forwarding:
 - USB4 Gen2/3 and TBT3/4: Detection of LFPS on any channel **shall** transition the LRD to Forwarding on the channel receiving the LFPS and the channel in the same lane in the opposite direction. Note that LFPS **may** be on only one channel (one lane in one direction) if the link is in CL0s.
 - USB4 Gen4: Detection of LFPS on any channel **shall** transition the LRD to Forwarding on both channels in both directions.
 - The LRD active cable **shall** consume a maximum of eight LFPS when transitioning to Forwarding.
 - The LRD **shall not** forward LFPS until it has enabled forwarding in both directions. Refer to Figure 6-15.
 - Meet the electrical requirements when forwarding LFPS
 - Snoop of ELT_Recovery
- Transition to CLd:
 - If SBU1 (or **optionally** SBU2) is low for greater than tDisconnectRx or if LT_Fall, or LT_LR_Off is detected.
 - The transmitter and receiver **shall** maintain the common-mode voltage and terminations.

Figure 6-15 USB4 Gen4 CL1/CL2 Exit Flow with LRD Active Cable



6.1.2.5.4.16.5 USB4 Gen4 Symmetric to Asymmetric Transitions

- Figure 6-16 provides an example of the Symmetric to Asymmetric flow for an LRD active cable.
- The LRD transitions from symmetrical to asymmetrical when it snoops the LT_SwitchRX2TX or LT_SwitchACK on SBU1.

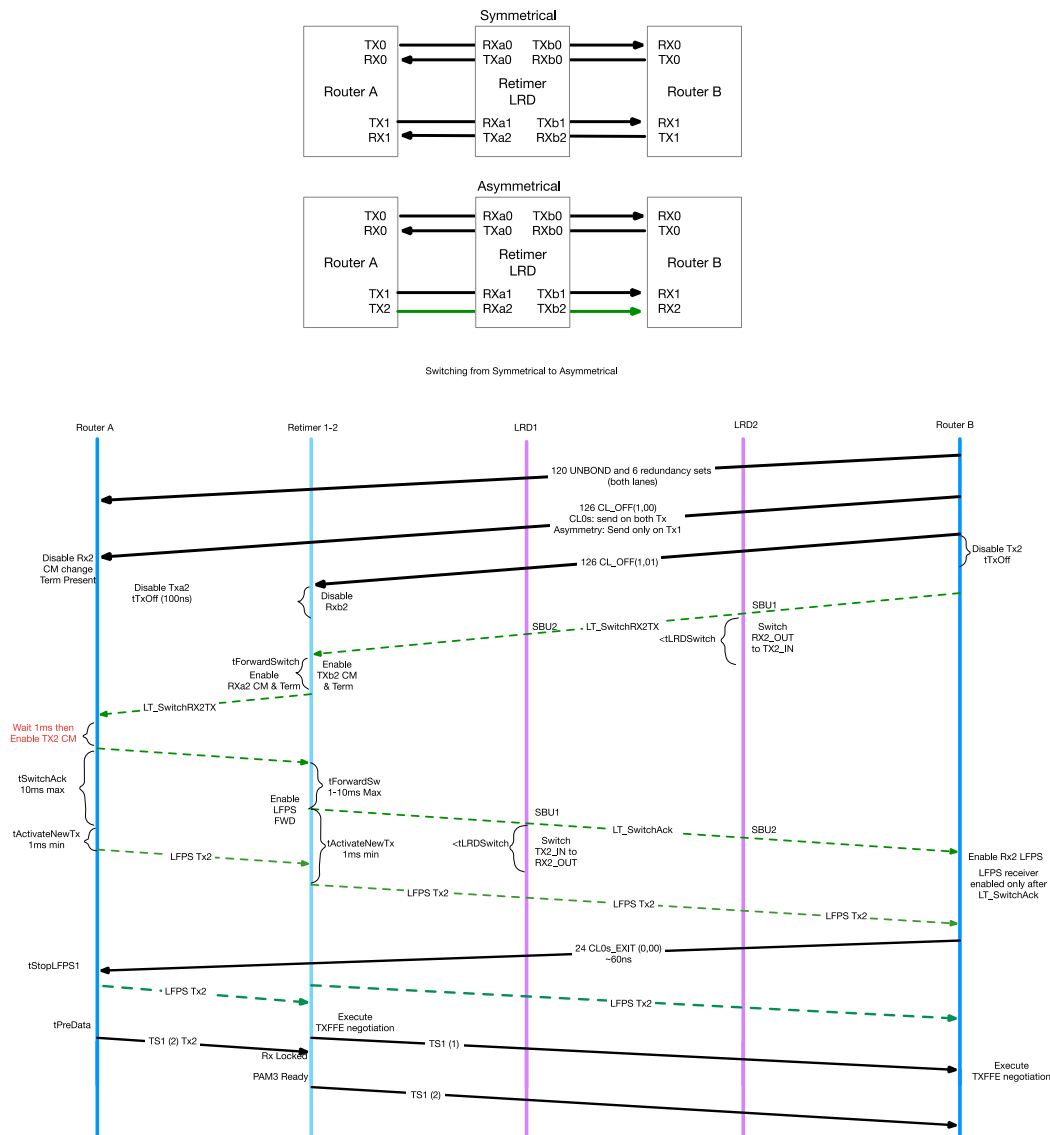
- The LRD switches its RX2_OUT to a TX2_IN on receipt of the LT_SwitchRX2TX on SBU1.
- The LRD **shall** check for high-speed signal when receiving the LT_SwitchRX2TX. If high speed signal is detected by the LRD on the lane which will switch direction, the LRD **shall** wait for no high-speed signal and then perform the direction switch within tLRDSwitch. The LRD **shall** assume this is a transition to Asymmetrical because the previous state was Symmetrical.
- The LRD switches its TX2_Out to RX2_IN on receipt of the LT_SwitchACK on SBU1 within tLRDSwitch.

The LRD transmitter and receiver **shall** modify their common-mode voltage after receipt of the LT_SwitchRX2TX or LT_SwitchACK if needed. The transmitter and receiver **shall not** exceed the V_DC_AC_CONN during the transition assuming the far side termination present. The transmitter and receiver **shall** apply terminations after the common-mode voltage ramp if the termination was removed during the common-mode ramp.

- The LRD **shall not** forward LFPS until it has completed direction switching.

Note that if no LT_SwitchACK is received within 1 second, the Link will disconnect by pulling SBTx low tDisconnectTx.

Figure 6-16 USB4 Symmetric to Asymmetric Transition

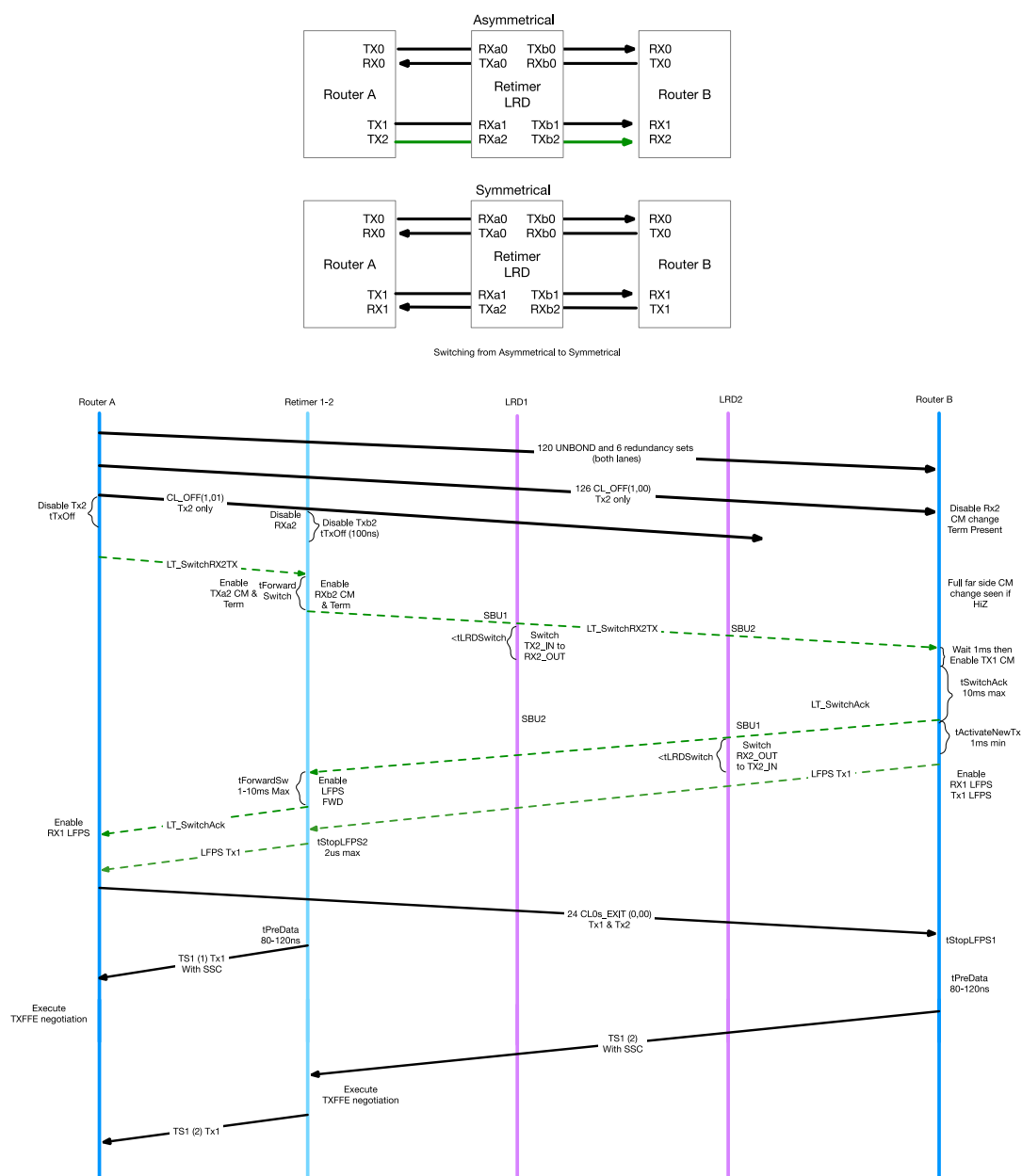


6.1.2.5.4.16.6 **USB4 Gen4 Asymmetric to Symmetric Transitions**

- Figure 6-17 provides an example of the Asymmetric to Symmetric flow for an LRD active cable.
- The LRD transitions from asymmetrical to symmetrical when it snoops LT_SwitchRX2TX or LT_SwitchACK on SBU1.
- The LRD switches its TX2_OUT to a RX_IN on receipt of the LT_SwitchRX2TX on SBU1.
- The LRD **shall** check for high-speed signal when receiving the LT_SwitchRX2TX. If high speed signal is detected by the LRD on the lane which will switch direction, the LRD **shall** wait for no high-speed signal and then perform the direction switch within tLRDSwitch. The LRD **shall** assume this is a transition to Symmetrical because the previous state was Asymmetrical.
- The LRD switches its RX2_Out to TX2_IN on receipt of the LT_SwitchACK on SBU1 within tLRDSwitch.
The LRD transmitter and receiver **may** ramp their common-mode voltage after receipt of the LT_SwitchRX2TX or LT_SwitchACK if needed. The transmitter and receiver **shall not** exceed the V_DC_AC_CONN during the transition assuming the far side termination present. The transmitter and receiver **shall** apply terminations after the common-mode voltage ramp if the termination was removed during the common-mode ramp.
- The LRD **shall not** forward LFPS until it has completed direction switching.

Note that if no LT_SwitchACK is received within 1s, the Link will disconnect by pulling SBTx low tDisconnectTx.

Figure 6-17 USB4 Asymmetric to Symmetric Transition



6.1.2.5.4.17 Cable Compliance

The LRD in the active cable **shall** support snooping the LRD Tuning register as defined in . The Compliance Software ~~only writes to the LRD Tuning Register if it needs to swap Lane1 and Lane2 in the active cable. The Compliance Software first writes to the Router LRD Tuning Register to set Force Lane Reversal=1b with Response Request=1b. The connected Router will then write to the LRD Tuning register in the test equipment with the same Force Lane Reversal=1b with Response Request=0. This allows LRD active cables which only snoop SBU1 to snoop the LRD Tuning Register write on the cable plug farthest from the Compliance Software and test equipment.~~

The Compliance Software then writes to the LRD Tuning register to set Force Lane Reversal=1b with Response Request=0b to inform the local cable plug.

The active cable **shall** switch back to normal Lane1 and Lane2 operation on disconnect.

Table – USB4 LRD Tuning Register Snooping

Register	Register Name	Byte	Bits	Field Name and Description	Type	Default Value
7	LRD Tuning	0	0	Response Request —When this bit is set to 1b, the USB4 Port shall send a write Command to the LRD Tuning register in the adjacent Router/Re-timer. The write shall contain the values of the local LRD Tuning register and shall have the Response Request bit set to 0b. The USB4 Port shall then set this bit to 0b.	R	0
			1	Force Lane Reversal —Identifies whether to override the functional Lane Reversal. 0b:—Cable shall use normal Lane1 and Lane2 operation relative to CC 1b:—Cable shall swap Lane1 and Lane2 Note: Sideband SBU1 and SBU2 shall remain in normal orientation regardless of the Force Lane Reversal setting.	R	0

6.1.2.6 Return Loss

Return loss is defined in the **USB 3.2** specification.

6.1.3 Mechanical Requirements

All active cables **shall** meet the mechanical requirements defined in the Section 3.8.

6.1.3.1 Thermal Requirements

6.1.3.1.1 Thermal Shutdown

All active cables **shall** implement a temperature sensor and place the **USB 3.2** signals in the eSS.Disabled state when the plug skin temperature reaches the maximum defined in Table 6-21. Active cables **shall** indicate they are in thermal shutdown if queried via the **USB PD Get_Status** command.

The Thermal Shutdown is cleared by the following events:

- Disconnect, and
- **USB PD** Hard Reset.

6.1.3.1.2 Maximum Skin Temperature

Active cable plug's skin temperature **shall not** exceed a maximum operating temperature of 30 °C above the ambient temperature for a plastic/rubber housing and 15 °C for a metal housing in any operating mode, when measured at an ambient temperature of 25 °C.

The active cable **shall** meet this requirement when tested connected to a port with a motherboard temperature of (ambient + 25 °C).

6.1.3.1.3 Thermal Reporting

Active cables **shall** implement reporting their maximum internal operating temperature in the **USB PD Discover_ID** Command. Active cables **shall** implement reporting their current internal operating temperature in the **USB PD Get_Status** Command on SOP' and SOP" when supported. Active cables **shall** update their reported Internal Temperature at least every 500 ms.

The plug's Internal Temperature is reported in °C and **shall** be monotonic. It is not the plug's skin temperature, but cable manufacturers **shall** correlate the maximum internal operating temperature with the maximum plug skin temperature to ensure shutdown when the maximum plug skin temperature is reached.

Sources and/or Sinks **may** take action to reduce VBUS current to reduce the cable plug internal operating temperature to below the reported maximum operating temperature. It is recommended Sources and/or Sinks poll the plug's Internal Temperature every 2 seconds.

Table 6-21 Cable Temperature Requirements

Temperature Requirements	
Maximum Internal to Skin Temperature Offset	Design Specific
Maximum Internal Operating Temperature	Design Specific
Maximum Skin Temperature Plastic/Rubber ¹	Ambient + 30 °C
Maximum Skin Temperature Metal ¹	Ambient + 15 °C
Thermal Shutdown Skin Temperature Plastic/Rubber ²	77 °C
Thermal Shutdown Skin Temperature Metal ²	60 °C
Notes: 1. When connected to a port with a motherboard temperature that is set to ambient +25 °C. 2. Based on IEC 62368-1.	

6.1.3.2 Plug Spacing

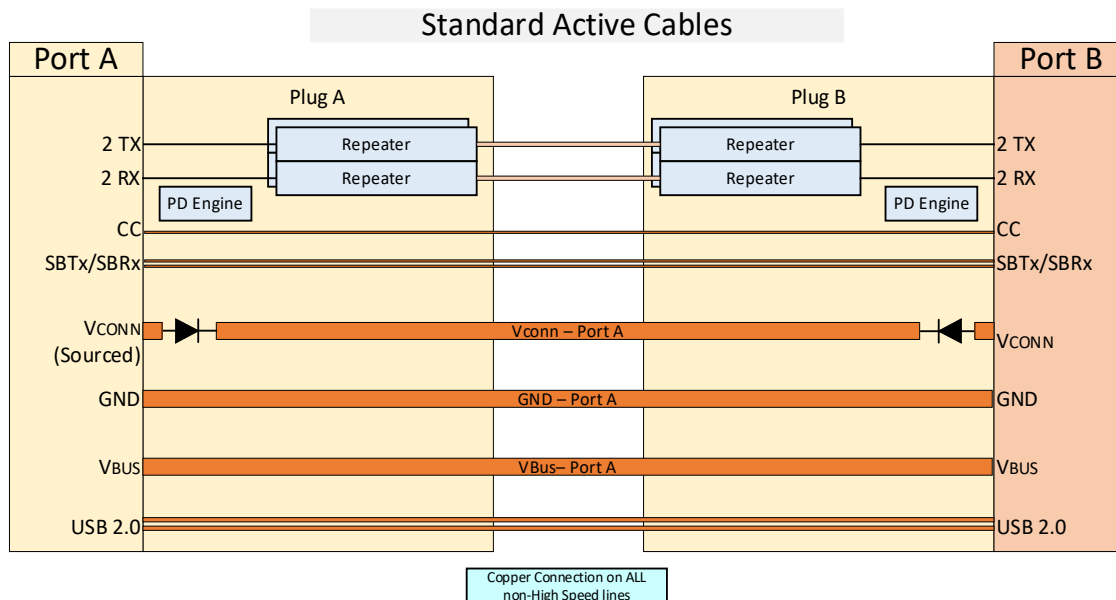
Active cables will support the USB Type-C vertical and horizontal spacing defined Section 3.10.2 when functioning in **USB 3.2** x1 operation. However, this spacing **may** impose thermal constraints. Appendix D provides system design guidance to minimize the thermal impact due to connector spacing. It is recommended that products designed for **USB 3.2** x2 operation with multiple adjacent USB Type-C connectors follow the design guidance in Appendix D to minimize the likelihood the active cable will go into thermal shutdown.

6.2 Additional Specifications for Copper and Hybrid-Optical Active Cables

Active cables include all active element cables including re-timer, re-driver, and hybrid optical-based cables. Hybrid Optical active cables use copper for power and low-speed data and optical for high-speed data.

Active cables minimally support **USB 3.2** Gen 2x2. **USB4** active cables support **USB4** as well as all **USB 3.2** rates. All **USB4** active cables **shall** be interoperable with **Thunderbolt 3** as defined in the **USB4** Specification.

6.2.1 Active Cable Block Diagram



6.2.2 USB4 Asymmetric Mode Support

All **USB4** Gen4-capable Active Cables **shall** support **USB4** Asymmetric Mode. The only exception is for OIAC as defined in Section 6.3. This support **shall** be reported in the **USB PD** Active Cable VDO2 (Bit 1).

6.2.3 Active Cable USB PD Requirements

Active cables **shall** support **USB PD** Revision 3, Version 1.2 or later. Active cables **shall** support **USB PD** Structured VDMs.

6.2.3.1 SOP' and SOP'' Requirements

SOP' and SOP'' are fixed in the active cable at the time of manufacture. See Section 4.5.2.4.

Active cables **shall** respond to Discover_Identity and **USB PD Get_Status** on SOP'. When the SOP'' Controller Present bit is set in the Active Cable VDO, they **shall** respond **USB PD Get_Status** on SOP'' as well.

6.2.3.2 Cable Status

The **USB PD Get_Status** Command is used to discover the current state of the active cable. Cable status **shall** be reported on SOP' and **shall** also be reported on SOP'' when the SOP'' Controller Presence bit is set in the Active Cable VDO.

6.2.4 Active Cable Behaviors in Response to USB PD Events

Each cable plug of the short active cable **shall** be capable of communicating on SOP' and SOP'' if reported in the **USB PD Discover_Identity** Command.

6.2.4.1 Data Role Swap

Active cables are transparent to the **USB PD** Data Role swap.

6.2.4.2 Power Role Swap

Active cables **shall** maintain **USB 3.2** signaling during a **USB PD** Power Role swap. The source of VCONN is not affected by a Power Role Swap.

6.2.4.3 VCONN Swap

Active cables **shall** maintain **USB 3.2** signaling during a **USB PD** VCONN swap. During a VCONN Swap, the original VCONN Source continues to supply VCONN for some time after the new VCONN Source begins to supply VCONN. This ensures that VCONN is never dropped.

6.2.4.4 Fast Role Swap

Active cables will drop **USB 3.2** signaling as a side-effect of a Fast Role Swap if VCONN is not maintained during the Fast Role Swap.

6.2.5 Active Cable Power Requirements

6.2.5.1 VBUS Requirements

Active cables **shall** meet the limits of the IR Drop on VBUS and ground defined in Section 4.4.1.

Active cables **shall** provide VBUS and support at least 3 A and **optionally** 5 A current.

6.2.5.2 VCONN Requirements

Active Cables **shall**:

- Meet the VCONN sink requirement defined in Table 4-6, Table 6-6 (U-State Requirements) and Table 6-12 (CLx Requirements).
- Connect VCONN as shown in Sections 6.2.1.

Active cables **shall** meet the VCONN requirements specified in Section 4.9.

6.3 Additional Specifications for Optically Isolated Active Cables

Optically Isolated Active Cables (OIACs) key features are that they support longer lengths up to 50 meters and provide electrical isolation between the two ends of the cable.

This specification defines support for the following OIACs:

- **USB 3.2** Gen2
- **USB4** Gen3 (**USB 3.2** Gen2, **USB4** Gen2, **TBT3**, and **USB4** Gen3)
- **USB4** Gen4 (**USB 3.2** Gen2, **USB4** Gen2, **TBT3**, **USB4** Gen3, **USB4** Gen4)

They are targeted for Industrial, Machine Vision, Remote Sensor, Pro Video, and Medical applications.

Since no power runs through an OIAC, VCONN must be provided from both ends and **USB PD** is required to correctly setup DFP-UFP link. Therefore, an OIAC can only be used to connect a Source DRD to a Source DRD or a Source DRD to a DFP, and at least one of the Sourcing DRDs must start out as a Sourcing DFP and ACCEPT a **DR_Swap** request to a UFP. **USB PD** Revision 3.0 or higher must be supported on both port partners for an OIAC to function. These requirements are detailed in Section 6.3.2.2 and 6.3.2.3.

OIACs have additional limitations on systems, even when a viable High Speed link has been established, including but not limited to some power saving modes, and features that require **Alternate Modes** or **USB 2.0**, which are not supported by the OIAC. Additional system limitations are defined in Section 6.3.2.

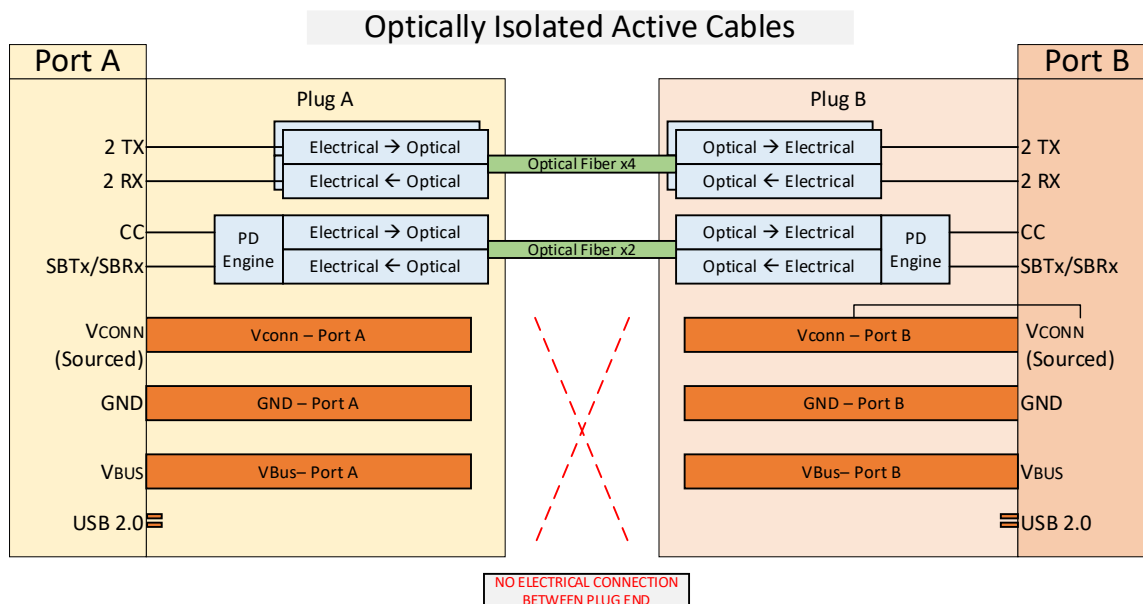
If a connection to a **USB 2.0** Device is required at the end of an OIAC, an adapter with a **USB 3.2** to **USB 2.0** transaction translator and VBUS/VCONN Source **may** be connected at the Device side of the cable to convert the **USB 3.2** signals to **USB 2.0** and provide power to the **USB 2.0** Device and the OIAC.

OIACs **may optionally** support other **Alternate Modes**. If an OIAC supports **Alternate Modes** that require the use of SBUs, the SBUs **shall** be optically isolated.

6.3.1 OIAC Block Diagrams

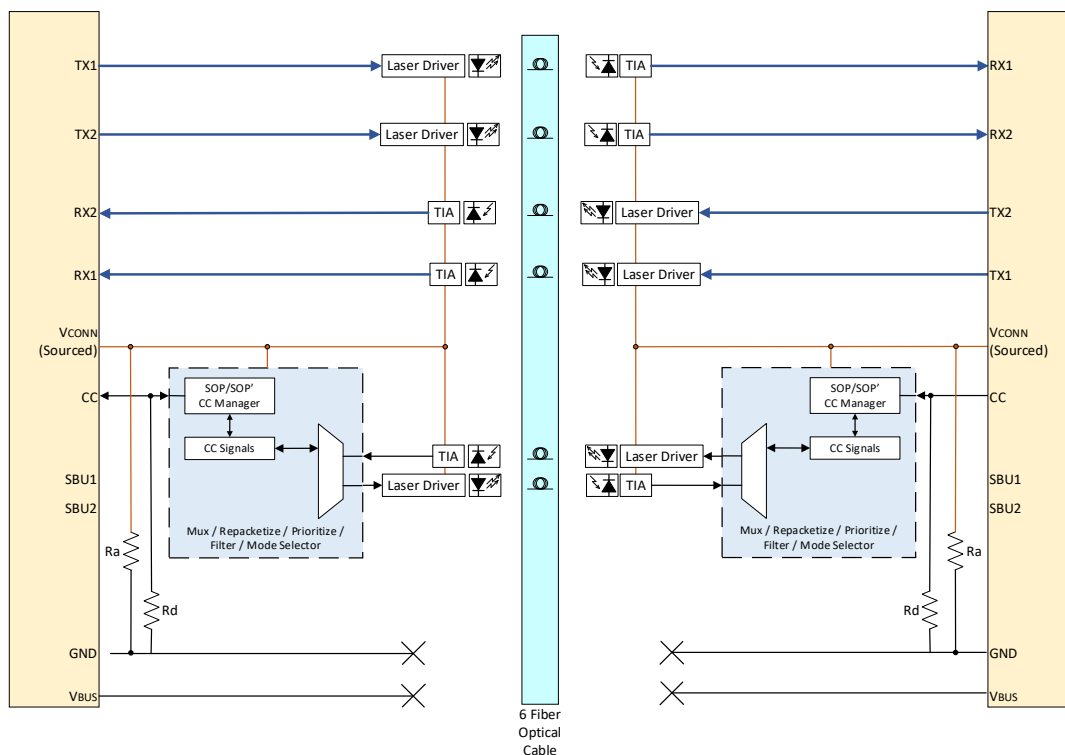
OIACs can be implemented in different ways. This section provides one possible implementation example for the **USB 3.2** and the corresponding **USB4** example. Other implementations can be used, as long as the all the electrical plug specification and characteristics are the same.

6.3.1.1 Simplified Optically Isolated Active Cable (OIAC) Block Diagram

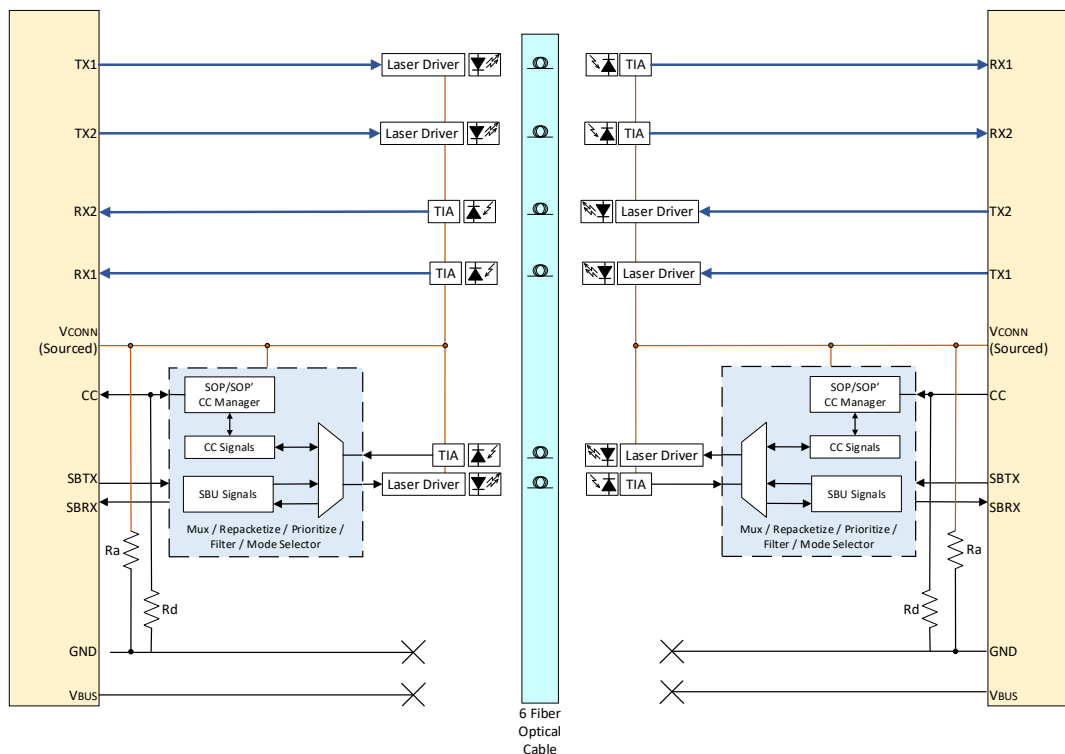


6.3.1.2 Example – Detailed OIAC Block Diagram – USB 3.2 Gen2

In this example, SBU lines are unused unless directed to enter an *Alternate Mode*.



6.3.1.3 Example – Detailed OIAC Block Diagram – USB4 Gen3/Gen4 Block Diagram



6.3.2 OIAC Limitations and General Requirements

6.3.2.1 Limitations of OIAC versus Active Cable

The Optically Isolated Active Cable (OIAC) **shall not** have any electrical wires running between the two plug ends. There are no electrical wires for VBUS, VCONN, GND, SBx, or **USB 2.0**.

VBUS, VCONN, and GND are completely isolated on each end of the cable, which means that the host/device must be both self-powered and supply VCONN to the OIAC.

The following features are **optional**:

- **DP Alternate Mode.**
- **USB 2.0.**
- **USB PD** Unchunked Messages.
- For **USB4** Gen4 OIAC, **USB4** Asymmetric Mode.

Cable VCONN CLd/U3 Power Requirement: The OIAC does not reduce the power requirement on VCONN during the CLd/U3 power state. The power consumption for VCONN power @ CL3/U3 \leq power at CL0/U0. Additional details are in Section 6.3.6.4.3 and Section 6.3.6.5.2.

The above limitations of the OIAC force additional requirements on the Host / Device connected to the OIAC, including be not limited to both VCONN and **USB PD** being required on each end of the cable. Additional requirements for the Host and Device and listed in Section 6.3.2.2 and Section 6.3.2.3.

6.3.2.2 Host Requirements

All USB certified hosts **should** meet all the requirements to support an OIAC.

However, the minimum requirements are listed here:

1. Source VCONN.
2. Support **USB PD**.
3. Initialize as a Sourcing DFP.
4. Dual Role Data (*).

(*) Conditional Requirement: In the scenario where the Host and OIAC is connected to Dual Role Data (DRD) Host/Device, at least one of the two (the Host on the near end or Host/Device on the far end) must be able to accept a **DR_Swap** when requested. If both ends **"Reject"** (or continuously respond with **"Wait"**) then the DFP-UFP link cannot be created, and a USB link will not be established.

6.3.2.3 Device Requirements

There are more device requirements to use an OIAC and a standard USB device. This is because the OIAC does not have a VBUS wire connecting the DFP and the UFP.

The device will need the meet all the following requirements:

1. Source VCONN.
2. Support **USB PD**.
3. Initialize as a Sourcing DFP.
4. Dual Role Data (*).

(*) Conditional Requirement: In the scenario where the Device and OIAC is connected to Dual Role Data (DRD) Host/Device, at least one of the two (the Device on the near end or Host/Device on the far end)

must be able to accept a **DR_Swap** when requested. If both ends “**Reject**” (or continuously respond with “**Wait**”) then the DFP-UFP link cannot be created, and a USB link will not be established.

Note: It is expected that the Device is a DRD Device that can support a **DR_Swap** request to a UFP.

6.3.2.4 Additional Limitations with using an OIAC with a Downstream Device (Dock/Hub/Device)

Assuming a **USB4** (or **USB 3.2**) link is established using an OIAC, some features that are normally available with a USB Type-C Full Featured Copper Cable **may not** be fully supported in the link including the following:

- Downstream **USB 2.0** ports:
Downstream **USB 2.0** ports will not work on a Dock/Hub/Device connected via an OIAC unless the OIAC implements proprietary support for **USB 2.0**.
- Downstream **Alternate Modes** ports:
Alternate Modes (e.g., **Thunderbolt 3, DP**) are only supported if the OIAC implements the mode.

6.3.3 OIAC Cable Power Requirements

6.3.3.1 VBUS Requirements

There are no VBUS wires across an OIAC.

6.3.3.2 VCONN Requirements

There is no VCONN wire between the two ends of an OIAC, therefore, it is possible for each OIAC plug to consume the entire 1.5 W allocated to VCONN from each port. However, the OIAC must meet all thermal and mechanical requirements of the USB Type-C Active Cable.

OIAC **shall**:

- Meet the VCONN sink requirement defined in Table 4-6, Table 6-6 (U-State Requirements) and Table 6-12 (CLx Requirements).
- Connect VCONN as shown in Section 6.3.1.

OIAC **shall** meet the VCONN requirements specified in Section 4.9.

6.3.3.3 OIAC Power Contract

While OIAC does not have VBUS going through the cable, each OIAC cable plug will have a power contract with their respective ports. This is required for any **USB PD** communication, which is needed for the OIAC to properly initialize.

The recommended power contract response for the OIAC is detailed in Section 6.3.4.3.2.1.2.

6.3.4 OIAC USB PD Requirements

OIACs **shall** support **USB PD** Revision 3, Version 1.6 or later. OIACs **shall** support **USB PD** Structured VDMs.

OIAC **may** support Unchunked Messages.

6.3.4.1 SOP' and SOP'' Requirements

OIAC **shall** respond to **USB PD Discover_Identity** and **USB PD Get_Status** of SOP'.

SOP'' controller is not required in the OIAC, however, a **USB PD Get_Status** of SOP' will respond with Thermal Shutdown and temperature information of the worst case of both cable ends. The OIAC will need to internally communicate thermal information between the plug ends and provide that information in the SOP' response.

OIACs will always have SOP' at the near end and SOP'' is the far end with respect to each port. This differs from other Active Cables where SOP' and SOP'' are predefined.

6.3.4.2 USB4 Asymmetric Mode Support

USB4 Gen4-capable OIAC *may optionally* support **USB4** Asymmetric Mode. This support *shall* be reported in the **USB PD** Active Cable VDO2 (Bit 1) response.

6.3.4.3 USB PD Message Handling

Table 6-22 OIAC USB PD Message Handling

Received Message	Local OIAC Plug Response (Sink)
Resets	
<i>Cable Reset</i>	Specified in Section 6.3.4.3.1
<i>Hard Reset</i>	Specified in Section 6.3.4.3.1
<i>Soft Reset</i>	Specified in Section 6.3.4.3.1
<i>Data Reset</i>	Specified in Section 6.3.4.3.1
Locally Handled Messages	
<i>Get_Sink_Capabilities</i>	Specified in Section 6.3.4.3.2.1.1
<i>Source_Capabilites</i>	Specified in Section 6.3.4.3.2.1.2
<i>PR_Swap</i>	Reject (Section 6.3.4.3.2.2)
<i>FR_Swap</i>	Reject (Section 6.3.4.3.2.2)
<i>VCONN_Swap</i>	Reject (Section 6.3.4.3.2.2)
<i>Get_Source_Capabilities</i>	Not Supported (Section 6.3.4.3.2.3)
<i>Get_Source_Capabilites_Extended</i>	Not Supported (Section 6.3.4.3.2.3)
<i>EPR_Get_Source_Capabilities</i>	Not Supported (Section 6.3.4.3.2.3)
<i>EPR_Get_Sink_Capabilites</i>	Not Supported (Section 6.3.4.3.2.3)
<i>BIST</i>	Specified in Section 6.3.4.3.2.4.1
<i>GoodCRC</i>	Specified in Section 6.3.4.3.2.4.2
<i>PS_RDY</i>	Specified in Section 6.3.4.3.2.4.2
<i>GotoMin</i>	Ignore (Section 6.3.4.3.2.4.3)
Conditionally Handled Messages	
<i>Accept</i>	Specified in Section 6.3.4.3.3.1
<i>Reject</i>	Specified in Section 6.3.4.3.3.2
<i>Wait</i>	Specified in Section 6.3.4.3.3.3
All other USB PD Message shall be allowed to traverse the OIAC	

6.3.4.3.1 USB PD Reset (Hard, Cable, Soft, Data)

All Resets (Cable, Hard, Data, Soft) will be handled the same by the OIAC and all be treated as Hard Reset, which resets VCONN.

For Hard (or Cable) Reset, an OIAC plug *shall*:

1. Detect the Hard (or Cable) Reset ordered set,
2. Forward “disconnect event” to the remote plug, and
3. Remove [Rd](#).

For Data (or Soft) Reset, an OIAC plug *shall*:

1. Detect the Data (or Soft) Reset Message,
2. Forward “disconnect event” to the remote plug, and
3. Remove **Rd**.

At the Remote OIAC Plug:

1. Receive “disconnect event”, and

Note: Forwarding a “disconnect event,” to the remote plug can be accomplished with a simple internal message or any other means the implemented decides.

2. Remove **Rd**.

A Hard Reset signal can occur at any time during normal operation of the cable and during the cable initialization. This signal will take precedent over the initialization state machine and immediately forward the Hard Reset Message to the remote plug, using an internal cable message.

6.3.4.3.2 Locally-Handled SOP *USB PD* Messages Received by the OIAC Plug

6.3.4.3.2.1 Specific Responses From the Local OIAC Plug

6.3.4.3.2.1.1 Get_Sink_Capabilities

The OIAC plug **shall** respond to this message directly and the suggested response is defined in Table 6-23 and Table 6-24 depending on which phase of the link bring up the OIAC is currently in.

Table 6-23 Recommended OIAC Sink_Capabilities PDO (SOP) on Initial Connection

Bit(s)	Value	Parameter
B31..30	00b	Fixed Supply
B29	0b	Not Dual-Role Power
B28	0b	Not higher capability
B27	0b	Not unconstrained power
B26	1b	USB Communications Capable
B25	0b	Dual-Role Data
B24..23	00b	Fast Role Swap not supported
B22..20	000b	Reserved
B19..10	00 0110 0100b	5V in 50mV units
B9..0	00 0000 0000b	Operational current in 10mA units

Table 6-24 Recommended OIAC Sink_Capabilities PDO (SOP) on Active Connection

Bit(s)	Value	Parameter
B31..30	00b	Fixed Supply
B29	0b	Not Dual-Role Power
B28	0b	Not higher capability
B27	0b	Not unconstrained power
B26	1b	USB Communications Capable
B25	As reported from Remote End	Dual-Role Data ¹
B24..23	00b	Fast Role Swap not supported
B22..20	000b	Reserved
B19..10	00 0110 0100b	5V in 50mV units
B9..0	00 0000 0000b	Operational current in 10mA units
Note 1: Reflection of far side connection.		

6.3.4.3.2.1.2 Source_Capabilities

A recommended Sink RDO is provided as an example in Table 6-25 is the same for both the initial connection and when the OIAC is in normal active mode.

The OIAC **shall** function when receiving the minimum Source PDO and the OIAC plug will request 5 V at 0 mA.

Table 6-25 Recommended OIAC Active Sink RDO (SOP)

Bit(s)	Value	Parameter
B31	0b	Reserved
B20..28	0 0000 0001b	Object Position
B27	0b	No GiveBack flag
B26	0b	No Capabilities Mismatch
B25	1b	USB Communication Capable
B24	1b	USB Suspend
B23	0b	Unchunked Extended Messages
B22..20	000b	Reserved
B19..10	0b	Operating Current in 10 mA units ¹
B9..0	0b	Maximum current in 10 mA units ¹
Note 1: Thermal design must be considered.		

6.3.4.3.2.2 “Rejected” Response From the Local OIAC Plug

The OIAC does not have any copper wires, therefore must supply power from each end independently, which makes any type of power swapping not possible. The following messages **shall** be intercepted by the OIAC and responded with a “**Reject**.”

- **PR_Swap.**
- **FR_Swap.**
- **VCONN_Swap.**

6.3.4.3.2.3 “Non Supported” Response From the Local OIAC Plug

Since the OIAC acting as sink (that must be powered by the local port) on both ends of the cable the following messages are not applicable. The OIAC **shall** respond with a “Not Supported.”

- **Get_Source_Cap.**
- **Get Source_Cap Extended.**
- **EPR_Get_Source_Cap.**
- **EPR_Get_Sink_Cap.**

6.3.4.3.2.4 Handled/Ignored by the Local OIAC Plug

6.3.4.3.2.4.1 BIST

BIST is a function that needs to be supported by the OIAC plug. The BIST message from the port **should** be intercepted and completely handled by the OIAC plug.

6.3.4.3.2.4.2 GoodCRC, PS_RDY

Both messages, GoodCRC and PS_RDY are confirmation messages that the previous message was successful. They just need to terminate in the OIAC plug.

6.3.4.3.2.4.3 GotoMin

GotoMin is a message that will just be completely ignored by the OIAC. As this *should* have no effect but is required to be terminated in the plug.

6.3.4.3.2.4.4 Soft Reset, Data Reset

See Section 6.3.4.3.1.

6.3.4.3.3 Conditional Response Messages

6.3.4.3.3.1 Accept

The **Accept** Message is a Valid response to only the following messages: **Request**, **EPR_Request**, **PR_Swap**, **DR_Swap**, **VCONN_Swap**, **FR_Swap**, **Soft_Reset**, and **USB PD Enter_USB**.

The OIAC, as the Sink, will never initiate any type of Power Swap or EPR request, therefore, an **Accept** response from the Source for those messages (**EPR_Request**, **PR_Swap**, **VCONN_Swap**, **FR_Swap**) *should* never occur.

Soft_Reset will be handled as a reset case and is defined in Section 6.3.4.3.1. There *should* never be an **Accept** message following a **Soft_Reset**.

A **Request** Message can and will initiate from the OIAC, so an **Accept** following that **Request** message will need to be processed within the OIAC plug and will not need to traverse the OIAC.

Both a **DR_Swap** and **USB PD Enter_USB** message can initiate from either end of the OIAC, and therefore *may* require an **Accept** message to traverse the OIAC.

6.3.4.3.3.2 Reject

The **USB PD Reject** message is a Valid response to only the following messages: **USB PD Request**, **EPR_Request**, **PR_Swap**, **DR_Swap**, **VCONN_Swap**, and **Enter_USB**.

The OIAC, as the Sink, will never initiate any type of Power Swap or EPR request, therefore, a **Wait** response from the Source for those messages (**EPR_Request**, **PR_Swap**, **VCONN_Swap**) *should* never occur.

A **Request** Message can initiate from the OIAC, so a **Reject** following that **Request** message will need to be processed within the OIAC plug and *should not* need to traverse the OIAC.

Both a **DR_Swap** and **USB PD Enter_USB** message can initiate from either end of the OIAC, and therefore *may* require a **Reject** message to traverse the OIAC.

6.3.4.3.3.3 Wait

The **Wait** Message is a Valid response to only the following messages: **Request**, **EPR_Request**, **PR_Swap**, **DR_Swap**, **VCONN_Swap**, and **USB PD Enter_USB**.

The OIAC, as the Sink, will never initiate any type of Power Swap or EPR request, therefore, a **Wait** response from the Source for those messages (**EPR_Request**, **PR_Swap**, **VCONN_Swap**) *should* never occur.

A **Request** Message can initiate from the OIAC, so a **Wait** following that **Request** message will need to be processed within the OIAC plug and *should not* need to traverse the OIAC.

Both a **DR_Swap** and **USB PD Enter_USB** message can initiate from either end of the OIAC, and therefore *may* require a **Wait** message to traverse the OIAC.

6.3.4.3.4 Other *USB PD* Messages

6.3.4.3.4.1 Get Status SOP'

OIACs are not required to support SOP', however, when reporting to a *USB PD Get_Status* message, the OIAC must provide accurate temperature information on the worst case of either cable plug, the higher temperature reading. And it must provide thermal shutdown information of either cable plug.

6.3.4.3.5 Messages That Traverse the OIAC

All other *USB PD* messages that are not listed in above section are considered "Generic" and will allowed to traverse the OIAC.

