USB Power Delivery ENGINEERING CHANGE NOTICE

Title: Sink No Response Timer

Applied to: USB Power Delivery Specification Revision 2.0 Version 1.2

Brief description of the functional changes:
The NoResponseTimer was partially removed from the Sink state operation but still appears in several parts of the specification, including as normative text. This ECR removes the remaining text.

Benefits as a result of the changes:
Operation of Sink without NoResponseTimer is clearly defined in the specification.

An assessment of the impact to the existing revision and systems that currently conform to the USB specification:
Aligns specification with the requirements expected on publication of the specification.

An analysis of the hardware implications:
None.

An analysis of the software implications:
None.

An analysis of the compliance testing implications:

Ensures that there are no unexpected timers or transitions to be tested in the state diagrams.

USB Power Delivery ENGINEERING CHANGE NOTICE

Actual Change

(a). Section 6.5.7, Page 195

From Text:

6.5.7 NoResponseTimer

The ***NoResponseTimer*** is used by the Policy Engine in a Source or Sink to determine that its Port Partner is not responding after a Hard Reset. When the ***NoResponseTimer*** times out, the Policy Engine shall issue up to ***nHardResetCount*** additional Hard Resets before determining that the Port Partner is non-responsive to USB Power Delivery messaging.

If the Sink fails to receive a ***Source_Capabilities*** Message received within ***tNoResponse*** of:

- The last bit of a ***Hard Reset*** Signaling being sent by the PHY Layer if the ***Hard Reset*** Signaling was initiated by the Sink
- The last bit of a ***Hard Reset*** Signaling being received by the PHY Layer if the ***Hard Reset*** Signaling was initiated by the Source

Then the Sink shall issue additional Hard Resets up to ***nHardResetCount*** times (see Section 6.7.2).

If the Source fails to receive a ***GoodCRC*** Message in response to a ***Source_Capabilities*** Message within ***tNoResponse*** of:

- The last bit of a ***Hard Reset*** Signaling being sent by the PHY Layer if the ***Hard Reset*** Signaling was initiated by the Sink
- The last bit of a ***Hard Reset*** Signaling being received by the PHY Layer if the ***Hard Reset*** Signaling was initiated by the Source

Then the Source shall issue additional Hard Resets up to ***nHardResetCount*** times (see Section 6.7.2).

For a non-responsive device, the Policy Engine in a Source may either decide to continue sending ***Source_Capabilities*** Messages or to go to non-USB Power Delivery operation and cease sending ***Source_Capabilities*** Messages.

To Text:

6.5.7 NoResponseTimer

The ***NoResponseTimer*** is used by the Policy Engine in a Source to determine that its Port Partner is not responding after a Hard Reset. When the ***NoResponseTimer*** times out, the Policy Engine shall issue up to ***nHardResetCount*** additional Hard Resets before determining that the Port Partner is non-responsive to USB Power Delivery messaging.

~~If the Sink fails to receive a ***Source_Capabilities*** Message received within ***tNoResponse*** of:~~

- ~~The last bit of a ***Hard Reset*** Signaling being sent by the PHY Layer if the ***Hard Reset*** Signaling was initiated by the Sink~~
- ~~The last bit of a ***Hard Reset*** Signaling being received by the PHY Layer if the ***Hard Reset*** Signaling was initiated by the Source~~

~~Then the Sink shall issue additional Hard Resets up to ***nHardResetCount*** times (see Section 6.7.2).~~

If the Source fails to receive a ***GoodCRC*** Message in response to a ***Source_Capabilities*** Message within ***tNoResponse*** of:

- The last bit of a ***Hard Reset*** Signaling being sent by the PHY Layer if the ***Hard Reset*** Signaling was initiated by the Sink
- The last bit of a ***Hard Reset*** Signaling being received by the PHY Layer if the ***Hard Reset*** Signaling was initiated by the Source

Then the Source shall issue additional Hard Resets up to ***nHardResetCount*** times (see Section 6.7.2).

For a non-responsive device, the Policy Engine in a Source may either decide to continue sending ***Source_Capabilities*** Messages or to go to non-USB Power Delivery operation and cease sending ***Source_Capabilities*** Messages.

(b). Section 8.3.2.6, Page 308

From Text:

8.3.2.6 Hard Reset

The following sections describe the steps required for a USB Power Delivery Hard Reset. The Hard Reset returns the operation of the USB Power Delivery to default role and operating voltage/current. During the Hard Reset USB Power Delivery PHY Layer communications shall be disabled preventing communication between the Port partners.

Note: Hard Reset, in this case, is applied to the USB Power Delivery capability of an individual Port on which the Hard Reset is requested. A side effect of the Hard Reset is that it might reset other functions on the Port such as USB.

8.3.2.6.1 Source Initiated Hard Reset

This is an example of a Hard Reset operation when initiated by a Source. Figure 8-8 shows the Messages as they flow across the bus and within the devices to accomplish the Hard Reset.

