31/10/2023 18:54:21

Compare Results

Old File:

USB_PD_R3_1 V1.8 2023-04_Ch5.pdf

29 pages (1.89 MB)

13/10/2023 19:35:33

versus

New File:

USB_PD_R3_2 V1.0 2023-10_Ch 5.pdf

36 pages (1.26 MB)

31/10/2023 18:08:22

Total Changes

547

Text only comparison

Content

308	Replacements
142	Insertions
97	Deletions

Styling and Annotations

0	Styling
0	Annotations

Go to First Change (page 1)

5. Physical Layer

5.1 Physical Layer Overview

The Physical Layer (PHY Layer) defines the signaling technology for USB Power Delivery. This chapter defines the electrical requirements and parameters of the PD Physical Layer required for interoperability between USB PD devices.

5.2 Physical Layer Functions

The USB PD Physical Layer consists of a pair of transmitters and receivers that communicate across a single signal wire (CC). All communication is half duplex. The PHY Layer practices collision avoidance to minimize communication errors on the channel.

The transmitter performs the following functions:

- Receive packet data from the protocol layer.
- Calculate and append a CRC.
- Encode the packet data including the CRC (i.e., the payload).
- Transmit the Packet (Preamble, ***SOP****, payload, CRC and ***EOP***) across the channel using Biphase Mark Coding (BMC) over CC.

The receiver performs the following functions:

- Recover the clock and lock onto the Packet from the Preamble.
- Detect the ***SOP****.
- Decode the received data including the CRC.
- Detect the ***EOP*** and validate the CRC:
    - If the CRC is ***Valid***, deliver the packet data to the protocol layer.
    - If the CRC is Invalid, flush the received data.

5.3 Symbol Encoding

Except for the Preamble, all communications on the line ***Shall*** be encoded with a line code to ensure a reasonable level of DC-balance and a suitable number of transitions. This encoding makes receiver design less complicated and allows for more variations in the receiver design.

4b5b line code ***Shall*** be used. This encodes 4-bit data to 5-bit symbols for transmission and decodes 5-bit symbols to 4-bit data for consumption by the receiver.

The 4b5b code provides data encoding along with special symbols. Special symbols are used to signal ***Hard Reset***, and delineate packet boundaries (see *Table 5.1 "4b5b Symbol Encoding Table"*).

Table 5.1 "4b5b Symbol Encoding Table"

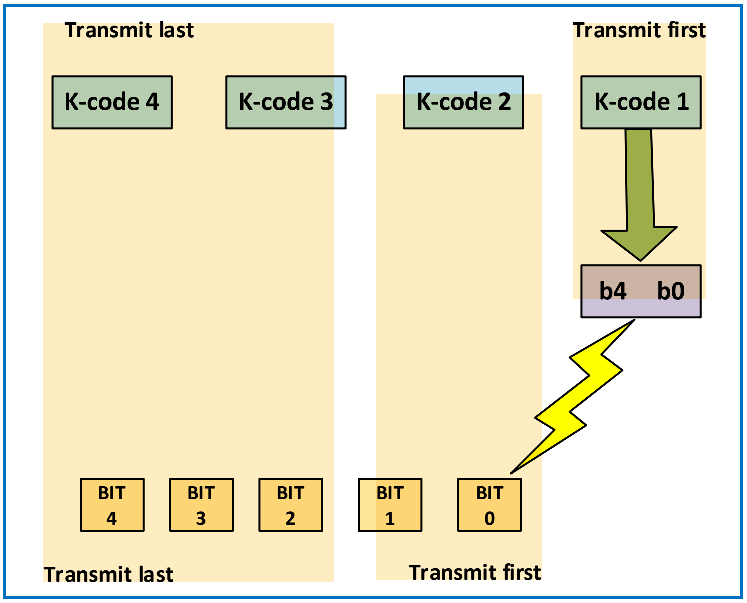
Name	4b	5b Symbol	Description
0	0000	11110	hex data 0
1	0001	01001	hex data 1
2	0010	10100	hex data 2
3	0011	10101	hex data 3
4	0100	01010	hex data 4
5	0101	01011	hex data 5
6	0110	01110	hex data 6
7	0111	01111	hex data 7
8	1000	10010	hex data 8
9	1001	10011	hex data 9
A	1010	10110	hex data A
B	1011	10111	hex data B
C	1100	11010	hex data C
D	1101	11011	hex data D
E	1110	11100	hex data E
F	1111	11101	hex data F
Sync-1	K-code	11000	Startsynch #1
Sync-2	K-code	10001	Startsynch #2
RST-1	K-code	00111	Hard Reset #1
RST-2	K-code	11001	Hard Reset #2
EOP	K-code	01101	EOP End of Packet
Reserved	Error	00000	***Shall Not*** be used
Reserved	Error	00001	***Shall Not*** be used
Reserved	Error	00010	***Shall Not*** be used
Reserved	Error	00011	***Shall Not*** be used
Reserved	Error	00100	***Shall Not*** be used
Reserved	Error	00101	***Shall Not*** be used
Sync-3	K-code	00110	Startsynch #3
Reserved	Error	01000	***Shall Not*** be used
Reserved	Error	01100	***Shall Not*** be used
Reserved	Error	10000	***Shall Not*** be used
Reserved	Error	11111	***Shall Not*** be used

5.4 Ordered Sets

Ordered sets ***Shall*** be interpreted according to *Figure 5-1 "Interpretation of ordered sets"*.

An ordered set consists of 4 K-codes sent as shown in *Figure 5-1 "Interpretation of ordered sets"*.

Figure 5-1 "Interpretation of ordered sets"



A list of the ordered sets used by USB Power Delivery can be seen in *Table 5.2 "Ordered Sets*. ***SOP****** is a generic term used in place of ***SOP***/***SOP'***/***SOP"***.

Table 5.2 "Ordered Sets"

Ordered Set	Reference
Cable Reset	*Section 5.6.5*
Hard Reset	*Section 5.6.4*
SOP	*Section 5.6.1.2.1*
SOP'	*Section 5.6.1.2.2*
SOP'_Debug	*Section 5.6.1.2.4*
SOP"	*Section 5.6.1.2.3*
SOP"_Debug	*Section 5.6.1.2.5*

The receiver ***Shall*** search for all four K-codes. When the receiver finds all four K-codes in the correct place, it ***Shall*** interpret this as a ***Valid*** ordered set. When the receiver finds three out of four K-codes in the correct place, it ***May*** interpret this as a ***Valid*** ordered set. The receiver ***Should*** ensure that all four K-codes are ***Valid*** to avoid ambiguity in detection (see *Table 5.3 "Validation of Ordered Sets*).

Table 5.3 “Validation of Ordered Sets”

	1st code	2nd code	3rd code	4th code
Valid[1]	Corrupt	K-code	K-code	K-code
Valid[1]	K-code	Corrupt	K-code	K-code
Valid[1]	K-code	K-code	Corrupt	K-code
Valid[1]	K-code	K-code	K-code	Corrupt
Valid[2] (perfect)	K-code	K-code	K-code	K-code
Invalid (example)	K-code	Corrupt	K-code	Corrupt

[1] ***May*** be interpreted as a ***Valid*** ordered set.

[2] ***Shall*** be interpreted as a ***Valid*** ordered set.

5.5 Transmitted Bit Ordering

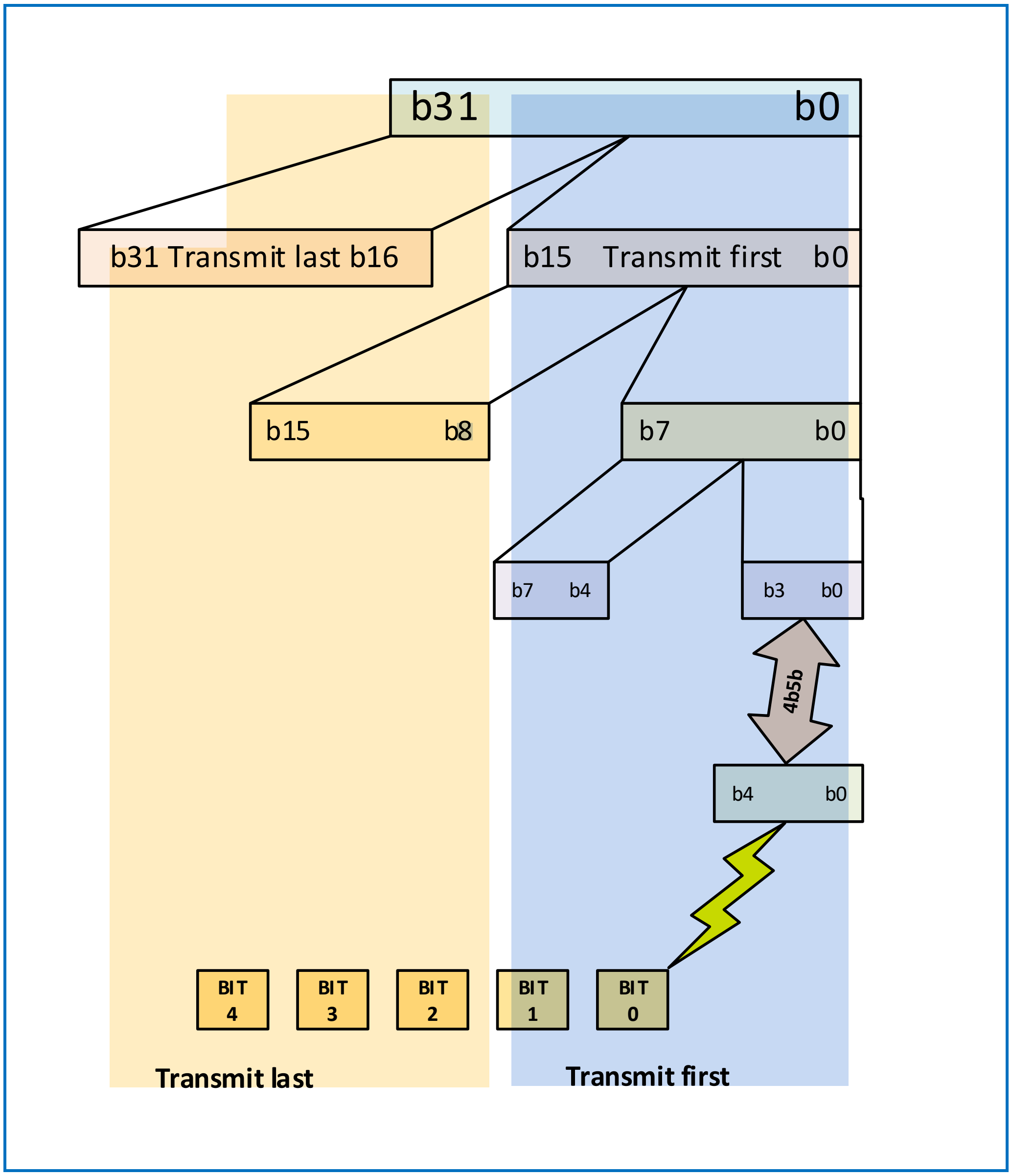
This section describes the order of bits on the wire that ***Shall*** be used when transmitting data of varying sizes. *Table 5.4 "Data Size"* shows the different data sizes that are possible.

Figure 5-2 "Transmit Order for Various Sizes of Data" shows the transmission order that ***Shall*** be followed.

Table 5.4 "Data Size"

	Unencoded	Encoded
Byte	8-bits	10-bits
Word	16-bits	20- bits
DWord	32-bits	40-bits

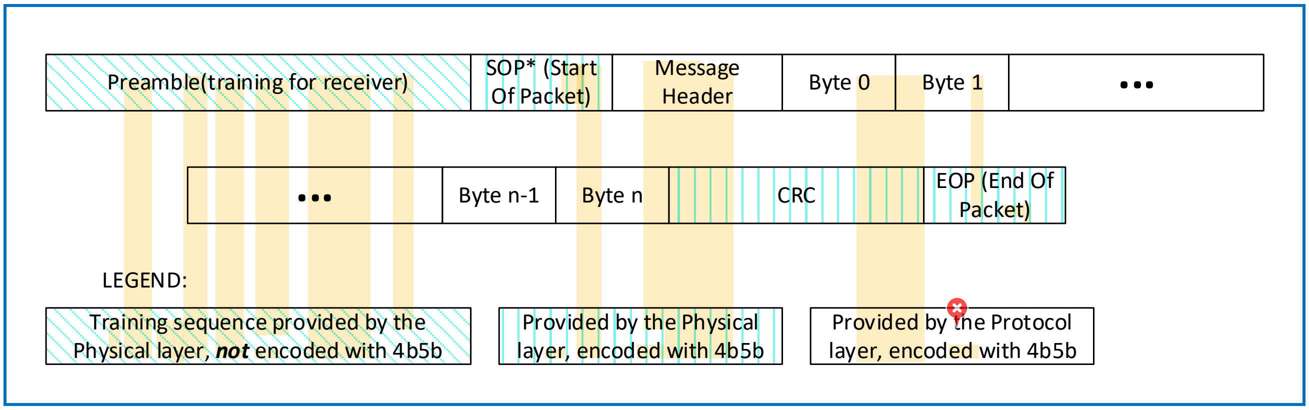
Figure 5-2 "Transmit Order for Various Sizes of Data"



5.6 Packet Format

The packet format ***Shall*** consist of a Preamble, an ***SOP****, (see *Section 5.6.1.2 "Start of Packet Sequences"*), packet data including the Message Header, a CRC and an ***EOP*** (see *Section 5.6.1.5 "End of Packet (EOP)"*). The packet format is shown in *Figure 5-3 "USB Power Delivery Packet Format"* and indicates which parts of the packet ***Shall*** be 4b/5b encoded. Once 4b/5b encoded, the entire Packet ***Shall*** be transmitted using BMC over CC. Note that all the bits in the Packet, including the Preamble, are BMC encoded. See *Section 6.2.1 "Message Construction"* for more details of the Packet construction for Control, Data and Extended Messages.

Figure 5-3 "USB Power Delivery Packet Format"


Preamble(training for receiver)
SOP* (Start Of Packet)
Message Header
Byte 0
Byte 1
•••
•••
Byte n-1
Byte n
CRC
EOP (End Of Packet)
LEGEND:
Training sequence provided by the Physical layer, *not* encoded with 4b5b
Provided by the Physical layer, encoded with 4b5b
Provided by the Protocol layer, encoded with 4b5b

5.6.1 Packet Framing

The transmission starts with a Preamble that is used to allow the receiver to lock onto the carrier. It is followed by a ***SOP**** (Start of Packet). The packet is terminated with an ***EOP*** (End of Packet) K-code.

5.6.1.1 Preamble

The Preamble is used to achieve lock in the receiver by presenting an alternating series of "0s" and "1s", so the average frequency is the carrier frequency. Unlike the rest of the packet, the Preamble ***Shall Not*** be 4b/5b encoded.

The Preamble ***Shall*** consist of a 64-bit sequence of alternating 0s and 1s. The Preamble ***Shall*** start with a "0" and ***Shall*** end with a "1".

5.6.1.2 Start of Packet Sequences

5.6.1.2.1 Start of Packet Sequence (SOP)

SOP is an ordered set. The ***SOP*** ordered set is defined as: three ***Sync-1*** K-codes followed by one ***Sync-2*** K-code (see *Table 5.5 "SOP ordered set"*).