6.3.4.3.6 OIAC Cable Status

The *USB PD Get_Status* Command is used to discover the current state of the active cable. Cable status *shall* be reported on SOP'.

6.3.4.3.7 OIAC *USB PD* Message Forwarding Timing

Figure 6-18 OIAC *USB PD* Message Forwarding

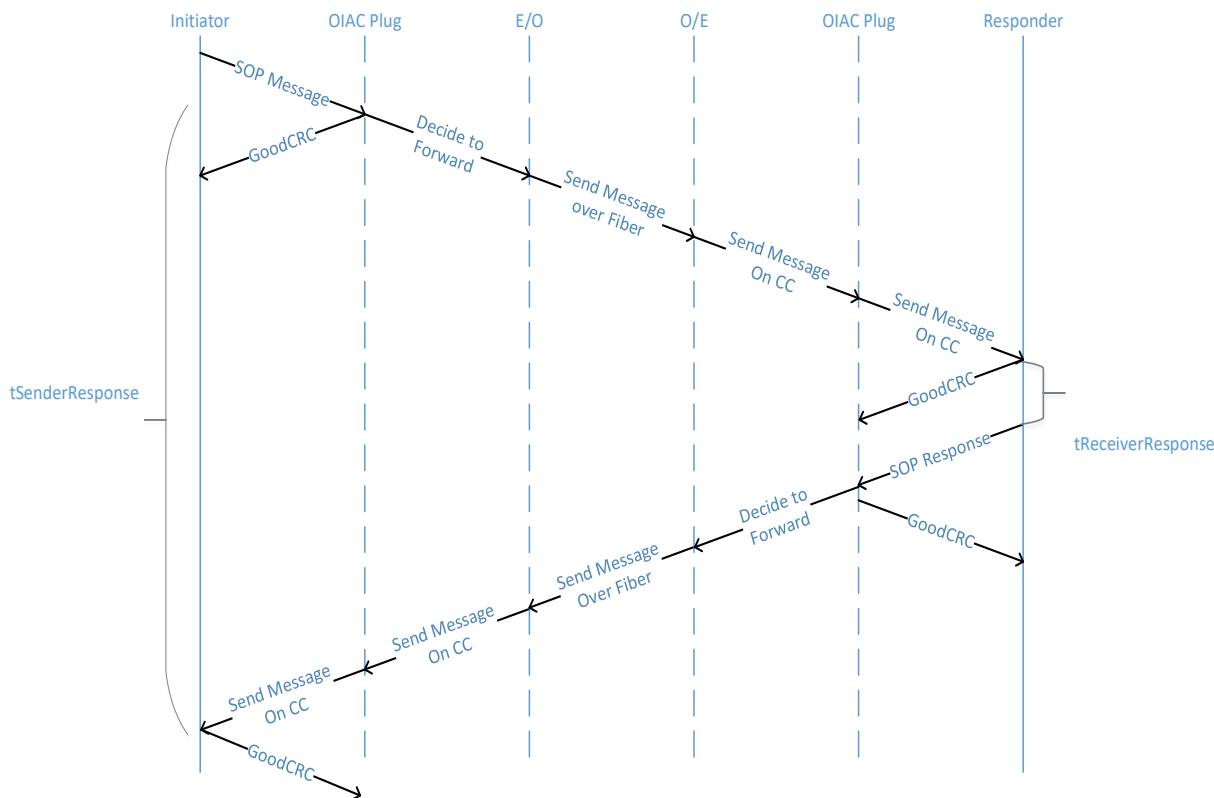


Table 6-26 OIAC *USB PD* Message Timing

Timer	Value (Min)	Value (Max)	Units	Reference
tDRSwapWait	100		ms	Time to wait if WAIT received before sending another <i>DR_Swap</i> . Defined in the <i>USB PD</i> Specification.
tForward		4	ms	Time to forward SOP Message one direction in OIAC.
tTransmit		195	μs	Time to send the GoodCRC. Defined in the <i>USB PD</i> Specification.
tInterFrameGap	25		μs	Time from the end of the last bit of a Frame until the start of the first bit of the next Preamble. Defined in the <i>USB PD</i> Specification.
tReceiverResponse		15	ms	Time for SOP Receiver to respond. Defined in the <i>USB PD</i> Specification.
tSenderResponse	27	33	ms	Time for SOP Initiator to receive response. Defined in the <i>USB PD</i> Specification.
tBootupCRCHoldoff		50	ms	Hold off on GoodCRC until ready to respond with Operational RDO in Phase 3.

Note: The ***USB PD*** specification revision ***shall*** take precedence over this table if any discrepancies exist.

6.3.4.3.8 Internal Cable Messages

All SOP messages ***shall*** be forwarded or terminated as defined in Sections 6.3.4.3.1 and 6.3.4.3.3, and will not be further described in this section.

The messages defined in this section provide ***informative*** guidance on internal messages for OIACs. The actual definition and implementation of each message is left to the implementer.

In this section and Section 6.3.1, there is a defined Plug-A and Plug-B to support ***USB PD*** communication through the OIAC cable. These designations are established at the time of manufacture and are completely internal to the cable. They are used to simplify cable initialization and internal messaging.

6.3.4.3.8.1 MSG_Keep_Alive (*Informative*)

A low duty cycle message that is meant to inform the remote cable plug that the local cable plug is still operational.

A simple example is that only Plug-A will send MSG_Keep_Alive and Plug-B must respond with MSG_Keep_Alive_ACK. Each end will have its own timeout for MSG_Keep_Alive and MSG_Keep_Alive_ACK.

6.3.4.3.8.2 MSG_Keep_Alive_ACK

Acknowledgement message to the MSG_Keep_Alive.

A simple example is that only Plug-A will send MSG_Keep_Alive and Plug-B must respond with MSG_Keep_Alive_ACK. Each end will have its own timeout for MSG_Keep_Alive and MSG_Keep_Alive_ACK.

6.3.4.3.8.3 MSG_Port_Capabilities

This message contains all relevant local port capabilities including but not limited to:

- Chunked/Unchunked capability, and
- DRD/DFP/UEP capabilities.

6.3.4.3.8.4 MSG_Cable_Config

This message contains the final cable configuration based on known system capabilities.

It will contain both relevant ports' capabilities and the final DFP/UEP roles for the system.

This message will also serve as the signal in Phase 2 for the cable plug to start the reboot process.

6.3.4.3.8.5 MSG_Release_Remote_SourceCap_GoodCRC

This is a synchronization message to attempt to bring up both ports at the same time.

It is used in Phase 3 and is the signal to release the GoodCRC message to the Source Capabilities message from the attached port. At the beginning of Phase 3, after each plug has been rebooted, and depending on the final DFP/UEP role, each plug *should* wait for MSG_Release_Remote_SourceCap_GoodCRC before it is allowed to release a GoodCRC in response to a Source_Capabilities message from the port.

6.3.4.3.8.6 MSG_DR_Swap_Init

Initial *DR_Swap* sent by Plug-A to Plug-B to perform a *DR_Swap*.

6.3.4.3.8.7 MSG_DR_Swap_Reject

Plug-B sends this message to report that the initial *DR_Swap* was rejected by its attached port.

This is needed by Plug-A to attempt to re-configure the cable such that the port associated to Plug-B can remain a DFP. This is part of the *DR_Swap* test in Phase 1, shown in the state diagram transition from M3 to M4 (or M3 to M5). It is also possible that this *may* be needed in Phase 3, if the port associated with Plug-B rejects the *DR_Swap*.

6.3.4.3.8.8 MSG_DR_Swap_Accept

Plug-B sends this message to report that the initial *DR_Swap* was accepted by its attached port.

This is needed by Plug-A to continue (M3 to M6 transition) in Phase 1 in the cable initialization.

6.3.4.3.8.9 MSG_Force_Detach

This message is to request the remote plug to disconnect from its attached port. The disconnect method can be done by raising the voltage on the CC line to above *vRd-Connected* or removing *Rd*.

This will cause the remote port to remove VCONN from the remote plug all the circuitry *should* be powered down, therefore resetting any action taken by the plug on the CC line to cause the disconnect.

6.3.4.3.8.10 MSG_Reset

This message is to forward a Hard_Reset signal to the remote plug and port.

An internal Hard Reset message *should* be responded to with an Acknowledgement.

6.3.4.3.8.11 MSG_Acknowledgement

This message is to acknowledge that a message was received.

This message has been explicitly defined in a few specific cases but can be used more broadly.

6.3.4.3.9 Data Role Swap in Active State (*Informative*)

OIACs *shall* support Data Role Swaps on SOP. Each OIAC plug discovers its plug port partner and determines if it is capable of a Data Role Swap during the initialization process described in Section 6.3.5. OIAC cable

plugs generate internal messages to communicate the **DR_Swap**, **Accept**, **Reject**, and **Wait** to the far side of the cable.

- The flow of a successful **DR_Swap** is shown in Figure 6-19.
- The flow of a Responder rejecting a **DR_Swap** is shown in Figure 6-20.
- The flow of a Responder issuing a **Wait** to a **DR_Swap** is shown in Figure 6-21. Note: The **USB PD Wait** and Retry timers **shall** follow all timing requirements in **USB PD**.
- The flow of an Initiator rejecting a cable plug **DR_Swap** due to an **Accept** from the Responder is shown in Figure 6-22.
- The flow of an Initiator issuing a **Wait** to a cable plug **DR_Swap** due to an **Accept** from the Responder is shown in Figure 6-23. Note: This does not follow the **USB PD Wait** timing, because a Hard Reset is initiated.

Figure 6-19 OIAC Successful Data Role Swap

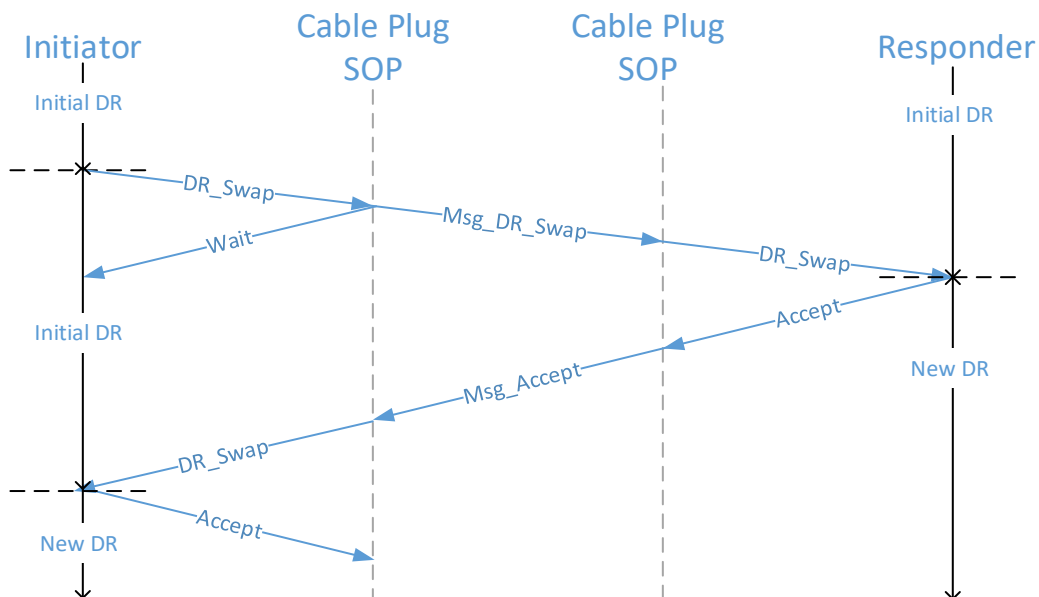


Figure 6-20 OIAC Rejected Data Role Swap

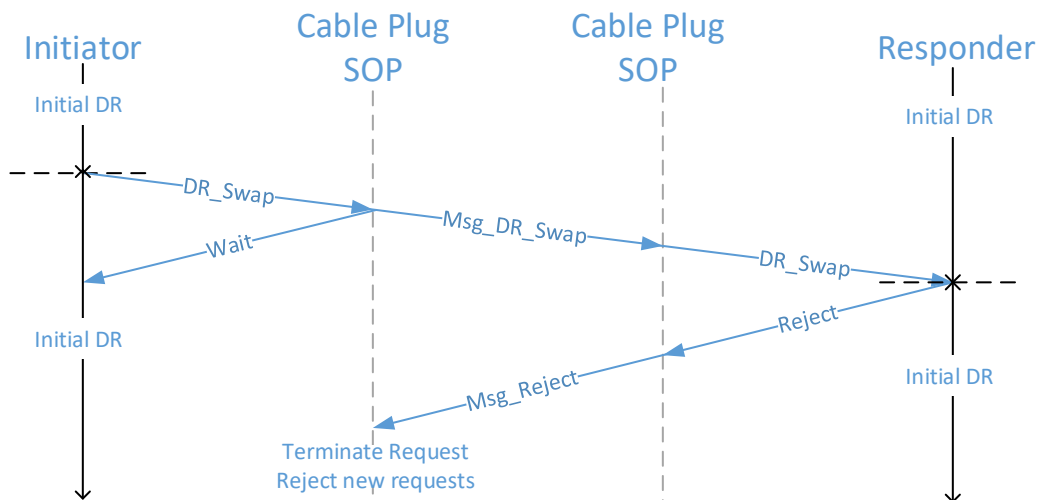


Figure 6-21 OIAC Wait Data Role Swap

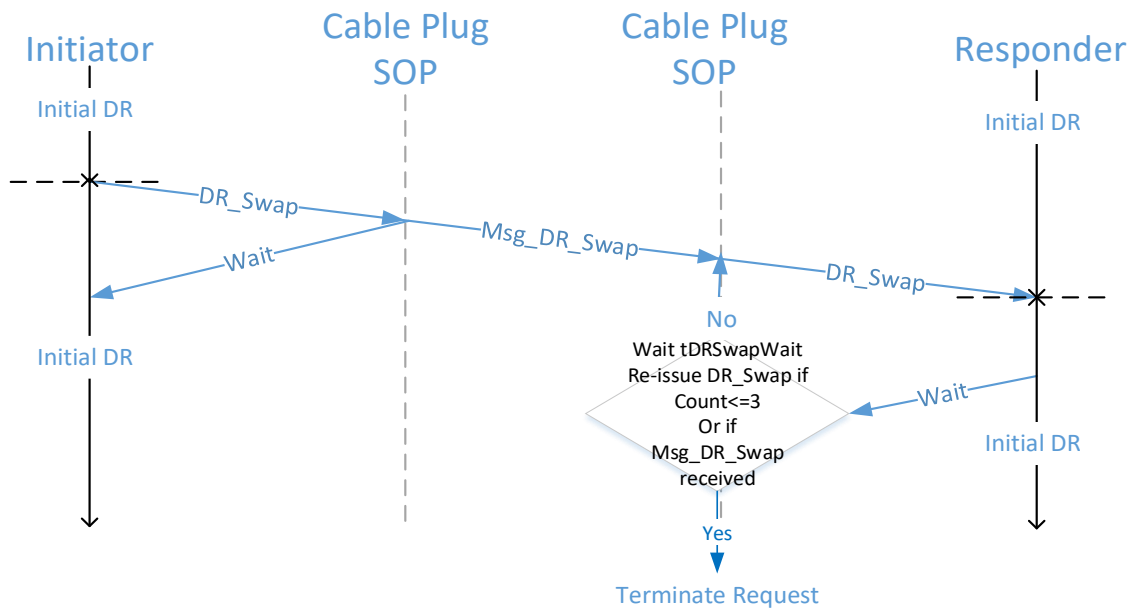


Figure 6-22 OIAC Initiator Reject Data Role Swap

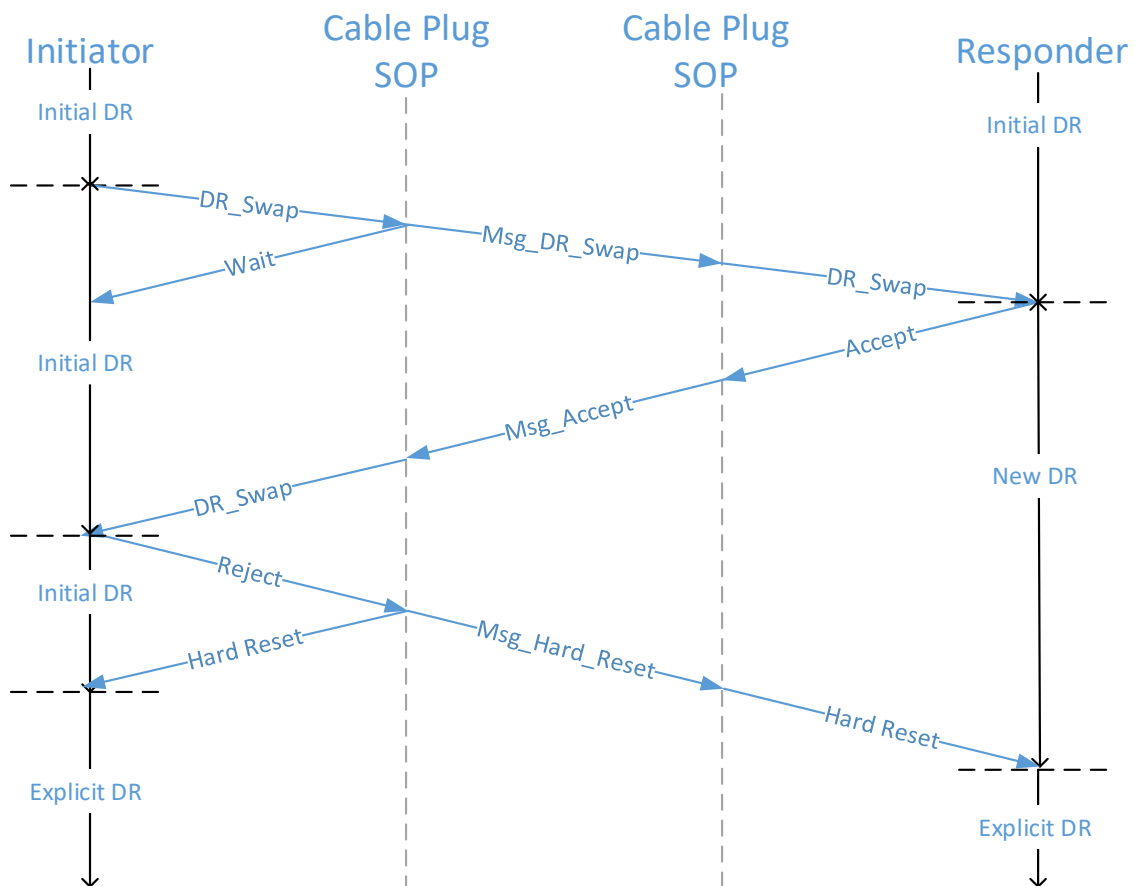
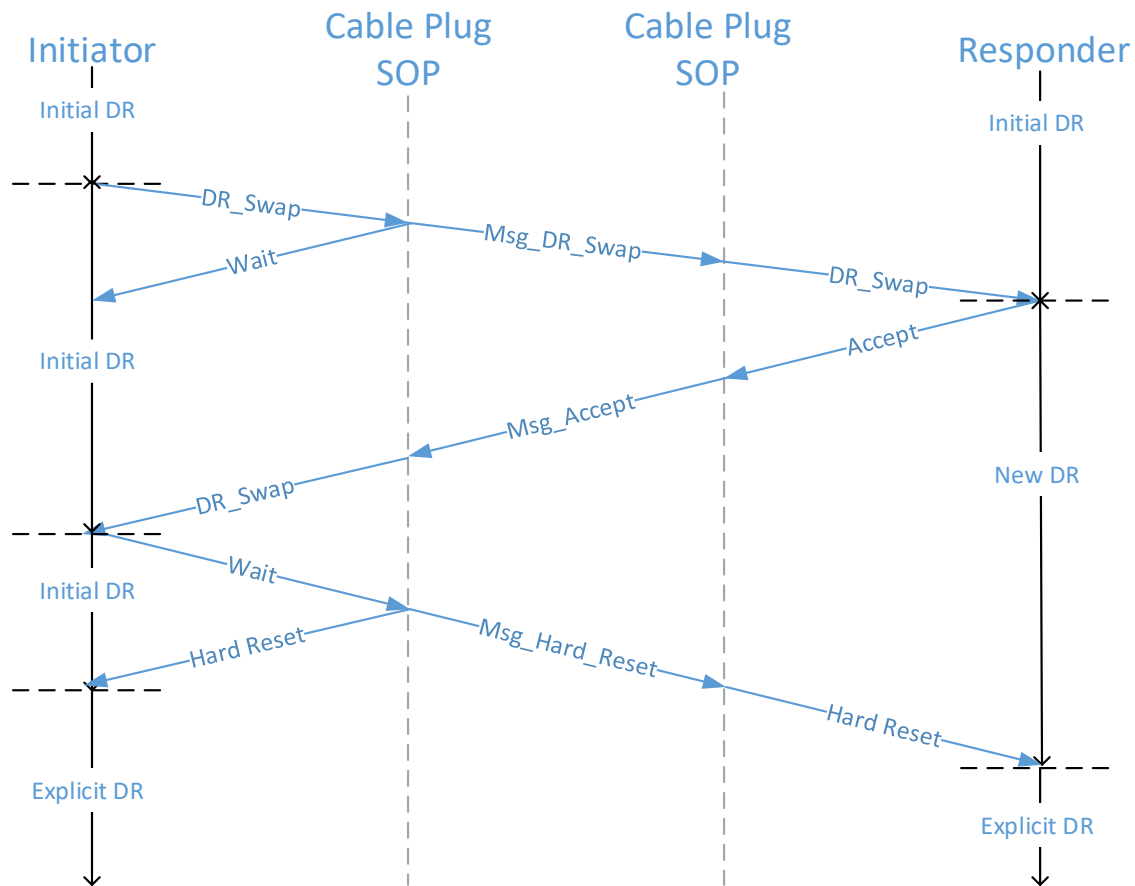


Figure 6-23 OIAC Initiator Wait Data Role Swap



6.3.5 OIAC Connection Flow and State Diagrams

This section defines the connection state diagrams for the OIAC. The connection sequence applies to both **USB 3.2** and **USB4** OIAC. To complete bringing up the **USB4** link, the **USB4** OIAC is also required to handle all **USB4**-related messages as defined in **USB PD**.

OIAC plug defined at time of manufacture as either Plug-A or Plug-B for **USB PD** communication. This in no way indicates the plug has more or less capability, rather it allows for a consistent behavior when making the initial end to end connection.

The OIAC communicates using **USB PD** with its plug partners to determine the partner capabilities. The OIAC performs a series of connect/disconnects to establish the correct UFP/DFP data role for the cable plug. The possible combinations for ports and cable plugs is defined in Table 6-27.

The connection and establishment of data roles is performed in three phases.

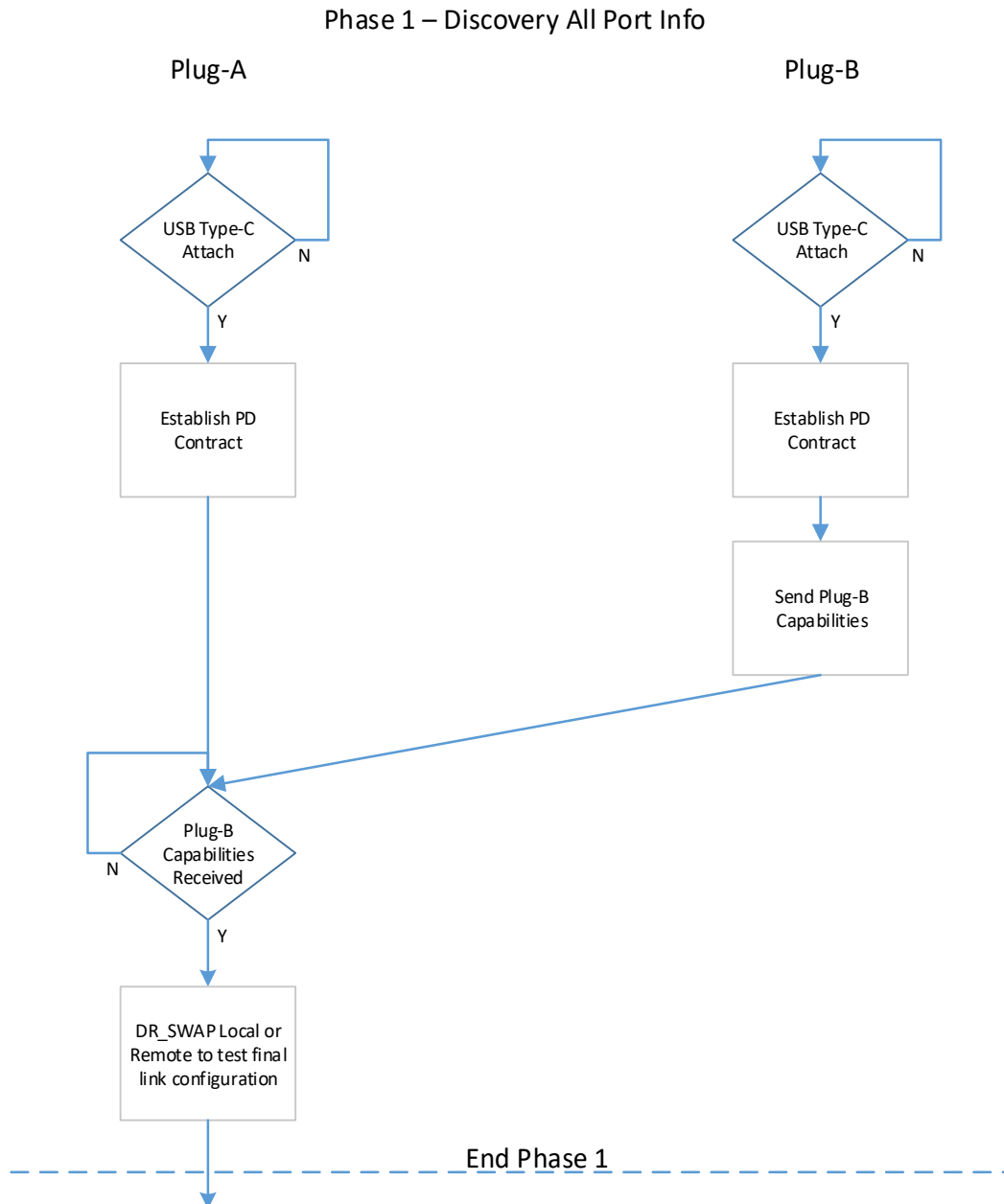
Table 6-27 Port and Plug Capabilities

Host/Device Port Capabilities	Cable Configuration		Host/Device Port Capabilities
	Plug-A Role	Plug-B Role	
DRD	DFP	UFP	DRD
DFP	UFP	DFP	DRD
DRD	DFP	UFP	DFP
DFP	Billboard		DFP
Any	Billboard if possible		UFP
UFP	Billboard if possible		Any

6.3.5.1 OIAC Connection Flow – Discovery – Phase 1

The OIAC cable plugs discover the capabilities of their port partners in the Discovery Phase.

Figure 6-24 OIAC Discovery – Phase 1



6.3.5.2 OIAC Connection Flow – Reboot – Phase 2

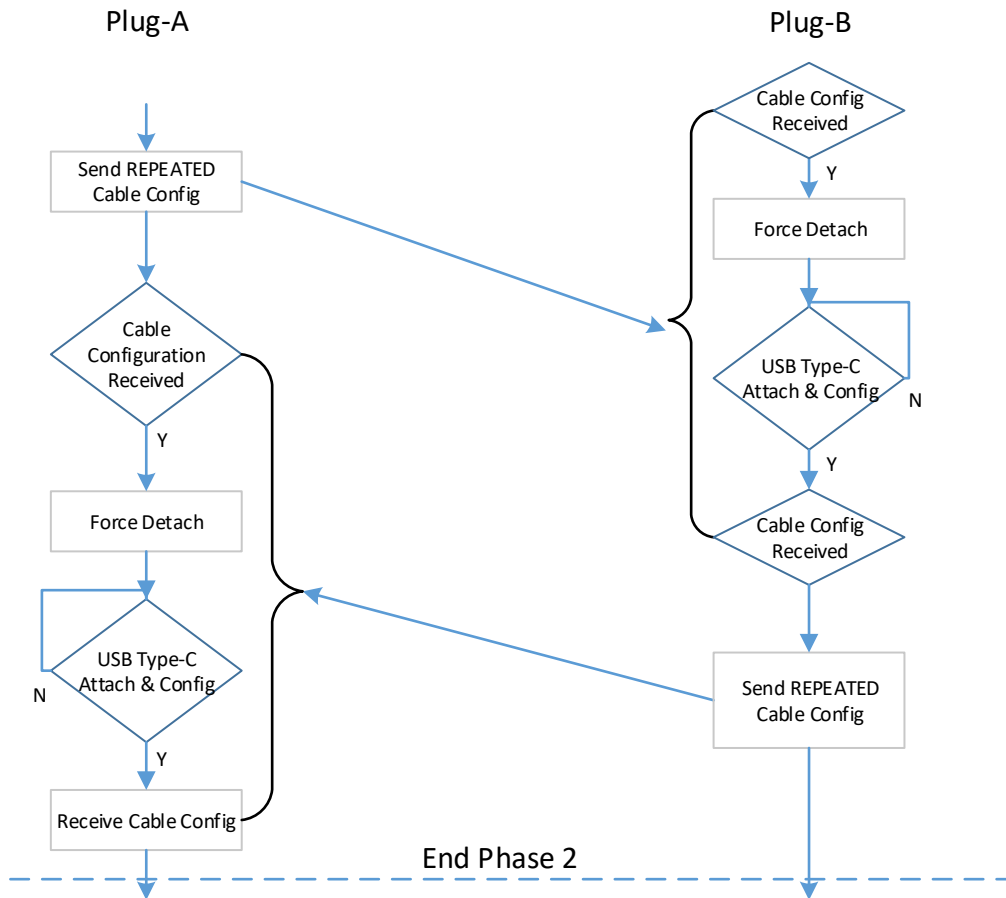
The OIAC cable plugs forward the capabilities of their plug partners and perform disconnect/reconnect.

- Plug-A will always start by repeatedly sending “MSG_Cable_Config” to Plug-B to start reboot.
- Plug-B Reboots.

- Plug-B sees “MSG_Cable_Config” and holds off on SourceCap GoodCRC until it is allowed to release it.
- Plug-B sends “MSG_Cable_Config” to Plug-A.
- Plug-A Reboots.
- Plug-A sees “MSG_Cable_Config” and holds off on SourceCap GoodCRC until it is allowed to release it.

Figure 6-25 OIAC Reboot - Phase 2

Phase 2 – Reboot Both Ends + Share Info



6.3.5.3 OIAC Connection Flow – Configuration – Phase 3

The OIAC Plug-A determines DFP/UEP roles for Plug-A and Plug-B. Plug-A releases the PLUG to be configured as the DFP and initiates a **DR_Swap**. The side that issues the **DR_Swap** send a **DR_Swap** and releases the other side SourceCap GoodCRC.

Figure 6-26 OIAC Plug-A Configured as DFP – Phase 3

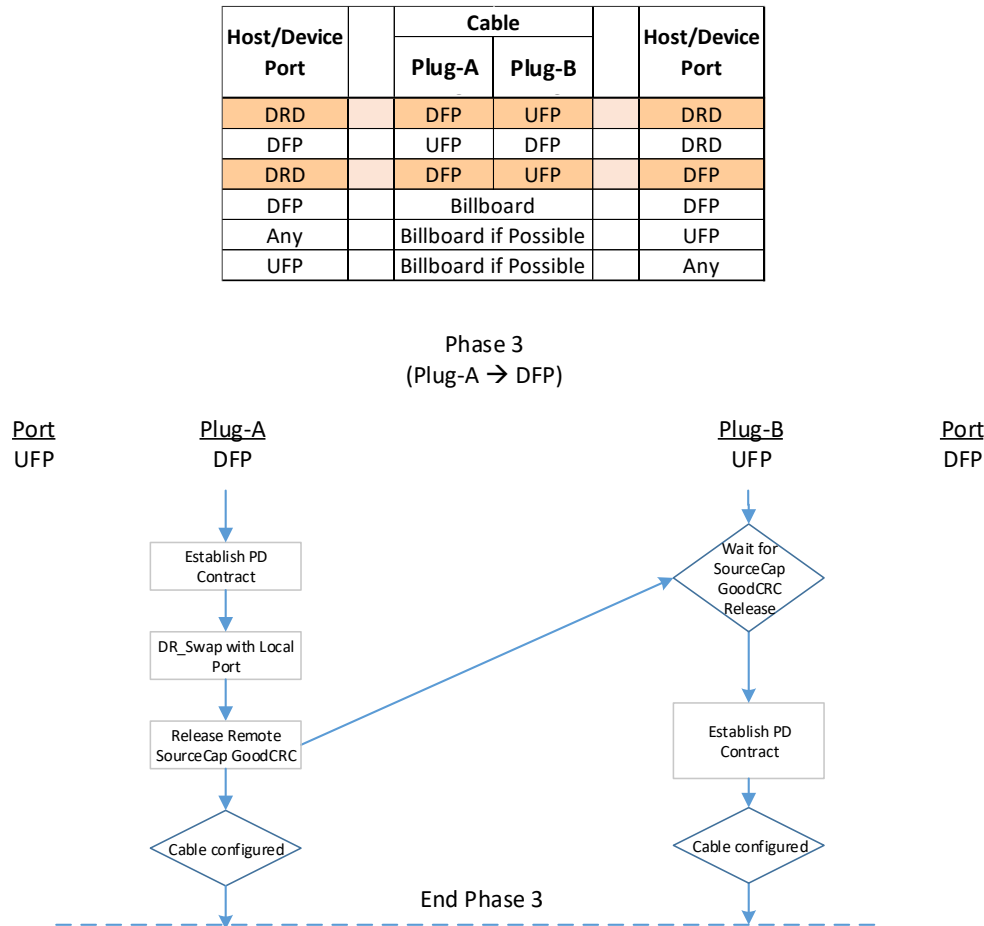


Figure 6-27 OIAC Plug-A Configured as UFP – Phase 3

Host/Device Port	Cable		Host/Device Port
	Plug-A	Plug-B	
DRD	DFP	UFP	DRD
DFP	UFP	DFP	DRD
DRD	DFP	UFP	DFP
DFP	Billboard		DFP
Any	Billboard if Possible		UFP
UFP	Billboard if Possible		Any

Phase 3
(Plug-A → UFP)

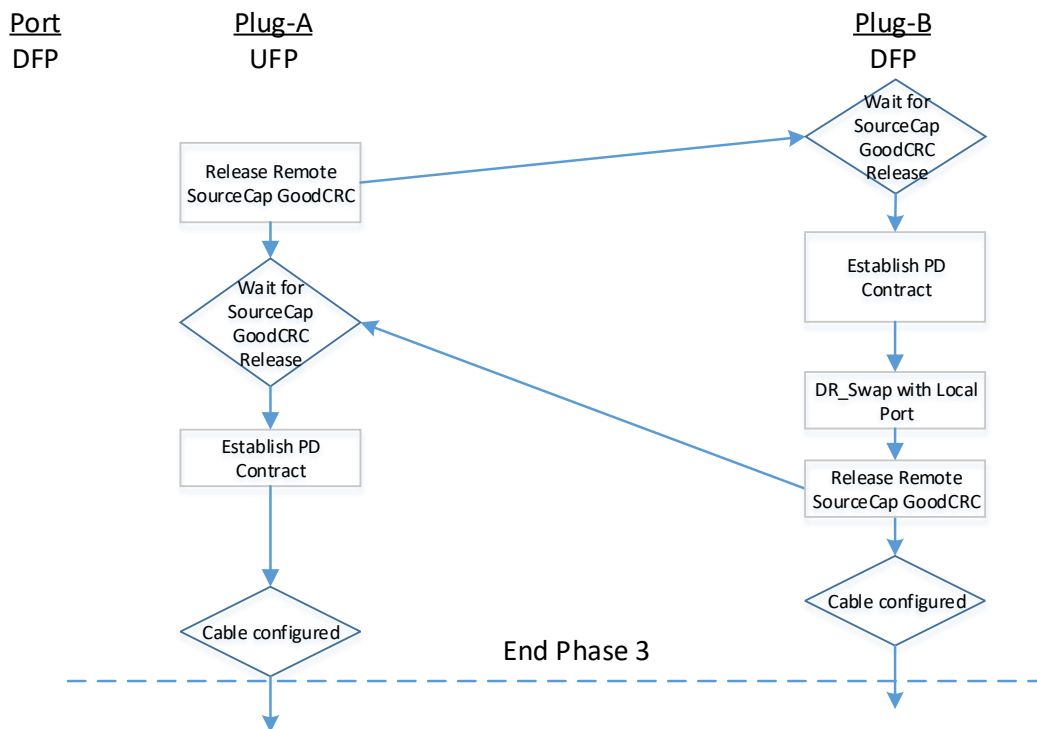
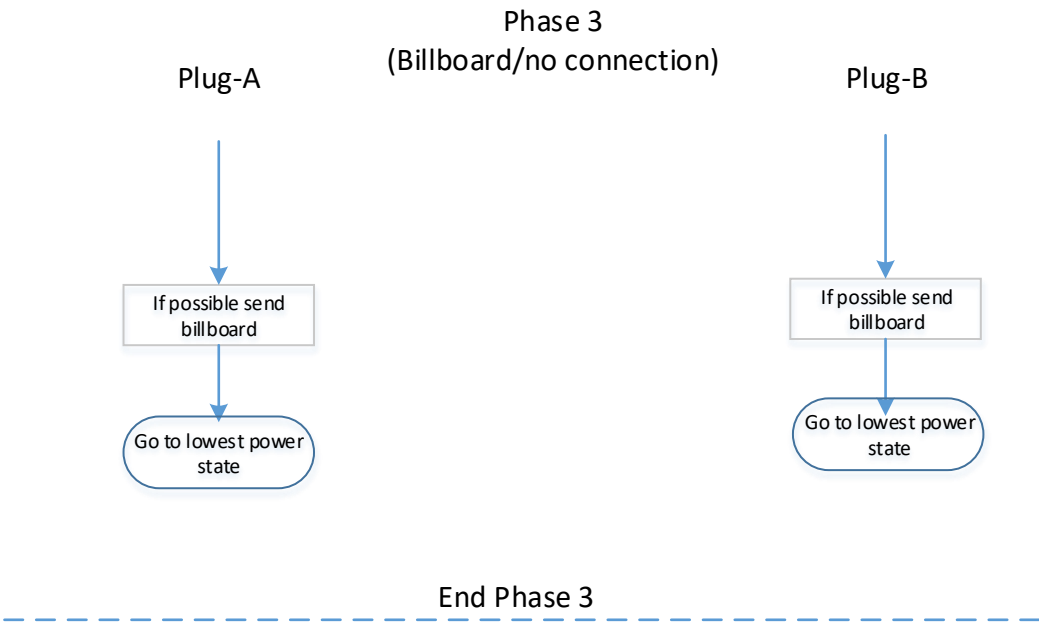


Figure 6-28 OIAC Plug-A No Connection Possible Billboard – Phase 3

Host/Device Port		Cable			Host/Device Port
		Plug-A	Plug-B		
DRD		DFP	UFP		DRD
DFP		UFP	DFP		DRD
DRD		DFP	UFP		DFP
DFP		Billboard			DFP
Any		Billboard if Possible			UFP
UFP		Billboard if Possible			Any



6.3.5.4 OIAC Connection State Diagram Plug-A

The following sections detail a possible OIAC state diagram for Plug-A.

Figure 6-29 OIAC Plug-A State Diagram Part 1 (Phase 1 and 2)

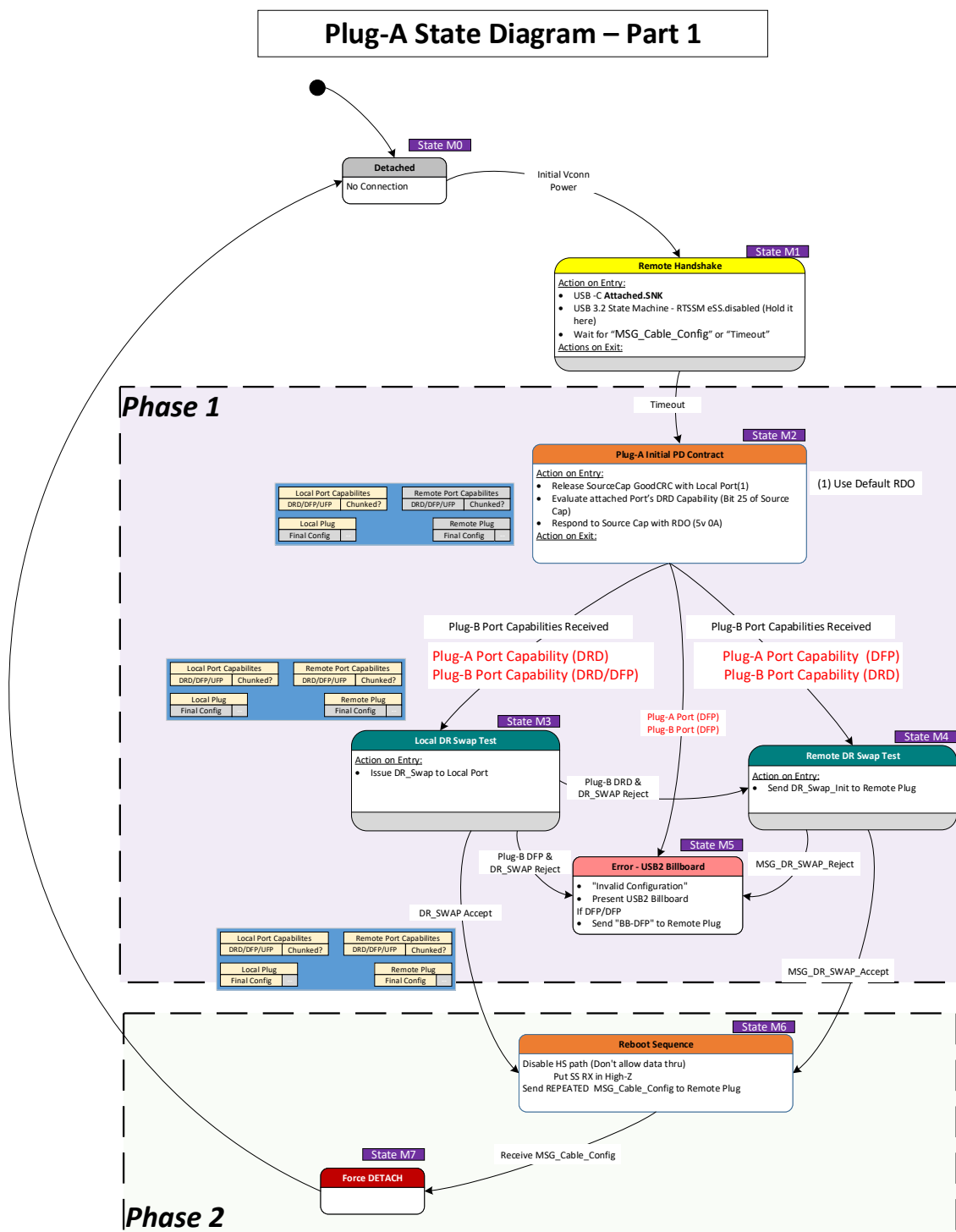
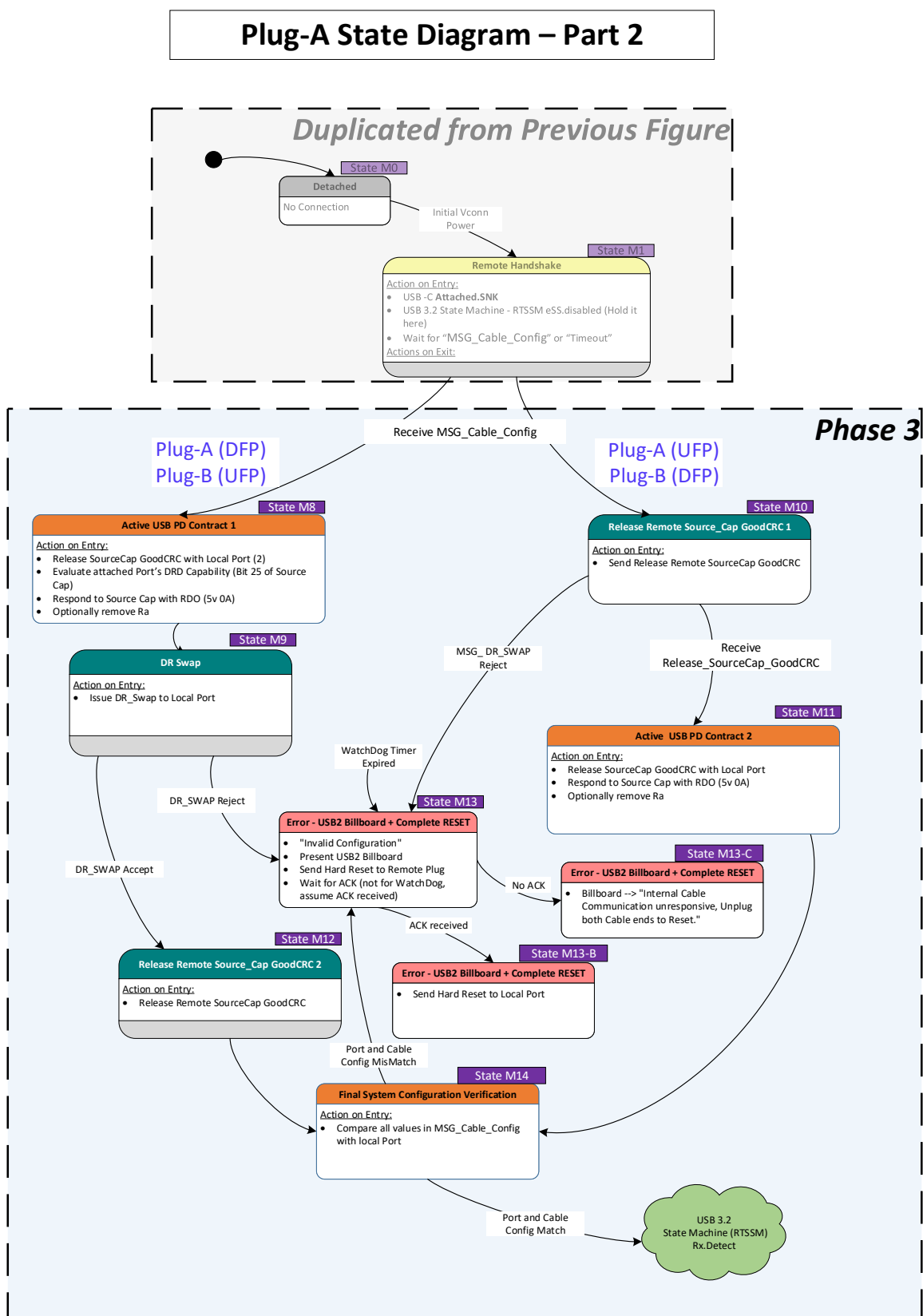


Figure 6-30 OIAC Plug-A State Diagram Part 2 (Phase 3)



6.3.5.4.1 Detached State (M0)

The plug is in **Detached** (M0) when no power is applied. The plug transitions to **Remote Handshake** (M1) when VCONN is applied.