Figure 8-8 Source initiated Hard Reset

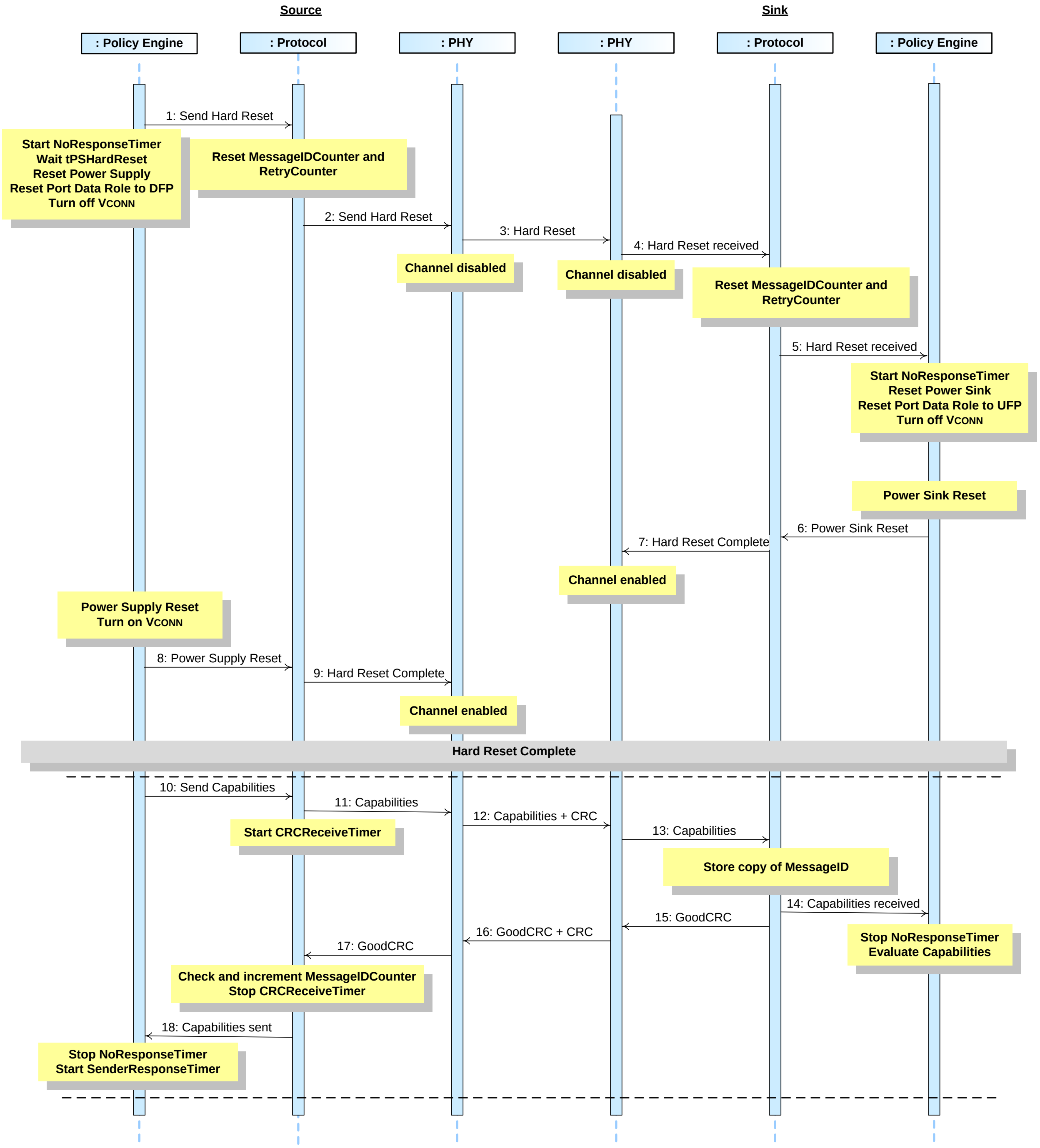


Table 8-8 Steps for Source initiated Hard Reset

Step	Source	Sink
1	The Policy Engine directs the Protocol Layer to generate ***Hard Reset*** Signaling. The Policy Engine starts the ***NoResponseTimer*** and requests the Device Policy Manager to reset the power supply to USB Default Operation. If this is a USB Type-C connector the Policy Engine requests the Device Policy Manager to reset the Port Data Role to DFP and to turn off VCONN if this is on.	
2	Protocol Layer resets ***MessageIDCounter*** and ***RetryCounter.*** Protocol Layer requests the Physical Layer send ***Hard Reset*** Signaling.	
3	Physical Layer sends ***Hard Reset*** Signaling and then disables the PHY Layer communications channel for transmission and reception.	Physical Layer receives the ***Hard Reset*** Signaling and disables the PHY Layer communications channel for transmission and reception.
4		Physical Layer informs the Protocol Layer of the Hard Reset. Protocol Layer resets ***MessageIDCounter*** and ***RetryCounter***.
5		The Protocol Layer informs the Policy Engine of the Hard Reset. The Policy Engine starts the ***NoResponseTimer*** and requests the Device Policy Manager to reset the Power Sink to default operation. If this is a USB Type-C connector the Policy Engine requests the Device Policy Manager to reset the Port Data Role to UFP and to turn off VCONN if this is on.
6		The Power Sink returns to default operation. The Policy Engine informs the Protocol Layer that the Power Sink has been reset.
7		The Protocol Layer informs the PHY Layer that the Hard Reset is complete. The PHY Layer enables the PHY Layer communications channel for transmission and reception.
8	The power supply is reset to default operation. If this is a USB Type-C connector VCONN is turned on. The Policy Engine informs the Protocol Layer that the power supply has been reset.	
9	The Protocol Layer informs the PHY Layer that the Hard Reset is complete. The PHY Layer enables the PHY Layer communications channel for transmission and reception.	
	The reset is complete and protocol communication can restart.	
10	Policy Engine directs the Protocol Layer to send a ***Source_Capabilities*** Message that represents the power supply's present capabilities.	
11	Protocol Layer creates the Message and passes to Physical Layer. Starts ***CRCReceiveTimer***.	
12	Physical Layer appends CRC and sends the ***Source_Capabilities*** Message.	Physical Layer receives the ***Source_Capabilities*** Message and checks the CRC to verify the Message.

Step	Source	Sink
13		Physical Layer removes the CRC and forwards the ***Source_Capabilities*** Message to the Protocol Layer.
14		Protocol Layer stores the ***MessageID*** of the incoming Message. The Protocol Layer forwards the received ***Source_Capabilities*** Message information to the Policy Engine that consumes it. The Policy Engine stops the ***NoResponseTimer***.
15		Protocol Layer generates a ***GoodCRC*** Message and passes it Physical Layer.
16	Physical Layer receives the ***GoodCRC*** Message and checks the CRC to verify the Message.	Physical Layer appends CRC and sends the ***GoodCRC*** Message.
17	Physical Layer removes the CRC and forwards the ***GoodCRC*** Message to the Protocol Layer.	
18	Protocol Layer verifies and increments the ***MessageIDCounter*** and stops ***CRCReceiveTimer***. Protocol Layer informs the Policy Engine that the ***Source_Capabilities*** Message was successfully sent. Policy Engine stops the ***NoResponseTimer*** and starts the ***SenderResponseTimer***.	
	USB Power Delivery communication is re-established.	

8.3.2.6.2 Sink Initiated Hard Reset

This is an example of a Hard Reset operation when initiated by a Sink. Figure 8-9 shows the Messages as they flow across the bus and within the devices to accomplish the Hard Reset.

Figure 8-9 Sink Initiated Hard Reset

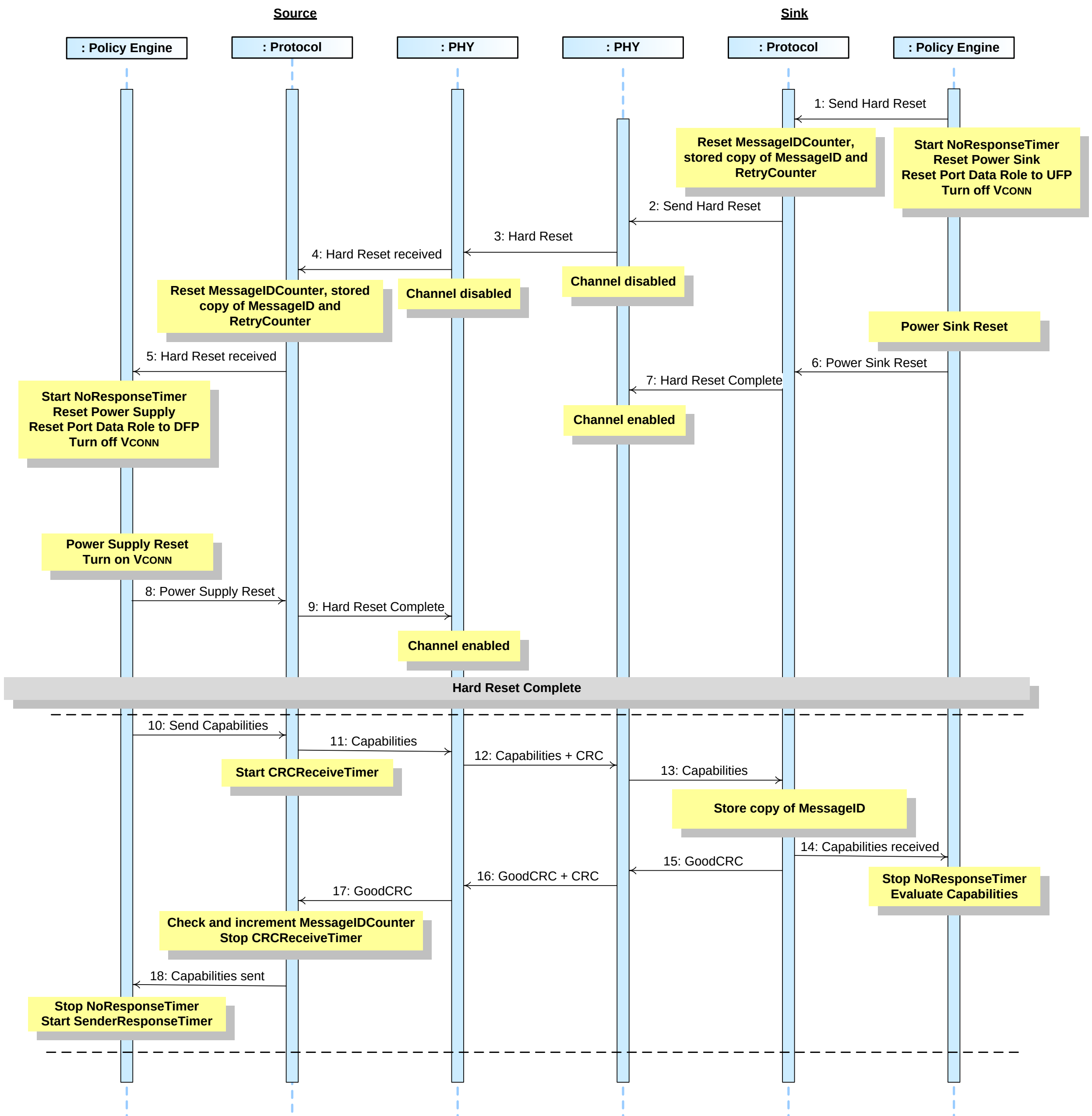


Table 8-9 Steps for Sink initiated Hard Reset

Step	Source	Sink
1		The Policy Engine directs the Protocol Layer to generate ***Hard Reset*** Signaling. The Policy Engine starts the ***NoResponseTimer*** and requests the Device Policy Manager to reset the power supply to USB Default Operation. If this is a USB Type-C connector the Policy Engine requests the Device Policy Manager to reset the Port Data Role to UFP and to turn off VCONN if this is on.
2		Protocol Layer resets ***MessageIDCounter***, stored copy of ***MessageID*** and ***RetryCounter***. Protocol Layer requests the Physical Layer send ***Hard Reset*** Signaling.
3	Physical Layer receives the ***Hard Reset*** Signaling and disables the PHY Layer communications channel for transmission and reception.	Physical Layer sends the ***Hard Reset*** Signaling and then disables the PHY Layer communications channel for transmission and reception.
4	Physical Layer informs the Protocol Layer of the Hard Reset. Protocol Layer resets ***MessageIDCounter***, stored copy of ***MessageID*** and ***RetryCounter***.	
5	The Protocol Layer Informs the Policy Engine of the Hard Reset. The Policy Engine starts the ***NoResponseTimer*** and requests the Device Policy Manager to reset the Power Sink to default operation. If this is a USB Type-C connector the Policy Engine requests the Device Policy Manager to reset the Port Data Role to DFP and to turn off VCONN if this is on.	
6		The Power Sink returns to USB Default Operation. The Policy Engine informs the Protocol Layer that the Power Sink has been reset.
7		The Protocol Layer informs the PHY Layer that the Hard Reset is complete. The PHY Layer enables the PHY Layer communications channel for transmission and reception.
8	The power supply is reset to USB Default Operation. If this is a USB Type-C connector VCONN is turned on. The Policy Engine informs the Protocol Layer that the power supply has been reset.	
9	The Protocol Layer informs the PHY Layer that the Hard Reset is complete. The PHY Layer enables the PHY Layer communications channel for transmission and reception.	
	The reset is complete and protocol communication can restart.	
10	Policy Engine directs the Protocol Layer to send a ***Source_Capabilities*** Message that represents the power supply's present capabilities.	
11	Protocol Layer creates the Message and passes to Physical Layer. Starts ***CRCReceiveTimer***.	
12	Physical Layer appends CRC and sends the ***Source_Capabilities*** Message.	Physical Layer receives the ***Source_Capabilities*** Message and checks the CRC to verify the Message.

