

# USB Type-C ENGINEERING CHANGE NOTICE

## Title: Try.SRC and Try.SNK Usage Clarifications

Applied to: USB Type-C® Specification Release 2.1, May 2021

<b>Brief description of the functional changes proposed:</b>
Adds clarification relating to the implementation and use of Try.SRC and Try.SNK.

<b>Benefits as a result of the proposed changes:</b>
The implementation and use of Try mechanisms are not consistently implemented with regard to dual-role products and thus there are use cases where a product not implementing the Try mechanism will end up locked into a power/data role state that is not useful when connected to a non-PD-based DRP. This clarification intends to provide better guidance regarding the implementation and use of Try mechanisms.

<b>An assessment of the impact to the existing revision and systems that currently conform to the USB specification:</b>
Unless existing products are updated, they might not always end up in their preferred power and data roles.

<b>An analysis of the hardware implications:</b>
May require changes to HW if the appropriate Try mechanism isn't implemented.

<b>An analysis of the software implications:</b>
No impact to SW.

<b>An analysis of the compliance testing implications:</b>
No changes to compliance expected.

# USB Type-C ENGINEERING CHANGE NOTICE

## Actual Change Requested

### Changes using modified or new text and tables

#### (a) Section 4.5.1.4.1

##### From Text:

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 effect 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.

[Try.SRC](#) and [Try.SNK](#) are intended to ensure more predictable power roles when initially connecting two DRPs, especially if the port partner does not support [USB PD](#). For example, a small mobile device may want to implement [Try.SNK](#), so that when attaching to a DRP laptop, the mobile device will always initially be the power sink. Similarly, a laptop or Power Bank may wish to implement [Try.SRC](#) to ensure it always sources power to attached DRPs. Self-powered devices such as AMAs or those whose primary function is a data UFP may also consider implementing [Try.SNK](#) to ensure they can properly expose their functionality. If both sides support [USB PD](#), the appropriate roles may then be further refined or swapped as per the [USB PD](#) specification.

##### To Text:

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 effect 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. ~~[Try.SRC](#) and [Try.SNK](#) are intended to ensure more predictable power roles when initially connecting two DRPs, especially if the port partner does not support [USB PD](#).~~ For example, a small mobile device ~~that prefers being a Sink or UFP may want to~~ should implement [Try.SNK](#), so that when ~~attaching~~ attached to a DRP ~~system such as a~~ laptop, the mobile device will always ~~initially~~ be the power sink. If the mobile device is connected to 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~~ laptop or Power Bank may wish to ~~that doesn't consume power over the port should~~ implement [Try.SRC](#) to ensure it always ~~sources power~~ ends up being a DFP when ~~to~~ attached ~~to~~ DRPs. ~~Self-powered devices such as AMAs or those whose primary function is a data UFP may also consider implementing [Try.SNK](#) to ensure they can properly expose their functionality. If both sides support [USB PD](#), the appropriate roles may then be further refined or swapped as per the [USB PD](#) specification.~~

# USB Type-C ENGINEERING CHANGE NOTICE

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 useable 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 and the Try mechanism might be the only opportunity for the product to get into a useable functional role.

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

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