6.3.5.4.2 Remote Handshake State (M1)

The plug is waiting for the “Timeout” timer expiration or a “MSG_Cable_Config” message from Plug-B.

Recommended Timeout time = ~100ms

The Timeout time is dependent on the duty cycle of Plug-B’s Repeat Port Capabilities messages and the maximum cable latency.

The plug starts in the **USB 3.2** RTSSM eSS.Disabled and remains in eSS.Disabled until cable initialization is complete at the end of Phase 3.

6.3.5.4.2.1 Entry to Remote Handshake

The plug enters **Attached.SNK** followed by entry to **Remote Handshake** (M1).

6.3.5.4.2.2 Exit from Remote Handshake

The plug transitions to:

- **Plug-A Initial PD Contract** (M2) when the “Timeout” timer has expired,
- **Active USB PD Contract 1** (M8) upon receipt of a “MSG_Cable_Config” message when Plug-A resolves to a DFP and Plug-B will resolve to a UFP, or
- **Release Remote Source_Cap GoodCRC 1** (M10) upon receipt of a “MSG_Cable_Config” message when Plug-A resolves to a UFP and Plug-B will resolve to a DFP.

6.3.5.4.3 Plug-A Initial PD Contract State (M2)

When the plug is in **Plug-A Initial PD Contract**, the plug has established an initial **USB PD** contract and evaluated its local port’s DRD capability.

6.3.5.4.3.1 Entry to Plug-A Initial PD Contract State

On Entry to **Plug-A Initial PD Contract**, the plug **shall**:

1. Send a GoodCRC in response to a Source Capabilities message from the local attached port
2. Evaluate the local attached port’s DRD capability (as indicated in the Source Capabilities message)
3. Respond to the Source Capabilities message with the “default” RDO specified in Section 6.3.4.3.2.1.2.

6.3.5.4.3.2 Exit from Plug-A Initial PD Contract State

The OIAC plug **shall** transition to:

- **Local DR Swap Test** (M3) upon receipt of a MSG_Port_Capabilities where the Plug-A port is a DRD and the Plug-B port is either a DRD or DFP,
- **Remote DR Swap Test** (M4) upon receipt of a MSG_Port_Capabilities where the Plug-A port is a DFP and the Plug-B port is DRD, or
- **Error – USB2 Billboard** (M5) upon determination both Plug-A and Plug-B are connected to DFPs.

6.3.5.4.4 Local DR Swap Test State (M3)

The **Local DR Swap Test** is a test to ensure that the Plug-A port that is defined as a DRD will accept a **DR_Swap** request.

6.3.5.4.4.1 Entry to Local DR Swap Test State

On entry to **Local DR Swap Test**, the plug **shall** issue **DR_Swap** request to its local port.

If the local port responds to the DR Swap with “**Wait**,” then the plug **shall** follow the tDRSwapWait timer and retry up to 3 times, after which it will error out and transition to state **Error – USB2 Billboard** (M5).

6.3.5.4.4.2 Exit from Local DR Swap Test State

The OIAC plug **shall** transition to:

- **Remote DR Swap Test** (M4) upon receipt of a **Reject** and the Plug-B port reported that it is a DRD,
- **Error – USB2 Billboard** (M5) upon receipt of a **Reject** and the Plug-B port reported that it is a DFP, or
- **Reboot Sequence** (M6) upon receipt of an **Accept**.

6.3.5.4.5 Remote DR Swap Test State (M4)

The **Remote DR Swap Test** is a test to ensure that the Plug-B port that is defined as a DRD will accept a **DR_Swap** request.

6.3.5.4.5.1 Entry to Remote DR Swap Test State

On entry to **Remote DR Swap Test**, the plug **shall** issue a **DR_Swap_Init** to the remote plug.

6.3.5.4.5.2 Exit from Remote DR Swap Test State

The OIAC plug **shall** transition to:

- **Reboot Sequence** (M6) upon receipt of a **MSG_DR_Swap_Accept**, or
- **Error – USB2 Billboard** (M5) upon receipt of a **MSG_DR_Swap_Reject**.

6.3.5.4.6 Error – USB2 Billboard (M5)

The plug presents a Billboard indicating an Invalid Configuration is present. For example: “Error: A DFP only device connected to one of the plugs.”

6.3.5.4.6.1 Entry to Error – USB2 Billboard

On entry **Error – USB2 Billboard**, the plug **shall** issue present a Billboard message over **USB 2.0** and then power down to its lowest possible state.

6.3.5.4.6.2 Exit from Error – USB2 Billboard

The only means of exiting this Error state, is either from a Reset that disconnects VCONN power or a disconnect event which also disconnects VCONN power.

6.3.5.4.7 Reboot Sequence State (M6)

When the plug is in the **Reboot Sequence**, it will disable the High Speed Data path and start to initiate a remote plug reboot.

6.3.5.4.7.1 Entry to Reboot Sequence State

On entry to Reboot Sequence, the plug **shall**:

- 1) Disable the HS path by changing the SS RX termination to High-Z.

- 2) Determine and store the final System Configuration for this link.
 - a. The System Configuration will contain:
 - i. Host/Device Port information
 - ii. Final Plug-A/Plug-B roles
 1. If coming from State M3
 - a. Plug-A → DFP (*DR_Swap*)
 - b. Plug-B → UFP
 2. If coming from State M4
 - a. Plug-A → UFP
 - b. Plug-B → DFP (*DR_Swap*)
 3. If coming from State M9
 - a. Plug-A → UFP
 - b. Plug-B → DFP (*DR_Swap*)
- 3) Continuously send “MSG_Cable_Config” message to the remote plug.

6.3.5.4.7.2 Exit from Reboot Sequence State

The OIAC plug *shall* transition to the **Force Detach** (M7) when a “MSG_Cable_Config” message is received that matches the final configuration that Plug-A sent to Plug-B.

6.3.5.4.8 Force Detach State (M7)

The plug *shall* transition to *SRC.Open* on both CC and VCONN and maintain this state for at least *tSRCDisconnect*.

6.3.5.4.8.1 Entry to Force Detach State

On entry to **Force Detach**, the plug *shall* raise the voltage on the CC-wire above *vRd-ConnectConnected*.

6.3.5.4.8.2 Exit from Force Detach State

The plug transitions to **Detached** upon exit from **Force Detach**.

6.3.5.4.9 Active USB PD Contract 1 State (M8)

Active USB Contract 1 is where OIAC Plug-A creates at **USB PD** contract with the local port.

6.3.5.4.9.1 Entry to USB PD Contract 1 State

The OIAC plug *shall* follow the steps listed below:

1. Begin responding to **USB PD** messages from its port partner with GoodCRCs.
2. Evaluate Fixed 5V PDO in the Source Capabilities message to check if its port partner is a DRD.
3. Request for a 5 V @ 0 A power contract.
4. May remove *Ra* to save power.

6.3.5.4.9.2 Exit from USB PD Contract 1 State

The plug transitions to the **DR Swap** (M9) when it receives an **Accept** followed by a PS_RDY message from its port partner.

6.3.5.4.10 DR Swap State (M9)

The **DR Swap** is used to set the final data role of the OIAC's Plug-A and signal to the OIAC Plug-B to complete its configuration.

6.3.5.4.10.1 Entry to DR Swap State

The OIAC plug **shall** issue a **DR_Swap** to its port partner.

If the local port responds to the **DR_Swap** with “**Wait**,” then the plug **shall** follow the tDRSwapWait timer and retry up to 3 times, after which it will error out and transition to state **Error – USB2 Billboard** (M5).

6.3.5.4.10.2 Exit from DR Swap State

If the **DR_Swap** message is responded to with:

- An **Accept**, it **shall** transition to the **Release Remote Source_Cap GoodCRC 2 State** (M12), or
- A **Reject** it **shall** transition to the **Error - USB2 Billboard + Complete Reset** (M13).

6.3.5.4.11 Release Remote Source_Cap GoodCRC 1 State (M10)

The OIAC is waiting for Release_Remote_Source_Cap_GoodCRC to better synchronize the power on of the two OIAC plug ends.

6.3.5.4.11.1 Entry to Release Remote Source_Cap GoodCRC 1 State

The OIAC plug **shall** release the Remote SourceCap GoodCRC.

6.3.5.4.11.2 Exit from Release Remote Source_Cap GoodCRC 1 State

The OIAC plug **shall** transition to:

- **Active USB PD Contract 2 State** (M11), when a “MSG_ Release_Remote_SourceCap_GoodCRC” message is received, or
- **Error – USB2 Billboard + Complete Reset** (M13) upon receipt of a MSG_DR_Swap_Reject.

6.3.5.4.12 Active USB PD Contract 2 State (M11)

Active USB Contract 2 is where OIAC Plug-A creates at **USB PD** contract with the local port and **should** end with viable link.

6.3.5.4.12.1 Entry to Active USB PD Contract 2 State

The OIAC plug **shall** follow the steps listed below:

1. Begin responding to **USB PD** messages from its port partner with GoodCRCs.
2. Request a 5 V @ 0 A power contract.
3. May remove **C** to save power.

6.3.5.4.12.2 Exit from Active USB PD Contract 2 State

The plug **shall** transition to **Final System Configuration Verification** (M13) for final system verification.

6.3.5.4.13 Release Remote Source_Cap GoodCRC 2 State (M12)

OIAC Source Plug configuration is completed, and the **USB 3.2** begins looking for a connection.

6.3.5.4.13.1 Entry to Release Remote Source_Cap GoodCRC 2 State

The OIAC plug **shall** release the Remote SourceCap GoodCRC.

6.3.5.4.13.2 Exit from Release Remote Source_Cap GoodCRC 2 State

The plug *shall* transition to **Final System Configuration Verification** (M13) for final system verification.

6.3.5.4.14 Error – USB2 Billboard + Complete Reset (M13)

The plug presents a Billboard indicating an Invalid Configuration is present. For example: “Error: An invalid configuration occurred. Full link will be reset.”

6.3.5.4.14.1 Entry to Error – USB2 Billboard + Complete Reset

On entry Error – USB2 Billboard + Complete Reset, the plug *shall*:

- 1) Present a USB2 Billboard message
- 2) Send a MSG_Hard_Reset to the remote Plug
- 3) Wait for Hard Reset Ack from Remote Plug (Unless entry is from an expired WatchDog Timer, in which case, go directly to M13-B)

Received ACK (State M13-B):

- 4) Send Hard Reset to the Local Port

Did NOT receive ACK (State M13-C):

- 5) Present USB2 Billboard Message: “Error: Internal Cable Communication Error. Unplug both cable ends to reset.”

6.3.5.4.14.2 Exit from Error – USB2 Billboard + Complete Reset

The only means of exiting this Error state, is either from a Reset that disconnects VCONN power or a disconnect event which also disconnects VCONN power from each port.

6.3.5.4.14.3 Watchdog Timer Entry

A watchdog timer *should* be implemented for internal cable messages that require a response. The watchdog timer will also provide an entry to an **Error State** (M13) if the far end plug is unresponsive for any reason.

There are a few states where the watchdog timer *shall not* be implemented including but not limited to M2, where it is possible that only a single end of the OIAC is connected and M6, where the reboot sequence can take a few seconds.

6.3.5.4.15 Final System Configuration Verification (M14)

Final System Configuration Verification is used to do one final check there were no unforeseen changes the local port and the final cable configuration defined by Plug-A.

6.3.5.4.15.1 Entry to Final System Configuration Verification

The OIAC plug *shall* check all values in the MSG_Cable_Config match that of the current local port’s configuration.

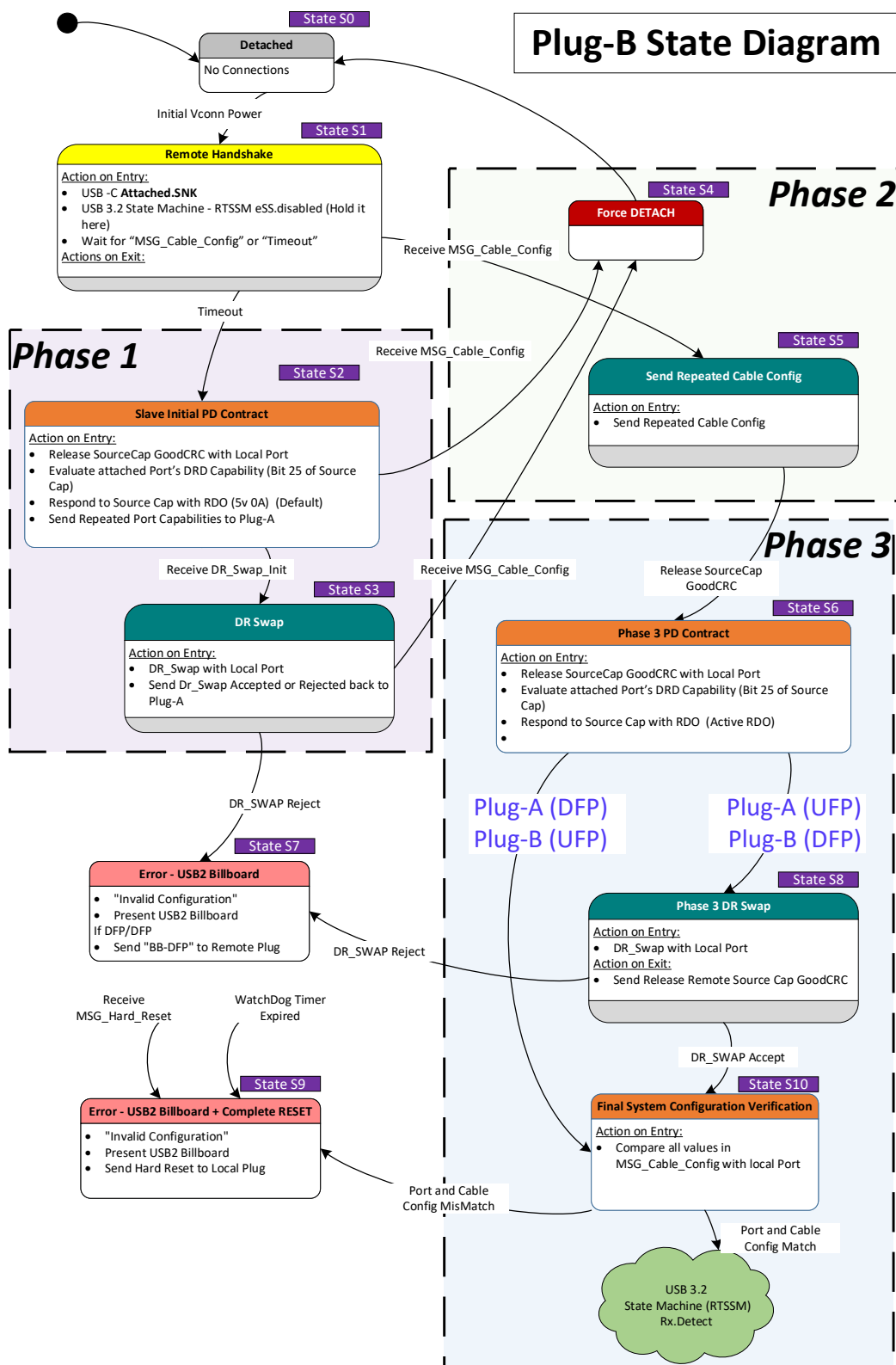
6.3.5.4.15.2 Exit from Final System Configuration Verification

The plug *shall* transition to:

- **Rx.Detect**, and start far-end receiver termination detection and the [USB 3.2](#) RTSSM State Machine after a successful match of the MSG_Cable_Config and the local port’s configuration, or
- **Error - USB2 Billboard + Complete RESET** (State M14) after unsuccessful match of the MSG_Cable_Config and the local port’s configuration.

The following sections detail a possible OIAC state diagram for Plug-B.

Plug-B State Diagram



6.3.5.5.1 Detached State (S0)

The plug is in **Detached** (S0) when no power is applied. The plug transitions to **Remote Handshake** (S1) when VCONN is applied.

6.3.5.5.2 Remote Handshake State (S1)

The plug is waiting for the “Timeout” timer expiration or a “MSG_Cable_Config” message from Plug-A.

Recommended Timeout time = ~100ms

The Timeout time is dependent on the duty cycle of the Plug-A’s Repeat MSG_Cable_Config messages and the maximum cable latency.

6.3.5.5.2.1 Entry to Remote Handshake State

The plug enters **Attached.SNK** followed by entry to **Remote Handshake** (S1).

The plug starts in the **USB 3.2** RTSSM eSS.Disabled and remains in eSS.Disable until cable initialization is complete at the end of Phase 3.

6.3.5.5.2.2 Exit from Remote Handshake State

The plug transitions to:

- **Plug-B Initial PD Contract** (S2) when the “Timeout” timer has expired, or
- **Send Repeated Cable Config** (S5) upon receipt of a “MSG_Cable_Config” message.

6.3.5.5.3 Plug-B Initial PD Contract (S2)

The plug takes the following actions in the order defined:

1. Send a GoodCRC in response to a Source Capabilities message from the local attached port.
2. Evaluate the local attached port’s DRD Capability (as indicated in the Source Capabilities message).
3. Send the initial RDO (5 V/0 A) and negotiate an explicit power contract.
4. Send “MSG_Port_Capabilities” to Plug-A. The plug **shall** continue to send “MSG_Port_Capabilities” until it exits this state.

6.3.5.5.3.1 Entry to Plug-B Initial PD Contract

The plug enters **Plug-B Initial PD Contract** (S2) upon “Timeout” timer expiration.

6.3.5.5.3.2 Exit from Plug-B Initial PD Contract

The plug transitions to:

- **DR Swap** (S3) upon receipt of a “DR_Swap_Init” message from Plug-A, or
- **Force Detach** (S4) upon receipt of a “MSG_Cable_Config” message from Plug-A.

6.3.5.5.4 DR Swap (S3)

The plug determines if a **DR_Swap** can be performed with the local attached port or Plug-A configured itself.

6.3.5.5.4.1 Entry to DR Swap

Upon entry into DR Swap (S3) the plug **shall**:

1. Send **DR_Swap** message to the local attached port.
2. Send MSG_DR_Swap_Accept when the plug receives **Accept** message from the local attached port.

3. Send MSG_DR_Swap_Reject when the plug receives **Reject** message from the local attached port.

If the local port responds to the DR Swap with “**Wait**,” then the plug **shall** follow the tDRSwapWait timer and retry up to 3 times, after which it will error out and transition to state **Error – USB2 Billboard** (S7).

6.3.5.5.4.2 Exit from DR Swap

The plug **shall** transition to:

- **Error – USB2 Billboard** (S7) upon rejection of the local attached port **DR_Swap**, or
- **Force Detach** (S4) upon receipt of a “MSG_Cable_Config” message from Plug-A.

6.3.5.5.5 Force Detach (S4)

The plug **shall** transition to **SRC.Open** on both CC and VCONN and maintain this state for at least **tSRCDisconnect**.

6.3.5.5.5.1 Entry to Force Detach

The plug enters **Force Detach** (S4) upon receipt of a “MSG_Cable_Config” message from Plug-A.

6.3.5.5.5.2 Exit from Force Detach

The plug transitions to **Detached** (S0) upon exit from **Force Detach**.

6.3.5.5.6 Send Repeated Cable Config (S5)

The plug repeatedly sends “MSG_Cable_Config” messages in this state to inform Plug-A of the configuration of the local attached port.

6.3.5.5.6.1 Entry to Send Repeated Cable Config

The plug enters **Repeated Cable Config** (S5) upon receipt of a “MSG_Cable_Config” message from Plug-A.

6.3.5.5.6.2 Exit from Send Repeated Cable Config

The plug transitions to **Phase 3 PD Contract** (S6) upon receipt of a “Release SourceCap GoodCRC” message from the Plug-A port.

6.3.5.5.7 Phase 3 PD Contract (S6)

The plug performs the following actions in this state:

1. Send a GoodCRC in response to a Source Capabilities message from the local attached port.
2. Evaluate the attached local attached port’s DRD Capability (as indicated in the Source Capabilities message).
3. Send the initial RDO (5 V/0 A) in response to the Source Capabilities message and negotiate an explicit power contract.

6.3.5.5.7.1 Entry to Phase 3 PD Contract

The plug enters **Phase 3 PD Contract** upon receipt of a “Release SourceCap GoodCRC” message from Plug-A.

6.3.5.5.7.2 Exit from Phase 3 PD Contract

The plug transitions to:

- **Final System Configuration Verification** (S10) for final system verification upon determination Plug-A is acting as a DFP and it is acting as a UFP, or
- **Phase 3 DR Swap** (S8) upon determination Plug-A is acting as a UFP and it is acting as a DFP.

6.3.5.5.8 Error – USB2 Billboard (S7)

The plug presents a Billboard indicating an Invalid Configuration is present. For example: “Error: A DFP only device connected to one of the plugs.”

6.3.5.5.8.1 Entry to Error – USB2 Billboard

The plug transitions to this state upon rejection of a *DR_Swap* by the local attached port.

6.3.5.5.8.2 Exit from Error – USB2 Billboard

The only means of exiting this Error state, is either from a Reset that disconnects VCONN power or a disconnect event which also disconnects VCONN power.

6.3.5.5.9 Phase 3 DR Swap (S8)

The plug issues a *DR_Swap* with its local attached port in this state.

6.3.5.5.9.1 Entry to Phase 3 DR Swap

The plug enters *Phase 3 DR Swap* upon determination Plug-A is connected as a UFP and Plug-B *should* connect as a DFP.

6.3.5.5.9.2 Exit from Phase 3 DR Swap

The plug *shall* transition to:

- *Final System Configuration Verification* (S10) for final system verification after successful completion of the *DR_Swap* with the local attached port, or
- *Error – USB2 Billboard* (S7) after unsuccessful completion of the *DR_Swap* with the local attached port.

6.3.5.5.10 Error – USB2 Billboard + Complete RESET (S9)

The plug presents a Billboard indicating an Invalid Configuration is present. For example: “Error: An invalid configuration occurred. Full link will be reset.”

6.3.5.5.10.1 Entry to Error – USB2 Billboard + Complete RESET

On entry *Error – USB2 Billboard + Complete RESET*, the plug *shall*:

1. Present a USB2 Billboard message.
2. Send Hard_Reset to the Local Plug.

6.3.5.5.10.2 Exit from Error – USB2 Billboard + Complete RESET

The only means of exiting this Error state, is either from a Reset that disconnects VCONN power or a disconnect event which also disconnects VCONN power.

6.3.5.5.10.3 Watchdog Timer Entry

A watchdog timer *should* be implemented for internal cable messages that require a response. The watchdog timer will also provide an entry to an *Error State* (M13) if the far end plug is unresponsive for any reason.

There are a few states where the watchdog timer *shall not* be implemented including but not limited to S5, where the reboot sequence can take a few seconds.

6.3.5.5.11 Final System Configuration Verification (S10)

The *Final System Configuration Verification* is used to one final check there were no unforeseen changes to the local port and the final cable configuration defined by Plug-A.

6.3.5.5.11.1 Entry to Final System Configuration Verification

The OIAC plug **shall** check all values in the MSG_Cable_Config match that of the current local port's configuration.

6.3.5.5.11.2 Exit from Final System Configuration Verification

The plug **shall** transition to:

- **Rx.Detect**, and start far-end receiver termination detection and the **USB 3.2** RTSSM State Machine after a successful match of the MSG_Cable_Config and the local port's configuration, or
- **Error – USB2 Billboard + Complete RESET** (State S9) after unsuccessful match of the MSG_Cable_Config and the local port's configuration.

6.3.6 Additional Electrical Requirements for OIAC

6.3.6.1 Shielding Effectiveness Requirement

OIACs **shall** meet the shielding effectiveness requirement defined in Section 3.7.7 and Figure 3-66.

6.3.6.2 Low Speed Signal Requirements

6.3.6.2.1 CC Channel Requirements

OIACs **shall** meet the Low-Speed Signal Requirements in Section 3.7.2.6.

6.3.6.2.2 SBU Requirements

6.3.6.2.2.1 USB 3.2

OIACs are not required to support SBU1/2 for **USB 3.2** support.

6.3.6.2.2.2 Alternate Modes

OIACs **may** support SBUs for **Alternate Modes**. SBUs are not usable until the cable has entered an **Alternate Mode**. OIACs which support SBU signals **shall** meet the requirements of the **Alternate Mode(s)** supported. Definition of the SBU requirements for **Alternate Modes** is outside the scope of this specification..

6.3.6.2.2.3 USB4

OIACs **shall** support SBTX and SBRX for **USB4** support. No additional support for SBU1/2 signaling is required.

OIAC SBU wires operating as **USB4** SBTX and SBRX **shall** meet the same voltage swing requirements as Active Cables as defined in Table 6-3.

6.3.6.3 USB 2.0

OIACs are not required to support **USB 2.0**, therefore **USB 2.0** is not expected to work across an OIAC.

The OIAC will take action to reset the link when the USB Device drops from **USB 3.2** to **USB 2.0**.

During the initial connection the OIAC **shall** present as a **USB 2.0** DFP and provide a 15K Ohm pull down on the D+/D- pins on both ends of the cable. The cable plug **shall not** issue a **USB 2.0** Reset in this state.

The OIAC cable plug **shall** issue a **USB 2.0** Reset upon detecting a **USB 2.0** connection on D+/D- (LS, FS, or HS **USB 2.0** connection). The cable plug **shall** issue a **USB 2.0** Bus Reset by pulling D+ and D- low for at least 50 ms.

The OIAC **shall** implement a tDisableCount counter to determine how many times the cable has transitioned from **USB 3.2** to **USB 2.0**. The tDisableCount counter **shall** be reset to zero on either condition:

- Power on Reset of the OIAC, or
- Successful transition to **USB 3.2** U0.

The OIAC **shall** present and latch a **USB 2.0** billboard when the tDisableCount counter reaches three.

6.3.6.4 **USB 3.2**

OIAC **shall** meet the requirements in this section regardless of length. OIACs **shall** incorporate AC-coupling from the plug to repeater on both the **USB 3.2** TX and RX signals. OIACs **shall** provide a discharge path for discharging the AC-coupling capacitors in the cable on unplug per **USB 3.2**.

6.3.6.4.1 **USB 3.2 Power-on and Rx.Detect**

OIAC **shall** have the same power-on and far-end receiver termination as defined for the Active Cable defined in Section 6.1.2.4.2.

OIACs **shall** reflect the receiver terminations across the cable to replicate the behavior of a passive cable.

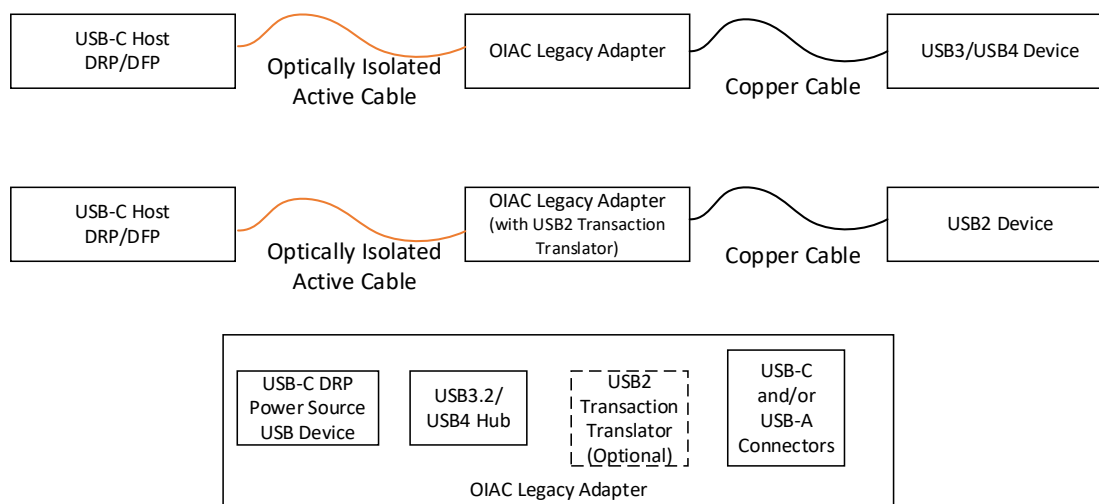
6.3.6.4.2 **OIAC Legacy Adapter**

The OIAC Legacy Adapter is specified here to provide a way to connect to a **USB 3.2** or **USB4** device that does not support OIACs. This adapter will need to be in place between the OIAC and the device.

The OIAC Adapter requires the following capabilities:

1. USB Type-C DRP/Source (Self-Powered) on the upward facing port connected to the OIAC,
2. Self-Powered
 - a. Needs to power the upstream OIAC
 - b. Needs to be able to power the downstream device
3. **USB PD** on the upward facing port connected to the OIAC,
4. **USB 3.2** Hub
5. **Optional: USB4** Hub
6. **Optional: USB 2.0** to **USB 3.2** transaction translator on the downstream facing port (only needed if connecting to **USB 2.0**-only devices, hubs, or hosts).

Figure 6-32 Illustration of Usages for OIAC that Require an Adapter or Hub



6.3.6.4.3 USB 3.2 U-State Power Requirements

OIACs **shall** meet the VCONN power requirements for **USB 3.2** operation in Table 6-28. These requirements are for the entire cable not just a cable plug.

Table 6-28 USB 3.2 U-State Requirements

State	Maximum VCONN Power Consumption	Power Consumption Notes
U0	1.0 W single-lane 1.5 W dual-lane	Applies to POLLING.LFPS, TRAINING, and RECOVERY states.
U1	≤ U0 power	Forwarding LFPS is required
U2	≤ U0 power	Forwarding LFPS is required
U3	≤ U0 power	Steady state power. eMarker in sleep.
Rx.Detect	≤ U0 power	Rx.Detect period may be lengthened when no USB 3.2 terminations have been detected. eMarker in sleep.
eSS.Disabled	≤ U0 power	USB 3.2 is disabled. eMarker in sleep.

6.3.6.4.4 USB 3.2 U-State Exit Latency

USB 3.2 U-State Exit Latency is TBD.

6.3.6.4.5 Signal Swing

OIACs **shall** meet the same electrical characteristics for **USB 3.2** Signal Swing as an Active Cable defined in Section 6.1.2.4.6.

6.3.6.5 USB4

The Optically Isolated Active Cable for **USB4** is defined as a Linear Optical Active Cable. As such, the electrical LRD requirements in Section 6.1.2.5.4 will generally apply to a **USB4** OIAC.

6.3.6.5.1 AC Coupling Capacitors

OIAC **shall** include AC-coupling capacitance between 135 nF and 265 nF inside their plugs placed at the output transmit path and between 300 nF and 363 nF at the input receive path. Discharge resistors between 200 K Ω and 242 K Ω **shall** be placed at the input receive path. See Figure 3-3 in the **USB4** specification for a diagram of the AC-coupling capacitors and discharge resistors.

6.3.6.5.2 USB4 CL-State Power Requirements for OIAC

OIACs are not required to meet the same CL-State power requirements as defined for Active Cables (Section 6.1.2.5.2).

OIACs **shall** meet the V_{CONN} power requirements for **USB4** operation in Table 6-29. These requirements are for the entire cable, not just a cable plug.

Table 6-29 USB4 CL-State Requirements for OIAC

State	Maximum V _{CONN} Power Consumption	Power Consumption Notes
CL0	1.5 W	Applies to training states and CL0.
CL0s	≤ CL0 power	
CL1	≤ CL0 power	
CL2	≤ CL0 power	
CLd	≤ CL0 power	Steady state power.

6.3.6.6 Return Loss

Return loss is defined in the **USB 3.2** specification.

6.3.7 Additional Mechanical Requirements for OIAC

The mechanical specifications for OIACs are the same as all other Active Cables (see Section 6.1.3). Additional requirements for OIACs are provided in this section.

6.3.7.1 Thermal

6.3.7.1.1 Thermal Shutdown

OIACs **shall** billboard in shutdown to both ports, regardless of which end of the cable plug is in thermal shutdown. For example: “Error: The Optical Cable has experienced a thermal shutdown.”

6.3.7.1.2 Thermal Reporting

The OIAC **shall** implement reporting their maximum internal operating temperature in the **USB PD Discover_ID** Command.

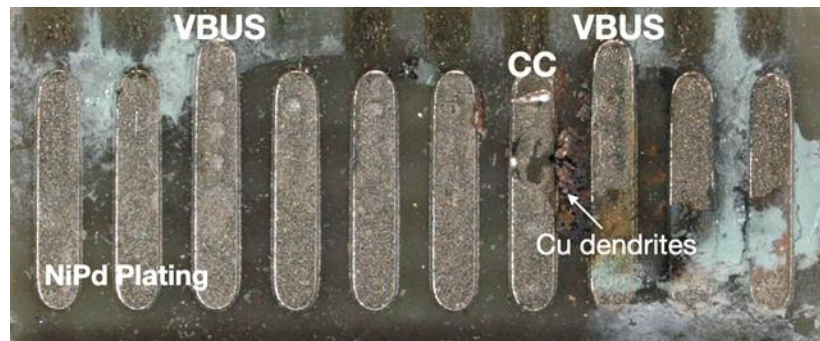
The OIAC **shall** report their current internal operating temperature (of the higher of the two cable plugs) in the **USB PD Get_Status** Command on SOP’.

A Liquid Corrosion Mitigation Mode

A.1 Overview

Ingress of every day liquids with electrolytes (sweat, soda, tap water, salt water, pool water, etc.) into the USB Type-C connector can easily corrode the connector pins when exposed to the liquid with electrical bias in the connector (VBUS, CC, etc.). The liquid forms a bridge between pins and when the voltage difference between the pins is high enough (above a few hundred millivolts), the exposed metal will dissolve into free ions and these ions will migrate from one pin to another. When these dissolved ions move and form oxides, they can lead to open or high impedance due to surface residue or partial metal pin material loss or they can lead to shorts due to various mechanisms like dendrite growth or metal-based particle depositions on plastic surface.

Figure A-1 Electrolytic Liquid Corrosion Example



When in a connected state, VBUS is 4.75 V – 50.4 V and CC/VCONN is 0.2 V – 5.5 V. Likewise, in an unconnected state, a Source or DRP will pull the CC pin to 3 V – 5.5 V. This results in always-present voltage bias that will corrode pins when exposed to electrolytic liquids.

The **Liquid Corrosion Mitigation Mode** is an optional normative mode that eliminates or reduces the rate of corrosion by removing or forcing the bias voltages in the connector to a voltage as close to 0 V as possible.

A.2 Detail

If a liquid detection scheme is employed by a device with a USB Type-C port, the port state can be directed to the **CorrosionMitigation** state by first going through **ErrorRecovery**. In the **CorrosionMitigation** state, the port shall pull low with a resistance on both CC1 and CC2 less than or equal to the Maximum Impedance of **Ra** as specified in Table 4-29. The port should pull low with a resistance stronger than **Ra** to pull the voltage as low as possible. The lower the voltage, the lower the rate of corrosion. Figure A-2 and Figure A-3 show the Corrosion Mitigation pull-down resistance for any port in the **CorrosionMitigation** state when connected to a Source. Alternatively, for solutions with the ability to only pull one pin low at a time, this resistance can be applied to only one of the CC pins while the other CC pin is kept open. Figure A-4 shows the pull-down resistance and open with this method. When this alternative approach is used, the voltage on the open CC pin shall be monitored to detect the presence of a pull-up from an attached Source (CC is **SNK.Rp**) due to a plug event with a cable or device in the opposite orientation. When the pull-up is detected, the Sink shall swap the pull-down and open between the CC pins.

Figure A-2 Corrosion Mitigation in Pull-Up/Pull-Down CC Model

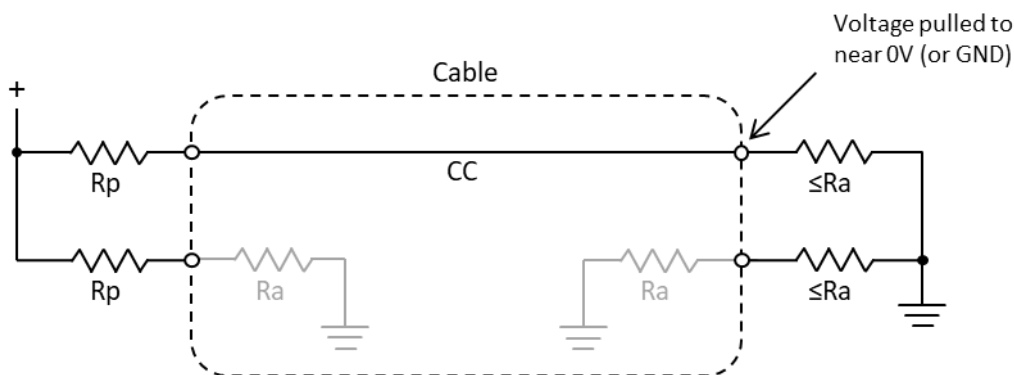


Figure A-3 Corrosion Mitigation in Current Source/Pull-Down CC Model

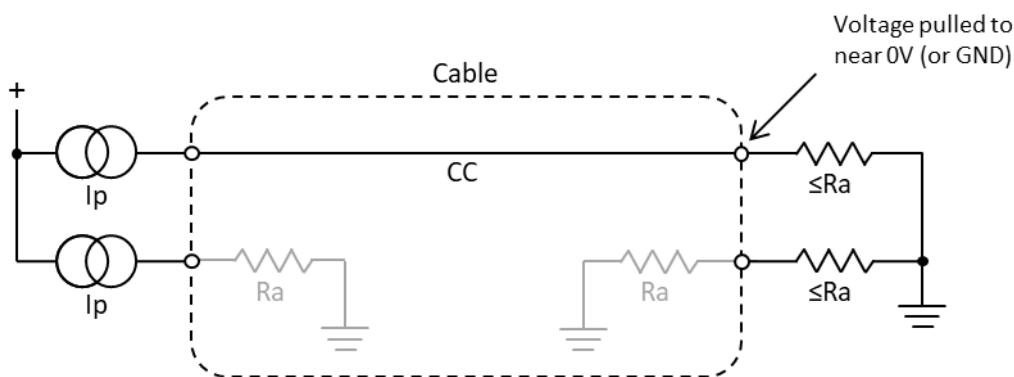
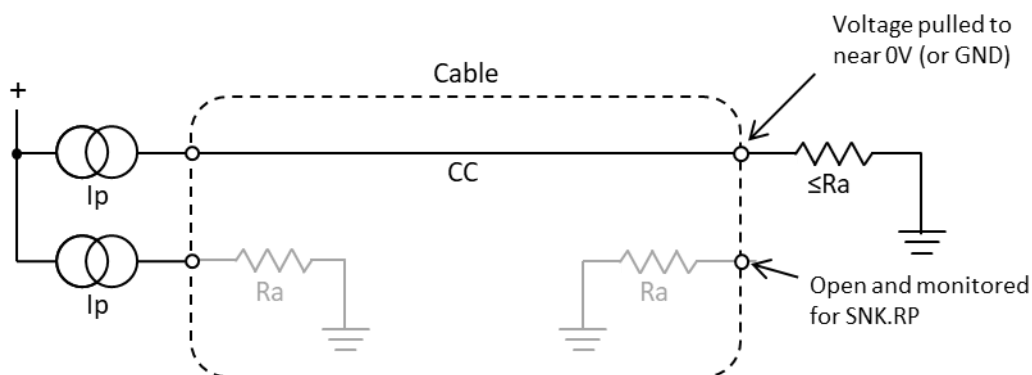


Figure A-4 Corrosion Mitigation in Current Source/Pull-Down CC Model with Alternate Mitigation



When a Source or DRP connects to a port in the **CorrosionMitigation** state, it will move to the **AttachWait.SRC** state (two **Ra** detected), stay in the **Unattached.SRC** state (Source with single **Ra** detected), or continue DRP toggling (DRP with single **Ra** detected) until the **CorrosionMitigation** state is exited by the port partner. In these states, VBUS and VCONN are at 0 V, and the CC pin will be pulled low by the attached resistance of the port partner. All other pins should be open or pulled low. This will hold all voltages in the port low enough to stop or greatly reduce the rate of corrosion. Once a liquid condition is no longer detected, the port should be directed to exit the **CorrosionMitigation** state.

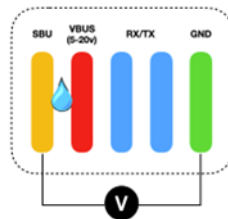
A.3 Liquid Detection Methods

Multiple methods exist to detect liquid in a port. Three possible methods are described, but this is not an exhaustive list. Other methods may be developed and used.

A.3.1 Liquid Measurement Method

Liquid may be detected by measuring the effect of leakage through the liquid between pins. The voltage on a pin can be measured to determine if there is a connection through this liquid to another pin. Figure A-5 shows an example of this measurement using one of the SBU pins. When the port is dry, the measurement pin should be biased to a known level based on the bias circuitry chosen. When the port is wet, DC voltages or signals may couple on the measurement pin which can be read by comparator or ADC. To use the method, various liquids should be characterized with the planned connector (metal, plating, spacing, etc.). Note, this method has low complexity, but may have only moderate sensitivity (accuracy of detection) to liquids resulting in some missed detections based on the amount of and type of liquid. This method may also have a high rate of missed detects due to the variation of the voltages to which the leakage may be connected (i.e., VBUS and CC). This method may also produce higher levels of false detections (other contaminants, high levels of coupling, etc.).

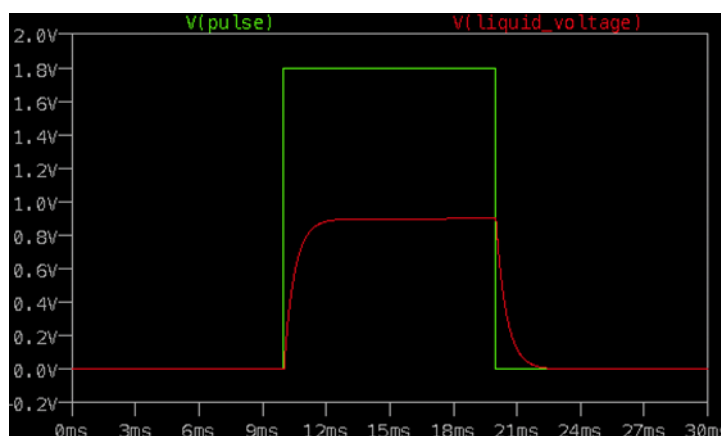
Figure A-5 Liquid Detection by Leakage Measurement



A.3.2 Pulsed Measurement Method

Liquid may be detected by measuring the effect of driving a pulse onto the measurement pin through a series resistance. The rise and fall behavior of the pin can be measured with a timer and comparator to determine the time it takes to cross a certain threshold. Figure A-6 shows an example of this measurement. When the port is dry, the pin voltage will pass the threshold quickly. When the port is wet, loading from the liquid and adjacent pins will cause a slower voltage rise or fall. Like the leakage measurement method, to use the pulsed measurement method, various liquids should be characterized with the planned connector (metal, plating, spacing, etc.). Note, this method is slightly more complex, but can also have only moderate sensitivity to liquids. Some missed detections based on the amount of, and type of liquid may still occur. This method may also miss detections due to other pulsed signals within the port like DRP toggling. This method may also produce false detections but at a reduced rate from the leakage measurement method.

Figure A-6 Liquid Detection by Pulsed Measurement



A.3.3 Impedance Measurement Method

Liquid may be detected by measuring the impedance of a measurement pin. This can be done by various methods. One example method is an I-V measurement that measures both the amplitude and phase shift from a known “dry” condition. Wet or Dry conditions can be determined by the measurement impedance characteristics of the port with the liquid. Like the previous measurement methods, to use the pulsed measurement method, various liquids should be characterized with the planned connector (metal, plating, spacing, etc.). Note, this method is much more complex, but has higher sensitivity and lower false detections. Figure A-7 through Figure A-11 show examples of this measurement on various liquids taken with a Keysight E4990A impedance analyzer sweeping from 20Hz to 1MHz. Table A-1 shows the conditions for the measurements shown.