Step	Source	Sink
13		Physical Layer removes the CRC and forwards the ***Source_Capabilities*** Message to the Protocol Layer.
14		Protocol Layer stores the ***MessageID*** of the incoming Message. The Protocol Layer forwards the received ***Source_Capabilities*** Message information to the Policy Engine that consumes it. The Policy Engine stops the ***NoResponseTimer***.
15		Protocol Layer generates a ***GoodCRC*** Message and passes it Physical Layer.
16	Physical Layer receives the ***GoodCRC*** Message and checks the CRC to verify the Message.	Physical Layer appends CRC and sends the ***GoodCRC*** Message.
17	Physical Layer removes the CRC and forwards the ***GoodCRC*** Message to the Protocol Layer.	
18	Protocol Layer verifies and increments the ***MessageIDCounter*** and stops ***CRCReceiveTimer***. Protocol Layer informs the Policy Engine that the ***Source_Capabilities*** Message was successfully sent. Policy Engine stops the ***NoResponseTimer*** and starts the ***SenderResponseTimer***.	
	USB Power Delivery communication is re-established.	

8.3.2.6.3 Source Initiated Hard Reset – Sink Long Reset

This is an example of a Hard Reset operation when initiated by a Source. In this example the Sink is slow responding to the reset causing the Source to send multiple ***Source_Capabilities*** Messages before it receives a

GoodCRC Message response. Figure 8-10 shows the Messages as they flow across the bus and within the devices to accomplish the Hard Reset.

Figure 8-10 Source initiated reset - Sink long reset

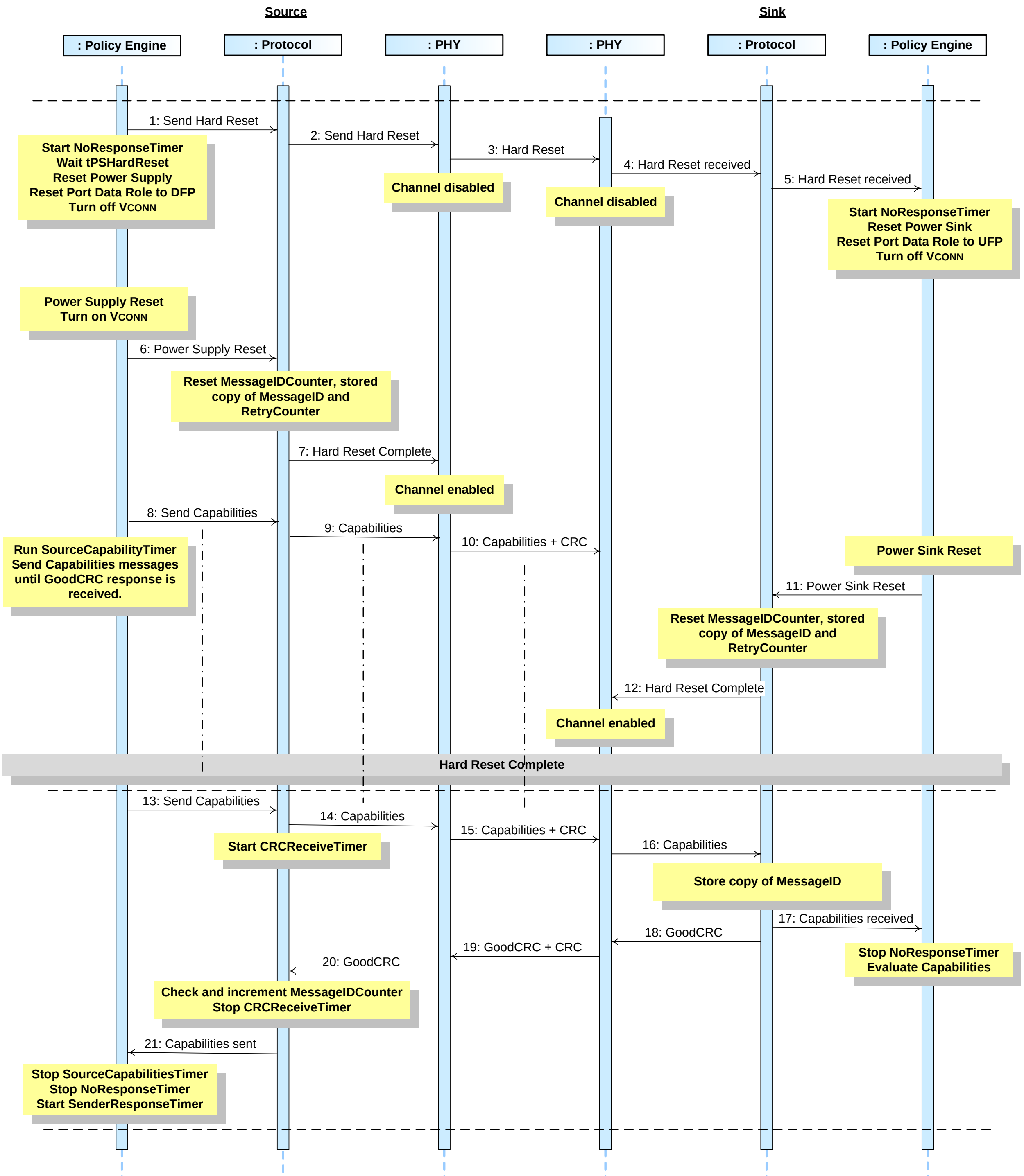


Table 8-10 Steps for Source initiated Hard Reset – Sink long reset

Step	Source	Sink
1	The Policy Engine directs the Protocol Layer to generate ***Hard Reset*** Signaling. The Policy Engine starts the ***NoResponseTimer*** and requests the Device Policy Manager to reset the power supply to USB Default Operation. If this is a USB Type-C connector the Policy Engine requests the Device Policy Manager to reset the Port Data Role to DFP and to turn off VCONN if this is on.	
2	Protocol Layer resets ***MessageIDCounter***, stored copy of ***MessageID*** and ***RetryCounter***. Protocol Layer requests the Physical Layer send ***Hard Reset*** Signaling.	
3	Physical Layer sends the ***Hard Reset*** Signaling and then disables the PHY Layer communications channel for transmission and reception.	Physical Layer receives the ***Hard Reset*** Signaling and disables the PHY Layer communications channel for transmission and reception.
4		Physical Layer informs the Protocol Layer of the Hard Reset. Protocol Layer resets ***MessageIDCounter***, stored copy of ***MessageID*** and ***RetryCounter***.
5		The Protocol Layer Informs the Policy Engine of the Hard Reset. The Policy Engine starts the ***NoResponseTimer*** and requests the Device Policy Manager to reset the Power Sink to default operation. If this is a USB Type-C connector the Policy Engine requests the Device Policy Manager to reset the Port Data Role to UFP and to turn off VCONN if this is on.
6	The power supply is reset to USB Default Operation. If this is a USB Type-C connector VCONN is turned on. The Policy Engine informs the Protocol Layer that the power supply has been reset.	
7	The Protocol Layer informs the PHY Layer that the Hard Reset is complete. The PHY Layer enables the PHY Layer communications channel for transmission and reception.	
	The reset is complete and protocol communication can restart.	
8	Policy Engine directs the Protocol Layer to send a ***Source_Capabilities*** Message that represents the power supply's present capabilities. Policy Engine starts the ***SourceCapabilityTimer***. The ***SourceCapabilityTimer*** times out one or more times until a ***GoodCRC*** Message response is received.	
9	Protocol Layer creates the Message and passes to Physical Layer. Starts ***CRCReceiveTimer***.	
10	Physical Layer appends CRC and sends the ***Source_Capabilities*** Message.	Note: ***Source_Capabilities*** Message not received since channel is disabled.
11		The Power Sink returns to USB Default Operation. The Policy Engine informs the Protocol Layer that the Power Sink has been reset.

Step	Source	Sink
12		The Protocol Layer informs the PHY Layer that the Hard Reset is complete. The PHY Layer enables the PHY Layer communications channel for transmission and reception.
	The reset is complete and protocol communication can restart.	
13	Policy Engine directs the Protocol Layer to send a ***Source_Capabilities*** Message that represents the power supply's present capabilities. Starts the ***SourceCapabilityTimer***.	
14	Protocol Layer creates the Message and passes to Physical Layer. Starts ***CRCReceiveTimer***.	
15	Physical Layer appends CRC and sends the ***Source_Capabilities*** Message.	Physical Layer receives the ***Source_Capabilities*** Message and checks the CRC to verify the Message.
16		Physical Layer removes the CRC and forwards the ***Source_Capabilities*** Message to the Protocol Layer.
17		Protocol Layer stores the ***MessageID*** of the incoming Message. The Protocol Layer forwards the received ***Source_Capabilities*** Message information to the Policy Engine that consumes it. The Policy Engine stops the ***NoResponseTimer*****.**
18		Protocol Layer generates a ***GoodCRC*** Message and passes it Physical Layer.
19	Physical Layer receives the ***GoodCRC*** Message and checks the CRC to verify the Message.	Physical Layer appends CRC and sends the ***GoodCRC*** Message.
20	Physical Layer removes the CRC and forwards the ***GoodCRC*** Message to the Protocol Layer.	
21	Protocol Layer verifies and increments the ***MessageIDCounter*** and stops ***CRCReceiveTimer***. Protocol Layer informs the Policy Engine that the ***Source_Capabilities*** Message was successfully sent. Policy Engine stops the ***SourceCapabilityTimer***, stops the ***NoResponseTimer*** and starts the ***SenderResponseTimer***.	
	USB Power Delivery communication is re-established.	