Table 5.5 "SOP ordered set"

K-code number	K-code in code table
1	***Sync-1***
2	***Sync-1***
3	***Sync-1***
4	***Sync-2***

A Power Delivery Capable Source or Sink ***Shall*** be able to detect and communicate with packets using ***SOP***. If a ***Valid SOP*** is not detected (see ***Table 5.3 "Validation of Ordered Sets"***) then the whole transmission ***Shall*** be ***Discarded***.

Sending and receiving of SOP Packets ***Shall*** be limited to PD Capable Ports on PDUSB Hosts and PDUSB Devices. Cable Plugs and VPDs ***Shall*** neither send nor receive SOP Packets. Note that PDUSB Devices, even if they have the physical form of a cable (e.g., AMAs), are still required to respond to SOP Packets.

5.6.1.2.2 Start of Packet Sequence Prime (SOP')

The ***SOP'*** ordered set is defined as: two ***Sync-1*** K-codes followed by two ***Sync-3*** K-codes (see ***Table 5.6 "SOP' ordered set"***).

Table 5.6 "SOP' ordered set"

K-code number	K-code in code table
1	***Sync-1***
2	***Sync-1***
3	***Sync-3***
4	***Sync-3***

A VPD ***Shall*** have SOP' Communication capability. A VPD and a Cable Plug capable of SOP' Communications ***Shall*** only detect and communicate with packets starting with ***SOP'***.

A Port needing to communicate with a Cable Plug capable of SOP' Communications, Attached between a Port Pair will be able to communicate using both packets starting with ***SOP'*** to communicate with the Cable Plug and starting with ***SOP*** to communicate with its Port Partner.

For a VPD or a Cable Plug supporting SOP' Communications, if a ***Valid SOP'*** is not detected (see ***Table 5.3 "Validation of Ordered Sets"***) then the whole transmission ***Shall*** be ***Discarded***. For a Port supporting SOP' Communications if a ***Valid SOP*** or ***SOP'*** is not detected (see ***Table 5.3 "Validation of Ordered Sets"***) then the whole transmission ***Shall*** be ***Discarded***. When there is no Explicit Contract or an Implicit Contract in place a Sink ***Shall Not*** send SOP' Packets and ***Shall Discard*** all packets starting with ***SOP'***.

5.6.1.2.3 Start of Packet Sequence Double Prime (SOP'')

The ***SOP''*** ordered set is defined as the following sequence of K-codes: ***Sync-1***, ***Sync-3***, ***Sync-1***, ***Sync-3*** (see ***Table 5.7 "SOP'' ordered set"***).

Table 5.7 "SOP'' ordered set"

K-code number	K-code in code table
1	***Sync-1***
2	***Sync-3***
3	***Sync-1***
4	***Sync-3***

A VPD ***Shall Not*** have SOP'' Communication capability. A Cable Plug capable of SOP'' Communication, ***Shall*** have a SOP' Communication capability in the other Cable Plug. No cable ***Shall*** only support SOP'' Communication. A Cable Plug to which SOP'' Communication is assigned ***Shall*** only detect and communicate with packets starting with ***SOP''*** and ***Shall Discard*** any other packets.

A Port needing to communicate with such a Cable Plug, Attached between a Port Pair will be able to communicate using packets starting with *SOP'* and *SOP''* to communicate with the Cable Plugs and packets starting with *SOP* to communicate with its Port Partner. A Port which supports SOP'' Communication ***Shall*** also support SOP' Communication and ***Shall*** co-ordinate SOP* Communication so as to avoid collisions.

For the Cable Plug supporting SOP'' Communication, if a ***Valid*** *SOP''* is not detected (see *Table 5.3 "Validation of Ordered Sets"*) then the whole transmission ***Shall*** be Discarded. For the Port if a ***Valid*** *SOP** is not detected (see *Table 5.3 "Validation of Ordered Sets"*) then the whole transmission ***Shall*** be Discarded.

5.6.1.2.4 Start of Packet Sequence Prime Debug (SOP'_Debug)

The *SOP'_Debug* ordered set is defined as the following sequence of K-codes: *Sync-1*, *RST-2*, *RST-2*, *Sync-3* (see *Table 5.8 "SOP'_Debug ordered set"*). The usage of this Ordered Set is presently undefined.

Table 5.8 "SOP'_Debug ordered set"

K-code number	K-code in code table
1	*Sync-1*
2	*RST-2*
3	*RST-2*
4	*Sync-3*

5.6.1.2.5 Start of Packet Sequence Double Prime Debug (SOP''_Debug)

The *SOP''_Debug* ordered set is defined as the following sequence of K-codes: *Sync-1*, *RST-2*, *Sync-3, Sync-2* (see *Table 5.9 "SOP''_Debug ordered set"*). The usage of this Ordered Set is presently undefined.

Table 5.9 "SOP''_Debug ordered set"

K-code number	K-code in code table
1	*Sync-1*
2	*RST-2*
3	*Sync-3*
4	*Sync-2*

5.6.1.3 Packet Payload

The packet payload is delivered from the protocol layer (see *Section 6.2 "Messages"*) and ***Shall*** be encoded with the hex data codes from *Table 5.1 "4b5b Symbol Encoding Table"*.

5.6.1.4 CRC

The CRC ***Shall*** be inserted just after the payload. It is described in *Section 5.6.2 "CRC"*.

5.6.1.5 End of Packet (EOP)

The end of packet marker ***Shall*** be a single *EOP* K-code as defined in *Figure 5-1 "Interpretation of ordered sets"*. This ***Shall*** mark the end of the CRC. After the *EOP*, the CRC-residual ***Shall*** be checked. If the CRC is not good, the whole transmission ***Shall*** be ***Discarded***, if it is good, the packet ***Shall*** be delivered to the Protocol Layer. Note an *EOP* ***May*** be used to prematurely terminate a Packet e.g., before sending *Hard Reset* Signaling.

5.6.2 CRC

The Message Header and data ***Shall*** be protected by a 32-bit CRC.

- CRC-32 protects the data integrity of the data payload. CRC-32 is defined as follows:
- The CRC-32 polynomial ***Shall*** be = 04C1_1DB7h.
- The CRC-32 Initial value ***Shall*** be = FFFF_FFFFh.
- CRC-32 ***Shall*** be calculated for all bytes of the payload not inclusive of any packet framing symbols (i.e., excludes the Preamble, ***SOP****, ***EOP***).
- CRC-32 calculation ***Shall*** begin at byte 0, bit 0 and continue to bit 7 of each of the bytes of the packet.
- The remainder of CRC-32 ***Shall*** be complemented.
- The residual of CRC-32 ***Shall*** be C704 DD7Bh.

This inversion of the CRC-32 remainder adds an offset of FFFF_FFFFh that will create a constant CRC-32 residual of C704_DD7Bh at the receiver side.

Note: The CRC implementation is identical to the one used in ***[USB 3.2]***.

Figure 5-4 "CRC 32 generation" is an illustration of CRC-32 generation. The output bit ordering ***Shall*** be as detailed in ***Table 5.10 "CRC-32 Mapping"***.

Figure 5-4 "CRC 32 generation"

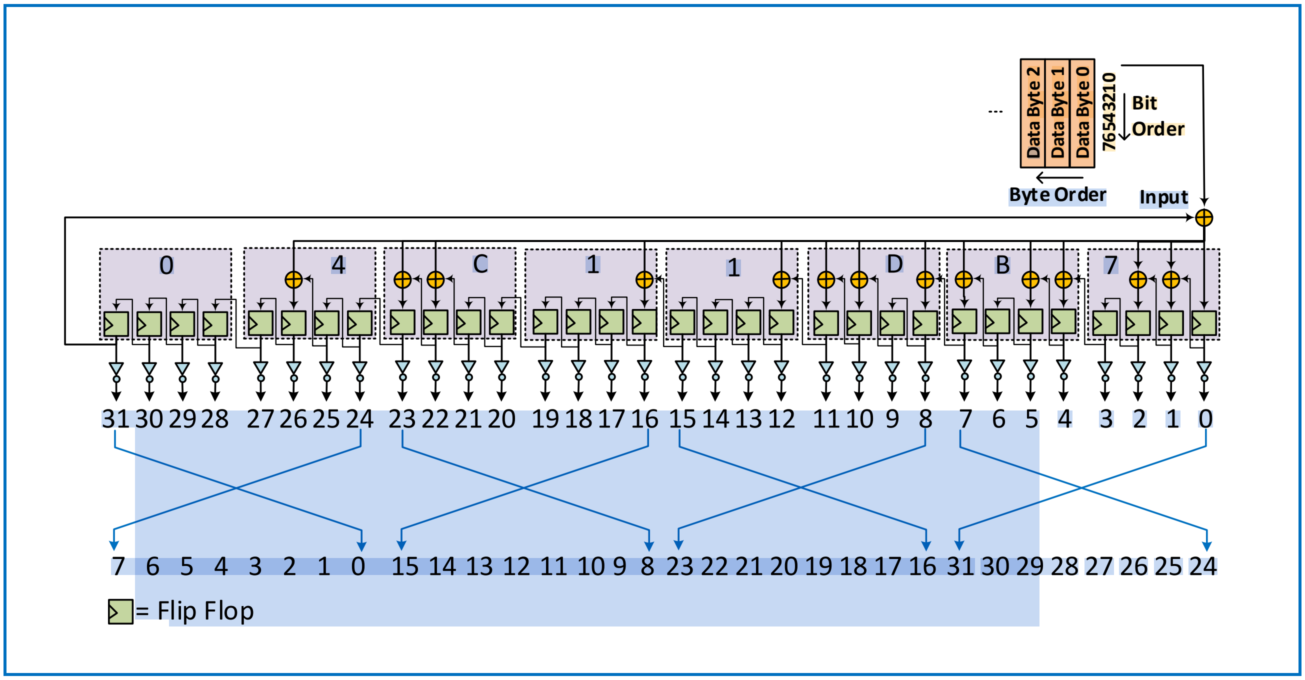


Table 5.10 “CRC-32 Mapping”

CRC-32	Result bit Position in CRC-32 Field
0	31
1	30
2	29
3	28
4	27
5	26
6	25
7	24
8	23
9	22
10	21
11	20
12	19
13	18
14	17
15	16
16	15
17	14
18	13
19	12
20	11
21	10
22	9
23	8
24	7
25	6
26	5
27	4
28	3
29	2
30	1
31	0

The CRC-32 ***Shall*** be encoded before transmission.

5.6.3 Packet Detection Errors

CRC errors, or errors detected while decoding encoded symbols using the code table, ***Shall*** be treated the same way; the Message ***Shall*** be ***Discarded*** and a ***GoodCRC*** Message ***Shall Not*** be returned.

While the receiver is processing a packet, if at any time the CC-line becomes idle the receiver ***Shall*** stop processing the packet and ***Discard*** it (no ***GoodCRC*** Message is returned). See ***Section 5.8.6.1 "Definition of Idle"*** for the definition of BMC idle.

5.6.4 Hard Reset

Hard Reset Signaling is an ordered set of bytes sent with the purpose to be recognized by the PHY Layer. The ***Hard Reset*** Signaling ordered set is defined as: three ***RST-1*** K-codes followed by one ***RST-2*** K-code (see ***Table 5.11 "Hard Reset ordered set"***).

Table 5.11 "Hard Reset ordered set"

K-code number	K-code in code table
1	***RST-1***
2	***RST-1***
3	***RST-1***
4	***RST-2***

A device ***Shall*** perform a Hard Reset when it receives ***Hard Reset*** Signaling. After receiving the ***Hard Reset*** Signaling, the device ***Shall*** reset as described in ***Section 6.8.3 "Hard Reset"***. If a ***Valid*** ***Hard Reset*** is not detected (see ***Table 5.3 "Validation of Ordered Sets"***) then the whole transmission ***Shall*** be ***Discarded***.

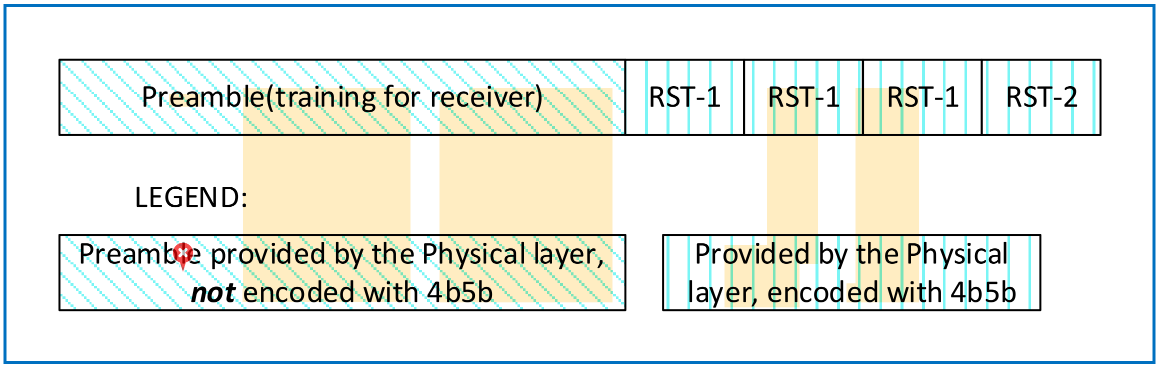
A Cable Plug ***Shall*** perform a Hard Reset when it detects ***Hard Reset*** Signaling being sent between the Port Partners. After receiving the ***Hard Reset*** Signaling, the device ***Shall*** reset as described in ***Section 6.8.3 "Hard Reset"***.

The procedure for sending ***Hard Reset*** Signaling ***Shall*** be as follows:

- If the PHY Layer is currently sending a Message, the Message ***Shall*** be interrupted by sending an ***EOP*** K-code and the rest of the Message ***Discarded***.
- If CC is not idle, wait for it to become idle (see ***Section 5.8.6.1 "Definition of Idle"***).
- Wait ***tInterFrameGap.***
- If CC is still idle send the Preamble followed by the 4 K-codes for ***Hard Reset*** Signaling.
- Disable the channel (i.e., stop sending and receiving), reset the PHY Layer and inform the Protocol Layer that the PHY Layer has been reset.
- Re-enable the channel when requested by the Protocol Layer.

Figure 5-5 "Line format of Hard Reset" shows the line format of ***Hard Reset*** Signaling which is a Preamble followed by the ***Hard Reset*** Ordered Set.

Figure 5-5 "Line format of Hard Reset"


Preamble(training for receiver)
RST-1
RST-1
RST-1
RST-2
LEGEND:
Preamble provided by the Physical layer, *not* encoded with 4b5b
Provided by the Physical layer, encoded with 4b5b

5.6.5 Cable Reset

Cable Reset Signaling is an ordered set of bytes sent with the purpose to be recognized by the PHY Layer. The *Cable Reset* Signaling ordered set is defined as the following sequence of K-codes: *RST-1*, *Sync-1*, *RST-1*, *Sync-3* (see *Table 5.12 "Cable Reset ordered set"*).

