

Targeted Host Compliance Test Specification

Date: Feb 27, 2019

Revision: 1.00

Purpose

An Embedded Host is defined as a Targeted Host that provides Targeted Host functionality over Standard-A or Micro-AB receptacles. This document broadens certification requirements to allow Targeted Host Certification over USB-C ports. It also clarifies requirements for hosts that are seeking certification on a non-Windows OS. Peripheral certification of a DRD (Dual Role Data) device is not covered in this document.

Test Requirements

Full Host Certification requires all the testing described in the [USB 3.2 Product Matrix](#), which requires Windows OS support on the Host Under Test. A full-featured host that does not support Windows can only be certified as a Targeted Host. The main differences between Full Host Certification and Targeted Host Certification are:

- Full Host Certification requires XHCICV, USB3CV, USB2CV, and Hub3CV tests as applicable.
- Targeted Host certification allows hosts to disable some features.
- Targeted Host certification allows limited peripheral support via a TPL (Targeted Peripheral List).

The following table lists required testing for Targeted Hosts.

Test Requirement	Reference
USB 2.0 Electrical	USB 2.0 Electrical CTS
USB 3.2 Electrical	Electrical CTS for SuperSpeed USB Electrical CTS for Enhanced SS USB
USB 3 Link Tests	USB 3.1 Link Layer Test Specification
Tests for USB-C Ports (USB Type-C Functional, PD, etc)	USB-C Product Matrix
Interoperability Testing with U1/U2 enabled	At a minimum, a host must pass EH interoperability. If a host TPL contains all the devices used for a full xHCI host certification, then testing will include the full device tree used in <i>xHCI Interoperability Test Procedures for Peripherals, Hubs and Hosts (Legacy,</i>

	<p><i>USB Type-C and USB Power Delivery</i>). This also includes a subset of 150 popular USB 2.0 devices.</p> <p>Compliance Plan for USB On-The-Go and Embedded Host 3.0, xHCI Interoperability Test Procedures for Peripherals, Hubs and Hosts (Legacy, USB Type-C and USB Power Delivery)</p>
Embedded Host Testing	Compliance Plan for USB On-The-Go and Embedded Host 3.0
BC 1.2	USB Battery Charging 1.2 Compliance Plan
xHC Debug Capability	Required for a full host implementation.

Full Host Implementations

If a host is seeking Targeted Host Certification only because it does not support Windows, it must pass all testing required in the USB 3.2 Product Matrix, except for CV testing (XHCICV, USB3CV, USB2CV, Hub3CV). These full-featured hosts must work with the full interoperability tree as defined in *xHCI Interoperability Test Procedures for Peripherals, Hubs and Hosts (Legacy, USB Type-C and USB Power Delivery)*. This includes Interoperability with the subset of 150 popular USB 2.0 devices.

Helps for Linux and Android Implementations

The Linux kernel, including the Ubuntu flavor, includes 2 drivers (modules) that are used for USB-IF testing: *usb_hset_test* and *lvs*.

USB 2.0 Electrical Testing

Embedded Host testing for USB 2.0 included the definition of a fixture to facilitate electrical testing without user involvement on the Host Under Test. See [PIDVID](#) and [HS Embedded Host Electrical Test Board](#) for purchasing information. The software implementation for this is included in Linux kernel v2.6.35 and above. It is also in the Ubuntu distribution, so can be included in Android builds. For directions on applying *usb_ehset_test* as a patch, go [here](#). *usb_ehset_test* source code can be found in Linux source at `drivers/usb/misc/ehset.c`.

This functionality is also available at <https://code.google.com/archive/p/openusbhset/>.

Link, USB-C Functional and 3.2 Electrical Testing

- LVS helpers

Some tests in Link and Type-C Functional require “helpers” that initiate particular test conditions (GetDescriptor, etc). The *lvs* module (driver) in Linux can be found in the latest Linux source at `drivers/usb/misc/lvstest.c`. This module creates attributes that allow initiation of host behavior for Link Tests and USB Type-C Functional Tests (`u1_timeout`, `u2_timeout`, `hot_reset`, `u3_entry`, `u3_exit`, `warm_reset`). The host implementation must include this module, as well as the ability to bind it and access the associated attributes. For an Android device this would typically be done by providing terminal access on a host that has been rooted. The *lvs* driver must be bound to the root hub of the host being tested, or to a hub if there is an embedded hub. Instructions for binding the

driver can be found in the initial commit log of `lvstest.c`. Descriptions for accessing the attributes can be found at [Documentation/ABI/testing/sysfs-bus-usb-lvstest](#).

- 3.2 Electrical Testing

xHCI hosts have the option of requiring software intervention to enable compliance testing. See CTE documentation in 4.19.1.2.4.1 in *eXtensible Host Controller Interface for Universal Serial Bus*. If the host requires software intervention, this can be done using the `enable_compliance` attribute of the `lvs` module.