Table A-1 Example Measurement Test Conditions

Item	Model	Description
Connector	JAE DX07S024JJ2R1300	USB Type-C receptacle, 24 pin SMD
Liquid Dispense	Various	0.5-10 μ L micropipette
Equipment	Keysight E4990A	Impedance Analyzer
Artificial Sweat	1700-0020	Pickering Laboratories Artificial Eccrine Perspiration Stabilized

Figure A-7 Example Measurement of a Dry Port

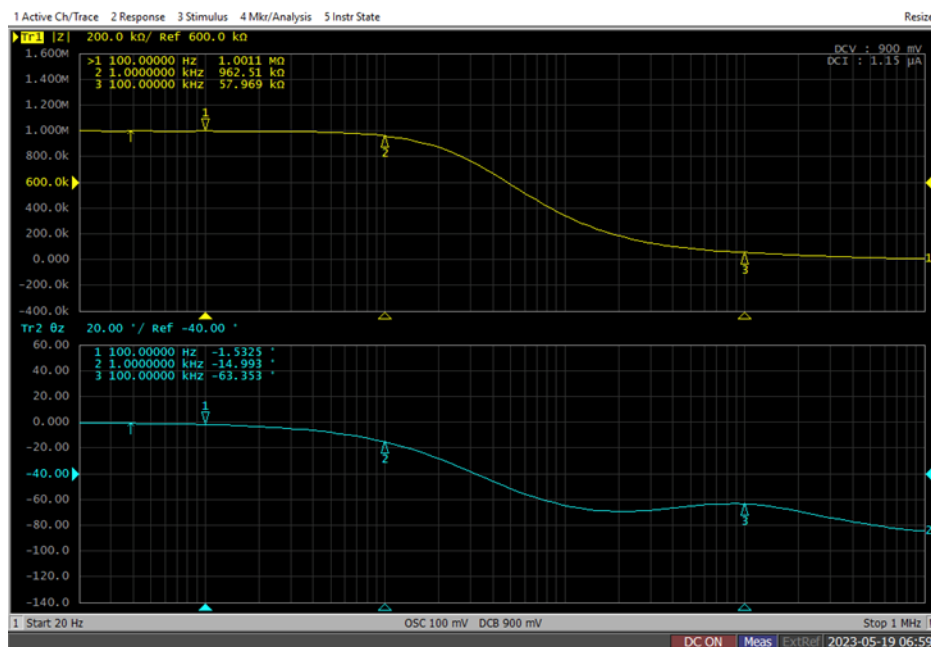


Figure A-8 Example Measurement of a Port with Reverse Osmosis Water

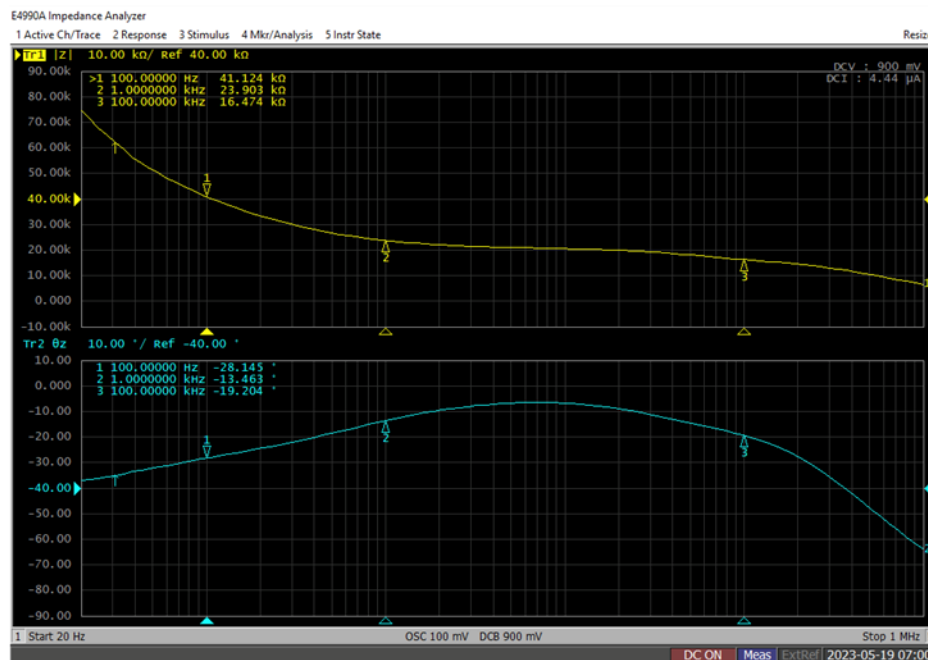


Figure A-9 Example Measurement of a Port with Tap Water

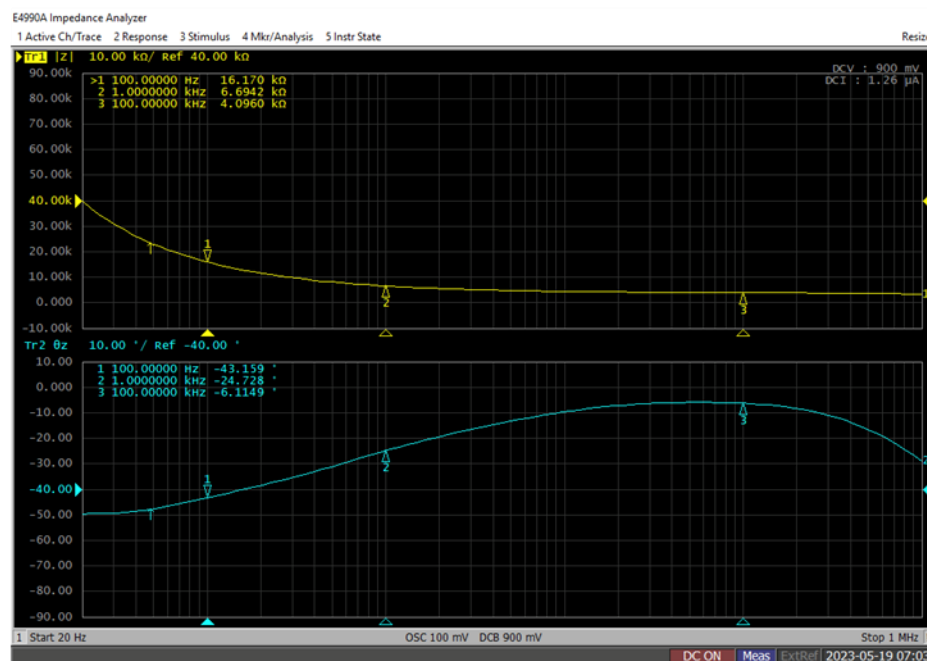


Figure A-10 Example Measurement of a Port with Artificial Sweat

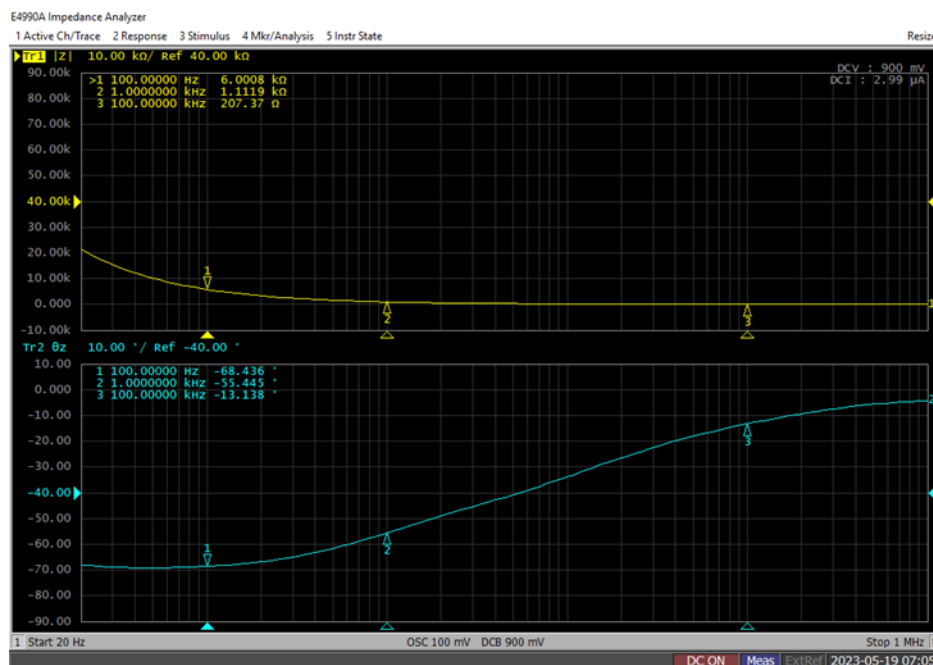
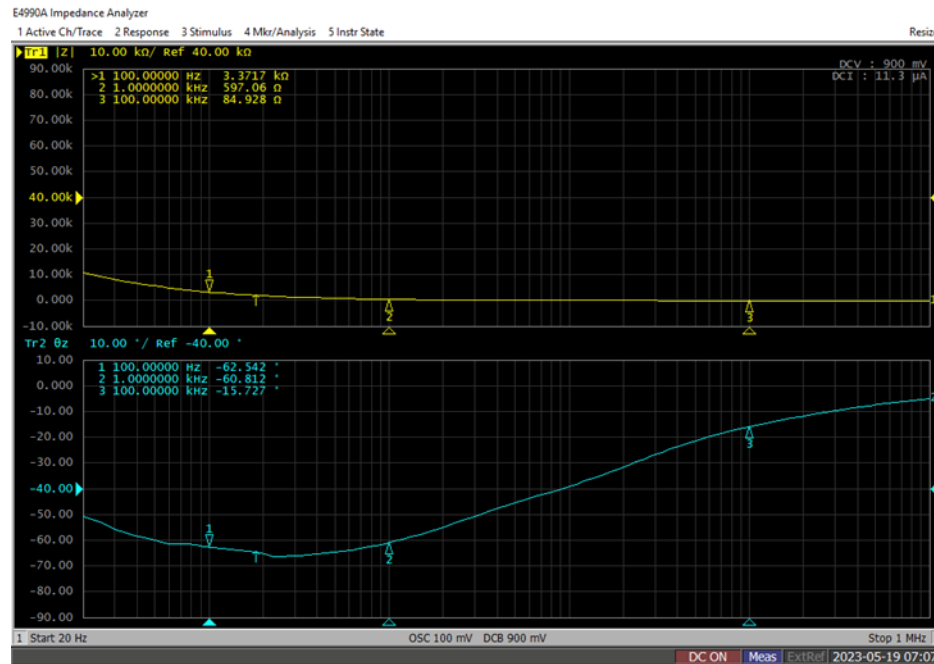


Figure A-11 Example Measurement of a Port with Ocean Water



A.4 Liquid Detection Pins in the Connector

Liquid may be detected by utilizing unused pin(s) in the connector or by adding dedicated pin(s)/pad(s) into the connector.

Key options and design areas that need attention.

1. When adding pin(s)/pad(s) to the connector, requirements found in Section 3.2.1 shall still be met.
2. This specification does not define standard locations for liquid detection pin(s)/pad(s). Figure A-12 shows a reference area within a receptacle where dedicated liquid detection (LD) pin(s)/pad(s) may be added. This region is between the normative pin mating area and the EMC pad. Utilizing this area for liquid detection pin(s)/pad(s) avoids interfering with either the normative plug pins or the EMC springs. It is recommended that the connector designer define the liquid detection pin(s)/pad(s) tolerances such that these pin(s)/pad(s) stay in the liquid detect zone in worse case tolerance.
3. Figure A-13 shows a reference receptacle with a liquid detection pad along the full width of the receptacle pins. Note, the reference receptacle shown is a PCB tongue with the ground mating locations on the metal frame encapsulating the edge.
4. Figure A-14 shows a reference receptacle with dedicated liquid detection pins between each VBUS and the adjacent CC and SBU pins. In this example, the liquid detection pins have been placed where typical corrosion damage is observed and where the connector is functionally most sensitive to shorts (impedance reduction) due to corrosion. Figure A-15 shows a view of an example surface mount footprint with the added pins. This can be applied to a vertical-mount, right-angle mount, or mid-mount receptacle.
5. Placement of the liquid detection pad or pins can affect the signal performance of the high-speed signaling path. The resulting shape and spacing of the high-speed pins should be analyzed to ensure it meets the signaling requirements of the high-speed protocol being used in the design. Interference of the high-speed signals from the liquid detection pin(s)/pad(s) due to the liquid detection method should also be analyzed.

Figure A-12 Example Liquid Detect Pin(s)/Pad(s) Location Area

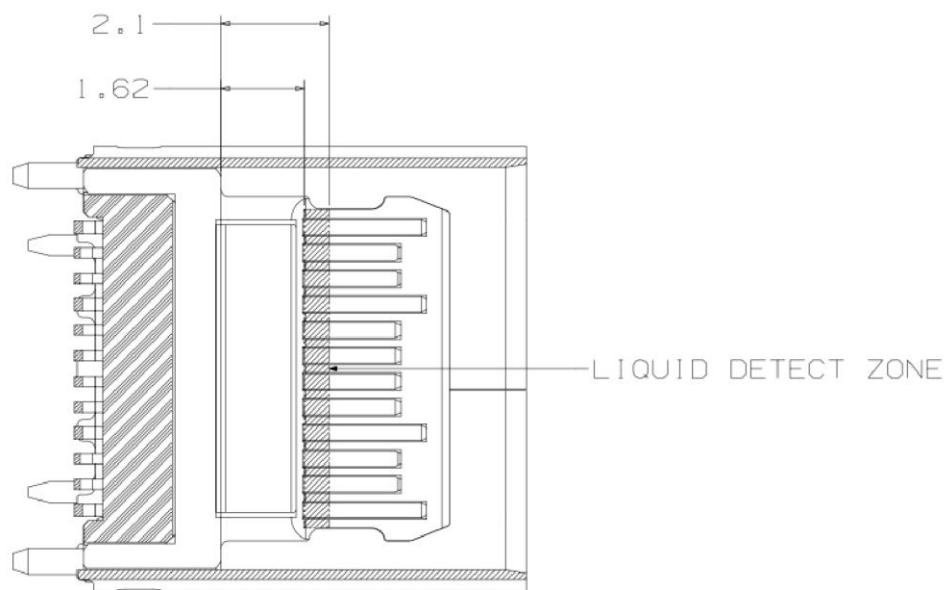


Figure A-13 Example Liquid Detect Pad Along All Connector Pins

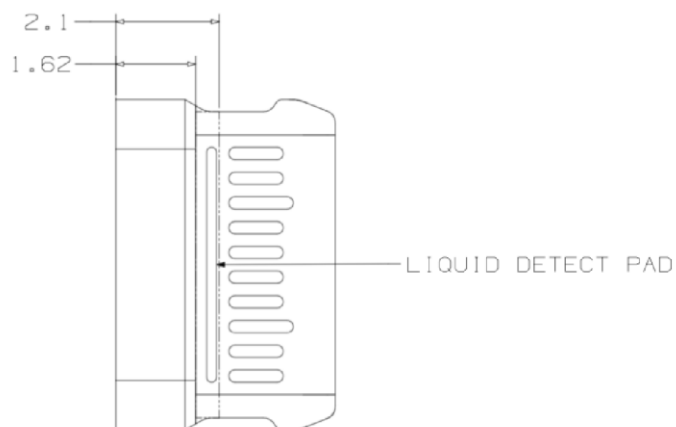


Figure A-14 Example Liquid Detect Pins Adjacent to VBUS/SBU/CC

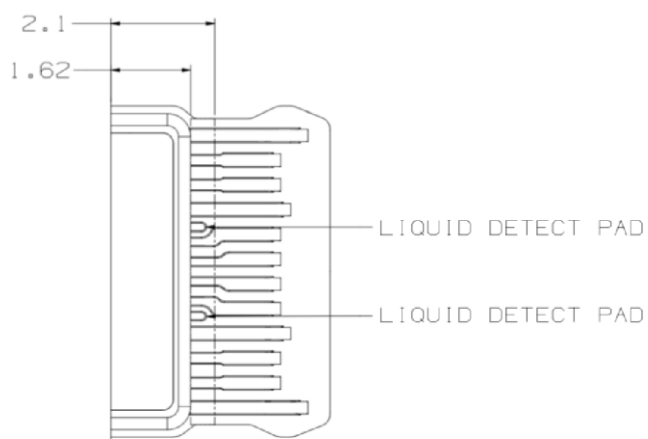
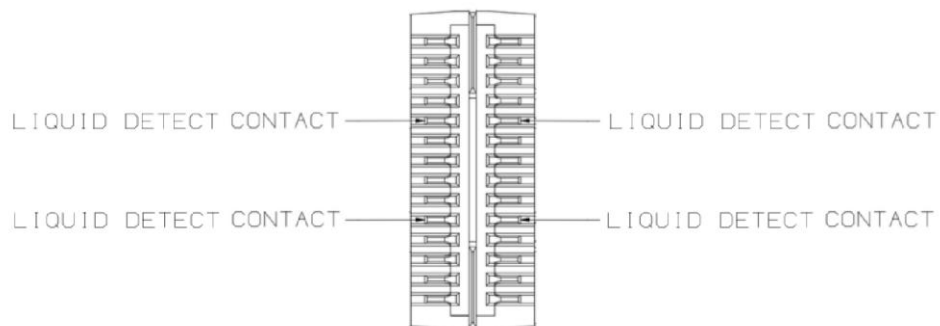


Figure A-15 Example Liquid Detect Connector Surface Mount View



B Debug Accessory Mode

B.1 Overview

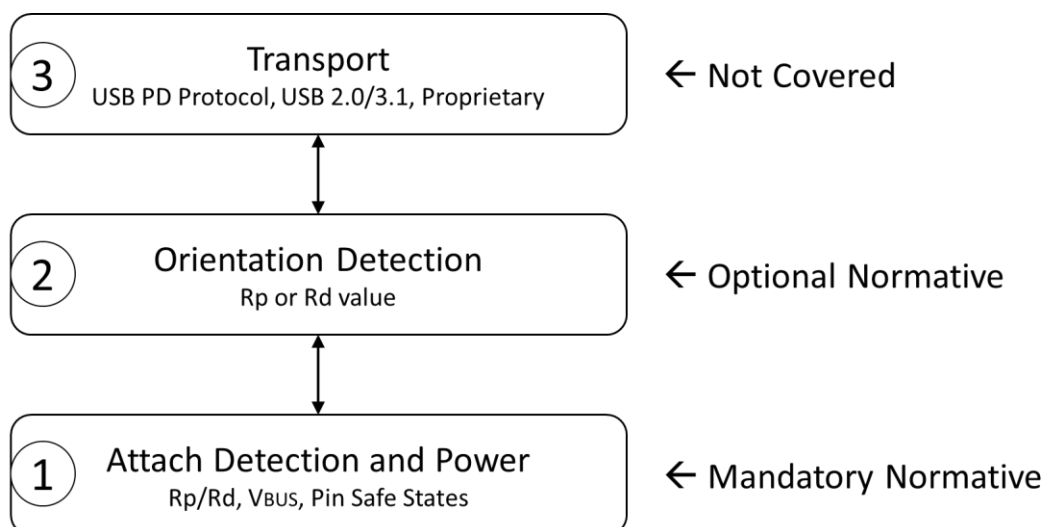
This appendix covers the functional requirements for the USB Type-C **Debug Accessory Mode (DAM)**, **Debug and Test System (DTS)**, and **Target System (TS)**. The USB Type-C connector is ideal for the debug of closed-chassis, form-factor devices. Debug covers many areas, ranging from detailed JTAG Test Access Port (TAP)-level debug in a lab to high-level debug of software applications in production. Lab debug requires early debug access to hardware registers soon after reset, whereas software debug uses kernel debuggers, etc. to access software state. **Debug Accessory Mode** in USB Type-C enables debug of closed-chassis, form-factor devices by re-defining the USB Type-C ports for debug purposes.

Basic debug requirements are defined as a standard feature, and additional debug features **may** be added as per vendor specifications.

B.2 Functional

The USB Type-C **Debug Accessory Mode** follows a layered structure as shown in Figure B-1, defining the minimum physical layer for Attach, Detection and Power. Orientation detection is **optional normative**. The transport layer is left proprietary and is not covered in this document.

Figure B-1 USB Type-C Debug Accessory Layered Behavior



B.2.1 Signal Summary

Figure B-2 shows the pin assignments of the DTS plug that are used to support DAM. The pins highlighted in yellow are those available to be configured for debug signals. Both CC1 and CC2 are used for current advertisement and **optional** orientation detection.

Figure B-2 DTS Plug Interface

A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
GND	RX2+	RX2-	VBUS	SBU1	D-	D+	CC1	VBUS	TX1-	TX1+	GND
GND	TX2+	TX2-	VBUS	CC2	D+	D-	SBU2	VBUS	RX1-	RX1+	GND
B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12

The DTS and TS must follow the USB Safe State as defined in the **USB PD** specification at all times (whether in DAM or not).

B.2.2 Port Interoperability

Table B-1 summarizes the expected results when interconnecting a DTS Source, Sink or DRP port to a TS Source, Sink or DRP port.

Table B-1 DTS to TS Port Interoperability

	DTS Source	DTS Sink	DTS DRP
TS Sink	Functional	Non-Functional ¹	Functional
TS Sink w/ Accessory Support	Functional	Non-Functional ¹	Functional
TS DRP	Functional	Functional	Functional
TS Source	Non-Functional ¹	Functional	Functional
Note 1: In the cases where no function results, neither port <i>shall</i> be harmed by this connection. Following the USB Safe State ensures this.			

B.2.3 Debug Accessory Mode Entry

The typical flow for the configuration of the interface in the general case of a DTS to a TS is as follows:

1. Detect a valid connection between the DTS (Source, Sink, or DRP) and TS (Source, Sink, or DRP)
2. **Optionally** determine orientation of the plug in the receptacle
3. **Optionally** establish **USB PD** communication over CC for advanced power delivery negotiation and **Alternate Modes**. **USB PD** communication is allowed only if the **optional** orientation of the plug is determined.
4. Establish test access connections with the available USB Type-C signals

The DTS DRP will connect as either a Source or a Sink, but its state diagram gives preference to the Source role.

B.2.3.1 Detecting a Valid DTS-to-TS Connection

The general concept for setting up a valid connection between a DTS and TS is based on being able to detect the typical USB Type-C termination resistances. However, detecting a **Debug Accessory Mode** connection requires that both CC pins must detect a pull-up (**Rp**) or pull-down (**Rd**) termination. A USB Type-C Cable does not pass both CC wires so a receptacle to receptacle **Debug Accessory Mode** connection cannot be detected.

A DTS is only allowed to connect to a TS that is presenting either **Rp/Rp** or **Rd/Rd**. Otherwise, the TS does not support **Debug Accessory Mode**.

To detect either an **Rp/Rp** or **Rd/Rd**, the DTS must be a captive cable or a direct-attach device with a USB Type-C plug and the TS must have a USB Type-C receptacle.

B.2.4 Connection State Diagrams

This section provides reference connection state diagrams for CC-based behaviors of the DTS. The TS connection state diagrams are found in Section 4.5.2.

Refer to Section B.2.4.1 for the specific state transition requirements related to each state shown in the diagrams.

Refer to Section B.2.4.3 for a description of which states are mandatory for each port type and a list of states where **USB PD** communication is permitted.

Figure B-3 illustrates a connection state diagram for a DTS Source.

Figure B-3 Connection State Diagram: DTS Source

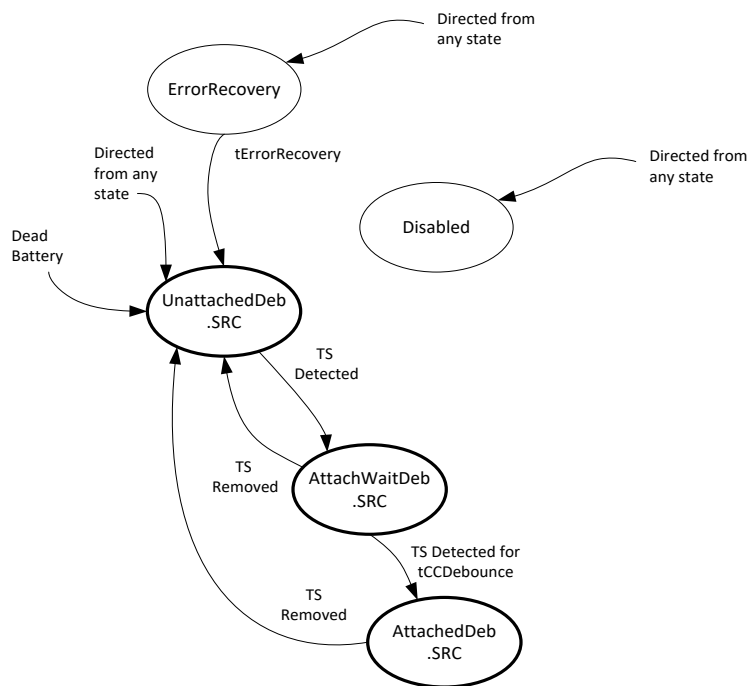


Figure B-4 illustrates a connection state diagram for a simple DTS Sink.

Figure B-4 Connection State Diagram: DTS Sink

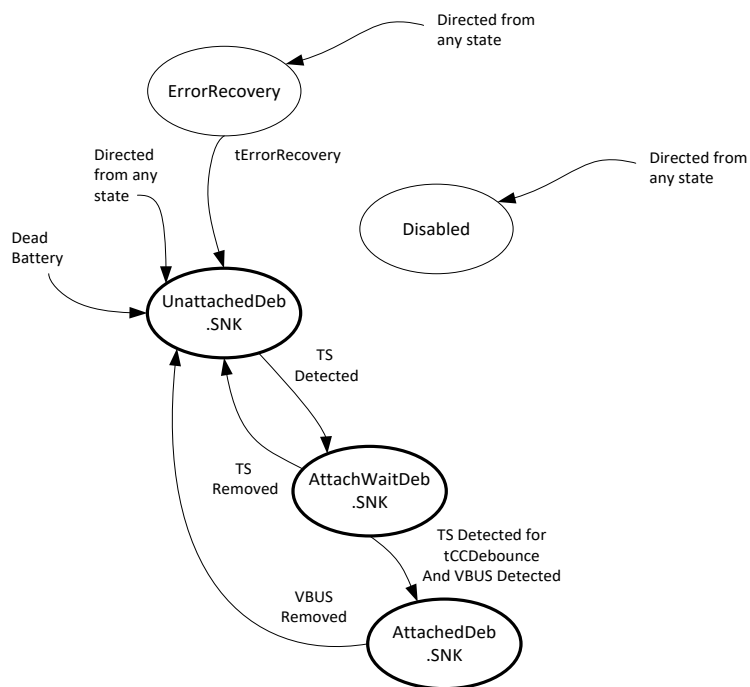
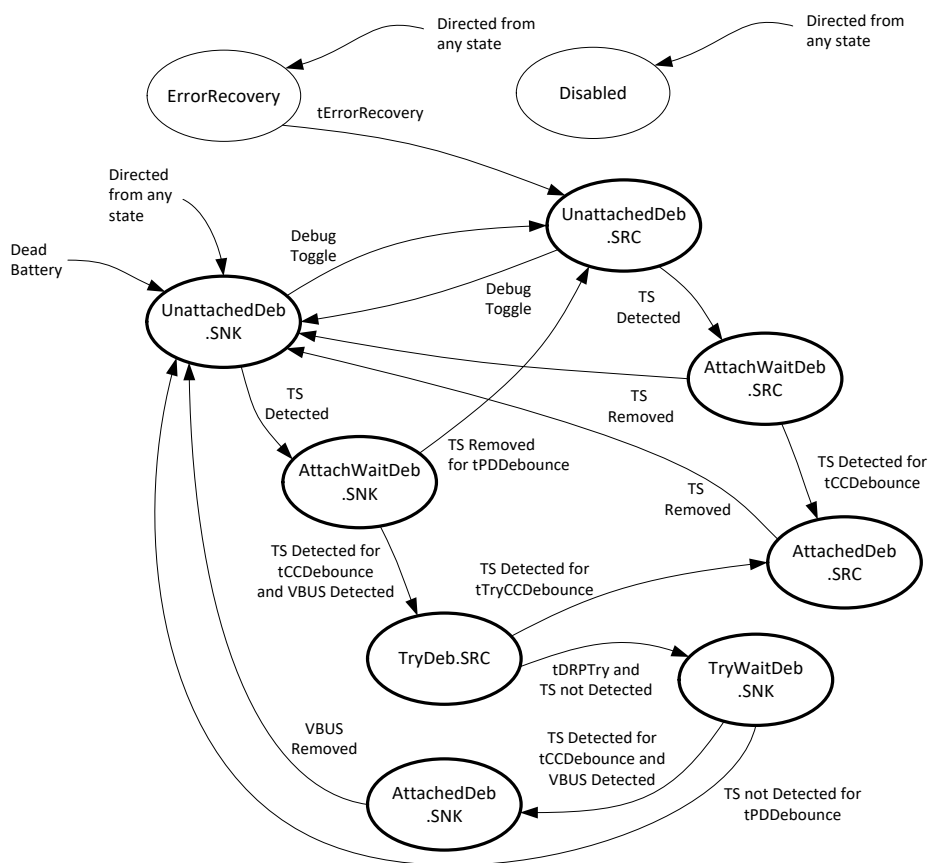


Figure B-5 illustrates a connection state diagram for a DTS DRP.

Figure B-5 Connection State Diagram: DTS DRP



B.2.4.1 Connection State Machine Requirements

The DTS state machine requirements follow those outlined in Section 4.5.2.2 for the general USB Type-C state machines with the additional following states defined.

Note, VCONN **shall not** be driven by any DTS or TS port in any state.

B.2.4.1.1 ErrorRecovery State

This state appears in Figure B-3, Figure B-4, and Figure B-5.

The **ErrorRecovery** state is where the DTS cycles its connection by removing all terminations from the CC pins for **tErrorRecovery** followed by transitioning to the appropriate **UnattachedDeb.SNK** or **UnattachedDeb.SRC** state based on DTS type.

The DTS **should** transition to the **ErrorRecovery** state from any other state when directed.

A DTS **may** choose not to support the **ErrorRecovery** state. If the **ErrorRecovery** state is not supported, the DTS **shall** be directed to the **Disabled** state if supported. If the **Disabled** state is not supported, the DTS **shall** be directed to either the **UnattachedDeb.SNK** or **UnattachedDeb.SRC** states.

A DTS Sink **shall** transition to **UnattachedDeb.SNK** after **tErrorRecovery**.

A DTS Source **shall** transition to **UnattachedDeb.SRC** after **tErrorRecovery**.

A DTS DRP **shall** transition to *UnattachedDeb.SRC* after *tErrorRecovery*.

B.2.4.1.2 UnattachedDeb.SNK State

This state appears in Figure B-4 and Figure B-5.

When in the *UnattachedDeb.SNK* state, the DTS is waiting to detect the presence of a TS Source.

A DTS with a dead battery **shall** enter this state while unpowered.

B.2.4.1.2.1 UnattachedDeb.SNK Requirements

The DTS **shall not** drive VBUS.

Both CC pins **shall** be independently terminated to ground through *Rd*.

B.2.4.1.2.2 Exiting from UnattachedDeb.SNK State

The DTS **shall** transition to *AttachWaitDeb.SNK* when a TS Source connection is detected, as indicated by the *SNK.Rp* state on both of its CC pins.

A DTS DRP **shall** transition to *UnattachedDeb.SRC* within *tDRPTransition* after the state of one or both CC pins is *SNK.Open* for $tDRP - dcSRC.DRP \cdot tDRP$, or if directed.

B.2.4.1.3 AttachWaitDeb.SNK State

This state appears in Figure B-4 and Figure B-5.

When in the *AttachWaitDeb.SNK* state, the DTS has detected the *SNK.Rp* state on both CC pins and is waiting for VBUS.

B.2.4.1.3.1 AttachWaitDeb.SNK Requirements

The requirements for this state are identical to *UnattachedDeb.SNK*.

B.2.4.1.3.2 Exiting from AttachWaitDeb.SNK State

A DTS Sink **shall** transition to *UnattachedDeb.SNK* when the state of one or both CC pins is *SNK.Open* for at least *tPDDebounce*.

A DTS DRP **shall** transition to *UnattachedDeb.SRC* when the state of one or both CC pins is *SNK.Open* for at least *tPDDebounce*.

A DTS Sink **shall** transition to *AttachedDeb.SNK* when neither CC pin is *SNK.Open* after *tCCDebounce* and VBUS is detected.

A DTS DRP **shall** transition to *TryDeb.SRC* when neither CC pin is *SNK.Open* after *tCCDebounce* and VBUS is detected.

B.2.4.1.4 AttachedDeb.SNK State

This state appears in Figure B-4 and Figure B-5.

When in the *AttachedDeb.SNK* state, the DTS is attached and operating as a DTS Sink.

B.2.4.1.4.1 AttachedDeb.SNK State Requirements

This mode is for debug only.

The port **shall not** drive VBUS.

The port **shall** provide an **Rd** as specified in Table 4-17 on both CC pins if orientation is not needed. See Section B.2.6 for orientation detection.

The port **shall** monitor to detect when VBUS is removed.

If the DTS needs to establish **USB PD** communications, it **shall** do so only after entry to this state. In this state, the DTS takes on the initial **USB PD** role of UFP/Sink.

The DTS **shall** connect the debug signals for **Debug Accessory Mode** operation only after entry to this state.

The DTS **may** follow the DAM **Sink Power Sub-State** behavior specified in Section B.2.4.2.

B.2.4.1.4.2 Exiting from AttachedDeb.SNK State

A DTS **shall** transition to **UnattachedDeb.SNK** when VBUS is no longer present.

B.2.4.1.5 UnattachedDeb.SRC State

This state appears in Figure B-3 and Figure B-5.

When in the **UnattachedDeb.SRC** state, the DTS is waiting to detect the presence of a TS Sink.

B.2.4.1.5.1 UnattachedDeb.SRC Requirements

The DTS **shall not** drive VBUS.

The DTS **shall** source current on both CC pins independently.

The DTS **shall** provide a unique **Rp** value on each CC pin as specified in Section B.2.4.2.

B.2.4.1.5.2 Exiting from UnattachedDeb.SRC State

The DTS **shall** transition to **AttachWaitDeb.SRC** when the **SRC.Rd** state is detected on both CC pins.

A DTS DRP **shall** transition to **UnattachedDeb.SNK** within **tDRPTransition** after **dcSRC.DRP · tDRP**, or if directed.

B.2.4.1.6 AttachWaitDeb.SRC State

This state appears in Figure B-3 and Figure B-5.

The **AttachWaitDeb.SRC** state is used to ensure that the state of both of the CC pins is stable after a TS Sink is connected.

B.2.4.1.6.1 AttachWaitDeb.SRC State Requirements

The requirements for this state are identical to **UnattachedDeb.SRC**.

B.2.4.1.6.2 Exiting from AttachWaitDeb.SRC State

The DTS **shall** transition to **AttachedDeb.SRC** when VBUS is at **vSafe0V** and the **SRC.Rd** state is detected on both of the CC pins for at least **tCCDebounce**.

A DTS Source **shall** transition to **UnattachedDeb.SRC** and a DTS DRP to **UnattachedDeb.SNK** when the **SRC.Open** state is detected on either of the CC pins.

B.2.4.1.7 AttachedDeb.SRC State

This state appears in Figure B-3 and Figure B-5.

When in the **AttachedDeb.SRC** state, the DTS is attached and operating as a DTS Source.

B.2.4.1.7.1 AttachedDeb.SRC State Requirements

The DTS **shall** provide a unique *Rp* value on each CC pin as specified in Section B.2.4.2.

The DTS **shall** supply VBUS current at the level it advertises. See Section B.2.6.1.1 for advertising current level.

The DTS **shall** supply VBUS within *tVBUSON* of entering this state, and for as long as it is operating as a power source.

If the DTS needs to establish **USB PD** communications, it **shall** do so only after entry to this state. The DTS **shall not** initiate any **USB PD** communications until VBUS reaches *vSafe5V*. In this state, the DTS takes on the initial **USB PD** role of DFP/Source.

The DTS **shall** connect the debug signals for *Debug Accessory Mode* operation only after entry to this state.

B.2.4.1.7.2 Exiting from AttachedDeb.SRC State

A DTS Source **shall** transition to *UnattachedDeb.SRC* when the *SRC.Open* state is detected on either CC pin.

A DTS DRP **shall** transition to *UnattachedDeb.SNK* when *SRC.Open* is detected on either CC pin.

A DTS **shall** cease to supply VBUS within *tVBUSOFF* of exiting *AttachedDeb.SRC*.

B.2.4.1.8 TryDeb.SRC State

This state appears in Figure B-5.

When in the *TryDeb.SRC* state, the DTS DRP is querying to determine if the TS is also a DRP, to favor the DTS taking the Source role.

B.2.4.1.8.1 TryDeb.SRC State Requirements

The DTS **shall not** drive VBUS.

The DTS **shall** source current on both CC pins independently.

The DTS **shall** provide a unique *Rp* value on each CC pin as specified in Section B.2.4.2.

B.2.4.1.8.2 Exiting from TryDeb.SRC State

The DTS **shall** transition to *AttachedDeb.SRC* when the *SRC.Rd* state is detected on both CC pins for at least *tTryCCDebounce*.

The DTS **shall** transition to *TryWaitDeb.SNK* after *tDRPTry* if the state of both CC pins is not *SRC.Rd*.

B.2.4.1.9 TryWaitDeb.SNK State

This state appears in Figure B-5.

When in the *TryWaitDeb.SNK* state, the DTS has failed to become a DTS Source and is waiting to attach as a DTS Sink.

B.2.4.1.9.1 TryWaitDeb.SNK State Requirements

The DTS **shall not** drive VBUS.

Both CC pins **shall** be independently terminated to ground through *Rd*.

B.2.4.1.9.2 Exiting from TryWaitDeb.SNK State

The DTS **shall** transition to **AttachedDeb.SNK** when neither CC pin is **SNK.Open** after **tCCDebounce** and VBUS is detected.

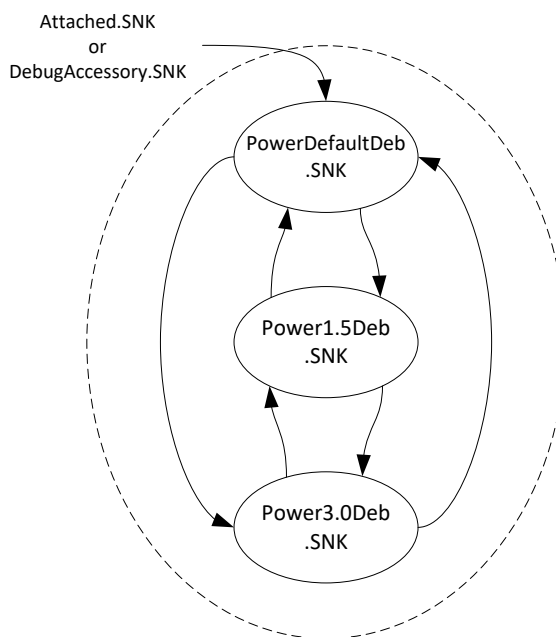
The DTS **shall** transition to **UnattachedDeb.SNK** when the state of one of the CC pins is **SNK.Open** for at least **tPDDebounce** or if VBUS is not detected within **tPDDebounce**.

B.2.4.2 Power Sub-State Requirements**B.2.4.2.1 TS Sink Power Sub-State Requirements**

When in the **DebugAccessory.SNK** state and the DTS Source is supplying default VBUS, the TS Sink **shall** operate in one of the sub-states shown in Figure B-6. The initial TS Sink Power Sub-State is **PowerDefaultDeb.SNK**. Subsequently, the TS Sink Power Sub-State is determined by the DTS Source's USB Type-C current advertisement determined by the **Rp** value on each CC pin as shown in Table B-2. The TS Sink in the attached state **shall** remain within the TS Sink Power Sub-States until either VBUS is removed, or a **USB PD** contract is established with the Source.

The TS Sink is only required to implement TS Sink Power Sub-State transitions if the TS Sink wants to consume more than default USB current.

Note, a TS Source will not use the values in Table B-2. A TS Source will present the same **Rp** on each CC pin using the standard **Rp** value for the desired current advertisement.

Figure B-6 TS Sink Power Sub-States**Table B-2 Rp/Rp Charging Current Values for a DTS Source**

Mode of Operation	CC1	CC2
Default USB Power	Rp for 3 A	Rp for 1.5 A
USB Type-C Current @ 1.5 A	Rp for 1.5 A	Rp for Default
USB Type-C Current @ 3 A	Rp for 3 A	Rp for Default

B.2.4.2.2 PowerDefaultDeb.SNK Sub-State

This sub-state supports DAM Sinks consuming current within the lowest range (default) of Source-supplied current.

B.2.4.2.2.1 PowerDefaultDeb.SNK Sub-State Requirements

The port **shall** draw no more than the default USB power from VBUS. See Section 4.6.2.1.

If the DTS Sink wants to consume more than the default USB power, it **shall** monitor **vRd** on both CC pins to determine if more current is available from the Source.

B.2.4.2.2.2 Exiting from PowerDefaultDeb.SNK Sub-State

For any change on CC indicating a change in allowable power, the DAM Sink **shall not** transition until the new **vRd** voltages on each CC pin have been stable for at least **tRpValueChange**.

For **vRd** voltages on the CC pins indicating 1.5 A mode, the DAM Sink **shall** transition to the Power1.5Deb.SNK Sub-State.

For **vRd** voltages on the CC pins indicating 3 A mode, the DAM Sink **shall** transition to the Power3.0Deb.SNK Sub-State.

B.2.4.2.3 Power1.5Deb.SNK Sub-State

This sub-state supports DAM Sinks consuming current within the two lower ranges (default and 1.5 A) of DAM Source-supplied current.

B.2.4.2.3.1 Power1.5Deb.SNK Sub-State Requirements

The DAM Sink **shall** draw no more than 1.5 A from VBUS.

The DAM Sink **shall** monitor both **vRd** voltages while it is in this sub-state.

B.2.4.2.3.2 Exiting from Power1.5Deb.SNK Sub-State

For any change on the CC pins indicating a change in allowable power, the DAM Sink **shall not** transition until the new **vRd** voltages on both CC pins have been stable for at least **tRpValueChange**.

For **vRd** voltages on the CC pins indicating Default USB Power mode, the port **shall** transition to the PowerDefaultDeb.SNK Sub-State and reduce its power consumption to the new range within **tSinkAdj**.

For **vRd** voltages on the CC pins indicating 3 A mode, the port **shall** transition to the Power3.0Deb.SNK Sub-State.

B.2.4.2.4 Power3.0Deb.SNK Sub-State

This sub-state supports DAM Sinks consuming current within all three ranges (default, 1.5 A and 3.0 A) of DAM Source-supplied current.

B.2.4.2.4.1 Power3.0Deb.SNK Sub-State Requirements

The port **shall** draw no more than 3.0 A from VBUS.

The port **shall** monitor both **vRd** voltages while it is in this sub-state.

B.2.4.2.4.2 Exiting from Power3.0Deb.SNK Sub-State

For any change on the CC pins indicating a change in allowable power, the port **shall not** transition until the new **vRd** voltages on both CC pins have been stable for at least **tRpValueChange**.

For **vRd** voltages on the CC pins indicating Default USB Power mode, the port **shall** transition to the PowerDefaultDeb.SNK Sub-State and reduce its power consumption to the new range within **tSinkAdj**.

For **vRd** voltages on the CC pins indicating 1.5 A mode, the DAM Sink **shall** transition to the Power1.5Deb.SNK Sub-State.

B.2.4.2.5 DTS Sink Power Sub-State Requirements

A DTS Sink follows the same power sub-states defined in Section 4.5.2.3. The TS Source will be advertising current with a standard **Rp** value that is the same for each CC pin. If **optional** orientation detection is performed, the DTS Sink will only be able to determine the **Rp** value from the CC pin that is set for **USB PD** communication.

B.2.4.3 Connection State Summary

Table B-3 defines the mandatory and **optional** states for each type of port. For states allowing **USB PD** communication, DAM connections requiring **USB PD** communication **shall** determine orientation by the steps described in Section B.2.6.

Table B-3 Mandatory and Optional States

	DTS Source	DTS Sink	DTS DRP	USB PD Communication and/or Debug Signal Activity
UnattachedDeb.SNK	N/A	Mandatory	Mandatory	Not Permitted
AttachWaitDeb.SNK	N/A	Mandatory	Mandatory	Not Permitted
AttachedDeb.SNK	N/A	Mandatory	Mandatory	Permitted
UnattachedDeb.SRC	Mandatory	N/A	Mandatory	Not Permitted
AttachWaitDeb.SRC	Mandatory	N/A	Mandatory	Not Permitted
AttachedDeb.SRC	Mandatory	N/A	Mandatory	Permitted
TryDeb.SRC	N/A	N/A	Mandatory	Not Permitted
TryWaitDeb.SNK	N/A	N/A	Mandatory	Not Permitted

B.2.5 DTS Port Interoperability Behaviors

This section describes interoperability behavior between DTS ports and TS ports.

B.2.5.1 DTS Port to TS Port Interoperability Behaviors

The following sub-sections describe typical port-to-port interoperability behaviors for the various combinations of DTS and TS Sources, Sinks and DRPs as presented in Table B-1.

B.2.5.1.1 DTS Source to TS Sink Behavior

The following describes the behavior when a DTS Source is connected to a TS Sink.

1. DTS Source and TS Sink in the unattached state.

2. DTS Source transitions from *UnattachedDeb.SRC* to *AttachedDeb.SRC* through *AttachWaitDeb.SRC*.
 - DTS Source detects the TS Sink's pull-downs on both CC pins and enters *AttachWaitDeb.SRC*. After *tCCDebounce* it then enters *AttachedDeb.SRC*.
 - DTS Source turns on VBUS.
3. TS Sink transitions from *Unattached.SNK* to *DebugAccessory.SNK* through *AttachWait.SNK*.
 - TS Sink in *Unattached.SNK* detects the DTS Source's pull-ups on both CC pins and enters *AttachWait.SNK*. After that state persists for *tCCDebounce* and it detects VBUS, it enters *DebugAccessory.SNK*.
4. While the DTS Source and TS Sink are in the attached state:
 - DTS Source adjusts both *Rp* values as needed for offered current.
 - TS Sink detects and monitors *vRd* on the CC pins for available current on VBUS and performs any orientation required.
 - DTS Source monitors both CC pins for detach and when detected on either pin, enters *UnattachedDeb.SRC*.
 - TS Sink monitors VBUS for detach and when detected, enters *Unattached.SNK*.

B.2.5.1.2 DTS Source to TS DRP Behavior

The following describes the behavior when a DTS Source is connected to a TS DRP.