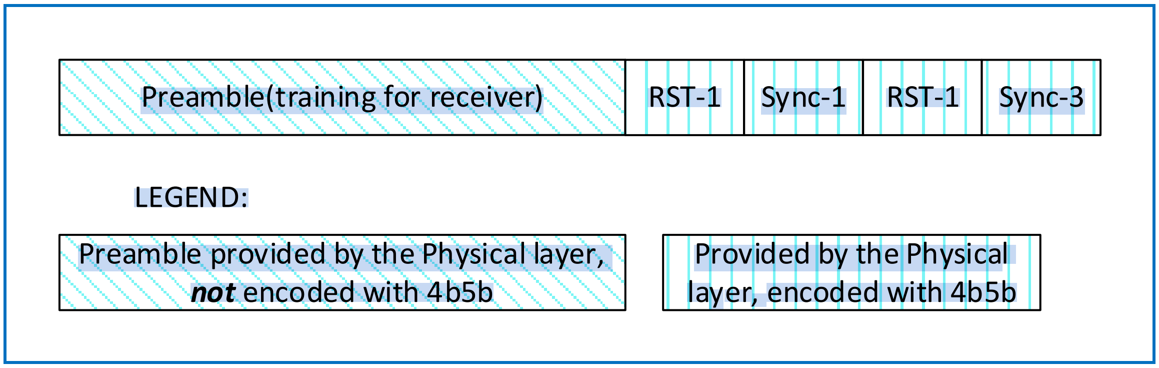
Table 5.12 "Cable Reset ordered set"

K-code number	K-code in code table
1	*RST-1*
2	*Sync-1*
3	*RST-1*
4	*Sync-3*

Cable Reset Signaling ***Shall*** only be sent by the DFP. The *Cable Reset* Ordered Set is used to reset the Cable Plugs without the need to Hard Reset the Port Partners. The state of the Cable Plug after the *Cable Reset* Signaling ***Shall*** be equivalent to power cycling the Cable Plug.

Figure 5-6 "Line format of Cable Reset" shows the line format of *Cable Reset* Signaling which is a Preamble followed by the *Cable Reset* Ordered Set.

Figure 5-6 "Line format of Cable Reset"


Preamble(training for receiver)
RST-1
Sync-1
RST-1
Sync-3
LEGEND:
Preamble provided by the Physical layer, *not* encoded with 4b5b
Provided by the Physical layer, encoded with 4b5b

5.7 Collision Avoidance

The PHY Layer ***Shall*** monitor the channel for data transmission and only initiate transmissions when CC is idle. If the bus idle condition is present, it ***Shall*** be considered safe to start a transmission provided the conditions detailed in ***Section 5.8.5.4 "Inter-Frame Gap"*** are met. The bus idle condition ***Shall*** be checked immediately prior to transmission. If transmission cannot be initiated, then the packet ***Shall*** be ***Discarded***. If the packet is ***Discarded*** because CC is not idle, the PHY Layer ***Shall*** signal to the protocol layer that it has ***Discarded*** the Message as soon as CC becomes idle. See ***Section 5.8.6.1 "Definition of Idle"*** for the definition of idle CC.

In addition, during an Explicit Contract, the PHY Layer ***Shall*** control the R_p resistor value to avoid collisions between Source and Sink transmissions. The Source ***Shall*** set an R_p value corresponding to a current of 3A to indicate to the Sink that it ***May*** initiate an AMS. The Source ***Shall*** set an R_p value corresponding to a current of 1.5A this ***Shall*** indicate to the Sink that it ***Shall Not*** initiate an AMS and ***Shall*** only respond to Messages as part of an AMS. See ***[USB Type-C 2.3]*** (USB Type-C®) for details of the corresponding R_p values. During the Implicit Contract that precedes an Explicit Contract (including Power Role Swap and Fast Role Swap) the R_p resistor value is used to specify USB Type-C® current and is not used for collision avoidance.

Table 5.13 "Rp values used for Collision Avoidance" details the R_p values that ***Shall*** be used by the Source to control Sink initiation of an AMS.

Table 5.13 "R_p values used for Collision Avoidance"

Source R_p	Parameter	Description	Sink operation	Source operation
1.5A@5V	***SinkTxNG***	Sink Transmit "No Go,"	The Sink ***Shall Not*** initiate an AMS once ***tSinkDelay*** has elapsed after ***SinkTxNG*** is asserted.	Source can initiate an AMS ***tSinkTx*** after setting R_p to this value.
3A@5V	***SinkTxOk***	Sink Transmit "Ok"	Sink can initiate an AMS.	Source cannot initiate an AMS while it has this value set.

See also ***Section 6.6.16 "Collision Avoidance Timers"*** and ***Section 6.10 "Collision Avoidance"***.

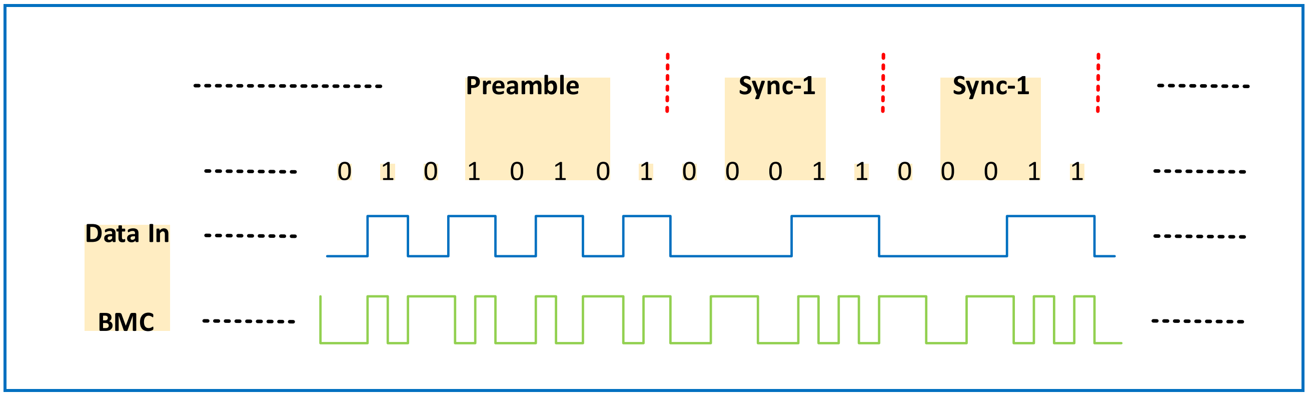
5.8 Biphase Mark Coding (BMC) Signaling Scheme

Biphase Mark Coding (BMC) is the physical layer Signaling Scheme for carrying USB Power Delivery Messages. This encoding assumes a dedicated DC connection, identified as the CC wire, which is used for sending PD Messages.

Biphase Mark Coding is a version of Manchester coding (see *[IEC 60958-1]*). In BMC, there is a transition at the start of every bit time (UI) and there is a second transition in the middle of the UI when a 1 is transmitted. BMC is effectively DC balanced, (each 1 is DC balanced and two successive zeroes are DC balanced, regardless of the number of intervening 1's). It has bounded disparity (limited to 1 bit over an arbitrary packet, so a very low DC level).

Figure 5-7 "BMC Example" illustrates Biphase Mark Coding. This example shows the transition from a Preamble to the *Sync-1* K-codes of the *SOP* Ordered Set at the start of a Message. Note that other K-codes can occur after the Preamble for Signaling such as *Hard Reset* and *Cable Reset*.

Figure 5-7 "BMC Example"



5.8.1 Encoding and signaling

BMC uses DC coupled baseband signaling on CC. *Figure 5-8 "BMC Transmitter Block Diagram"* shows a block diagram for a Transmitter and *Figure 5-9 "BMC Receiver Block Diagram"* shows a block diagram for the corresponding Receiver.

Figure 5-8 "BMC Transmitter Block Diagram"

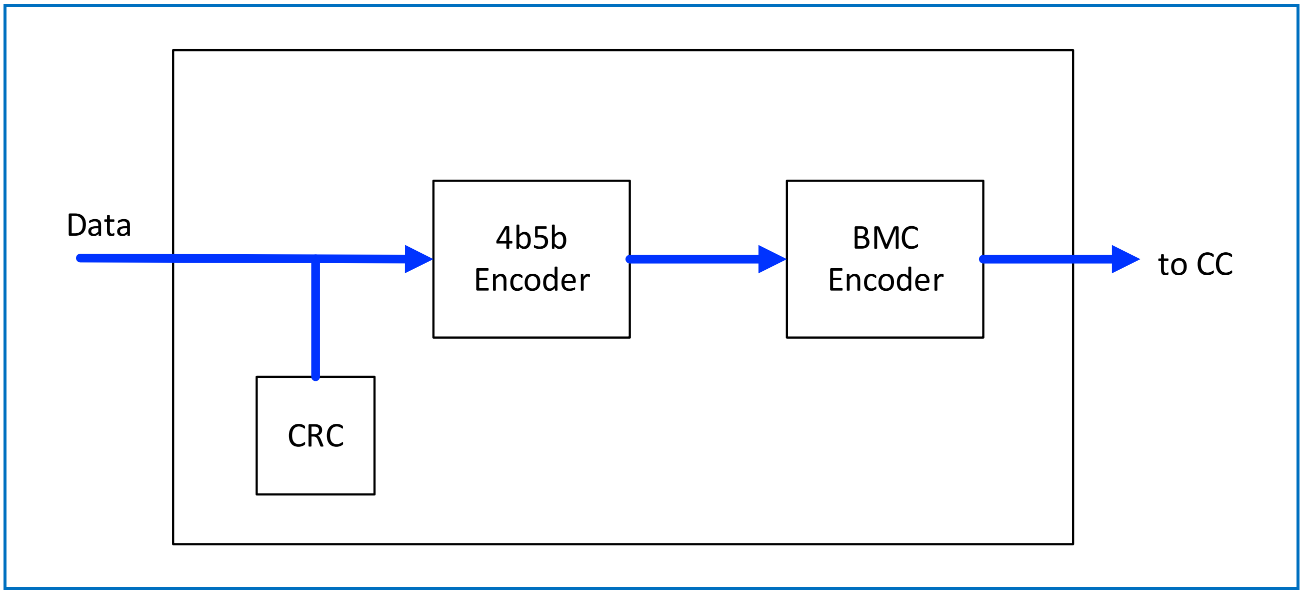
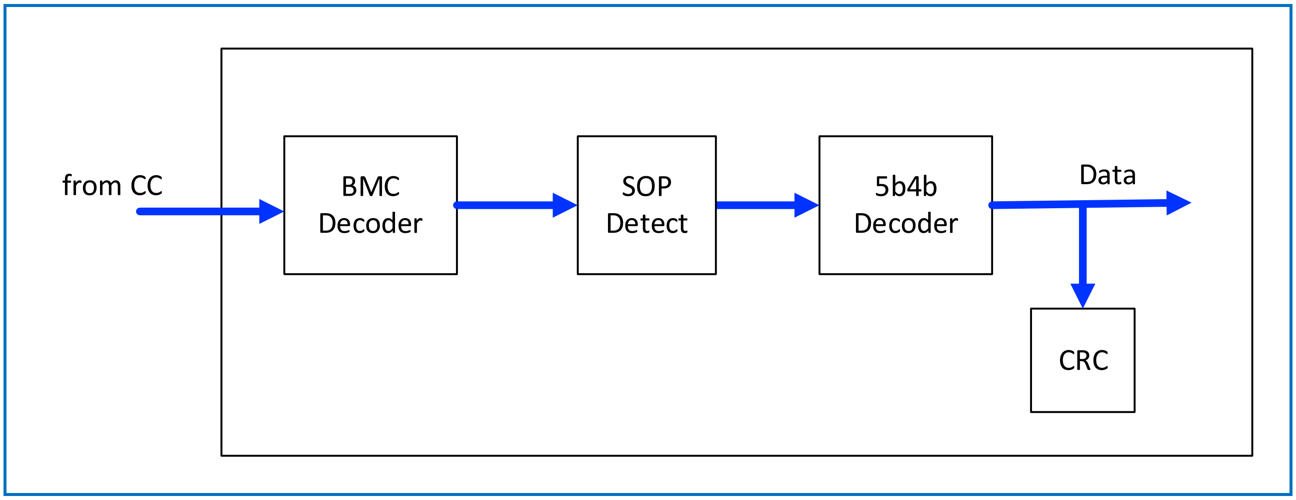


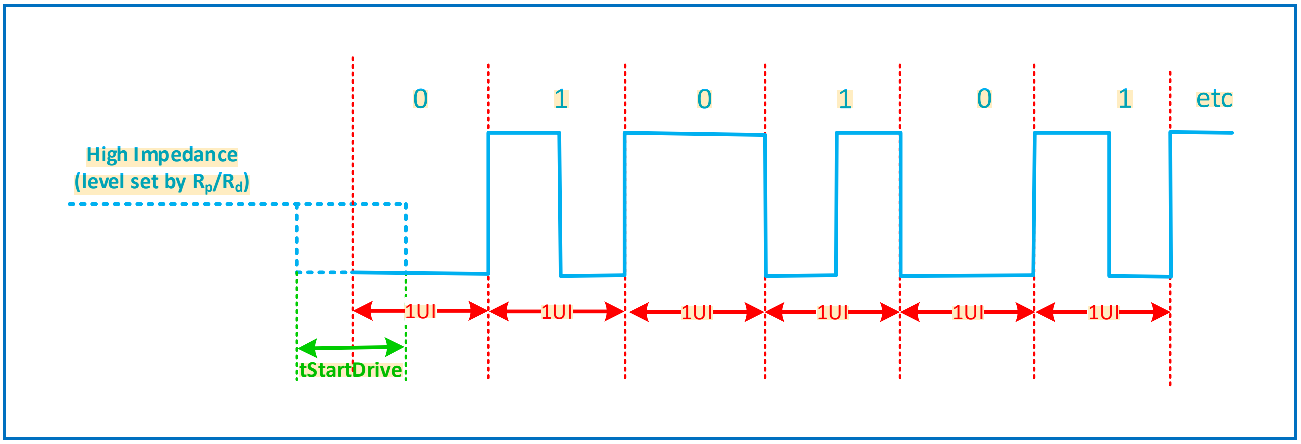
Figure 5-9 “BMC Receiver Block Diagram”



The USB PD baseband signal ***Shall*** be driven on the CC wire with a tristate driver that ***Shall*** cause a ***vSwing*** swing on CC. The tristate driver is slew rate limited (see min rise/fall time in *Section 5.8.5 “BMC Transmitter Specifications”*) to limit coupling to D+/D- and to other signal lines in the USB Type-C® fully featured cables (see *[USB Type-C 2.3]*). This slew rate limiting can be performed either with driver design or an RC filter on the driver output.

When sending the Preamble, the transmitter ***Shall*** start by transmitting a low level. The receiver ***Shall*** tolerate the loss of the first edge. The transmitter ***May*** vary the start of the Preamble by ***tStartDrive*** min (see *Figure 5-10 “BMC Encoded Start of Preamble”*).

Figure 5-10 “BMC Encoded Start of Preamble”



The transmitter ***Shall*** terminate the final bit of the Frame by an edge (the “trailing edge”) to help ensure that the receiver clocks the final bit. If the trailing edge results in the transmitter driving CC low (i.e., the final half-UI of the frame is high), then the transmitter:

- ***Shall*** continue to drive CC low for ***tHoldLowBMC***.
- Then ***Shall*** continue to drive CC low for ***tEndDriveBMC*** measured from the trailing edge of the final bit of the Frame.
- Then ***Shall*** release CC to high impedance.