To Text:

8.3.2.6 Hard Reset

The following sections describe the steps required for a USB Power Delivery Hard Reset. The Hard Reset returns the operation of the USB Power Delivery to default role and operating voltage/current. During the Hard Reset USB Power Delivery PHY Layer communications shall be disabled preventing communication between the Port partners.

Note: Hard Reset, in this case, is applied to the USB Power Delivery capability of an individual Port on which the Hard Reset is requested. A side effect of the Hard Reset is that it might reset other functions on the Port such as USB.

8.3.2.6.1 Source Initiated Hard Reset

This is an example of a Hard Reset operation when initiated by a Source. Figure 8-8 shows the Messages as they flow across the bus and within the devices to accomplish the Hard Reset.

Figure 8-8 Source initiated Hard Reset

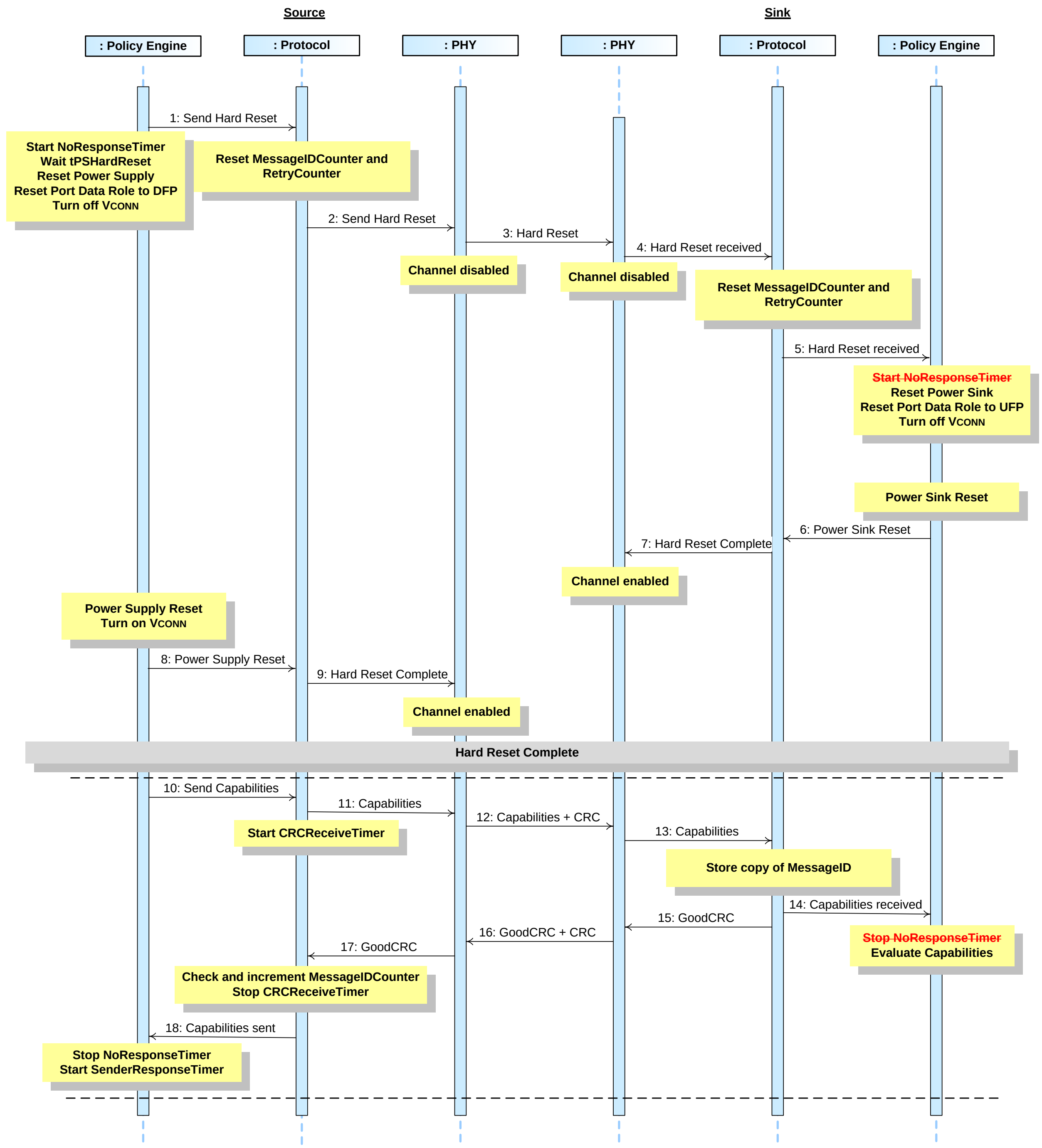


Table 8-8 Steps for Source initiated Hard Reset

Step	Source	Sink
1	The Policy Engine directs the Protocol Layer to generate ***Hard Reset*** Signaling. The Policy Engine starts the ***NoResponseTimer*** and requests the Device Policy Manager to reset the power supply to USB Default Operation. If this is a USB Type-C connector the Policy Engine requests the Device Policy Manager to reset the Port Data Role to DFP and to turn off VCONN if this is on.	
2	Protocol Layer resets ***MessageIDCounter*** and ***RetryCounter.*** Protocol Layer requests the Physical Layer send ***Hard Reset*** Signaling.	
3	Physical Layer sends ***Hard Reset*** Signaling and then disables the PHY Layer communications channel for transmission and reception.	Physical Layer receives the ***Hard Reset*** Signaling and disables the PHY Layer communications channel for transmission and reception.
4		Physical Layer informs the Protocol Layer of the Hard Reset. Protocol Layer resets ***MessageIDCounter*** and ***RetryCounter***.
5		The Protocol Layer informs the Policy Engine of the Hard Reset. The Policy Engine ~~starts the ***NoResponseTimer*** and~~ requests the Device Policy Manager to reset the Power Sink to default operation. If this is a USB Type-C connector the Policy Engine requests the Device Policy Manager to reset the Port Data Role to UFP and to turn off VCONN if this is on.
6		The Power Sink returns to default operation. The Policy Engine informs the Protocol Layer that the Power Sink has been reset.
7		The Protocol Layer informs the PHY Layer that the Hard Reset is complete. The PHY Layer enables the PHY Layer communications channel for transmission and reception.
8	The power supply is reset to default operation. If this is a USB Type-C connector VCONN is turned on. The Policy Engine informs the Protocol Layer that the power supply has been reset.	
9	The Protocol Layer informs the PHY Layer that the Hard Reset is complete. The PHY Layer enables the PHY Layer communications channel for transmission and reception.	
	The reset is complete and protocol communication can restart.	
10	Policy Engine directs the Protocol Layer to send a ***Source_Capabilities*** Message that represents the power supply's present capabilities.	
11	Protocol Layer creates the Message and passes to Physical Layer. Starts ***CRCReceiveTimer***.	
12	Physical Layer appends CRC and sends the ***Source_Capabilities*** Message.	Physical Layer receives the ***Source_Capabilities*** Message and checks the CRC to verify the Message.

USB Power Delivery ENGINEERING CHANGE NOTICE

Step	Source	Sink
13		Physical Layer removes the CRC and forwards the ***Source_Capabilities*** Message to the Protocol Layer.
14		Protocol Layer stores the ***MessageID*** of the incoming Message. The Protocol Layer forwards the received ***Source_Capabilities*** Message information to the Policy Engine that consumes it. ~~The Policy Engine stops the *NoResponseTimer*.~~
15		Protocol Layer generates a ***GoodCRC*** Message and passes it Physical Layer.
16	Physical Layer receives the ***GoodCRC*** Message and checks the CRC to verify the Message.	Physical Layer appends CRC and sends the ***GoodCRC*** Message.
17	Physical Layer removes the CRC and forwards the ***GoodCRC*** Message to the Protocol Layer.	
18	Protocol Layer verifies and increments the ***MessageIDCounter*** and stops ***CRCReceiveTimer***. Protocol Layer informs the Policy Engine that the ***Source_Capabilities*** Message was successfully sent. Policy Engine stops the ***NoResponseTimer*** and starts the ***SenderResponseTimer***.	
	USB Power Delivery communication is re-established.	

8.3.2.6.2 Sink Initiated Hard Reset

This is an example of a Hard Reset operation when initiated by a Sink. Figure 8-9 shows the Messages as they flow across the bus and within the devices to accomplish the Hard Reset.