1. DTS Source and TS DRP in the unattached state.
 - TS DRP alternates between *Unattached.SRC* and *Unattached.SNK*.
2. DTS Source transitions from *UnattachedDeb.SRC* to *AttachedDeb.SRC* through *AttachWaitDeb.SRC*.
 - DTS Source detects the TS DRP's pull-downs on both CC pins and enters *AttachWaitDeb.SRC*. After *tCCDebounce* it then enters *AttachedDeb.SRC*.
 - DTS Source turns on VBUS.
3. TS DRP transitions from *Unattached.SNK* to *DebugAccessory.SNK* through *AttachWait.SNK*.
 - TS DRP in *Unattached.SNK* detects the DTS Source's pull-ups on both CC pins and enters *AttachWait.SNK*. After that state persists for *tCCDebounce* and it detects VBUS, it enters *DebugAccessory.SNK*.
4. While the DTS Source and TS DRP are in their respective attached states:
 - DTS Source adjusts both *Rp* values as needed for offered current.
 - TS DRP detects and monitors *vRd* on both CC pins for available current on VBUS and performs any orientation required.
 - DTS Source monitors both CC pins for detach, and when detected, enters *UnattachedDeb.SRC*.
 - TS DRP monitors VBUS for detach and when detected, enters *Unattached.SNK* (and resumes toggling between *Unattached.SNK* and *Unattached.SRC*).

B.2.5.1.3 DTS Sink to TS Source Behavior

The following describes the behavior when a DTS Sink is connected to a TS Source.

1. TS Source and DTS Sink in the unattached state.
2. TS Source transitions from *Unattached.SRC* to *UnorientedDebugAccessory.SRC* through *AttachWait.SRC*.
 - TS Source detects the DTS Sink's pull-downs on both CC pins and enters *AttachWait.SRC*. After *tCCDebounce*, it enters *UnorientedDebugAccessory.SRC*.

- TS Source turns on VBUS.
- 3. DTS Sink transitions from *UnattachedDeb.SNK* to *AttachedDeb.SNK* through *AttachWaitDeb.SNK*.
 - DTS Sink in *UnattachedDeb.SNK* detects the TS Source's pull-ups on both CC pins and enters *AttachWaitDeb.SNK*.
 - DTS Sink in *AttachWaitDeb.SNK* detects that the pull-ups on both CC pins persist for *tCCDebounce* and it detects VBUS. It enters *AttachedDeb.SNK*.
 - DTS sink determines advertised current from *vRd* on either CC pin.
- 4. If orientation supported, DTS Sink adjusts *Rd* on the non-CC communication pin as needed for orientation detection.
- 5. If orientation supported, TS Source detects change in *vRd* of one of the CC pins and transitions from *UnorientedDebugAccessory.SRC* to *OrientedDebugAccessory.SRC* and performs any orientation required.
- 6. While the TS Source and DTS Sink are in the attached state:
 - If orientation is supported, DTS sink determines any change in advertised current from *vRd* of the CC pin that has been set as the CC communication pin.
 - TS Source monitors both CC pins for detach and when detected, enters *Unattached.SRC*.
 - DTS Sink monitors VBUS for detach and when detected, enters *UnattachedDeb.SNK*.

B.2.5.1.4 DTS Sink to TS DRP Behavior

The following describes the behavior when a DTS Sink is connected to a TS DRP.

1. DTS Sink and TS DRP in the unattached state.
 - TS DRP alternates between *Unattached.SRC* and *Unattached.SNK*.
2. TS DRP transitions from *Unattached.SRC* to *UnorientedDebugAccessory.SRC* through *AttachWait.SRC*.
 - TS DRP in *Unattached.SRC* detects both CC pull-downs of DTS Sink in *UnattachedDeb.SNK* and enters *AttachWait.SRC*.
 - TS DRP in *AttachWait.SRC* detects that the pull-downs on both CC pins persist for *tCCDebounce*. It then enters *UnorientedDebugAccessory.SRC* and turns on VBUS.
3. DTS Sink transitions from *UnattachedDeb.SNK* to *AttachedDeb.SNK* through *AttachWaitDeb.SNK*.
 - DTS Sink in *UnattachedDeb.SNK* detects the TS DRP's pull-ups on both CC pins and enters *AttachWaitDeb.SNK*. After that state persists for *tCCDebounce* and it detects VBUS, it enters *AttachedDeb.SNK*.
 - DTS sink determines advertised current from *vRd* on either CC pin.
7. If orientation is supported, DTS Sink adjusts *Rd* on the non-CC communication pin as needed for orientation detection.
8. If orientation supported, TS DRP detects change in *vRd* on one of the CC pins and transitions to *OrientedDebugAccessory.SRC* and performs the required orientation.
9. While the TS DRP and DTS Sink are in the attached state:
 - If orientation is supported, DTS sink determines any change in advertised current from *vRd* of the CC pin that has been set as the CC communication pin.
 - TS DRP monitors both CC pins for detach and when detected, enters *Unattached.SNK*.
 - DTS Sink monitors VBUS for detach and when detected, enters *UnattachedDeb.SNK*.

B.2.5.1.5 DTS DRP to TS Sink Behavior

The following describes the behavior when a DTS DRP is connected to a TS Sink.

1. DTS DRP and TS Sink in the unattached state.
 - DTS DRP alternates between *UnattachedDeb.SRC* and *UnattachedDeb.SNK*.
2. DTS DRP transitions from *UnattachedDeb.SRC* to *AttachedDeb.SRC* through *AttachWaitDeb.SRC*.
 - DTS DRP in *UnattachedDeb.SRC* detects both of the CC pull-downs of TS Sink enters *AttachWaitDeb.SRC*.
 - DTS DRP in *AttachWaitDeb.SRC* detects that the pull-downs on both CC pins persist for *tCCDebounce*. It then enters *AttachedDeb.SRC*.
 - DTS DRP turns on VBUS.
3. TS Sink transitions from *Unattached.SNK* to *DebugAccessory.SNK* through *AttachWait.SNK*.
 - TS Sink in *Unattached.SNK* detects the DTS DRP's pull-ups on both CC pins and enters *AttachWait.SNK*.
 - TS Sink in *AttachWait.SNK* detects that the pull-ups on both CC pins persist for *tCCDebounce* and it detects VBUS. It enters *DebugAccessory.SNK*.
4. While the DTS DRP and TS Sink are in their respective attached states:
 - DTS DRP adjusts *Rp* as needed for offered current.
 - TS Sink detects and monitors *vRd* on the CC pins for available current on VBUS and performs any orientation required.
 - DTS DRP monitors both CC pins for detach and when detected, enters *UnattachedDeb.SNK*.
 - TS Sink monitors VBUS for detach and when detected, enters *Unattached.SNK*.

B.2.5.1.6 DTS DRP to TS DRP Behavior

The following describes the behavior when a DTS DRP is connected to TS DRP.

Case #1:

1. Both DRPs in the unattached state.
 - DTS DRP alternates between *UnattachedDeb.SRC* and *UnattachedDeb.SNK*.
 - TS DRP alternate between *Unattached.SRC* and *Unattached.SNK*.
2. DTS DRP transitions from *UnattachedDeb.SRC* to *AttachWaitDeb.SRC*.
 - DTS DRP in *UnattachedDeb.SRC* detects both CC pull-downs of TS DRP in *Unattached.SNK* and enters *AttachWaitDeb.SRC*.
3. TS DRP transitions from *Unattached.SNK* to *AttachWait.SNK*.
 - TS DRP in *Unattached.SNK* detects both CC pull-ups of DTS DRP and enters *AttachWait.SNK*.
4. DTS DRP transitions from *AttachWaitDeb.SRC* to *AttachedDeb.SRC*.
 - DTS DRP in *AttachWaitDeb.SRC* continues to see both CC pull-downs of TS DRP for *tCCDebounce*, enters *AttachedDeb.SRC* and turns on VBUS.
5. TS DRP transitions from *AttachWait.SNK* to *DebugAccessory.SNK*.
 - TS DRP detects DTS DRP's pull-ups on both CC pins for *tCCDebounce* and detects VBUS and enters *DebugAccessory.SNK*.
 - TS DRP detects and monitors *vRd* on the CC pins for available current on VBUS and performs any orientation required.

6. While the TS DRP and DTS DRP are in the attached state:
 - TS DRP monitors VBUS for detach and when detected, enters *Unattached.SNK*.
 - DTS DRP monitors both CC pins for detach and when detected, enters *UnattachedDeb.SNK*.

Case #2:

1. Both DRPs in the unattached state.
 - DTS DRP alternates between *UnattachedDeb.SRC* and *UnattachedDeb.SNK*.
 - TS DRP alternates between *Unattached.SRC* and *Unattached.SNK*.
2. DTS DRP transitions from *UnattachedDeb.SNK* to *AttachWaitDeb.SNK*.
 - DTS DRP in *UnattachedDeb.SNK* detects both CC pull-ups of TS DRP in *Unattached.SRC* and enters *AttachWaitDeb.SNK*.
3. TS DRP transitions from *Unattached.SRC* to *UnorientedDebugAccessory.SRC* through *AttachWait.SRC*.
 - TS DRP in *Unattached.SRC* detects both CC pull-downs of DTS DRP and enters *AttachWait.SRC*.
 - TS DRP in *AttachWait.SRC* continues to see both CC pull-downs of TS DRP for *tCCDebounce*, enters *UnorientedDebugAccessory.SRC* and turns on VBUS.
4. DTS DRP transitions from *AttachWaitDeb.SNK* to *TryDeb.SRC*.
 - DTS DRP in *AttachWaitDeb.SNK* continues to see both CC pull-ups of TS DRP for *tCCDebounce* and detects VBUS, enters *TryDeb.SRC*.
5. TS DRP transitions from *UnorientedDebugAccessory.SRC* to *Unattached.SNK*.
 - TS DRP in *UnorientedDebugAccessory.SRC* detects the removal of both CC pull-downs of DTS DRP and enters *Unattached.SNK*.
6. TS DRP transitions from *Unattached.SNK* to *AttachWait.SNK*.
 - TS DRP in *Unattached.SNK* detects both CC pull-ups of DTS DRP and enters *AttachWait.SNK*.
7. DTS DRP transitions from *TryDeb.SRC* to *AttachedDeb.SRC*.
 - DTS DRP in *TryDeb.SRC* detects both CC pull-downs of TS DRP for *tTryCCDebounce* and enters *AttachedDeb.SRC*.
 - DTS DRP turns on VBUS.
8. TS DRP transitions from *AttachWait.SNK* to *DebugAccessory.SNK*.
 - TS DRP detects DTS DRP's pull-ups on both CC pins for *tCCDebounce* and detects VBUS and enters *DebugAccessory.SNK*.
9. While the DTS DRP and TS DRP are in their respective attached states:
 - DTS DRP adjusts *Rp* as needed for offered current.
 - TS DRP detects and monitors *vRd* on the CC pins for available current on VBUS and performs any orientation required.
 - DTS DRP monitors both CC pins for detach and when detected, enters *UnattachedDeb.SNK*.
 - TS DRP monitors VBUS for detach and when detected, enters *Unattached.SNK*.

B.2.5.1.7 DTS DRP to TS Source Behavior

The following describes the behavior when a DTS DRP is connected to TS Source.

1. DTS DRP and TS Source in the unattached state.

- DTS DRP alternates between *UnattachedDeb.SRC* and *UnattachedDeb.SNK*.
- TS Source in *Unattached.SRC*.
- 2. DTS DRP transitions from *UnattachedDeb.SNK* to *AttachWaitDeb.SNK*.
 - DTS DRP in *UnattachedDeb.SNK* detects pull-ups on both CC pins and enters *AttachWaitDeb.SNK*.
- 3. TS Source transitions from *Unattached.SRC* to *UnorientedDebugAccessory.SRC* through *AttachWait.SRC*.
 - TS Source in *Unattached.SRC* detects both CC pull-downs of DTS DRP and enters *AttachWait.SRC*.
 - TS Source in *AttachWait.SRC* continues to see both CC pull-downs of DTS DRP for *tCCDebounce*, enters *UnorientedDebugAccessory.SRC* and turns on VBUS.
- 4. DTS DRP transitions from *AttachWaitDeb.SNK* to *TryDeb.SRC*.
 - DTS DRP in *AttachWaitDeb.SNK* continues to see both CC pull-ups of TS DRP for *tCCDebounce* and detects VBUS, enters *TryDeb.SRC*.
- 5. TS Source transitions from *UnorientedDebugAccessory.SRC* to *Unattached.SRC*.
 - TS Source in *UnorientedDebugAccessory.SRC* detects the removal of both CC pull-downs of DTS DRP and enters *Unattached.SRC*.
- 6. DTS DRP transitions from *TryDeb.SRC* to *TryWaitDeb.SNK*.
 - After *tDRPTry*, DTS DRP does not see pull-downs on both CC pin and enters *TryWaitDeb.SNK*.
- 7. TS Source transitions from *Unattached.SRC* to *UnorientedDebugAccessory.SRC*.
 - TS Source in *Unattached.SRC* detects pull-downs on both CC pins and enters *AttachWait.SRC*.
 - TS Source continues to detect pull-downs on both CC pins for *tCCDebounce* and enters *UnorientedDebugAccessory.SRC* and outputs VBUS.
- 8. DTS DRP transitions from *TryWaitDeb.SNK* to *AttachedDeb.SNK*.
 - DTS DRP sees pull-ups on both CC pins for *tCCDebounce* and detects VBUS and enters *AttachedDeb.SNK*.
 - If orientation required, DTS DRP adjusts *Rd* on the non-CC communication pin as needed for orientation detection
- 9. If orientation supported, TS Source detects change in *vRd* on one of the CC pins and transitions to *OrientedDebugAccessory.SRC* and performs the required orientation.
- 10. While the TS Source and DTS DRP are in the attached state:
 - If orientation is supported, DTS DRP determines any change in advertised current from *vRd* of the CC pin that has been set as the CC communication pin.
 - TS Source monitors both CC pins for detach and when detected, enters *Unattached.SRC*.
 - DTS DRP monitors VBUS for detach and when detected, enters *UnattachedDeb.SNK*.

B.2.5.2 DTS Port to non-DAM TS Port Interoperability Behaviors

The following sub-sections describe the non-functional port-to-port interoperability behaviors for the various combinations of DTS and TS Sources, Sinks, and DRPs that do not support DAM.

B.2.5.2.1 DTS Source to non-DAM TS Sink Behavior

The following describes the behavior when a DTS Source is connected to a non-DAM TS Sink.

1. DTS Source and TS Sink in the unattached state.

2. DTS Source transitions from *UnattachedDeb.SRC* to *AttachedDeb.SRC* through *AttachWaitDeb.SRC*.
 - DTS Source detects the non-DAM TS Sink's pull-downs on both CC pins and enters *AttachWaitDeb.SRC*. After *tCCDebounce* it then enters *AttachedDeb.SRC*.
 - DTS Source turns on VBUS.
3. Non-DAM TS Sink transitions from *Unattached.SNK* to *AttachWait.SNK*.
 - Non-DAM TS Sink in *Unattached.SNK* detects the DTS Source's pull-ups on both CC pins and enters *AttachWait.SNK*.
 - Non-DAM TS Sink continues to detect pull-ups on both CC pins and stays in *AttachWait.SNK* because it does not support DAM (will not enter *Attached.SNK* because it does not detect *SNK.Open* on either pin).
4. While the DTS Source and non-DAM TS Sink are in their final state:
 - DTS Source adjusts *Rp* as needed for offered current.
 - Non-DAM TS Sink *may* draw USB default current from DTS Source as permitted by Section 4.5.2.3 but will not enter DAM.
 - DTS Source monitors both CC pins for detach and when detected, enters *UnattachedDeb.SRC*.
 - Non-DAM TS Sink monitors both CC pins for detach and when detected, enters *Unattached.SNK*.

B.2.5.2.2 DTS Source to non-DAM TS DRP Behavior

The following describes the behavior when a DTS Source is connected to a non-DAM TS DRP.

1. DTS Source and non-DAM TS DRP in the unattached state.
 - Non-DAM TS DRP alternates between *Unattached.SRC* and *Unattached.SNK*.
2. DTS Source transitions from *UnattachedDeb.SRC* to *AttachedDeb.SRC* through *AttachWaitDeb.SRC*.
 - DTS Source detects the non-DAM TS Sink's pull-downs on both CC pins and enters *AttachWaitDeb.SRC*. After *tCCDebounce* it then enters *AttachedDeb.SRC*.
 - DTS Source turns on VBUS.
3. Non-DAM TS DRP transitions from *Unattached.SNK* to *AttachWait.SNK*.
 - Non-DAM TS DRP in *Unattached.SNK* detects the DTS Source's pull-ups on both CC pins and enters *AttachWait.SNK*.
 - Non-DAM TS DRP continues to detect pull-downs on both CC pins and stays in *AttachWait.SNK* because it does not support DAM (will not enter *Attached.SNK* because it does not detect *SNK.Open* on either pin)
4. While the DTS Source and non-DAM TS DRP are in their final state:
 - DTS Source adjusts *Rp* as needed for offered current.
 - Non-DAM TS DRP *may* draw USB default current from DTS Source as permitted by Section 4.5.2.3 but will not enter DAM.
 - DTS Source monitors both CC pins for detach and when detected, enters *UnattachedDeb.SRC*.
 - Non-DAM TS DRP monitors both CC pins for detach and when detected, enters *Unattached.SRC*.

B.2.5.2.3 DTS Sink to non-DAM TS Source Behavior

The following describes the behavior when a DTS Sink is connected to a non-DAM TS Source.

1. Non-DAM TS Source and DTS Sink in the unattached state.

2. Non-DAM TS Source transitions from *Unattached.SRC* to *AttachWait.SRC*.
 - Non-DAM TS Source detects the DTS Sink's pull-downs on both CC pins and enters *AttachWait.SRC*.
 - Non-DAM TS Source continues to detect pull-downs on both CC pins and stays in *AttachWait.SRC* because it does not support DAM (will not enter *Attached.SRC* because it does not detect *SRC.Rd* on only one CC pin).
3. DTS Sink transitions from *UnattachedDeb.SNK* to *AttachWaitDeb.SNK*.
 - DTS Sink in *UnattachedDeb.SNK* detects the non-DAM TS Source's pull-ups on both CC pins and enters *AttachWaitDeb.SNK*.
 - DTS Sink remains in *AttachWaitDeb.SNK* because it does not detect VBUS.
4. While the non-DAM TS Source and DTS Sink are in their final state:
 - Non-DAM TS Source monitors both CC pins for detach and when detected, enters *Unattached.SRC*
 - DTS Sink monitors VBUS for attach and both CC pins for detach and enters *UnattachedDeb.SNK* when both CC pins go to *SNK.Open*.

B.2.5.2.4 DTS Sink to non-DAM TS DRP Behavior

The following describes the behavior when a DTS Sink is connected to a non-DAM TS DRP.

1. DTS Sink and non-DAM TS DRP in the unattached state.
 - Non-DAM TS DRP alternates between *Unattached.SRC* and *Unattached.SNK*.
 - DTS Sink in *UnattachedDeb.SNK*.
2. Non-DAM TS DRP transitions from *Unattached.SRC* to *AttachWait.SRC*.
 - Non-DAM TS DRP detects the DTS Sink's pull-downs on both CC pins and enters *AttachWait.SRC*.
 - Non-DAM TS DRP continues to detect pull-downs on both CC pins and stays in *AttachWait.SRC* because it does not support DAM (will not enter *Attached.SRC* because it does not detect *SRC.Rd* on only one CC pin).
3. DTS Sink transitions from *UnattachedDeb.SNK* to *AttachWaitDeb.SNK*.
 - DTS Sink in *UnattachedDeb.SNK* detects the non-DAM TS DRP's pull-ups on both CC pins and enters *AttachWaitDeb.SNK*.
 - DTS Sink remains in *AttachWaitDeb.SNK* because it does not detect VBUS.
4. While the non-DAM TS DRP and DTS Sink are in their final state:
 - Non-DAM TS DRP monitors both CC pins for detach and when detected, enters *Unattached.SNK*.
 - DTS Sink monitors VBUS for attach and both CC pins for detach and enters *UnattachedDeb.SNK* when both CC pin go to *SNK.Open*.

B.2.5.2.5 DTS DRP to non-DAM TS Sink Behavior

The DTS DRP to non-DAM TS Sink behavior follows the flow in Section B.2.5.2.1.

B.2.5.2.6 DTS DRP to non-DAM TS DRP Behavior

The DTS DRP to non-DAM TS DRP behavior follows the flows in Section B.2.5.2.2 and Section B.2.5.2.4 depending on the role forced by the non-DAM TS DRP.

B.2.5.2.7 DTS DRP to non-DAM TS Source Behavior

The following describes the behavior when a DTS DRP is connected to non-DAM TS Source.

1. DTS DRP and non-DAM TS Source in the unattached state.
 - DTS DRP alternates between *UnattachedDeb.SRC* and *UnattachedDeb.SNK*.
 - Non-DAM TS Source in *Unattached.SRC*.
2. DTS DRP transitions from *UnattachedDeb.SNK* to *AttachWaitDeb.SNK*.
 - DTS DRP in *UnattachedDeb.SNK* detects pull-ups on both CC pins and enters *AttachWaitDeb.SNK*.
3. Non-DAM TS Source transitions from *Unattached.SRC* to *AttachWait.SRC*.
 - Non-DAM TS Source in *Unattached.SRC* detects pull-downs on both CC pins and enters *AttachWait.SRC*.
 - Non-DAM TS Source continues to detect pull-downs on both CC pins and stays in *AttachWait.SRC* because it does not support DAM (will not enter *Attached.SRC* because it does not detect *SRC.Rd* on only one CC pin).
 - DTS Sink remains in *AttachWaitDeb.SNK* because it does not detect VBUS.
5. While the non-DAM TS Source and DTS DRP are in their final state:
 - Non-DAM TS Source monitors both CC pins for detach and when detected, enters *Unattached.SRC*.
 - DTS DRP monitors VBUS for attach and both CC pins for detach and enters *UnattachedDeb.SRC* when both CC pin go to *SNK.Open*.

B.2.5.2.8 DTS Sink to non-DAM TS Sink with Accessory Support Behavior

The following describes the behavior when a DTS Sink is connected to a non-DAM USB Type-C TS Sink with Accessory Support.

1. DTS Sink and non-DAM TS Sink with Accessory Support (“non-DAM TS Sink” for the remainder of this flow) in the unattached state.
 - Non-DAM TS Sink alternates between *Unattached.SNK* and *Unattached.Accessory*.
 - DTS Sink in *UnattachedDeb.SNK*.
2. Non-DAM TS Sink transitions from *Unattached.Accessory* to *AttachWait.Accessory*.
 - Non-DAM TS Sink detects the DTS Sink’s pull-downs on both CC pins and enters *AttachWait.Accessory*.
 - Non-DAM TS Sink continues to detect pull-downs on both CC pins and enters USB Type-C *Debug Accessory Mode*.
3. DTS Sink transitions from *UnattachedDeb.SNK* to *AttachWaitDeb.SNK*.
 - DTS Sink in *UnattachedDeb.SNK* detects the non-DAM TS Sinks pull-ups on both CC pins and enters *AttachWaitDeb.SNK*.
 - DTS Sink remains in *AttachWaitDeb.SNK* because it does not detect VBUS.
4. While the non-DAM TS DRP and DTS Sink are in their final state:
 - Non-DAM TS Sink monitors both CC pins for detach and when detected, enters *Unattached.SNK*.
 - DTS Sink monitors both CC pins for detach and enters *UnattachedDeb.SNK* when both CC pins go to *SNK.Open*.

B.2.6 Orientation Detection

Orientation detection is **optional normative**. A USB Type-C port supporting **Debug Accessory Mode** is not required to perform orientation detection. If orientation detection is required, this method **shall** be followed.

B.2.6.1 Orientation Detection using Rd and/or Rp Values

In this **optional normative** flow, the DTS **shall** always initiate an orientation detection sequence, independent of its role as Source, Sink, or DRP. This means that the TS must detect this orientation sequence and perform multiplexing to orient and connect the port signals to the proper channels as well as determine the proper CC pin for **USB PD** communication.

B.2.6.1.1 Orientation Detection with DTS as a Source

When the DTS is presenting an **Rp**, it **shall** present asymmetric **Rp** values (Rp1/Rp2) on CC1/CC2 to indicate orientation to the TS. The DTS as a source **shall** indicate a weaker resistive value on CC2. Table B-2 shows the values of **Rp** resistance on each CC pin to indicate orientation and advertise the USB Type-C current available on VBUS. See Table 4-27 for the **Rp** resistance ranges.

Once the TS sink enters the **DebugAccessory.SNK** state, after the **vRd** on both CC pins is stable for **tRpValueChange**, it will orient its signal multiplexor based on the detected orientation indicated by the relative voltages of the CC pins. The CC pin with the greater voltage is the plug CC pin, which establishes the orientation of the DTS plug in the TS receptacle and also indicates the **USB PD** CC communication wire. The TS Sink cannot perform **USB PD** communication or connect any orientation-sensitive debug signals until orientation is determined.

B.2.6.1.2 Orientation Detection with DTS as a Sink

When the DTS is a sink, it **shall** follow a two-step approach.

1. The DTS sink **shall** present **Rd/Rd** on the CC pins of the debug accessory plug. This will put the system into **Debug Accessory Mode**
2. Once the DTS sink enters **AttachedDeb.SNK** state, it **shall** present a resistance to GND of $\leq Ra$ on B5 (CC2)

The asymmetric signaling is detected by the TS Source in the **UnorientedDebugAccessory.SRC** state. Once Detected, the TS Source will move to the **OrientedDebugAccessory.SRC**. Once the TS source enters the **OrientedDebugAccessory.SRC** state, after the **SRC.Ra** level is detected on one of the CC pins, it will orient its signal multiplexor based on the detected orientation indicated by the relative voltages of the CC pins. The CC pin with the greater voltage is the plug CC pin, which establishes the orientation of the DTS plug in the TS receptacle and also indicates the **USB PD** CC communication wire. The TS Source cannot perform **USB PD** communication or connect any orientation-sensitive debug signals until orientation is determined.

B.3 Security/Privacy Requirements

Debug port(s) typically provide system access beyond the normal operation of USB hardware and protocol. Additional protection against unintended use is needed. The design must incorporate appropriate measures to prohibit unauthorized access or modification of the unit under test and to prevent exposure of private user data on the unit under test. The method of protection is not explicitly defined in this specification.

The vendor **shall** assert as part of USB compliance certification that:

- The device has met the requirement to protect the system's security and user's privacy in its vendor-specific implementation of the port, and
- The device requires the user to take an explicit action to authorize access to or modification of the unit.

C USB Type-C Digital Audio

C.1 Overview

One of the goals of USB Type-C is to help reduce the number of I/O connectors on a host platform. One connector type that could be eliminated is the legacy 3.5 mm audio device jack. A USB Type-C digital audio solution based on native USB protocol is a simple solution and is interoperable with both the host platform and audio device being connected directly without the need for adapters and operates seamlessly through existing USB topologies (e.g., through hubs and docks).

This appendix is for the **optional normative** definition of digital audio support on USB Type-C-based products. Any USB Audio Class product, having either a USB Type-C plug or receptacle, and whether it is a host system, typically an audio source, and an audio device, typically an audio sink, **shall** meet the requirements of this appendix in addition to all other applicable USB specification requirements.

C.2 USB Type-C Digital Audio Specifications

USB Type-C Digital Audio (TCDA), when implemented per this specification, **shall** be compliant with either the USB Audio Device Class 1.0, 2.0, 3.0 or 4.0 specifications as listed below. While allowed, basing a TCDA on USB Audio Device Class 1.0 or USB Audio Device Class 3.0 is not recommended. As USB Audio Device Class 4.0 improves power efficiency by providing new selective configuration tools and supporting burst mode audio data transfers, supports new CODEC types and data formats for consumer audio applications, and provides numerous extensions to support various changes in the core specification, its use is *strongly recommended*.

USB Audio Device Class 1.0 (*not recommended*) including:

- USB Device Class Definition for Audio Devices, Release 1.0
- USB Device Class Definition for Audio Data Formats, Release 1.0
- USB Device Class Definition for Audio Terminal Types, Release 1.0

USB Audio Device Class 2.0 including:

- USB Device Class Definition for Audio Devices, Release 2.0
- USB Device Class Definition for Audio Data Formats, Release 2.0
- USB Device Class Definition for Audio Terminal Types, Release 2.0

USB Audio Device Class 3.0 (*not recommended*) including:

- USB Device Class Definition for Audio Devices, Revision 3.0
- USB Device Class Definition for Audio Data Formats, Release 3.0
- USB Device Class Definition for Audio Terminal Types, Release 3.0
- USB Device Class Definition for Basic Audio Functions, Release 3.0

USB Audio Device Class 4.0 including:

- USB Device Class Definition for Audio Devices, Revision 4.0

USB Audio Device Class 3.0 specifications include the definition of basic audio function profiles (Basic Audio Device Definition, BADD). TCDA devices based on USB Audio Device Class 3.0 will implement one of the defined profiles. TCDA-capable hosts based on USB Audio Device Class 3.0 will recognize and typically implement all of the profiles that are relevant to the capabilities and usage models for the host.

TCDA devices **shall** fall into one of the following two configurations:

- a traditional VBUS-powered USB device that has a USB Type-C receptacle for use with a standard USB Type-C cable, or
- a **VCONN-Powered USB Device (VPD)** that has a captive cable with a USB Type-C plug (including thumb drive style products).

USB Type-C plug-based TCDA devices **shall not** be implemented as a variant of the USB Type-C **Analog Audio Adapter Accessory** (Appendix A in USB Type-C specifications prior to Release 2.3).

D Thermal Design Considerations for Active Cables

D.1 Introduction

USB Type C active cables use active circuitry to realize a longer link than passive cables and to maintain the electrical performance at high speed data transmission (**USB 3.2** or **USB4**). The additional power dissipation due to active components in the plug over-mold, creates a thermal challenge to passively dissipate power from its active components off the limited outer surface area of cable over-mold. Furthermore, the VBUS current, up to 5 A for power delivery, generates joule heat from the conductors along VBUS and GND lines, including copper wires, solder joints, contact pins inside connectors and copper traces on paddle board.

This appendix provides some case studies, based on **USB 3.2** active cable designs, to show the thermal impacts of certain factors affecting the maximum over-mold surface temperature T_S such as IC power inside over-mold (PO), thermal boundary, VBUS current level, and port to port spacing. The case study provided is for a specific mechanical design of the cable. When a different mechanical design (geometry or material, etc.) are used, these impacts need further investigation. The methodology of the study is thermal modeling. The modeling results have been validated for some cases (1.5 W PO and 5A VBUS) with lab test results within ± 3 °C, but not for all cases. Note that this appendix is not a full factorial or complete Design of Experiment (DOE) study and whether there is interaction among any of these factors are not covered here.

To meet thermal requirements specified in Section 6.1.3.1, as well as the junction temperature T_J requirement of any active components, an active cable **should** be carefully designed to facilitate the desired heat flow paths. A desirable thermal resistance between powered IC to over-mold surface is achieved when neither T_S nor T_J exceeds their specifications. This appendix focuses solely on T_S as output of the study, as the T_J requirement varies depending on the IC requirements.

It is recommended that system integrator such as host or device designer **should** take into consideration the heat transferred to or from an active cable in the system level thermal analysis.

Nomenclature used in this appendix:

- T_A = ambient temperature (°C)
- T_J = junction temperature (°C)
- T_S = plug over-mold outer surface maximum temperature (°C)
- T_{MB} = motherboard/thermal boundary temperature (°C)
- P_O = active component power (W) inside the over-mold that directly plugged in the host or device at each end of cable.

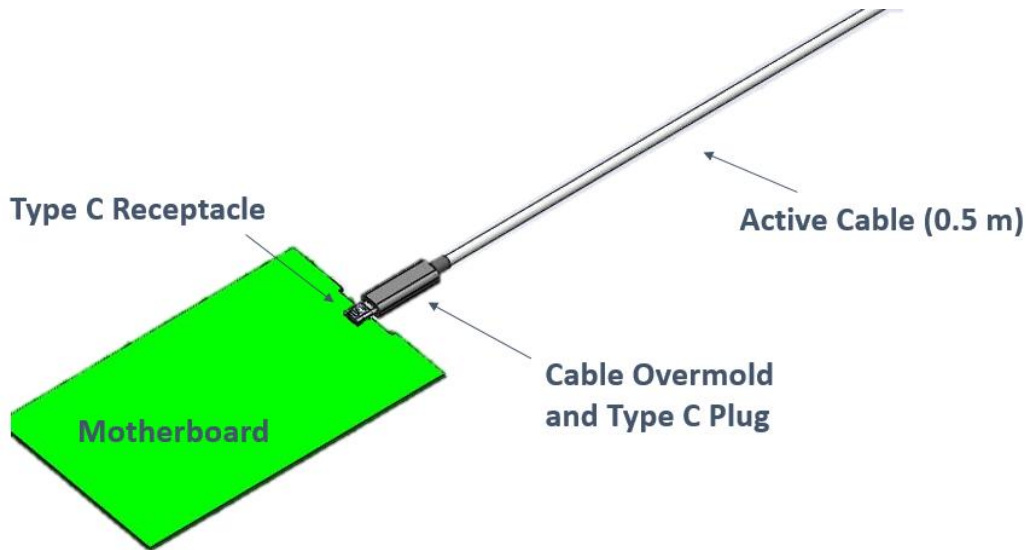
D.2 Model

D.2.1 Assumptions

A system model was built which includes a half active cable with one over-mold on the end, a mated pair of connectors (plug and receptacle) and a motherboard as its host or device side thermal boundary. The model assumes the cable is symmetric with VCONN power to be equally divided and each end of cable consumes half of VCONN power for the active components.

It is a Computational Fluid Dynamics (CFD) model with heat transfer of conduction, natural convection, and radiation. The emissivity of the plug over-mold and cable jacket is assumed to be 0.92 and the connector metal surfaces is assumed to be 0.05.

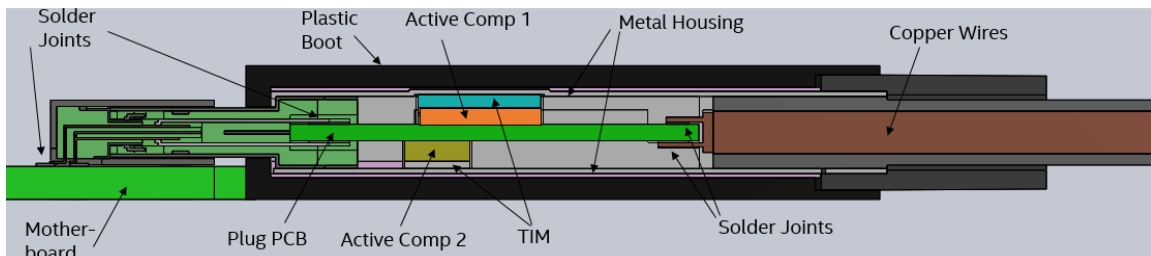
Figure D-1 Active Cable Model (Single Port, Top Mount Receptacle)



D.2.2 Model Architecture

The specific system and cable architecture used in the model is shown below.

Figure D-2 Model Architecture



The simplified cable model uses a pure copper cable, representing a typical short active cable, with total cross section of the copper conductors being about 3.8 mm².

The cable model incorporates a plastic boot for the over-mold which allows a higher surface temperature threshold than some other materials such as metal or glass. The over-mold length in the study was 35 mm.

In this specific cable design, two active components are surface mounted on plug PCB (or paddle board). Thermal Interface Material (TIM) are placed between “hot components” and “heat spreading material” such as metal housings to reduce thermal resistance between component junctions to ambient. Metal shells help to reduce T_s by spreading heat across the over-mold surface and avoid hot spots.

The plug PCB and motherboard are assumed to be FR4 based material. The motherboard is a bulk model assumed to be at a constant temperature without a point heat source on it. The receptacle is top mounted on the motherboard in single port and horizontal stacked cases, Figure D-6; and is vertically mounted in vertical stack up cases, Figure D-4 and Figure D-5.

D.2.3 Heat Sources

Main heat sources include:

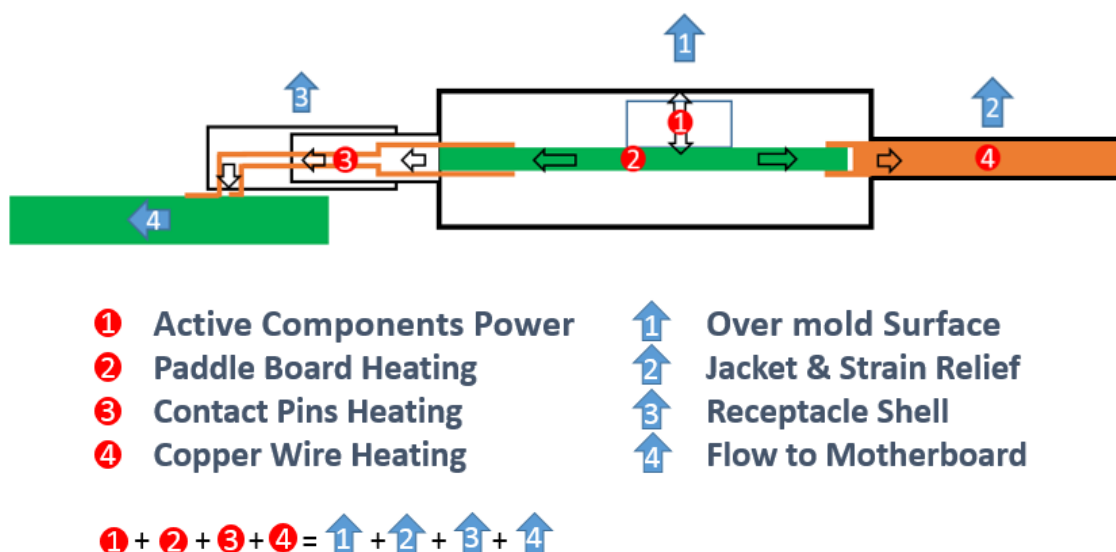
- Active component power such as re-timer, voltage regulator, etc.; the overall power inside over-mold is P_0 , which is about half of V_{CONN} power consumed by the full cable.

- Joule heat from the any conductor that carries high current, e.g., raw cable VBUS and GND copper wire, the plug PCB copper traces, contact pins of connectors, etc.

D.2.4 Heat Flow

The main power sources and heat flow paths are illustrated in Figure D-3. The overall heat generated from the cable is mainly dissipated from over-mold surfaces, cable jacket and path to motherboard. The higher thermal resistance of one heat path, the more heat it will “push off” to other heat paths and the more risk that active component junction is overheated. Since heat flow to motherboard is not a desired path from the perspective of system design, cable and over-mold design are critical to achieve balanced heat dissipation paths so not to violate either T_S or T_J requirements.

Figure D-3 Heat Sources and Heat Flow Paths



The overall heat generated from the cable **should** be consistent with the overall power dissipated by the cable. An example of half a 1.0 m active cable consuming 1.5 W and sourcing 5 A VBUS is shown below:

Table D-1 Heat Sources and Heat Dissipation Example (1.5 W cable and 5 A)

(a) Heat Sources		
Index	Heat Source	Power (W)
1	Active Components	0.750
2	Pin Heating	0.330
3	PCB Heating	0.135
4	Cable Heating	0.805
Total Power Generation:		2.020
(b) Heat Dissipation		
Index	Heat Dissipation	Power (W)
1	Plug Surface	0.500
2	Cable & SR	1.120
3	Receptacle	0.050
4	Flow to Motherboard	0.350
Total Power Dissipation:		2.020

D.3 USB 3.2 Single-Lane Active Cable

Based on the assumption that V_{CONN} power consumption is equally split between two ends of the cable and the 1 W maximum V_{CONN} power dissipation in the USB Type-C active cable (See Table 4-5), active component power in each end or over-mold power (P_0) can go up to 0.5 W in a **USB 3.2** active cable.

D.3.1 USB 3.2 Single-Lane Active Cable Design Considerations

The active cable designer *should* design for T_s less than 30 °C above T_A in the condition where thermal boundary T_{MB} is of 25 °C above T_A per Section 6.1.3.1.

D.3.1.1 USB 3.2 Single-Lane Active Cable in a Single Port Configuration

An active cable connected to a single port in a host or device can take full advantage of the overall plug surface area for heat dissipation. Table D-2 shows that when P_0 is 0.5 W, it is achievable to keep the plug over-mold surface temperature T_s of a single cable below the requirement, at both 3 A and 5 A V_{BUS} , assuming the motherboard temperature is no higher than ($T_A + 25$) °C.

Table D-2 USB 3.2 Active Cable Design Single Port Case Study at 35 °C Ambient and 60 °C Thermal Boundary (Single Lane)

	3 A V_{BUS}	5 A V_{BUS}
T_s (°C)	57	60

D.3.1.2 USB 3.2 Single-Lane Active Cable in a Multiple Port Configuration

When multi-port connector spacing is small, there is heat transfer between cables resulting in heat dissipation through natural convection being less effective than in the single port case. Radiation is also less effective due to the proximity of hot surfaces. This section lists a few typical 3-port configurations to show the impacts of receptacle spacing to the thermal performance of an active cable. For Figure D-4 and Figure D-5 minimum spacing center to center is 7 mm; for Figure D-6 it is 12.85 mm.

Figure D-4 Vertically Stacked Horizontal Connectors 3x1 Configuration (VERT)

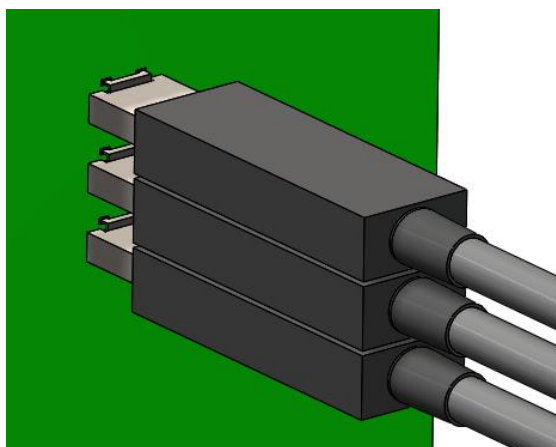


Figure D-5 Horizontally Stacked Vertical Connectors 1x3 Configuration (HZ90)

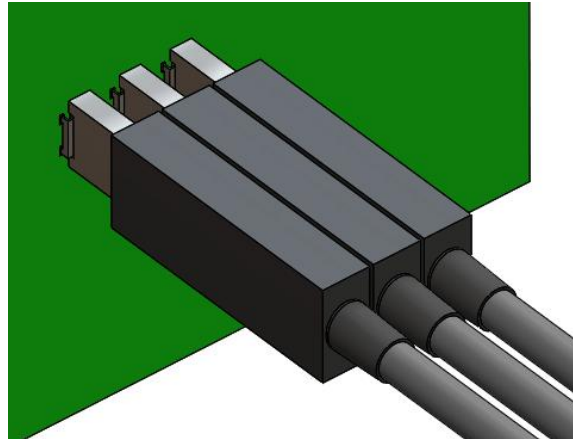
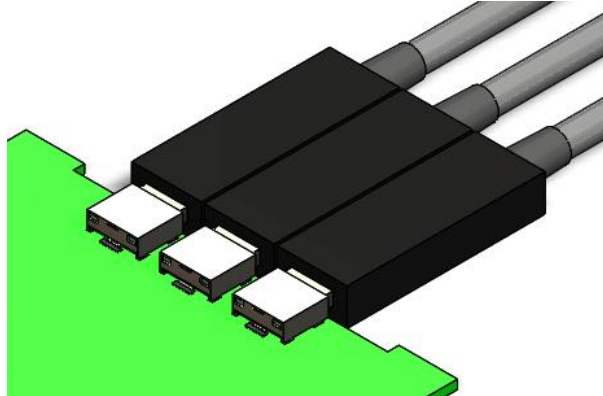


Figure D-6 Horizontally Stacked Horizontal Connectors 1x3 Configuration (HORZ)



D.3.1.2.1 USB 3.2 Single-Lane 3A Active Cable in a 3-Port Configuration

When three active cables are stacked up, the port in the center position is usually in the worst situation for heat transfer. Figure D-7 shows the temperature difference between maximum over-mold surface temperature T_s of three ports and the ambient temperature T_A when three **USB 3.2** 3A cables are plugged on a 60 °C motherboard in 35°C ambient.

In all 3-port configurations shown in Figure D-4, Figure D-5, and Figure D-6, it is achievable to keep the all three plug over-mold surface temperature T_s below the requirement, at 3 A V_{BUS} , assuming the motherboard temperature is no higher than $(T_A + 25)$ °C. Specific cable design *should* be tested and validated because the margin of center port in VERT and HZ90 is less than 1 °C at minimum port spacing in thermal modeling.