Figure 5-11 "Transmitting or Receiving BMC Encoded Frame Terminated by Zero with High-to-Low Last Transition" illustrates the end of a BMC encoded Frame with an encoded zero for which the final bit of the Frame is terminated by a high to low transition. *Figure 5-12 "Transmitting or Receiving BMC Encoded Frame Terminated by One with High-to-Low Last Transition"* illustrates the end of a BMC Encoded frame with an encoded one for which the final bit of the Frame is terminated by a high to low transition. Both figures also illustrate the ***tInterFrameGap*** timing requirement before the start of the next Frame when the Port has either been transmitting or receiving the previous Frame (see *Section 5.8.5.4 "Inter-Frame Gap"*).

Figure 5-11 "Transmitting or Receiving BMC Encoded Frame Terminated by Zero with High-to-Low Last Transition"

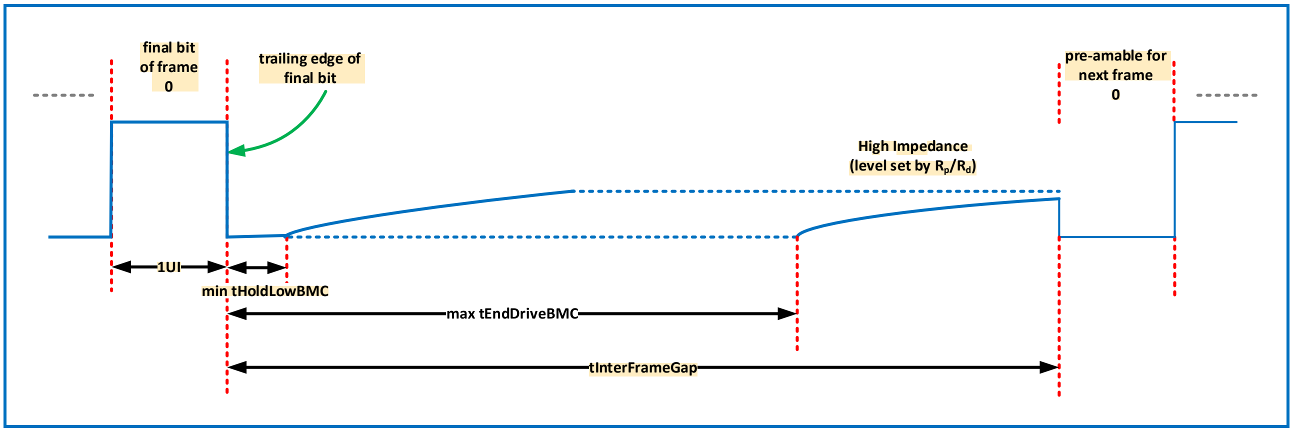
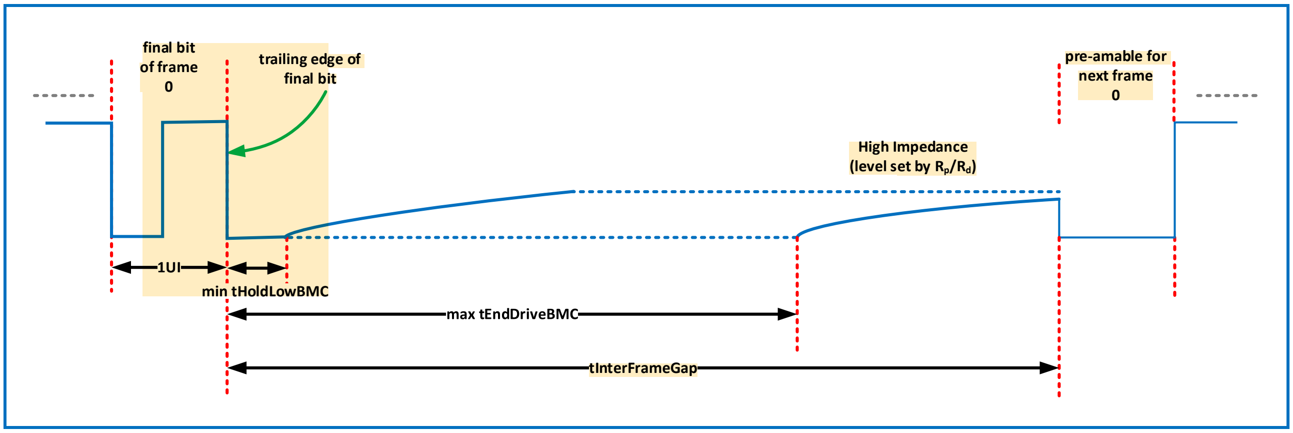

final bit of frame 0
trailing edge of final bit
pre-amable for next frame 0
High Impedance (level set by R_p/R_d)
1UI
min tHoldLowBMC
max tEndDriveBMC
tInterFrameGap

Figure 5-12 "Transmitting or Receiving BMC Encoded Frame Terminated by One with High-to-Low Last Transition"


final bit of frame 0
trailing edge of final bit
pre-amable for next frame 0
High Impedance (level set by R_p/R_d)
1UI
min tHoldLowBMC
max tEndDriveBMC
tInterFrameGap

If the trailing edge results in the transmitter driving CC high (i.e., the final half-UI of the frame is low), then the transmitter:

- ***Shall*** continue to drive CC high for 1 UI.
- Then ***Shall*** drive CC low for ***tHoldLowBMC***.
- Then ***Shall*** continue to drive CC low for ***tEndDriveBMC*** measured from the final edge of the final bit of the Frame.
- Then ***Shall*** release CC to high impedance.

Figure 5-13 "Transmitting or Receiving BMC Encoded Frame Terminated by Zero with Low to High Last Transition" illustrates the ending of a BMC encoded Frame that ends with an encoded zero for which the final bit of the Frame is terminated by a low to high transition. *Figure 5-14 "Transmitting or Receiving BMC Encoded Frame Terminated by One with Low to High Last Transition"* illustrates the ending of a BMC encoded Frame that ends with an encoded one for which the final bit of the Frame is terminated by a low to high transition. Both figures also illustrate the ***tInterFrameGap*** timing requirement before the start of the next Frame when the Port has either been transmitting or receiving the previous Frame (see *Section 5.8.5.4 "Inter-Frame Gap"*).

Figure 5-13 "Transmitting or Receiving BMC Encoded Frame Terminated by Zero with Low to High Last Transition"

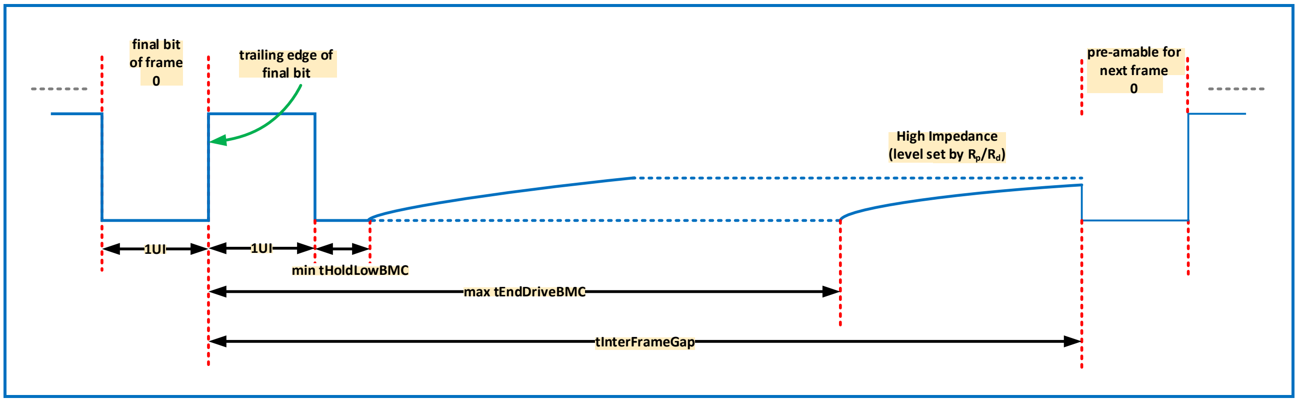
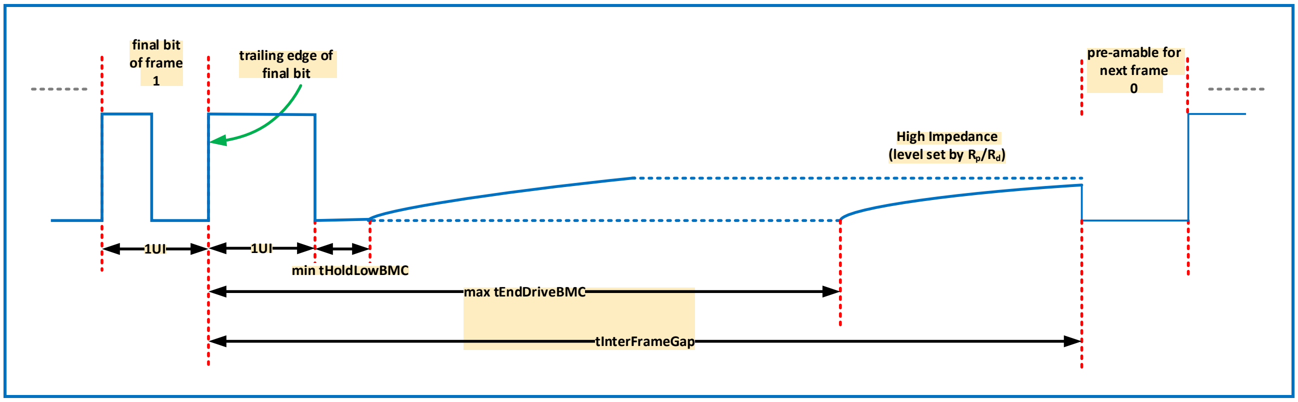


Figure 5-14 "Transmitting or Receiving BMC Encoded Frame Terminated by One with Low to High Last Transition"



Note: There is no requirement to maintain a timing phase relationship between back-to-back packets.

5.8.2 Transmit and Receive Masks

5.8.2.1 Transmit Masks

The transmitted signal ***Shall Not*** violate the masks defined in *Figure 5-15 "BMC Tx 'ONE' Mask"*, *Figure 5-16 "BMC Tx 'ZERO' Mask"*, *Table 5.14 "BMC Tx Mask Definition, X Values"* and *Table 5.15 "BMC Tx Mask Definition, Y Values"* at the output of a load equivalent to the cable model and receiver load model described in *Section 5.8.3 "Transmitter Load Model"*. The masks apply to the full range of R_p/R_d values as defined in ***[USB Type-C 2.3]***.

Note: the measurement of the transmitter does not need to accommodate a change in signal offset due to the ground offset when current is flowing in the cable.

The transmitted signal ***Shall*** have a rise time no faster than ***tRise***. The transmitted signal ***Shall*** have a fall time no faster than ***tFall***. The maximum limits on the rise and fall times are enforced by the Tx inner masks.

Figure 5-15 "BMC Tx 'ONE' Mask"

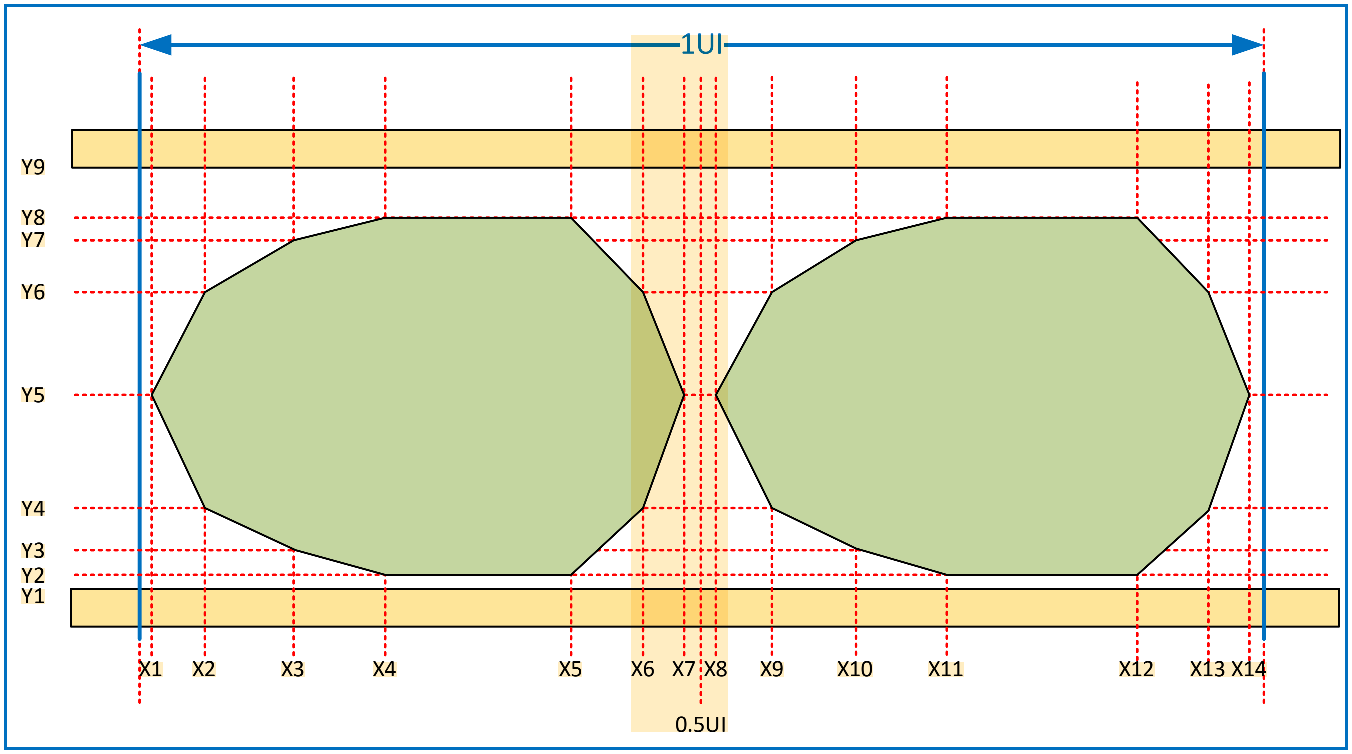


Figure 5-16 "BMC Tx 'ZERO' Mask"