Figure 8-9 Sink Initiated Hard Reset

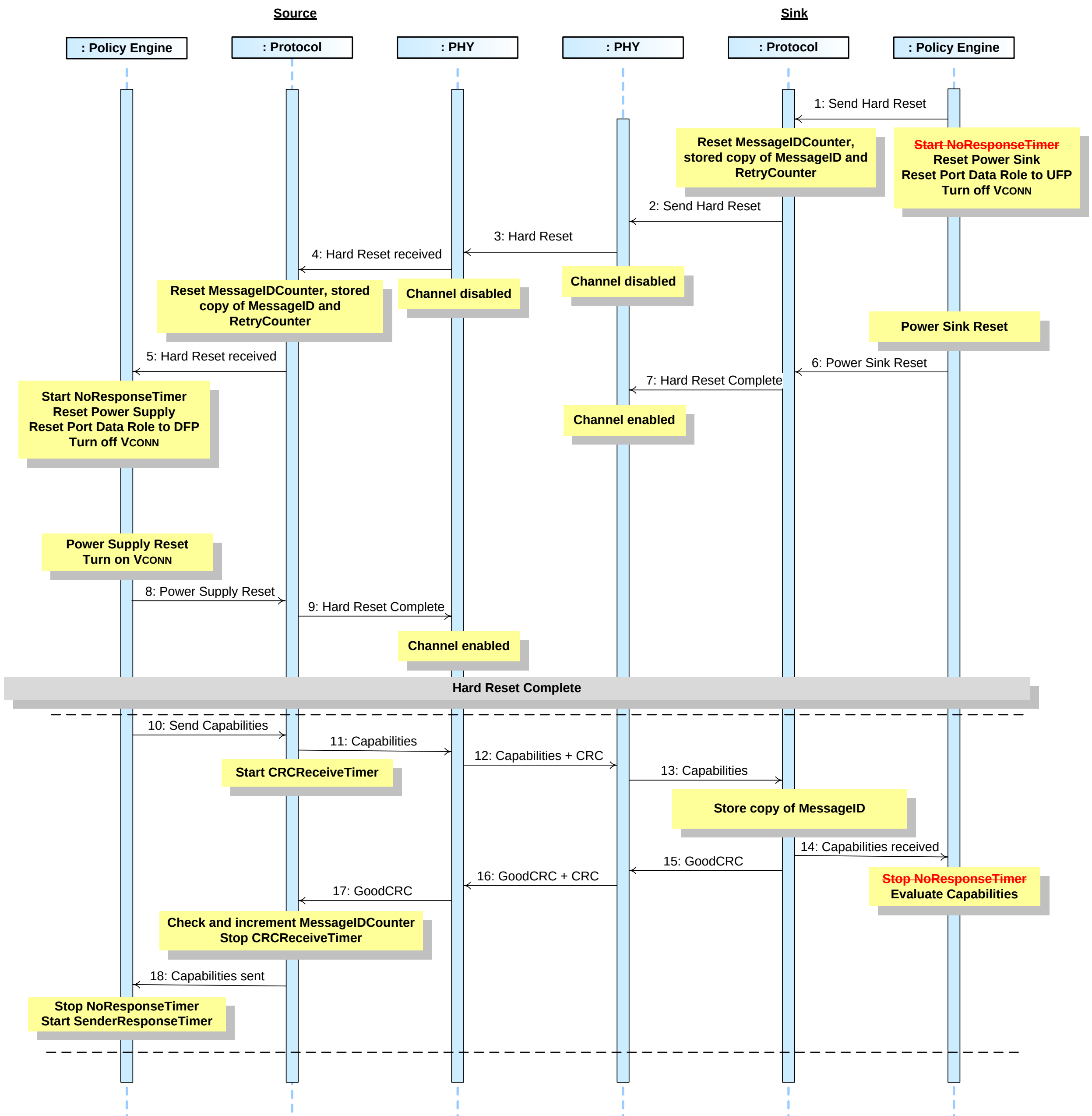


Table 8-9 Steps for Sink initiated Hard Reset

Step	Source	Sink
1		The Policy Engine directs the Protocol Layer to generate ***Hard Reset*** Signaling. The Policy Engine ~~starts the NoResponseTimer and~~ requests the Device Policy Manager to reset the power supply to USB Default Operation. If this is a USB Type-C connector the Policy Engine requests the Device Policy Manager to reset the Port Data Role to UFP and to turn off VCONN if this is on.
2		Protocol Layer resets ***MessageIDCounter***, stored copy of ***MessageID*** and ***RetryCounter***. Protocol Layer requests the Physical Layer send ***Hard Reset*** Signaling.
3	Physical Layer receives the ***Hard Reset*** Signaling and disables the PHY Layer communications channel for transmission and reception.	Physical Layer sends the ***Hard Reset*** Signaling and then disables the PHY Layer communications channel for transmission and reception.
4	Physical Layer informs the Protocol Layer of the Hard Reset. Protocol Layer resets ***MessageIDCounter***, stored copy of ***MessageID*** and ***RetryCounter***.	
5	The Protocol Layer Informs the Policy Engine of the Hard Reset. The Policy Engine starts the ***NoResponseTimer*** and requests the Device Policy Manager to reset the Power Sink to default operation. If this is a USB Type-C connector the Policy Engine requests the Device Policy Manager to reset the Port Data Role to DFP and to turn off VCONN if this is on.	
6		The Power Sink returns to USB Default Operation. The Policy Engine informs the Protocol Layer that the Power Sink has been reset.
7		The Protocol Layer informs the PHY Layer that the Hard Reset is complete. The PHY Layer enables the PHY Layer communications channel for transmission and reception.
8	The power supply is reset to USB Default Operation. If this is a USB Type-C connector VCONN is turned on. The Policy Engine informs the Protocol Layer that the power supply has been reset.	
9	The Protocol Layer informs the PHY Layer that the Hard Reset is complete. The PHY Layer enables the PHY Layer communications channel for transmission and reception.	
	The reset is complete and protocol communication can restart.	
10	Policy Engine directs the Protocol Layer to send a ***Source_Capabilities*** Message that represents the power supply's present capabilities.	
11	Protocol Layer creates the Message and passes to Physical Layer. Starts ***CRCReceiveTimer***.	
12	Physical Layer appends CRC and sends the ***Source_Capabilities*** Message.	Physical Layer receives the ***Source_Capabilities*** Message and checks the CRC to verify the Message.

Step	Source	Sink
13		Physical Layer removes the CRC and forwards the ***Source_Capabilities*** Message to the Protocol Layer.
14		Protocol Layer stores the ***MessageID*** of the incoming Message. The Protocol Layer forwards the received ***Source_Capabilities*** Message information to the Policy Engine that consumes it. ~~The Policy Engine stops the *NoResponseTimer*.~~
15		Protocol Layer generates a ***GoodCRC*** Message and passes it Physical Layer.
16	Physical Layer receives the ***GoodCRC*** Message and checks the CRC to verify the Message.	Physical Layer appends CRC and sends the ***GoodCRC*** Message.
17	Physical Layer removes the CRC and forwards the ***GoodCRC*** Message to the Protocol Layer.	
18	Protocol Layer verifies and increments the ***MessageIDCounter*** and stops ***CRCReceiveTimer***. Protocol Layer informs the Policy Engine that the ***Source_Capabilities*** Message was successfully sent. Policy Engine stops the ***NoResponseTimer*** and starts the ***SenderResponseTimer***.	
	USB Power Delivery communication is re-established.	

8.3.2.6.3 Source Initiated Hard Reset – Sink Long Reset

This is an example of a Hard Reset operation when initiated by a Source. In this example the Sink is slow responding to the reset causing the Source to send multiple ***Source_Capabilities*** Messages before it receives a

GoodCRC Message response. Figure 8-10 shows the Messages as they flow across the bus and within the devices to accomplish the Hard Reset.