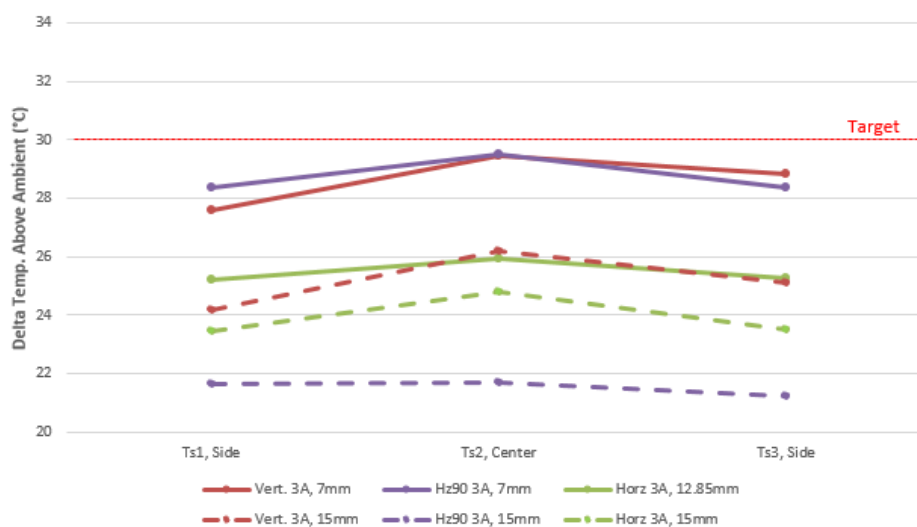
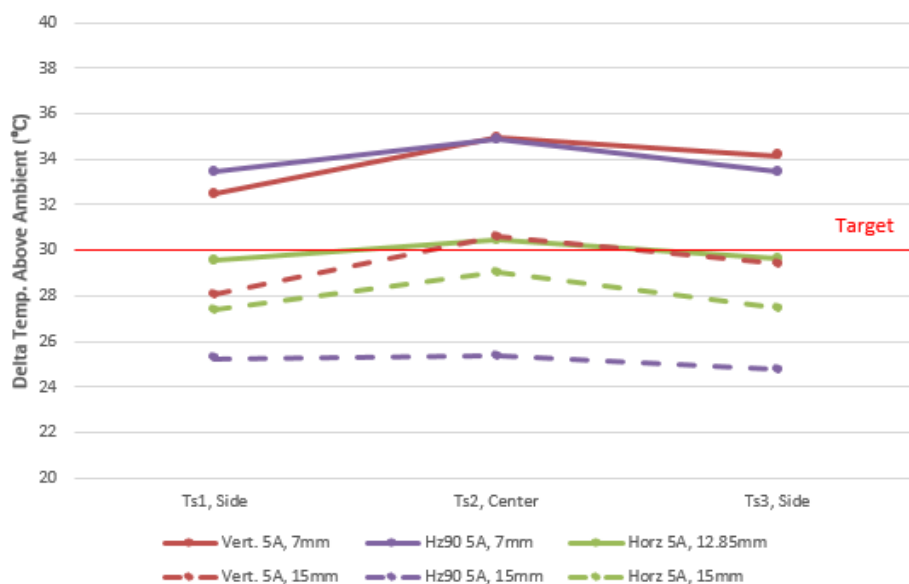
Figure D-7 USB 3.2 Single-Lane 3A Active Cable in a 3-Port Configuration**D.3.1.2.2 USB 3.2 Single-Lane 5A Active Cable in a 3-Port Configuration**

Figure D-8 shows the temperature difference between maximum over-mold surface temperature T_s of three ports and the ambient temperature T_A when three **USB 3.2** 5A cables are plugged on a 60 °C motherboard in 35 °C ambient.

All solid lines indicate the minimum spacing cases and dash lines the enlarged spacing cases. Center port is the worst case in all configurations. Three 5A cables at VERT and HZ90 configurations at minimum spacing could exceed the ($T_A + 30$ °C) specification by up to 5 °C. HORZ configuration marginally meet spec on side ports but failed on center port.

Enlarging spacing between ports greatly reduce T_s . Especially in HZ90 configuration, spacing from 7 mm to 15 mm reduced T_s by about 8 °C.

Figure D-8 USB 3.2 Single-Lane 5A Active Cable in a 3-Port Configuration

D.4 Dual-Lane Active Cables

USB 3.2 and **USB4** defines two lanes of USB high-speed data and in dual-lane operation typically has higher active component power consumption than **USB 3.2** single-lane Gen2 active cables. Higher power could heat up the over-mold and raise T_s above user comfort zone when plugging or unplugging the cable.

USB 3.2 and **USB4** dual-lane active cable *may* consume up to 1.5 W of power from V_{CONN} . This compares with the 1 W allowed for **USB 3.2** single-lane active cables.

Section D.4.1 shows T_s resulting from 0.75 W over-mold power P_0 in a 1.5 W dual-lane **USB 3.2** active cable for a certain design, in both single-port and multiple-port configurations. Results reveal that a thermal solution is necessary to meeting cable design requirements, especially in multiple-port configuration.

Both over-mold power P_0 and thermal boundary of the cable T_{MB} have impacts on T_s . The correlation of three are studied in Section D.4.1.2 which helps system and cable designer to take both factors into consideration.

D.4.1 USB 3.2 Dual-Lane Active Cable Design Considerations

The cable designer *should* design for T_s of the over-mold less than 30 °C above T_A in the condition where thermal boundary T_{MB} is of 25 °C above T_A per Section 6.1.3.1.

D.4.1.1 USB 3.2 Dual-Lane Active Cable in a Single Port Configuration

An active cable connected to a single port in a host or device can take full advantage of the overall plug surface area for heat dissipation. Table D-3 shows that when P_0 is 0.75 W, it is achievable to keep the plug over-mold surface temperature T_s of a single cable below $(T_A + 30)$ °C at both 3 A VBUS and 5 A VBUS, assuming the motherboard temperature is no higher than $(T_A + 25)$ °C.

Table D-3 USB 3.2 Active Cable Design Single Port Case Study at 35 °C Ambient and 60 °C Thermal Boundary (Dual Lane)

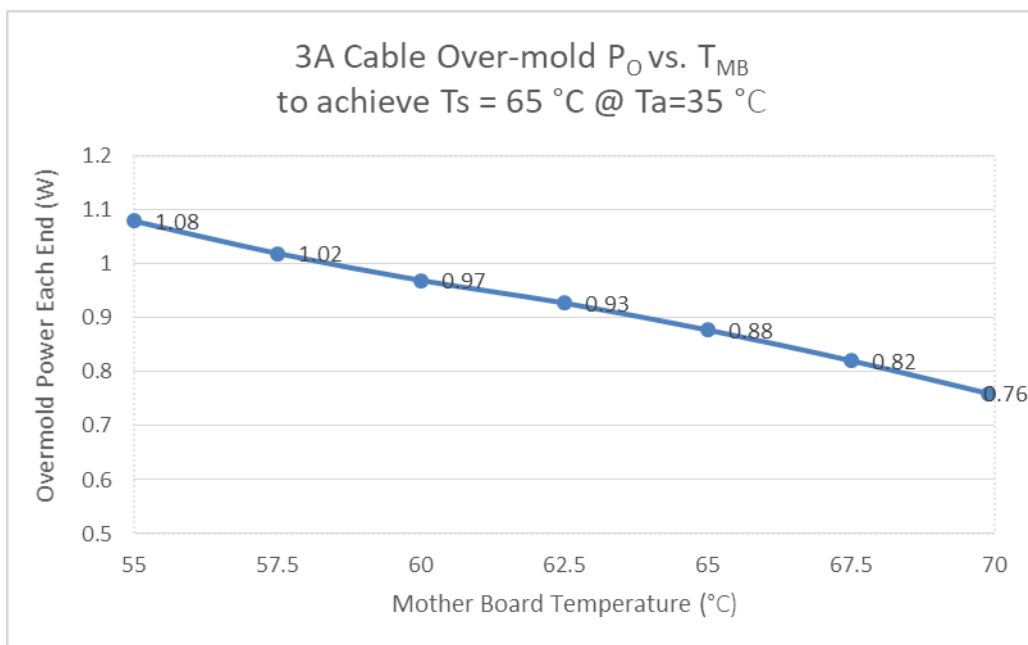
	3 A VBUS	5 A VBUS
T_s (°C)	61	64

In 5 A VBUS case, T_s is much closer to specified limit than 3 A VBUS case (Section D.3.1.1), so test and verification of thermal design is highly recommended.

D.4.1.2 Impact of Over-mold Power P_0 and Thermal Boundary Temperature T_{MB}

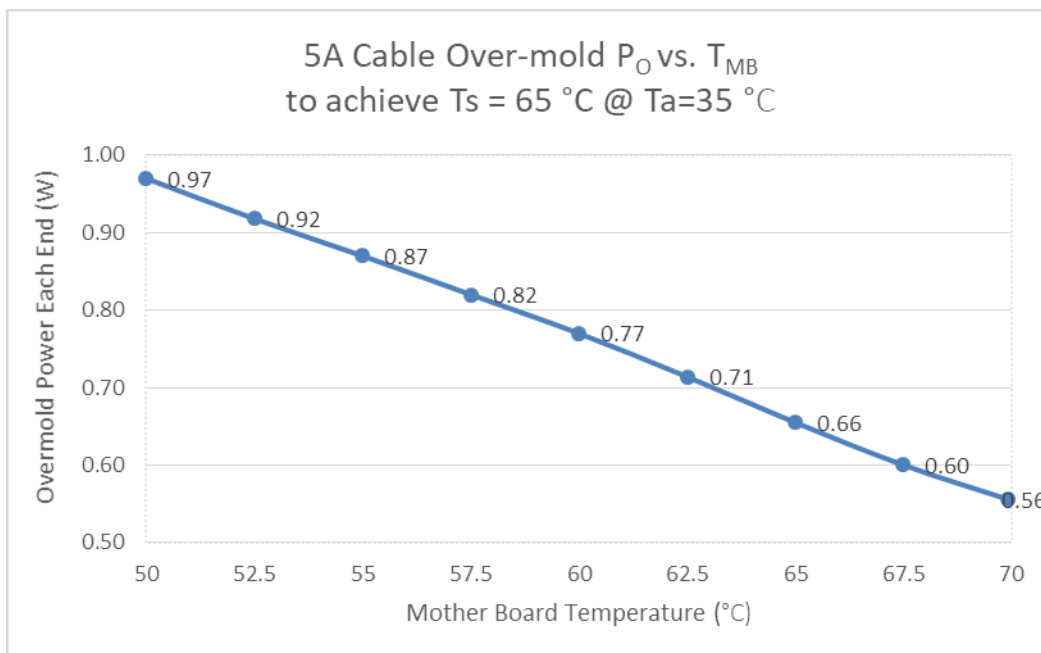
In Figure D-9, the area under graph indicates the combination of over-mold power P_0 and thermal boundary temperature T_{MB} that can achieve $T_s < (T_A + 30)$ °C in a single port configuration in a 3 A VBUS application.

Figure D-9 Impact of Overmold Power P_O and Thermal Boundary Temperature T_{MB} at 3 A VBUS in a Single Port Configuration



In Figure D-10, the area under graph indicates the combination of over-mold power P_O and thermal boundary temperature T_{MB} that can achieve $T_S < (T_A + 30)^\circ\text{C}$ in a single port configuration in a 5 A VBUS application.

Figure D-10 Impact of Overmold Power P_O and Thermal Boundary Temperature T_{MB} at 5 A VBUS in a Single Port Configuration



D.4.1.3 Dongle Cable

When overall active component power is higher than the maximum over-mold power P_0 that could meet T_S requirement, cable **may** be re-designed to move the thermal load away from the USB Type-C plug over-mold such as in a dongle cable as illustrated in Figure D-11.

Figure D-11 USB 3.2 Active Cable Dongle Design (One End Shown)



The cable **should** be designed so that the over-mold directly plugged in the host or device dissipates no more than maximum P_0 and extra heat is migrated to another part of the cable such as a dongle, so neither extra heat will flow into host and device, nor over-mold surface temperature is too hot for users to touch.

D.4.2 USB 3.2 Dual-Lane Active Cable in a Multiple Port Configuration

Multi-port connector spacing results in less effective heat dissipation by natural convection and radiation. This section lists a few typical 3-port configurations to show the impacts of receptacle spacing to the thermal performance of **USB 3.2** active cables. Naming of configurations used in this section are the same as in Section D.3.1.2.

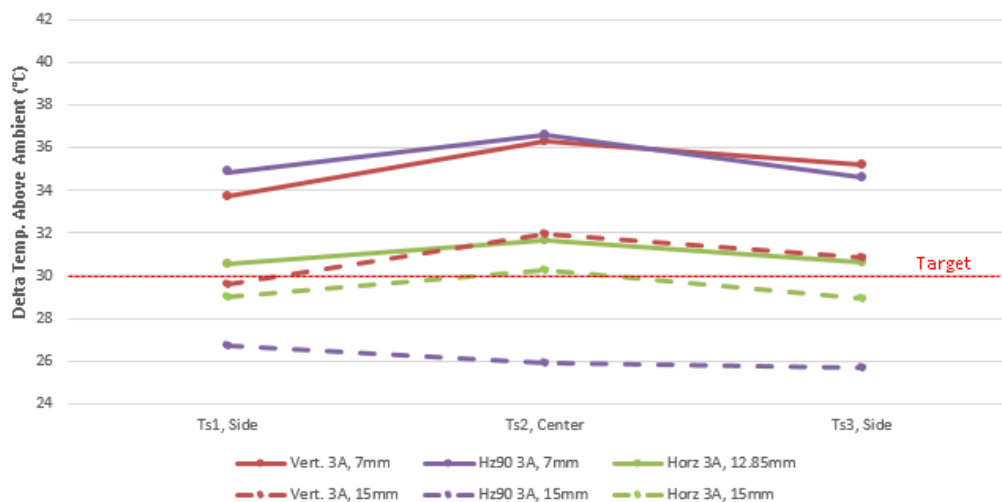
D.4.2.1 USB 3.2 Dual-Lane 3A Active Cable in a 3-Port Configuration

Figure D-12 shows the temperature difference between maximum over-mold surface temperature T_S of three ports and the ambient temperature T_A when three **USB 3.2** dual-lane 3A VBUS and 1.5 W cables are plugged on a 60 °C motherboard in 35 °C ambient. The port in the center position is usually in the worst situation for heat transfer.

All solid lines indicate the minimum spacing cases and dash lines the enlarged spacing cases. Center port is the worst case in all configurations. T_S of center port in VERT and HZ90 configurations at minimum spacing could be more than 6 °C over the ($T_A + 30$ °C) specification and in HORZ configuration about 2 °C over specification.

Enlarging spacing between ports could greatly reduce T_S . Especially in HZ90 configuration, spacing from 7 mm to 15 mm reduced T_S by about 11 °C, which helps to reduce T_S to meet specification.

Figure D-12 USB 3.2 Dual-Lane 3A Active Cable in a 3-Port Configuration



D.4.2.2 USB 3.2 Dual-Lane 5A Active Cable in a 3-Port Configuration

Figure D-13 shows the temperature difference between maximum over-mold surface temperature T_s of three ports and the ambient temperature T_A when three **USB 3.2** dual-lane 5 A VBUS and 1.5 W cables are plugged on a 60 °C motherboard in 35 °C ambient. The T_s port in the center position is still the highest of all three in all cases.

In all 3-port configurations listed in Figure D-4, Figure D-5, and Figure D-6, plug over-mold surface temperature T_s of all three ports have exceeded the requirement, at 5 A VBUS, assuming the motherboard temperature is at $(T_A + 25)$ °C. T_s of center port in VERT and HZ90 configurations at minimum spacing are the highest, near 12 °C over the $(T_A + 30)$ °C specification and in HORZ configuration about 6 °C over specification.

Enlarging spacing between ports could help reduce T_s . The largest reduction is seen in HZ90 configuration, which is near 12 °C and it brings T_s back close to target, when spacing is enlarged from 7 mm to 15 mm. However, when port spacing is not sufficient to bring T_s down to desired range, further design options in cable and host/device **should** be investigated.

Figure D-13 USB 3.2 Dual-Lane 5A Active Cable in a 3-Port Configuration



D.5 USB 3.2 Host and Device Design Considerations

Multi-port **USB 3.2** systems **should** follow the connector minimum spacing requirement defined in Section 3.10.2.

From heat flow schematics (Section D.2.4), when flow path 1 (over-mold surface dissipation) is less effective due to the limited spacing between cables, more heat would flow to motherboard and cable. It is recommended that system designer evaluate the heat flow to the system in a system level thermal analysis and provide a heat solution at the system level to reduce the motherboard temperature at these ports if necessary.

D.5.1 Heat Spreading or Heat Sinking from Host or Device

Proper thermal solutions **may** be needed on a host or device to meet cable thermal requirements. Below are examples of placement of thermal interface material on host or device USB Type-C receptacle connector to spread heat or conduct heat away from chassis. This is to help either direct heat away from active components inside cable plug or limit amount of heat from flowing from host or device into the cable plug.

Both would prevent the increased junction temperature of active components and increased cable plug surface temperature over the finger touch temperature limit. The heat management solutions shown below are not limited to certain type or size.

Figure D-14 Example: Additional Heat Spreader on Receptacle in Host or Device

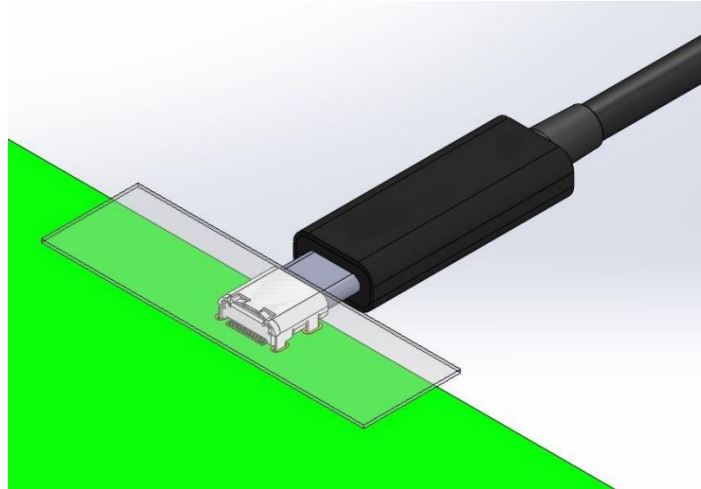
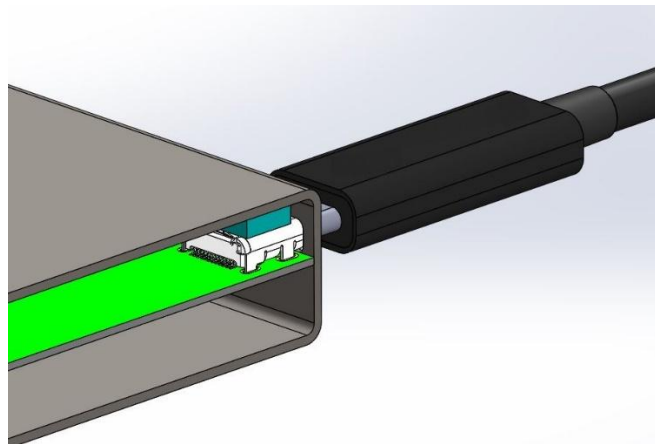


Figure D-15 Example: Heat Sinking on Chassis of Host or Device



D.5.2 Motherboard Temperature Control

Motherboard as a thermal boundary for the cable, could impact the thermal performance of cable greatly. Lowered mother temperature especially the area local around the receptacles could help reduce plug surface temperature T_s and component junction temperature T_j . See more discussion in Section D.4.1.1.

D.5.3 Wider Port Spacing for Multi-Port Applications

Wider spacing between receptacle connectors, especially when no additional heat sinking is available, is recommended for multiport application. Section D.3.1.2.1 and section D.3.1.2.2 show the impact from adjustment of port spacing.

D.5.4 Power Policies

To be added in a future update.

E Alternate Modes

All hosts and devices (except chargers and clearly marked charge-through ports) using a USB Type-C receptacle **shall** expose a USB interface (minimally **USB 2.0**). In the case where the host or device **optionally** supports **Alternate Modes**:

- The host and device **shall** use **USB Power Delivery** Structured Vendor Defined Messages (Structured VDMs) to discover, configure and enter/exit modes to enable **Alternate Modes**.
- The device is strongly encouraged to provide equivalent functionality over **USB 3.2** and/or **USB 2.0** where such exists for best user experience.
- Where no equivalent functionality is implemented, the device **shall** provide a USB interface exposing a **USB Billboard Device Class** used to provide information needed to identify the device. A device is not required to provide a USB interface exposing a **USB Billboard Device Class** for non-user facing modes (e.g., diagnostic modes).

For **Alternate Modes** support on **USB 2.0** and **USB 3.2** hubs, see Section 4.7.

For **Alternate Modes** support on **USB4** hubs, see Section 5.2.3.2.

There are **Alternate Mode** devices that look like a USB hub – the downstream facing ports of such devices are USB Type-C receptacles that support **Alternate Modes**. These devices are referred to as **Alternate Mode** expanders:

- The **Alternate Mode** port expander's downstream facing USB Type-C receptacles **shall** expose a **USB 2.0** interface.

An **Alternate Mode** port expander with the capability to pass SuperSpeed USB through its upstream facing port **should** expose SuperSpeed USB on its downstream facing USB Type-C receptacles.

E.1 Alternate Mode Architecture

The **USB Power Delivery** Structured VDMs are defined to extend the functionality a device exposes. Only Structured VDMs **shall** be used to alter the USB functionality or reconfigure the pins the USB Type-C Connector exposes. Structured VDMs provide a standard method to identify the modes a device supports and to command the device to enter and exit a mode. The use of Structured VDMs is in addition to the normal **USB PD** messages used to manage power. Structured VDMs **may** be interspersed within the normal **USB PD** messaging stream, however they **shall not** be inserted in the middle of an ongoing PD power negotiation.

The Structured VDMs consist of a request followed by a response. The response is either a successful completion of the request (ACK), an indication that the device needs time before it can service a request (BUSY), or a rejection of the request (NAK). A host and device do not enter a mode when either a NAK or BUSY is returned.

Multiple modes **may** exist and/or function concurrently. For example, a Structured VDM **may** be used to manage an active cable at the same time that another Structured VDM is used to manage the device so that both the cable and device are operating in a compatible mode.

E.2 Alternate Mode Requirements

The host and device **shall** negotiate a **USB PD** Explicit Contract before Structured VDMs **may** be used to discover or enter an **Alternate Mode**.

The ACK **shall** be sent after switching to the **Alternate Mode** has been completed by the UFP for Enter Mode and Exit Mode requests. See [Section 6.4.4 in the USB Power Delivery Specification regarding Structured VDM Usage](#). On initial connect, a UFP that reassigns pins when in its **Alternate Mode** **shall** hold off exposing **USB 2.0** speed identification (Section 7.1.5 of the **USB 2.0** specification) and **USB 3.2** receiver terminations until completing entry (including any pin reassignments) into the **Alternate Mode** or the **tAMTimeout** has expired.

If a device fails to successfully enter an **Alternate Mode** within **tAMTimeout** then the device **shall** minimally expose a **USB 2.0** interface (**USB Billboard Device Class**) that is powered by VBUS. If the device additionally supports **USB4**, then the device **should** defer exposing a **USB 2.0** interface (**USB Billboard Device Class**) due to an **Alternate Mode** timeout until the **USB4** discovery and entry process has completed (See Section 5.2.2).

When a device offers multiple modes, especially where multiple **Alternate Mode** definitions are needed in order to be compatible with multiple host-side implementations, successfully entering an **Alternate Mode** **may** be predicated on only one of the available modes being successfully recognized by a host. In this case, the device is not required to expose but **may** still expose a **USB Billboard Device Class** interface to indicate to the host the availability and status of the modes it supports.

The host **may** send an Enter Mode after **tAMTimeout**. If the device enters the mode, it **shall** respond with an ACK and discontinue exposing the **USB Billboard Device Class** interface. The device **may** expose the **USB Billboard Device Class** interface again with updated capabilities.

The current supplied over VCONN **may** be redefined by a specific **Alternate Mode**, but the power **shall not** exceed the current rating of the pin (See Section 3.7.8.4).

E.2.1 Alternate Mode Pin Reassignment

Figure E-1 illustrates the only pins that **shall** be available for functional reconfiguration in a full-featured cable. The pins highlighted in yellow are the only pins that **shall** be reconfigured.

Figure E-1 Pins Available for Reconfiguration over the Full-Featured Cable

A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
GND	RX2+	RX2-	VBUS	SBU1	D-	D+	CC	VBUS	TX1-	TX1+	GND
GND	TX2+	TX2-	VBUS	VCONN			SBU2	VBUS	RX1-	RX1+	GND
B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12

Figure E-2 illustrates the only pins that **shall** be available for functional reconfiguration in direct connect applications such as a cradle dock, captive cable, or a detachable notebook. The pins highlighted in yellow are the only pins that **shall** be reconfigured. Five additional pins are available because this configuration is not limited by the cable wiring.

Figure E-2 Pins Available for Reconfiguration for Direct Connect Applications

A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
GND	RX2+	RX2-	VBUS	SBU1	D-	D+	CC	VBUS	TX1-	TX1+	GND
GND	TX2+	TX2-	VBUS	VCONN			SBU2	VBUS	RX1-	RX1+	GND
B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12

The **USB 2.0** data pins (A6, A7) **shall** remain connected to the USB host controller during entry, while in and during exit of an **Alternate Mode** except in the case of a direct connect application that remaps A6 and A7. Direct connect applications that remap A6 and A7 through the use of an **Alternate Mode** **shall** provide a USB Billboard Class device that is presented if the remapped **Alternate Mode** is not entered within **tAMTimeout**.

E.2.2 Alternate Mode Electrical Requirements

Signaling during the use of **Alternate Modes** **shall** comply with all relevant cable assembly, adapter assembly and electrical requirements of Chapter 3.

Several requirements are specified in order to minimize risk of damage to the USB transmitters and receivers in a USB host or device when operating in an **Alternate Mode**:

- If pin pairs B11, B10 (RX1) and A11, A10 (RX2) are used on a captive cable, they **shall** be AC coupled either before or in the USB Type-C plug.
- If pin pairs B11, B10 (RX1) and A11, A10 (RX2) are used on a USB Type-C receptacle, they **may** be AC coupled and discharged per **USB 3.2** before the receptacle.
- AC coupling on pin pairs A2, A3 (TX1) and B2, B3 (TX2) as defined for SuperSpeed USB signaling per **USB 3.2** **shall** be used for **Alternate Mode** signaling.
- Signals being received at the USB Type-C receptacle **shall not** exceed the value specified for $V_{TX-DIFF-PP}$ in Table 6-18 of the **USB 3.2** specification.
- Direct Connect applications that remap pins A6 and A7 **shall** place pins A6 and A7 in a hi-Z state before transmitting the **USB PD** Enter_Mode command to the Sink. The Source **shall not** enable the alternate use of the A6 and A7 pins until an ACK has been received by the Source. In the event of a failure to enter the **Alternate Mode** after transmission of the **USB PD Enter_Mode** command, the Source **shall** restore pins A6 and A7 to the **normative USB 2.0** operation.

Direct connect applications **shall** ensure that any stubs introduced by repurposing the extra D+/D- pair do not interfere with USB communication with compliant hosts that short the pairs of pins together on the receptacle. This can be ensured by placing the **Alternate Mode** switch close to the plug, by adding inductors to eliminate the stubs at **USB 2.0** frequencies, by AC-terminating the long stubs to remove reflections at the cost of attenuated signal, or by other means.

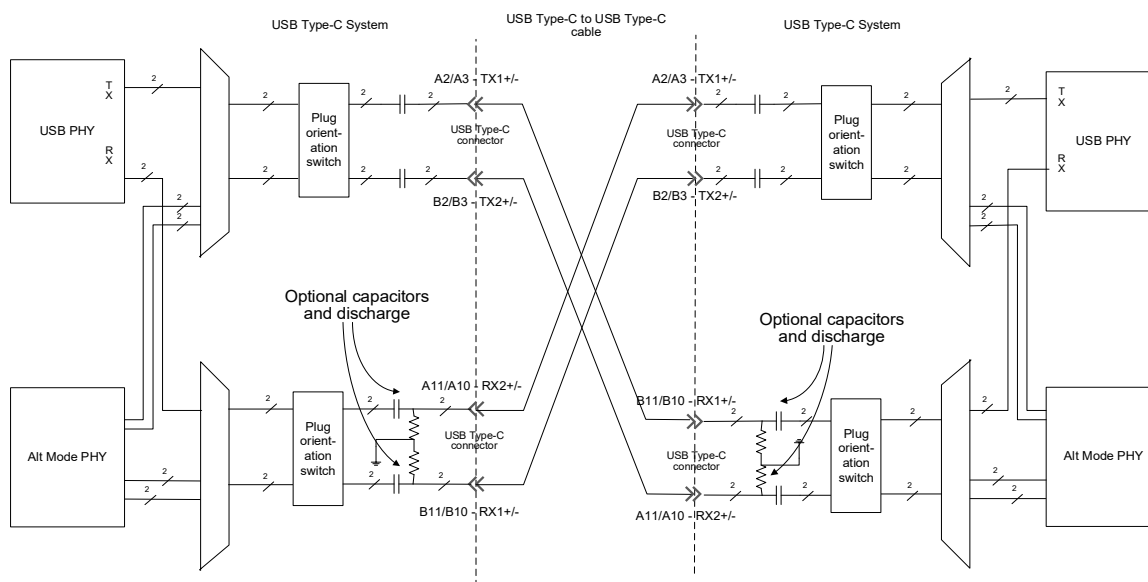
When in an **Alternate Mode**, activity on the SBU lines **shall not** interfere with **USB PD** BMC communications or interfere with detach detection.

The AC coupling requirement are the same as defined in the **USB 3.2** specification. The TX signals **shall** be AC coupled within the system before the physical connector. The RX signals **may** be DC coupled or AC coupled and discharged within the system.

It **should** be noted that the AC coupling capacitor is placed in the system next to the USB Type-C receptacle, so that the system components (the orientation switch, the **Alternate Mode** selection multiplexer, and other system components) operate within the common mode limits set by the local PHY. This applies, in the SuperSpeed USB operation, to both the transmit path and the receive path within the local system. The receive path is isolated from the common mode of the port partner by the AC coupling capacitors that are implemented on the TX path in the port partner.

Figure E-3 shows the key components in a typical **Alternate Mode** implementation using a USB Type-C to USB Type-C full featured cable. This implementation meets the AC coupling requirements, as the capacitors required to be in or before the USB Type-C plug are implemented behind the TX pins in the port partner.

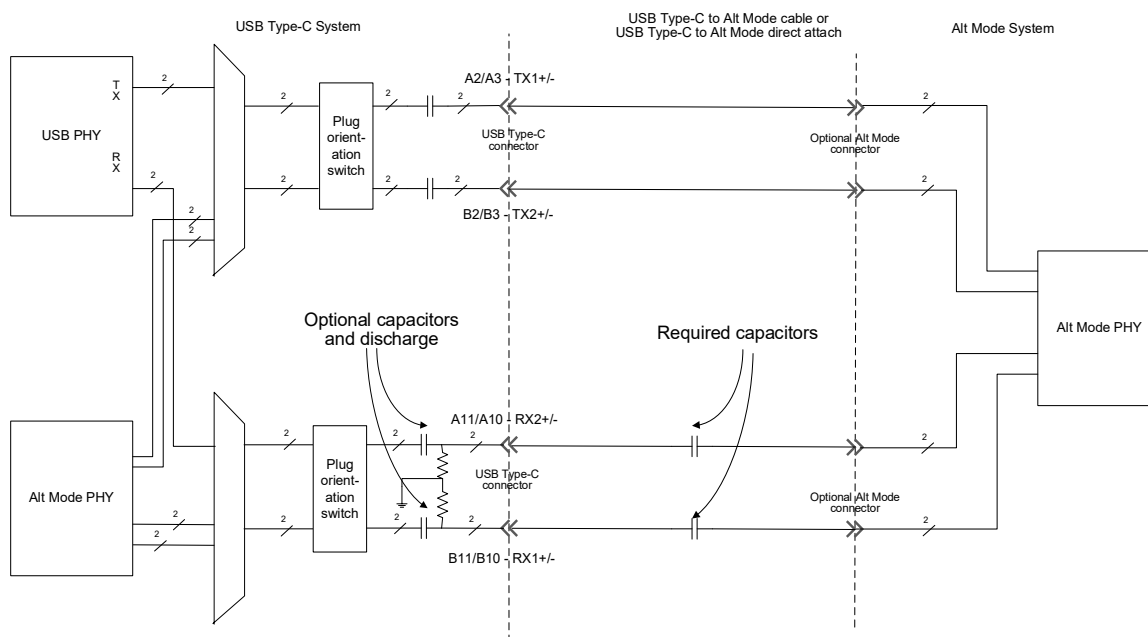
Figure E-3 Alternate Mode Implementation using a USB Type-C to USB Type-C Cable



In the case where the **Alternate Mode** System is required to implement DC blocking capacitors within the system between active system components and the **Alternate Mode** connector, then this provides the necessary isolation and further capacitors in the USB Type-C to **Alternate Mode** adapter cable are not necessary and *may* indeed impair signal integrity.

Figure E-4 shows the key components in a typical **Alternate Mode** implementation using either a USB Type-C to **Alternate Mode** connector cable, or a USB Type-C **Alternate Mode** Direct Attach device. In both cases it is necessary that the system path behind the RX pins on the USB receptacle be isolated from external common mode. This requirement is met by incorporating capacitors in or behind the USB Type-C plug on the **Alternate Mode** cable or **Alternate Mode** device.

Figure E-4 Alternate Mode Implementation using a USB Type-C to Alternate Mode Cable or Device



The USB Safe State is defined by the **USB PD** specification. The USB Safe State defines an electrical state for the SBU1/2 and TX/RX for DFPs, UFPs, and Active Cables when transitioning between USB and an **Alternate Mode**. SBU1/2 and TX/RX must transition to the USB Safe State before entering to or exiting from an **Alternate Mode**. Table E-1 defines the electrical requirements for the USB Safe State. See the **USB PD** Specification for more detail on entry/exit mechanisms to the USB Safe State.

Table E-1 USB Safe State Electrical Requirements

	SBU1/2	TX ^{1,2}	RX ²	A6/A7/B6/B7 ⁴
Common-mode voltage	0 to 1.5 V	0 to 1.5 V	0 to 1.5 V	0 to 1.5 V
Impedance to ground³	< 4 MΩ	< 4 MΩ	25 KΩ – 4 MΩ	< 4 MΩ
Notes: 1. TX common-mode voltage is defined on the integrated circuit side of the AC coupling capacitors. 2. Unused TX and RX signals should transition to USB Safe State if wired to the connector but not used. 3. The DFP and UFP shall provide a discharge path to ground in USB Safe State when a connection to the USB Type-C receptacle is present. 4. Applies to docking solutions/direct connect applications that redefine pins A6, A7, B6 and B7.				

E.3 Parameter Values

Table E-2 provides the timeout requirement for a device that supports **Alternate Modes** to enable a **USB Billboard Device Class** interface when none of the modes supported by the device are successfully recognized and configured by the DFP to which the device is attached.

Table E-2 USB Billboard Device Class Availability Following Alternate Mode Entry Failure

	Maximum	Description
<i>tAMTimeout</i>	1000 ms	The time between a Sink attach until a USB Billboard Device Class interface is exposed when an Alternate Mode is not successfully entered.

While operating in an **Alternate Mode**, the signaling **shall not** cause noise ingress onto USB signals operating concurrently that exceeds the **Vnoise** parameters given in Table E-3.

Table E-3 Alternate Mode Signal Noise Ingression Requirements

	Limit	Bandwidth
VNOISE on BMC during BMC Active	30 mV	100 ns time constant filter
VNOISE on BMC during BMC Idle	100 mV	100 ns time constant filter
VNOISE on D+/D- (Single-ended)	40 mV	500 MHz
VNOISE on D+/D- (Differential)	10 mV	500 MHz
Note: Each VNOISE parameter is the max noise ingress level allowed onto the respective interface that is due to two SBU aggressors from the Alternate Mode signaling, under respective worse case scenarios. The coupling between SBU_A/SBU_B and CC within a USB Type-C cable shall meet the requirement described in Section 3.7.2.6.4. The coupling between SBU_A/SBU_B and USB D+/D- within a USB Type-C cable shall meet the requirement described in Section 3.7.2.6.5.		

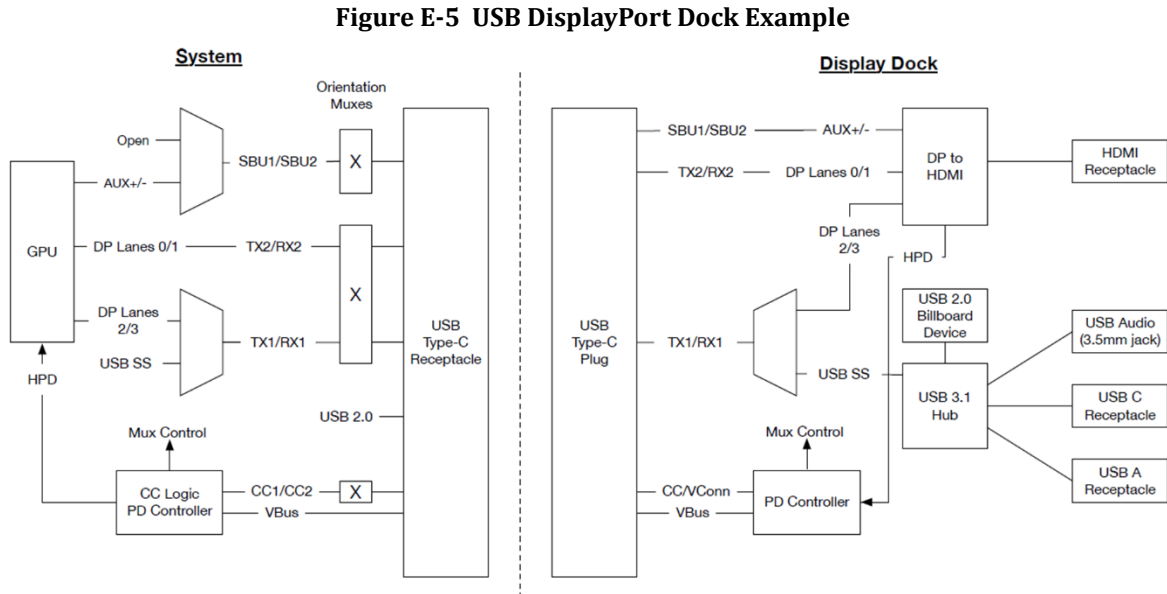
E.4 Example Alternate Mode – USB DisplayPort™ Dock

This example illustrates the use of Structured VDMs to expose and access functionality beyond the basic functionality defined by the USB Type-C Connector. The device uses its USB Type-C connector to make connection when placed in a cradle dock. This example only illustrates the functional connections.

E.4.1 USB DisplayPort™ Dock Example

- The cradle dock provides mechanical alignment and attachment in addition to those provided by the USB Type C connector allowing for only one orientation eliminating the need for an orientation MUX in the dock.
- The dock and system use **USB PD** to manage charging and power.
- The dock uses **DisplayPort** to drive a **DisplayPort-to-HDMI** adapter to support connecting an **HDMI** monitor.
- The dock has a USB hub that exposes two external USB ports and attached internal USB Devices, e.g., a USB audio Device (a 3.5 mm audio jack), and a USB Billboard Device.

Figure E-5 illustrates the USB **DisplayPort** Dock example in a block diagram form.



The system uses **USB PD** Structured VDMs to communicate with the dock to discover that it supports a compatible **Alternate Mode**. The system then uses a Structured VDM to enter the dock mode. Since **USB PD** is used, it **may** also be used to negotiate power for the system and dock. In this example, the SuperSpeed USB signals allow the dock to work as a USB-only dock when attached to a system that does not fully support the dock or even **USB PD**.

E.4.2 Functional Overview

The following summarizes the behavior resulting from attaching the example USB **DisplayPort** Dock for three likely host system cases.

1. Host system does not support **USB PD** or supports **USB PD** without Structured VDMs
 - The host does not support **USB PD**, or supports **USB PD** but not Structured VDMs, so it will not look for SVIDs using the Structured VDM method.
 - The host will discover the USB hub and operate as it would when connected to any USB hub.

- Since the host will not send an Enter Mode command, after *tAMETimeout* the dock will expose a **USB Billboard Device Class** interface that the host will enumerate. The host then reports to the user that an unsupported Device has been connected, identifying the type of Device from the **USB Billboard Device Class** information.
- 2. Host system supports **USB PD** and Structured VDMs but does not support this specific USB **DisplayPort** Dock
 - The host discovers the USB hub and operates as it would when connected to any USB hub.
 - The Host looks for SVIDs that it recognizes. The VID associated with this USB **DisplayPort** Dock *may* or *may not* be recognized by the Host.
 - If that VID is recognized by the Host, the Host then requests the modes associated with this VID. The mode associated with this USB **DisplayPort** Dock is not recognized by the Host.
 - Since the host does not recognize the mode as being supported hence will not send the Enter Mode command, after *tAMETimeout* the dock will expose a **USB Billboard Device Class** interface that the host will enumerate. The host then reports to the user that an unsupported Device has been connected, identifying the type of Device from the **USB Billboard Device Class** information.
- 3. Host system supports this specific USB **DisplayPort** Dock
 - The Host looks for SVIDs that it recognizes. The VID associated with this USB **DisplayPort** Dock is recognized by the Host.
 - The Host then requests the modes associated with this VID. The mode associated with this USB/Display Dock is recognized by the Host.
 - Since this mode is recognized as supported, the Host uses the Enter Mode command to reconfigure the USB Type-C receptacle and enter the USB **DisplayPort** Dock mode.
 - The USB **DisplayPort** Dock *may optionally* expose the **USB Billboard Device Class** interface to provide additional information to the OS.

E.4.3 Operational Summary

The following summarizes the basic process of discovery through configuration when the USB **DisplayPort** Dock is attached to the Host.

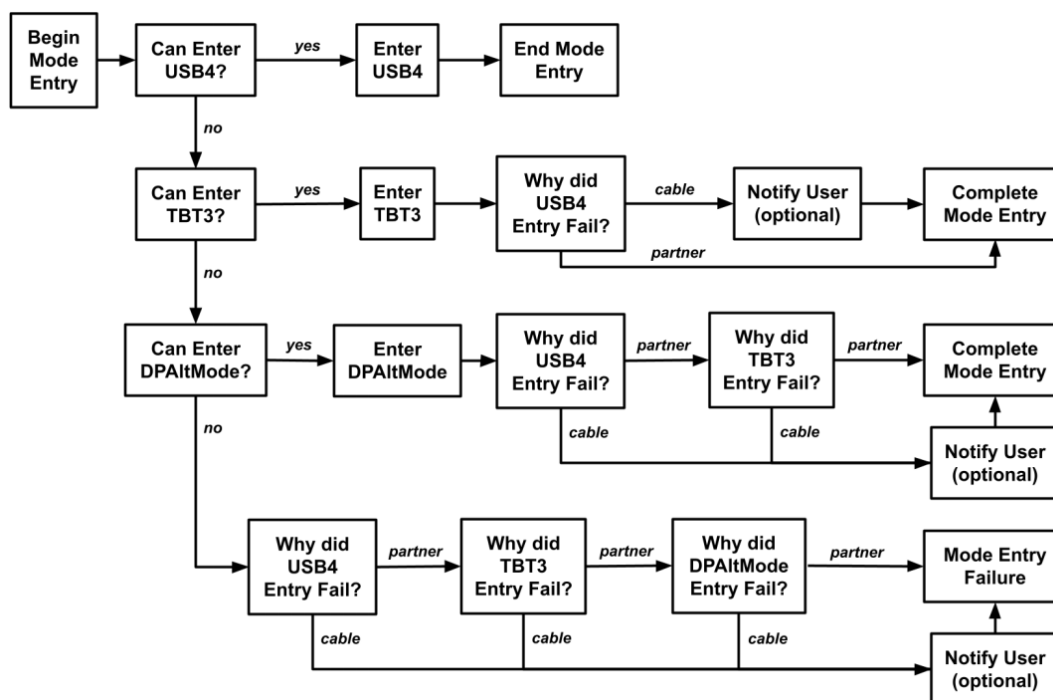
1. Host detects the presence of a device (CC pins) and the connector orientation.
2. Host applies default VBUS.
3. Host applies VCONN because the dock presents *Ra*.
4. Host uses **USB PD** to make power contract with the USB **DisplayPort** Dock.
5. Host runs the Discover Identify process.
 - a. Sends **Discover Identity** message.
 - b. Receives an ACK message with information identifying the cable.
6. Host runs the Discover SVIDs process.
 - a. Sends Discover SVID message.
 - b. Receives an ACK message with list of SVIDs for which the Dock device has modes.
7. Host runs the Discover Modes process.
 - a. Sends Discover Modes VDM for the VIDs previously discovered.
 - b. Receives an ACK message with a list of modes associated with each VID.

- c. If USB **DisplayPort** Dock mode is not found, dock will timeout and present the **USB Billboard Device Class** interface and the OS will inform the user of the error – *Done*.
 - d. *Else ...*
 8. Host runs the Enter Mode process.
 - a. Sends Enter Mode VDM with VID and USB **DisplayPort** Dock mode.
 - b. Receives an ACK message – Host is now attached to the USB **DisplayPort** Dock and supports **DisplayPort** signaling to interface additional functions in combination with USB signaling.
 9. Host stays in the USB **DisplayPort** Dock mode until.
 - a. Explicitly exited by an Exit Mode VDM.
 - b. System physically disconnected from the USB **DisplayPort** Dock.
 - c. Hard Reset on **USB PD**.
 - d. VBUS is removed.

E.5 Example Alternate Mode Entry Flow with Cable Warnings

A single USB Type-C port *may* support multiple **Alternate Modes** if it is acting as a DFP. Which **Alternate Mode** the port enters depends on two factors. Firstly, what the port, partner and cable can support. Secondly, how the DFP's **Alternate Modes** are prioritized. Figure E-6 gives an example of how an **Alternate Mode** entry flow *may* work for a DFP **USB4** port with notifications to warn the user when a cable is preventing **Alternate Mode** entry.

Figure E-6 **USB4** DFP Mode Entry Flow Example



The port in this system example supports **USB4**, **Thunderbolt 3 Alternate Mode** and **DisplayPort Alternate Mode**. These **Alternate Modes** are mutually exclusive because they use pin reassignment as detailed in Section E.2.1. The port will prioritize entering **USB4**, followed by **Thunderbolt 3 Alternate Mode**, and then **DisplayPort Alternate Mode**. It will only attempt **Thunderbolt 3 Alternate Mode** discovery if **USB4** discovery fails. It will only attempt **DisplayPort Alternate Mode** discovery if both **USB4** discovery and **Thunderbolt 3 Alternate Mode** discovery fail.

E.5.1 Operational Summary

This subsection describes how the DFP **USB4** port in this example attempts **Alternate Mode** discovery and entry when a USB Type-C device has been connected. Before attempting any **Alternate Mode** entry, both the partner and cable need to have completed **USB PD** discovery. For each **Alternate Mode**, the DFP will attempt a discovery flow to determine if it can enter that **Alternate Mode** with the attached partner and cable. It will only enter an **Alternate Mode** after successfully completing the mode discovery flow.