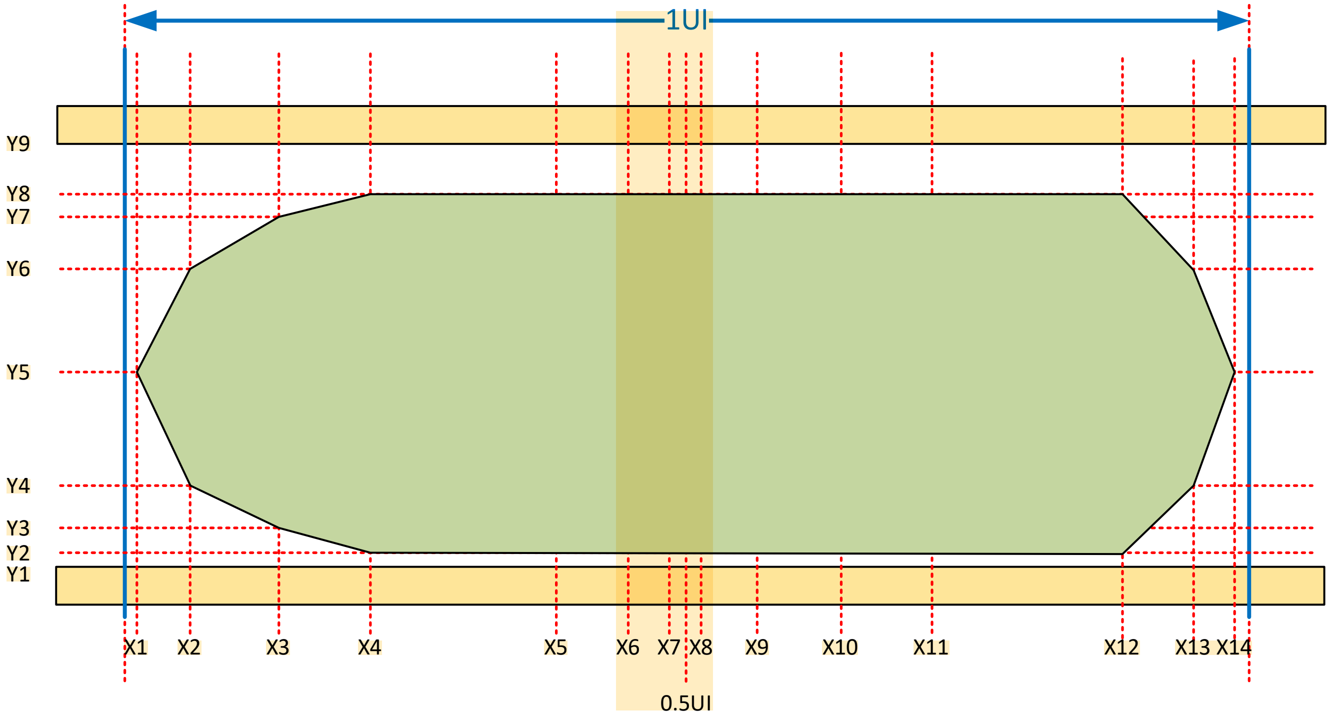


Table 5.14 "BMC Tx Mask Definition, X Values"

Name	Description	Value	Units
X1Tx	Left Edge of Mask	0.015	UI
X2Tx	see figure	0.07	UI
X3Tx	see figure	0.15	UI
X4Tx	see figure	0.25	UI
X5Tx	see figure	0.35	UI
X6Tx	see figure	0.43	UI
X7Tx	see figure	0.485	UI
X8Tx	see figure	0.515	UI
X9Tx	see figure	0.57	UI
X10Tx	see figure	0.65	UI
X11Tx	see figure	0.75	UI
X12Tx	see figure	0.85	UI
X13Tx	see figure	0.93	UI
X14Tx	Right Edge of Mask	0.985	UI

Table 5.15 “BMC Tx Mask Definition, Y Values”

Name	Description	Value	Units
Y1Tx	Lower bound of Outer mask	-0.075	V
Y2Tx	Lower bound of inner mask	0.075	V
Y3Tx	see figure	0.15	V
Y4Tx	see figure	0.325	V
Y5Tx	Inner mask vertical midpoint	0.5625	V
Y6Tx	see figure	0.8	V
Y7Tx	see figure	0.975	V
Y8Tx	see figure	1.04	V
Y9Tx	Upper Bound of Outer mask	1.2	V

5.8.2.2 Receive Masks

A Source using the BMC Signaling Scheme ***Shall*** be capable of receiving a signal that complies with the mask when sourcing power as defined in *Figure 5-17 “BMC Rx ‘ONE’ Mask when Sourcing Power”*, *Figure 5-18 “BMC Rx ‘ZERO’ Mask when Sourcing Power”* and *Table 5.16 “BMC Rx Mask Definition”*. The Source Rx mask is bounded by sweeping a Tx mask compliant signal, with added ***vNoiseActive*** between power neutral and Source offsets.

A Consumer using the BMC Signaling Scheme ***Shall*** be capable of receiving a signal that complies with the mask when sinking power as defined in *Figure 5-21 “BMC Rx ‘ONE’ Mask when Sinking Power”, Figure 5-22 “BMC Rx ‘ZERO’ Mask when Sinking Power”* and *Table 5.16 “BMC Rx Mask Definition”*. The Consumer Rx mask is bounded by sweeping a Tx mask compliant signal, with added ***vNoiseActive*** between power neutral and Consumer offsets.

Every product using the BMC Signaling Scheme ***Shall*** be capable of receiving a signal that complies with the mask when power neutral as defined in *Figure 5-19 “BMC Rx ‘ONE’ Mask when Power neutral”*, *Figure 5-20 “BMC Rx ‘ZERO’ Mask when Power neutral”* and *Table 5.16 “BMC Rx Mask Definition”*.

Dual-Role Power Devices ***Shall*** meet the receiver requirements for a Source when providing power during any transmission using the BMC Signaling Scheme or a Sink when consuming power during any transmission using the BMC Signaling Scheme.

Cable Plugs ***Shall*** meet the receiver requirements for both a Source and a Sink during any transmission using the BMC Signaling Scheme.

The parameters used in the masks are specified to be appropriate to either edge triggered or oversampling receiver implementations.

The masks are defined for ‘ONE’ and ‘ZERO’ separately as BMC enforces a transition at the midpoint of the unit interval while a ‘ONE’ is transmitted.

The Rx masks are defined to bound the Rx noise after the Rx bandwidth limiting filter with the time constant ***tRxFilter*** has been applied.

The boundaries of Rx outer mask, ***Y1Rx*** and ***Y5Rx***, are specified according to ***vSwing*** max and accommodate half of ***vNoiseActive*** from cable noise coupling and the signal offset ***vIRDropGNDC*** due to the ground offset when current is flowing in the cable.

The vertical dimension of the Rx inner mask, ***Y4Rx*** - ***Y2Rx***, for power neutral is derived by reducing the vertical dimension of the Tx inner mask, ***Y7Tx*** - ***Y3Tx***, at time location ***X3Tx*** by ***vNoiseActive*** to account for cable noise coupling. The received signal is composed of a waveform compliant to the Tx mask plus ***vNoiseActive***.

The vertical dimension of the Rx inner mask for sourcing power is derived by reducing the vertical dimension of the Tx inner mask by ***vNoiseActive*** and ***vIRDropGNDC*** to account for both cable noise coupling and signal DC offset. The received signal is composed of a waveform compliant to the Tx mask plus the maximum value of ***vNoiseActive*** plus ***vIRDropGNDC*** where the ***vIRDropGNDC*** value transitions between the minimum and the maximum values as allowed in this spec.

The vertical dimension of the Rx inner mask for sinking power is derived by reducing the vertical dimension of the Tx inner mask by ***vNoiseActive*** max and ***vIRDropGNDC*** max for account for both cable noise coupling and signal DC offset. The received signal is composed of a waveform compliant to the Tx mask plus the maximum value of ***vNoiseActive*** plus ***vIRDropGNDC*** where the ***vIRDropGNDC*** value transitions between the minimum and the maximum values as allowed in this spec.

The center line of the Rx inner mask, ***Y3Rx***, is at half of the nominal ***vSwing*** for power neutral, and is shifted up by half of ***vIRDropGNDC*** max for sourcing power and is shifted down by half of ***vIRDropGNDC*** max for sinking power.

The receiver sensitivity ***Shall*** be set such that the receiver does not treat noise on an undriven signal path as an incoming signal. Signal amplitudes below ***vNoiseIdle*** max ***Shall*** be treated as noise when BMC is idle.

Figure 5-17 "BMC Rx 'ONE' Mask when Sourcing Power"

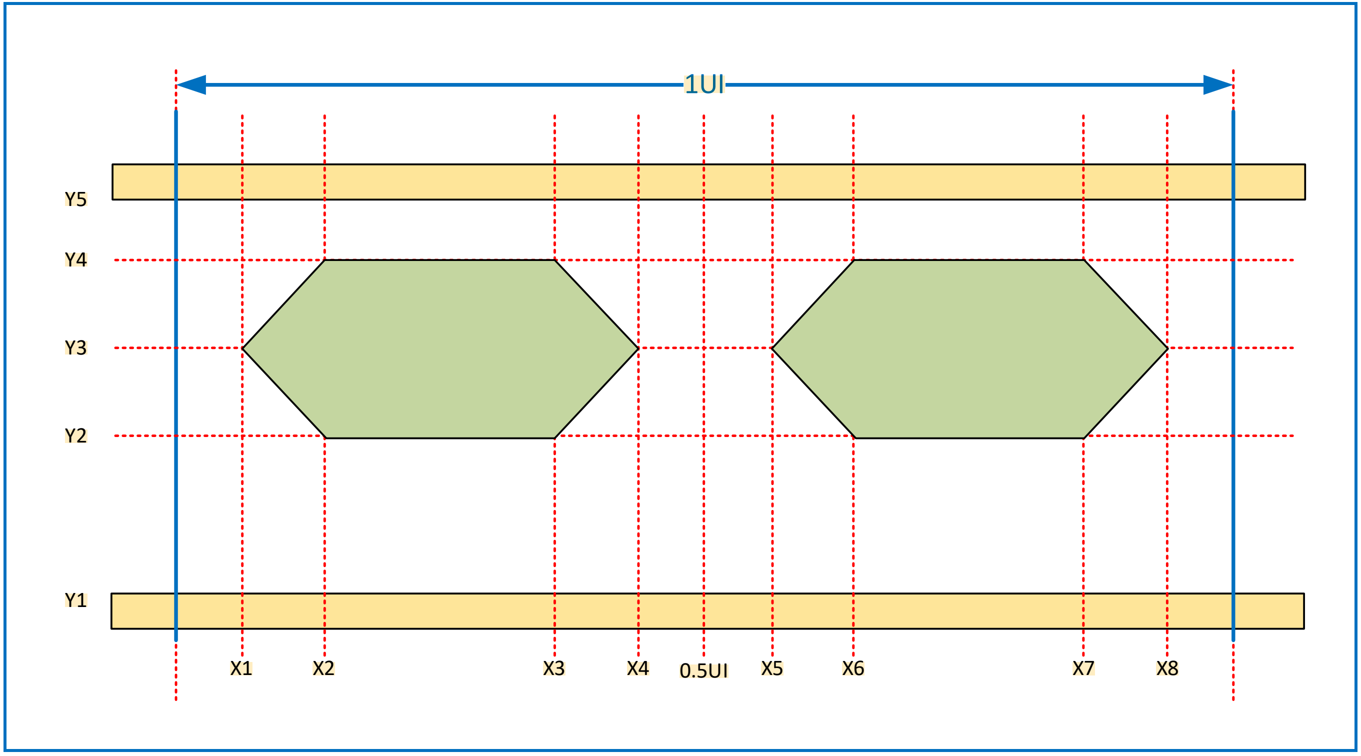


Figure 5-18 “BMC Rx ‘ZERO’ Mask when Sourcing Power”

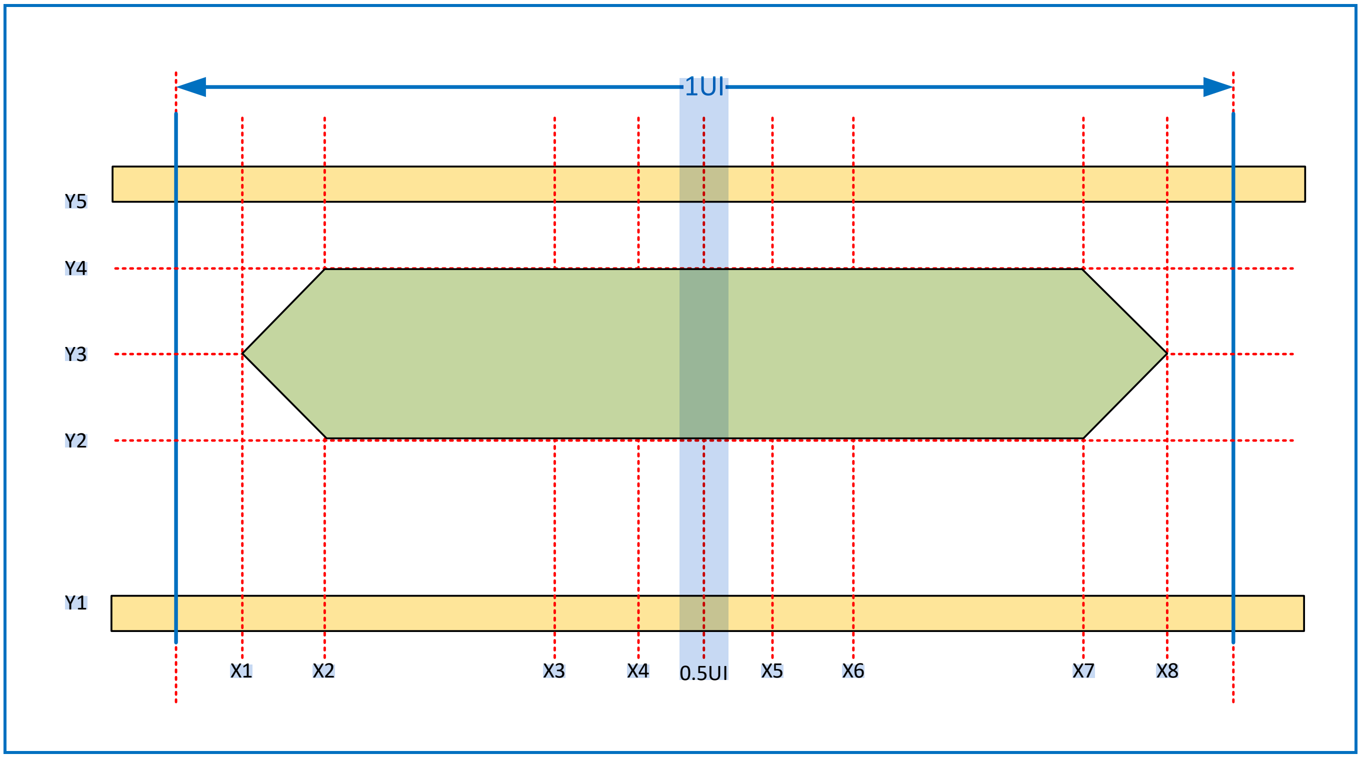


Figure 5-19 “BMC Rx ‘ONE’ Mask when Power neutral”