Figure 8-10 Source initiated reset - Sink long reset

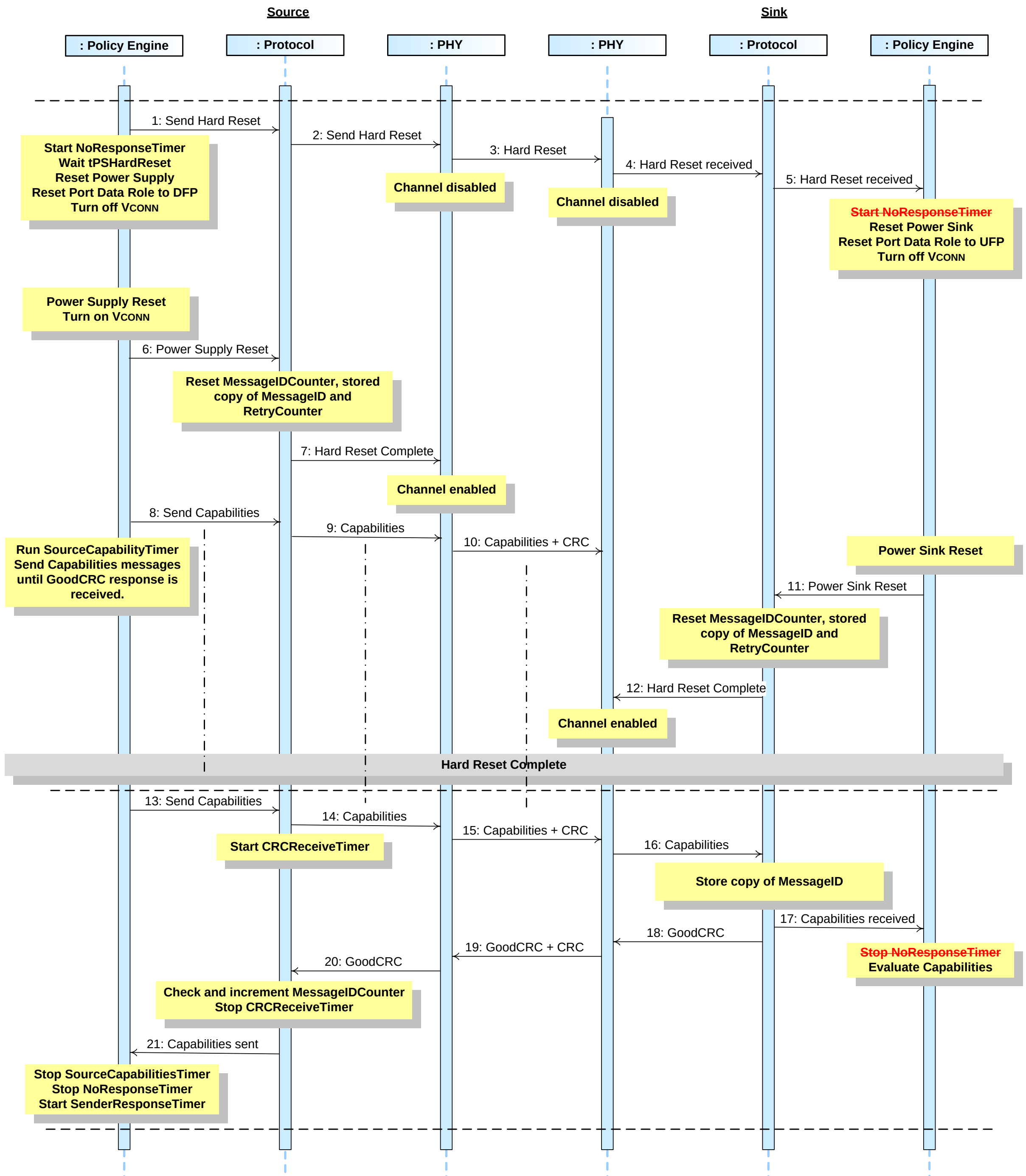


Table 8-10 Steps for Source initiated Hard Reset – Sink long reset

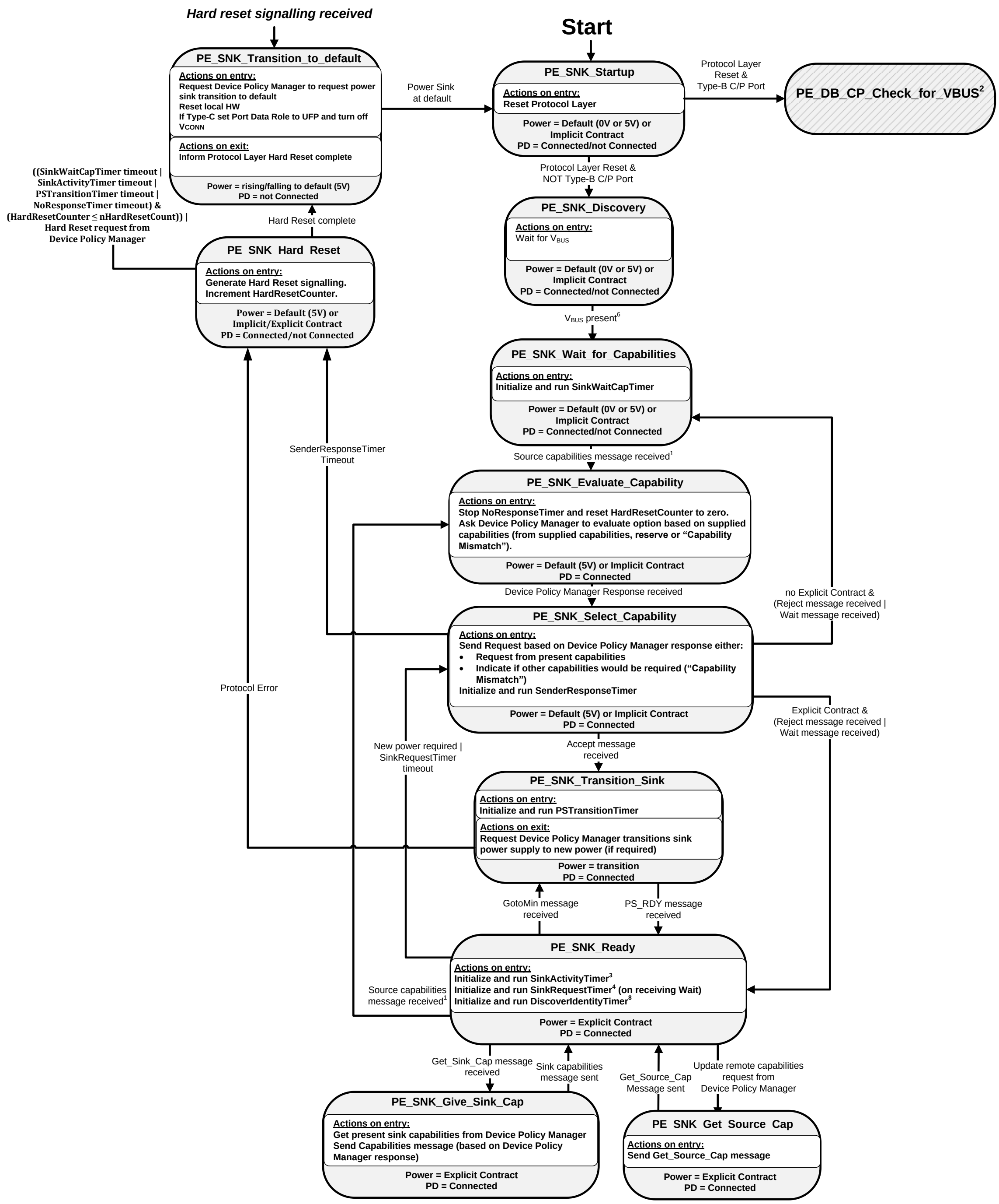
Step	Source	Sink
1	The Policy Engine directs the Protocol Layer to generate ***Hard Reset*** Signaling. The Policy Engine starts the ***NoResponseTimer*** and requests the Device Policy Manager to reset the power supply to USB Default Operation. If this is a USB Type-C connector the Policy Engine requests the Device Policy Manager to reset the Port Data Role to DFP and to turn off VCONN if this is on.	
2	Protocol Layer resets ***MessageIDCounter***, stored copy of ***MessageID*** and ***RetryCounter***. Protocol Layer requests the Physical Layer send ***Hard Reset*** Signaling.	
3	Physical Layer sends the ***Hard Reset*** Signaling and then disables the PHY Layer communications channel for transmission and reception.	Physical Layer receives the ***Hard Reset*** Signaling and disables the PHY Layer communications channel for transmission and reception.
4		Physical Layer informs the Protocol Layer of the Hard Reset. Protocol Layer resets ***MessageIDCounter***, stored copy of ***MessageID*** and ***RetryCounter***.
5		The Protocol Layer Informs the Policy Engine of the Hard Reset. The Policy Engine ~~starts the **_NoResponseTimer_** and~~ requests the Device Policy Manager to reset the Power Sink to default operation. If this is a USB Type-C connector the Policy Engine requests the Device Policy Manager to reset the Port Data Role to UFP and to turn off VCONN if this is on.
6	The power supply is reset to USB Default Operation. If this is a USB Type-C connector VCONN is turned on. The Policy Engine informs the Protocol Layer that the power supply has been reset.	
7	The Protocol Layer informs the PHY Layer that the Hard Reset is complete. The PHY Layer enables the PHY Layer communications channel for transmission and reception.	
	The reset is complete and protocol communication can restart.	
8	Policy Engine directs the Protocol Layer to send a ***Source_Capabilities*** Message that represents the power supply's present capabilities. Policy Engine starts the ***SourceCapabilityTimer***. The ***SourceCapabilityTimer*** times out one or more times until a ***GoodCRC*** Message response is received.	
9	Protocol Layer creates the Message and passes to Physical Layer. Starts ***CRCReceiveTimer***.	
10	Physical Layer appends CRC and sends the ***Source_Capabilities*** Message.	Note: ***Source_Capabilities*** Message not received since channel is disabled.
11		The Power Sink returns to USB Default Operation. The Policy Engine informs the Protocol Layer that the Power Sink has been reset.

Step	Source	Sink
12		The Protocol Layer informs the PHY Layer that the Hard Reset is complete. The PHY Layer enables the PHY Layer communications channel for transmission and reception.
	The reset is complete and protocol communication can restart.	
13	Policy Engine directs the Protocol Layer to send a ***Source_Capabilities*** Message that represents the power supply's present capabilities. Starts the ***SourceCapabilityTimer***.	
14	Protocol Layer creates the Message and passes to Physical Layer. Starts ***CRCReceiveTimer***.	
15	Physical Layer appends CRC and sends the ***Source_Capabilities*** Message.	Physical Layer receives the ***Source_Capabilities*** Message and checks the CRC to verify the Message.
16		Physical Layer removes the CRC and forwards the ***Source_Capabilities*** Message to the Protocol Layer.
17		Protocol Layer stores the ***MessageID*** of the incoming Message. The Protocol Layer forwards the received ***Source_Capabilities*** Message information to the Policy Engine that consumes it. ~~The Policy Engine stops the~~ ~~***NoResponseTimer***.~~
18		Protocol Layer generates a ***GoodCRC*** Message and passes it Physical Layer.
19	Physical Layer receives the ***GoodCRC*** Message and checks the CRC to verify the Message.	Physical Layer appends CRC and sends the ***GoodCRC*** Message.
20	Physical Layer removes the CRC and forwards the ***GoodCRC*** Message to the Protocol Layer.	
21	Protocol Layer verifies and increments the ***MessageIDCounter*** and stops ***CRCReceiveTimer***. Protocol Layer informs the Policy Engine that the ***Source_Capabilities*** Message was successfully sent. Policy Engine stops the ***SourceCapabilityTimer***, stops the ***NoResponseTimer*** and starts the ***SenderResponseTimer***.	
	USB Power Delivery communication is re-established.	

(c). Section 8.3.3.3, Page 404, Figure 8-43

From Text:

Figure 8-43 Sink Port state diagram



1 Source capabilities messages received in states other than ***PE_SNK_Wait_for_Capabilities*** and ***PE_SNK_Ready*** constitute a Protocol Error.

2 The ***SinkActivityTimer*** shall not be run when operating at ***vSafe5V*** or when two systems using the USB Type-C connector are communicating, since ***Ping*** messages are optional.

[4] The ***SinkRequestTimer*** should not be stopped if a ***Ping*** Message is received in the ***PE_SNK_Ready*** state since it represents the maximum time between requests after a ***Wait*** Message which is not reset by a ***Ping*** Message.

[5] For USB Type-C connectors Error Recovery steps can be taken at this point which are defined in the ***[USB Type-C 1.2]*** specification and are outside the scope of this specification.

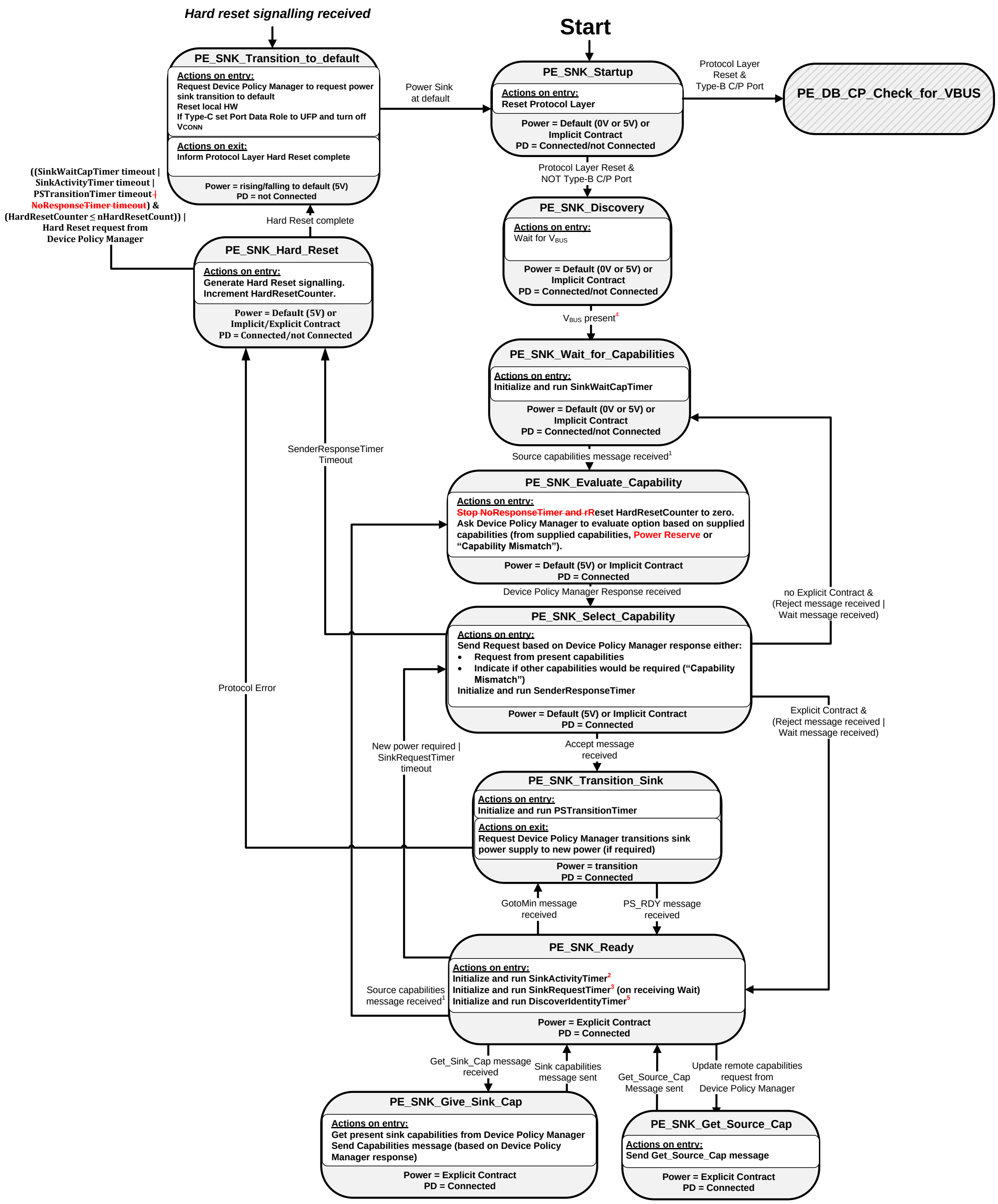
[6] During a Hard Reset the Source voltage will transition to ***vSafe0V*** and then transition to ***vSafe5V***. Sinks need to ensure that V_{BUS} present is not indicated until after the Source has completed the Hard Reset process by detecting both of these transitions.

[7] PD Connected is defined as a situation when the Port Partners have exchanged a Message and ***GoodCRC*** response. The Port Partners remain PD Connected after a Swap or the connector is able to identify a disconnect (USB Type-C, USB Type-A with insert detect, USB Micro-AB).

[8] The ***DiscoverIdentityTimer*** is run when this is a VCONN Source and a PD Connection with a Cable Plug needs to be established i.e. no ***GoodCRC*** Message has yet been received in response to a ***Discover Identity*** Command.PE_SNK_Startup state

To Text:

Figure 8-43 Sink Port state diagram



[1] Source capabilities messages received in states other than ***PE_SNK_Wait_for_Capabilities*** and ***PE_SNK_Ready*** constitute a Protocol Error.

[2] The ***SinkActivityTimer*** shall not be run when operating at ***vSafe5V*** or when two systems using the USB Type-C connector are communicating, since ***Ping*** messages are optional.

[3] The ***SinkRequestTimer*** should not be stopped if a ***Ping*** Message is received in the ***PE_SNK_Ready*** state since it represents the maximum time between requests after a ***Wait*** Message which is not reset by a ***Ping*** Message.

~~[5] For USB Type-C connectors Error Recovery steps can be taken at this point which are defined in the ***[USB Type-C 1.2]*** specification and are outside the scope of this specification.~~

[4] During a Hard Reset the Source voltage will transition to ***vSafe0V*** and then transition to ***vSafe5V***. Sinks need to ensure that V_{BUS} present is not indicated until after the Source has completed the Hard Reset process by detecting both of these transitions.

~~[7] PD Connected is defined as a situation when the Port Partners have exchanged a Message and ***GoodCRC*** response. The Port Partners remain PD Connected after a Swap or the connector is able to identify a disconnect (USB Type-C, USB Type-A with insert detect, USB Micro-AB).~~

[5] The ***DiscoverIdentityTimer*** is run when this is a VCONN Source and a PD Connection with a Cable Plug needs to be established i.e. no ***GoodCRC*** Message has yet been received in response to a ***Discover Identity*** Command.~~PE_SNK_Startup state~~

(d). Section 8.3.3.3.2, Page 405

From Text:

8.3.3.3.2 PE_SNK_Discovery state

In the ***PE_SNK_Discovery*** state the Sink Policy Engine waits for V_{BUS} to be present.

The Policy Engine shall transition to the ***PE_SNK_Wait_for_Capabilities*** state when:

- The Device Policy Manager indicates that V_{BUS} has been detected.

The Policy Engine shall transition to the ***ErrorRecovery*** state when:

- The Port is a USB Type-C connector and
- The Port Partners have previously been PD Connected (the USB Type-C Port remains Attached to a Port it has had a PD Connection with during this attachment) and
- There has been a ***NoResponseTimer*** timeout and
- The ***HardResetCounter*** > ***nHardResetCount***.

The Policy Engine shall transition to the PE_SNK_Hard_Reset state when:

- The Port is not a USB Type-C connector and
- There has been a ***NoResponseTimer*** timeout and
- The ***HardResetCounter*** ≤ ***nHardResetCount***.

To Text:

8.3.3.3.2 PE_SNK_Discovery state

In the ***PE_SNK_Discovery*** state the Sink Policy Engine waits for V_{BUS} to be present.

The Policy Engine shall transition to the ***PE_SNK_Wait_for_Capabilities*** state when:

- The Device Policy Manager indicates that V_{BUS} has been detected.

~~The Policy Engine shall transition to the ***ErrorRecovery*** state when:~~

- ~~The Port is a USB Type-C connector and~~
- ~~The Port Partners have previously been PD Connected (the USB Type-C Port remains Attached to a Port it has had a PD Connection with during this attachment) and~~
- ~~There has been a ***NoResponseTimer*** timeout and~~
- ~~The ***HardResetCounter*** > ***nHardResetCount***.~~

The Policy Engine shall transition to the ***PE_SNK_Hard_Reset*** state when:

- The Port is not a USB Type-C connector and
- There has been a ***NoResponseTimer*** timeout and
- The ***HardResetCounter*** ≤ ***nHardResetCount***.

(e). Section 8.3.3.3.9, Page 408

From Text:

8.3.3.3.9 PE_SNK_Transition_to_default state

The Policy Engine shall transition from any state to ***PE_SNK_Transition_to_default*** state when:

- ***Hard Reset*** Signaling is detected.

When ***Hard Reset*** Signaling is received or transmitted then the Policy Engine shall transition from any state to ***PE_SNK_Transition_to_default***. This state can also be entered from the ***PE_SNK_Hard_Reset*** state.

On entry to the ***PE_SNK_Transition_to_default*** state the Policy Engine shall:

- indicate to the Device Policy Manager that the Sink shall transition to default
- request a reset of the local hardware
- for a USB Type-C connector shall request that the Port Data Role is set to UFP.

On exit from the ***PE_SNK_Transition_to_default*** state the Policy Engine shall initialize and run the ***NoResponseTimer*** and inform the Protocol Layer that the Hard Reset is complete. Note that the ***NoResponseTimer*** shall continue to run in every state until it is stopped or times out.

The Policy Engine shall transition to the ***PE_SNK_Startup*** state when:

- The Device Policy Manager indicates that the Sink has reached the default level.

To Text:

8.3.3.3.9 PE_SNK_Transition_to_default state

The Policy Engine shall transition from any state to ***PE_SNK_Transition_to_default*** state when:

- ***Hard Reset*** Signaling is detected.

When ***Hard Reset*** Signaling is received or transmitted then the Policy Engine shall transition from any state to ***PE_SNK_Transition_to_default***. This state can also be entered from the ***PE_SNK_Hard_Reset*** state.

On entry to the ***PE_SNK_Transition_to_default*** state the Policy Engine shall:

- indicate to the Device Policy Manager that the Sink shall transition to default

- request a reset of the local hardware
- for a USB Type-C connector shall request that the Port Data Role is set to UFP.

~~On exit from the ***PE_SNK_Transition_to_default*** state the Policy Engine shall initialize and run the ***NoResponseTimer*** and inform the Protocol Layer that the Hard Reset is complete. Note that the ***NoResponseTimer*** shall continue to run in every state until it is stopped or times out.~~

The Policy Engine shall transition to the ***PE_SNK_Startup*** state when:

- The Device Policy Manager indicates that the Sink has reached the default level.

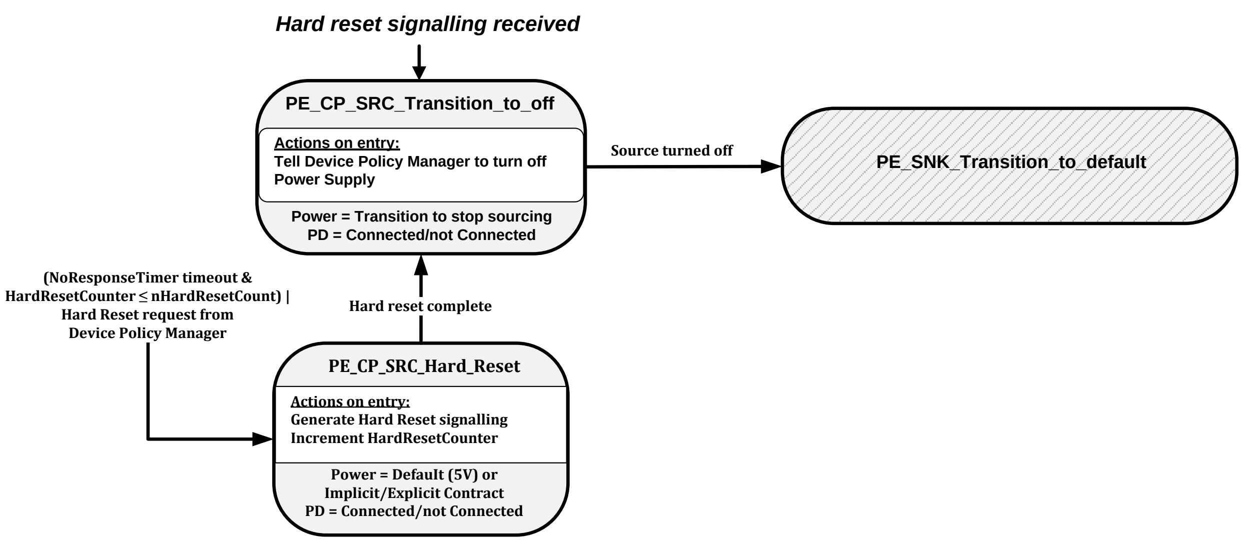
(f). Section 8.3.3.6.1.3, Page 413

From Text:

8.3.3.6.1.3 Type-A/B Hard Reset of Policy Engine in a Provider/Consumer in Sink Role

Figure 8-49 shows the state diagram in the case where a Provider/Consumer with a Port operating in Sink Role is required to perform a Hard Reset.

Figure 8-49 State Diagram for Hard Reset of P/C in Sink Role



8.3.3.6.1.3.1 PE_PC_SNK_Hard_Reset state

The Policy Engine shall transition to the ***PE_PC_SNK_Hard_Reset*** state for a Provider/Consumer Port in Sink Role from any state when:

- ((***SinkWaitCapTimer*** timeout |
- ***SinkActivityTimer*** timeout |
- ***PSTransitionTimer*** timeout |
- ***NoResponseTimer*** timeout) &
- (***HardResetCounter*** ≤ ***nHardResetCount***)) |
- Hard Reset request from Device Policy Manager

The Policy Engine shall transition to the ***PE_PC_SNK_Swap_Recovery*** state when:

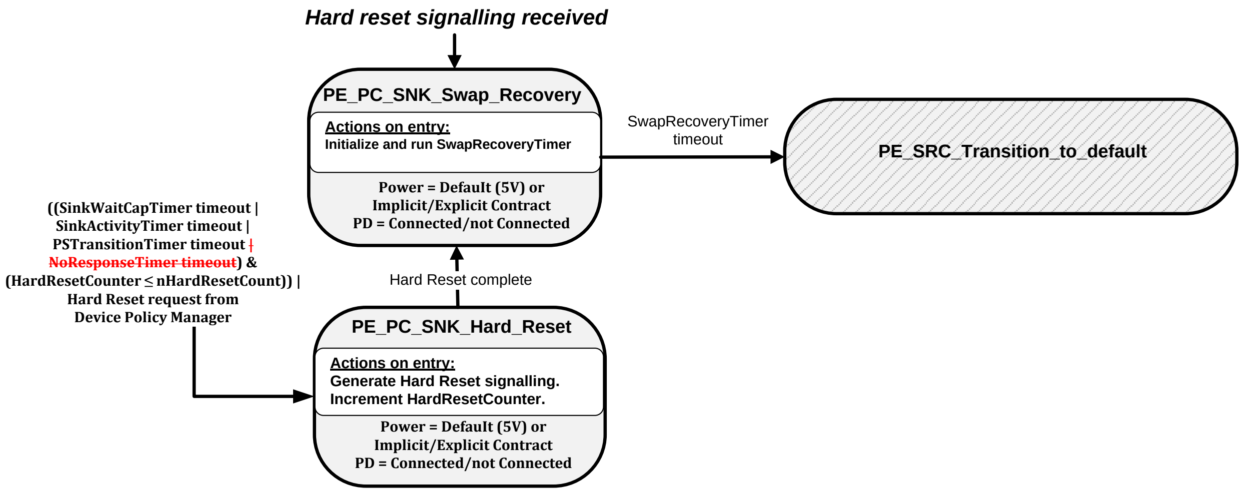
- The Hard Reset is complete.

To Text:

8.3.3.6.1.3 Type-A/B Hard Reset of Policy Engine in a Provider/Consumer in Sink Role

Figure 8-49 shows the state diagram in the case where a Provider/Consumer with a Port operating in Sink Role is required to perform a Hard Reset.

Figure 8-49 State Diagram for Hard Reset of P/C in Sink Role



8.3.3.6.1.3.1 PE_PC_SNK_Hard_Reset state

The Policy Engine shall transition to the ***PE_PC_SNK_Hard_Reset*** state for a Provider/Consumer Port in Sink Role from any state when:

- ((***SinkWaitCapTimer*** timeout |
- ***SinkActivityTimer*** timeout |
- ***PSTransitionTimer*** timeout~~|~~
- ~~***NoResponseTimer*** timeout~~) &
- (***HardResetCounter*** ≤ ***nHardResetCount***)) |
- Hard Reset request from Device Policy Manager

The Policy Engine shall transition to the ***PE_PC_SNK_Swap_Recovery*** state when:

- The Hard Reset is complete.