1. The DFP attempts **USB4** discovery defined in Section 5.1 and illustrated in Figure 5-1.
 - a. If successful, the DFP enters **USB4** and completes mode entry.
2. If the DFP fails **USB4** discovery, it attempts **Thunderbolt 3 Alternate Mode** discovery defined in Appendix F and illustrated in Figure F-1.
 - a. If successful, the DFP enters **Thunderbolt 3 Alternate Mode**.
 - b. **Optionally**, the DFP can check why **USB4** discovery failed and inform the user. Figure E-6 illustrates a case where the user is notified when **USB4** discovery fails due to the cable.
 - c. After entering **Thunderbolt 3 Alternate Mode**, the DFP completes mode entry.
3. If the DFP fails **Thunderbolt 3 Alternate Mode** discovery, it will attempt **DisplayPort Alternate Mode** discovery described in Section E.4.3.
 - a. Unlike **USB4** and **Thunderbolt 3 Alternate Mode**, **DisplayPort Alternate Mode** does not explicitly define cable conditions required for mode entry. A DFP **should** still check the cable to confirm it supports a sufficient data rate for **DP** signals as a condition of **DisplayPort Alternate Mode** discovery. While this will prevent entering **DisplayPort Alternate Mode** with a cable that **should not** work for connecting displays, it will also prevent mode entry in cases where non-compliant cables would work for connecting displays but do not have a proper eMarker. Figure E-6 shows a case where cable checks are part of the **DisplayPort Alternate Mode** discovery and entry.
 - b. If **DisplayPort Alternate Mode** entry succeeds, the DFP can **optionally** check why **USB4** and **Thunderbolt 3 Alternate Mode** discovery failed and inform the user.
 - c. If **DisplayPort Alternate Mode** entry failed, the DFP can **optionally** check why **USB4**, **Thunderbolt 3 Alternate Mode** and **DisplayPort Alternate Mode** discovery failed and notify the user.
 - d. After attempting discovery for all modes and entering the highest priority mode that successfully finished its discovery flow, the DFP completes mode entry.

F Thunderbolt™ 3 Compatibility Discovery and Entry

The **USB4** specification includes defined support for compatibility between **USB4** products that are designed to interoperate with existing **Thunderbolt™ 3 (TBT3)** products. This appendix documents the **normative** methodology to discover and enter into **TBT3** between two port partners – this methodology relies on **Alternate Mode** protocol as defined in Appendix E of this specification and the **USB Power Delivery** specification.

Thunderbolt 3 technology is organized into two primary product categories: hosts and devices. Most **TBT3** devices include at least one upstream and one downstream port although a **TBT3** device *may* include more than one downstream port in a manner similar to a hub or no downstream ports in a manner similar to a peripheral.

F.1 TBT3 Compatibility Mode Functional Requirements

In order to successfully interoperate with existing **TBT3** products, the functional requirements in the following subsections must be met.

F.1.1 TBT3-Compatible Power Requirements

Before two **TBT3**-compatible port partners can enter **TBT3** mode, a **USB PD** explicit power contract **shall** be established.

F.1.2 TBT3-Compatible Host Requirements

All **TBT3**-compatible host ports **shall** meet the following requirements.

- Support DRP operation.
- If resolved to a UFP, use **USB PD DR_Swap** to attempt to switch into the DFP data role when DFP is preferred.
- If resolved to a DFP, do not accept **USB PD DR_Swap** to remain in the DFP data role when DFP is preferred.

F.1.3 TBT3-Compatible Device Upstream Requirements

F.1.3.1 Self-Powered Device

The **TBT3**-compatible upstream port of a self-powered device **shall** meet the following requirements.

- Support DRP operation.
- Prefer Sink/UFP through the implementation and use of **Try.SNK** as needed.
- If resolved to a DFP, accept **USB PD DR_Swap** to switch into the UFP data role.

F.1.3.2 Bus-Powered Device

The **TBT3**-compatible upstream port of a bus-powered device **shall** meet the following requirements.

- Support Sink/UFP operation.
- Reject **USB PD DR_Swap** to remain in the UFP data role.

F.1.4 TBT3-Compatible Device Downstream Requirements

F.1.4.1 Self-Powered Device

The **TBT3**-compatible downstream port of a self-powered device **shall** meet the following requirements.

- Support DRP operation.
- Prefer Source/DFP through the implementation and use of **Try.SRC** as needed.

- If resolved to a DFP, do not accept **USB PD DR_Swap** to switch into the DFP data role.

F.1.4.2 Bus-Powered Device

The **TBT3**-compatible downstream port of a bus-powered device **shall** meet the following requirements.

- Support Sink.

F.1.5 TBT3-Compatible Self-Powered Device Without Predefined Upstream Port Rules

A **TBT3**-compatible device port **may** behave as either a downstream or upstream port based on its connection state to a **TBT3**-compatible host as described below.

- When no **TBT3**-compatible host is connected, the USB Type-C ports **shall**:
 - Prefer to be configured as a UFP.
 - Implement and use **Try.SNK** as needed to get into the UFP state.
 - If resolved to a DFP, initiate, or accept **USB PD DR_Swap** to switch to the UFP data role.
 - Accept **USB PD DR_Swap** to switch to the DFP data role.
 - When resolved to a UFP, identify this port as being connected to the host.
 - Put the remaining downstream ports into the **ErrorRecovery** state.
- After a **TBT3**-compatible host is initially connected, the remaining downstream USB Type-C ports **shall**:
 - Implement and use **Try.SRC** as needed to get into the DFP state.
 - Issue a Hard Reset if a **USB PD DR_Swap** is received when both a connection is present, and an **Alternate Mode** is in place.
 - Issue a **USB PD DR_Swap** to switch to the DFP data role if a connection is present but no **Alternate Mode** has been entered (this includes performing a disconnect/reconnect on the port).
 - Accept **USB PD DR_Swap** to switch to the DFP data role if a connection is present but no **Alternate Mode** has been entered (this includes performing a disconnect/reconnect on the port).
- When a **TBT3**-compatible host that was identified as a host is disconnected, the downstream USB Type-C ports **shall**:
 - Enter the **ErrorRecovery** state.
 - Behave as if no host is connected.

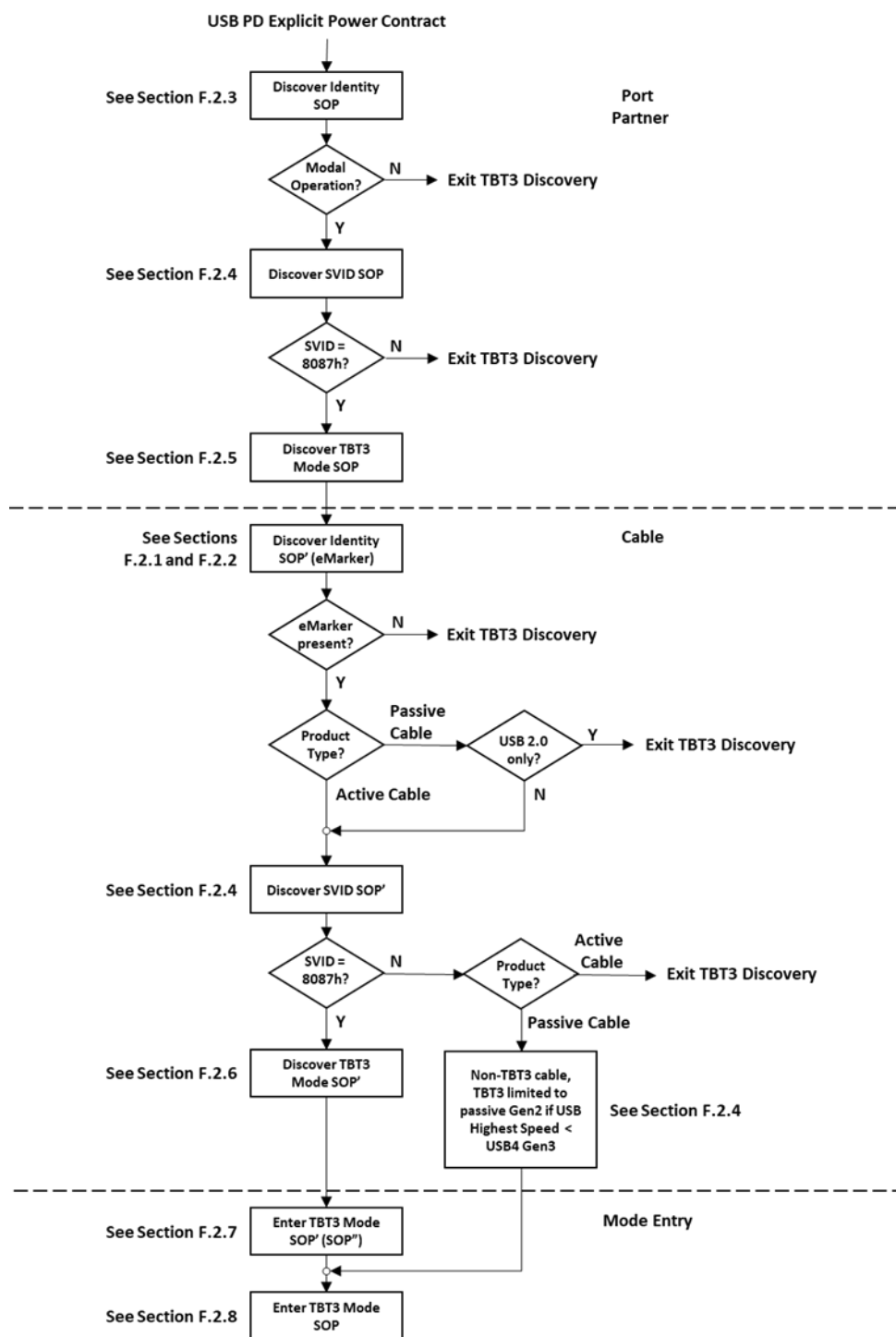
F.1.6 TBT3-Compatible Devices with a Captive Cable

TBT3-compatible devices with a captive cable **shall** respond to **USB PD** messages both SOP and SOP'.

F.2 TBT3 Discovery and Entry Flow

Figure F-1 describes the flow for a **TBT3**-compatible DFP port to discover and enter into the **TBT3** Compatibility Mode of **USB4** with a connected UFP. For each functional block, refer to the sub-sections identified for additional details.

Figure F-1 TBT3 Discovery Flow



F.2.1 TBT3 Passive Cable Discover Identity Responses

Table F-1, along with Table F-2 and Table F-3, defines the expected **Discover Identity** VDO responses for a **TBT3** passive cable. In fields where multiple values are listed, cable responses *may* vary to match the specific connected cable's capabilities.

Table F-1 TBT3 Passive Cable Discovery Identity VDO Responses

Message Header							
Rsvd	Number of Objects	Message ID	Cable Plug		Spec Revision	Rsvd	Message Type
0	5...6	0...7	1 = Cable Plug		10b or 01b	0	1111b
VDM Header							
SVID	VDM Type	VDM Version	Rsvd	Object Position	Command Type	Rsvd	Command
0xFF00	1	01b	0	000b	001b	0	00001b

Bit(s)	Value	Parameter
ID Header VDO		
B31	0	USB Communications Capable as USB Host
B30	0	USB Communications Capable as a USB Device
B29...27	011b = Passive Cable	Product Type (Cable Plug)
B26	1 = Modal Operation	Modal Operation Supported
B25...23	000b	Product Type (DFP)
B22...16	0	Reserved
B15...0	Per vendor	USB Vendor ID
Cert Stat VDO		
B31...0	0x00000000...0xFFFFFFFF	XID assigned by USB-IF
Product VDO		
B31...16	0x0000...0xFFFF	USB Product ID
B15...0	0x0000...0xFFFF	bcdDevice
Passive Cable VDO		
Depending on USB PD Specification Revision: See Table F-2 for USB PD Revision 2.0 or Table F-3 for USB PD Revision 3.0		

Table F-2 TBT3 Passive Cable VDO for USB PD Revision 2.0, Version 1.3

Bit(s)	Value	Parameter
B31...28	0000b...1111b	HW Version
B27...24	0000b...1111b	Firmware Version
B23...20	0	Reserved
B19...18	10b = USB Type-C	USB Type-C plug to USB Type-C/Captive
B17	0	Reserved
B16...13	0001b – <10ns (~1m) 0010b – 10ns to 20ns (~2m)	Cable Latency
B12...11	00b = VCONN not required	Cable Termination Type
B10	0 = Fixed	SSTX1 Directionality Support
B9	0 = Fixed	SSTX2 Directionality Support
B8	0 = Fixed	SSRX1 Directionality Support
B7	0 = Fixed	SSRX2 Directionality Support
B6...5	01b = 3A 10b = 5A	VBUS Current Handling Capacity
B4	1 = Yes	VBUS through cable
B3	0 = No	SOP” controller present
B2...0	010b = [USB 3.1] Gen1 and Gen2	SuperSpeed USB Signaling Support

Table F-3 TBT3 Passive Cable VDO for USB PD Revision 3.0, Version 1.2

Bit(s)	Value	Parameter
B31...28	0000b...1111b	HW Version
B27...24	0000b...1111b	Firmware Version
B23...21	000b = Version 1.0	VDO Version
B20	0	Reserved
B19...18	10b = USB Type-C	USB Type-C plug to USB Type-C/Captive
B17	0	Reserved
B16...13	0001b – <10ns (~1m) 0010b – 10ns to 20ns (~2m)	Cable Latency
B12...11	00b = VCONN not required	Cable Termination Type
B10...9	00b = 20V	Maximum VBUS Voltage
B8...7	00b	Reserved
B6...5	01b = 3A 10b = 5A	VBUS Current Handling Capacity
B4...3	00b	Reserved
B2...0	010b = [USB 3.2] Gen1 and Gen2	SuperSpeed USB Signaling Support

F.2.2 TBT3 Active Cable Discover Identity Responses

Table F-4, along with Table F-5, Table F-6 and Table F-7, defines the expected **Discover Identity** VDO responses for a **TBT3** active cable. In fields where multiple values are listed, cable responses **may** vary to match the specific connected cable's capabilities.

Table F-4 TBT3 Active Cable Discovery Identity VDO Responses

Message Header							
Rsvd	Number of Objects	Message ID	Cable Plug	Spec Revision	Rsvd	Message Type	
0	5...6	0...7	1 = Cable Plug	10b or 01b	0	1111b	
VDM Header							
SVID	VDM Type	VDM Version	Rsvd	Object Position	Command Type	Rsvd	Command
0xFF00	1	01b	0	000b	001b	0	00001b

Bit(s)	Value	Parameter
ID Header VDO		
B31	0	USB Communications Capable as USB Host
B30	0	USB Communications Capable as a USB Device
B29...27	100b = Active Cable	Product Type (Cable Plug)
B26	1 = Modal Operation Supported	Modal Operation Supported
B25...23	000b	Product Type (DFP)
B22...16	0	<i>Reserved</i>
B15...0	Per vendor	USB Vendor ID
Cert Stat VDO		
B31...0	0x00000000...0xFFFFFFFF	XID assigned by <i>USB-IF</i>
Product VDO		
B31...16	0x0000...0xFFFF	USB Product ID
B15...0	0x0000...0xFFFF	bcdDevice
Active Cable VDO 1		
Depending on <i>USB PD</i> Specification Revision: See Table F-5 for <i>USB PD</i> Revision 2.0 or Table F-6 for <i>USB PD</i> Revision 3.0		
Active Cable VDO 2		
Applicable only for <i>USB PD</i> Revision 3.0 – See Table F-7		

Table F-5 TBT3 Active Cable VDO for USB PD Revision 2.0, Version 1.3

Bit(s)	Value	Parameter
B31...28	0000b...1111b	HW Version
B27...24	0000b...1111b	Firmware Version
B23...20	0	Reserved
B19...18	10b = USB Type-C	USB Type-C plug to USB Type-C/Captive
B17	0	Reserved
B16...13	0001b – <10ns (~1m)	Cable Latency
B12...11	11b = Both ends Active, VCONN required	Cable Termination Type
B10	0 = Fixed	SSTX1 Directionality Support
B9	0 = Fixed 1 = Configurable	SSTX2 Directionality Support
B8	0 = Fixed 1 = Configurable	SSRX1 Directionality Support
B7	0 = Fixed 1 = Configurable	SSRX2 Directionality Support
B6...5	01b = 3A 10b = 5A	VBUS Current Handling Capacity
B4	1 = Yes	VBUS through cable
B3	0 = No 1 = Yes	SOP' controller present
B2...0	010b = [USB 3.1] Gen1 and Gen2	SuperSpeed USB Signaling Support

Table F-6 TBT3 Active Cable VDO 1 for USB PD Revision 3.0, Version 1.2

Bit(s)	Value	Parameter
B31...28	0000b...1111b	HW Version
B27...24	0000b...1111b	Firmware Version
B23...21	010b = Version 1.2	VDO Version
B20	0	Reserved
B19...18	10b = USB Type-C	USB Type-C plug to USB Type-C/Captive
B17	0	Reserved
B16...13	xxxxb (e.g., 0010b – 10ns to 20ns (~2m))	Cable Latency
B12...11	11b = Both ends Active, VCONN required	Cable Termination Type
B10...9	00b = 20V	Maximum VBUS Voltage
B8	1 = Yes	SBU Supported
B7	0 = Passive 1 = Active	SBU Type
B6...5	01b = 3A 10b = 5A	VBUS Current Handling Capacity
B4	0 = No 1 = Yes	VBUS through cable
B3	0 = No 1 = Yes	SOP' controller present
B2...0	010b = [USB 3.2] Gen1 and Gen2	SuperSpeed USB Signaling Support

Table F-7 TBT3 Active Cable VDO 2 for USB PD Revision 3.0, Version 1.2

Bit(s)	Value	Parameter
B31...24	xxxxxxx (e.g., 10101010b = 170 °C)	Maximum Operating Temperature
B23...16	xxxxxxx (e.g., 11010010b = 210 °C)	Shutdown Temperature
B15	0	Reserved
B14...12	000b ≥ 10 mW	U3 Power
B11	0	U3 to U0 Transition mode
B10...8	000b	Reserved
B7...6	00b	USB 2.0 Hub Hops Consumed
B5	0 = Supported	USB 2.0 Supported
B4	0 = Supported	SuperSpeed Supported
B3	1 = two lanes	SuperSpeed Lanes Supported
B2	0	Reserved
B1...0	00b = Gen1 01b = Gen2	SuperSpeed Signaling

F.2.3 TBT3 Device Discover Identity Responses

Table F-8 defines the expected **Discover Identity** VDO responses for a **TBT3** device. In fields where multiple values are listed, device responses *may* vary to match the specific connected device's capabilities.

Table F-8 TBT3 Device Discovery Identity VDO Responses

Message Header							
Rsvd	Number of Objects	Message ID	Cable Plug		Spec Revision	Rsvd	Message Type
0	4	0...7	0 = UFP		10b or 01b	0	1111b
VDM Header							
SVID	VDM Type	VDM Version	Rsvd	Object Position	Command Type	Rsvd	Command
0xFF00	1	01b	0	000b	001b	0	00001b

Bit(s)	Value	Parameter
ID Header VDO		
B31	0 or 1	USB Communications Capable as USB Host
B30	0 or 1	USB Communications Capable as a USB Device
B29...27	001b = PDUSB Hub 010b = PDUSB Peripheral 101b = Alternate Mode Adapter (AMA) 110b = VCONN-Powered USB Device (VPD)	Product Type (UFP)
B26	1 = Modal Operation Supported	Modal Operation Supported
B25...23	001b = PDUSB Hub 010b = PDUSB Host 100b = Alternate Mode Controller (AMC)	Product Type (DFP)
B22...16	0	Reserved
B15...0	Per vendor	USB Vendor ID
Cert Stat VDO		
B31...0	0x00000000...0xFFFFFFFF	XID assigned by USB-IF
Product VDO		
B31...16	0x0000...0xFFFF	USB Product ID
B15...0	0x0000...0xFFFF	bcdDevice

F.2.4 TBT3 Discover SVID Responses

Table F-9 defines the expected Discover SVID VDO responses for a **TBT3** device or cable.

Table F-9 TBT3 Discover SVID VDO Responses

Message Header							
Rsvd	Number of Objects	Message ID	Cable Plug		Spec Revision	Rsvd	Message Type
0	3	0...7	0 = UFP		10b or 01b	0	1111b
VDM Header							
SVID	VDM Type	VDM Version	Rsvd	Object Position	Command Type	Rsvd	Command
0xFF00	1	01b	0	000b	001b	0	00010b

Bit(s)	Value	Parameter
VDO 1		
B31...16	0x8087 = Intel/ <i>TBT3</i>	SVID 0
B15...0	0xFF01 = VESA DP (if supported)	SVID 1
VDO 2		
B31...16	0x0000 – 0xFFFF (normally the cable manufacturer VID)	SVID 2
B15...0	0x0000 (normally)	SVID 3

If the Intel/**TBT3** SVID of 0x8087 is not returned in response to the Discover SVID command, the cable is a Non-**TBT3** cable.

If a Non-**TBT3** cable's Product Type is Active Cable, it **shall** be regarded as not compatible with **TBT3**, and **TBT3** Discovery **shall** exit.

If a Non-**TBT3** cable's Product Type is Passive Cable, the USB Highest Speed field in the cable's Passive Cable VDO **shall** determine **TBT3** functionality and speed. If USB Highest speed is "**USB4** Gen3" or higher, the cable **shall** be regarded as a **TBT3** capable cable at Gen3 performance. If USB Highest speed is "**USB 3.2** Gen1" or "**USB 3.2/USB4** Gen2", it **shall** be regarded as a **TBT3** capable cable limited to passive Gen2 performance. If USB Highest Speed indicates "**USB 2.0**-only, No SuperSpeed", **TBT3** Discovery **shall** exit.

Note: Legacy **TBT3** platforms **may not** recognize **USB4** Gen3 or higher passive cables that don't also include the **TBT3** Passive Cable **Discover Identity** VDOs. When this happens, the **USB4** Gen3 or higher passive cable will still function but will only be used at Gen2 speeds.

F.2.5 TBT3 Device Discover Mode Responses

Table F-10 defines the expected Discover Mode VDO responses for a **TBT3** device. In fields where multiple values are listed, device responses *may* vary to match the specific connected device's capabilities.

Table F-10 TBT3 Device Discover Mode VDO Responses

Message Header							
Rsvd	Number of Objects	Message ID	Cable Plug		Spec Revision	Rsvd	Message Type
0	2	0...7	0 = UFP		10b or 01b	0	1111b
VDM Header							
SVID	VDM Type	VDM Version	Rsvd	Object Position	Command Type	Rsvd	Command
0x8087	1	01b	0	000b	001b	0	00011b

Bit(s)	Value	Parameter
TBT3 SOP VDO		
B31...30	00b	<i>Reserved</i> for legacy use
B29...27	000b	<i>Reserved</i>
B26	Defined by the Intel TBT3 specification.	Intel defined
B25...17	000000000b	<i>Reserved</i>
B16	0 = TBT3 Adapter 1 = TBT2 Legacy Adapter	TBT Adapter
B15...0	0x0001 = TBT Mode	TBT Alternate Mode

Description of the TBT3 Mode VDO Response Bits:

- B31 and B30: **Shall** be set to 00b.
- B26: Intel defined.

F.2.6 TBT3 Cable Discover Mode Responses

Table F-11 defines the expected Discover Mode VDO responses for a **TBT3** cable. In fields where multiple values are listed, cable responses *may* vary to match the specific connected cable's capabilities.

Table F-11 TBT3 Cable Discover Mode VDO Responses

Message Header							
Rsvd	Number of Objects	Message ID	Cable Plug		Spec Revision	Rsvd	Message Type
0	2	0...7	1 = Cable Plug		10b or 01b	0	1111b
VDM Header							
SVID	VDM Type	VDM Version	Rsvd	Object Position	Command Type	Rsvd	Command
0x8087	1	01b	0	000b	001b	0	00011b

Bit(s)	Value	Parameter
TBT3 SOP' VDO		
B31...26	00000000b	Reserved
B25	0 = Passive cable 1 = Active cable	Active_Passive
B24	0b	Reserved
B23	0 = Active with bi-directional LSRX ¹ communication or when Passive 1 = Active with uni-directional LSRX ¹ communication	Active Cable Plug Link Training
B22	0 = Not re-timer 1 = Re-timer	Re-timer
B21	0 = Non-Optical 1 = Optical	Cable Type
B20...19	00b = 3 rd Gen Non-Rounded TBT 01b = 3 rd & 4 th Gen Rounded and Non-Rounded TBT 10b...11b = Reserved	TBT_Rounded_Support
B18...16	000b = Reserved 001b = USB3.1 Gen1 Cable (10 Gbps TBT support) 010b = 10 Gbps (USB 3.2 Gen1 and Gen2 passive cables) 011b = 10 Gbps and 20 Gbps (TBT 3 rd Gen active cables and 20 Gbps passive cables) 100b...111b = Reserved	Cable Speed
B15...0	0x0001 = TBT Mode	TBT Alternate Mode

Notes:

1. LSRX in TBT3 is the same communication channel as SBRX in USB4.

F.2.7 TBT3 Cable Enter Mode Command

Table F-12 defines the Enter Mode Command that *shall* be sent to the SOP' (and if needed the SOP'') of an active **TBT3** cable to enable the cable for **TBT3** operation.

Table F-12 TBT3 Cable Enter Mode Command

Message Header							
Rsvd	Number of Objects	Message ID	Cable Plug		Spec Revision	Rsvd	Message Type
0	1	0...7	1 = Cable Plug		10b or 01b	0	1111b
VDM Header							
SVID	VDM Type	VDM Version	Rsvd	Object Position	Command Type	Rsvd	Command
0x8087	1	01b	0	000b	000b	0	00100b

F.2.8 TBT3 Device Enter Mode Command

Table F-13 defines the Enter Mode Command that *shall* be sent to the SOP of a **TBT3** device to enable the device for **TBT3** operation.

Table F-13 TBT3 Device Enter Mode Command

Message Header							
Rsvd	Number of Objects	Message ID	Cable Plug		Spec Revision	Rsvd	Message Type
0	2	0...7	0 = UFP		10b or 01b	0	1111b
VDM Header							
SVID	VDM Type	VDM Version	Rsvd	Object Position	Command Type	Rsvd	Command
0x8087	1	01b	0	000b	000b	0	00100b

Bit(s)	Value	Parameter
TBT3 SOP VDO		
B31...30	00b	<i>Reserved</i> for legacy use
B29...27	000b	<i>Reserved</i>
B26	Defined by the Intel TBT3 specification.	Intel Defined
B25	0 = Passive cable 1 = Active cable	Active_Passive
B24	0 = TBT3 Adapter 1 = TBT2 Legacy Adapter	TBT Adapter
B23	0 = Active with bi-directional LSRX ¹ communication or when Passive 1 = Active with uni-directional LSRX ¹ communication	Active Cable Link Training
B22	0 = Not re-timer 1 = Re-timer	Re-timer
B21	0 = Non-Optical 1 = Optical	Cable Type
B20...19	00b = 3 rd Gen Non-Rounded TBT 01b = 3 rd & 4 th Gen Rounded and Non-Rounded TBT 10b...11b = Reserved	TBT_Rounded_Support
B18...16	000b = Reserved 001b = USB3.1 Gen1 Cable (10 Gbps TBT support) 010b = 10 Gbps (USB 3.2 Gen1 and Gen2 passive cables) 011b = 10 Gbps and 20 Gbps (TBT 3 rd Gen active cables and 20 Gbps passive cables) 100b...111b = Reserved	Cable Speed
B15...0	0x0001 = TBT Mode	TBT Alternate Mode

Notes:

1. LSRX in **TBT3** is the same communication channel as SBRX in **USB4**.

The values to be used when sending the **TBT3** Device Enter Mode command to the SOP of a **TBT3** device are determined based on information retained from earlier in the discovery flow as follows:

- B31 and B30: The DFP *shall* set these bits to 0b. The UFP *shall* ignore these bits.
- B29...27: The DFP *shall* set these bits to 0b. The UFP *shall* ignore these bits.
- B26: The DFP *shall* set this bit to 0b unless otherwise instructed by Intel to set to 1b.

- B25: return the value received in the B25 field of the **TBT3** Cable Discover Mode Response.
- B24: return the value received in the B16 field of the **TBT3** Device Discover Mode Response.
- B23: if using a **TBT3** cable, return the value received in the B23 field of the **TBT3** Cable Discover Mode Response, otherwise set to 0.
- B22: if using a **TBT3** cable, return the value received in the B22 field of the **TBT3** Cable Discover Mode Response, otherwise set to 0.
- B21: if using a **TBT3** cable, return the value received in the B21 field of the **TBT3** Cable Discover Mode Response, otherwise set to 0.
- B20...19: if using a **TBT3** cable, return the value received in the B20...19 field of the **TBT3** Cable Discover Mode Response, otherwise set to 00b.
- B18...16: if using a **TBT3** cable, return the value received in the B18...16 field of the **TBT3** Cable Discover Mode Response, otherwise set to 010b.

F.2.9 TBT3 Cable Functional Difference Summary

Table F-14 provides a summary of existing **TBT3** cables and the unique functional differences between them.

Table F-14 TBT3 Cable Functional Difference Summary

Cable	Function					SOP' Configuration					
						ID Header VDO	Discover Mode (8087)				
	USB2	USB3	TBT3 Limitations	USB4	DP	Passive/ Active B29...27	Re- timer B22	Passive/ Active B25	Uni/Bi Directional LSRX ¹ B23	Rounded/ none B20...19	Optical/ none B21
Passive	Yes	Yes		Yes	Yes	011b	0b	0b	N/A (0b)	N/A (0b)	0b
TBT3 Re-timer	Yes	No	TBT3 Legacy ³	No	No	100b	1b	0b	0b	00b	0b
USB4 Re- Timer (with TBT3)	Yes	Yes		Yes	<i>Optional</i>	100b	1b	0b/1b	1b	01b	0b
USB4 Re- Driver (with TBT3)	Yes	Yes		Yes	<i>Optional</i>	011b ²	0b	1b	1b	01b	0b
TBT3 Limit Optical	No	No	No CLx No CC ⁴	No	No	100b	0b	0b	1b	00b	1b
Linear Optical Re- Driver	No	Yes		Yes	No	100b	0b	1b	1b	01b	1b
Notes: 1. LSRX in TBT3 is the same communication channel as SBRX in USB4 . 2. This cable is an active cable, however, to support backward compatibility with TBT3 legacy devices, B29...27 <i>should</i> be set to 011b. 3. Per USB4 Chapter 13 definition. 4. This cable does not support end-to-end USB PD communication.											

Notes:

1. **TBT3** Re-timer cables only support **TBT3** and does not support **USB4** operation.
2. All other re-timer cables are as defined in Chapter 6.
3. Limit Optical cables are as defined in Chapter 6 as optically-isolated active cables.

G Extracting Pulse Response from Sampled Data and Calculating Non-Linearity Noise

The following procedure is used to determine the linear fit pulse response and error for Linear Re-Driver (LRD) based cables.

1. The transmitter **shall** be configured to transmit PRBS15 pattern (as defined in **USB4** Specification section 4.2.1.3.4).
2. Extract the linear fit pulse from the measured waveform using the parameters specified in Table G-1.
3. Define an input pattern $x(n)$ to be a single PRBS period of length N_{seq} and an output signal to be the captured waveform $y(n)$, sampled at M times the signal baud rate.
4. Average the captured waveform at intervals of 2 PRBS repetitions ($2 \cdot N \cdot M$ samples) for filtering out uncorrelated noise.
5. Correlate the averaged waveform and the reference input pattern for extracting output signal $y_1(n)$ aligned to the input pattern $x(n)$ ($N \cdot M$ samples).
6. Concatenate the post-cursor input pattern corresponding to the first waveform sample at the left of the input vector $x(n)$, and the pre-cursor input pattern corresponding to the last waveform sample at the right of the input vector as following:

$$x_1[n] = [\{x(N-N_{post}+1), x(N-N_{post}+2), \dots, x(N)\}, \{x(1), x(2), \dots, x(N)\}, \{x(1), x(2), \dots, x(N_{pre})\}]$$

7. Zero pad $x_1(n)$ to yield $x_z(n)$ such that $M-1$ zeros are inserted between each adjacent entry, before the first entry and after the last entry of $x_1(n)$.
8. Present the output signal $y_1(n)$ as the convolution of $x_z(n)$ and FIR filter $h(n)$ containing $N_{taps} \cdot M$ coefficients:

$$y_1(n) = \sum_{k=0}^{N_{taps} \cdot M} x_z(n-k) \cdot h(k)$$

and in matrix representation:

$$y_1^{[(N_{seq} \cdot M) \times 1]} = X_z^{[(N_{seq} \cdot M) \times (N_{taps} \cdot M)]} \cdot h^{[(N_{taps} \cdot M) \times 1]}$$

$$X_z = \begin{bmatrix} x_z(N_{taps} \cdot M) & x_z(N_{taps} \cdot M - 1) & \dots & x_z(3) & x_z(2) & x_z(1) \\ x_z(N_{taps} \cdot M + 1) & x_z(N_{taps} \cdot M) & \dots & x_z(4) & x_z(3) & x_z(2) \\ x_z(N_{taps} \cdot M + 2) & x_z(N_{taps} \cdot M + 1) & \dots & x_z(5) & x_z(4) & x_z(3) \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ x_z(N_{seq} \cdot M) & x_z(N_{seq} \cdot M - 1) & \dots & x_z(N_{seq} \cdot M - N_{taps} \cdot M + 1) & \dots & \dots \end{bmatrix}$$

9. Extract the filter h coefficients by applying least-squares fitting:

$$h = [X_z^T \cdot X_z]^{-1} \cdot X_z^T \cdot y_1$$

where the superscript “T” denotes the matrix transpose operation.

10. Extract the linear fitting error waveform:

$$e = y_1 - X_z \cdot h$$

11. Since e has M phases, need to calculate e for each phase.

- a. Align the e vector to be in length of $M \cdot N$

- b. Create a matrix $E^{[NxM]}$ such as $E = \begin{bmatrix} e_1 & e_2 & \dots & e_M \\ e_{M+1} & e_{M+2} & \dots & e_{2M} \\ e_{2M+1} & e_{2M+2} & \dots & e_{3M} \\ \vdots & \vdots & \vdots & \vdots \\ e_{NM+1} & e_{NM+2} & \dots & e_{NM+M} \end{bmatrix}$
- c. Calculate the standard deviation of each phase. $e_{std} = std(E)$
- d. $\sigma_e = \max(e_{std})$
- e. The value **should** be normalized to the input swing: $\sigma_e = \frac{0.4}{TX_{amp}} \cdot \sigma_e$

The following parameters **shall** be used in the linear fit pulse calculation:

Table G-1 Linear Fit Pulse Extraction Parameters

Parameter	Value
Ntaps	100
Npost	Ntaps – Npre – 1
Npre	5
M	8

H USB PD High-Voltage Design Considerations

This appendix considers the impact of having higher voltages across the USB Type-C interface. It addresses contact lifetime and product safety when the USB Type-C cable gets unplugged while high-voltage operation is still enabled. It is applicable for the **USB Power Delivery** defined Extended Power Range (EPR) voltages (28 V – 48 V) and the Standard Power Range (SPR) at 20 V when supply current is high.

H.1 Potential for Arcing Damage During Cable Withdrawal

Arcing can occur when the connector is unplugged if the voltage differential across the gap between the plug and receptacle VBUS contacts is greater than 12 volts. Figure H-1 shows examples of the damage that could result from arcing.

Figure H-1 Arcing Damage to USB Type-C VBUS Contacts



The following conditions can result in arcing when the cable is withdrawn:

1. Source
 - Voltage regulation when the load is suddenly removed.
2. Sink
 - Length of time for the Sink to hold the voltage on its VBUS contact.
3. Cable
 - The inductive kick on VBUS occurs in sub-100 ns.
 - Ringing on VBUS occurs in microseconds.
 - The loss of IR voltage drop from the source to the plug due to load removal occurring in 0.1 to 1 μ s.

An arc can be formed when the voltage difference between the Source and Sink across the gap between connector contacts is as low as 12 volts. The arcing voltage of 12 volts only applies within a gap distance of less than 7.5 – 10 μ m at normal atmospheric pressures. Above this gap distance, the arcing voltage increases significantly. Assuming 12 volts across the entire range provides a significant margin for analysis and application.

H.2 Arcing During USB Type-C Cable Withdrawal

There are two separate mechanisms that can create the voltage differential necessary to arc and with enough current can potentially damage contacts through excessive heating.

1. Inductive kickback
2. Sink discharge

Even before either of these mechanisms occur, there is initial heating because of all the current being channeled through a very small contact point where the power density creates enough heat to potentially melt metal.

Inductive Kickback

The first arcing mechanism is due to inductive kickback which can readily create a voltage delta of 12 volts or more. This event starts at contact break and lasts less than approximately 100 ns. Inductive kickback induced arcing occurs at any VBUS voltage – it happens regardless of the starting DC voltage on VBUS. This arcing has not been seen to cause long term damage to USB Type-C cables in the past as the current is likely too low to super heat the metal (beyond forming a temporary micron-sized molten bridge) to a point where it is permanently destructive. Calculating the energy of the inductive arc as $\frac{1}{2} Li^2$ results in approximately 5 µjoules which is too low of an energy to damage the metal and correlates well with observation over the lifetime of USB Type-C connections in practice.

Sink Discharge

The second mechanism creating a voltage differential that can result in arcing is due to the discharge of the voltage at the Sink-side VBUS contact while the Source-side VBUS contact remains high. This arcing mechanism is the one with the most potential to create and sustain an arc at high enough current to heat and damage the connector contacts.

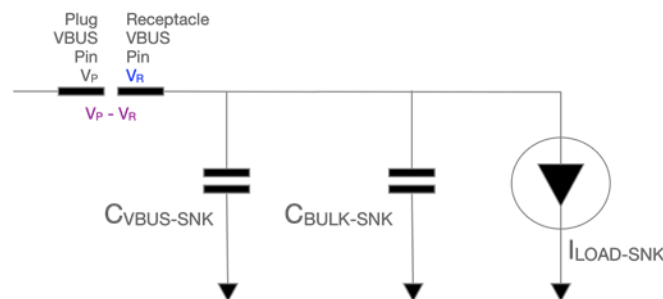
This analysis will assume that the disconnect occurs at the Sink end of a cable, but a disconnect at the Source end has the same effect.

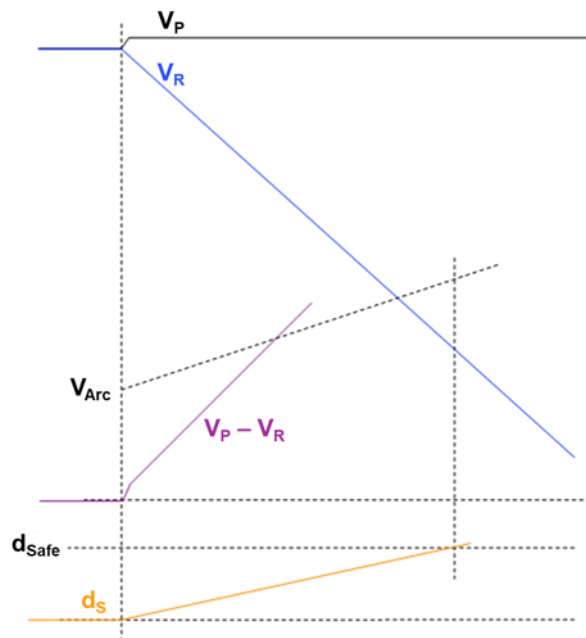
When the disconnect occurs, the Source will continue to supply power until it detects that the Sink has been disconnected which **may** take up to 650 ms ($t_{VBUSOFF}$). The VBUS contact voltage on the Source-side **may** quickly ($\sim 1 \mu s$) step up in voltage due to load regulation and the elimination of any IR voltage drop through the cable. At the same time, the Sink-side contact voltage will discharge due to the load current until the Sink detects the disconnect and removes the load. This creates a voltage difference that is increasing in time.

The withdrawal velocity is a factor in whether an arc will occur or not. If it is fast enough, then there is insufficient time to reach the voltage differential needed to form an arc. In practice, the withdrawal rate **may not** always be fast enough to keep the differential voltage below the threshold of arcing. In essence, there is a race between the contacts reaching a safe distance such that arcing will not occur at the voltages within the USB Type-C range and the voltage differential between the pins reaching the arcing voltage of 12 V.

Figure H-2 shows the potential for arcing due to Sink Discharge. At disconnect, the Source-side plug contact voltage V_P increases in voltage while the Sink-side receptacle contact starts decreasing in voltage, thus the differential voltage across the contact gap begins increasing. Note, the discharge rate of the Sink-side contact voltage is determined by the Sink load current and the Sink VBUS and **USB PD** bulk capacitance ($c_{SnkBulkPd}$).

Figure H-2 Arcing Due to Discharge





In Figure H-2, the graph shows the voltage difference $V_P - V_R$ reaching the potential arcing voltage V_{Arc} before the contact separation d_S reaches the safe distance d_{Safe} . This would result in an arc forming and damaging the pins if the current is sufficient.

The calculated energy of this event if V_{Arc} is 15 Volts for example for 1 ms would result in 75 milli-joules which will boil the surface of the contacts resulting in significant damage.

H.3 Mitigating Arcing Damage During Cable Withdrawal Due to Sink Discharge

The goal of arcing mitigation is not necessarily to entirely prevent arcing but to prevent damage to the connector pins due to arcing that *may* still occur. An arc *may* occur without damaging the connector pins if the energy of the arcing is sufficiently low. Experimental data suggests that arcing with a current less than 1 A does not generate enough heat to damage the connector pins.

To mitigate arcing damage due to Sink discharge, the voltage between the disconnected contacts must not reach the arcing voltage of 12 volts until the distance between the contacts reaches a safe distance or the current sinking capability of the Sink must be sufficiently low at the time of arcing. The actual arcing voltage increases significantly as the gap distance increases and is not constant and has been seen to range from 12 volts at 0 μm gap distance to as high as 300 volts with a gap distance of 7.5 – 10 μm . Assuming the minimum of 12 volts throughout gives margin and is a practical design target. Likewise, the safe distance is somewhere between 7.5 – 10 μm therefore assuming 20 μm for the following analyses gives plenty of margin.

To help mitigate arcing due to Sink discharge, the Sink *should* manage the discharge slew rate in combination with detecting the disconnection and internally disconnecting its load. Given that it is practical, a Sink design could solely focus on limiting the slew rate to a safe level versus designing with a functional balance between the load capacitance and the time needed to detect the disconnect and remove the device's primary load. However, unplug speed and resulting distance by a human is statistical. This means that the extraction speed

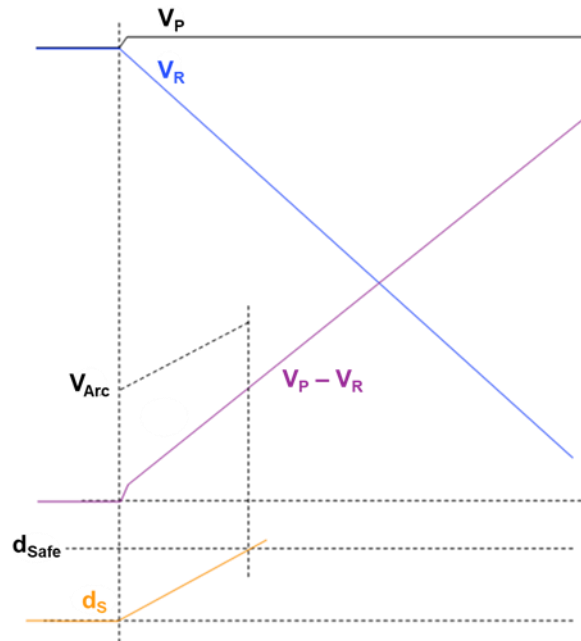
has a statistical chance to be very slow relative to the discharge time and has a statistical chance to stop at an unsafe distance. This means that there will always be observed arcing with enough unplug events.

The best approach is to limit the Sink discharge rate so that an arcing voltage will not be reached. While limiting the discharge speed in the Sink does mitigate the chance of an arc by itself, it **should** be used in conjunction with removing the load current. Both approaches are discussed in the following examples.

H.3.1 Limiting Sink Discharge Rate

Figure H-3 shows that arcing is prevented when the discharge rate of the Sink VBUS is slow enough such that the voltage difference $V_P - V_R$ between the contacts does not reach the arcing voltage V_{Arc} before the contact distance d_S reaches the safe distance d_{Safe} .

Figure H-3 Arcing Prevention During Sink Discharge by Limiting Slew Rate



The slew rate of the Sink VBUS discharge is set by the max load current in the Sink and the bulk capacitance on VBUS in the Sink. Increasing the bulk capacitance slows the discharge rate. With some limited testing, the disconnect velocity of a properly designed USB Type-C connector with retention springs has been observed to be as slow as 90 mm/s. It is assumed that there will be little degradation over lifetime due to the required minimum breaking force of the connector. The time to reach the safe distance d_S of 20 μm with a breaking velocity of 90 mm/s is 220 μs . For the timing requirement in Section 4.6.2.6, 250 μs is specified therefore in this analysis, the Sink must not discharge at a rate such that 12 V is reached before 250 μs after disconnect.

Note, the plug VBUS voltage will increase immediately at disconnect when there is load current. It must be assumed that the plug VBUS voltage **may** increase from being at the minimum Sink VBUS voltage for the **USB PD** contract at disconnect to the maximum Sink VBUS voltage of the **USB PD** contract minus the 750 mV defined in **vSinkPD_min**. This means that at disconnect, as the discharge of the Sink VBUS begins, the differential voltage **may** start anywhere between $V_{SinkPD_max} - V_{SinkPD_min}$.

A. Example of preventing arcing by slew rate limitation for a 20 Volt USB PD fixed contract with a max 5 A load current:

$d_S = 20 \mu\text{m}$ = Safe distance to prevent arcing.

$V_E = 90 \text{ mm/s}$ = Minimum contact breaking velocity.

$V_{Arc} = 12 \text{ V}$ = Minimum Arcing Voltage.

$V_{SinkPD_max} = 20 \text{ V} \times 1.05 = 21 \text{ V}$ = Maximum Sink voltage at disconnect.

$V_{SinkPD_min} = 20 \text{ V} \times 0.95 - 750\text{mV} = 18.25\text{V}$ = Minimum Sink voltage at disconnect.

$V_{Dis} = V_{SinkPD_max} - V_{SinkPD_min} = 2.75\text{V}$ = Maximum voltage between the contacts at disconnect.