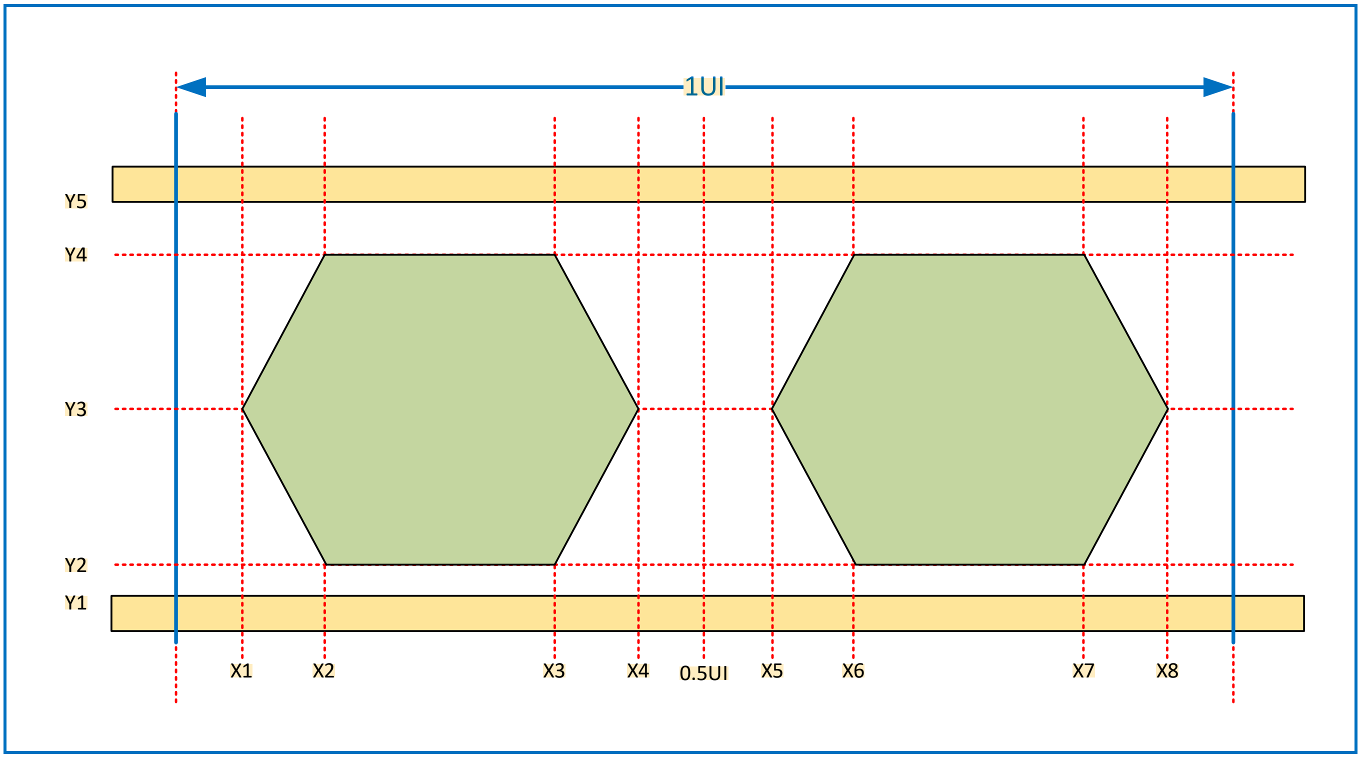


Figure 5-20 “BMC Rx ‘ZERO’ Mask when Power neutral”

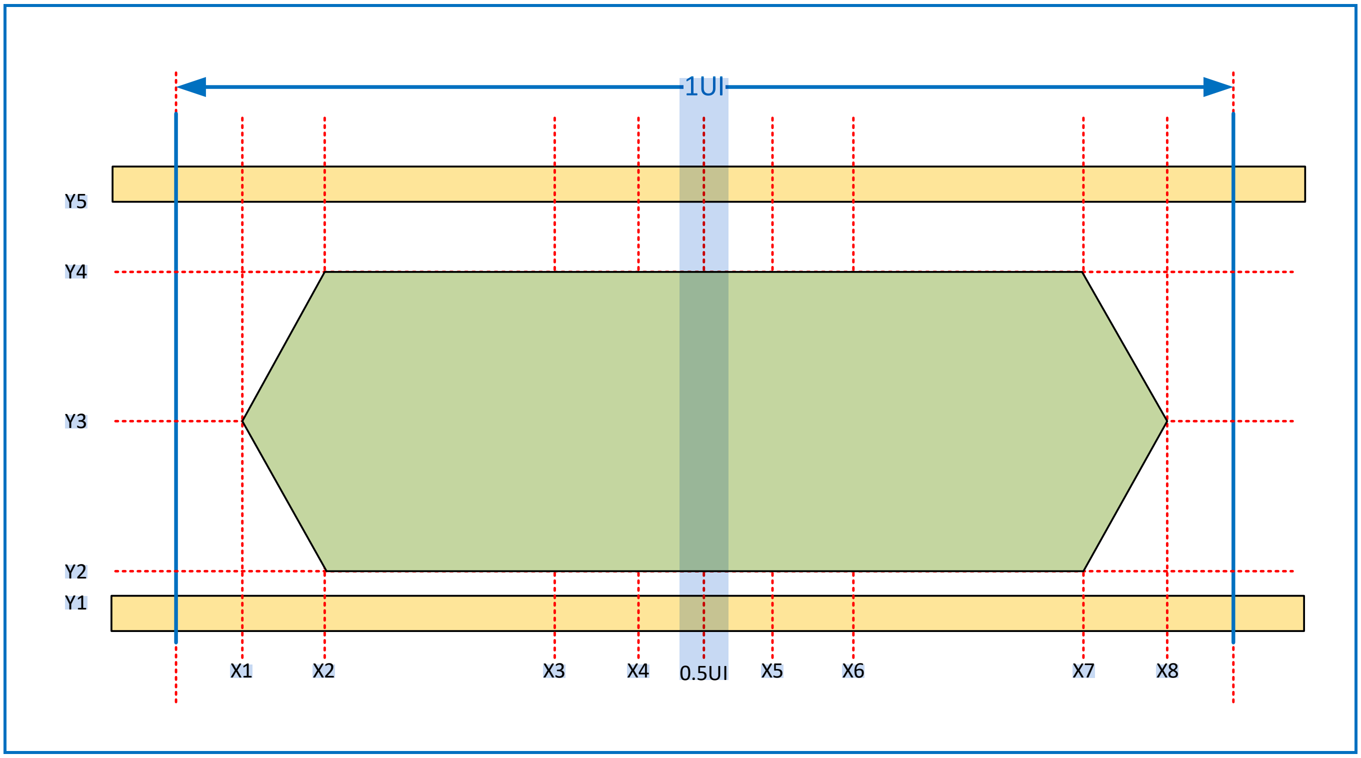


Figure 5-21 “BMC Rx ‘ONE’ Mask when Sinking Power”

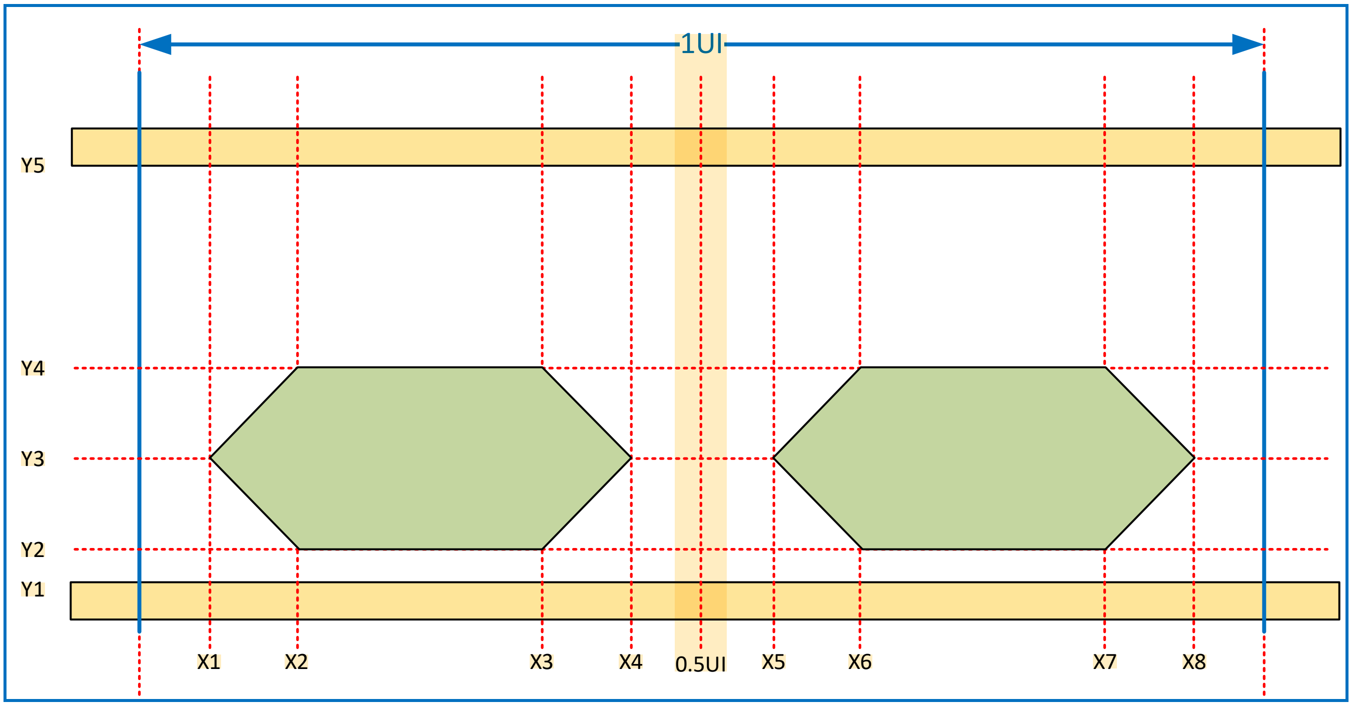


Figure 5-22 “BMC Rx ‘ZERO’ Mask when Sinking Power”

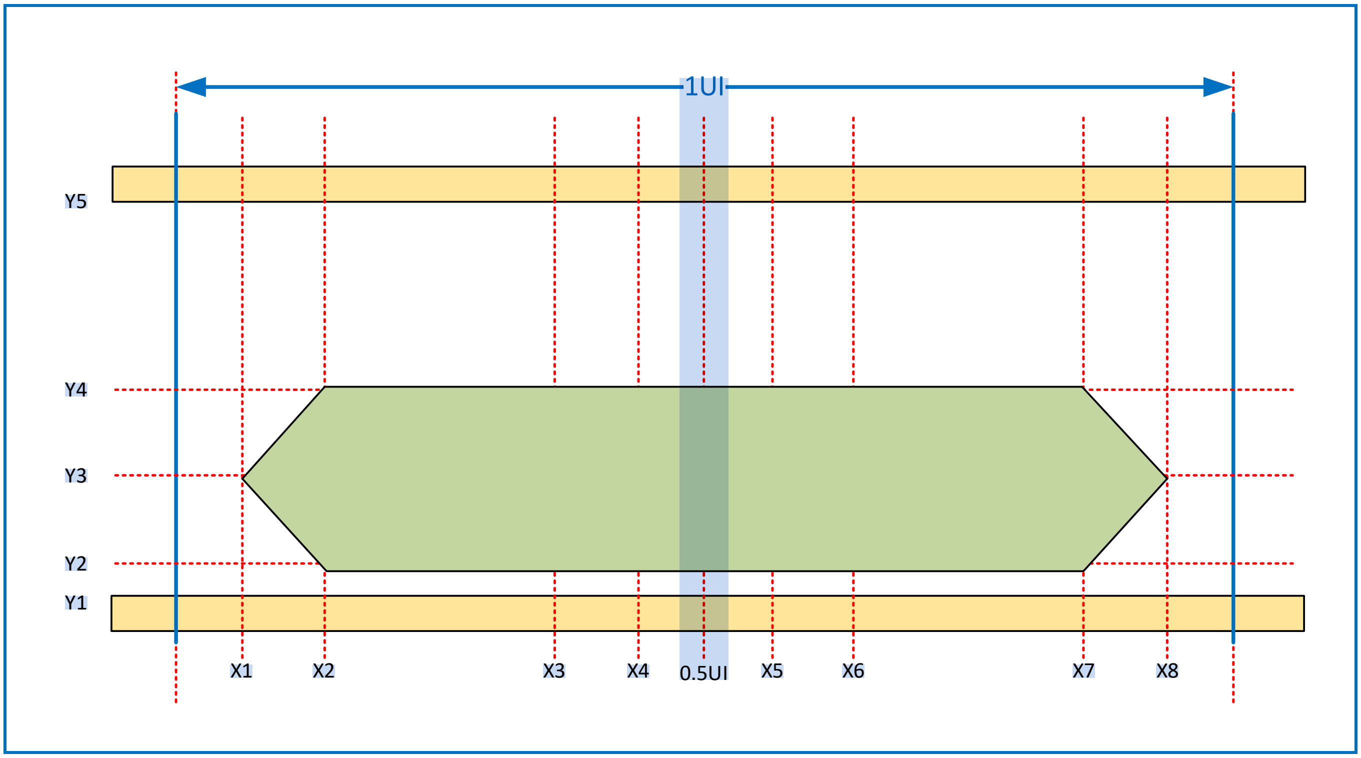


Table 5.16 "BMC Rx Mask Definition"

Name	Description	Value	Units
X1Rx	Left Edge of Mask	0.07	UI
X2Rx	Top Edge of Mask	0.15	UI
X3Rx	See figure	0.35	UI
X4Rx	See figure	0.43	UI
X5Rx	See figure	0.57	UI
X6Rx	See figure	0.65	UI
X7Rx	See figure	0.85	UI
X8Rx	See figure	0.93	UI
Y1Rx	Lower bound of Outer Mask	-0.3325	V
Y2Rx	Lower Bound of Inner Mask	*Y3Rx* – 0.205 when sourcing power[1] or sinking power[1]. *Y3Rx* – 0.33 when power neutral1	V
Y3Rx	Center line of Inner Mask	0.6875 Sourcing Power[1] 0.5625 Power Neutral[1] 0.4375 Sinking Power[1]	V
Y4Rx	Upper bound of Inner mask	*Y3Rx* + 0.205 when sourcing power[1] or sinking power[1]. *Y3Rx* + 0.33 when power neutral[1]	V
Y5Rx	Upper bound of the Outer mask	1.5325	V

[1] The position of the center line of the Inner Mask is dependent on whether the receiver is Sourcing or Sinking power or is Power Neutral (see earlier in this section).

5.8.3 Transmitter Load Model

The transmitter load model ***Shall*** be equivalent to the circuit outlined in *Figure 5-23 "Transmitter Load Model for BMC Tx from a Source"* for a Source and *Figure 5-24 "Transmitter Load Model for BMC Tx from a Sink"* for a Sink. It is formed by the concatenation of a cable load model and a receiver load model. See ***[USB Type-C 2.3]*** for details of the R_p and R_d resistors. Note the parameters *zCable_CC*, *tCableDelay_CC* and *cCablePlug_CC* are defined in ***[USB Type-C 2.3]***.

Figure 5-23 "Transmitter Load Model for BMC Tx from a Source"

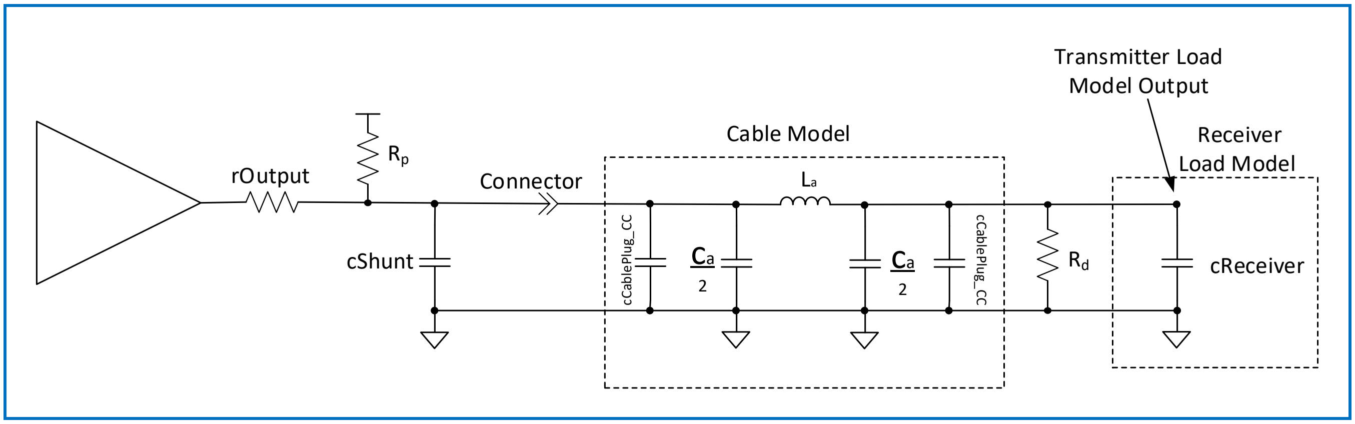
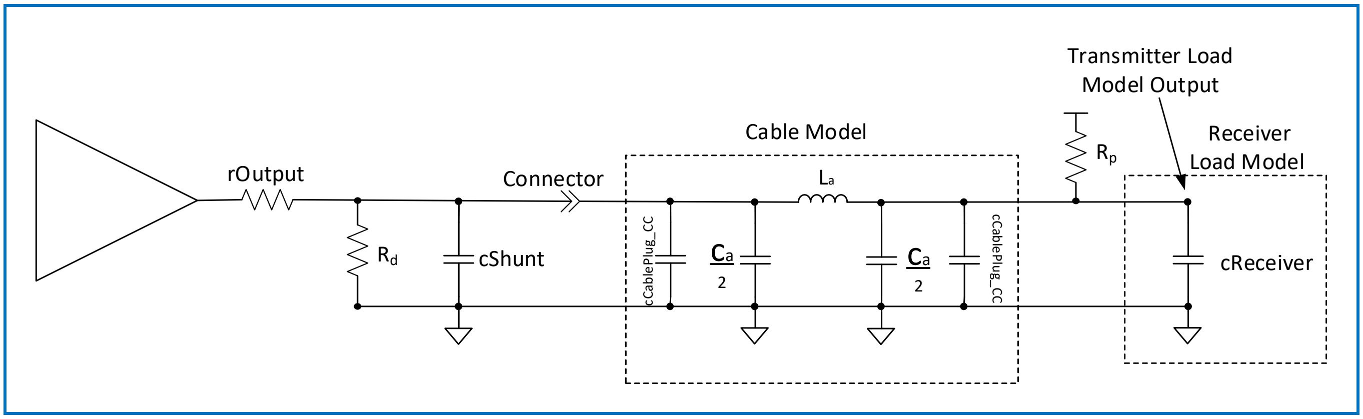


Figure 5-24 "Transmitter Load Model for BMC Tx from a Sink"



The transmitter system components *rOutput* and *cShunt* are illustrated for ***Informative*** purposes, and do not form part of the transmitter load model. See *Section 5.8.5 "BMC Transmitter Specifications"* for a description of the transmitter system design.

The value of the modeled cable inductance, La, (in nH) ***Shall*** be calculated from the following formula:

$$La = \ tCableDelay\_CC_{max} * \ zCable\_CC_{min}$$

tCableDelay_CC is the modeled signal propagation delay through the cable, and *zCable_CC* is the modeled cable impedance.

The modeled cable inductance is 640 nH for a cable with $zCable\_CC_{min}$ = 32 Ω and $tCableDelay\_CC_{max}$ = 20 nS.

The value of the modeled cable capacitance, Ca, (in pF) ***Shall*** be calculated from the following formula:

$$Ca = \frac{tCableDelay\_CC_{max}}{zCable\_CC_{min}}$$

The modeled cable capacitance is Ca = 625 pF for a cable with *zCable_CC*$_{\text{min}}$ = 32 Ω and *tCableDelay_CC*$_{\text{max}}$ = 20 nS. Therefore, Ca/2 = 312.5 pF.

cCablePlug_CC models the capacitance of the plug at each end of the cable. ***cReceiver*** models the capacitance of the receiver. The maximum values ***Shall*** be used in each case.

Note: the transmitter load model assumes that there are no other return currents on the ground path.

5.8.4 BMC Common specifications

This section defines the common receiver and transmitter requirements.

5.8.4.1 BMC Common Parameters

The electrical requirements specified in *Table 5.17 "BMC Common Normative Requirements"* ***Shall*** apply to both the transmitter and receiver.

Table 5.17 "BMC Common Normative Requirements"

Name	Description	Min	Nom	Max	Units	Comment
fBitRate	Bit rate	270	300	330	Kbps	
tUnitInterval[1]	Unit Interval	3.03		3.70	µs	1/***fBitRate***

[1] ***tUnitInterval*** denotes the time to transmit an unencoded data bit, not the shortest high or low times on the wire after encoding with BMC. A single data bit cell has duration of 1UI, but a data bit cell with value 1 will contain a centrally placed 01 or 10 transition in addition to the transition at the start of the cell.

5.8.5 BMC Transmitter Specifications

The transmitter ***Shall*** meet the specifications defined in *Table 5.18 "BMC Transmitter Normative Requirements"*.

Table 5.18 “BMC Transmitter Normative Requirements”

Name	Description	Min	Nom	Max	Units	Comment
pBitRate	Maximum difference between the bitrate during the part of the packet following the Preamble and the reference bitrate.			0.25	%	The reference bit rate is the average bit rate of the last 32 bits of the Preamble.
rFRSwapTx	Fast Role Swap Request transmit driver resistance (excluding cable resistance)			5	Ω	Maximum driver resistance of a Fast Role Swap Request transmitter. Assumes a worst case cable resistance of 15Ω as defined in ***[USB Type-C 2.3]***. Note: based on this value the maximum combined driver and cable resistance of a Fast Role Swap Request transmitter is 20Ω.
tEndDriveBMC	Time to cease driving the line after the end of the last bit of the Frame.			23	µs	Min value is limited by ***tHoldLowBMC***.
tFall	Fall Time	300			ns	10 % and 90 % amplitude points, minimum is under an unloaded condition.
tHoldLowBMC	Time to cease driving the line after the final high-to-low transition.	1			µs	Max value is limited by ***tEndDriveBMC***.
tInterFrameGap	Time from the end of last bit of a Frame until the start of the first bit of the next Preamble.	25			µs	
tFRSwapTx	Fast Role Swap Request transmit duration	60		120	µs	Fast Role Swap Request is indicated from the initial Source to the initial Sink by driving CC low for this time.
tRise	Rise time	300			ns	10 % and 90 % amplitude points, minimum is under an unloaded condition.
tStartDrive	Time before the start of the first bit of the Preamble when the transmitter ***Shall*** start driving the line.	-1		1	µs	
vSwing	Voltage Swing	1.05	1.125	1.2	V	Applies to both no load condition and under the load condition specified in Section 5.8.3.
zDriver	Transmitter output impedance	33		75	Ω	Source output impedance at the Nyquist frequency of ***[USB 2.0]*** low speed (750 kHz) while the source is driving the CC line.

5.8.5.1 Capacitance when not transmitting

cReceiver is the capacitance that a DFP or UFP ***Shall*** present on the CC line when the DFP or UFP's receiver is not transmitting on the line. The transmitter ***May*** have more capacitance than ***cReceiver*** while driving the CC line, but ***Shall*** meet the waveform mask requirements. Once transmission is complete, the transmitter ***Shall*** disengage capacitance in excess of ***cReceiver*** from the CC wire within ***tInterFrameGap***.

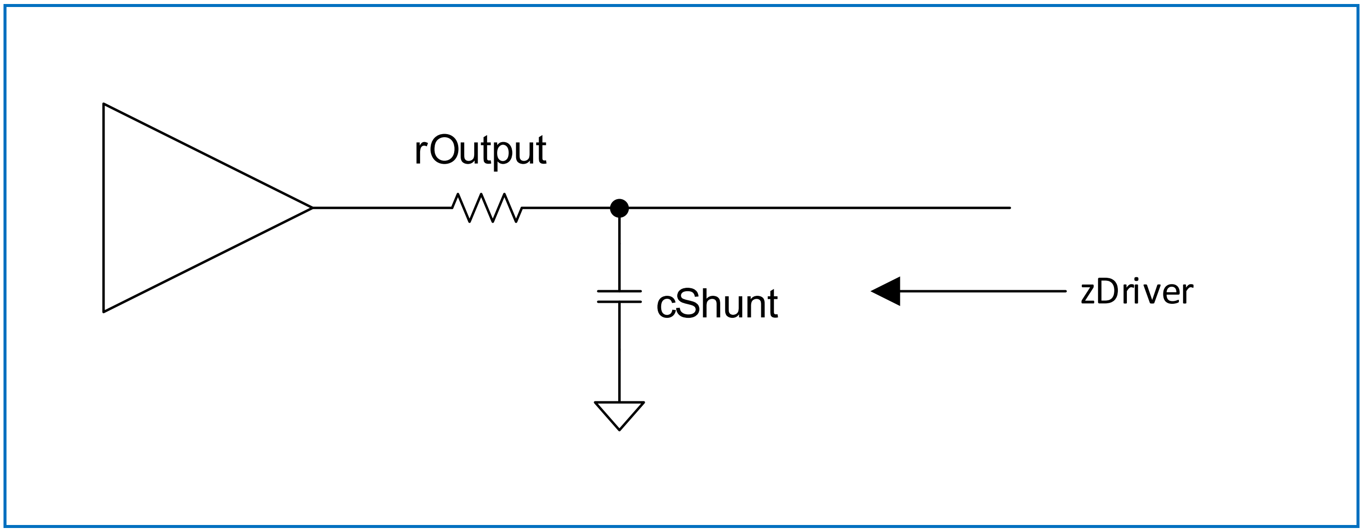
5.8.5.2 Source Output Impedance

Source output impedance ***zDriver*** is determined by the driver resistance and the shunt capacitance of the source and is hence a frequency dependent term. ***zDriver*** impacts the noise ingression in the cable. It is specified such that the noise at the Receiver is bounded.

zDriver is defined by the following equation:

$$zDriver = \frac{rOutput}{1 + s * rOutput * cShunt}$$

Figure 5-25 Transmitter diagram illustrating zDriver



cShunt ***Shall Not*** cause a violation of ***cReceiver*** when not transmitting.

5.8.5.3 Bit Rate Drift

Limits on the drift in ***fBitRate*** are set to help low-complexity receiver implementations.

fBitRate is the reciprocal of the average bit duration from the previous 32 bits at a given portion of the packet. The change in ***fBitRate*** during a packet ***Shall*** be less than ***pBitRate***. The reference bit rate (refBitRate) is the average ***fBitRate*** over the last 32 bits of the Preamble. ***fBitRate*** throughout the packet, including the ***EOP***, ***Shall*** be within ***pBitRate*** of refBitRate. ***pBitRate*** is expressed as a percentage:

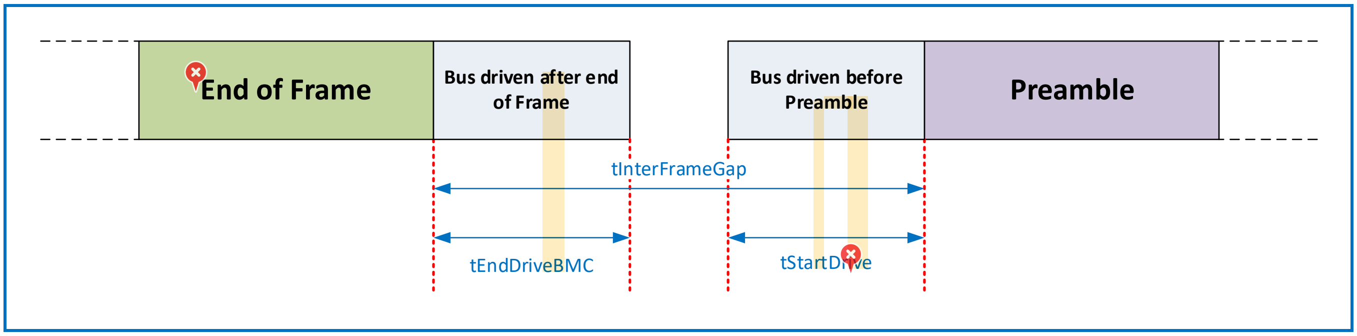
pBitRate = | ***fBitRate*** – refBitRate | / refBitRate x 100%

The transmitter ***Shall*** have the same ***pBitRate*** for all packet types. The ***BIST Carrier Mode*** and Bit Stream signals are continuous signals without a payload. When checking ***pBitRate*** any set of 1044 bits (20 bit ***SOP*** followed by 1024 PRBS bits) within a continuous signal ***May*** be considered as the part of the packet following the Preamble and the 32 preceding bits considered to be the last 32 bits of the Preamble used to compute refBitRate.

5.8.5.4 Inter-Frame Gap

Figure 5-26 "Inter-Frame Gap Timings" illustrates the inter-Frame gap timings.

Figure 5-26 "Inter-Frame Gap Timings"



The transmitter ***Shall*** drive the bus for no longer than ***tEndDriveBMC*** after transmitting the final bit of the Frame.

Before starting to transmit the next Frame's Preamble the transmitter of the next Frame ***Shall*** ensure that it waits for ***tInterFrameGap*** after either:

- Transmitting the previous frame, for example sending the next Message in an AMS immediately after having sent a ***GoodCRC*** Message, or
- Receiving the previous frame, for example when responding to a received Message with a ***GoodCRC*** Message, or
- Observing an idle condition on CC (see *Section 5.7 "Collision Avoidance"*). In this case the Port is waiting to initiate an AMS observes idle (see *Section 5.8.6.1 "Definition of Idle"*) and then waits ***tInterFrameGap*** before transmitting the Frame. See also *Section 5.7 "Collision Avoidance"* for details on when an AMS can be initiated.

Note: the transmitter is also required to verify a bus idle condition immediately prior to starting transmission of the next Frame (see *Section 5.8.6.1 "Definition of Idle"*).

The transmitter of the next Frame ***May*** vary the start of the Preamble by ***tStartDrive*** (see *Section 5.8.1 "Encoding and signaling"*).

See also *Section 5.8.1 "Encoding and signaling"* for figures detailing the timings relating to transmitting, receiving, and observing idle in relating to Frames.

5.8.5.5 Shorting of Transmitter Output

A Transmitter in a Port or Cable Plug ***Shall*** tolerate having its output be shorted to ground for ***tFRSwapTx*** max. This is due to the potential for Fast Role Swap to be signaled while the Transmitter is in the process of transmitting (see *Section 5.8.5.6 "Fast Role Swap Transmission"*).

5.8.5.6 Fast Role Swap Transmission

The Fast Role Swap process is intended for use by a PDUSB HUB that presently has an external wall supply and is providing power both through its downstream Ports to USB Devices and upstream to a USB Host such as a notebook. On removal of the external wall supply Fast Role Swap enables a V_{BUS} supply to be maintained by allowing the USB Host to apply ***vSafe5V*** when it sees V_{BUS} droop below ***vSafe5V*** after having detected Fast Role Swap signaling. The Fast Role Swap AMS is then used to correctly assign Source/Sink roles and configure the R_p/R_d resistors (see *Section 8.3.2.9 "Fast Role Swap"*).

The initial Source ***Shall*** signal a Fast Role Swap Request by driving CC to ground with a resistance of less than ***rFRSwapTx*** for ***tFRSwapTx***. The initial Source ***Shall*** only signal a Fast Role Swap when it has an Explicit Contract. The initial Source ***May*** signal a Fast Role Swap even if it has not yet had its Sink Capabilities queried by the initial Sink. On transmission of the Fast Role Swap signal any pending Messages ***Shall*** be ***Discarded*** (see ***Section 6.12.2.2.1 "Common Protocol Layer Message Transmission State Diagram"***).

The Fast Role Swap signal ***May*** override any active transmissions.

Since the initial Sink's response to the Fast Role Swap signal is to send an ***FR_Swap*** Message, the initial Source ***Shall*** ensure R_p is set to ***SinkTxOk*** once the Fast Role Swap signal is complete.

5.8.6 BMC Receiver Specifications

The receiver ***Shall*** meet the specifications defined in *Table 5.19 “BMC Receiver Normative Requirements”*.

Table 5.19 “BMC Receiver Normative Requirements”

Name	Description	Min	Nom	Max	Units	Comment
cReceiver	CC receiver capacitance	200		600	pF	The DFP or UFP system ***Shall*** have capacitance within this range when not transmitting on the line.
nBER	Bit error rate, S/N = 25 dB			10-6		
nTransitionCount	Transitions for signal detect	3				Number of transitions to be detected to declare bus non-idle.
tFRSwapRx	Fast Role Swap Request detection time	30		50	µs	A Fast Role Swap Request results in the receiver detecting a signal low for at least this amount of time.
tRxFilter	Rx bandwidth limiting filter (digital or analog)	100			ns	Time constant of a single pole filter to limit broad-band noise ingression1.
tTransitionWindow	Time window for detecting non-idle	12		20	µs	
vFRSwapCableTx	Fast Role Swap Request Voltage detection threshold	490	520	550	mV	The Fast Role Swap Request must be below this Voltage threshold to be detected.
vIRDropGNDC	Cable Ground IR Drop			250	mV	As specified in *[USB Type-C 2.3]*.
vNoiseActive	Noise amplitude when BMC is active.			165	mV	Peak-to-peak noise from VBUS, *[USB 2.0]* and SBU lines after the Rx bandwidth limiting filter with the time constant **tRxFilter** has been applied.
vNoiseIdle	Noise amplitude when BMC is idle.			300	mV	Peak-to-peak noise from VBUS, *[USB 2.0]* and SBU lines after the Rx bandwidth limiting filter with the time constant **tRxFilter** has been applied.
zBmcRx	Receiver Input Impedance	1			MΩ	

1) Broad-band noise ingression is due to coupling in the cable interconnect.

5.8.6.1 Definition of Idle

BMC packet collision is avoided by the detection of signal transitions at the receiver. Detection is active when *nTransitionCount* transitions occur at the receiver within a time window of *tTransitionWindow*. After waiting *tTransitionWindow* without detecting *nTransitionCount* transitions the bus ***Shall*** be declared idle.

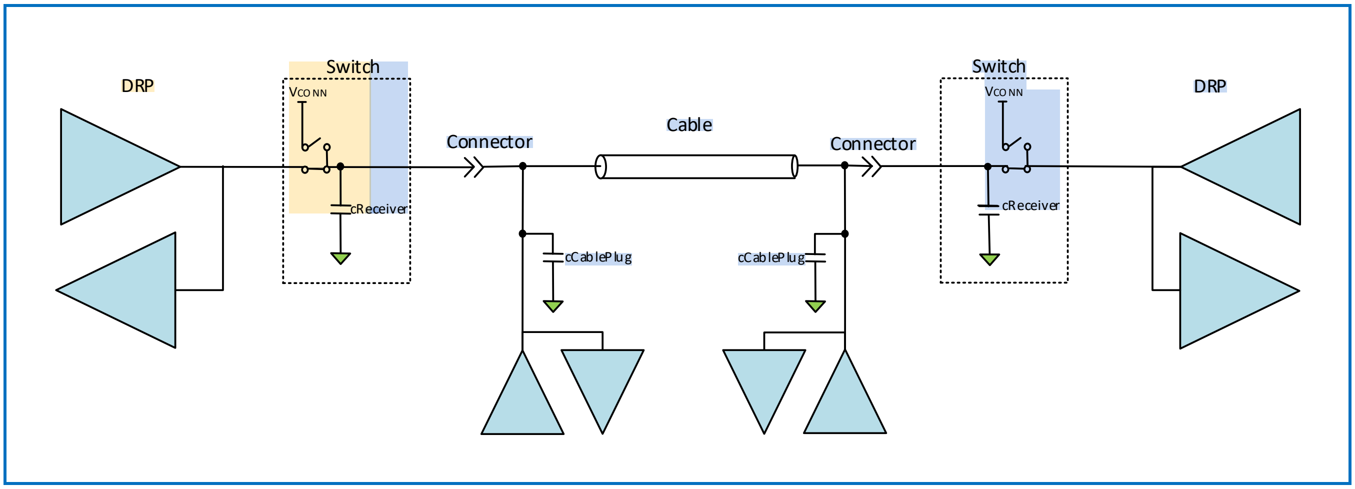
Refer to *Section 5.8.5.4 “Inter-Frame Gap”* for details of when transmissions ***May*** start.

5.8.6.2 Multi-Drop

The BMC Signaling Scheme is suitable for use in multi-Drop configurations containing one or two BMC Multi-Drop transceivers connected to the CC wire, for example where one or both ends of a cable contains a multi-Drop transceiver. In this specification the location of the multi-Drop transceiver is referred to as the Cable Plug.

Figure 5-27 "Example Multi-Drop Configuration showing two DRPs" below illustrates a typical Multi Drop configuration with two DRPs.

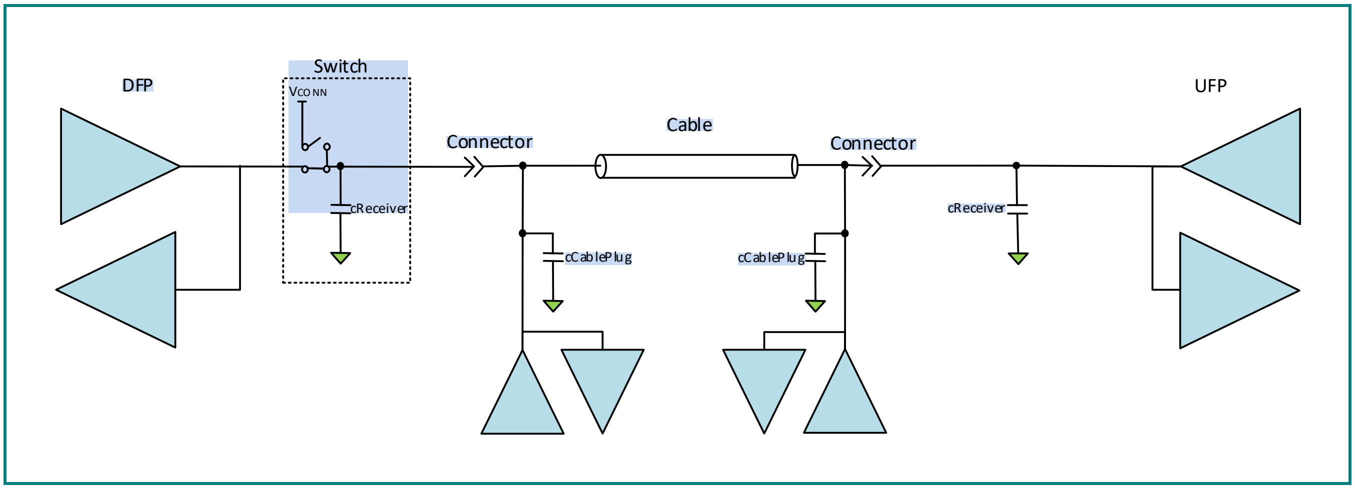
Figure 5-27 "Example Multi-Drop Configuration showing two DRPs"



The Multi-Drop transceiver ***Shall*** obey all the electrical characteristics specified in this section except for those relating to capacitance. The maximum capacitance allowed for the multi-Drop node when not driving the line is *cCablePlug_CC* defined in ***[USB Type-C 2.3]***. There are no constraints as to the distance of the multi-Drop transceiver from the end of the plug. The Multi-Drop transceiver(s) ***May*** be located anywhere along the cable including the plugs. The Multi-Drop transceiver suffers less from ground offset compared to the transceivers in the host or device and contributes no significant reflections.

It is possible to have a configuration at Attach where one Port can be a VCONN Source and the other Port is not able to be a VCONN Source, such that there is no switch in the second Port. An example of a DFP with a switch Attached to a UFP without a switch is outlined in *Figure 5-28 "Example Multi-Drop Configuration showing a DFP and UFP"*. The capacitance on the CC line for a Port not able to be a VCONN Source ***Shall*** still be within ***cReceiver*** except when transmitting.

Figure 5-28 "Example Multi-Drop Configuration showing a DFP and UFP"



5.8.6.3 Fast Role Swap Detection

An initial Sink prepares for a Fast Role Swap by ensuring that once it has detected the Fast Role Swap signal its power supply is ready to respond by applying ***vSafe5V*** according to the timing detailed in *Section 7.1.13 "Fast Role Swap"*. The initial Sink ***Shall*** only respond to the Fast Role Swap signal when all the following conditions have been met:

- An Explicit Contract has been established and the Sink Capabilities of the initial Source have been received by, and at the request of, the initial Sink.
- The ***Sink_Capabilities*** Message received from the initial Source has at least one of the Fast Role Swap bits set in its 5V fixed PDO.
- The initial Sink is able and willing to source the current requested by the initial Source in the Fast Role Swap bits of its ***Sink_Capabilities*** Message.

On detection of the Fast Role Swap signal any pending Messages ***Shall*** be ***Discarded*** (see *Section 6.12.2.2.1 "Common Protocol Layer Message Transmission State Diagram"*).

When the initial Sink is prepared for a Fast Role Swap and the bus is idle the CC Voltage averaged over ***tFRSwapRx*** min remains above 0.7V (see ***[USB Type-C 2.3]***) since the Source R_p is either 1.5A or 3.0A. However, ***vNoiseIdle*** noise ***May*** cause the CC line Voltage to reach 0.7V-***vNoiseIdle***/2 for short durations. When the initial Sink is prepared for a Fast Role swap while it is transmitting and the initial Source is signaling a Fast Role Swap Request, the transmission will be attenuated such that the peak CC Voltage will not exceed ***vFRSwapCableTx*** min. Therefore, when the initial Sink is prepared for a Fast Role Swap, it ***Shall Not*** detect a Fast Swap signal when the CC Voltage, averaged over ***tFRSwapRx*** min, is above 0.7V. When the initial Sink is prepared for a Fast Role Swap, it ***Shall*** detect a CC Voltage lower than ***vFRSwapCableTx*** min for ***tFRSwapRx*** as a Fast Role Swap Request.

Note: the initial Sink is not required to average the CC Voltage to meet these requirements.

The initial Sink ***Shall*** initiate the Fast Role Swap AMS within ***tFRSwapInit*** of detecting the Fast Role Swap Request in order to assign the R_p/R_d resistors to the correct Ports and to re-synchronize the state machines (see *Section 6.3.19 "FR_Swap Message"*).

The initial Sink ***Shall*** become the new Source and ***Shall*** start supplying ***vSafe5V*** at USB Type-C® Current (see ***[USB Type-C 2.3]***) no later than ***tSrcFRSwap*** after V$_{BUS}$ has dropped below ***vSafe5V***. An initial Sink ***Shall*** disable its V$_{BUS}$ Disconnect Threshold detection circuitry while Fast Role Swap detection is active.

Note: while power is transitioning the VCONN Source to the Cable Plug(s) cannot be guaranteed.

5.9 Built in Self-Test (BIST)

The following sections define BIST functionality which ***Shall*** be supported.

5.9.1 BIST Carrier Mode

In ***BIST Carrier Mode***, the Physical Layer ***Shall*** send out a BMC encoded continuous string of alternating "1"s and “0” s. This enables the measurement of power supply noise and frequency drift.

Note that this transmission is a purely a sequence of alternating bits and ***Shall Not*** be formatted as a Packet.

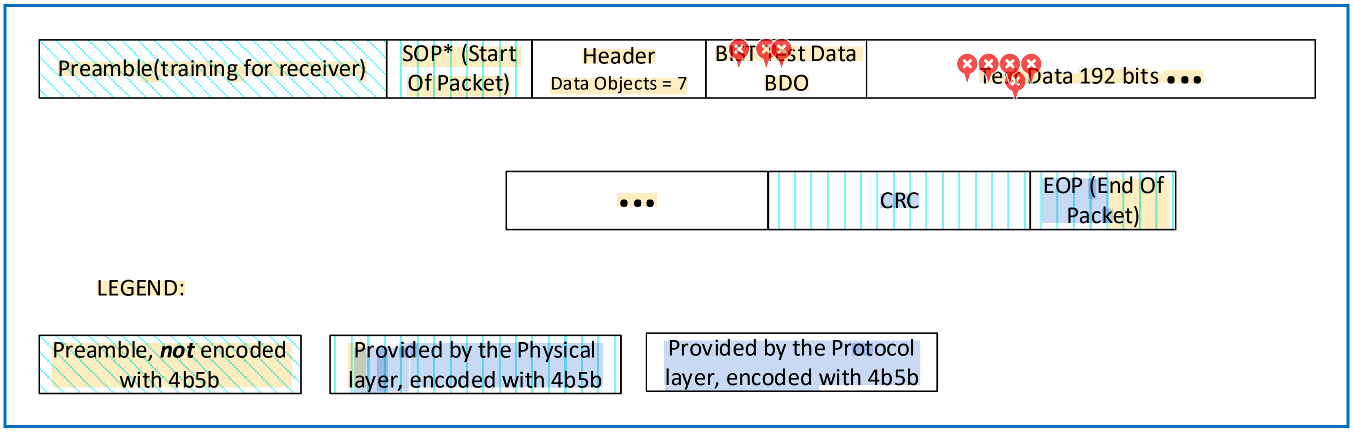
See also *Section 6.4.3 “BIST Message”.*

5.9.2 BIST Test Data

A ***BIST Test Data*** Message is used by the Tester to send various Tester generated test patterns to the UUT in order to test the UUT’s receiver. See also *Section 6.4.3 “BIST Message”.*.

Figure 5-29 “Test Data Frame” shows the Test Data Frame which ***Shall*** be sent by the Tester to the UUT. The ***BIST*** Message, with a ***BIST Test Data*** BIST Data Object consists of a Preamble, followed by ***SOP****, followed by the Message Header with a data length of 7 Data Objects, followed a ***BIST Test Data*** BIST Data Object, followed by 6 Data Objects containing Test data, followed by the CRC and then an ***EOP***.

Figure 5-29 “Test Data Frame”


Preamble(training for receiver)
SOP* (Start Of Packet)
Header
Data Objects = 7
BIST Test Data BDO
Test Data 192 bits •••
•••
CRC
EOP (End Of Packet)
LEGEND:
Preamble, *not* encoded with 4b5b
Provided by the Physical layer, encoded with 4b5b
Provided by the Protocol layer, encoded with 4b5b