$I_{Load} = 5 \text{ A}$.

$t_{Safe} = d_s / V_E = 250 \mu\text{s}$ = Time to reach the safe distance.

$dV_{max} = V_{Arc} - V_{Dis} = 12 \text{ V} - 2.75 \text{ V} = 9.25 \text{ V}$ = Maximum discharge voltage before reaching V_{Arc} .

$dV / dt = dV_{max} / t_{Safe} = 9.25 \text{ V} / 250 \mu\text{s} = 37 \text{ mV} / \mu\text{s}$ = Maximum slew rate.

$C_{Bulk} = I_{Load} / (dV / dT) = 5 \text{ A} / 37 \text{ mV} / \mu\text{s} = 135 \mu\text{F}$.

In this first example, the Sink Bulk Capacitance to prevent the arcing voltage from being reached before the contacts are at a safe distance is 135 μF .

B. Example of preventing arcing by slew rate limitation for a USB PD EPR 48V contract operating at 48 V with a max 5 A load current:

In this example, the analysis is essentially the same as the first example but using the highest voltage request for the given EPR voltage range, in this case 48 volts.

With all other parameters and assumptions remaining the same, the follow adjustments are made:

$V_{SinkPD_max} = 48 \text{ V} \times 1.05 = 50.4 \text{ V}$.

$V_{SinkPD_min} = 48 \text{ V} \times 0.95 - 750\text{mV} = 44.85 \text{ V}$.

$V_{Dis} = V_{SinkPD_max} - V_{SinkPD_min} = 5.55 \text{ V}$.

$dV_{max} = V_{Arc} - V_{Dis} = 12 \text{ V} - 5.55 \text{ V} = 6.45 \text{ V}$

$dV / dt = dV_{max} / t_{Safe} = 6.45 \text{ V} / 250 \mu\text{s} = 25.8 \text{ mV} / \mu\text{s}$.

$C_{Bulk} = I_{Load} / (dV / dT) = 5 \text{ A} / 25.8 \text{ mV} / \mu\text{s} = 194 \mu\text{F}$.

In this second example, the Sink Bulk Capacitance to prevent the arcing voltage from being reached before the contacts are at a safe distance is 194 μF .

Note, due to the nature of **USB PD** contracts, the starting differential voltage between the contacts V_{Dis} at disconnect increases with increasing nominal or variable contract. This results in the dV_{max} being lower as contract voltage gets higher. Thus, higher contract voltages will need to slow down the slew rate with higher C_{Bulk} or lower I_{Load} .

H.3.2 Load Removal

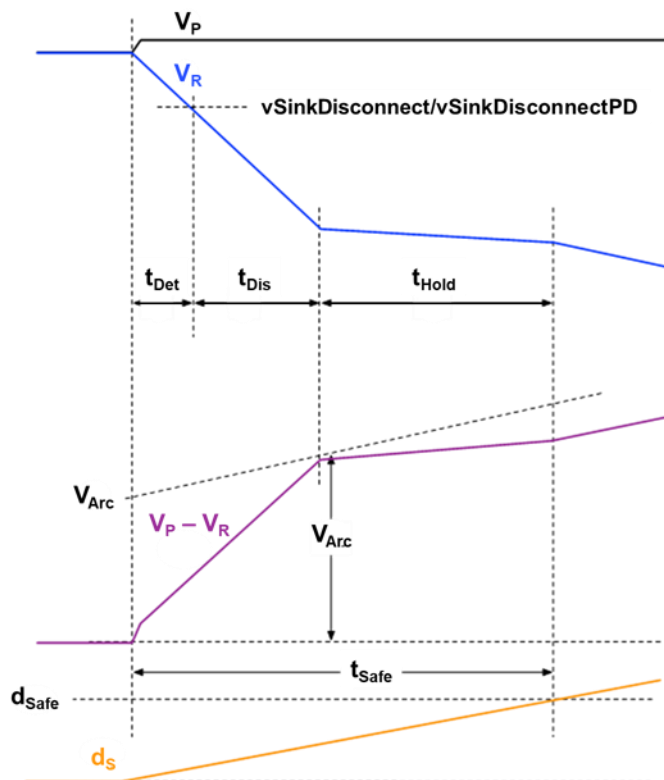
Quickly detecting the disconnect and reducing the Sink's primary load is an alternative to simply relying on managing the slew rate via adjusting the load capacitance. Based on the Sink requirements given in Section 4.6.2.6, the method analysis that follows is for detecting disconnect by the Sink based on monitoring the V_{BUS} voltage for a drop to below the defined value for ***vSinkDisconnectPD*** and disconnecting the Sink's primary load to reduce the current that flows through the arc if an arc occurs.

Note: As a secondary method and in addition to monitoring the V_{BUS} voltage for disconnect, monitoring the CC voltage **may** give the earliest indication of disconnection – this is due to the CC contacts separating before the V_{BUS} contacts in the connector design. The amount of time between V_{BUS} and CC contact breaks depends

upon the relative distance between the VBUS contact and the CC contact in the receptacle and the extraction speed. To make CC detection of the unplug most effective, a receptacle with a distance of at least 0.3 mm **should** be used (see Figure 3-1). For example, a system with at least 100 μF operational capacitance that only transmits mandatory **USB PD** messages (durations < 1.6 ms) and debounces the CC pin for no more than 200 μs will detect the detach before the VBUS contact breaks if it uses a 0.3 mm receptacle.

Figure H-4 shows an alternative to increasing the capacitance relative to the load current. In this example, the Sink detects the disconnect using **vSinkDisconnect** or **vSinkDisconnectPD** depending on the contract type to disconnect the load before V_{Arc} is reached.

Figure H-4 Arcing Prevention During Sink Discharge by Load Removal



In this example, three timings are introduced. t_{Det} is the time from disconnect to reaching the minimum detection voltage. t_{Dis} is the remaining time for the Sink VBUS to discharge after reaching the minimum detection voltage t_{Det} . The sink must remove the load current within t_{Dis} to stop the discharge before the differential voltage between the contacts reaches V_{Arc} . t_{Hold} is the remaining time the discharge of VBUS must be halted or reduced such that V_{Arc} is not reached until t_{Safe} has expired.

A. Example of preventing arcing by load removal for a 20 Volt USB PD fixed contract with a max 5 A load current:

In this analysis, some values are identical to the Limiting Sink Discharge example.

$$d_s = 20 \mu\text{m}.$$

$$V_E = 90 \text{ mm/s}.$$

$$V_{\text{Arc}} = 12 \text{ V}.$$

$$V_{\text{SinkPD_max}} = 21 \text{ V}.$$

$$V_{\text{SinkPD_min}} = 18.25 \text{ V}.$$

$$V_{Dis} = 2.75 \text{ V.}$$

$$I_{Load} = 5 \text{ A.}$$

$$t_{Safe} = 250 \text{ }\mu\text{s.}$$

$$dV \text{ max} = 9.25 \text{ V} - 1 \text{ V} = \text{Maximum discharge voltage with 1 V margin before reaching } V_{Arc}.$$

In this case, 1 V margin has been added to ensure that the sink disconnects before reaching V_{Arc} .

$$C_{Bulk} = 20 \text{ }\mu\text{F} = \text{Bulk capacitance in Sink.}$$

In this case, C_{Bulk} is chosen to be lower than the previously calculated minimum bulk capacitance to prevent arcing by slow discharge.

$$dV / dt = I_{Load} / C_{Bulk} = 5 \text{ A} / 20 \text{ }\mu\text{F} = 250 \text{ mV}/\mu\text{s} = \text{Maximum slew rate.}$$

Based on this maximum slew rate, the sum of ($t_{Det} + t_{Dis}$) can be calculated. The voltage detector and load disconnect switch can vary by implementation but the sum of these two processes needs to occur within this calculated total.

$$(t_{Det} + t_{Dis}) = dV \text{ max} / (dV / dt) = 8.25 \text{ V} / 250 \text{ mV}/\mu\text{s} = 33 \text{ }\mu\text{s} = \text{time from contact disconnect to removing the load.}$$

$$t_{Hold} = t_{Safe} - (t_{Det} + t_{Dis}) = 250 \text{ }\mu\text{s} - 33 \text{ }\mu\text{s} = 217 \text{ }\mu\text{s.}$$

In this example, to prevent arcing when the bulk capacitance is 20 μF , which is not enough to keep the differential voltage between the contacts from reaching the arcing voltage V_{Arc} before the contacts reach a safe distance, the load must be removed within 33 μs after V_{BUS} contacts start to disconnect and must be held from reaching the arcing voltage for another 217 μs .

B. Example of preventing arcing by load removal for a USB PD EPR 48V contract operating at 48 V with a max 5 A load current:

In this example, the analysis is essentially the same as the first example but using the highest voltage request for the given EPR voltage range, in this case 48 volts. In this example, the bulk capacitance used has been increased to better illustrate balancing the mitigation approach between limiting Sink discharge rate and Sink load removal.

With all other parameters and assumptions remaining the same, the follow adjustments are made:

$$V_{SinkPD_max} = 48 \text{ V} \times 1.05 = 50.4 \text{ V.}$$

$$V_{SinkPD_min} = 48 \text{ V} \times 0.95 - 750\text{mV} = 44.85 \text{ V.}$$

$$V_{Dis} = V_{SinkPD_max} - V_{SinkPD_min} = 5.55 \text{ V.}$$

$$dV \text{ max} = V_{Arc} - V_{Dis} - 1 \text{ V} = 12 \text{ V} - 5.55 \text{ V} - 1 \text{ V} = 5.45 \text{ V (includes the 1 V margin)}$$

$$C_{Bulk} = 100 \text{ }\mu\text{F} = \text{Bulk capacitance in Sink.}$$

$$dV / dt = I_{Load} / C_{Bulk} = 5 \text{ A} / 100 \text{ }\mu\text{F} = 50 \text{ mV}/\mu\text{s}$$

$$(t_{Det} + t_{Dis}) = dV \text{ max} / (dV / dt) = 5.45 \text{ V} / 50 \text{ mV}/\mu\text{s} = 109.0 \text{ }\mu\text{s}$$

$$t_{Hold} = t_{Safe} - (t_{Det} + t_{Dis}) = 250 \text{ }\mu\text{s} - 109.0 \text{ }\mu\text{s} = 141.0 \text{ }\mu\text{s.}$$

The slew rate in this example assumes the bulk capacitance is 100 μF resulting in the load removal to be completed within 109.0 μs and the needed hold time of 141.0 μs .

If the bulk capacitance were to be increased to 194 μF as calculated in the limiting Sink discharge rate Example B in Section H.3.1, the slew rate would decrease to 25.8 mV/ μs resulting in the load removal needing to be completed within 211.2 μs and the hold time decreased to 38.8 μs .

H.3.3 Limiting Source Current Capability

To further aid in mitigating the chance of damage from an arc, the Source *should* monitor for disconnect and remove the sourcing capability as well as source bulk capacitance as quickly as possible. Since the source does not have a mechanism to see the sink-side voltage, it has no direct way to determine when it *should* have the source and bulk capacitance disconnected from VBUS. The faster the removal of the source and bulk capacitance, the less chance there is for damage from an arc.

The simplest mechanism for detecting the disconnect is the defined monitoring of the CC voltage. Note, when the source is transmitting **USB PD** traffic, it cannot detect the disconnect with the CC voltage until the transmitted packet is finished. **USB PD** transmission from the source *should* be a relatively low percentage of connect time resulting in a statistically low chance of hitting this scenario. Combined with the Sink properly removing the load current for arc mitigation further reduces the chance. Another mechanism for detecting disconnect would be to monitor the load drop on VBUS. A disconnect load drop is much faster than the allowed load step from a sink defined in **USB PD**. Detection circuitry can be added that distinguishes the faster load drop such that the disconnect can be detected during **USB PD** traffic transmission.

I USB PD Encoding Guidelines for USB Type-C Product Types

This appendix provides guidance for the use of the UFP and DFP product types and UFP and DFP VDO responses for USB Type-C product examples. Refer to [Section 6.4.4.3.1 \(~~Discover Identity~~\)](#) in the *USB Power Delivery* specification for more information [on *Discovery Identity*](#).

I.1 USB Type-C Product Type Definitions and USB PD Encodings

When establishing a USB Type-C connection, the *Discover Identity* command *may* be received over the link between the two products being connected. It *should* be noted that an Explicit Contract must already exist between the port partners before the *Discover Identity* command *may* be initiated by either product.

The product that receives a *Discover Identity* Command responds with a *Discover Identity* Command ACK which will contain multiple pieces of information including the Product Types for its UFP, DFP or both (depending on applicability) and its UFP VDO, DFP VDO or both (depending on applicability). The encoding information described below and used in the tables that follow is only a portion of the overall Command ACK response – refer to the *USB PD* specification for the complete *Discover Identity* Command ACK response requirements.

When a product has multiple USB Type-C ports, these definitions are specific to the port that is being queried by its port partner. Typically, a product with multiple ports will expose the same functionality across all of its ports but there are some situations where this isn't the case, for example, a hub will describe its upstream port differently than its collection of downstream ports. In some cases, a port *may* have a special purpose that is not equivalent to the other exposed ports, for example, a user-identifiable port dedicated to charging a product whereas the other port or ports have data capabilities.

In this appendix, the following *Discover Identity* Command ACK response items are included.

- **UFP Product Types** (Ref. *USB PD* Section 6.4.4.3.1.1.3)
 - **Undefined** (000b) – The responding port is not a UFP, i.e., it has no device function or device controller connected to this port. This port also does not consume power although it *may* connect as a Sink for the purpose of establishing a functional relationship with a Source.
 - **PDUSB Hub** (001b) – The responding port has only upstream hub functionality connected to this port. No other device function or device controller is connected to this port. The response needs to also include a UFP VDO.
 - **PDUSB Peripheral** (010b) – The responding port has at least one device function or device controller connected to this port. The response needs to also include a UFP VDO.
 - **PSD** (011b) – The responding port has no data functionality but does consume power via this port.
- **DFP Product Types** (Ref. *USB PD* Section 6.4.4.3.1.1.6)
 - **Undefined** (000b) – The responding port is not a DFP, i.e., it had no host controller or hub downstream functionality connected to this port. This port also does not supply power to its port partner.
 - **PDUSB Hub** (001b) – The responding port has downstream hub functionality. The response needs to also include a DFP VDO.
 - **PDUSB Host** (010b) – The responding port has a host controller connected to this port. The response needs to also include a DFP VDO.
 - **Power Brick** (011b) – The responding port has no data functionality but does supply power via this port. The response needs to also include a DFP VDO.

- **UFP VDO – Device Capability** (Ref. **USB PD** Section 6.4.4.3.1.4)
 - **USB 2.0 Device Capable** – The responding port has at least one **USB 2.0** device function or device controller. This bit is specific to all device function types except a **USB 2.0** Billboard Device Class function. When the **USB 3.2** Device Capable bit is set, this bit is required to be set as well. When the **USB4** Device Capable bit is set, this bit **may** or **may not** be set depending on if at least one of the **USB4** Device Capable functions maps to a **USB 2.0** Device Class function.
 - **USB 2.0 Billboard only** – this bit is specific to the presence or not of a **USB 2.0** Billboard Device Class function. Other **USB 2.0** Device Capable functions **may** or **may not** exist, and if such a function does exist, the **USB 2.0** Device Capable bit will be used to indicate this.
 - **USB 3.2 Device Capable** – The responding port has at least one **USB 3.2** device function or device controller. When the **USB4** Device Capable bit is set, this bit **may** or **may not** be set depending on if at least one of the **USB4** Device Capable functions maps to a **USB 3.2** Device Class function.
 - **USB4 Device Capable** – The responding port is connected to the upstream port of a **USB4** device router with one or more **USB4** peripheral functions (i.e., **USB3**, **DisplayPort** or **PCIe**).
- **UFP VDO – Highest Speed** (Ref. **USB PD** Section 6.4.4.3.1.4)
 - 000b = **USB 2.0 only**, no SuperSpeed USB support.
 - 001b = **USB 3.2 Gen1** – **USB 2.0** is also supported.
 - 010b = **USB 3.2 Gen2 / USB4 Gen2** – **USB 2.0** and **USB 3.2** Gen1 are also supported. This encoding is used both for **USB 3.2** Gen2 or **USB4** Gen2 products, and if for a **USB4** Gen2 product, **USB 3.2** Gen2 is also supported.
 - 011b = **USB4 Gen3** – **USB 2.0**, **USB 3.2** Gen1, **USB 3.2** Gen2 and **USB4** Gen2 are also supported.
- **DFP VDO – Host Capability** (Ref. **USB PD** Section 6.4.4.3.1.5)
 - **USB 2.0 Host Capable** – The responding port is connected to the downstream side of a **USB 2.0** host controller. When either the **USB 3.2** Host Capable or **USB4** Host Capable bits are set, this bit is required to be set as well.
 - **USB 3.2 Host Capable** – The responding port is connected to the downstream side of a **USB 3.2** host controller. When the **USB4** Host Capable bit is set, this bit is required to be set as well.
 - **USB4 Host Capable** – The responding port is connected to the downstream side of a **USB4** host or device router.

I.2 USB PD Encoding Guidelines Tables

Table I-1 lists examples of USB Type-C-based products and provides brief clarifying notes for each. All of the included product examples implement **USB PD** as either a Source, Sink or both, and will use **USB PD** for other functional configuration (**USB4**, **Alternate Modes**, etc.) as appropriate to the product. This list of examples is not exhaustive and other valid combinations of features can also be described by the available **USB PD** encodings. Products that don't implement **USB PD** are not relevant to these guidelines.

Table I-2 lists the same examples (in the same order as Table I-1) and provides the recommended **USB PD** encoding information for the UFP and DFP Product Types along with the UFP and DFP VDOs fields associated with those product types.

Notes for these tables:

1. DC = USB Device Controller. This controller provides access to either **USB 2.0**, **USB 3.2** or **USB4** peripheral functionality when the port is operating in its UFP data role.

2. Upstream and Downstream are in reference to the physical port locations on either a hub or dock product. By definition, a hub has only one identifiable upstream port and will have one or more downstream ports. Docks included here are hubs with extended functionality.
3. For product types: PSDs are power sinks only and don't support a UFP data connection; Power Bricks are power sources only and don't support a DFP data connection.

Table I-1 USB Type-C Product Example Clarifying Notes

Product Example	Notes
USB 3.2 Host – no DC (e.g., Desktop PC)	Typical of a Desktop PC that provides power to attached peripherals. This product has no UFP device controller functionality, nor does it consume power via the port.
USB 3.2 Host – no DC – power consumer	Typical of a notebook PC that can provide power to attached peripherals but also can consume power over this port from either a hub, dock or charger. This product has no UFP device controller functionality.
USB 3.2 Host – USB 2.0 DC	This product has user-visible USB 2.0 -based UFP device controller functionality. This product may or may not consume power over this port from either a hub, dock or charger.
USB 3.2 Host – USB 3.2 DC	This product has user-visible USB 3.2 -based UFP device controller functionality that also will function over a USB 2.0 connection. This product may or may not consume power over this port from either a hub, dock or charger.
USB4 Host – no DC (e.g., Desktop PC)	Typical of a Desktop PC that provides power to attached peripherals. This product has no UFP device controller functionality, nor does it receive power via the port. Even though this host does not consume power, it will still behave as a DRP in order to be able to establish a USB4 data connection with another USB4 Host.
USB4 Host – no DC – power consumer	Typical of a notebook PC that can provide power to attached peripherals but also can consume power over this port from either a hub, dock, or charger. This product has no UFP device controller functionality.
USB4 Host – USB 2.0 DC	This product has user-visible USB 2.0 -based UFP device controller functionality. This product may or may not consume power over this port from either a hub, dock or charger.
USB4 Host – USB 3.2 DC	This product has user-visible USB 3.2 -based UFP device controller functionality that also will function over a USB 2.0 connection. This product may or may not consume power over this port from either a hub, dock or charger.
USB4 Host – USB4 DC	This product has user-visible USB4 -based UFP device router functionality and that functionality is also available when the port is operating in either a USB 2.0 or USB 3.2 connection. This product may or may not consume power over this port from either a hub, dock, or charger.
USB4 Host – USB4 DC – partial USB equiv.	This product has user-visible USB4 -based UFP device router functionality and only a portion of this functionality is available when the port is operating in either a USB 2.0 or USB 3.2 connection. This product may or may not consume power over this port from either a hub, dock, or charger.
USB4 Host – USB4 DC – no USB equiv.	This product has user-visible USB4 -based UFP device router functionality and none of this functionality is available when the port is operating in either a USB 2.0 or USB 3.2 connection. This product may or may not consume power over this port from either a hub, dock, or charger.
USB 3.2 Peripheral	This product provides one or more user-visible functions. Power for this product is provided either over the USB Type-C connection or by a separate power supply not associated with the USB connection. This product is not capable of supplying power to the other product.

Product Example	Notes
USB 3.2 Peripheral – power provider	This product provides one or more user-visible functions. Power for this product is provided either over the USB Type-C connection or by a separate power supply not associated with the USB connection. This product is also capable of supplying power to the other product.
USB4 Peripheral	This product provides one or more user-visible functions, and that functionality is also available when the port is operating in either a USB 2.0 or USB 3.2 connection. Power for this product is provided either over the USB Type-C connection or by a separate power supply not associated with the USB connection. This product is not capable of supplying power to the other product.
USB4 Peripheral – power provider	This product provides one or more user-visible functions, and that functionality is also available when the port is operating in either a USB 2.0 or USB 3.2 connection. Power for this product is provided either over the USB Type-C connection or by a separate power supply not associated with the USB connection. This product is also capable of supplying power to the other product.
USB4 Peripheral – partial USB equiv.	This product provides one or more user-visible functions and only a portion of this functionality is available when the port is operating in either a USB 2.0 or USB 3.2 connection. Power for this product is provided either over the USB Type-C connection or by a separate power supply not associated with the USB connection. This product is not capable of supplying power to the other product.
USB4 Peripheral – no USB equiv.	This product provides one or more user-visible functions and none of this functionality is available when the port is operating in either a USB 2.0 or USB 3.2 connection. Power for this product is provided either over the USB Type-C connection or by a separate power supply not associated with the USB connection. This product is not capable of supplying power to the other product.
USB 3.2 Hub (Upstream)	This is the dedicated upstream port of a USB 3.2 hub.
USB 3.2 Hub (Upstream) – power provider	This is the dedicated upstream port of a USB 3.2 hub. This port can also supply power.
USB 3.2 Hub (Downstream)	This is one of the downstream ports of a USB 3.2 hub.
USB 3.2 Hub (Downstream) – power consumer	This is one of the downstream ports of a USB 3.2 hub. This port can also consume power.
USB4 Hub (Upstream)	This is the dedicated upstream port of a USB4 hub.
USB4 Hub (Upstream) – power provider	This is the dedicated upstream port of a USB4 hub. This port can also supply power.
USB4 Hub (Downstream)	This is one of the downstream ports of a USB4 hub.
USB4 Hub (Downstream) – power consumer	This is one of the downstream ports of a USB4 hub. This port can also consume power.
USB4 Dock (Upstream)	This is the dedicated upstream port of a USB4 -based dock that includes a USB4 hub with one or more downstream USB Type-C ports. This product provides one or more user-visible functions, and that functionality is also available when the port is operating in either a USB 2.0 or USB 3.2 connection.
USB4 Dock (Upstream) – power provider	This is the dedicated upstream port of a USB4 -based dock that includes a USB4 hub with one or more downstream USB Type-C ports. This product provides one or more user-visible functions, and that functionality is also available when the port is operating in either a USB 2.0 or USB 3.2 connection. This port can also supply power.
USB4 Dock (Upstream) – partial USB equiv.	This is the dedicated upstream port of a USB4 -based dock that includes a USB4 hub with one or more downstream USB Type-C ports. This product provides one or more user-visible functions and that only a portion of this functionality is available when the port is operating in either a USB 2.0 or USB 3.2 connection.

Product Example	Notes
USB4 Dock (Upstream) – power provider - partial USB equiv.	This is the dedicated upstream port of a USB4 -based dock that includes a USB4 hub with one or more downstream USB Type-C ports. This product provides one or more user-visible functions and only a portion of this functionality is available when the port is operating in either a USB 2.0 or USB 3.2 connection. This port can also supply power.
USB4 Dock (Upstream) – no USB equiv.	This is the dedicated upstream port of a USB4 -based dock that includes a USB4 hub with one or more downstream USB Type-C ports. This product provides one or more user-visible functions and that none of this functionality is available when the port is operating in either a USB 2.0 or USB 3.2 connection.
USB4 Dock (Upstream) – power provider – no USB equiv.	This is the dedicated upstream port of a USB4 -based dock that includes a USB4 hub with one or more downstream USB Type-C ports. This product provides one or more user-visible functions and that none of this functionality is available when the port is operating in either a USB 2.0 or USB 3.2 connection. This port can also supply power.
USB4 Dock (Downstream)	This is one of the downstream ports of a USB4 dock that includes a USB4 hub.
USB4 Dock (Downstream) – power consumer	This is one of the downstream ports of a USB4 dock that includes a USB4 hub. This port can also supply power.
USB Type-C Charger	This product's only function is to be a USB PD power source.
USB Type-C Power Bank	This product's primary function is to be a USB PD power source using an internal battery as its source of power. The product also consumes power to charge its internal battery or pass power through to another port of the product.
USB PD-based Power Sinking Device	This product has no USB data functionality and only uses power supplied over a USB PD connection for its non-USB purpose.

Table I-2 USB PD Encoding Guidelines for Example Products

<div>Notes:</div> <div>1. DC = USB Device Controller (UFP)</div> <div>2. Upstream and Downstream are in reference to the physical port locations.</div> <div>3. For product types: PSDs are power sinks only; Power Bricks are power sources only.</div> <div>Product Examples</div>	UFP Product Type		DFP Product Type		UFP Device Cap and Highest Speed				DFP Host Capability			
	Type	VDO Required	Type	VDO Required	000b = <i>USB 2.0</i> only				<i>USB 2.0</i> Host			
	Undefined	None	Undefined	None	001b = Gen1				<i>USB 3.2</i> Host			
	PDUSB Hub	UFP VDO	PDUSB Hub	DFP VDO	010b = Gen2				<i>USB 4</i> Host			
	PDUSB Peripheral	UFP VDO	PDUSB Host	DFP VDO	011b = Gen3							
	PSD	None	Power Brick	DFP VDO	100b...111b = Reserved							
	Undefined		PDUSB Host						1	1	0	
	PSD		PDUSB Host						1	1	0	
	PDUSB Peripheral		PDUSB Host		1	0	0	0	000b	1	1	0
	PDUSB Peripheral		PDUSB Host		1	0	1	0	001b or 010b	1	1	0
	Undefined		PDUSB Host							1	1	1
	PSD		PDUSB Host							1	1	1
	PDUSB Peripheral		PDUSB Host		1	0	0	0	000b	1	1	1
	PDUSB Peripheral		PDUSB Host		1	0	1	0	001b or 010b	1	1	1
	PDUSB Peripheral		PDUSB Host		1	0	1	1	010b or 011b	1	1	1
PDUSB Peripheral		PDUSB Host		1	0	1	1	010b or 011b	1	1	1	
PDUSB Peripheral		PDUSB Host		0	1	0	1	010b or 011b	1	1	1	
PDUSB Peripheral		Undefined		1	0	1	0	001b or 010b				
PDUSB Peripheral		Power Brick		1	0	1	0	001b or 010b	0	0	0	
PDUSB Peripheral		Undefined		1	0	1	1	010b or 011b				
PDUSB Peripheral		Power Brick		1	0	1	1	010b or 011b	0	0	0	
PDUSB Peripheral		Undefined		1	0	1	1	010b or 011b				
PDUSB Peripheral		Undefined		0	1	0	1	010b or 011b				

Table I-2 USB PD Encoding Guidelines for Example Products, cont.

	UFP Product Type		DFP Product Type		UFP Device Cap and Highest Speed				DFP Host Capability		
	Type	VDO Required	Type	VDO Required	USB 2.0 Device	USB 2.0 Billboard	USB 3.2 Device	USB4 Device	000b = USB 2.0 only	USB 2.0 Host	USB4 Host
	Undefined	None	Undefined	None					001b = Gen1	0 0	
	PDUSB Hub	UFP VDO	PDUSB Hub	DFP VDO					010b = Gen2	0 0	
Product Examples	PDUSB Peripheral	UFP VDO	PDUSB Host	DFP VDO					011b = Gen3	1 1	
	PSD	None	Power Brick	DFP VDO					100b...111b = Reserved	1 1	0
	PDUSB Hub		Undefined		1	0	1	0	001b or 010b		
	PDUSB Hub		Power Brick		1	0	1	0	001b or 010b	0 0	0
	Undefined		PDUSB Hub							1 1	0
	PSD		PDUSB Hub							1 1	0
	PDUSB Hub		Undefined		1	0	1	1	010b or 011b		
	PDUSB Hub		Power Brick		1	0	1	1	010b or 011b	0 0	0
	Undefined		PDUSB Hub							1 1	1
	PSD		PDUSB Hub							1 1	1
	PDUSB Peripheral		Undefined		1	0	1	1	010b or 011b		
	PDUSB Peripheral		Power Brick		1	0	1	1	010b or 011b	0 0	0
	PDUSB Peripheral		Undefined		1	0	1	1	010b or 011b		
	PDUSB Peripheral		Power Brick		1	0	1	1	010b or 011b	0 0	0
	PDUSB Peripheral		Undefined		0	1	0	1	010b or 011b		
	PDUSB Peripheral		Power Brick		0	1	0	1	010b or 011b	0 0	0
	Undefined		PDUSB Hub							1 1	1
	PSD		PDUSB Hub							1 1	1

Notes:
1. DC = USB Device Controller (UFP)
2. Upstream and Downstream are in reference to the physical port locations.
3. For product types: PSDs are power sinks only; Power Bricks are power sources only.

Table I-2 USB PD Encoding Guidelines for Example Products, cont.

Notes: 1. DC = USB Device Controller (UFP) 2. Upstream and Downstream are in reference to the physical port locations. 3. For product types: PSDs are power sinks only; Power Bricks are power sources only.	UFP Product Type		DFP Product Type		UFP Device Cap and Highest Speed				DFP Host Capability		
	Type	VDO Required	Type	VDO Required	000b = USB 2.0 only				USB4 Host		
	Undefined	None	Undefined	None	001b = Gen1				USB 3.2 Host		
	PDUSB Hub	UFP VDO	PDUSB Hub	DFP VDO	010b = Gen2				USB 2.0 Host		
	PDUSB Peripheral	UFP VDO	PDUSB Host	DFP VDO	011b = Gen3						
Product Examples	PSD	None	Power Brick	DFP VDO	100b...111b = Reserved						
	Undefined		Power Brick								
	PSD		Power Brick								
	PSD		Undefined								
	PSD		Undefined								
USB Type-C Charger	Undefined		Power Brick						0	0	0
USB Type-C Power Bank	PSD		Power Brick						0	0	0
USB PD-based Power Sinking Device	PSD		Undefined						0	0	0

I.3 Detailed Examples

This section includes a number of detailed examples drawn from the list of product examples listed in Table I-1 and using the guidance provided by Table I-2. This set of examples was chosen to provide additional explanatory comments and for pointing out some notable characteristics.

All of the product examples fall into one of the following USB Type-C power and data role categories:

1. **Sink / UFP** – this product can only consume power and be a UFP in a data connection.
2. **Source / DFP** – this product can only supply power and be a DFP in a data connection.
3. **DRP (Source or Sink) / Not DRD (UFP)** – this product can supply or consume power and only be a UFP in a data connection.
4. **DRP (Source or Sink) / Not DRD (DFP)** – this product can supply or consume power and only be a DFP in a data connection.
5. **DRP (Source or Sink) / DRD** – this product can supply or consume power and be either a UFP or DFP in a data connection.

I.3.1 USB 3.2 Host – no DC – Power Consumer : **DRP (Source or Sink) / Not DRD (DFP)**

The following summarizes the connection results for connecting this product to any of the five distinct product power/data role combinations:

	USB 3.2 Host – no DC – power consumer	
	DRP (Source or Sink)	Not DRD (DFP)
Sink / UFP	Source	DFP
Source / DFP	Sink	No connect
DRP / Not DRD (UFP)	Source or Sink	DFP
DRP / Not DRD (DFP)	Source or Sink	No connect
DRP / DRD	Source or Sink	DFP

This **USB 3.2** host is typical of a notebook PC that when connected to a charger or power-sourcing hub, dock or peripheral, can sink power using **USB PD**. As this product has no UFP device controller functionality, it will not establish a data connection to any other product that isn't in a UFP data role.

As this host can sink power, the UFP Product Type is set to PSD and no UFP VDO is needed.

The DFP Product Type is set to PDUSB Host and the DFP VDO indicates that it supports **USB 2.0** and **USB 3.2** Host Capabilities.

I.3.2 USB 3.2 Peripheral – Power Provider : **DRP (Source or Sink) / Not DRD (UFP)**

The following summarizes the connection results for connecting this product to any of the five distinct product power/data role combinations:

	USB 3.2 Peripheral – power provider	
	DRP (Source or Sink)	Not DRD (UFP)
Sink / UFP	Source	No connect
Source / DFP	Sink	UFP
DRP / Not DRD (UFP)	Source or Sink	No connect
DRP / Not DRD (DFP)	Source or Sink	UFP
DRP / DRD	Source or Sink	UFP

This **USB 3.2** peripheral is unusual in that goes beyond a traditional peripheral by being able to supply power to a connected host, hub or dock that is capable of consuming power over the connection. An example might be a **USB 3.2**-based dock where all of its functional capabilities connect in the UFP role after the Source/Sink relationship with its port partner are established via USB Type-C and **USB PD** connection protocol. If this peripheral ends up in the Source power role, the use of a **USB PD** Data Role Swap will be used to correctly establish the UFP data role of the peripheral's port.

The UFP Product Type is set to PDUSB Peripheral and the UFP VDO indicates **USB 2.0** and **USB 3.2** device capabilities and the highest **USB 3.2** speed of either Gen1 (001b) or Gen2 (010b).

As this peripheral can supply power, the DFP Product Type is set to Power Brick and the DFP VDO indicates no support for any of the USB Host Capabilities (**USB 2.0**, **USB 3.2** or **USB4**).

1.3.3 **USB4 Host – Power Consumer : DRP (Source or Sink) / Not DRD (DFP)**

The following summarizes the connection results for connecting this product to any of the five distinct product power/data role combinations:

	USB4 Host – no DC – power consumer	
	DRP (Source or Sink)	Not DRD (DFP)
Sink / UFP	Source	DFP
Source / DFP	Sink	No connect or USB4 *
DRP / Not DRD (UFP)	Source or Sink	DFP
DRP / Not DRD (DFP)	Source or Sink	No connect or USB4 *
DRP / DRD	Source or Sink	DFP

* If the port partner is also a **USB4** DFP.

This **USB4** host is similar to the **USB 3.2** host described earlier (see 1.3.1) but differs with regard to how it works with other hosts, hubs or docks depending on the data capabilities of those other products. Both host examples do not include UFP data role support but the **USB4** host is capable of communicating with another **USB4** host as an inter-domain link even as it does not include any USB device controller functionality.

To enable two **USB4** DFPs to discover and establish the **USB4** data connection, the **USB4** discovery and entry process of the Source will discover the DFP VDO of the Sink which will indicate **USB4** Host Capability and as long as the cable also supports **USB4**, the **USB4** data connection will be enabled. Following that, the **USB4** Connection Manager of each host will discover the inter-domain link and enable the host-to-host data connections supported by the application layer.

As this host can sink power, the UFP Product Type is set to PSD and no UFP VDO is needed.

The DFP Product Type is set to PDUSB Host and the DFP VDO indicates that it supports **USB 2.0**, **USB 3.2** and **USB4** Host Capabilities.

1.3.4 **USB4 Host – USB4 DC – Partial or no USB Equivalent : DRP (Source or Sink) / DRD**

The following summarizes the connection results for connecting this product to any of the five distinct product power/data role combinations:

	USB4 Host – USB4 DC	
	DRP (Source or Sink)	DRD
Sink / UFP	Source	DFP
Source / DFP	Sink	UFP
DRP / Not DRD (UFP)	Source or Sink	DFP
DRP / Not DRD (DFP)	Source or Sink	UFP
DRP / DRD	Source or Sink	DFP or UFP

This **USB4** host differs from the **USB4** host described earlier (see I.3.3) in that it does include some **USB4**-based UFP device functionality and therefore is a DRD. This specific example is included here to illustrate what is needed if only a portion of this device functionally is available when the port is operating in either a **USB 2.0** or **USB 3.2** connection.

When this example host is connected to another **USB4** host, its device functionality is fully available to the other host via functions that are connected to one or more of the available **USB4** tunneling protocols (**USB**, **DisplayPort** or **PCIe**). But when this host is connected to either a **USB 2.0** or **USB 3.2** host, some or all of the **USB4**-based device functionality is not available because it doesn't map to one of the existing USB Device Classes – in this case, the example host is required to expose a Billboard Device Class function specific to the missing functionality on the **USB 2.0** interface when in the UFP data role.

As this host has device controller functionality in a UFP data role, the UFP Product Type is set to PDUSB Peripheral and the UFP VDO indicates **USB4** device capabilities and the highest **USB4** speed of either Gen2 (010b) or Gen3 (011b).

Additionally, since this host supports only partial or no USB equivalent functionality when connected to a non-**USB4** host, the UFP VDO will indicate what USB equivalent functionality is supported (**USB 2.0** or **USB 3.2**) along with providing a **USB 2.0** Billboard or indicate only the presence of a **USB 2.0** Billboard. The following lists the possible UFP VDO Device Capabilities combinations for this example.

USB 2.0 Device	USB 2.0 Billboard	USB 3.2 Device	USB4 Device	
Interpretation of the USB4 host's UFP Device Capabilities				
0	1	0	1	None of the USB4 device functionality maps to either USB 2.0 or USB 3.2 Device Class equivalents.
1	0	0	1	At least some of the USB4 device functionality maps to USB 2.0 Device Class equivalents but none maps to USB 3.2 Device Class equivalents.
1	0	1	1	At least some of the USB4 device functionality maps to USB 2.0 and USB 3.2 equivalents.

The DFP Product Type is set to PDUSB Host and the DFP VDO indicates that it supports **USB 2.0**, **USB 3.2** and **USB4** Host Capabilities.

I.3.5 **USB4 Peripheral : UFP (Sink)**

The following summarizes the connection results for connecting this product to any of the five distinct product power/data role combinations:

	USB4 Peripheral	
	Sink	UFP
Sink / UFP	No connect	No connect
Source / DFP	Sink	UFP
DRP / Not DRD (UFP)	Sink	No connect
DRP / Not DRD (DFP)	Sink	UFP
DRP / DRD	Sink	UFP

This **USB4** peripheral provides one or more user-visible functions and that functionally is also available when the port is operating in either a **USB 2.0** or **USB 3.2** connection. Power for this product is provided either over the USB Type-C connection or by a separate power supply not associated with the USB connection.

The UFP Product Type is set to PDUSB Peripheral and the UFP VDO indicates **USB 2.0**, **USB 3.2** and **USB4** device capabilities and the highest **USB4** speed of either Gen2 (010b) or Gen3 (011b).

As this peripheral can only sink power and has no DFP data functionality, the DFP Product Type is set to Undefined and no DFP VDO is needed.

1.3.6 **USB4 Dock (Upstream) – Power Provider : DRP (Source or Sink) / Not DRD (UFP)**

The following summarizes the connection results for connecting this product to any of the five distinct product power/data role combinations:

	USB4 Dock (Upstream) – power provider	
	DRP (Source or Sink)	Not DRD (UFP)
Sink / UFP	Source	No connect
Source / DFP	Sink	UFP
DRP / Not DRD (UFP)	Source or Sink	No connect
DRP / Not DRD (DFP)	Source or Sink	UFP
DRP / DRD	Source or Sink	UFP

This is the dedicated upstream port of a **USB4**-based dock that includes a **USB4** hub with one or more downstream USB Type-C ports along with one or more user-visible functions and that functionally is also available when the port is operating in either a **USB 2.0** or **USB 3.2** connection. This upstream port can also supply power.

This dock's upstream port UFP Product Type is set to PDUSB Peripheral and the UFP VDO indicates **USB 2.0**, **USB 3.2** and **USB4** device capabilities and the highest **USB4** speed of either Gen2 (010b) or Gen3 (011b).

Since this dock's upstream port can also source power but has no DFP data role functionality, its DFP Product Type is Power Brick and the DFP VDO indicates no support for any of the USB Host Capabilities (**USB 2.0**, **USB 3.2** or **USB4**).

1.3.7 **USB4 Hub (Downstream) – DFP (Source)**

The following summarizes the connection results for connecting this product to any of the five distinct product power/data role combinations:

USB4 Hub (Downstream) – power provider		
	Source	DFP
Sink / UFP	Source	DFP
Source / DFP	No connect	No connect
DRP / Not DRD (UFP)	Source	DFP
DRP / Not DRD (DFP)	Source	No connect or USB4 *
DRP / DRD	Source	DFP

* If the port partner is also a **USB4** DFP.

This is one of the downstream ports of a **USB4** hub.

When a USB host product is connected to a **USB4** hub downstream port, it functions similarly to **USB 2.0** and **USB 3.2** hub downstream ports by not establishing a data connection when that host is only **USB 2.0** or **USB 3.2** capable. If that USB host product is capable of **USB4**, the **USB4** hub will use its **USB4** discovery and entry process to discover the DFP VDO of the attached **USB4** host and as long as the cable also supports **USB4**, the **USB4** data connection will be enabled.

To enable two **USB4** DFPs to discover and establish the **USB4** data connection, the. Following that, the **USB4** Connection Manager of each host will discover the inter-domain link and enable the host-to-host data connections supported by the application layer

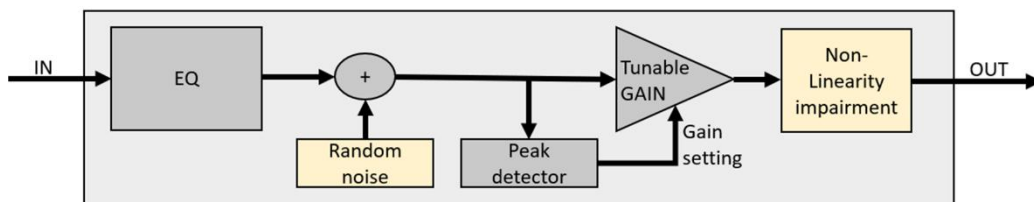
Since this hub's downstream port does not support receiving power nor does it offer any UFP data functionality in a UFP data role, the UFP Product Type is set to Undefined.

This hub's downstream port DFP Product Type is set to PDUSB Hub and the DFP VDO indicates support for all USB Host Capabilities (**USB 2.0**, **USB 3.2** and **USB4**).

J Design Assumptions for the **USB4** Gen4 LRD Cable Specification

This appendix describes the design assumptions that were used during the spec development of the **USB4** LRD cable. The description here is **informative** only and might be used as a reference for design and/or system simulations.

This is the LRD IC model used for the spec development:



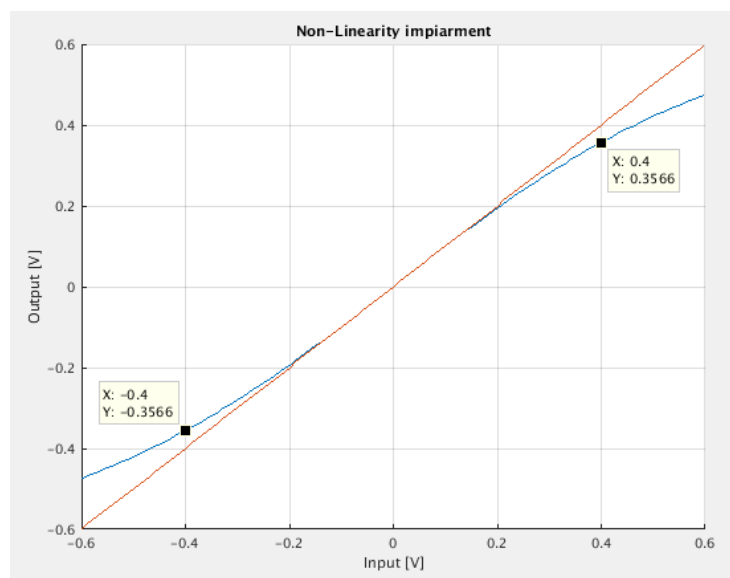
The following parameters were assumed:

1. Reference CTLE is given by:

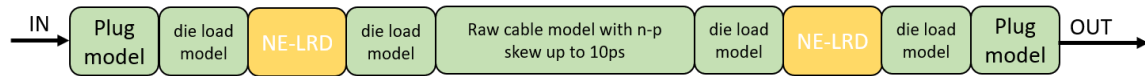
$$H(s) = \omega_{p2} \cdot \frac{S + 10^{\frac{A_{DC}}{20}} \cdot \omega_{p1}}{(S + \omega_{p1}) \cdot (S + \omega_{p2})}$$

- a. Where:

- i. A_{DC} is the DC gain (in dB)
- ii. S is the frequency in Laplace domain
- iii. $\omega_{p1} = 2 \cdot \pi \cdot 4.26e9 \frac{rad}{sec}$
- iv. $\omega_{p2} = 2 \cdot \pi \cdot 20e9 \frac{rad}{sec}$
2. Random noise injection with 1.2 mV
3. Tunable gain controlled by peak detector with threshold set to 700 mV-pp
4. Gain limit of 8 dB
5. Non-linearity modeling with $1.53 \cdot \tanh(1/1.53)$
 - a. This reflects 1 dB compression point of 800 mV-pp (reference to input)



This is a cable system model used for the specification development:



The settings of the NE (near-end, i.e., closer to the TX) and FE (far-end, i.e., closer to the RX) LRDs are as follows:

1. NE EQ setting: $A_{DC} = -7dB$
2. FE EQ setting: $A_{DC} = -5dB$
3. Self-tuning gain enabled in both NE and FE LRDs
4. Raw cable n-p skew is limited to 10 ps

The response of the cable including the LRD's EQ and gain is:

