Specification for C-PHY℠

Version 1.2 – 26 November 2016

MIPI Board Adopted 28 March 2017

* NOTE TO IMPLEMENTERS *

This document is a Specification. MIPI member companies' rights and obligations apply to this Specification as defined in the MIPI Membership Agreement and MIPI Bylaws.

This page intentionally left blank.

Specification for C-PHY℠

Version 1.2

26 November 2016

MIPI Board Adopted 28 March 2017

Further technical changes to this document are expected as work continues in the C-PHY Subgroup of the PHY Working Group.

NOTICE OF DISCLAIMER

The material contained herein is not a license, either expressly or impliedly, to any IPR owned or controlled by any of the authors or developers of this material or MIPI®. The material contained herein is provided on an "AS IS" basis and to the maximum extent permitted by applicable law, this material is provided AS IS AND WITH ALL FAULTS, and the authors and developers of this material and MIPI hereby disclaim all other warranties and conditions, either express, implied or statutory, including, but not limited to, any (if any) implied warranties, duties or conditions of merchantability, of fitness for a particular purpose, of accuracy or completeness of responses, of results, of workmanlike effort, of lack of viruses, and of lack of negligence.

All materials contained herein are protected by copyright laws, and may not be reproduced, republished, distributed, transmitted, displayed, broadcast or otherwise exploited in any manner without the express prior written permission of MIPI Alliance. MIPI, MIPI Alliance and the dotted rainbow arch and all related trademarks, tradenames, and other intellectual property are the exclusive property of MIPI Alliance and cannot be used without its express prior written permission.

ALSO, THERE IS NO WARRANTY OF CONDITION OF TITLE, QUIET ENJOYMENT, QUIET POSSESSION, CORRESPONDENCE TO DESCRIPTION OR NON-INFRINGEMENT WITH REGARD TO THIS MATERIAL OR THE CONTENTS OF THIS DOCUMENT. IN NO EVENT WILL ANY AUTHOR OR DEVELOPER OF THIS MATERIAL OR THE CONTENTS OF THIS DOCUMENT OR MIPI BE LIABLE TO ANY OTHER PARTY FOR THE COST OF PROCURING SUBSTITUTE GOODS OR SERVICES, LOST PROFITS, LOSS OF USE, LOSS OF DATA, OR ANY INCIDENTAL, CONSEQUENTIAL, DIRECT, INDIRECT, OR SPECIAL DAMAGES WHETHER UNDER CONTRACT, TORT, WARRANTY, OR OTHERWISE, ARISING IN ANY WAY OUT OF THIS OR ANY OTHER AGREEMENT, SPECIFICATION OR DOCUMENT RELATING TO THIS MATERIAL, WHETHER OR NOT SUCH PARTY HAD ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Without limiting the generality of this Disclaimer stated above, the user of the contents of this Document is further notified that MIPI: (a) does not evaluate, test or verify the accuracy, soundness or credibility of the contents of this Document; (b) does not monitor or enforce compliance with the contents of this Document; and (c) does not certify, test, or in any manner investigate products or services or any claims of compliance with the contents of this Document. The use or implementation of the contents of this Document may involve or require the use of intellectual property rights ("IPR") including (but not limited to) patents, patent applications, or copyrights owned by one or more parties, whether or not Members of MIPI. MIPI does not make any search or investigation for IPR, nor does MIPI require or request the disclosure of any IPR or claims of IPR as respects the contents of this Document or otherwise.

Questions pertaining to this document, or the terms or conditions of its provision, should be addressed to:

MIPI Alliance, Inc.
c/o IEEE-ISTO
445 Hoes Lane
Piscataway, NJ 08854
Attn: Board Secretary

Contents

Figures

Tables

Release History

Date	Version	Description
2014-Oct-07	v1.0	Initial Board Approved Release.
2016-Feb-11	v1.1	Board Approved Release.
2017-Mar-28	v1.2	Board Approved Release.

This page intentionally left blank.

1 Introduction

This document describes a High-Speed serial interface called C-PHY, which provides high throughput performance over bandwidth limited channels for connecting to peripherals, including displays and cameras. (This includes display Chip-on-Glass receiver channels and image sensor transmitters that exhibit bandwidth limitations.)

The C-PHY is based on 3-Phase symbol encoding technology delivering 2.28 bits per symbol over three-wire trios, and is targeting 2.5Gsymbols/s. C-PHY has many characteristics in-common with D-PHY [MIPI01]; many parts of C-PHY were adapted from D-PHY. C-PHY was designed to be able to coexist on the same IC pins as D-PHY so that dual-mode devices can be developed.

1.1 Scope

The scope of this document is to describe the lowest layers of the High-Speed interfaces to be applied by MIPI Alliance application or protocol level specifications. This includes the physical interface, electrical interface, low-level timing and the PHY-level protocol. The goal has been to define a C-PHY High-Speed interface that can coexist on the same pins as the MIPI D-PHY interface. These functional areas taken together are known as C-PHY.

The C-PHY specification shall always be used in combination with a higher layer MIPI specification that references this specification. Any other use of the C-PHY specification is strictly prohibited, unless approved in advance by the MIPI Board of Directors.

The following topics are outside the scope of this document:

- **Explicit specification of signals of the clock generator unit.** The C-PHY specification does implicitly require a minimum performance of the internal clock signals in order to meet the defined specifications of the external signals. Intentionally, only the behavior on the interface pins is constrained. Therefore, the clock generation unit is excluded from this specification, and will be a separate functional unit that provides the required clock signals to the C-PHY in order to meet the specification. This allows many implementation trade-offs as long as these do not violate this specification.
- **Procedure to resolve contention situations.** The C-PHY contains several mechanisms to detect Link contention. However, certain contention situations can only be detected at higher levels and are therefore not included in this specification.
- **Ensure proper operation of a connection between different Lane Module types.** There are several different Lane Module types to optimally support the different functional requirements of several applications. This means that next to some base-functionality there are optional features which can be included or excluded. This specification only ensures correct operation for a connection between matched Lane Modules types, which means: Modules that support the same features and have complementary functionality. In case the two sides of the Lane are not the same type, and these are supposed to work correctly, it shall be ensured by the manufacturer(s) of the Lane Module(s) that the provided additional functionality does not corrupt operation. This can be most easily accomplished if the additional functionality can be disabled by other means independent of the MIPI C-PHY interface, such that the Lane Modules behave as if they were the same type.
- **ESD protection level of the IO.** The required level of ESD protection will depend on a particular application environment and product type.
- **Exact symbol error rate value.** The actual value of the achieved symbol error rate depends on the total system integration and the hostility of the environment. Therefore, it is impossible to specify a symbol error rate for individual parts of the Link. This specification allows for implementations with a symbol error rate less than 10^{-12}.

- **Specification of the PHY-Protocol Interface.** The C-PHY specification includes a PHY-Protocol Interface (PPI) annex that provides one possible solution for this interface. This annex is limited to the essential signals for normal operation in order to clarify the kind of signals needed at this interface. For power reasons this interface will be internal for most applications. Practical implementations may be different without being inconsistent with the C-PHY specification.
- **Implementations.** This specification is intended to restrict the implementation as little as possible. Various sections of this specification use block diagrams or example circuits to illustrate the concept and are not in any way claimed to be the preferred or required implementation. Only the behavior on the C-PHY interface pins is normative. Regulatory compliance methods are not within the scope of this document. It is the responsibility of product manufacturers to ensure that their designs comply with all applicable regulatory requirements.

The C-PHY Specification evolution is primarily driven by the need to achieve higher data rates and better efficiency, while at the same time respecting backward compatibility. In this process the previous version of the specification is taken and modifications are added, without compromising backward compatibility. Each new version of the specification that is derived both preserves all the specification components of the previous version, and adds the new changes. Due to technology evolution, some parameters are changed to optimize for newer technologies.

It is recommended to always follow the latest version of the C-PHY Specification, irrespective of the targeted data rate. The product data sheet should mention both the targeted C-PHY Specification version and data rates. This will enable the system integrator to make proper decisions to achieve interoperability goals.

Items that are outside of the scope of this document are generally the same as those described in the D-PHY specification, except for a detailed description of built-in test circuitry and test patterns. The built-in test circuit description is included as an informative chapter to be followed at the option of the system or device implementer. This document deviates from the norm and defines the behaviors of the test circuitry because the general functionality of C-PHY is different from most other PHY implementations. It is useful to provide a common test circuit description that can be followed by device implementers and test equipment providers so there will be compatibility between devices implementing C-PHY and test equipment.

Coexistence with D-PHY on the same IC pins is possible, and likely in many applications; the means of doing so is beyond the scope of this standard.

Regulatory compliance methods are not within the scope of this document. It is the responsibility of product manufacturers to ensure that their designs comply with all applicable regulatory requirements.

1.2 Purpose

The C-PHY specification is used by manufacturers to design products that adhere to MIPI Alliance interface specifications for mobile devices such as, but not limited to, camera, display and unified protocol interfaces.

Implementing this specification reduces the time-to-market and design cost of mobile devices by standardizing the interface between products from different manufacturers. In addition, richer feature sets requiring high data rates can be realized by implementing this specification. Finally, adding new features to mobile devices is simplified due to the extensible nature of the MIPI Alliance Specifications.

2 Terminology

2.1 Use of Special Terms

The MIPI Alliance has adopted Section 13.1 of the *IEEE Standards Style Manual*, which dictates use of the words “shall”, “should”, “may”, and “can” in the development of documentation, as follows:

> The word *shall* is used to indicate mandatory requirements strictly to be followed in order to conform to the Specification and from which no deviation is permitted (*shall* equals *is required to*).
>
> The use of the word *must* is deprecated and shall not be used when stating mandatory requirements; *must* is used only to describe unavoidable situations.
>
> The use of the word *will* is deprecated and shall not be used when stating mandatory requirements; *will* is only used in statements of fact.
>
> The word *should* is used to indicate that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain course of action is deprecated but not prohibited (*should* equals *is recommended that*).
>
> The word *may* is used to indicate a course of action permissible within the limits of the Specification (*may* equals *is permitted to*).
>
> The word *can* is used for statements of possibility and capability, whether material, physical, or causal (*can* equals *is able to*).

All sections are normative, unless they are explicitly indicated to be informative.

2.2 Definitions

Bi-directional: A single Lane that supports communication in both the Forward Direction and the Reverse Direction.

C-PHY: The PHY defined in this document. C-PHYs may be used in channel-limited applications, hence the use of the character “C.”

Escape Mode: An optional mode of operation for Lanes that allows low bit-rate commands and data to be transferred at very low power.

Forward Direction: The signal direction is defined relative to the direction of the High-Speed data. The main direction of data communication, from source to sink, is denoted as the Forward Direction. Data is sent in the Forward Direction when the system is initialized.

Lane: A Lane consists of two complementary Lane Modules communicating via three-Line, point-to-point Lane Interconnects. The term “Lane” is often used to denote interconnect only.

Lane Interconnect: Three-Line, point-to-point interconnect used for both differential High-Speed signaling and Low-Power, single-ended signaling.

Lane Module: Module at each side of the Lane for driving and/or receiving signals on the Lane.

Line: An interconnect wire used to connect a driver to a receiver. Three Lines are required to create a Lane Interconnect.

Link: A connection between two devices containing at least one Lane. A Link consists of at least two PHYs and one Lane Interconnect.

Master: The Master is the PHY on the side of a Link that is the main data source. The Master transmits data in the Forward Direction and receives data in the Reverse Direction. In some Bi-directional systems the functions of the Master and Slave are nearly equivalent. The difference between being Master or Slave is

simply that one side is identified as the Master, and the other side is identified as the Slave. The side that is defined as the Master does not change after the system is initialized.

PHY: A functional block that implements the features necessary to communicate over the Lane Interconnect. A PHY consists of one or more Lane Modules and a PHY Adapter layer.

PHY Adapter: A protocol layer that converts symbols from an APPI to the signals used by a specific PHY PPI.

PHY Configuration: A set of Lanes that represent a possible Link. A PHY Configuration consists of one or more Lanes.

Reverse Direction: Reverse Direction is the direction that is opposite to the Forward Direction (see definition above). Data is sent in the Forward Direction when the system is initialized.

Slave: The Slave is the PHY on the side of a Link that is the main data sink. The Slave receives data in the Forward Direction and transmits data in the Reverse Direction. In some Bi-directional systems the functions of the Master and Slave are nearly equivalent. The difference between being Master or Slave is simply that one side is identified as the Master, and the other side is identified as the Slave. The side that is defined as the Slave does not change after the system is initialized.

Turnaround: Reversing the direction of communication on a Lane.

Unidirectional: A single Lane that supports communication in the Forward Direction only.

Wire State: the combination of signal levels driven on the three Lines of a Lane.

2.3 Abbreviations

e.g.	For example (Latin: exempli gratia)
i.e.	That is (Latin: id est)

2.4 Acronyms

ALP	Alternate Low-Power: identifier for operation mode
APPI	Abstracted PHY-Protocol Interface
BER	Bit Error Rate
CIL	Control and Interface Logic
DDR	Double Data Rate
EMI	Electro Magnetic Interference
EoT	End of Transmission
HS	High-Speed; identifier for operation mode
HS-RX	High-Speed Receiver (Low-Swing Differential)
HS-TX	High-Speed Transmitter (Low-Swing Differential)
IEEE	Institute of Electrical and Electronics Engineers
IO	Input-Output
ISTO	Industry Standards and Technology Organization
LP	Low-Power: identifier for operation mode
LP-CD	Low-Power Contention Detector
LPDT	Low-Power Data Transmission

LP-RX	Low-Power Receiver (Large-Swing Single-Ended)
LP-TX	Low-Power Transmitter (Large-Swing Single-Ended)
LPS	Low-Power State(s)
LSB	Least Significant Bit
LVLP	Low Voltage Low Power; optional signal voltage range in LP mode
Mbps	Megabits per second
MSB	Most Significant Bit
PHY	Physical Layer
PICS	Protocol Implementation Conformance Statement
PLL	Phase-Locked Loop
PPI	PHY-Protocol Interface
PWB	Printed Wiring Board
PRBS	Pseudorandom Binary Sequence
RF	Radio Frequency
RX	Receiver
SE	Single-Ended
SoT	Start of Transmission
TLIS	Transmission-Line Interconnect Structure: physical interconnect realization between Master and Slave
TX	Transmitter
UI	Unit Interval, equal to the duration of any HS state
ULPS	Ultra-Low Power State

3 References

[MIPI01] *MIPI Alliance Specification for D-PHY*, version 1.2, MIPI Alliance, Inc., 10 September 2014.

[MIPI02] *MIPI Alliance Specification for C-PHY*, version 1.0, MIPI Alliance, Inc., 5 August 2014.

[PET01] Peterson, W. Wesley; Weldon, E. J. Jr., *Error-Correcting Codes*, Second Edition, Massachusetts Institute of Technology, 1972.

4 C-PHY Overview

C-PHY describes a High-Speed, rate-efficient PHY, especially suited for mobile applications where channel rate limitations are a factor. The needs of rate limited channels are accomplished through the use of 3-Phase symbol encoding technology delivering approximately 2.28 bits per symbol over a three-wire group of conductors. This C-PHY specification has been written primarily for the connection of cameras and displays to a host processor. Nevertheless, it can be applied to many other applications.

It is envisioned that C-PHY will sometimes be used in dual-simplex or full-duplex configurations for interconnections in a more generic communication network. The symbol rates for the Forward and Reverse Directions can be either equal or asymmetrical, depending upon the system requirements. Since the C-PHY Lane has an embedded clock, the Reverse Direction symbol rate can be either faster or slower than the Forward Direction symbol rate. The capability to transmit or receive HS Reverse data is optional. Exploiting the HS Reverse feature is attractive for applications with asymmetrical data traffic requirements, and when the cost of separate interconnects for a return channel is too high. While the HS Reverse feature is optional, it avoids mandatory overhead costs for applications that do not have return traffic requirements, or that want to apply physically distinct return communication channels.

C-PHY has re-used many parts of the D-PHY standard. C-PHY was designed to coexist on the same IC pins as D-PHY, so that dual-mode devices can be developed. C-PHY High-Speed data coding differs substantially from the D-PHY clock-forwarding system, although the High-Speed signal levels and terminations bear some similarity. The low-power mode of D-PHY is reused almost completely, and the transitions to and from the High-Speed and low-power modes is very similar to the D-PHY standard.

Key characteristics of C-PHY coding are:

- Uses a group of three conductors, rather than conventional pairs. The group of three wires is called a Lane, and the individual Lines of the Lane are called A, B and C. C-PHY does not have a separate clock Lane.
- Within a three-wire Lane, two of the three wires are driven to opposite levels; the third wire is terminated to a mid-level (at either one end or both ends), and the voltages at which the wires are driven changes at every symbol.
- Multiple bits are encoded into each symbol epoch, the data rate is ~2.28x the symbol rate. There is no additional overhead for line coding, such as 8b10b, which is not needed.
- Clock timing is encoded into each symbol. This is accomplished by requiring that the combination of voltages driven onto the wires must change at every symbol boundary. This simplifies clock recovery.
- The signal is received using a group of three differential receivers.
- The C-PHY interface can co-exist on the same pins/pads as the D-PHY interface signals.

4.1 Summary of PHY Functionality

The C-PHY provides a synchronous connection between Master and Slave. A practical PHY Configuration consists of one or more three-wire Lanes. The Link includes a High-Speed signaling mode for fast-data traffic and a low-power signaling mode for control purposes. Optionally, a low-power Escape Mode can be used for low speed asynchronous data communication. High-Speed data communication appears in bursts with an arbitrary number of payload data words. The low-power mode and Escape Mode remain the same as defined in the D-PHY specification.

The PHY uses three wires per Lane, so three wires are required for the minimum PHY Configuration. In High-Speed mode each Lane is terminated on both sides and driven by a low-swing, 3-Phase signal. In low-power mode all wires are operated single-ended and non-terminated. To minimize EMI, the drivers for this mode shall be slew-rate controlled and current limited.

The maximum achievable bit rate in High-Speed mode is determined by the performance of transmitter, receiver and interconnect implementations. The maximum symbol rate capability ranges from 0.8 Gsps to

3 Gsps, depending upon the combination of features and channel characteristics. The symbol rate ranges for different combinations of transmitters, receivers and channels are described in ***Section 10.3.3***. For a fixed clock frequency, the available data capacity of a PHY Configuration can be increased by using more Lanes. Effective data throughput can be reduced by employing burst mode communication. The maximum data rate in low-power mode is 10 Mbps.

The features introduced by this specification can be applied to any High-Speed symbol rate.

4.1.1 Summary of Lane Signaling States

Figure 1 shows the current flow through the Lane for all six Wire States. The circuit examples below are simplified for the data encoding example; they are intended to illustrate the basic signaling states on the three-wire Link. ***Figure 1*** shows the positive-polarity Wire States on the left and negative-polarity Wire States on the right. The three rotation states (x, y and z) are shown from top to bottom. The six driven states (called Wire States) on a C-PHY Lane are called: +x, -x, +y, -y, +z, and -z. The positive polarity Wire States have the same wires driven as the corresponding negative polarity states, but the polarity is opposite on the driven pair of wires. For example: the +x Wire State is defined as A being driven high and B driven low, while the -x Wire State is B driven high and A driven low. The “undriven” conductor can be undriven when operating at lower symbol rates, or is actually driven by a termination at a voltage half way between the highest and lowest driven levels if operating at higher symbol rates.

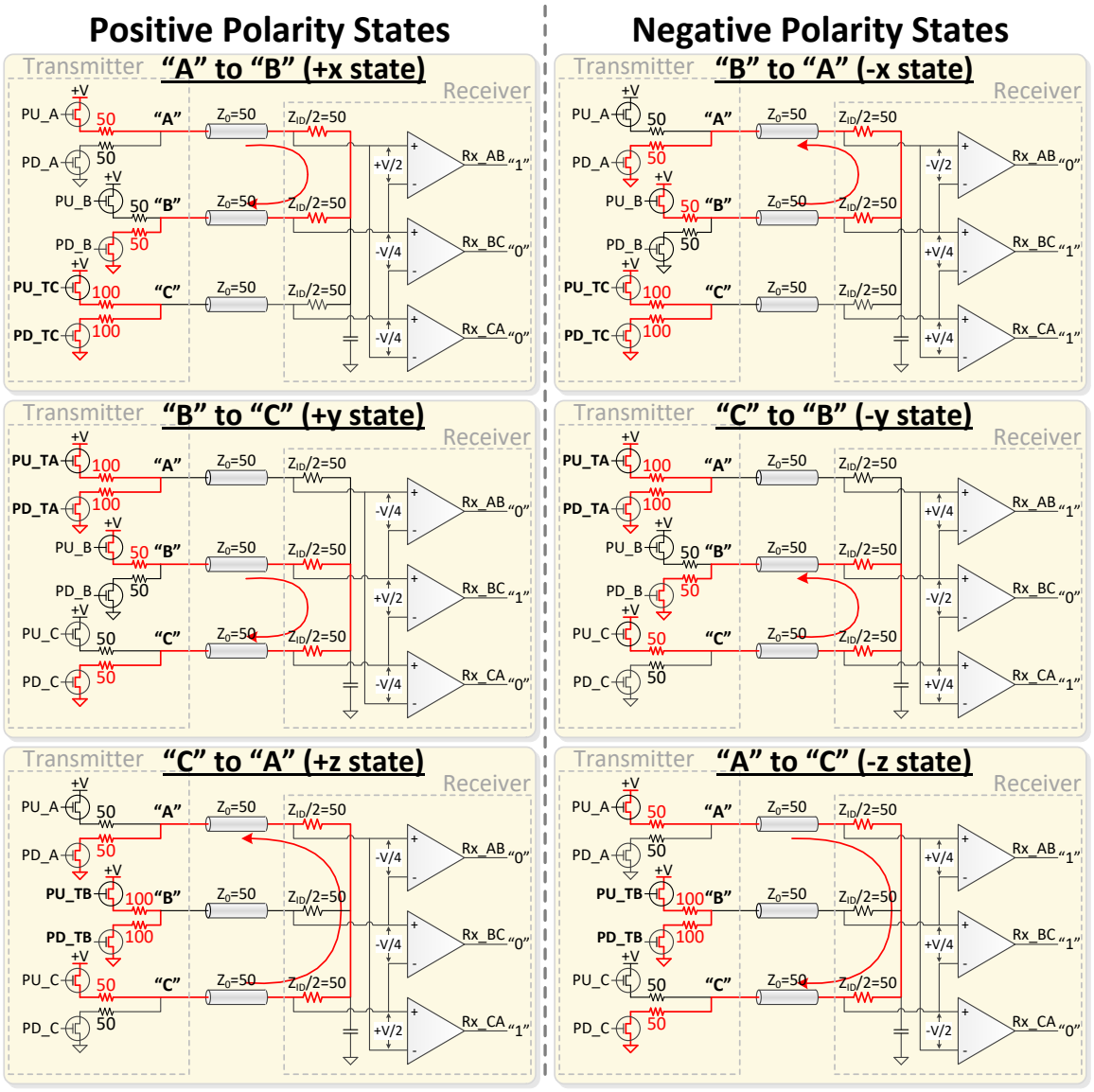


Figure 1 Six Physical Layer Wire States of C-PHY Encoding, Nominal Values Shown

Table 1 Signal Voltage and Differential Voltage for the Six C-PHY Wire States

Wire State	Wire Amplitude			Receiver diff input voltage			Receiver digital output		
	A	B	C	A – B	B – C	C – A	Rx_AB	Rx_BC	Rx_CA
+x	¾ V	¼ V	½ V	+½ V	-¼ V	-¼ V	1	0	0
-x	¼ V	¾ V	½ V	-½ V	+¼ V	+¼ V	0	1	1
+y	½ V	¾ V	¼ V	-¼ V	+½ V	-¼ V	0	1	0
-y	½ V	¼ V	¾ V	+¼ V	-½ V	+¼ V	1	0	1
+z	¼ V	½ V	¾ V	-¼ V	-¼ V	+½ V	0	0	1
-z	¾ V	½ V	¼ V	+¼ V	+¼ V	-½ V	1	1	0

4.1.2 Representation of Symbols in High-Speed Mode

One of the symbol to Wire State encoding rules is that a state-transition exists at every symbol boundary. The reason for this rule is that it encodes the clock timing within the symbol, which has a number of advantages.

With six possible Wire States (as shown in ***Figure 1*** and ***Table 1***) there are always 5 possible transitions to the next Wire State from any present Wire State. The possible state transitions are illustrated in the state diagram in ***Figure 2***. The symbol value is defined by the change in Wire State values from one unit interval to the next. Note that more than two bits of information (actually $\log_2(5) = 2.3219$ bits) can be encoded into each symbol. Seven consecutive symbols are used to transmit 16 bits of information. (Note that $5^7 = 78{,}125$ permutations in seven consecutive symbols, with five possible Wire State transitions that define each symbol. The information encoded in seven symbols is more than sufficient to represent a 16-bit binary value, $2^{16} = 65{,}536$.)

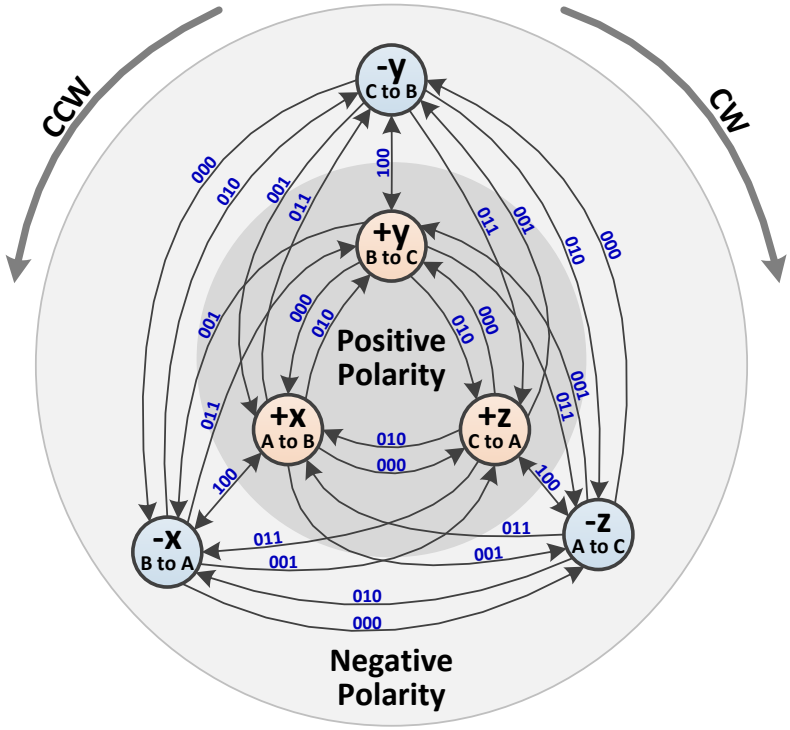


Figure 2 State Diagram Showing All Six Wire States, and All Possible Transitions

4.1.3 Representation of High-Speed Signaling States

Figure 3 shows all processes involved in the transmission of 16-bit data from the transmitter to the reception of 16-bit words in the receiver. 16-bit words at the transmitter are converted to seven channel symbols by a Mapper. Then the seven symbols are serialized and sent one symbol at a time to a Symbol Encoder and 3-wire driver which drives the three signals of the Lane (A, B and C Lines) at the transmitting end. At the receiving end, there are three differential receivers that receive A minus B, B minus C, and C minus A. The digital outputs of the differential receivers connect to a Symbol Decoder and clock recovery circuit. The output of the Symbol Decoder is fed to a serial-to-parallel converter, and every group of 7 symbols output by the Symbol Decoder is presented to the De-Mapper where they are converted back to a 16-bit word. The functional details of the Mapper, Symbol Encoder, Symbol Decoder and De-Mapper are described in ***Section 6.1.3***.

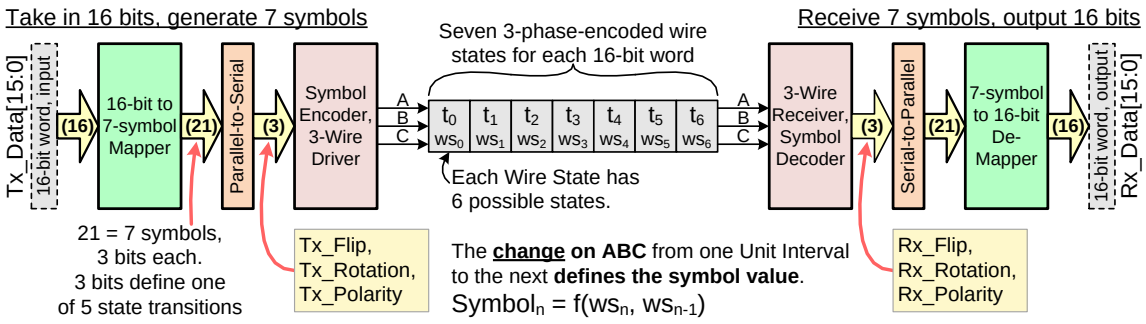


Figure 3 End-to-End Transmission of Data, 16-bit Word Conversion to Channel States

4.2 Mandatory Functionality

All functionality that is specified in this document shall be implemented for all C-PHY Configurations, unless it is specifically stated as informative or specified as optional in ***Section 5.5***.

5 Architecture

This Section describes the internal structure of the PHY including its functions at the behavioral level. Furthermore, several possible PHY Configurations are given. Each configuration can be considered as a suitable combination from a set of basic modules.

5.1 Lane Modules

A PHY Configuration consists of one or more Lane Modules. Each of these PHY Lane Modules communicates via three Lines to a complementary part at the other side of the Lane Interconnect.

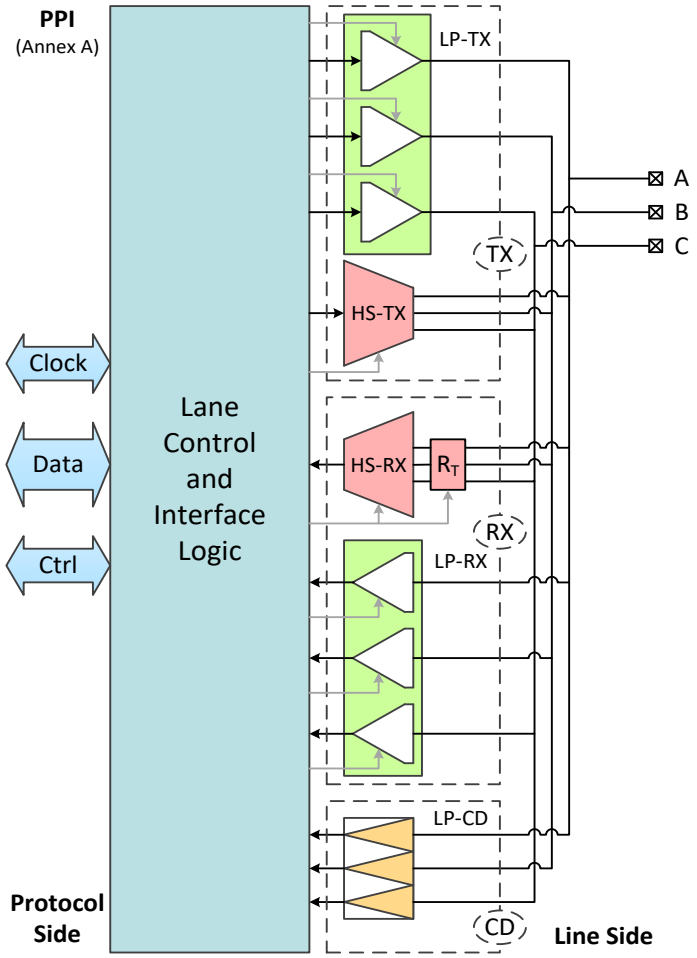


Figure 4 Universal Lane Module Functions

Each Lane Module consists of one or more High-Speed functions utilizing three interconnect wires simultaneously, one or more single-ended low-power functions operating on each of the interconnect wires individually, and control & interface logic. An overview of all functions is shown in ***Figure 4***. High-Speed signals have a low voltage swing, e.g. 250 mV, while low-power signals have a large swing, e.g. 1.2V. High-Speed functions are used for High-Speed data transmission. The low-power functions are mainly used for control, but have other, optional, use cases. The I/O functions are controlled by a Lane Control and Interface Logic block. This block interfaces with the higher layer protocol unit and determines the global operation of the Lane Module.

High-Speed functions include a differential transmitter (HS-TX) and a differential receiver (HS-RX).

A Lane Module may contain a HS-TX, a HS-RX, or both. If a Lane Module contains both a HS-TX and a HS-RX, they are never both enabled simultaneously during normal operation. An enabled High-Speed function shall terminate the Lane on its side of the Lane Interconnect as defined in ***Section 9.1.1*** and ***Section 9.2.1***. If a High-Speed function in the Lane Module is not enabled, then the function shall be put into a high impedance state.

Low-power functions include single-ended Transmitters (LP-TX), Receivers (LP-RX), and low-power Contention-Detectors (LP-CD). Low-power functions are always associated with a Lane, as these are single-ended functions operating on all three of the interconnect wires individually. An LP-TX may support an optional Low Voltage Low Power (LVLP) operation, in which the maximum voltage is limited in comparison to the normal Low-Power mode. An LP-RX, which meets the V_{IH} specification in ***Section 9.2.2***, supports LVLP operation.

Presence of High-Speed and low-power functions is correlated. That is, if a Lane Module contains a HS-TX, then it shall also contain a LP-TX. A similar constraint holds for HS-RX and LP-RX.

If a Lane Module containing a LP-RX is powered, then that LP-RX shall always be active and continuously monitor Line levels. A LP-TX shall be enabled only when driving low-power states. The LP-CD function is required only for Bi-Directional operation. If present, the LP-CD function is enabled to detect contention situations while the LP-TX is driving low-power states. The LP-CD checks for contention before driving a new state on the Line, except in ULPS.

The activities of LP-TX, HS-TX, and HS-RX in a single Lane Module are mutually exclusive, except for some short crossover periods. For detailed specification of the Line-side signals, and the HS-TX, HS-RX, LP-TX, LP-RX, and LP-CD functions, refer to ***Section 9*** and ***Section 10***.

For proper operation, the set of functions in the Lane Modules on both sides of the Lane Interconnect has to be matched. This means for each HS and LP transmit or receive function on one side of the Lane Interconnect, a complementary HS or LP receive or transmit function must be present on the other side. In addition, a contention detector is needed in any Lane Module that combines TX and RX functions.

5.2 Master and Slave

Each Link has a Master side and a Slave side. For Links that are capable of sending HS data in only one direction, the Master is the main data source and the Slave is the main data sink. The main direction of data communication, from source to sink, is called the Forward Direction. Data communication in the opposite direction is called Reverse transmission. Only Bi-Directional Lanes can transmit in the Reverse Direction. Bi-directional Lanes can be turned around, so that they source High-Speed data from the Slave and sink data at the Master.

5.3 High Frequency Clock Generation

In many cases, a PLL Clock Multiplier is needed for the high frequency clock generation at the Master side. The C-PHY specification uses an architectural model where a separate Clock Multiplier Unit outside the PHY generates the required high frequency clock signals for the PHY. Whether this Clock Multiplier Unit in practice is integrated inside the PHY is left to the implementer.

Some examples of timing sources for the Reverse Direction Link are shown in ***Figure 5***.

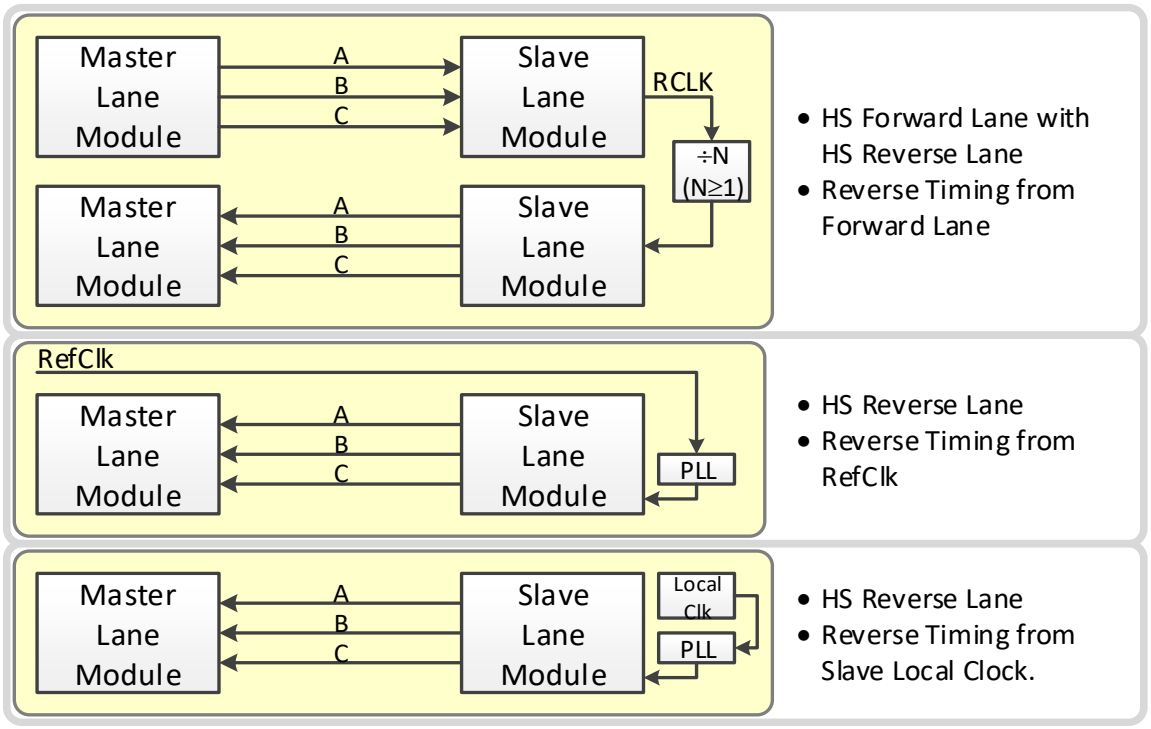


Figure 5 Examples of Reverse Link Timing Sources

5.4 Lanes and the PHY-Protocol Interface

A complete Link contains, beside Lane Modules, a PHY Adapter Layer that ties all Lanes, the Clock Multiplier Unit, and the PHY-Protocol Interface together. ***Figure 6*** shows a PHY Configuration example for a Link with three Lanes plus a separate Clock Multiplier Unit. The PHY Adapter Layer, though a component of a PHY, is not within the scope of this Specification.

The logical PHY-Protocol interface (PPI) for each individual Lane includes a set of signals to cover the functionality of that Lane. As shown in ***Figure 6***, Clock signals may be shared for all Lanes. The reference clock and control signals for the Clock Multiplier Unit are not within the scope of this Specification.

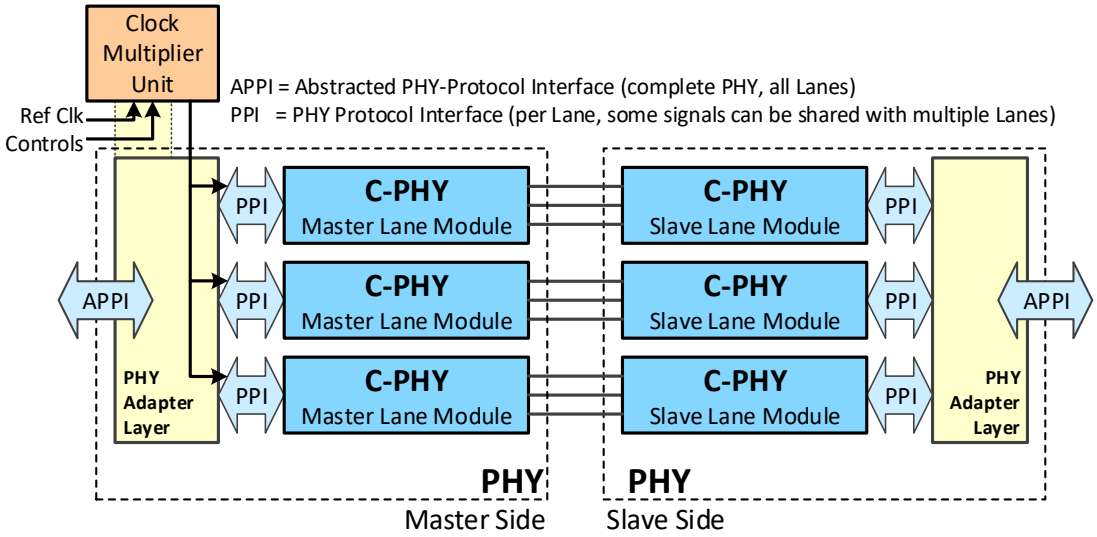


Figure 6 Three Lane PHY Configuration

5.5 Selectable Lane Options

A PHY Configuration consists of one or more Lanes. All Lanes shall support High-Speed transmission and Escape Mode in the Forward Direction.

There are two main types of Lanes:

- Bi-Directional (featuring Turnaround and some Reverse communication functionality)
- Unidirectional (without Turnaround or any kind of Reverse communication functionality)

Bi-Directional Lanes shall include one or both of the following Reverse communication options:

- High-Speed reverse data communication
- Low-power Reverse Escape Mode (including or excluding LPDT).

All Lanes shall include Escape Mode support for ULPS and Triggers in the Forward Direction. Other Escape Mode functionality is optional; all possible Escape Mode features are described in ***Section 6.6***. Applications shall define what additional Escape Mode functionality is required and, for Bi-Directional Lanes, shall select Escape Mode functionality for each direction individually.

This results in many options for complete PHY Configurations. The degrees of freedom are:

- Single or multiple Lanes
- Bi-Directional and/or Unidirectional Lane (per Lane)
- Supported types of reverse communication (per Lane)
- Functionality supported by Escape Mode (for each direction, per Lane)

Figure 7 is a flow graph of the option selection process. Practical configuration examples can be found in ***Section 5.7***.

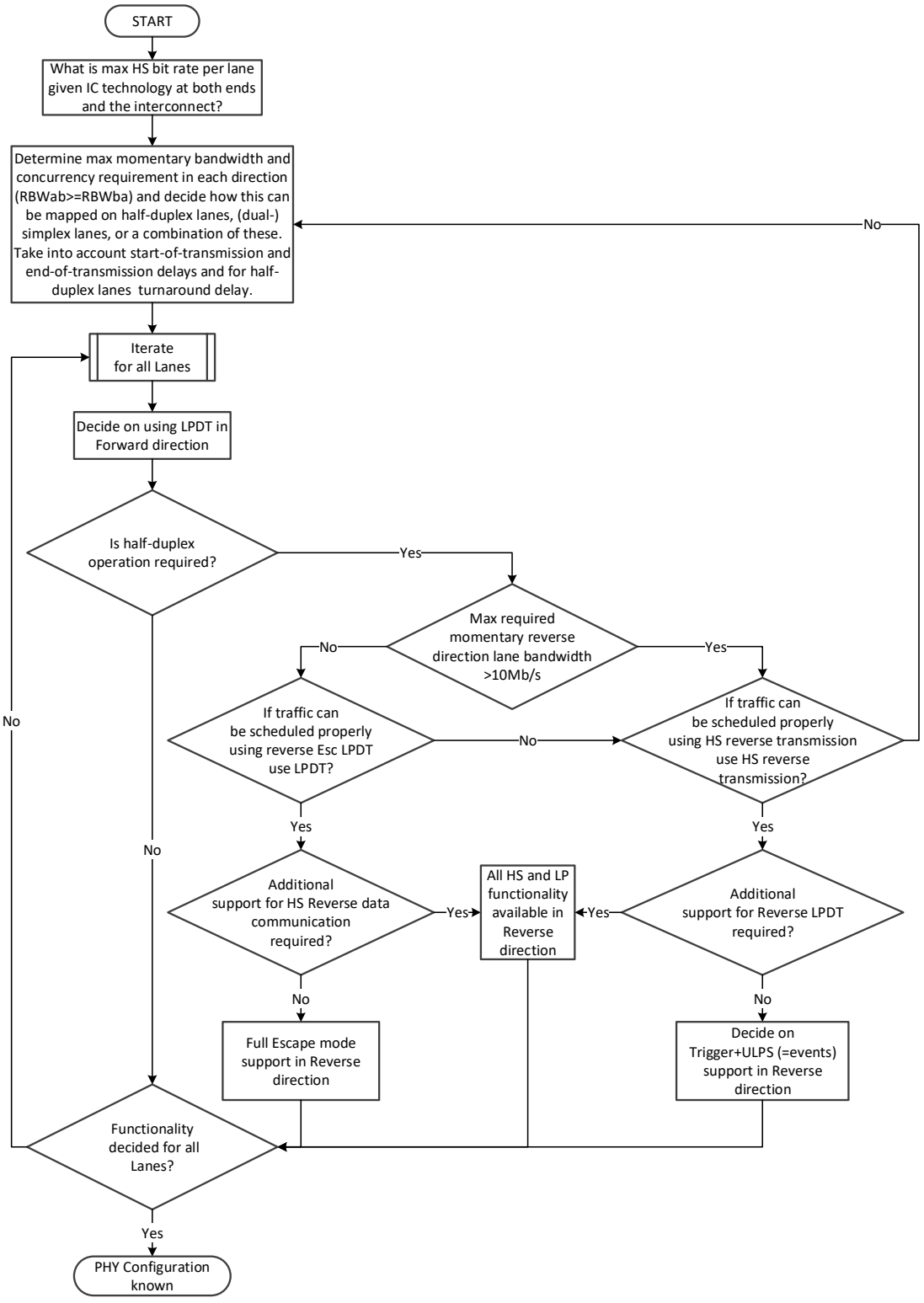


Figure 7 Option Selection Flow Graph

5.6 Lane Module Types

The required functions in a Lane Module depend on the Lane type, and on which side of the Lane Interconnect the Lane Module is located. There are two main Lane types: Unidirectional Lane and Bi-Directional Lane. Several PHY Configurations can be constructed with these Lane types. See ***Figure 7*** for more information on selecting Lane options.

Figure 8 shows a universal Lane Module diagram with a global overview of internal functionality of the CIL function. This Universal Module can be used for all Lane types. The requirements for the 'Control and Interface Logic' (CIL) function depend on the Lane type and Lane side. ***Section 6*** and ***Annex A*** implicitly specify the contents of the CIL function. The actual realization is left to the implementer.

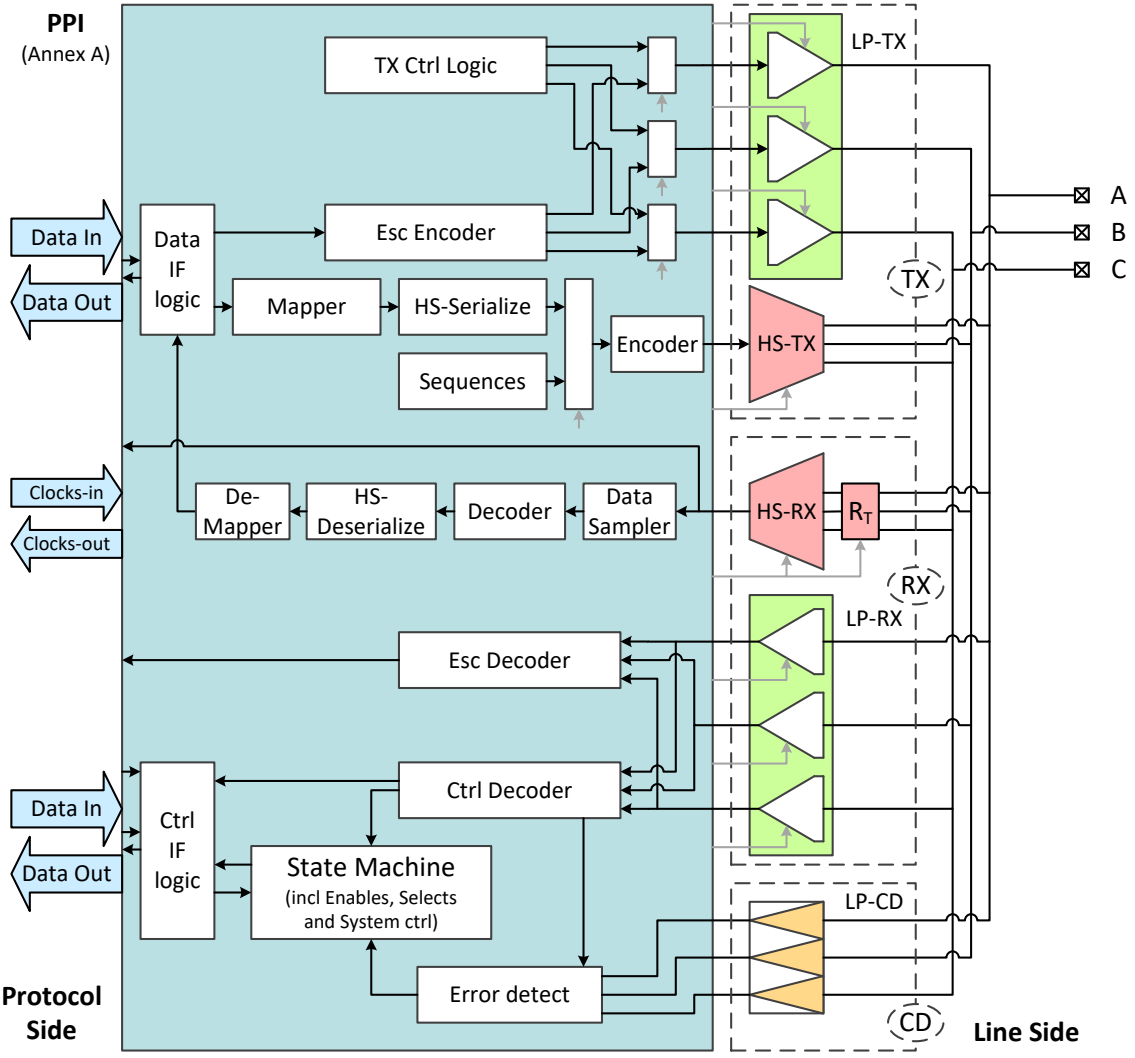


Figure 8 Universal Lane Module Architecture

Of course, stripped-down versions of the universal Lane Module that just support the required functionality for a particular Lane type are possible. These stripped-down versions are identified by the acronyms in ***Table*** 2. For simplification reasons, any of the four identification characters can be replaced by an X, which means that this can be any of the available options. For example, a CIL-MFEN is therefore a stripped-down CIL function for the Master side of a Unidirectional Lane with Escape Mode functionality only in the Forward Direction. A CIL-SRXX is a CIL function for the Slave of a Lane with support for Bi-directional High-Speed

communication and any allowed subset of Escape Mode. Note that a CIL-XFXN implies a Unidirectional Link, while either a CIL-XRXX or CIL-XXXY block implies a Bi-directional Link.

Table 2 Lane Type Descriptors

Prefix	Lane Interconnect Side	High-Speed Capabilities	Forward Direction Escape Mode Features Supported	Reverse Direction Escape Mode Features Supported
CIL-	M – Master S – Slave X – Don't Care[2]	F – Forward Only R – Reverse and Forward X – Don't Care[2]	A – All (including LPDT) E – Events – Triggers and ULPS Only X – Don't Care[2]	A – All (including LPDT) E – Events – Triggers and ULPS Only N – None Y – Any[1] (A, E or A and E) X – Don't Care[2]

Note:

1. *"Any" means any combination of one or more functions*
2. *"X" Means "F or R"*

The recommended PHY-Protocol Interface contains Data-in and Data-out in word format, Input and/or output Clock signals, and Control signals. Control signals include requests, handshakes, test settings, and initialization. A proposal for a logical internal interface is described in ***Annex A***. Although not a requirement, it may be very useful to use the proposed PPI as a guide. For external use on ICs an implementation may multiplex many signals on the same pins. However, for power efficiency reasons, the PPI is normally within an IC.

5.6.1 Unidirectional Lane

For a Unidirectional Lane the Master module shall contain at least a HS-TX, a LP-TX, and a CIL-MFXN function. The Slave shall contain at least a HS-RX, a LP-RX, and a CIL-SFXN.

5.6.2 Bi-Directional Lanes

A Bi-Directional Lane Module includes some form of Reverse communication; either High-Speed reverse communication, reverse Escape Mode, or both. The functions required depend on what methods of reverse communication are included in the Lane Module.

5.6.2.1 Bi-Directional Lane Modes

A Bi-Directional Lane Module without High-Speed reverse communication shall include a low-power reverse Escape Mode (including or excluding LPDT). The Master-side Lane Module includes a HS-TX, LP-TX, LP-RX, LP-CD, and CIL-MFXY. The Slave side consists of a HS-RX, LP-RX, LP-TX, LP-CD, and a CIL-SFXY.

5.6.2.2 Bi-Directional Lane with High-Speed Reverse Communication

A Bi-directional Lane Module with High-Speed reverse communication shall include a reverse Escape Mode. The Master-side Lane Module includes a HS-TX, HS-RX, LP-TX, LP-RX, LP-CD, and CIL-MRXX. The Slave-side consists of a HS-RX, HS-TX, LP-RX, LP-TX, LP-CD, and a CIL-SRXX.

This type of Lane Module might seem suitable for both Master and Slave, but because of the asymmetry of the Link one side shall be configured as Master and the other side as Slave.

5.7 Configurations

This Section outlines several common PHY Configurations, but should not be considered an exhaustive list of all possible arrangements. Any other configuration that does not violate the requirements of this document is also allowed.

In order to create an abstraction level, the Lane Modules are represented in this Section by Lane Module symbols. ***Figure 9*** shows the syntax and meaning of Lane Module symbols.

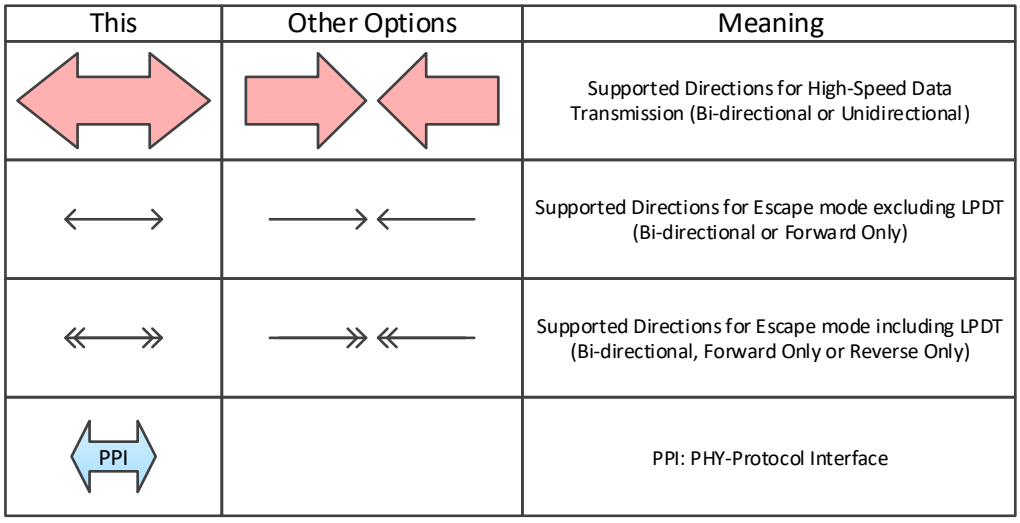


Figure 9 Lane Module Symbol Macros and Symbols Legend

For multiple Lanes, a large variety of configurations is possible. ***Figure 10*** shows an overview of symbolic representations for different Lane types. The acronyms mentioned for each Lane type represent the functionality of each Lane Module in a short way. This also sets the requirements for the CIL function inside each Lane Module.

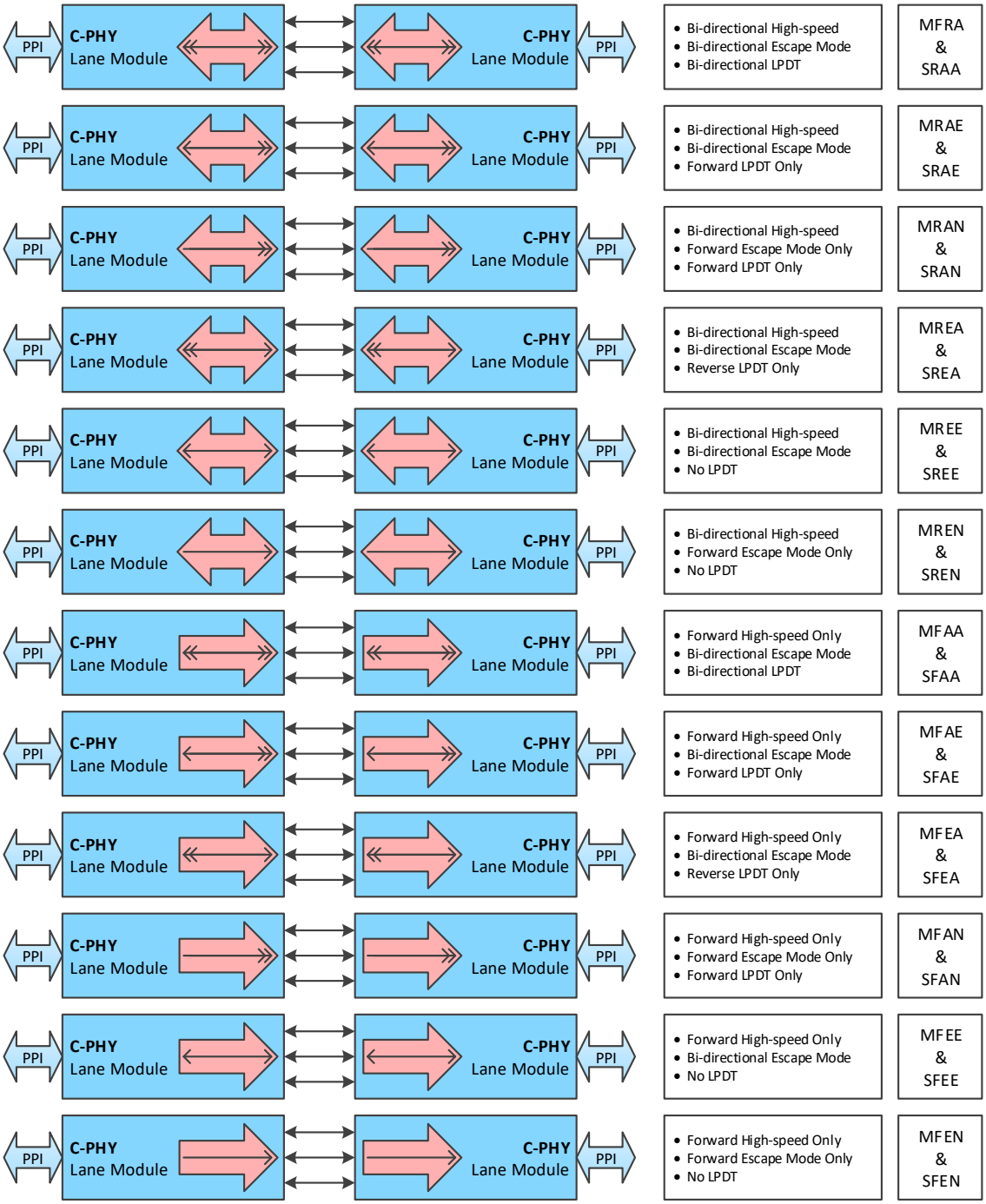


Figure 10 All Possible Lane Types

5.7.1 Unidirectional Configurations

All Unidirectional configurations are constructed with one or more Unidirectional Lanes. Two basic configurations can be distinguished: single-Lane and multiple-Lanes. For completeness a dual-simplex configuration is also shown. At the PHY level there is no difference between a dual-simplex configuration and two independent Unidirectional configurations.

5.7.1.1 PHY Configuration with a Single Lane

This configuration includes one Unidirectional Lane from Master to Slave. Communication is therefore only possible in the Forward Direction. ***Figure 11*** shows an example configuration without LPDT. This configuration requires three interconnect signal wires.

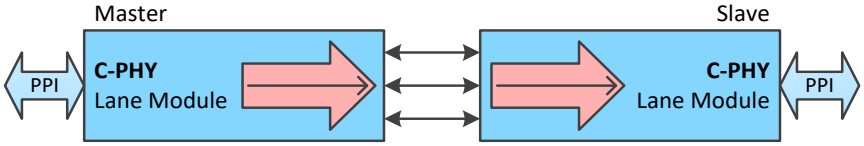


Figure 11 Unidirectional Single Lane Configuration

5.7.1.2 PHY Configuration with Multiple Lanes

This configuration includes multiple Unidirectional Lanes from Master to Slave. Bandwidth is extended, but communication is only possible in the Forward Direction. The PHY specification does not require all Lanes to be active simultaneously. In fact, the protocol layer controls all Lanes individually. ***Figure 12*** shows an example of this configuration for three Lanes. If N is the number of Lanes, then this configuration requires $3 \cdot N$ interconnect wires.

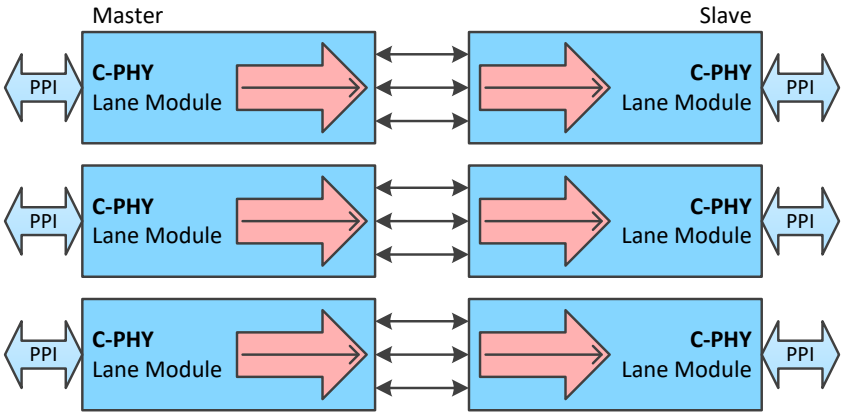


Figure 12 Unidirectional Multiple Lane Configuration without LPDT

5.7.1.3 Dual-Simplex (Two Directions with Unidirectional Lanes)

This case is the same as two independent (dual), Unidirectional (simplex) Links: one for each direction. Each direction may contain either a single Lane, or multiple Lanes. Please note that the Master and Slave Lane Modules for the two different directions are opposite. The PHY Configuration for each direction shall comply with the C-PHY specifications. As both directions are conceptually independent, the bit rates for each direction do not have to match. However, for practical implementations, it is attractive to match rates and share some internal signals as long as both links fulfill all specifications externally. ***Figure 13*** shows an example of this dual PHY Configuration.

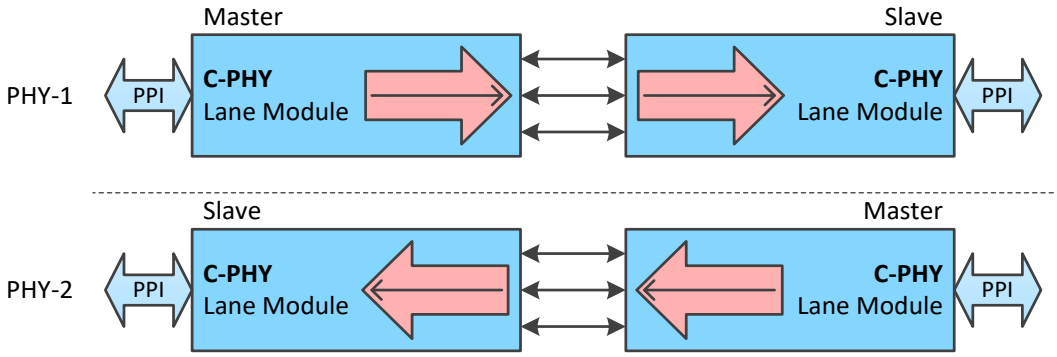


Figure 13 Two Directions Using Two Independent Unidirectional PHYs without LPDT

5.7.2 Bi-Directional Half-Duplex Configurations

Bi-Directional configurations consist of one or more Bi-Directional Lanes. Half-duplex operation enables Bi-Directional traffic across shared interconnect wires. This configuration saves wires compared to the dual-simplex configuration. However, time on the Link is shared between Forward Direction traffic and Reverse Direction traffic and Link Turnaround. The High-Speed symbol rate in the Reverse Direction can be any rate that meets all specifications. There is no requirement for the reverse symbol rate to have any specific relationship with respect to the symbol rate in the Forward Direction. LPDT can have similar rates in the Forward and Reverse Directions. This configuration is especially useful for cases with asymmetrical data traffic.

5.7.2.1 PHY Configurations with a Bi-Directional Single Lane

This configuration includes one of any kind of Bi-Directional Lane. This allows time-multiplexed data traffic in both Forward and Reverse Directions. ***Figure 14*** shows this configuration, with a Lane that supports both High-Speed and escape (without LPDT) communication in both directions. Other possibilities are that only one type of reverse communication is supported, or that LPDT is also included in one or both directions. All these configurations require three interconnect wires.

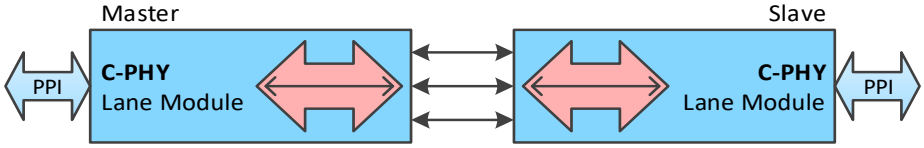


Figure 14 Bi-Directional Single Lane Configuration

5.7.2.2 PHY Configurations with Multiple Lanes

This configuration includes multiple Bi-Directional Lanes. Communication is possible in both the Forward and Reverse Direction for each individual Lane. The maximum available bandwidth scales with the number of Lanes for each direction. The PHY specification does not require all Lanes to be active simultaneously, or even to be operating in the same direction. In fact, the protocol layer controls all Lanes individually. ***Figure*** 15 shows an example configuration with two Lanes. If N is the number of Lanes, this configuration requires $3 \cdot N$ interconnect wires.

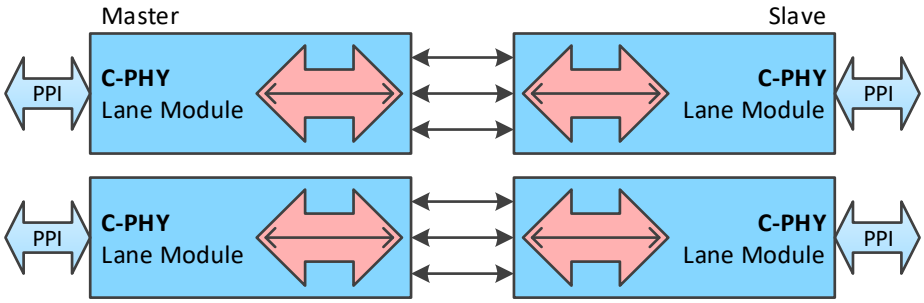


Figure 15 Bi-Directional Multiple Lane Configuration

5.7.3 Mixed Lane Configurations

Instead of using only one Lane type, PHY Configurations may combine different Unidirectional and Bi-directional Lane types. ***Figure 16*** shows an example mixed type configuration with one Bi-directional and one Unidirectional Lane, both without LPDT.

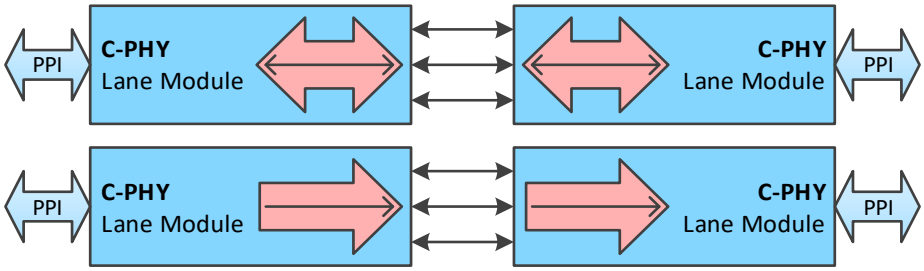


Figure 16 Mixed Type Multiple Lane Configuration

6 Global Operation

This Section specifies operation of C-PHY including signaling types, communication mechanisms, operating modes and coding schemes. Detailed specifications of the required electrical functions can be found in ***Section 9***.

6.1 Transmission Data Structure

During High-Speed, or low-power, transmission, the Link transports payload data provided by the protocol layer to the other side of the Link. This Section specifies the restrictions for the transmitted and received payload data.

6.1.1 Data Units

The minimum payload data unit for the High-Speed mode of transmission shall be one 16-bit word. Data provided to a TX and taken from a RX on any Lane operating in High-Speed mode shall be an integer number of 16-bit words.

The minimum payload data unit for the low-power mode of transmission shall be one byte. Data provided to a TX and taken from a RX on any Lane operating in low-power mode shall be an integer number of bytes.

6.1.2 Bit Order, Serialization, and De-Serialization

For serial transmission, the data shall be serialized in the transmitting PHY and de-serialized in the receiving PHY. For High-Speed data transmission the PHY maps 16-bit words into groups of seven symbols as described in ***Section 6.1.3.3***. Symbol s6 is defined as the most significant symbol of the seven symbol group, and symbol s0 is defined as the least significant symbol. The group of seven symbols shall be transmitted in the sequence of: s0, s1, s2, s3, s4, s5, s6, where s0 shall be transmitted first. Symbol s1 shall be transmitted in the UI immediately following symbol s0, s2 shall be transmitted in the UI immediately following s1, and so on, and s6 shall be transmitted in the UI immediately following s5. Symbol s0 of the next group of seven symbols shall be transmitted in the UI immediately following s6 of the present group.

6.1.3 Encoding, Decoding, Mapping and De-Mapping

C-PHY shall use two layers of coding with serialization and deserialization in between, as illustrated in ***Figure 3***:

1. **Mapping and De-Mapping** – A Mapper converts a 16-bit data unit to be transmitted into a group of seven transmitted symbols. A De-Mapper converts a group of 7 received symbols into a 16-bit data unit.
2. **Serialization and Deserialization** – A parallel to serial converter accepts a group of 7 symbols from the Mapper and presents one symbol at a time to the Symbol Encoder. A serial to parallel converter accepts one symbol at a time from the Symbol Decoder and presents a group of 7 symbols to the De-Mapper.
3. **Encoding and Decoding** – A Symbol Encoder converts one symbol into a Wire State to be sent over the Lane based on the present 3-bit symbol value and the Wire State that was transmitted in the previous UI. A Symbol Decoder computes a received symbol value based on the Wire State received in the present UI and the Wire State received in the previous UI.

6.1.3.1 Wire States

One of six possible High-Speed Wire States shall be driven onto a Lane during a High-Speed unit interval (UI). Each of the Lines of a Lane shall be driven to one of three signal levels: low, middle or high. In some implementations the middle signal level can be the result of the transmitter not driving the signal. Each of the three Lines in a Lane shall be at a different signal level than the other two lines. The six Wire States consist of the six possible permutations of driving the three Lines of a Lane with a different signal level on

each Line. The six Wire States shall be called +x, -x, +y, -y, +z and –z are defined as described in ***Table 3***. Examples of the Wire States are shown in ***Figure 1*** and ***Table 1***.

The Wire States defined as having a positive polarity are: +x, +y and +z. The Wire States defined as having a negative polarity are: -x, -y and -z.

Table 3 Definition of Wire States

Wire State Name	High-Speed State Code Name	Line Signal Levels		
		A	B	C
+x	HS_+X	High	Low	Middle
-x	HS_-X	Low	High	Middle
+y	HS_+Y	Middle	High	Low
-y	HS_-Y	Middle	Low	High
+z	HS_+Z	Low	Middle	High
-z	HS_-Z	High	Middle	Low

6.1.3.2 Symbol Encoding and Decoding

Each symbol shall be represented using a 3-bit number having one of five values: 000, 001, 010, 011 and 100. The symbol values are based on the specific transitions between the Wire States as shown in ***Figure 2***. These transitions are derived from the 3-bit symbol value where each bit defines a particular Wire State change parameter: flip, rotate, and polarity. The flip, rotate and polarity bits affect the Wire State as follows:

- The least significant bit of the 3-bit symbol value is Polarity, which indicates whether the polarity changes state from the previous symbol. When Polarity is “one” then the Wire State transmitted during symbol interval N has a polarity that is opposite that of the Wire State transmitted during symbol interval N-1 (i.e. from positive: +x, +y, +z to negative: -x, -y, -z; or negative to positive); else if Polarity is “zero” then the polarity of the Wire State transmitted during symbol interval N remains the same as the Wire State transmitted during symbol interval N-1.
- The next least significant bit of the 3-bit value is Rotation, which indicates the direction of rotation from the Wire State transmitted during symbol interval N-1 compared to the Wire State transmitted during symbol interval N. When Rotation is one, then the direction of rotation is clockwise; else, when Rotation is zero then the direction of rotation is counterclockwise.
- The most significant bit of the 3-bit symbol value is Flip, which indicates there is only a polarity change in the next symbol but the Wire State will not rotate to a different phase. A Flip causes a transition between states +x and -x, between +y and -y, or between +z and -z. The flip transitions are represented by the blue arrows in ***Figure 2***. When Flip is one then the phase is the same as the previous symbol but the polarity is opposite of the polarity in the previous symbol. Also, when Flip is one then the values of Rotation and Polarity for the same corresponding symbol are ignored, and are both set to zero.

The flip, rotate, and polarity bits are shown for all 30 state transitions in ***Figure 2***.

6.1.3.2.1 Encoding

The symbol encoding shall be performed as described in ***Table 4***, which defines the symbol encoding algorithm. The translation defined in ***Table 4*** converts one 3-bit symbol value into a Wire State to be sent over the Lane based on the present 3-bit symbol value and the Wire State that was transmitted in the previous UI. Every transition in the state diagram of ***Figure 2*** is represented in ***Table 4***. The present Wire State is determined by the previous Wire State and the symbol input value. For example: if the 3-bit symbol value is 011 (no flip, CW rotation, opposite polarity) and the previous Wire State is +y, then the next Wire State is -z.

Table 4 Five Possible Transitions from Previous State to Present State

Symbol Input Value	Previous Wire State, interval N-1						What Happens
	+x	-x	+y	-y	+z	-z	
000	+z	-z	+x	-x	+y	-y	Rotate CCW, polarity is Same
001	-z	+z	-x	+x	-y	+y	Rotate CCW, polarity is Opposite
010	+y	-y	+z	-z	+x	-x	Rotate CW, polarity is Same
011	-y	+y	-z	+z	-x	+x	Rotate CW, polarity is Opposite
1xx	-x	+x	-y	+y	-z	+z	Same phase, polarity is Opposite

Note:

1. *Symbol Input value is: [Tx_Flip, Tx_Rotation, Tx_Polarity]*
2. *Values in the table show the Present Wire State transmitted during interval N, as a function of the Previous Wire State transmitted during interval N-1, and 3-bit Symbol Value.*

An example Transmit Encoder and driver circuit is shown in ***Figure 17***. The 3-bit binary values that represent the previous and present Wire State in ***Figure 17*** exist only to make the example easier to follow. This is to break the process in the example into two steps: Transmit Symbol Encoding Logic and Transmit Pre-driver Control Logic. The Wire State binary values are internal to the Symbol Encoder and Transmitter circuit, so the values that describe the Wire States within these blocks are an implementation choice. For example: the actual logic circuit could use the decoded 6-bit pre-driver value [PU_A, PD_A, PU_B, PD_B, PU_C, PD_C] to define the present Wire State value (using only 6 of the 64 possible values) instead of using the intermediate 3-bit Wire State value.

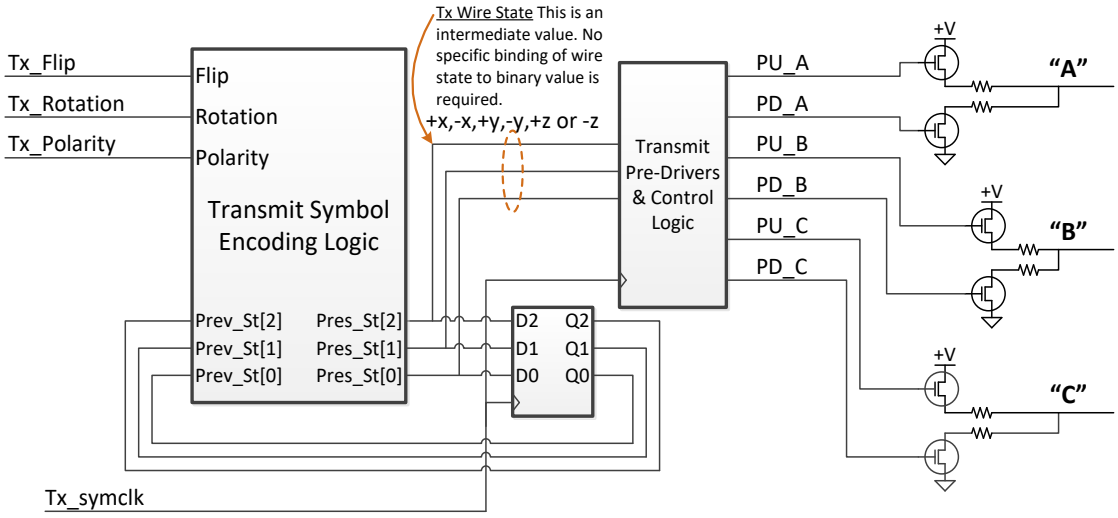


Figure 17 Encoder and Transmitter Example

Table 5 Transmit Pre-Driver Control Logic

Wire State	VA	VB	VC	Driver PU_A	Driver PD_A	Driver PU_B	Driver PD_B	Driver PU_C	Driver PD_C
+x	¾ V	¼ V	½ V	1	0	0	1	0	0
-x	¼ V	¾ V	½ V	0	1	1	0	0	0
+y	½ V	¾ V	¼ V	0	0	1	0	0	1
-y	½ V	¼ V	¾ V	0	0	0	1	1	0
+z	¼ V	½ V	¾ V	0	1	0	0	1	0
-z	¾ V	½ V	¼ V	1	0	0	0	0	1

Note that the Transmit Pre-Driver Control Logic table is shown only to illustrate the logic function of the translation of Wire State to driver control signals. The actual implementation may require carefully designed routing and signal gating to precisely control the skew between the A, B and C wires of the Lane.

6.1.3.2.2 Decoding

The symbol decoding function shall be performed as described in ***Table 6***. Every transition in the state diagram of ***Figure 2*** is represented in ***Table 6***. Note that there are no transitions to the same state (because there is always a Wire State transition at each symbol boundary) so the table shows "n/a" to indicate that these transitions are not applicable. The symbol value shall be determined based on the transition from the previous Wire State (interval N-1) to the present Wire State (interval N). For example: if the previous Wire State is +y which results in "010" at the output of the receivers (prev_Rx_AB=0, prev_Rx_BC=1, prev_Rx_CA=0) and the present Wire State is -z which results in "110" at the output of the receivers (Rx_AB=1, Rx_BC=1, Rx_CA=0), then the symbol value is 011 which represents: no flip, a CW rotation, and polarity change. This symbol value of 011 is located in ***Table 6*** where the "+y" column and the "-z state" row intersect. ("y" toward "z" is CW rotation, and "+" to "-" is a polarity change.)

Table 6 Receive Transition Mapping

Present Wire State [Rx_AB, Rx_BC, Rx_CA] (received during interval N)	Previous Wire State [prev_Rx_AB, prev_Rx_BC, prev_Rx_CA] (Wire State received during interval N-1)					
	+x [100]	-x [011]	+y [010]	-y [101]	+z [001]	-z [110]
+x state [100]	n/a	1xx	000	001	010	011
-x state [011]	1xx	n/a	001	000	011	010
+y state [010]	010	011	n/a	1xx	000	001
-y state [101]	011	010	1xx	n/a	001	000
+z state [001]	000	001	010	011	n/a	1xx
-z state [110]	001	000	011	010	1xx	n/a
	symbol value above = [Rx_Flip, Rx_Rotation, Rx_Polarity]					

An example receiver circuit with symbol decoding is shown below in ***Figure 18***.

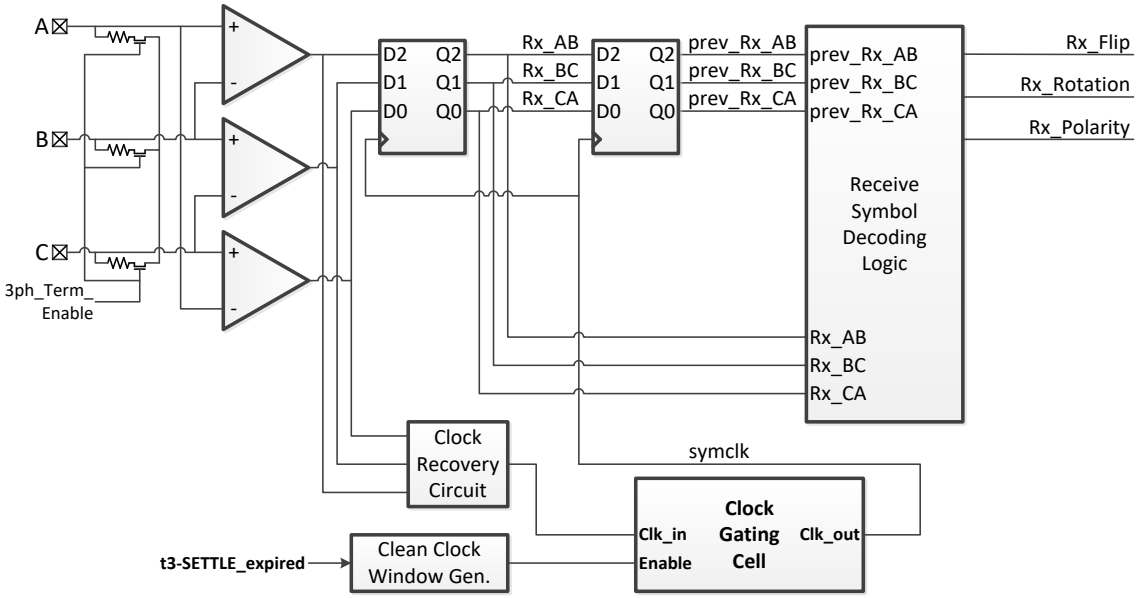


Figure 18 Receiver and Symbol Decoder Example

6.1.3.3 16-to-7 Mapping and 7-to-16 De-Mapping Circuit Implementation

The Mapper and De-Mapper are the outer-most functions in the C-PHY digital coding system. The Mapper is the first function that occurs on the transmit side, and the De-Mapper is the last function performed on the receive side. This order of functions is shown in ***Figure 3***. The Mapper converts a 16-bit word into a group of seven symbols at the transmitting end. At the receiving end, a 7-to-16 De-Mapper converts groups of seven symbols back to 16-bit words. The mapping process is described in ***Section 6.1.3.3.1***, and the de-mapping process is described in ***Section 6.1.3.3.2***.

The 16-to-7 Mapper and 7-to-16 De-Mapper perform a mapping and inverse mapping function between 16-bit input/output values and a group of 7 symbols. The Mapper and De-Mapper shall conform to the process defined in ***Figure 19***.

The group of seven symbols is comprised of seven 3-bit symbol values. Each symbol is comprised of a flip, rotate and polarity bit, so for any particular symbol, n, sn = [Flip[n], Rotation[n], Polarity[n]].

A seven-symbol Mapper output value is defined for every possible 16-bit Mapper input value. However, not all possible seven symbol sequences correspond to a 16-bit mapper input value. The output of the De-Mapper is not defined when an invalid group of seven symbols is presented to the input of the De-Mapper. An invalid group of seven symbols is defined as the collection of symbols having a value of “4” (the state of Flip[6:0]) that does not correspond to one of the 28 mapped regions defined in ***Figure 19***. The De-Mapper output is not specified for these invalid groups of seven symbols, so the De-Mapper output should be an implementation choice when an invalid seven symbol group is presented to the De-Mapper.

Since the mapping and inverse mapping functions are completely feed-forward functions, pipeline registers can be inserted between intermediate stages if necessary to lessen timing constraints in systems that operate at a high symbol rate.

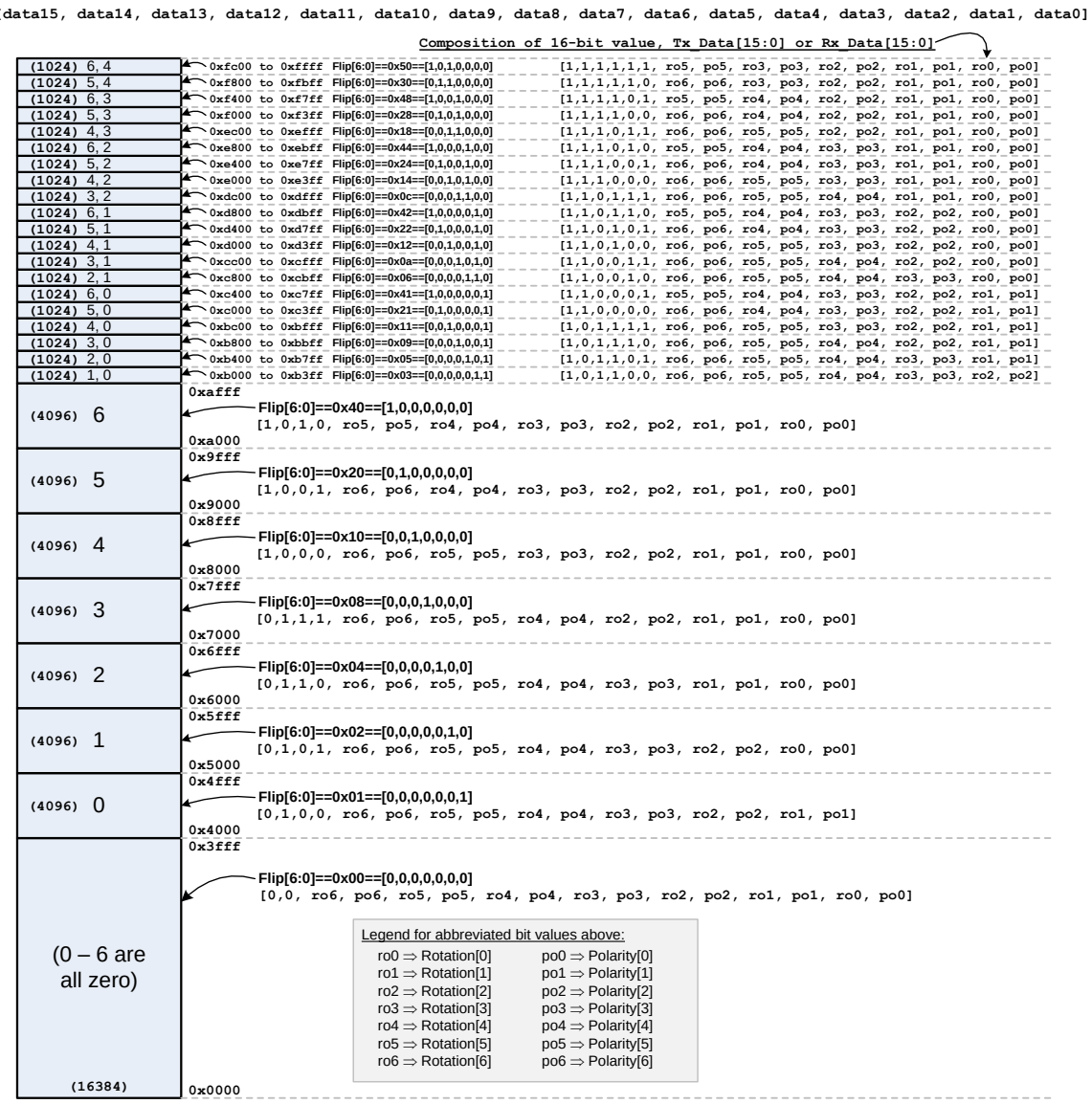


Figure 19 Data Mapping Between Seven Symbols and a 16-Bit Word

Figure 19 specifies the mapping of 16-bit words to groups of seven symbols. The value of the 16-bit word is shown on the left, ranging from 0x0000 at the bottom to 0xffff at the top. The vectors enclosed in square brackets to the right show the correspondence of Rotation and Polarity values to specific bits in the 16-bit word. The seven Flip bits of the seven symbols define one of 28 different regions within the 16-bit range. One region contains 16,384 values, 7 regions contain 4,096 values each, and 20 regions contain 1,024 values each. The use of these varying sized regions simplifies the mapping function.

6.1.3.3.1 Tx 16-Bit to 7-Symbol Mapper

The Mapper shall perform a translation of a 16-bit value to a seven symbol group per the mapping function defined in ***Figure 19***.

A high-level diagram of an example of a 16-to-7 Mapper is shown in ***Figure 20***, and a low-level implementation of the Mapper example is shown in ***Figure 21***.

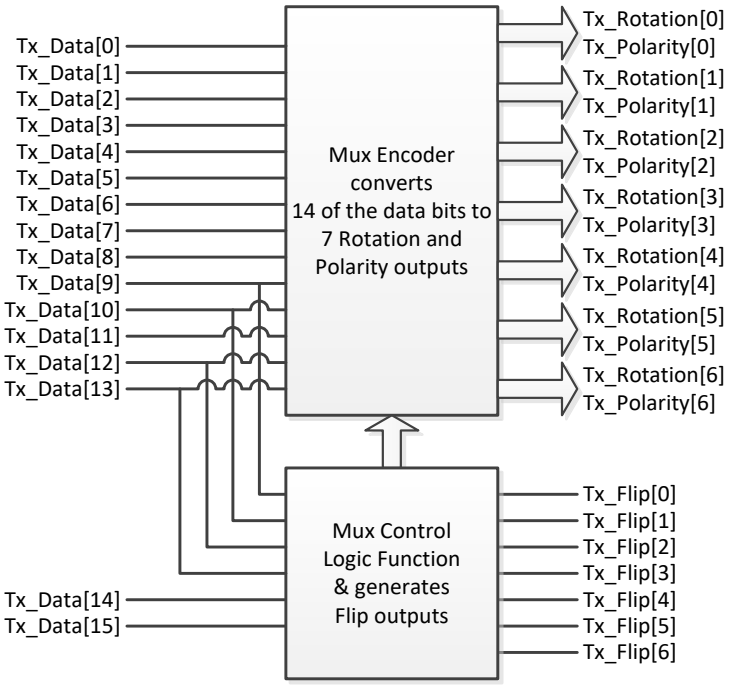


Figure 20 Example, Mapping Circuit Converts 16-bit Word to Seven Symbols

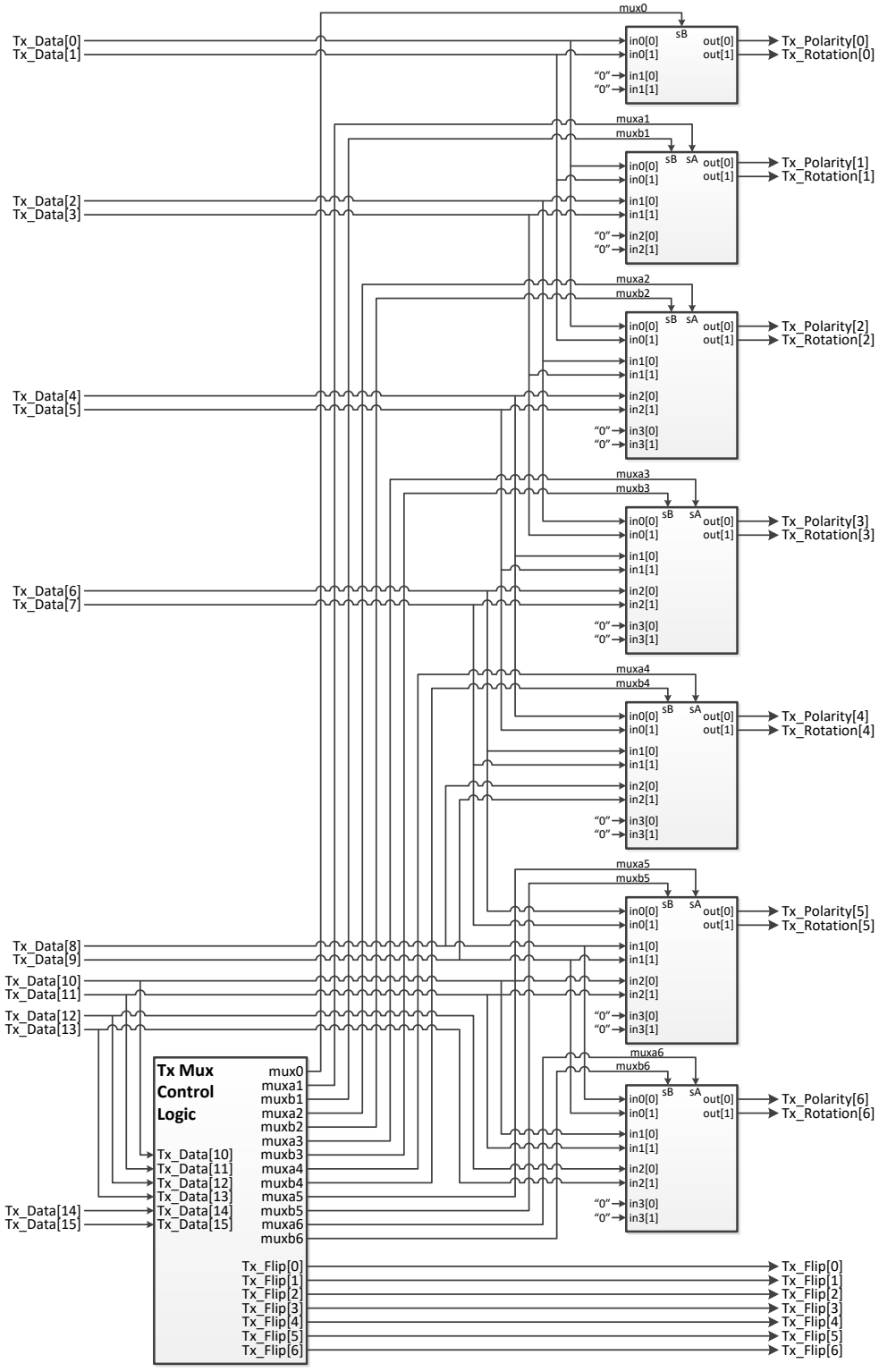


Figure 21 Example, Detailed Logic Diagram of 16-bit Word to 7-Symbol Mapping Circuit

Figure 21 shows the low-level circuit of the 16-to-7 Mapper example in the block diagram of ***Figure 20*** that performs the conversion of a 16-bit data word to be transmitted into seven consecutive symbols. The logic function of the "Tx Mux Control Logic" in the ***Figure 21*** example appears below in ***Table 7***.

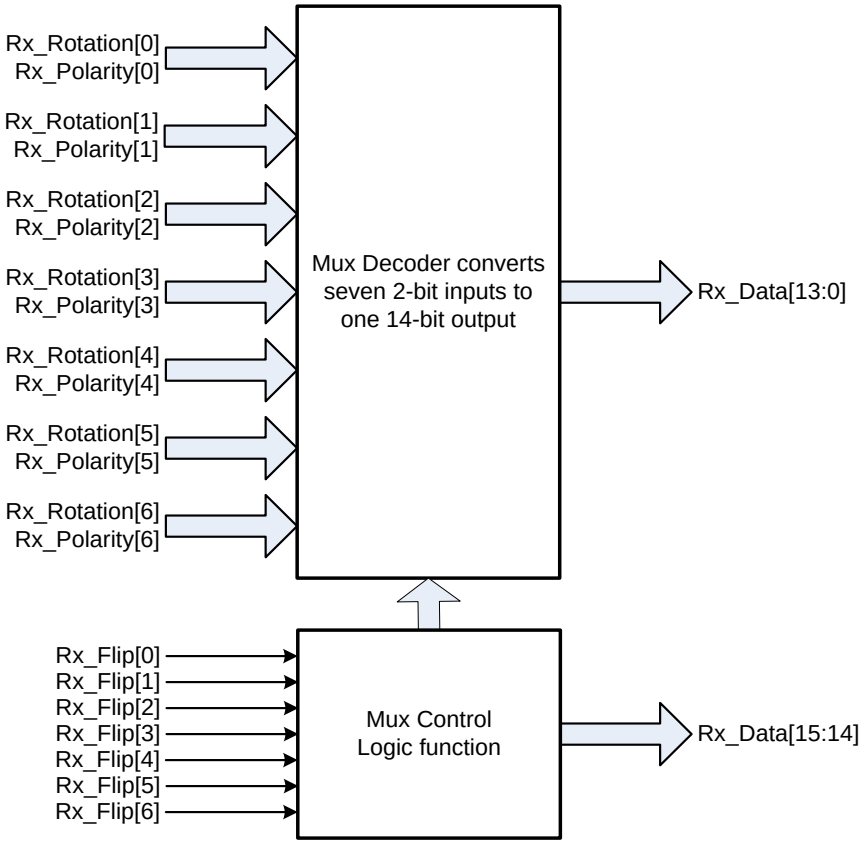
Table 7 Truth Table of the "Tx Mux Control Logic" in Figure 21

Tx_Data [15:10]	muxb6	muxa6	muxb5	muxa5	muxb4	muxa4	muxb3	muxa3	muxb2	muxa2	muxb1	muxa1	mux0	Flip[6:0]
0x00 – 0x0f	1	0	1	0	1	0	1	0	1	0	0	1	0	0x00
0x10 – 0x13	0	1	0	1	0	1	0	1	0	1	0	0	1	0x01
0x14 – 0x17	0	1	0	1	0	1	0	1	0	1	1	0	0	0x02
0x18 – 0x1b	0	1	0	1	0	1	0	1	1	1	0	1	0	0x04
0x1c – 0x1f	0	1	0	1	0	1	1	1	1	0	0	1	0	0x08
0x20 – 0x23	0	1	0	1	1	1	1	0	1	0	0	1	0	0x10
0x24 – 0x27	0	1	1	1	1	0	1	0	1	0	0	1	0	0x20
0x28 – 0x2b	1	1	1	0	1	0	1	0	1	0	0	1	0	0x40
0x2c	0	0	0	0	0	0	0	0	0	0	1	0	1	0x03
0x2d	0	0	0	0	0	0	0	0	1	1	0	0	1	0x05
0x2e	0	0	0	0	0	0	1	1	0	1	0	0	1	0x09
0x2f	0	0	0	0	1	1	0	1	0	1	0	0	1	0x11
0x30	0	0	1	1	0	1	0	1	0	1	0	0	1	0x21
0x31	1	1	0	1	0	1	0	1	0	1	0	0	1	0x41
0x32	0	0	0	0	0	0	0	0	1	1	1	0	0	0x06
0x33	0	0	0	0	0	0	1	1	0	1	1	0	0	0x0a
0x34	0	0	0	0	1	1	0	1	0	1	1	0	0	0x12
0x35	0	0	1	1	0	1	0	1	0	1	1	0	0	0x22
0x36	1	1	0	1	0	1	0	1	0	1	1	0	0	0x42
0x37	0	0	0	0	0	0	1	1	1	1	0	1	0	0x0c
0x38	0	0	0	0	1	1	0	1	1	1	0	1	0	0x14
0x39	0	0	1	1	0	1	0	1	1	1	0	1	0	0x24
0x3a	1	1	0	1	0	1	0	1	1	1	0	1	0	0x44
0x3b	0	0	0	0	1	1	1	1	1	0	0	1	0	0x18
0x3c	0	0	1	1	0	1	1	1	1	0	0	1	0	0x28
0x3d	1	1	0	1	0	1	1	1	1	0	0	1	0	0x48
0x3e	0	0	1	1	1	1	1	0	1	0	0	1	0	0x30
0x3f	1	1	0	1	1	1	1	0	1	0	0	1	0	0x50

6.1.3.3.2 Rx 7-Symbol to 16-Bit De-Mapper

The De-Mapper shall perform a translation of a seven symbol group to a 16-bit value per the mapping function defined in ***Figure 19***.

587 A high-level diagram of an example of a 7-to-16 De-Mapping circuit is shown in ***Figure 22***, and a detailed
588 low-level implementation of the De-Mapper example is shown in ***Figure 23***.



589

Figure 22 Example, De-Mapping Circuit Converts Seven Symbols to a 16-Bit Word

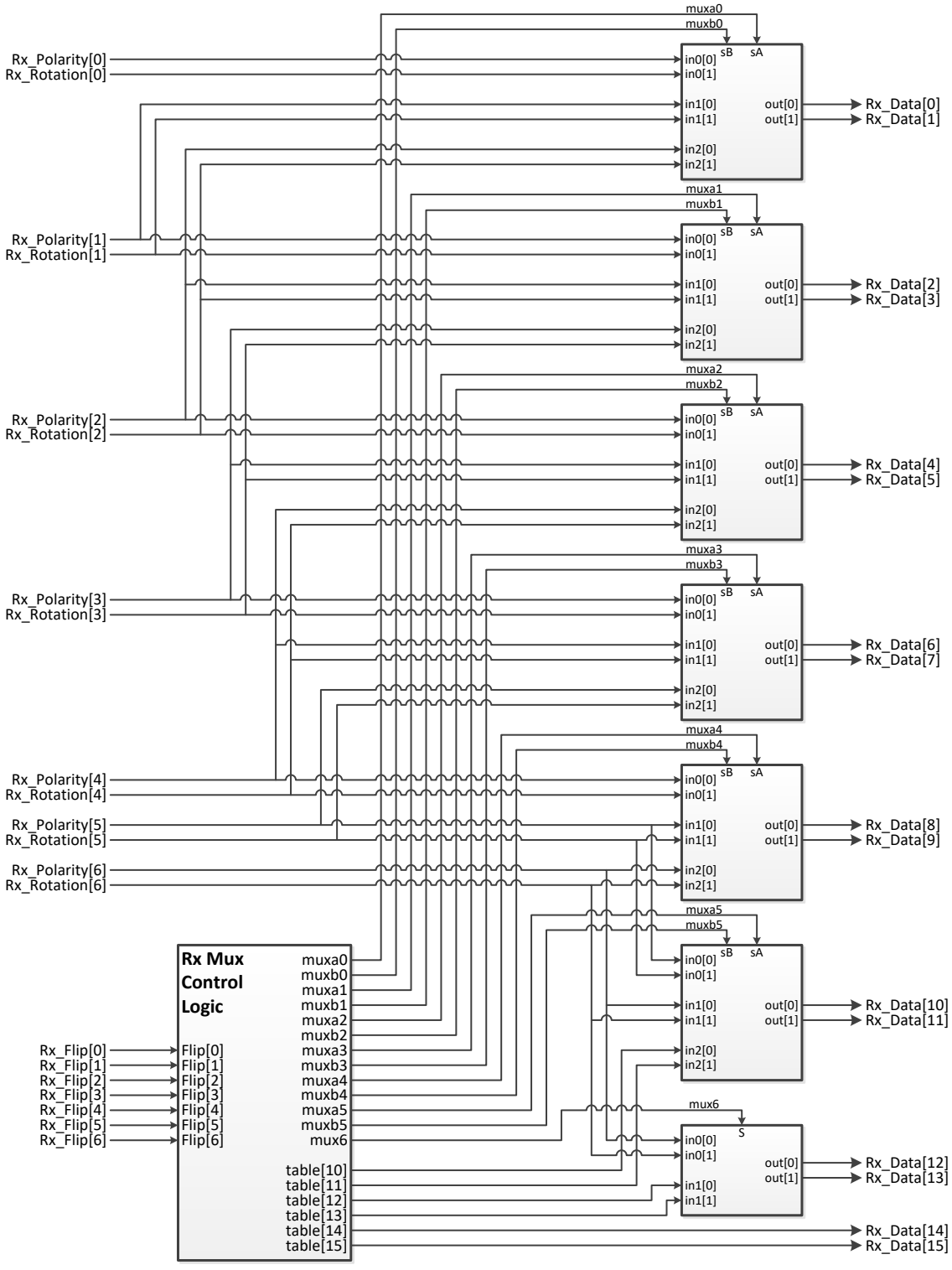


Figure 23 Detailed Logic Diagram Example of a 7-Symbol to 16-Bit Word De-Mapper

The logic function of the "Rx Mux Control Logic" in ***Figure 23*** is shown below in ***Table 8***.

592 **Table 8 Truth Table of the “Rx Mux Control Logic” in Figure 23**

Rx_Flip[6:0]	muxa6	muxb5	muxa5	muxb4	muxa4	muxb3	muxa3	muxb2	muxa2	muxb1	muxa1	muxb0	muxa0	table[15:14]	table[13:12]	table[11:10]
0x00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x	x
0x01	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	x
0x02	1	0	1	0	1	0	1	0	1	0	1	0	0	1	1	x
0x04	1	0	1	0	1	0	1	0	1	0	0	0	0	1	2	x
0x08	1	0	1	0	1	0	1	0	0	0	0	0	0	1	3	x
0x10	1	0	1	0	1	0	0	0	0	0	0	0	0	2	0	x
0x20	1	0	1	0	0	0	0	0	0	0	0	0	0	2	1	x
0x40	1	0	0	0	0	0	0	0	0	0	0	0	0	2	2	x
0x03	1	1	0	1	0	1	0	1	0	1	0	1	0	2	3	0
0x05	1	1	0	1	0	1	0	1	0	1	0	0	1	2	3	1
0x09	1	1	0	1	0	1	0	1	0	0	1	0	1	2	3	2
0x11	1	1	0	1	0	1	0	0	1	0	1	0	1	2	3	3
0x21	1	1	0	1	0	0	1	0	1	0	1	0	1	3	0	0
0x41	1	1	0	0	1	0	1	0	1	0	1	0	1	3	0	1
0x06	1	1	0	1	0	1	0	1	0	1	0	0	0	3	0	2
0x0a	1	1	0	1	0	1	0	1	0	0	1	0	0	3	0	3
0x12	1	1	0	1	0	1	0	0	1	0	1	0	0	3	1	0
0x22	1	1	0	1	0	0	1	0	1	0	1	0	0	3	1	1
0x42	1	1	0	0	1	0	1	0	1	0	1	0	0	3	1	2
0x0c	1	1	0	1	0	1	0	1	0	0	0	0	0	3	1	3
0x14	1	1	0	1	0	1	0	0	1	0	0	0	0	3	2	0
0x24	1	1	0	1	0	0	1	0	1	0	0	0	0	3	2	1
0x44	1	1	0	0	1	0	1	0	1	0	0	0	0	3	2	2
0x18	1	1	0	1	0	1	0	0	0	0	0	0	0	3	2	3
0x28	1	1	0	1	0	0	1	0	0	0	0	0	0	3	3	0
0x48	1	1	0	0	1	0	1	0	0	0	0	0	0	3	3	1
0x30	1	1	0	1	0	0	0	0	0	0	0	0	0	3	3	2
0x50	1	1	0	0	1	0	0	0	0	0	0	0	0	3	3	3

6.1.4 Data Buffering

593 Data transmission takes place on protocol request. As soon as communication starts, the protocol layer at the
594 transmit side shall provide valid data as long as it does not stop its transmission request. For Lanes that use
595 Line coding, control symbols can also be inserted into the transmission. The protocol layer on the receive
596 side shall take the data as soon as delivered by the receiving PHY. The signaling concept, and therefore the
597 PHY protocol handshake, does not allow data throttling. Any data buffering for this purpose shall be inside
598 the protocol layer.

6.2 Lane States and Line Levels

Transmitter functions determine the Lane state by driving certain Line levels. The system shall operate in one of two different modes which are defined by the combinations of Line levels used in each mode. The primary mode uses a combination of HS differential signaling and LP mode single-ended high voltage signaling. Alternate Low-Power (ALP) mode uses solely differential signaling with a special state where the signals can cease toggling and collapse to zero.

6.2.1 HS and LP Mode Line States and Line Levels

In normal operation with HS and LP modes, either a HS-TX or a LP-TX is driving a Lane. A HS-TX always drives the Lane differentially. The three LP-TX's drive the three Lines of a Lane independently and single-ended. This results in six possible High-Speed Lane states and four possible low-power Lane states: LP-000, LP-001, LP-100 and LP-111. The High-Speed Lane states are: +x, -x, +y, -y, +z and -z. The interpretation of low-power Lane states depends on the mode of operation. The LP-Receivers shall always interpret any of the six High-Speed states as LP-000.

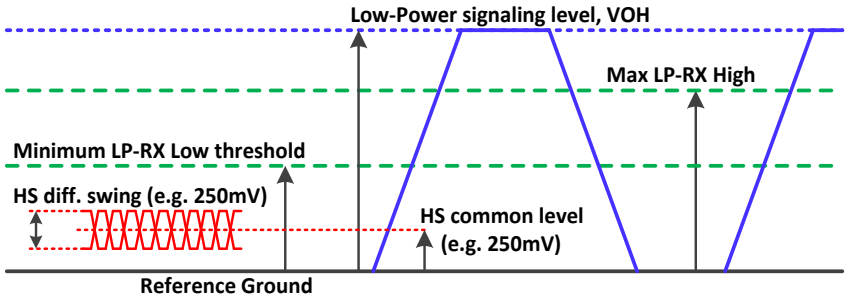


Figure 24 Line Levels in HS and LP Modes

The Stop state has a very exclusive and central function. If the Line levels show a Stop state for the minimum required time, then the PHY state machine shall return to the Stop state regardless of the previous state. This can be in RX or TX mode, depending on the most recent operating direction. ***Table 9*** lists all the states that can appear on a Lane during normal operation. Detailed specifications of electrical levels can be found in ***Section 9***.

All LP state periods shall be at least t_{LPX} in duration. State transitions shall be smooth and exclude glitch effects. A clock signal can be reconstructed by exclusive-ORing the A and C lines. Ideally, the reconstructed clock duration is at least $2 \cdot t_{LPX}$, but may have a duty cycle other than 50% due to signal slope and trip levels effects.

620 **Table 9 HS and LP Mode Lane State Descriptions**

State Code	Line Voltage Levels			High-Speed	Low-Power	
	A Line	B Line	C Line	Burst Mode	Control Mode	Escape Mode
HS_+X	HS High	HS Low	HS Mid	+x state	N/A, Note 1	N/A, Note 1
HS+-X	HS Low	HS High	HS Mid	-x state	N/A, Note 1	N/A, Note 1
HS_+Y	HS Mid	HS High	HS Low	+y state	N/A, Note 1	N/A, Note 1
HS_-Y	HS Mid	HS Low	HS High	-y state	N/A, Note 1	N/A, Note 1
HS_+Z	HS Low	HS Mid	HS High	+z state	N/A, Note 1	N/A, Note 1
HS_-Z	HS High	HS Mid	HS Low	-z state	N/A, Note 1	N/A, Note 1
LP-000	LP Low	LP Low	LP Low	N/A	Bridge	Space
LP-001	LP Low	LP Low	LP High	N/A	HS-Rqst	Mark-0
LP-100	LP High	LP Low	LP Low	N/A	LP-Rqst	Mark-1
LP-111	LP High	LP High	LP High	N/A	Stop	N/A, Note 2

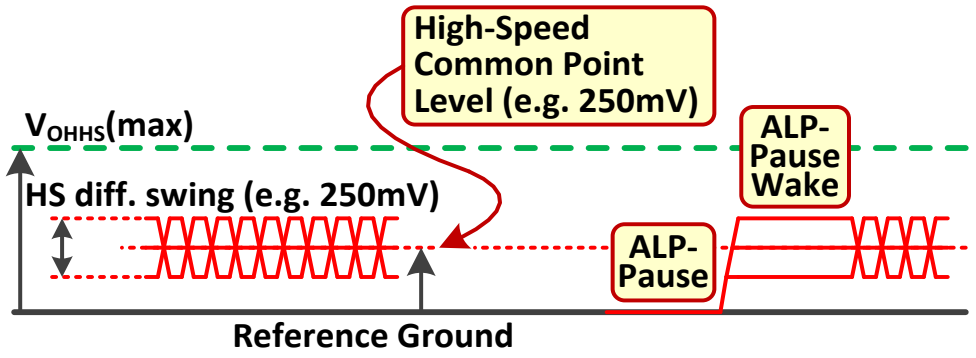
Note:

1. *During High-Speed transmission the low-power Receivers observe LP-000 on the Lines.*
2. *If LP-111 occurs during Escape Mode, the Lane returns to Stop state (Control Mode LP-111).*
3. *Only 4 of the 8 possible low-power states are defined, because the C-PHY low-power states correspond exactly to the D-PHY low-power states so that D-PHY low-power mode can be duplicated.*

6.2.2 ALP Mode Line States and Line Levels

621 ALP Mode replaces the high-voltage Line levels by the transmission of unique code words that are used only
622 for Lane signaling events. These unique codes are never produced by the 3-Phase mapping function described
623 in ***Section 6.1.3***, so there is never ambiguity in the interpretation of these codes at the receiver. The High-
624 Speed Lane states are: +x, -x, +y, -y, +z, -z.

625 In ALP mode the transmission of certain unique code words perform the exact same functions as the
626 transmission of LP pulse sequences in ALP mode. The Stop state has a special sequence consisting of a
627 unique code word (Post2) followed by setting the voltage of all three Lines of a Lane to the same value. This
628 line state is called ALP-Pause, which can be further defined as either ALP-Pause Stop or ALP-Pause ULPS.
629 ALP-Pause Stop and ALP-Pause ULPS are defined by the following relationships of the Line levels:
630 $V_A = V_B = V_C$, and $V_{OD\_AB} = V_{OD\_BC} = V_{OD\_CA} = 0$. To minimize power consumption while Lane activity has
631 ceased during one of the ALP-Pause states, a special receiver with an offset input threshold voltage is used
632 to detect the difference in differential levels between the ALP-Pause state ($V_{OD} = 0$) and ALP-Pause Wake
633 state ($V_{OD} = |V_{OD}|$ Strong). The Line levels of high-speed signaling, of the ALP-Pause and the ALP-Pause
634 Wake states, and are illustrated in ***Figure 25***.



635 **Figure 25 Line Levels in ALP Mode**

Table 10 ALP Mode Lane State Descriptions

State Code	Line Voltage Levels			High-Speed Burst Mode	ALP-Pause or ALP-Pause Wake
	A Line	B Line	C Line		
HS_+X	HS High	HS Low	HS Mid	+x state	ALP-Pause Wake, Note 1
HS_-X	HS Low	HS High	HS Mid	-x state	Alternative ALP-Pause Wake, Note 2
HS_+Y	HS Mid	HS High	HS Low	+y state	Alternative ALP-Pause Wake, Note 2
HS_-Y	HS Mid	HS Low	HS High	-y state	Alternative ALP-Pause Wake, Note 2
HS_+Z	HS Low	HS Mid	HS High	+z state	Alternative ALP-Pause Wake, Note 2
HS_-Z	HS High	HS Mid	HS Low	-z state	Alternative ALP-Pause Wake, Note 2
ALP-Pause	$V_A=V_B=V_C$	$V_A=V_B=V_C$	$V_A=V_B=V_C$	N/A	ALP-Pause, $V_A=V_B=V_C$

Note:

1. *The +x state is the high-speed line state detected in the receiver to determine that the ALP-Pause Wake condition is being driven by the transmitter.*
2. *The transmitter should be programmed to drive any of the six high-speed line states during ALP-Pause Wake so that permutations of the Lines in the Lane interconnect can be resolved, and the receiver can detect ALP-Pause Wake by monitoring the state of only the A and B lines.*

6.3 Operating Modes: Control, High-Speed, and Escape

6.3.1 HS and LP Operating Modes

During normal operation in HS and LP modes, a Lane will be either in Control or High-Speed mode. High-Speed data transmission happens in bursts, and starts from and ends at a Stop state (LP-111), which is by definition in Control mode. The Lane is only in High-Speed mode during data bursts. The sequence to enter High-Speed mode is: LP-111, LP-001, LP-000 at which point the Lane remains in High-Speed mode until a LP-111 is received. The Escape Mode can only be entered via a request within Control mode. The Lane shall always exit Escape Mode and return to control mode after detection of a Stop state. If not in High-Speed or Escape Mode, the Lane shall stay in Control mode. The Stop state serves as a general standby state and may last for any period of time > t_{LPX}. Possible events starting from the Stop state are High-Speed data transmission request (LP-111, LP-001, LP-000), Escape Mode request (LP-111, LP-100, LP-000, LP-001, LP-000) or Turnaround request (LP-111, LP-100, LP-000, LP-100, LP-000).

6.3.2 ALP Operating Modes

During normal operation in ALP Mode, a Lane will be in the ALP-Pause state, ALP-Pause Wake state, or High-Speed data transmission mode. High-Speed data transmission starts from and ends at the ALP-Pause state. The Lane is only in High-Speed mode during data bursts and during ALP signaling. The sequence to enter High-Speed mode for data bursts or ALP signaling is: ALP-Pause Wake followed by a preamble consisting of all 1's or all 3's symbols. This is illustrated in ***Figure 30***. Code words sent after the preamble will define the specific purpose of the burst. All of the codes used for ALP signaling perform identical functions as corresponding LP mode states or Escape mode pulse sequences. The Lane shall always exit the High-Speed mode and return to ALP-Pause state after detection of a Post2 code (described later). If not in High-Speed mode, the Lane shall stay in the ALP-Pause state. The ALP-Pause state serves as a general standby state and may last for any period of time > $t_{3\text{-}ALP\text{-}PAUSE(TX)}$. Any event is possible starting from one of the ALP-Pause Stop or ALP-Pause ULPS states. Note that the ALP mode equivalent of the Turnaround procedure is not yet defined.

6.4 High-Speed Data Transmission

High-Speed data transmission occurs in bursts. To aid receiver synchronization, data bursts shall be extended on the transmitter side with a Preamble and Post sequence. The effects of the High-Speed transmitter shall be eliminated on the receiver side by detecting certain events designed to be detected by the receiver so it can ignore the ambiguous operating states during the mode transitions. These Preamble and Post sequences can therefore only be observed on the transmission lines.

Transmission starts from, and ends with, a Stop state. During the intermediate time between bursts a Lane shall remain in the Stop state, unless a Turnaround or escape request is presented on the Lane. During a HS Data Burst the Lane shall be in High-Speed mode, and will be constantly toggling per the encoding rules, thus providing High-Speed data timing to the receiver.

6.4.1 Burst Payload Data

The payload data of a burst shall always represent an integer number of payload data words with a minimum length of one word. Note that for short bursts the Start and End overhead consumes much more time than the actual transfer of the payload data. There is no maximum number of words implied by the PHY. However, in the PHY there is no autonomous way of error recovery during a HS data burst. The practical symbol error rate is nearly zero. It is important to consider for every individual protocol what the best choice is for maximum burst length.

6.4.2 Start-of-Transmission

After a Transmit request, a Lane leaves the Stop state and prepares for High-Speed mode by means of a Start-of-Transmission (SoT) procedure. ***Table 11*** describes the sequence of events on TX and RX side.

Table 11 Start-of-Transmission Sequence

Tx Side	Rx Side
Drives Stop state (LP-111)	Observes Stop state
Drives HS-Rqst state (LP-001) for time t_{LPX}	Observes transition from LP-111 to LP-001 on the lines
Drives Bridge state (LP-000) for time $t_{3\text{-PREPARE}}$	Observes transition from LP-001 to LP-000 on the lines, enables Line termination after time $t_{3\text{-TERM-EN}}$
Enables High-Speed driver and disables low-power drivers simultaneously.	–
Drives Preamble sequence for time $t_{3\text{-PREAMBLE}}$	Enables HS-RX and waits for timer $t_{3\text{-SETTLE}}$ to expire in order to neglect transition effects
–	Starts looking for the Sync Word sequence
Inserts the Sync Word Sequence	–
–	Synchronizes upon recognition of Sync Word Sequence
Continues to transmit High-Speed payload data	–
–	Receives payload data

6.4.3 End-of-Transmission

At the end of a Data Burst, a Lane leaves High-Speed Transmission mode and enters the Stop state by means of an End-of-Transmission (EoT) procedure. ***Table 12*** shows a possible sequence of events during the EoT procedure. EoT processing may be performed by the protocol layer or by the C-PHY.

Table 12 End-of-Transmission Sequence

Tx Side	Rx Side
Completes Transmission of payload data	Receives payload data
Transmits the Post Sequence immediately after last payload data bit for time $t_{3\text{-POST}}$	Detect Post Sequence to determine last valid Data word and skip the remainder of the Post sequence
Disables the HS-TX, enables the LP-TX, and drives Stop state (LP-111) for time $t_{HS\text{-EXIT}}$	Detects the lines leaving LP-000 state and entering Stop state (LP-111) and disables Termination

6.4.4 HS Data Transmission Burst

Figure 26 shows the sequence of events during the transmission of a Data Burst. Transmission can be started and ended independently for any Lane by the protocol layer. However, for most applications the Lanes will start synchronously but may end at different times due to an unequal amount of transmitted bytes per Lane. The handshake with the protocol layer is described in Annex A.

Beginning from the Stop state, LP-111, the signals transition to LP-001 and then LP-000 to signal that High-Speed data transmission will begin soon. At the end of $t_{3\text{-PREPARE}}$ the low-power drivers are disabled and the High-Speed drivers are enabled simultaneously. The TX state machine should transmit the same Wire State as the first High-Speed Wire State at the beginning of $t_{3\text{-PREBEGIN}}$ for each Data Burst in order for the test equipment to measure $t_{3\text{-PREBEGIN}}$ value consistently. This does not correspond to any particular symbol value because there is no previous HS Wire State before it. It is likely that the first few Wire States of the $t_{3\text{-PREBEGIN}}$ interval will not be seen at the High-Speed receiver. This is because there will be some delay for the High-Speed drivers to reach their required signal levels at the beginning of $t_{3\text{-PREBEGIN}}$, and also the High-Speed receivers will be enabled and at some point start producing outputs toward the end of $t_{3\text{-SETTLE}}$. The receive circuitry shall be enabled toward the end of $t_{3\text{-SETTLE}}$ when it is safe for it to reliably decode the “3” symbols during $t_{3\text{-PRE-BEGIN}}$. It is not guaranteed at exactly which symbol clock generation and symbol decoding will begin at the end of $t_{3\text{-SETTLE}}$. The $t_{3\text{-PREBEGIN}}$ field may often consist of multiple groups of seven “3” symbols to provide a sufficient number of clocks to the upper layer protocol to initialize any pipeline stages prior to receiving data. The length of $t_{3\text{-PREBEGIN}}$ is a programmable value set in the transmitter.

The transmitter may output a programmable sequence during $t_{3\text{-PROGSEQ}}$ of the preamble, if it is enabled using a programmable sequence enable bit such as the MSB of the control register described in ***Section 12.5.3***. The symbol values transmitted in the programmable sequence, or whether the programmable sequence is used at all, is a choice of the system designer.

Figure 26 shows examples of the preamble with and without the programmable sequence. Seven symbols of value “3” are sent during $t_{3\text{-PREEND}}$ just prior to sending the Sync Word.

The length of the preamble output by the transmitter should be an adjustable value and it is described in a note below ***Table 24***.

The minimum required total length of the preamble, $t_{3\text{-PREAMBLE}}$, at the receiver depends upon the specific implementation of the C-PHY receiver and protocol receiver circuitry. Therefore, the minimum length of $t_{3\text{-PREAMBLE}}$ shall be specified in the data sheet of the receiving device, and this minimum total preamble length is the appropriate parameter to use to evaluate the minimum performance of the receiver.

The Sync Word precisely identifies the beginning of the Packet Data, and also identifies the timing alignment of word boundaries in the Packet Data. The Sync Word contains a sequence of five “4” symbols which does not occur in any sequence of symbols generated by the Mapper. The Sync Word may also be transmitted later in the burst to mark the beginning of redundant Packet Headers transmitted by the upper layer protocol.

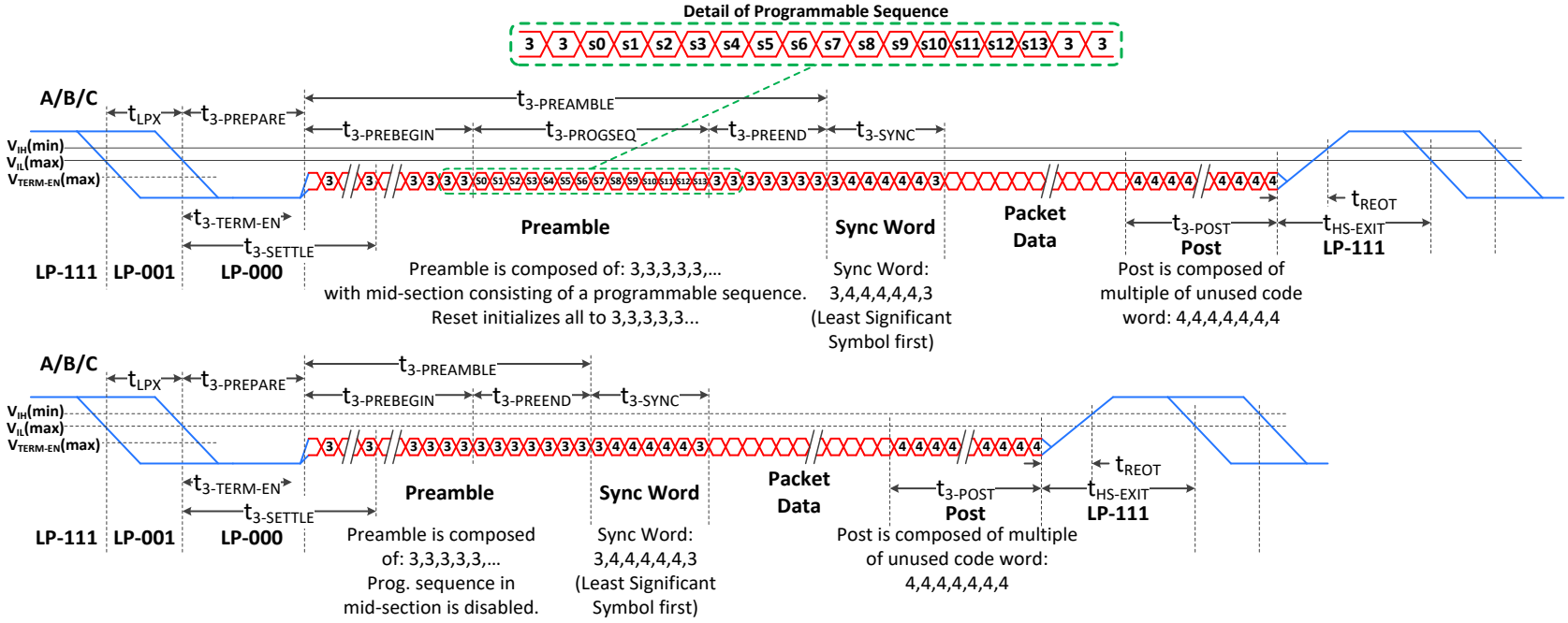
The end of Packet Data is identified by a unique sequence of “4” symbols in $t_{3\text{-POST}}$. The receiver identifies the end of Packet Data when it detects a sequence of seven consecutive “4” symbols. The Post field may often consist of multiple groups of seven “4” symbols to provide a sufficient number of clocks to the upper layer protocol to clear out any pipeline stages that may contain received data. The length of the Post field is

a programmable value set in the transmitter, for example: the post length field of the register described in ***Section 12.5.4***.

At the end of $t_{3\text{-}POST}$ the High-Speed drivers are disabled and the low-power drivers are enabled simultaneously, and all three signals of the Lane are driven high together to LP-111, the Stop state.

The length of Post output by the transmitter should be an adjustable value and it is described in a note below ***Table 24***.

The minimum required length of Post, $t_{3\text{-}POST}$, at the receiver depends upon the specific implementation of the C-PHY receiver and protocol receiver circuitry. Therefore, the minimum length of $t_{3\text{-}POST}$ shall be specified in the data sheet of the receiving device, and this minimum Post length is the appropriate parameter to use to evaluate the minimum performance of the receiver.



729

Figure 26 High-Speed Data Transmission in Burst

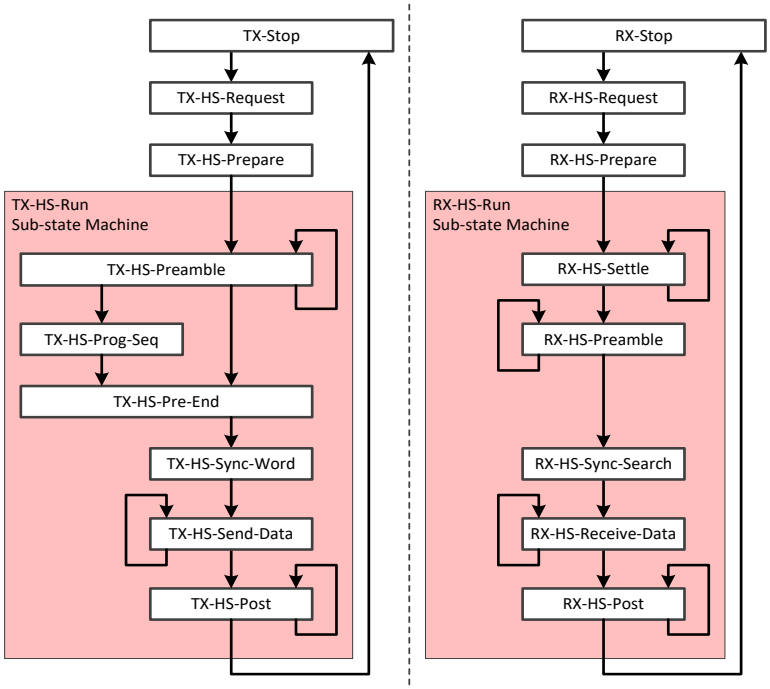


Figure 27 TX and RX State Machines for High-Speed Data Transmission

Table 13 High-Speed Data Transmission State Machine Description

State	Line Condition or Line State	Exit State	Exit Conditions
TX-Stop	Transmit LP-111	TX-HS-Request	On request of Protocol for High-Speed Transmission
TX-HS-Request	Transmit LP-001	TX-HS-Prepare	End of timed interval t_{LPX}
TX-HS-Prepare	Transmit LP-000	TX-HS-Preamble	End of interval $t_{3\text{-}PREPARE}$
TX-HS-Preamble	Preamble 3,3,3,3…	TX-HS-Prog-Seq	End of Preamble & Prog-Seq selected
		TX-HS-Pre-End	End of Preamble & Prog-Seq not selected
		TX-HS-Preamble	Preamble words remaining count > 0
TX-HS-Prog-Seq	Prog-Seq	TX-HS-Pre-End	End of Prog-Seq
TX-HS-Pre-End	Pre-End	TX-HS-Sync-Word	End of Pre-End
TX-HS-Sync-Word	Sync Word	TX-HS-Send-Data	End of Sync-Word
TX-HS-Send-Data	Packet Data	TX-HS-Post	End of Packet Data
		TX-HS-Send-Data	Packet Data available to send
TX-HS-Post	Post 4,4,4,4…	TX-Stop	Last word of Post sent
		TX-HS-Post	Post words remaining count > 0

Table 14 High-Speed Data Reception State Machine Description

State	Line Condition or Line State	Exit State	Exit Conditions
RX-Stop	LP-111	RX-HS-Request	Line transition to LP-001 detected
RX-HS-Request	LP-001	RX-HS-Prepare	Line transition to LP-000 detected
PX-HS-Prepare	LP-000	RX-HS-Settle	$t_{3\text{-TERM-EN}}$ expired
RX-HS-Settle	LP-000 or Preamble 3,3,3,3…	RX-HS-Prog-Seq	$t_{3\text{-SETTLE}}$ expired & Prog-Seq selected
		RX-HS-Sync-Search	$t_{3\text{-SETTLE}}$ expired & Prog-Seq not selected
		RX-HS-Settle	$t_{3\text{-SETTLE}}$ time not expired
RX-HS-Preamble	Any part of Preamble	RX-HS-Sync-Search	High-Speed circuit initialization complete
		RX-HS-Preamble	High-Speed circuit initialization not complete
RX-HS-Sync-Search	Preamble or Sync Word	RX-HS-Receive-Data	Sync Word detected
RX-HS-Receive-Data	Packet Data	RX-HS-Post	First word of Post detected
		RX-HS-Receive-Data	Post not yet detected
RX-HS-Post	Post 4,4,4,4…	RX-Stop	Line transition to LP-111 detected
		RX-HS-Post	Receiving Post, waiting for LP-111

6.4.4.1 Sync Word for Packet Header Resynchronization

A given C-PHY Wire State Link error causes two symbol decoding errors. If the erroneous Wire State also matches the Wire State immediately preceding and/or following it, then the C-PHY receiver's clock and data recovery circuit will miss one or two symbol clocks, thereby causing loss of 7-symbol word alignment. If left uncorrected, this loss of word alignment will likely cause extensive de-mapping errors in the rest of the received packet payload.

In order to facilitate restoration of the transmitted word alignment, this specification defines an unmapped, 7-symbol Sync Word which may be inserted by the C-PHY transmitter at a point directed by the protocol layer via the PPI. The 7-symbol value of the Sync Word is [344444x], which is the same value used for the Sync Word as described in ***Section 6.4.4***.

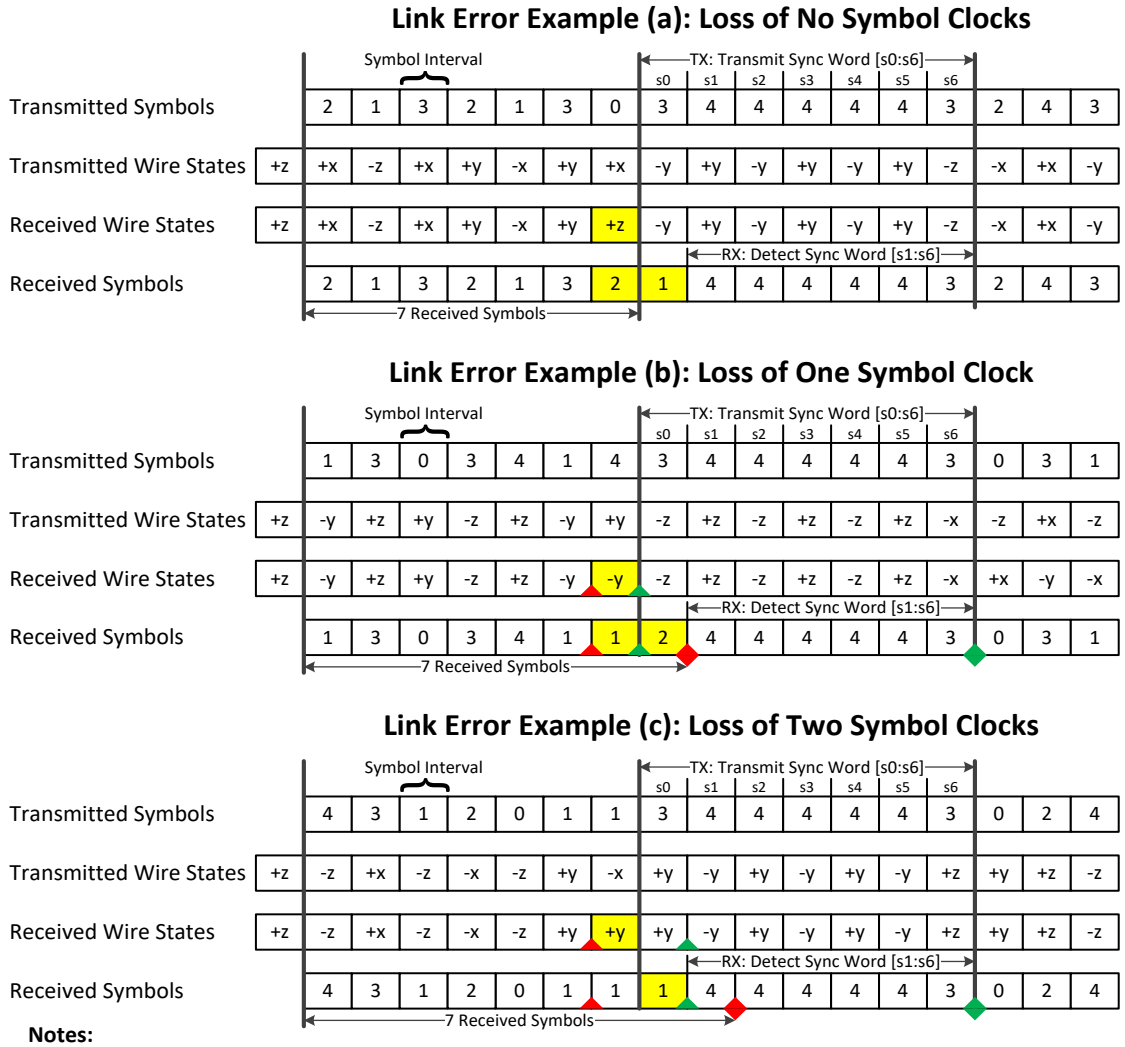
The C-PHY transmitter directly encodes the Sync Word into a series of seven Wire States, bypassing the symbol mapping block which normally processes 16-bit packet payload words.

The C-PHY receiver recognizes a Sync Word by detecting any sequence of five consecutive {4} symbols immediately followed by a {3} symbol at the output of the symbol decoder; i.e. detection of the least significant symbol of the Sync Word is not required. Ignoring the latter symbol actually makes Sync Word detection more robust, because a Wire State error occurring during the most significant symbol of the 7-symbol payload word immediately preceding the Sync Word also causes corruption of the Sync Word's least significant symbol. See ***Figure 28*** for examples of Sync Word detection.

The C-PHY receiver detects the Sync Word directly at the output of the symbol decoder, prior to the symbol de-mapping block. Any 16-bit word generated by the symbol de-mapping block in response to the Sync Word shall be ignored and not passed to the receiver protocol layer.

Upon detecting the Sync Word, the C-PHY receiver resets 7-symbol word alignment to start with the first symbol immediately following Sync Word detection (i.e. the first symbol immediately following the {3} symbol). Word realignment points are shown in the Sync Word detection examples of ***Figure 28***.

All C-PHY receivers shall support Sync Word detection and realignment. While this specification itself does not require a C-PHY transmitter to support Sync Word insertion, it may instead be required by the protocol layer specification used in conjunction with the transmitter.



Notes:

- Symbols are transmitted serially from left to right
- Wire state and symbol errors are highlighted in yellow
- ▲ : point at which symbol clock is lost
- ▲ : point at which symbol clock is restored
- ◆ : point of incorrect word alignment
- ◆ : point at which correct word alignment is restored

Figure 28 Link Error and Sync Word Detection Examples

6.4.4.2 Sync Word Sync Type

The Sync Word can be any of five possible symbol sequences, where the six most significant symbols are 344444 and the least significant symbol (which is transmitted first) can be any symbol value from 0 to 4. In other words: any of the five possible symbol sequences shown in ***Table 15*** are valid Sync Words. The

transmitter shall have the capability to send any of the five possible Sync Words and the receiver shall be able to detect any of the five possible Sync Words as a valid Sync Word.

Table 15 Symbol Sequence Values of the Different Sync Types

Sync Type	Sync Value	TxSyncTypeHS0[2:0], TxSyncTypeHS1[2:0]	RxSyncTypeHS0[2:0], RxSyncTypeHS1[2:0]
Sync Type 0	3444440	0	0
Sync Type 1	3444441	1	1
Sync Type 2	3444442	2	2
Sync Type 3	3444443	3	3
Sync Type 4	3444444	4	4

Note:

Sync Type 3 is the default Sync Word value which can be selected by the transmit protocol unit for backward compatibility with systems that do not support multiple Sync Types.

The PPI described in ***Annex A*** defines TxSyncTypeHS0 and TxSyncTypeHS1 fields to allow the transmit protocol unit to select a specific Sync Word sequence corresponding to a specific Sync Type. Similarly, for receiver operation, the PPI described in ***Annex A*** defines RxSyncTypeHS0 and RxSyncTypeHS1 fields to inform the receive protocol unit that specific Sync Type values were received.

The first Sync Word in a burst, which is the Sync Word that follows the normal preamble or calibration preamble, shall be Sync Type 3 which is 3444443. This first Sync Word is transmitted automatically by the C-PHY and is not transmitted as a result of the TxSendSyncHS signals in the PPI. Subsequent Sync Words (following the first Sync Word) transmitted in the High-Speed packet data under control of the transmit protocol unit can be any of the five possible Sync Type values described in ***Table 15***. ***Figure 29*** shows an example of Sync Words with different Sync Type values transmitted in the Packet Data field.

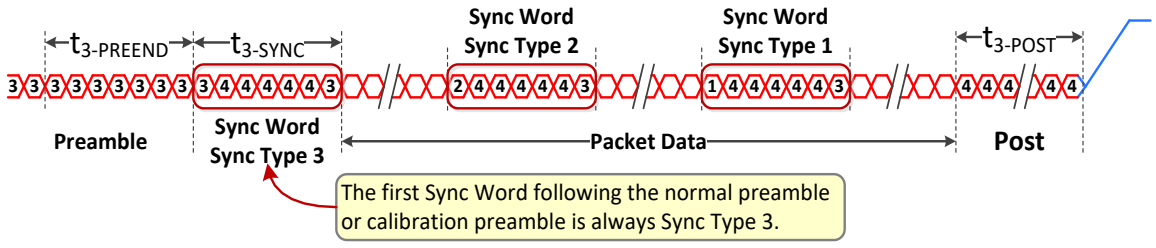


Figure 29 Example of Sync Words with Different Sync Type Values

6.4.5 Alternate Low Power (ALP) Mode Transmission Burst

Figure 30 shows the general burst format of ALP Mode. The Lane starts in either the ALP-Pause Stop or ALP-Pause ULPS state where all three Lines of the Lane are driven to the same level. In this state, $V_A = V_B = V_C$ and $V_{OD\_AB} = V_{OD\_BC} = V_{OD\_CA} = 0$. The ALP-Pause Stop and ALP-Pause ULPS states are very similar to the LP Mode Stop and ULPS states, respectively. The transmitter drives either the +x state or one of the other high-speed states for an extended period of time to wake up the receiver, and then the transmitter begins to transmit the preamble. The preamble consists of an all 3's symbol sequence in a data burst or an ALP command burst or an all 1's symbol sequence in a calibration burst. At the end of any burst the transmitter sends one of the following:

- the Stop Code followed by the Post2 sequence to inform the receiver that the link will immediately return to the ALP-Pause Stop state
- the ULPS Code followed by the Post2 sequence to inform the receiver that the link will immediately return to the ALP-Pause ULPS state

The only difference between the ALP-Pause Stop state and the ALP-Pause ULPS state at the PHY protocol level is that the receiver is given a longer time to wake up from the ALP-Pause ULPS state. By having this longer time to wake up, the receiver can generally afford to have a lower power consumption in the ALP-Pause ULPS state compared to the ALP-Pause Stop state.

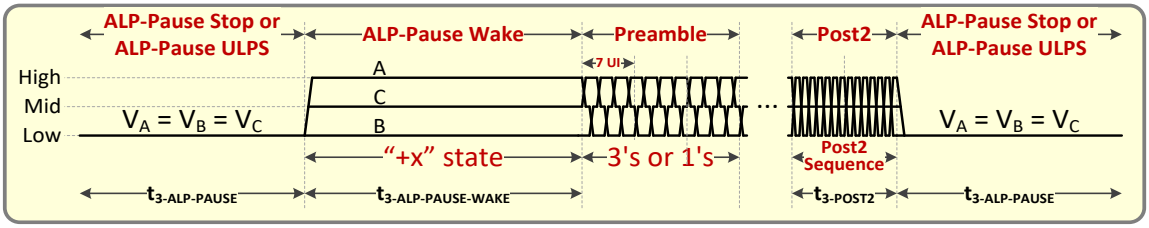


Figure 30 ALP Mode General Burst Format

Figure 31 shows an example of high-speed and ALP-Pause Wake receivers used in a single Lane. Three high-speed receivers are used to receive the A-B, B-C and C-A pairs. A low-power receiver with an offset (non-zero) input threshold voltage is used to detect the ALP-Pause Wake signal on the A-B pair when the system wakes from the ALP-Pause Stop State. A similar lower-power receiver with an offset input threshold voltage is used to detect the ALP-Pause Wake signal on the A-B pair when the system wakes from the ALP-Pause ULPS State. The circuit designer may choose to optimize the implementation of the two ALP-Pause Wake receivers by using a single low-power differential receiver having a switchable bias.

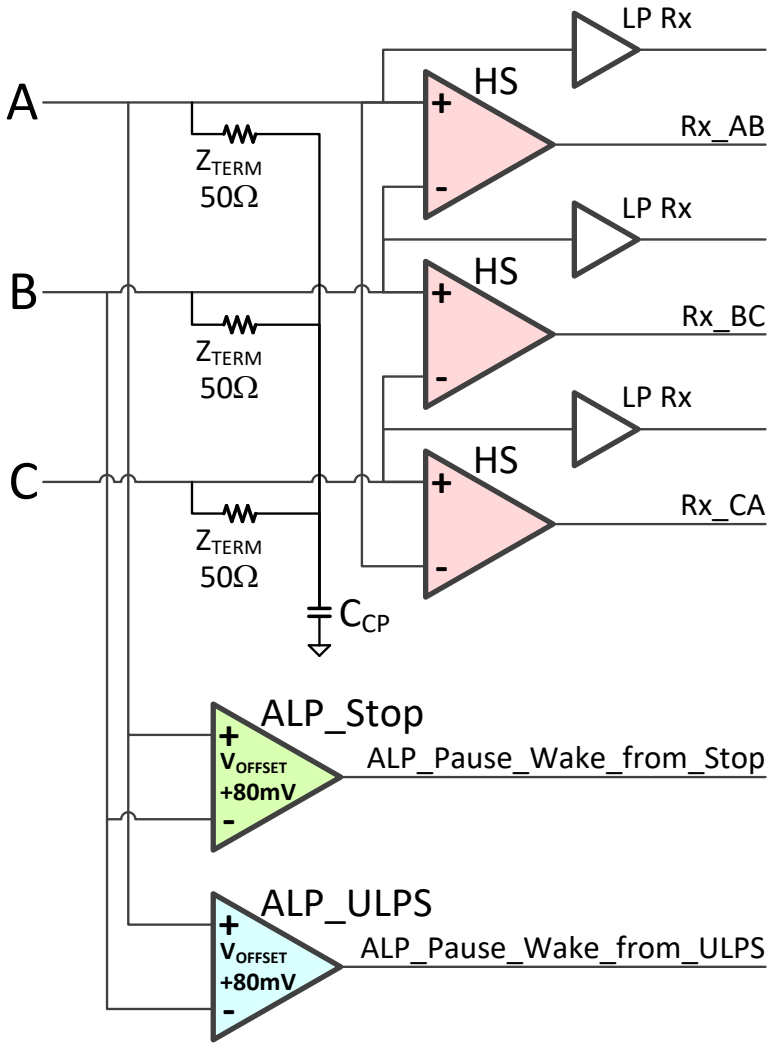


Figure 31 High-Speed and ALP-Pause Wake Receiver Example

Figure 32 shows examples of different types of bursts used to send high-speed data or ALP commands or a combination of both.

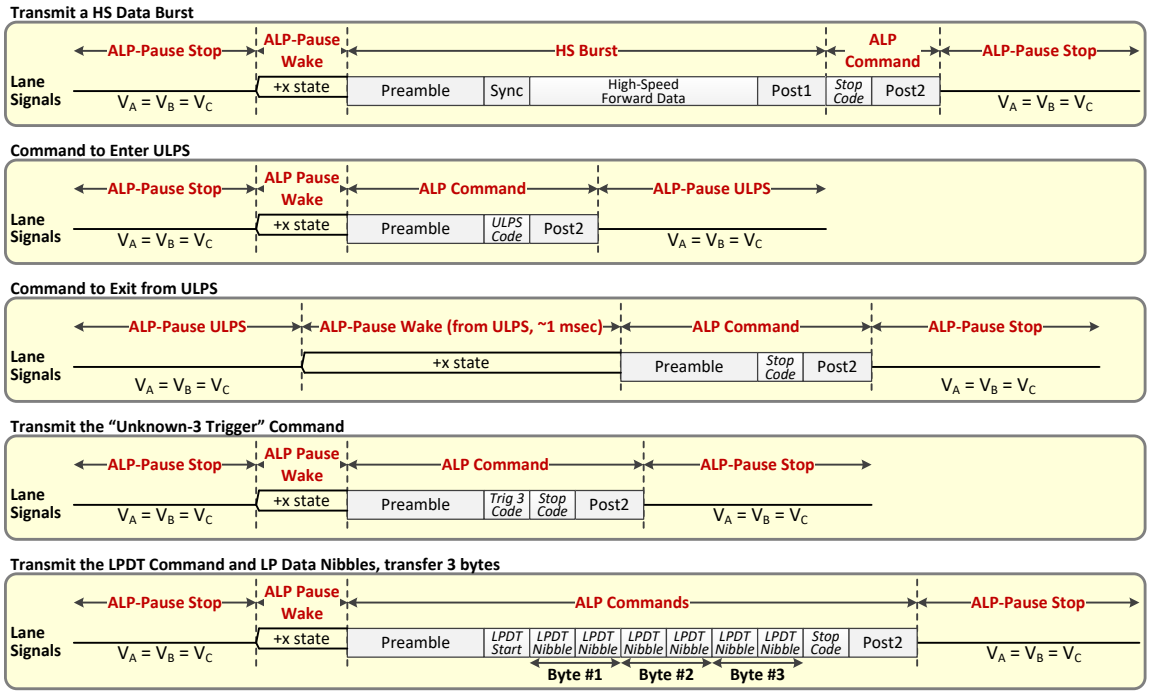


Figure 32 Examples of Bursts to Send High-Speed Data and ALP Commands

The seven-symbol code sequences in ***Table 16*** shall be used when ALP codes are transmitted. The corresponding LP mode or Escape mode action is described, where applicable.

Table 16 ALP Code Definitions

ALP Code	Symbol Sequence	PPI ALP Code	Corresponding LP State or Escape Mode Sequence
Stop Code	0244440	0b0000	LP-111 (End of Transmission, or EoT)
ULPS Code	0244441	0b0001	Escape Mode Entry + Ultra-Low Power State (ULPS)
Trig 1 Code	0244442	0b0010	Escape Mode Entry + Undefined-1 Trigger
Trig 2 Code	1244440	0b0011	Escape Mode Entry + Undefined-2 Trigger
Reset Trig Code	1244441	0b0100	Escape Mode Entry + Reset-Trigger [Remote Application]
Trig 3 Code	1244442	0b0101	Escape Mode Entry + Unknown-3 Trigger
Trig 4 Code	2244440	0b0110	Escape Mode Entry + Unknown-4 Trigger
Trig 5 Code	2244441	0b0111	Escape Mode Entry + Unknown-5 Trigger
Spacer Code	2244442	0b1000	n/a
LPDT Start Code	0144440	0b1001	Escape Mode Entry + LP Data Transfer Command
LPDT Nibble Code	$S_1$04444S_0	0b1010	LP Data, 4 bits of data
Post2	4444444	0b1011	n/a

6.4.5.1 Stop Code and ULPS Code

The Stop Code or ULPS Code shall be the last code transmitted in a burst prior to Post2. The primary purpose of these codes is to inform the receiver that the link will shut down following the Post2 field. After the Stop Code followed by Post2 is transmitted, the receiver shall enter the ALP-Pause Stop state. After the ULPS Code followed by Post2 is transmitted, the receiver shall enter the ALP-Pause ULPS state.

6.4.5.2 Trigger Codes

Trigger Codes are a mechanism to send a flag to the protocol layer at the receiving side, on request of the protocol layer on the transmitting side. This can be either in the Forward Direction or in the Reverse Direction, depending on the direction of operation and available Escape Mode functionality. The Trigger Codes in ALP mode are intended to perform the same function as the Remote Triggers that are defined for Escape mode, but the PPI defines separate signals for the transmission and reception of ALP Trigger Codes and Escape mode Remote Triggers.

The trigger codes are: Trig 1 Code, Trig 2 Code, Reset Trig, Trig 3 Code, Trig 4 code and Trig 5 code. These codes perform the same functions as the Escape mode pulse sequences: Escape Mode Entry + Undefined-1 Trigger, Escape Mode Entry + Undefined-2 Trigger, Escape Mode Entry + Reset-Trigger [Remote Application], Escape Mode Entry + Unknown-3 Trigger, Escape Mode Entry + Unknown-4 Trigger and Escape Mode Entry + Unknown-5 Trigger, respectively. Commands to transmit the trigger codes are received through the PPI from the upper layer transmit protocol unit. When the receiver receives any of the trigger codes it reports the identity of the trigger code through the PPI to the upper layer receive protocol unit.

6.4.5.3 Spacer Code

The Spacer Code may be transmitted at any time after the preamble and prior to transmission of the Stop Code or ULPS Code. The Spacer Code performs no specific function in the receiver. The purpose of the Spacer Code is to provide a means for the transmitter to insert a delay between other codes or to be able to transmit a code instead of momentarily pausing the Lane timing.

6.4.5.4 LPDT Start Code and LPDT Nibble Code

The LPDT Start Code and LPDT Nibble Code together perform the same functions as the Escape mode pulse sequences: Escape Mode Entry + LP Data Transfer Command followed by the LP Data. The transmitter shall transmit LP Data using a series of LPDT Nibble Codes. The byte ordering of the LP Data shall be to transmit the least significant nibble first followed by the most significant nibble. When the LPDT Start Code is transmitted the next LPDT Nibble Code shall correspond to a least significant nibble of the byte data being transmitted. In other words, the LPDT Start Code resets the byte alignment. Transmission of the preamble also resets the byte alignment so that the first LPDT Nibble Code transmitted after the preamble shall correspond to a least significant nibble of the byte data being transmitted. Although it is possible to transmit LP Data without the use of the LPDT Start Code, transmission of the LPDT Start Code immediately prior to transmission of a sequence of LPDT Nibble Codes is strongly recommended to remove any ambiguity in the interpretation of the byte alignment of the LP nibble data. ***Table 17*** describes how LP data is encoded into the LPDT Nibble Code sequence $S_1 04444 S_0$, where S_0 is the least significant symbol which is transmitted first, and S_1 is the most significant symbol which is transmitted last.

Table 17 LP Nibble Data Encoding

S_1	S_0	Data	S_1	S_0	Data	S_1	S_0	Data	S_1	S_0	Data
0	0	0	1	0	4	2	0	8	3	0	c
0	1	1	1	1	5	2	1	9	3	1	d
0	2	2	1	2	6	2	2	a	3	2	e
0	3	3	1	3	7	2	3	b	3	3	f

6.4.5.5 Post2 Code

The Post2 field shall be transmitted following the Stop Code or the ULPS Code. The Transmitter shall be able to control the length of $t_{3\text{-POST2}}$ in increments of seven unit intervals (UI). The length of $t_{3\text{-POST2}}$ is adjustable in the transmitter with a range from a minimum of one group of seven UI, to a maximum of 64 groups of seven UI. The length of this field in the transmitter and receiver is specified in ***Table 18***. The purpose of the Post2 Code is to provide the receiver with a sufficient number of clocks to process the Stop Code or ULPS Code and for the receiver to prepare itself to shut down to the ALP-Pause Stop state or ALP-Pause ULPS state. The number of clocks needed by the receiver is design-dependent and is addressed by the adjustment of the timing parameters in ***Table 18***. The transmitter is expected to set the duration of Post2 to confirm to the receiver's needs.

6.4.5.6 ALP Timing Parameters

The timing parameters for ALP mode are specified in ***Table 18***.

Table 18 ALP Timing Parameters

Parameter	Description	Min	Typ	Max	Unit	Notes
$t_{3\text{-POST2(TX)}}$	Time for which the Transmitter drives the sequence of all "4" symbols	7	–	224	UI	–
$t_{3\text{-POST2(RX)}}$	Time for which the Receiver accepts the sequence of all "4" symbols	Note 1	–	224	UI	1
$t_{3\text{-ALP-PAUSE(TX)}}$	Time that the Transmitter asserts the ALP-Pause Stop state.	90			ns	
	Time that the Transmitter asserts the ALP-Pause ULPS state.	0.5			ms	
$t_{3\text{-ALP-PAUSE(RX)}}$	The Receiver shall respond to an ALP-Pause Wake pulse after being in the ALP-Pause Stop state for this duration.	80			ns	
	The Receiver shall respond to an ALP-Pause Wake pulse after being in the ALP-Pause ULPS state for this duration.	0.45			ms	
$t_{3\text{-ALP-PAUSE-WAKE(TX)}}$	Time that the Transmitter asserts the ALP-Pause Wake pulse from the ALP-Pause Stop state.	90			ns	
	Time that the Transmitter asserts the ALP-Pause Wake pulse from the ALP-Pause ULPS state.	1.0			ms	
$t_{3\text{-ALP-PAUSE-WAKE(RX)}}$	In the ALP-Pause Stop state the Receiver shall respond to an ALP-Pause Wake pulse having a pulse-width in this range.	80			ns	
	In the ALP-Pause ULPS state the Receiver shall respond to an ALP-Pause Wake pulse having a pulse-width in this range.	0.9			ms	

Note:

1. *The system designer must ensure that the length of $t_{3\text{-POST2\_TX}}$ in the transmitter is compatible with the receiver's requirements. The receiver shall operate with $t_{3\text{-POST2\_RX}}$ up to the maximum specified value. The minimum value of $t_{3\text{-POST2\_RX}}$ shall be specified in the receiver's datasheet or other similar document.*

6.4.5.7 ALP Operation State Diagram

The Overall state diagram for Lane operation including High-Speed mode, calibration and ALP mode is shown in ***Figure 33***.

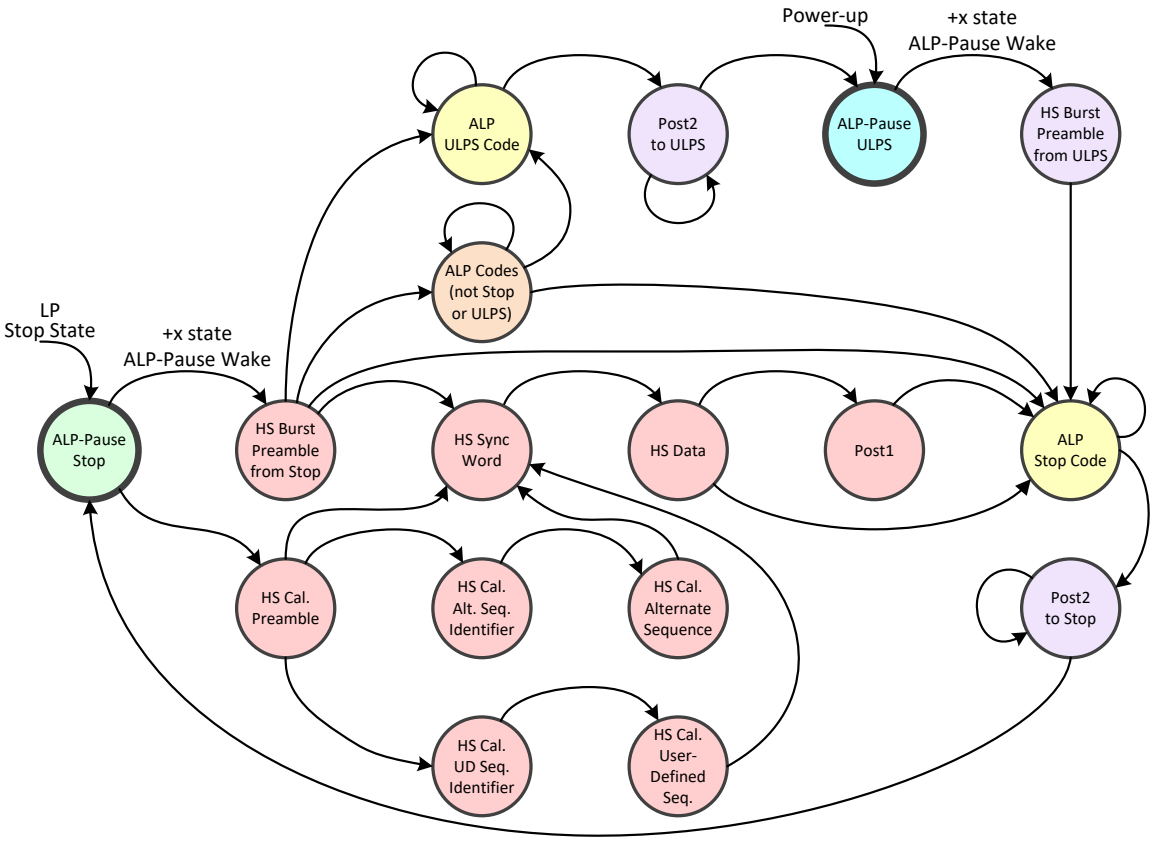


Figure 33 State Diagram for Lane Operation with High-Speed and ALP Modes

Table 19 ALP TX State Transition Description

State	Exit State	Tx Exit Conditions	Rx Exit Conditions
ALP-Pause Stop (V_A=V_B=V_C)	HS Cal. Preamble	Tx ready to send data burst or ALP burst	Rx detects ALP-Pause Wake, then 3's preamble
	HS Burst Preamble from Stop	Tx ready to send calibration burst	Rx detects ALP-Pause Wake, then 1's preamble
HS Cal. Preamble	HS Sync Word	1's Preamble complete, Format 1 burst	Rx detects Sync Word
	HS Cal. Alt. Seq. Identifier	1's Preamble complete, Format 2 burst	Rx detects Alternate Sequence Identifier
	HS Cal. UD Seq. Identifier	1's Preamble complete, Format 3 burst	Rx detects User-Defined Sequence Identifier
HS Cal. Alt. Seq. Identifier	HS Cal. Alternate Sequence	Alternate Sequence Identifier complete	Rx ready to process the Alternate Sequence
HS Cal. Alternate Sequence	HS Sync Word	Alternate Sequence complete	Rx detects Sync Word

State	Exit State	Tx Exit Conditions	Rx Exit Conditions
HS Cal. UD Seq. Identifier	HS Cal. User-Defined Seq.	User-Defined Sequence Identifier complete	Rx ready to process the User-Defined Sequence
HS Cal. User-Defined Seq.	HS Sync Word	User-Defined Sequence Complete	Rx detects Sync Word
HS Burst Preamble from Stop	ALP ULPS Code	3's Preamble complete, ready to send ALP ULPS code	Rx detects the ALP ULPS Code
	ALP Codes (not Stop or ULPS)	3's Preamble complete, ready to send other ALP codes	Rx detects ALP Codes other than Stop or ULPS
	ALP Stop Code	3's Preamble complete, ready to send ALP Stop Code	Rx detects ALP Stop Code
	HS Sync Word	3's Preamble complete, ready to send Sync Word	Rx detects Sync Word
HS Sync Word	HS Data	Sync Word complete	Rx receives HS packet data after the Sync Word
HS Data	Post1	HS Data complete	Rx detects Post
Post1	ALP Stop Code	Post1 complete	Rx detects the ALP Stop Code
ALP Stop Code	Post2 to Stop	ALP Stop Code complete	Rx receives Post2 following the ALP Stop Code, prepares to enter ALP-Pause Stop
	ALP Stop Code	Send additional ALP Stop Code	Rx detects an additional ALP Stop Code
Post2 to Stop	ALP-Pause Stop	Post2 complete	Rx has completed internal operations and can shut down to ALP-Pause Stop
	Post2 to Stop	Additional Post 2 to send	Rx detects additional Post2
ALP Codes (not Stop or ULPS)	ALP Codes (not Stop or ULPS)	Additional ALP codes ready to send	Rx detects ALP Codes other than Stop or ULPS and processes ALP Codes
	ALP ULPS Code	No other ALP Codes to send, ready to go to ALP-Pause ULPS state	Rx detects the ALP ULPS Code
	ALP Stop Code	No other ALP Codes to send, ready to go to ALP-Pause Stop state	Rx detects the ALP Stop Code
ALP ULPS Code	Post2 to ULPS	ALP ULPS Code complete	Rx receives Post2 following the ALP Stop Code, prepares to enter ALP-Pause ULPS
	ALP ULPS Code	Send additional ALP ULPS Code	Rx detects an additional ALP ULPS Code

State	Exit State	Tx Exit Conditions	Rx Exit Conditions
Post2 to ULPS	ALP-Pause ULPS	Post2 to ULPS complete	Rx has completed internal operations and can shut down to ALP-Pause ULPS
	Post2 to ULPS	Additional Post 2 to send	Rx detects additional Post2
ALP-Pause ULPS (V_A=V_B=V_C)	HS Burst Preamble	Ready to wake from ALP-Pause ULPS	Rx detects ALP-Pause Wake, then 3's preamble
HS Burst Preamble from ULPS	ALP Stop Code	3's Burst Preamble from UPLS complete	Rx detects the ALP Stop Code
any state	ALP-Pause ULPS	Power-up	Power-up
any state	ALP-Pause Stop	Detect LP mode Stop State	Detect LP mode Stop State

6.4.5.8 Concurrent LP and ALP Operation

The High-Speed state machine in ***Figure 33*** describes the processing of ALP mode commands integrated with the transmission and reception of High-Speed data and calibration bursts. The functions that use LP mode processing are performed separately in different state machines as described in ***Sections 6.5*** and ***6.6***, so it is possible to simultaneously process LP and ALP signaling.

The rules for LP and ALP command processing are as follows:

- LP operation shall be supported.
- ALP operation should be supported.
- The system shall be configured for LP operation at power-up. During initialization, the system can be configured for LP-only operation, or ALP-only operation or concurrent LP-ALP operation.

When the system is supporting concurrent LP-ALP operation the LP-111 LP mode line condition shall have priority to force the system to the Stop State.

6.5 Bi-Directional Lane Turnaround

The transmission direction of a Bi-Directional Lane can be swapped by means of a Link Turnaround procedure. This procedure enables information transfer in the opposite direction of the current direction. The procedure is the same for either a change from Forward Direction to Reverse Direction, or a change from Reverse Direction to Forward Direction. Notice that Master and Slave side shall not be changed by Turnaround. Link Turnaround shall be handled completely in Control mode. ***Table 20*** lists the sequence of events during Turnaround.

Table 20 Link Turnaround Sequence

Initial TX Side = Final RX Side	Initial RX Side = Final TX Side
Drives Stop state (LP-111)	Observes Stop state
Drives LP-Rqst state (LP-100) for time t_{LPX}	Observes transition from LP-111 to LP-100 states
Drives Bridge state (LP-000) for time t_{LPX}	Observes transition from LP-100 to LP-000 states
Drives LP-100 for time t_{LPX}	Observes transition from LP-000 to LP-100 states
Drives Bridge state (LP-000) for time $t_{TA\text{-}GO}$	Observes the transition from LP-100 to Bridge state and waits for time $t_{TA\text{-}SURE}$. After correct completion of this time-out this side knows it is in control.
–	Drives Bridge state (LP-000) for a period $t_{TA\text{-}GET}$

Initial TX Side = Final RX Side	Initial RX Side = Final TX Side
Stops driving the lines and observes the Line states with its LP-RX in order to see an acknowledgement.	–
–	Drives LP-100 for a period t_{LPX}
Observes LP-100 on the lines, interprets this as acknowledge that the other side has indeed taken control. Waits for Stop state to complete Turnaround procedure.	–
–	Drives Stop state (LP-111) for a period t_{LPX}
Observes transition to Stop state (LP-111) on the lines, interprets this as Turnaround completion acknowledgement, switches to normal LP receive mode and waits for further actions from the other side	–

Figure 34 shows the Turnaround procedure graphically.

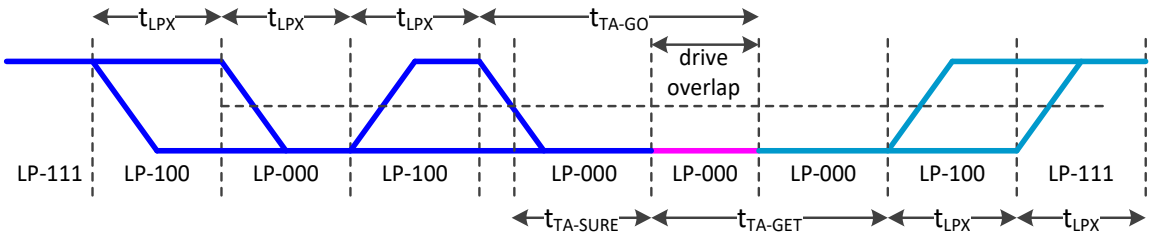
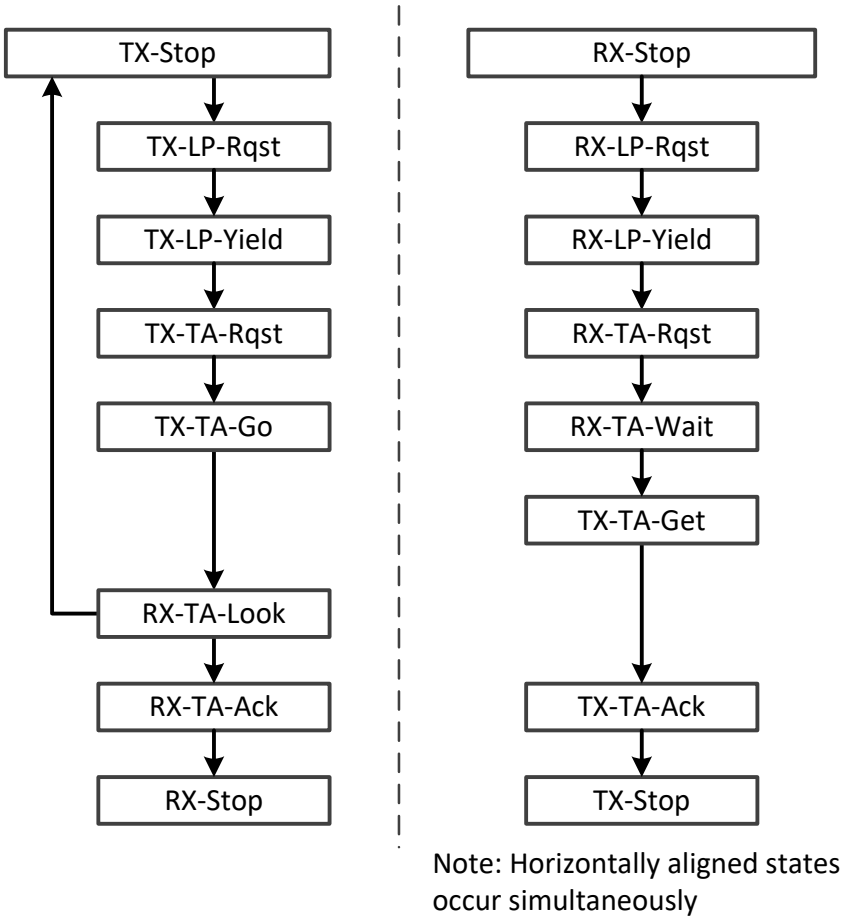


Figure 34 Turnaround Procedure

The low-power clock timing for both sides of the Link does not have to be the same, but may differ. However, the ratio between the low-power state periods, t_{LPX}, is constrained to ensure proper Turnaround behavior. See ***Table 24*** for the ratio of $t_{LPX(MASTER)}$ to $t_{LPX(SLAVE)}$.

The Turnaround procedure can be interrupted if the Lane is not yet driven into TX-LP-Yield by means of driving a Stop state. Driving the Stop state shall abort the Turnaround procedure and return the Lane to the Stop state. The PHY shall ensure against interruption of the procedure after the end of TX-TA-Rqst, RX-TA-Rqst, or TX-TA-GO. Once the PHY drives TX-LP-Yield, it shall not abort the Turnaround procedure. The protocol layer may take appropriate action if it determines an error has occurred because the Turnaround procedure did not complete within a certain time. See ***Section 7.3.5*** for more details. ***Figure 35*** shows the Turnaround state machine that is described in ***Table 21***.



889

Figure 35 Turnaround State Machine

Table 21 Turnaround State Machine Description

State	Line Condition or Line State	Exit State	Exit Conditions
Any Rx state	Any Received	RX-Stop	Observe LP-111 at lines
TX-Stop	Transmit LP-111	TX-LP-Rqst	On request of protocol for Turnaround
TX-LP-Rqst	Transmit LP-100	TX-LP-Yield	End of timed interval t_{LPX}
TX-LP-Yield	Transmit LP-000	TX-TA-Rqst	End of timed interval t_{LPX}
TX-TA-Rqst	Transmit LP-100	TX-TA-Go	End of timed interval t_{LPX}
TX-TA-Go	Transmit LP-000	RX-TA-Look	End of timed interval $T_{TA\text{-}GO}$
RX-TA-Look	Receive LP-000	RX-TA-Ack	Line transition to LP-100
RX-TA-Ack	Receive LP-100	RX-Stop	Line transition to LP-111
RX-Stop	Receive LP-111	RX-LP-Rqst	Line transition to LP-100
RX-LP-Rqst	Receive LP-100	RX-LP-Yield	Line transition to LP-000
RX-LP-Yield	Receive LP-000	RX-TA-Rqst	Line transition to LP-100
RX-TA-Rqst	Receive LP-100	RX-TA-Wait	Line transition to LP-000
RX-TA-Wait	Receive LP-000	TX-TA-Get	End of timed interval $t_{TA\text{-}SURE}$
TX-TA-Get	Transmit LP-000	TX-TA-Ack	End of timed interval $t_{TA\text{-}GET}$
TX-TA-Ack	Transmit LP-100	TX-Stop	End of timed interval t_{LPX}

Note:

During RX-TA-Look, the protocol layer may cause the PHY to transition to TX-Stop.

During High-Speed data transmission, Stop states (TX-Stop, RX-Stop) have multiple valid exit states.

6.6 Escape Mode

Escape Mode is a special mode of operation for Lanes using Low-Power states. Escape Mode applies only to the LP mode of operation. Similar capabilities exist for ALP mode that are described in ***Section 6.4.4.2***. With this mode some additional functionality becomes available. Escape Mode operation shall be supported in the Forward Direction, and is optional in the Reverse Direction. If supported, Escape Mode does not have to include all available features.

A Lane shall enter Escape Mode via an Escape Mode entry procedure (LP-111, LP-100, LP-000, LP-001, LP-000). As soon as the final Bridge state (LP-000) is observed on the Lines, the Lane shall enter Escape Mode in space state (LP-000). If an LP-111 is detected at any time before the final Bridge state (LP-000), the Escape Mode entry procedure shall be aborted and the receive side shall wait for, or return to, the Stop state.

Once Escape Mode is entered, the transmitter shall send an 8-bit entry command to indicate the requested action. ***Table 22*** lists all currently available Escape Mode commands and actions. All unassigned commands are reserved for future expansion.

The Stop state shall be used to exit Escape Mode and cannot occur during Escape Mode operation because of the spaced-one-hot encoding. The Stop state immediately returns the Lane to Control mode. If the entry command doesn’t match a supported command, that particular Escape Mode action shall be ignored and the receive side waits until the transmit side returns to the Stop state.

The PHY in Escape Mode shall apply spaced-one-hot bit encoding for asynchronous communication. Therefore, operation of a Lane in this mode does not depend on a separate clock signal. The complete Escape Mode action for a Trigger-Reset command is shown in ***Figure 36***.

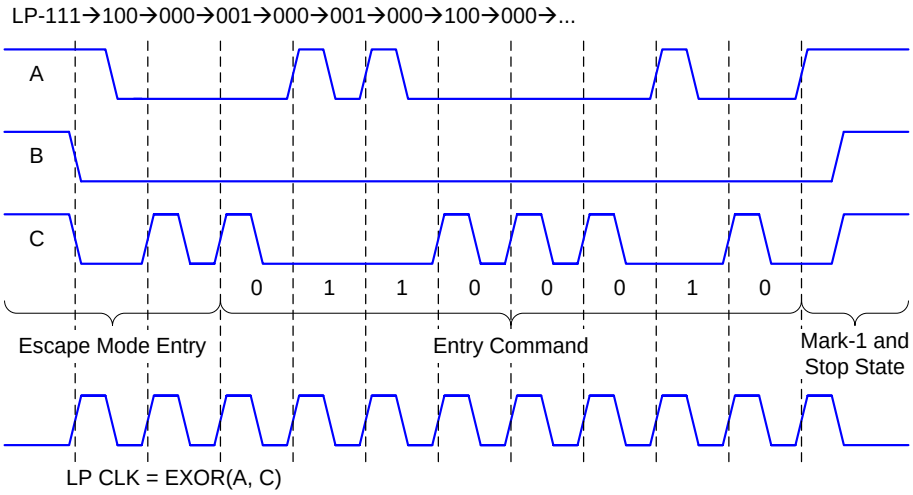


Figure 36 Trigger-Reset Command in Escape Mode

Spaced-one-hot coding means that each mark state is interleaved with a space state. Each symbol consists therefore of two parts: a one-hot phase (Mark-0 or Mark-1) and a space phase. The TX shall send Mark-0 followed by a space to transmit a ‘zero-bit’ and it shall send a Mark-1 followed by a space to transmit a ‘one-bit’. A mark that is not followed by a space does not represent a bit. The last phase before exiting Escape Mode with a Stop state shall be a Mark-1 state that is not part of the communicated bits, as it is not followed by a space state. The Clock can be derived from the two Line signals, A and C, by means of an exclusive-OR function. The length of each individual LP state period shall be at least $t_{LPX,MIN}$.

Table 22 Escape Entry Codes

Escape Mode Action	Command Type	Entry Command Pattern (first bit transmitted to last bit transmitted)
Low-Power Data Transmission	mode	11100001
Ultra-Low Power State	mode	00011110
Undefined-1	mode	10011111
Undefined-2	mode	11011110
Reset-Trigger [Remote Application]	Trigger	01100010
Unknown-3	Trigger	01011101
Unknown-4	Trigger	00100001
Unknown-5	Trigger	10100000

6.6.1 Remote Triggers

Trigger signaling is the mechanism to send a flag to the protocol layer at the receiving side, on request of the protocol layer on the transmitting side. This can be either in the Forward Direction or in the Reverse Direction, depending on the direction of operation and available Escape Mode functionality. Trigger signaling requires Escape Mode capability and at least one matching Trigger Escape entry command on both sides of the interface.

Figure 36 shows an example of an Escape Mode Reset-Trigger action. The Lane enters Escape Mode via the Escape Mode entry procedure. If the entry command Pattern matches the Reset-Trigger Command a Trigger is flagged to the protocol layer at the receive side via the logical PPI. Any bit received after a Trigger Command but before the Lines go to the Stop state shall be ignored. Therefore, dummy bytes can be concatenated in order to provide Clock information to the receive side.

Note that Trigger signaling including Reset-Trigger is a generic messaging system. The Trigger commands do not impact the behavior of the PHY itself. Therefore, Triggers can be used for any purpose by the protocol layer.

6.6.2 Low-Power Data Transmission

If the Escape Mode entry procedure is followed-up by the entry command for Low-Power Data Transmission (LPDT), then Data can be communicated by the protocol layer at low speed, while the Lane remains in low-power mode.

Data shall be encoded on the Lines with the same spaced-one-hot code as used for the entry commands. The data is self-clocked by the applied bit encoding and does not rely on a supplemental clock signal. The Lane can pause while using LPDT by maintaining a space state on the Lines. A Stop state on the Lines stops LPDT, exits Escape Mode, and switches the Lane to control mode. The last phase before Stop state shall be a Mark-1 state, which does not represent a data-bit. ***Figure 37*** shows a two-byte transmission with a pause period between the two bytes.

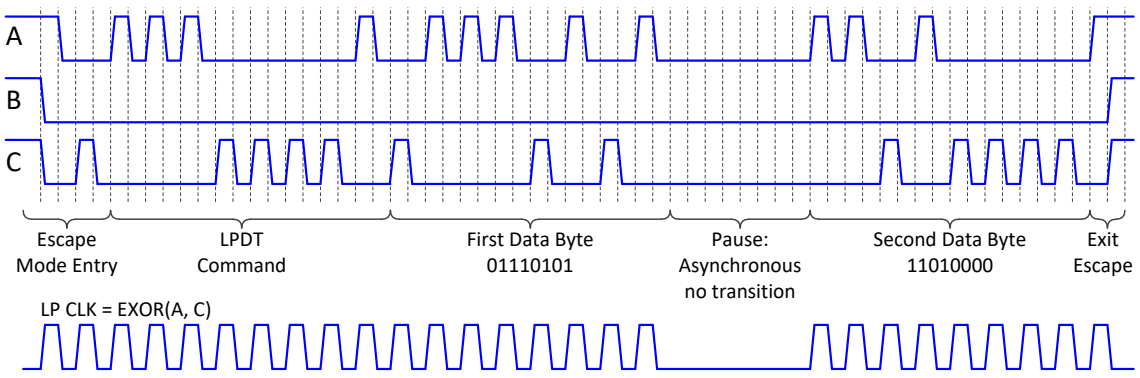


Figure 37 Two Data Byte Low-Power Data Transmission Example

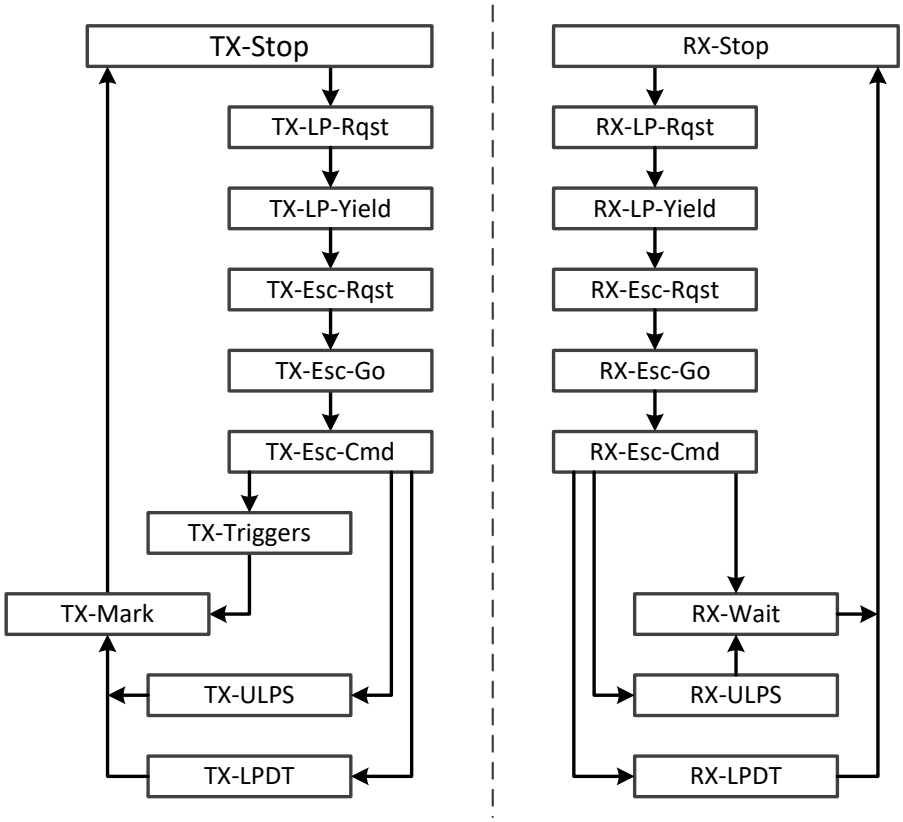
Using LPDT, a low-power (bit) clock signal ($f_{MOMENTARY}$ < 20MHz) provided to the transmit side is used to transmit data. Data reception is self-timed by the bit encoding. Therefore, a variable clock rate can be allowed. At the end of LPDT the Lane shall return to the Stop state.

6.6.3 Ultra-Low Power State

If the Ultra-Low Power State entry command is sent after an Escape Mode entry command, the Lane shall enter the Ultra-Low Power State (ULPS). This command shall be flagged to the receive side protocol. During this state, the Lines are in the space state (LP-000). Ultra-Low Power State is exited by means of a Mark-1 state with a length of t_{WAKEUP}, followed by a Stop state. ***Annex A*** describes an example of an exit procedure and a procedure to control the length of time spent in the Mark-1 state.

6.6.4 Escape Mode State Machine

The state machine for Escape Mode operation is shown in ***Figure 38*** and described in ***Table 23***.



951

Figure 38 Escape Mode State Machine

952 **Table 23 Escape Mode State Machine Description**

State	Line Condition or Line State	Exit State	Exit Conditions
Any RX state	Any Received	RX-Stop	Observe LP-111 at lines
TX-Stop	Transmit LP-111	TX-LP-Rqst	On request of protocol layer for Esc mode (PPI)
TX-LP-Rqst	Transmit LP-100	TX-LP-Yield	After time t_{LPX}
TX-LP-Yield	Transmit LP-000	TX-Esc-Rqst	After time t_{LPX}
TX-Esc-Rqst	Transmit LP-001	TX-Esc-Go	After time t_{LPX}
TX-Esc-Go	Transmit LP-000	TX-Esc-Cmd	After time t_{LPX}
TX-Esc-Cmd	Transmit sequence of 8-bit (16-Line-states) spaced-one-hot encoded entry command	TX-Triggers	After a Trigger Command
		TX-ULPS	After Ultra-Low Power Command
		TX-LPDT	After Low-Power Data Transmission Command
TX-Triggers	Space state or optional dummy bytes for the purpose of generating clocks	TX-Mark	Exit of the Trigger State on request of the protocol layer (via the PPI)
TX-ULPS	Transmit LP-000	TX-Mark	End of ULP State on request of the protocol layer (via the PPI)
TX-LPDT	Transmit serialized, spaced-one-hot encoded payload data		After last transmitted data bit
TX-Mark	Mark-1	TX-Stop	Next driven state after time t_{LPX}, or t_{WAKEUP} if leaving ULP State
RX-Stop	Receive LP-111	RX-LP-Rqst	Line transition to LP-100
RX-LP-Rqst	Receive LP-100	RX-LP-Yield	Line transition to LP-000
RX-LP-Yield	Receive LP-000	RX-Esc-Rqst	Line transition to LP-001
RX-Esc-Rqst	Receive LP-001	RX-Esc-Go	Line transition to LP-000
RX-Esc-Go	Receive LP-000	RX-Esc-Cmd	Line transition out of LP-000
RX-Esc-Cmd	Receive sequence of 8-bit (16-Line-states) spaced-one-hot encoded entry Command	RX-Wait	After Trigger and Unrecognized Commands
		RX-ULPS	After Ultra-Low Power Command
		RX-LPDT	After Low-Power Data Transmission Command
RX-ULPS	Receive LP-000	RX-Wait	Line transition to LP-100
RX-LPDT	Receive serial, Spaced-One-Hot encoded payload data	RX-Stop	Line transition to LP-111 (Last state should be a Mark-1)
RX-Wait	Any, except LP-111	RX-Stop	Line transition to LP-111

Note:

During High-Speed data transmission, Stop states (TX-Stop, RX-Stop) have multiple valid exit states.

6.7 (Not Used)

Note:

This Section is null for the C-PHY Specification. The Section heading has been retained in order to synchronize Section numbering with the D-PHY Specification [MIPI01].

6.8 (Not Used)

Note:

This Section is null for the C-PHY Specification. The Section heading has been retained in order to synchronize Section numbering with the D-PHY Specification [MIPI01].

6.9 Global Operation Timing Parameters

Table 24 lists the ranges for all timing parameters used in this Section. The values in the table assume a UI variation in the range defined by ΔUI (see ***Table 52***).

Transmitters shall support all Transmitter-specific timing parameters defined in ***Table 24***.

Receivers shall support all Receiver-specific timing parameters in defined in ***Table 24***.

Also note that while corresponding receiver tolerances are not defined for every transmitter-specific parameter, receivers shall also support reception of all allowed conformant values for all transmitter specific timing parameters in ***Table 24*** for all HS UI values up to, and including, the maximum supported HS symbol rate specified in the receiver's datasheet.

967 **Table 24 Global Operation Timing Parameters**

Parameter	Description	Min	Max	Unit	Notes
$t_{3\text{-PREPARE}}$	Time that the transmitter drives the 3-wire LP-000 Line state immediately before the start of the HS transmission.	38	95	ns	2
$t_{3\text{-TERM-EN}}$	Time for the receiver to enable the HS Line termination, starting from the time point when the A, B and C wire cross V_{IL_MAX}	Note 5	38	ns	3
$t_{3\text{-SETTLE}}$	Time interval during which the HS receiver should ignore any HS transitions on the Lane, starting from the beginning of $t_{3\text{-PREPARE}}$	Note 6	Note 7	ns	3, 4, 6, 7
$t_{HS\text{-EXIT}}$	Time that the transmitter drives LP-111 following a HS burst.	100	–	ns	2
t_{LPX}	Transmitted length of any low-power state period	50	–	ns	1, 2
t_{INIT}	See ***Section 6.11***.	100	–	µs	2
Ratio t_{LPX}	Ratio of $t_{LPX(MASTER)}/t_{LPX(SLAVE)}$ between Master and Slave side	2/3	–	3/2	
$t_{TA\text{-GET}}$	Time that the new transmitter drives the Bridge state (LP-000) after accepting control during a Link Turnaround.	$5\cdot t_{LPX}$		–	2
$t_{TA\text{-GO}}$	Time that the transmitter drives the Bridge state (LP-000) before releasing control during a Link Turnaround.	$4\cdot t_{LPX}$		–	2
$t_{TA\text{-SURE}}$	Time that the new transmitter waits after the LP-100 state before transmitting the Bridge state (LP-000) during a Link Turnaround.	t_{LPX}	$2\cdot t_{LPX}$	ns	2
t_{WAKEUP}	Time that a transmitter drives a Mark-1 state prior to a Stop state in order to initiate an exit from ULPS.	1	–	ms	2

Note:

1. *t_{LPX} is an internal state machine timing reference. Externally measured values may differ slightly from the specified values due to asymmetrical rise and fall times.*
2. *Transmitter-specific parameter.*
3. *Receiver-specific parameter.*
4. *The stated values are considered informative guidelines rather than normative requirements since this parameter is untestable in typical applications.*
5. *As specified in **Section 9.2.1**, the receiver termination impedances shall not be enabled until the single-ended voltages on all of A, B and C fall below $V_{TERM\text{-}EN}$.*
6. *$t_{3\text{-SETTLE}}$ should be configurable in the receiver. $t_{3\text{-SETTLE}}$ should end after the recovered clock is stable at the receiver, typically $t_{3\text{-PREPARE}}$ + 6 UI. The time for the recovered clock to be stable depends on the time to enable the HS drivers in the transmitter and time to enable the HS receivers. The HS receivers are typically ready before the HS drivers are enabled because the HS receivers are enabled starting at the beginning of $t_{3\text{-PREPARE}}$.*
7. *$t_{3\text{-SETTLE}}$ should end before the end of $t_{3\text{-PREBEGIN}}$ so the receiver's internal clock can run for a sufficient time to be able to initialize the PHY and protocol circuitry in the receiver. The length of $t_{3\text{-PREBEGIN}}$ is configured so that this condition can be met.*

968 $t_{3\text{-SETTLE}} > t_{3\text{-PREPARE}}$ $t_{3\text{-PROGSEQ}} = 14$ UI or 0 UI

969 $t_{3\text{-SETTLE}} < t_{3\text{-PREPARE}} + t_{3\text{-PREAMBLE}}$ $t_{3\text{-PREEND}} = 7$ UI

970 $t_{3\text{-PREAMBLE}} = t_{3\text{-PREBEGIN}} + t_{3\text{-PROGSEQ}} + t_{3\text{-PREEND}}$ $t_{3\text{-SYNC}} = 7$ UI

971 *$t_{3\text{-PREBEGIN}}$ should be adjustable at the transmitter from 7 UI minimum to 448 UI maximum in*
972 *increments of 7 UI. An example method to specify the length of $t_{3\text{-PREBEGIN}}$ is provided in*
973 **Section 12.5.3**. *$t_{3\text{-POST}}$ should be adjustable at the transmitter from 7 UI minimum to 224 UI maximum*

in increments of 7 UI. An example method to specify the length of $t_{3\text{-}POST}$ is provided in **Section 12.5.4**.

6.10 System Power States

Each Lane within a PHY Configuration, that is powered and enabled, has potentially three different power consumption levels: High-Speed transmission mode, Low-Power mode (ALP-Pause Stop), and Ultra-Low Power State (ALP-Pause ULPS). For details on Ultra-Low Power State see ***Section 6.6.3***. For details regarding ALP-Pause Stop and ALP-Pause ULPS, refer so ***Section 6.4.4.2***. The transition between these modes shall be handled by the PHY.

6.11 Initialization

6.11.1 LP Initialization

After power-up, the Slave side PHY shall be initialized when the Master PHY drives a Stop state (LP-111) for a period longer than t_{INIT}. The first Stop state longer than the specified t_{INIT} is called the initialization period. The Master PHY itself shall be initialized by a system or protocol layer input signal (PPI). The Master side shall ensure that a Stop state longer than t_{INIT} does not occur on the lines before the Master is initialized. The Slave side shall ignore all Line states during an interval of unspecified length prior to the initialization period. In multi-Lane configurations, all Lanes shall be initialized simultaneously.

Note that t_{INIT} is considered a protocol-dependent parameter, and thus the exact requirements for $t_{INIT,MASTER}$ and $t_{INIT,SLAVE}$ (transmitter and receiver initialization Stop state lengths, respectively,) are defined by the protocol layer specification and are outside the scope of this document. However, the C-PHY specification does place a minimum bound on the lengths of $t_{INIT,MASTER}$ and $t_{INIT,SLAVE}$, which each shall be no less than 100 µs. A protocol layer specification using the C-PHY specification may specify any values greater than this limit, for example, $t_{INIT,MASTER} \geq 1$ ms and $t_{INIT,SLAVE} = 500$ to 800 µs.

Table 25 LP Initialization States

State	Entry Conditions	Exit State	Exit Conditions	Line Levels
Master Off	Power-down	Master Initialization	Power-up	Any LP level except Stop states for periods>100µs
Master Init	Power-up or protocol request	TX-stop	A first Stop state for a period longer than $t_{INIT,MASTER}$ as specified by the protocol layer	Any LP signaling sequence that ends with a long initialization Stop state
Slave Off	Power-down	Any LP state	Power-up	Any
Slave Init	Power-up or protocol layer request	RX-Stop	Observe Stop state at the inputs for a period $t_{INIT,SLAVE}$ as specified by the protocol layer	Any LP signaling sequence which ends with the first long initialization stop period

6.11.2 ALP Initialization

After power-up, the Slave side PHY shall be initialized when the Master PHY transmits an ALP or HS burst ending with an ALP Stop Code and Post2 without sending a subsequent ALP burst or HS burst a period longer than t_{INIT}. This is approximately the time that the Master remains in the ALP-Pause Stop state. The sequence of transmitting a burst ending with an ALP Stop Code and Post2 followed by no other ALP burst or HS burst for a period longer than the specified t_{INIT} is called the initialization period. The Master PHY

itself shall be initialized by a system or protocol layer input signal (PPI). The Master side shall ensure that a the sequence of transmitting a burst ending with a Stop Code followed by no other burst for a period longer than t_{INIT} does not occur on the Lane before the Master is initialized. The Slave side shall not respond to other ALP Codes (other than the Preamble, ALP Stop Code, ALP Post 2 or Sync Word) during an interval of unspecified length prior to the initialization period. In multi-Lane configurations, all Lanes shall be initialized simultaneously.

Note that t_{INIT} is considered a protocol-dependent parameter, and thus the exact requirements for $t_{INIT,MASTER}$ and $t_{INIT,SLAVE}$ (transmitter and receiver initialization ALP Stop Code separation durations, respectively,) are defined by the protocol layer specification and are outside the scope of this document. However, the C-PHY specification does place a minimum bound on the lengths of $t_{INIT,MASTER}$ and $t_{INIT,SLAVE}$, which each shall be no less than 100 µs. A protocol layer specification using the C-PHY specification may specify any values greater than this limit, for example, $t_{INIT,MASTER} \geq 1$ ms and $t_{INIT,SLAVE} = 500$ to 800 µs.

Table 26 ALP Initialization States

State	Entry Conditions	Exit State	Exit Conditions	Line Levels
Master Off	Power-down	Master Initialization	Power-up	ALP-Pause
Master Init	Power-up or protocol request	TX-Stop	Transmission of an HS Burst ending with an ALP Stop Code and Post2, followed by no other burst for a period longer than $t_{INIT,MASTER}$ as specified by the protocol layer.	ALP-Pause after completion of an HS Burst ending with an ALP Stop Code and Post2
Slave Off	Power-down	ALP-Pause ULPS	Power-up	ALP-Pause
Slave Init	Power-up or protocol layer request	RX-Stop	Observe reception of an HS Burst ending with an ALP Stop Code and Post2, followed by no other burst for a period longer than $t_{INIT,SLAVE}$ as specified by the protocol layer	ALP-Pause after completion of an HS Burst ending with an ALP Stop Code and Post2

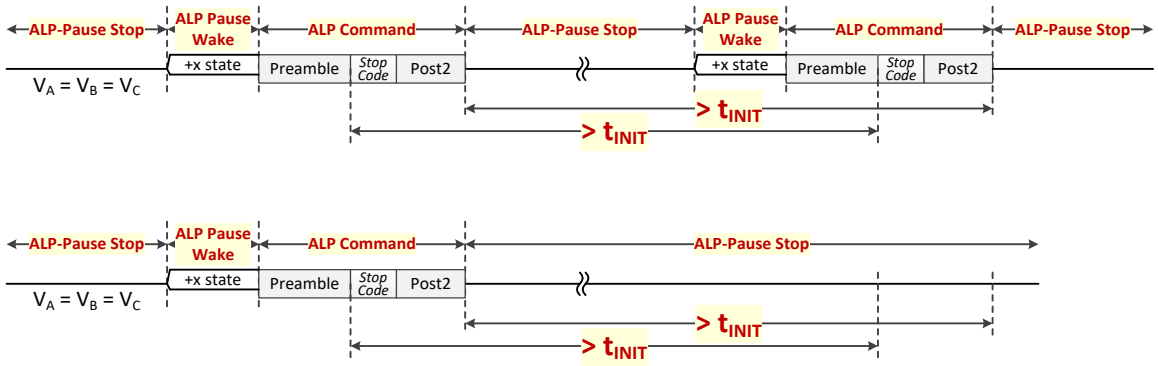


Figure 39 Examples of the Initialization Period after Power-Up

6.12 Calibration

At higher symbol rates, it is useful to calibrate delay circuits in the Receiver. This calibration compensates for natural variations of the Receiver circuitry due to process, voltage, and temperature. Calibration shall be supported by Transmitters and Receivers that operate above 3.0 Gsps. Calibration may be used at 3.0 Gsps or lower if it is supported in both the Transmitter and the Receiver. It will be known at the time the system is configured whether calibration is supported in both the Transmitter and the Receiver, so calibration can be used if it is supported in both devices.

A Transmitter causes a Receiver to perform calibration by transmitting a unique, and potentially extended-length, data burst preamble consisting of single-transition symbols. High-Speed packet data can be sent in the data bursts with the calibration preamble, in the same manner that packet data is sent in data bursts having the normal preamble. The calibration preamble begins with a sequence of "1" symbols instead of a sequence of "3" symbols. Because the Receiver can easily differentiate the "1" and "3" symbols without being calibrated, it can distinguish between a normal preamble and a calibration preamble. The calibration preamble may optionally include an Alternate Sequence field, in order to further enhance the Receiver's ability to adapt to conditions present in the system at the time when the calibration is performed.

Figure 40 highlights data burst format differences between normal preambles and calibration preambles. Data bursts with the normal preamble are described in detail in ***Section 6.4.4***. A Receiver able to calibrate itself using the data bursts with the normal preamble, without using data burst with the calibration preamble, may do so.

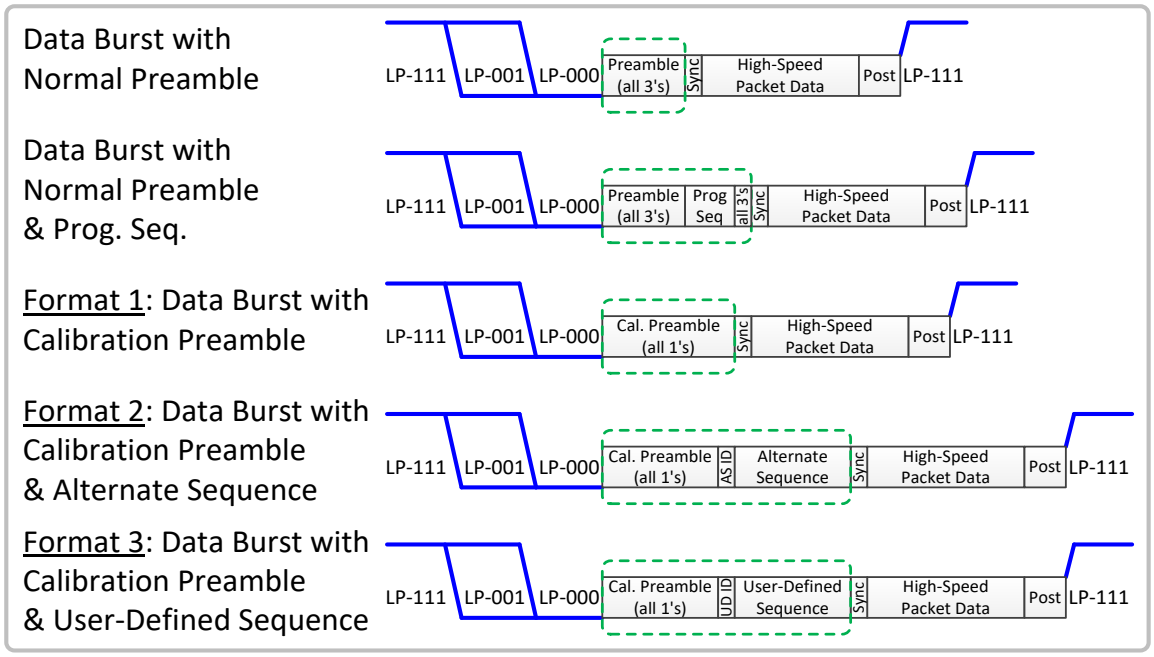


Figure 40 Normal Preambles Vs. Calibration Preambles

6.12.1 Calibration Preamble Formats

The calibration preamble shall be transmitted using one of the three formats illustrated in ***Figure 41***. The general nature of the calibration burst is identical to that of a normal burst described in ***Section 6.4.4***, except that the preamble duration and value of the symbols in the calibration preamble are different. A calibration preamble shall have one of three possible formats.

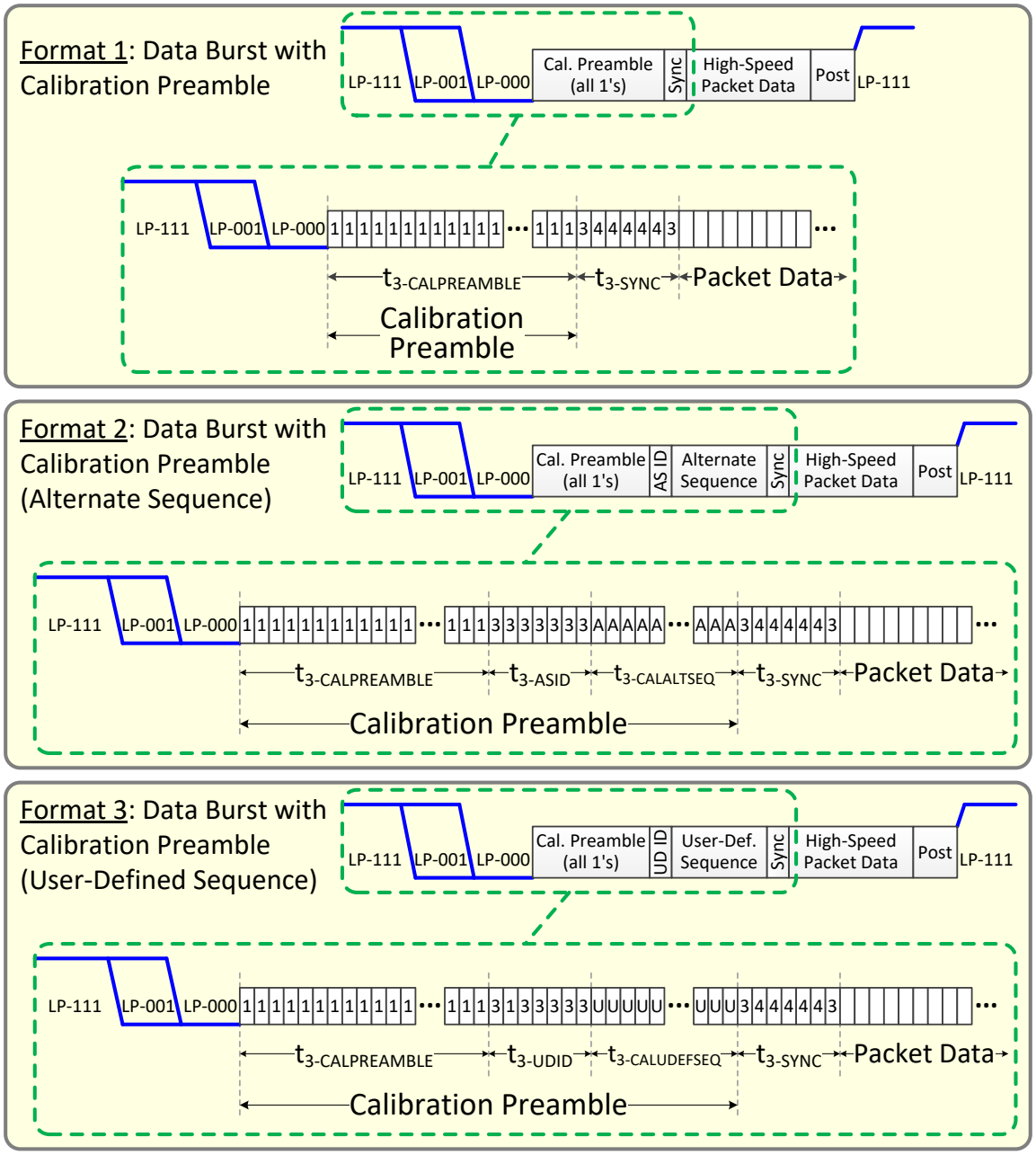


Figure 41 High-Speed Data Transmission in Rx-Calibration

6.12.1.1 Calibration Preamble Format 1

The first calibration preamble format illustrated in ***Figure 41*** consists only of a calibration preamble symbol sequence, called $t_{3\text{-CALPREAMBLE}}$. It consists of a sequence of "1" symbols (flip of 0, rotation of 0, and polarity of 1). The Transmitter shall be able to control the length of $t_{3\text{-CALPREAMBLE}}$ in increments of seven unit intervals (UI). The length of $t_{3\text{-CALPREAMBLE}}$ is adjustable in the transmitter with a range from a minimum of one group of seven UI, to a maximum of 256 groups of seven UI. The length of this field in the transmitter and receiver is specified in ***Table 32***.

6.12.1.2 Calibration Preamble Format 2

Figure 41 also illustrates the second calibration preamble format, which consists of three fields transmitted in the following sequence:

1. $t_{3\text{-CALPREAMBLE}}$: A calibration preamble symbol sequence consisting of a sequence of "1" symbols (flip of 0, rotation of 0, and polarity of 1). The length of this field is adjustable in the transmitter in increments of seven unit intervals (UI) with a range from a minimum of one group of seven UI, to a maximum of 256 groups of seven UI. The length of this field in the transmitter and receiver is specified in ***Table 32***.
2. $t_{3\text{-ASID}}$: Alternate sequence identifier. A sequence of seven "3" symbols (flip of 0, rotation of 1, and polarity of 1). The purpose of this field is to inform the Receiver that the Alternate Sequence field follows. The length of this field shall be fixed at 7 UI.
3. $t_{3\text{-CALALTSEQ}}$: Alternate Sequence field consisting of mapped and encoded PRBS9 data. The length of this field is adjustable in the transmitter in increments of seven unit intervals (UI) with a range from a minimum of one group of seven UI, to a maximum of 2048 groups of seven UI. The length of this field in the transmitter and receiver is specified in ***Table 32***.

The alternate sequence field, $t_{3\text{-CALALTSEQ}}$, shall consist of a PRBS9 pseudorandom sequence defined by the polynomial $x^0 + x^5 + x^9$. The output of the PRBS9 sequence generator shall be mapped and encoded into a sequence of symbols using the C-PHY mapping and encoding defined in ***Section 6***. Since this pattern is produced by the 3-phase mapper and encoder it will not contain the Sync Word or Post in any shifted form. The PRBS9 sequence generator shall be implemented using the method shown in ***Figure 42***.

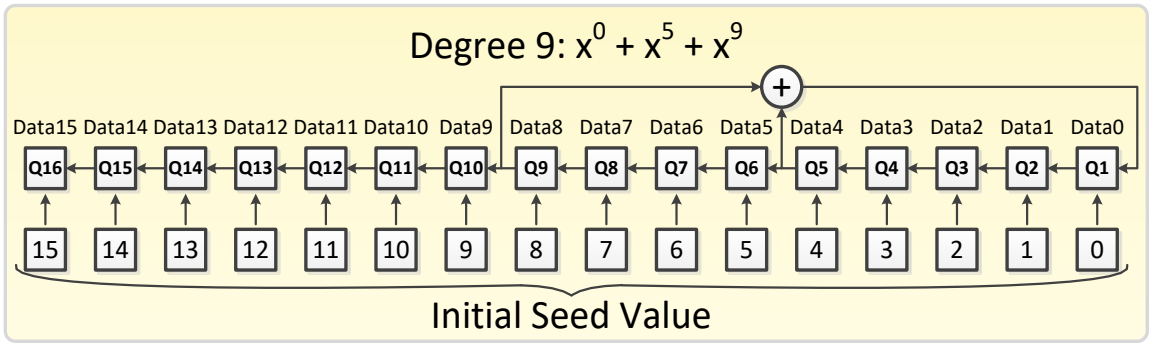


Figure 42 PRBS9 Sequence Generator for the Alternate Sequence

The PRBS9 sequence generator shall be initialized prior to the beginning of the Alternate Sequence using an initial seed value that is programmable in the transmitter. The system designer can choose a seed value for each Lane that is most appropriate for the application. It is possible to use the same seed or different seed value for each Lane within a Link. It is also possible to change the seed value during the course of the system operation. It is preferred to always use the same seed value for calibration. Guidance for choosing specific seed values is beyond the scope of this specification.

After the PRBS9 sequence generator is initialized using the initial seed value, the first 16-bit word of the sequence is the seed value contained in the generator register Q1 through Q16. The first 16-bit word output from the PRBS9 sequence generator is Data[15:0] which is equal to the initial seed value. The PRBS9 sequence generator is shifted 16 times before each successive 16-bit value is output on Data[15:0]. The 16-bit words output on Data[15:0] are presented to the 3-phase mapper. An example sequence of 16-bit values output by the PRBS9 sequence generator using a seed value of 0x0001 is shown in ***Table 27***.

Table 27 Example Sequence of 16-bit Values and Corresponding Transmission States

Word # in sequence	Word Value	Symbol							Wire State[1]						
		0	1	2	3	4	5	6	0	1	2	3	4	5	6
0 (initial seed)	0x0001	1	0	0	0	0	0	0	-z	-y	-x	-z	-y	-x	-z
1	0x08c2	2	0	0	3	0	2	0	-x	-z	-y	+z	+y	+z	+y
2	0x72ac	0	3	2	4	2	2	0	+x	-y	-z	+z	+x	+y	+x
3	0x37a6	2	1	2	2	3	1	3	+y	-x	-y	-z	+x	-z	+x
4	0xe450	0	0	4	1	1	4	0	+z	+y	-y	+x	-z	+z	+y
5	0xad3f	3	3	3	0	1	3	4	-z	+x	-y	-x	+z	-x	+x
6	0x6496	2	1	4	1	2	0	1	+y	-x	+x	-z	-x	-z	+y

Note:

1. *In this example, the wire state of each symbol is based on an initial wire state of +x, which would have been the wire state transmitted in the last Unit Interval of $t_{3\text{-ASID}}$ prior to $t_{3\text{-CALALTSEQ}}$. Any of the six wire states are possible as an initial value. The +x state was chosen for this example.*

6.12.1.3 Calibration Preamble Format 3

Figure 41 also illustrates the third calibration preamble format, which consists of three fields transmitted in the following sequence:

- $t_{3\text{-CALPREAMBLE}}$: A calibration preamble symbol sequence consisting of a sequence of "1" symbols (flip of 0, rotation of 0, and polarity of 1). The length of this field is adjustable in the transmitter in increments of seven unit intervals (UI) with a range from a minimum of one group of seven UI, to a maximum of 256 groups of seven UI. The length of this field in the transmitter and receiver is specified in ***Table 32***.
- $t_{3\text{-UDID}}$: User-defined sequence identifier. A unique sequence of seven symbols, 3333313 (from most significant symbol, s6, to least significant symbol, s0). The order of symbol transmission is defined in ***Section 6.1.2***. The purpose of this field is to inform the Receiver that the user-defined sequence field follows. The length of this field shall be fixed at 7 UI.
- $t_{3\text{-CALUDEFSEQ}}$: User-defined sequence field consisting of a user-defined sequence of symbols. The length of this field shall be adjustable in the transmitter in increments of seven unit intervals (UI) with a range from a minimum of one group of seven UI, to a maximum of 2048 groups of seven UI. The length of this field in the transmitter and receiver is specified in ***Table 32***.

The user-defined sequence field, $t_{3\text{-CALUDEFSEQ}}$, consists of a sequence of symbols selected to support calibration functions performed in the receiver. It is highly recommended to avoid unmapped sequences of symbols that are used for specific purposes, such as the Sync Word or Post.

6.12.2 Calibration Operations

Transmitters and Receivers that support the calibration capability shall observe the specifications detailed in this Section.

- Transmitters and Receivers that support the calibration capability shall have the capability to transmit the Format 1 calibration preamble.
- Transmitters that support the calibration capability shall have the capability to transmit the Format 2 calibration preamble. Receivers may support the Format 2 calibration preamble, however such support is not required.
- Transmitters and Receivers that support the calibration capability may support the Format 3 calibration preamble, however such support is not required.

Table 28 summarizes the requirements for the capability to transmit the different types of calibration preambles for transmitters and receivers.

Table 28 Summary of Calibration Burst Requirements for Transmitters and Receivers

	Tx	Rx
Calibration Preamble Format 1	Required if calibration is supported.	Required if calibration is supported.
Calibration Preamble Format 2	Required if calibration is supported.	Optional if calibration is supported.
Calibration Preamble Format 3	Optional if calibration is supported.	Optional if calibration is supported.

The Receiver shall detect the calibration preamble pattern, and shall initiate calibration upon detection of the calibration preamble pattern. Although the protocol unit is not required to request that calibration be performed simultaneously on all Lanes of a Link, it may be convenient to do so, so that the lengths of all of the fields of the preamble are the same on all Lanes of a Link. A calibration burst also has a High-Speed packet data field. The sending of application layer packets in a calibration burst is determined by the upper layer protocol controller. The Format 1 calibration sequence without the alternate sequence field is described in ***Table 29***, the Format 2 calibration sequence with the alternate sequence field is described in ***Table 30***, and the Format 3 calibration sequence with the user-defined sequence field is described in ***Table 31***. The end-of-transmission sequence is identical to the standard end of transmission sequence described in described in ***Table 12***.

Table 29 Start of Format 1 Calibration Sequence

Tx Side	Rx Side
Drives Stop state (LP-111)	Observes Stop state
Drives HS-Rqst state (LP-001) for time t_{LPX}	Observes transition from LP-111 to LP-001 on the Lines
Drives Bridge state (LP-000) for time $t_{3\text{-PREPARE}}$	Observes transition from LP-001 to LP-000 on the Lines, enables Line termination after time $t_{3\text{-TERM-EN}}$
Enables High-Speed driver and disables low-power drivers simultaneously.	–
Drives Calibration Preamble sequence for time $t_{3\text{-CALPREAMBLE}}$	Enables HS-RX and waits for timer $t_{3\text{-SETTLE}}$ to expire, in order to neglect transition effects
–	Detects sequence of all “1” symbols, performs calibration in the receiver.
–	Starts looking for the Sync Word sequence
Inserts the Sync Word Sequence	–
–	Synchronizes upon recognition of Sync Word Sequence
Continues to transmit High-Speed payload data	–
–	Receives payload data

Note:

The Receiver shall complete the calibration function during the time while the calibration preamble is transmitted. No capability for the Receiver to notify the Transmitter of completion of the calibration function is defined in this Specification.

1122 **Table 30 Start of Format 2 Calibration Sequence**

Tx Side	Rx Side
Drives Stop state (LP-111)	Observes Stop state
Drives HS-Rqst state (LP-001) for time t_{LPX}	Observes transition from LP-111 to LP-001 on the Lines
Drives Bridge state (LP-000) for time $t_{3\text{-PREPARE}}$	Observes transition from LP-001 to LP-000 on the Lines, enables Line termination after time $t_{3\text{-TERM-EN}}$
Enables High-Speed driver and disables low-power drivers simultaneously.	–
Drives Calibration Preamble sequence for time $t_{3\text{-CALPREAMBLE}}$	Enables HS-RX and waits for timer $t_{3\text{-SETTLE}}$ to expire, in order to neglect transition effects
–	Detects sequence of all “1” symbols, performs calibration in the receiver.
–	Starts looking for alternate sequence identifier of all “3” symbols
Drives alternate sequence identifier of all “3” symbols for time $t_{3\text{-ASID}}$, in order to indicate that the alternate sequence field follows	–
–	Detects alternate sequence identifier of all “3” symbols and prepares to receive the alternate sequence field
Drives the alternate sequence field for time $t_{3\text{-CALALTSEQ}}$	Performs necessary operations using the alternate sequence field
–	Starts looking for the Sync Word sequence
Inserts the Sync Word Sequence	–
–	Synchronizes upon recognition of Sync Word Sequence
Continues to transmit High-Speed payload data	–
–	Receives payload data

1123 **Table 31 Start of Format 3 Calibration Sequence**

Tx Side	Rx Side
Drives Stop state (LP-111)	Observes Stop state
Drives HS-Rqst state (LP-001) for time t_{LPX}	Observes transition from LP-111 to LP-001 on the Lines
Drives Bridge state (LP-000) for time $t_{3\text{-PREPARE}}$	Observes transition from LP-001 to LP-000 on the Lines, enables Line termination after time $t_{3\text{-TERM-EN}}$
Enables High-Speed driver and disables low-power drivers simultaneously.	–
Drives Calibration Preamble sequence for time $t_{3\text{-CALPREAMBLE}}$	Enables HS-RX and waits for timer $t_{3\text{-SETTLE}}$ to expire, in order to neglect transition effects
–	Detects sequence of all “1” symbols, performs calibration in the receiver.
–	Starts looking for user-defined sequence identifier, 3333313 symbol sequence

Drives user-defined sequence identifier, 3333313 symbol sequence, for time $t_{3\text{-UDID}}$, in order to indicate that the user-defined sequence field follows	–
–	Detects user-defined sequence identifier, 3333313 symbol sequence, and prepares to receive the alternate sequence field
Drives the user-defined sequence field for time $t_{3\text{-CALUDEFSEQ}}$	Performs necessary operations using the user-defined sequence field
–	Starts looking for the Sync Word sequence
Inserts the Sync Word Sequence	–
–	Synchronizes upon recognition of Sync Word Sequence
Continues to transmit High-Speed payload data	–
–	Receives payload data

1124 Timing parameters for the calibration preamble are summarized in ***Table 32***.

1125 **Table 32 Calibration Preamble Timing Parameters**

Parameter	Description	Min	Typ	Max	Unit	Notes
$t_{3\text{-CALPREAMBLE\_TX}}$	Time for which the Transmitter drives the sequence of all “1” symbols	7	–	1792	UI	–
$t_{3\text{-CALPREAMBLE\_RX}}$	Time for which the Receiver accepts the sequence of all “1” symbols	Note 1	–	1792	UI	1
$t_{3\text{-ASID}}$	Time for which the Transmitter transmits the alternate sequence identifier (after the $t_{3\text{-CALPREAMBLE}}$ and before $t_{3\text{-CALALTSEQ}}$)	–	7	–	UI	–
$t_{3\text{-UDID}}$	Time for which the Transmitter transmits the user-defined sequence identifier (after the $t_{3\text{-CALPREAMBLE}}$ and before $t_{3\text{-CALUDEFSEQ}}$)	–	7	–	UI	–
$t_{3\text{-CALALTSEQ\_TX}}$	Time for which the Transmitter transmits the alternate sequence field	7	–	14,336	UI	–
$t_{3\text{-CALALTSEQ\_RX}}$	Time for which the Receiver accepts the alternate sequence field	Note 2	–	14,336	UI	2
$t_{3\text{-CALUDEFSEQ\_TX}}$	Time for which the Transmitter transmits the user-defined sequence field	7	–	14,336	UI	–
$t_{3\text{-CALUDEFSEQ\_RX}}$	Time for which the Receiver accepts the user-defined sequence field	Note 3	–	14,336	UI	3

Note:

1. *The system designer must ensure that the length of $t_{3\text{-CALPREAMBLE\_TX}}$ in the transmitter is compatible with the receiver’s requirements. The receiver shall operate with $t_{3\text{-CALPREAMBLE\_RX}}$ up to the maximum specified value. The minimum value of $t_{3\text{-CALPREAMBLE\_RX}}$ shall be specified in the receiver’s datasheet or other similar document.*
2. *The system designer must ensure that the length of $t_{3\text{-CALALTSEQ\_TX}}$ in the transmitter is compatible with the receiver’s requirements. The receiver shall operate with $t_{3\text{-CALALTSEQ\_RX}}$ up to the maximum specified value. The minimum value of $t_{3\text{-CALALTSEQ\_RX}}$ shall be specified in the receiver’s datasheet or other similar document.*
3. *The system designer must ensure that the length of $t_{3\text{-CALUDEFSEQ\_TX}}$ in the transmitter is compatible with the receiver’s requirements. The receiver shall operate with $t_{3\text{-CALUDEFSEQ\_RX}}$ up to the maximum*

specified value. The minimum value of $t_{3\text{-}CALUDEFSEQ\_RX}$ shall be specified in the receiver's datasheet or other similar document.

The symbol rate can be reduced (i.e., to a lower symbol rate than the symbol rate that was in use at the time the calibration sequence was transmitted) without adversely affecting Receiver operation. For example: if calibration is performed at a rate "R", then the operating symbol rate can be reduced to R/2 without having to recalibrate. The amount by which the symbol rate can be increased after calibration without having to recalibrate depends upon Receiver implementation.

6.13 Global Operation Flow Diagram

All previously described aspects of operation, either including or excluding optional parts, are contained in Lane Modules. ***Figure 43*** shows the operational flow diagram for a Lane Module. Within both TX and RX five main processes can be distinguished: High-Speed Transmission, Escape Mode, Turnaround, ALP-Pause (ALP-Pause Stop and ALP-Pause ULPS), and initialization.

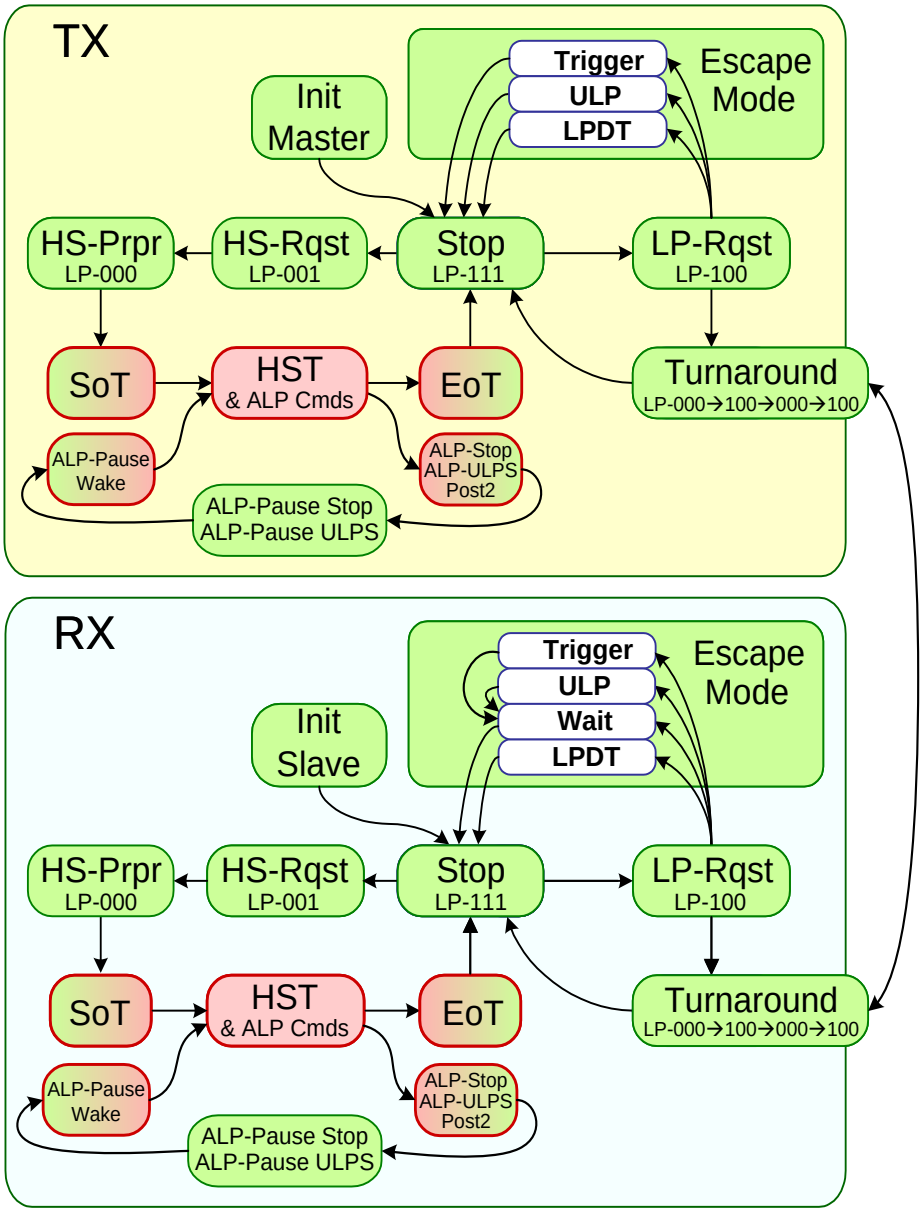


Figure 43 Lane Module State Diagram

6.14 Data Rate Dependent Parameters (Informative)

The High-Speed data transfer rate of the C-PHY may be programmable to values determined by a particular implementation. Any individual data transfer between SoT and EoT sequences must take place at a given, fixed rate. However, reprogramming the data rate of the C-PHY High-Speed transfer is allowed at initialization, before starting the exit from ULP state or in Stop state. The method of data rate reprogramming is out of the scope of this document.

6.14.1 Parameters Containing Only UI Values

Certain parameters are specified as a number of UI intervals. Often this shall be a multiple of 7 UI which simplifies the implementation of both the transmitter and receiver because decisions regarding the transmission of fields can take place at a word clock interval. Parameters specified in units of UI are:

- $t_{3\text{-PREBEGIN}}$ – the length of the first part of the Preamble
- $t_{3\text{-PROGSEQ}}$ – the length of the programmable sequence Section of the preamble
- $t_{3\text{-PREEND}}$ – the length of the end of the preamble
- $t_{3\text{-PREAMBLE}}$ – the length of the entire preamble including $t_{3\text{-BEGIN}}$, $t_{3\text{-PROGSEQ}}$ and $t_{3\text{-PREEND}}$
- $t_{3\text{-SYNC}}$ – the length of the Sync Word
- $t_{3\text{-POST}}$ – the length of the Post sequence at the end of the burst

6.14.2 Parameters Containing Time and UI values

There are no parameters specified as the sum of an explicit time and a number of UI.

6.14.3 Parameters Containing Only Time Values

Several parameters are specified only as explicit time values. These explicit time values are typically derived from the time needed to charge and discharge the interconnect and are, therefore, not data rate dependent. It is conceivable to use an analog timer or counter clocked by the UI to ensure the implementation satisfies these parameters. However, if these time values are implemented by counting UI only, then the count value is a function of the data rate and, therefore, must be changed when the data rate is changed.

The following parameters are based on time values alone:

- $t_{3\text{-PREPARE}}$
- $t_{3\text{-TERM-EN}}$
- $t_{3\text{-SETTLE}}$
- $t_{HS\text{-EXIT}}$

6.14.4 Parameters Containing Only Time Values That Are Not Data Rate Dependent

The remaining parameters in ***Table 24*** shall be complied with even when the High-Speed clock is off. These parameters include low-power and initialization state durations and LP signaling intervals. Though these parameters are not HS data rate dependent, some implementations of C-PHY may need to adjust these values when the data rate is changed.

7 Fault Detection

There are three different mechanisms to detect malfunctioning of the Link. Bus contention and error detection functions are contained within the C-PHY. These functions should detect many typical faults. However, some faults cannot be detected within the C-PHY and require a protocol-level solution. Therefore, the third detection mechanism is a set of application specific watchdog timers.

7.1 Contention Detection

If a Bi-directional Lane Module and a Unidirectional module are combined in one Lane, only Unidirectional functionality is available. Because in this case the additional functionality of one Bi-directional PHY Module cannot be reliably controlled from the limited functionality PHY side, the Bi-directional features of the Bi-directional Module shall be safely disabled. Otherwise in some cases deadlock may occur which can only be resolved with a system power-down and re-initialization procedure.

During normal operation one and only one side of a Link shall drive a Lane at any given time except for certain transition periods. Due to errors or system malfunction a Lane may end up in an undesirable state, where the Lane is driven from two sides or not driven at all. This condition eventually results in a state conflict and is called Contention.

All Lane Modules with LP Bi-directionality shall include contention detection functions to detect the following contention conditions:

- Modules on both sides of the same Line drive opposite LP levels against each other. In this case, the Line voltage will settle to some value between $V_{OL,MIN}$ and $V_{OH,MAX}$. Because V_{IL} is greater than V_{IHCD}, the settled value will always be either higher than V_{IHCD}, lower than V_{IL}, or both. Refer to ***Section 9.3***. This ensures that at least one side of the Link, possibly both, will detect the fault condition.
- The Module at one side drives LP-high while the other side drives HS-low on the same Line. In this case, the Line voltage will settle to a value lower than V_{IL}. The contention shall be detected at the side that is transmitting the LP-high.

The first condition can be detected by the combination of LP-CD and LP-RX functions. The LP-RX function should be able to detect the second contention condition. Details on the LP-CD and LP-RX electrical specifications can be found in ***Section 9***. Except when the previous state was TX-ULPS, contention shall be checked before the transition to a new state. Contention detection in ULPS is not required because the bit period is not defined and a clock might not be available.

After contention has been detected, the protocol layer shall take proper measures to resolve the situation.

7.2 Sequence Error Detection

If for any reason the Lane signal is corrupted the receiving PHY may detect signal sequence errors. Errors detected inside the PHY may be communicated to the protocol layer via the PPI. This kind of error detection is optional, but strongly recommended as it enhances reliability. The following sequence errors can be distinguished:

- SoT Error
- SoT Sync Error
- EoT Sync Error
- Escape Entry Command Error
- LP Transmission Sync Error
- False Control Error

7.2.1 SoT Error

The Sync Word for Start of High-Speed Transmission is fault tolerant of errors in the least significant symbol, as described in ***Section 6.4.4.1***. Therefore, the Sync Word is usable to establish word boundary synchronization and identification of the start of data, but confidence in the payload data is lower. If this situation occurs, an SoT Error is indicated.

7.2.2 SoT Sync Error

If the Sync Word is corrupted in a way that proper synchronization cannot be expected, a SoT Sync Error is indicated.

7.2.3 EoT Sync Error

The EoT Sync Error is not applicable to C-PHY. Redundant Post words follow the packet data, and an error in the misalignment of data words is easily detected by the upper layer protocol.

7.2.4 Escape Mode Entry Command Error

If the receiving Lane Module does not recognize the received entry command for Escape Mode an Escape Mode entry command error is indicated.

7.2.5 LP Transmission Sync Error

At the end of a low-power data transmission procedure, if data is not synchronized to a Byte boundary an Escape Sync Error signal is indicated.

7.2.6 False Control Error

If a LP-Rqst (LP-100) is not followed by the remainder of a valid escape or Turnaround sequence, a False Control Error is indicated. This error is also indicated if a HS-Rqst (LP-001) is not correctly followed by a Bridge State (LP-000).

7.3 Protocol Watchdog Timers (Informative)

It is not possible for the PHY to detect all fault cases. Therefore, additional protocol-level time-out mechanisms are necessary in order to limit the maximum duration of certain modes and states.

7.3.1 HS RX Timeout

In HS RX mode if no EoT is received within a certain period the protocol layer should time-out. The timeout period can be protocol specific.

7.3.2 HS TX Timeout

The maximum transmission length in HS TX is bounded. The timeout period is protocol specific.

7.3.3 Escape Mode Timeout

A device may timeout during Escape Mode. The timeout should be greater than the Escape Mode Silence Limit of the other device. The timeout period is protocol specific.

7.3.4 Escape Mode Silence Timeout

A device may have a bounded length for LP TX-000 during Escape Mode, after which the other device may timeout. The timeout period is protocol specific. For example, a display module should have an Escape Mode Silence Limit, after which the host processor can timeout.

7.3.5 Turnaround Errors

A Turnaround procedure always starts from a Stop state. The procedure begins with a sequence of Low-Power States ending with a Bridge State (LP-000) during which drive sides are swapped. The procedure is finalized by the response including a Turn State followed by a Stop state driven from the other side. If the actual sequence of events violates the normal Turnaround procedure a "False Control Error" may be flagged to the protocol layer. See ***Section 7.2.6***. The Turn State response serves as an acknowledgement for the correctly completed Turnaround procedure. If no acknowledgement is observed within a certain time period the protocol layer should time-out and take appropriate action. This period should be larger than the maximum possible Turnaround time for a particular system. There is no time-out for this condition in the PHY.

8 Interconnect and Lane Configuration

The interconnect between transmitter and receiver carries all signals used in C-PHY communication. This includes both High-Speed, low voltage signaling I/O technology and low speed, low power signaling for control functions. For this reason, the physical connection should be implemented by means of point-to-point transmission lines referenced to ground. The total interconnect may consist of several cascaded transmission line segments, such as, printed circuit boards, flex-foils, and cable connections.

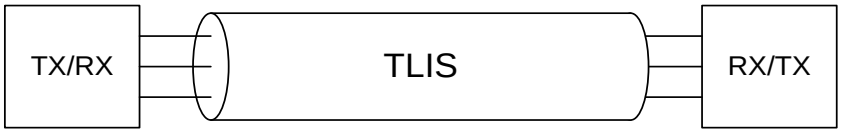


Figure 44 Point-to-Point Interconnect

8.1 Lane Configuration

The complete physical connection of a Lane consists of a transmitter (TX), and/or receiver (RX) at each side, with some Transmission-Line-Interconnect-Structure (TLIS) in between. The overall Lane performance is therefore determined by the combination of these three elements. The split between these elements is defined to be on the module (IC) pins. This Section defines both the required performance of the Transmission-Line-Interconnect-Structure for the signal routing as well as the I/O-cell Reflection properties of TX and RX. This way the correct overall operation of the Lane can be ensured.

With respect to physical dimensions, the Transmission-Line-Interconnect-Structure will typically be the largest part. Besides printed circuit board and flex-foil traces, this may also include elements such as vias and connectors.

8.2 Boundary Conditions

The reference characteristic impedance level is 100 Ω differential, 50 Ω single-ended per Line, and 25 Ω common-mode for any two lines together. The 50 Ω impedance level for single-ended operation is also convenient for test and characterization purposes.

This typical impedance level is required for all three parts of the Lane: TX, TLIS, and RX. The tolerances for characteristic impedances of the interconnect and the tolerance on Line termination impedances for TX and RX are specified by means of S-parameter templates over the whole operating frequency range.

The differential channel is also used for LP single-ended signaling. Therefore, it is strongly recommended to apply only very loosely coupled differential transmission lines.

8.3 Definitions

The frequency ‘fh’ is the highest fundamental frequency for data transmission and is equal to $1/(2*UI_{INST,MIN})$. Implementers should specify a value $UI_{INST,MIN}$ that represents the minimum instantaneous UI possible within a High-Speed data transfer for a given implementation.

The frequency ‘$f_{LP,MAX}$’ is the maximum toggle frequency for low-power mode.

RF interference frequencies are denoted by ‘f_{INT}’, where $f_{INT,MIN}$ defines the lower bound for the band of relevant RF interferers. The frequency f_{MAX} is defined by

$$f_{MAX} = \frac{3}{4} \cdot \frac{1}{UI_{INST,MIN}}$$

8.4 S-Parameter Specifications

The required performance of the physical connection is specified by means of S-parameter requirements for TX, TLIS, and RX, for TLIS by mixed-mode, 4-port parameters, and for RX and TX by mixed-mode, reflection (return loss) parameters. The S-parameter limits are defined over the whole operating frequency range by means of templates.

The differential transmission properties are most relevant and therefore this specification uses mixed-mode parameters. As the performance needs depend on the targeted bit rates, most S-parameter requirements are specified on a normalized frequency axis with respect to bit rate. Only the parameters that are important for the suppression of external (RF) interference are specified on an absolute frequency scale. This scale extends up to f_{MAX}. Beyond this frequency the circuitry itself should suppress the high-frequency interference signals sufficiently.

Only the overall performance of the TLIS and the maximum reflection of RX and TX are specified. This fully specifies the signal behavior at the RX/TX-module pins. The subdivision of losses, reflections and mode-conversion budget to individual physical fractions of the TLIS is left to the system designer. Annex B includes some rules of thumb for system design and signal routing guidelines.

8.5 Characterization Conditions

All S-parameter definitions are based on a 50 Ω impedance reference level. The characterization can be done with a measurement system, as shown in ***Figure 45***.

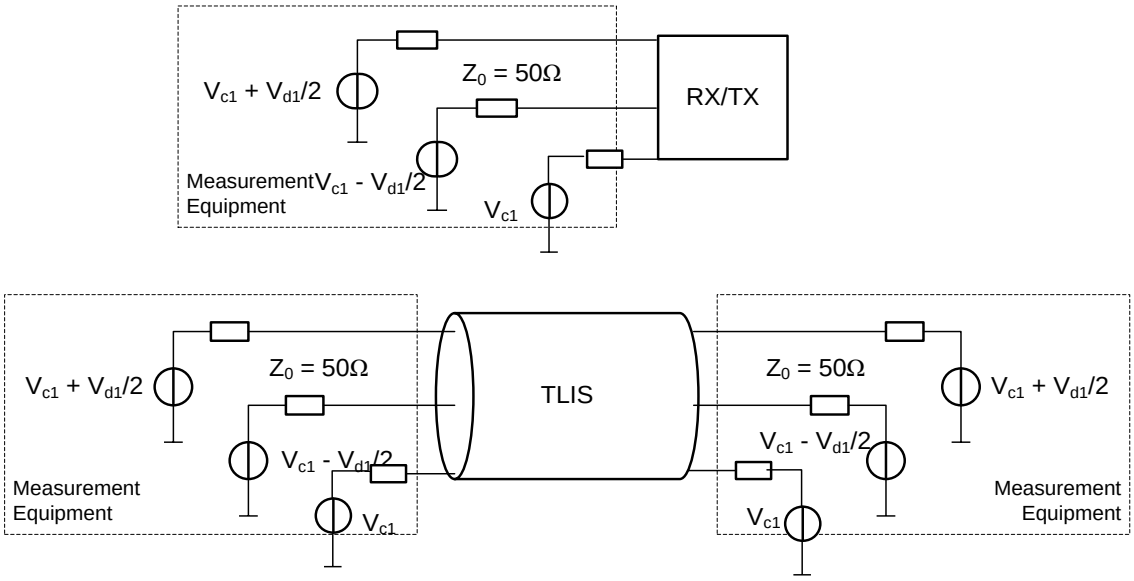


Figure 45 Set-up for S-Parameter Characterization of RX, TX and TLIS

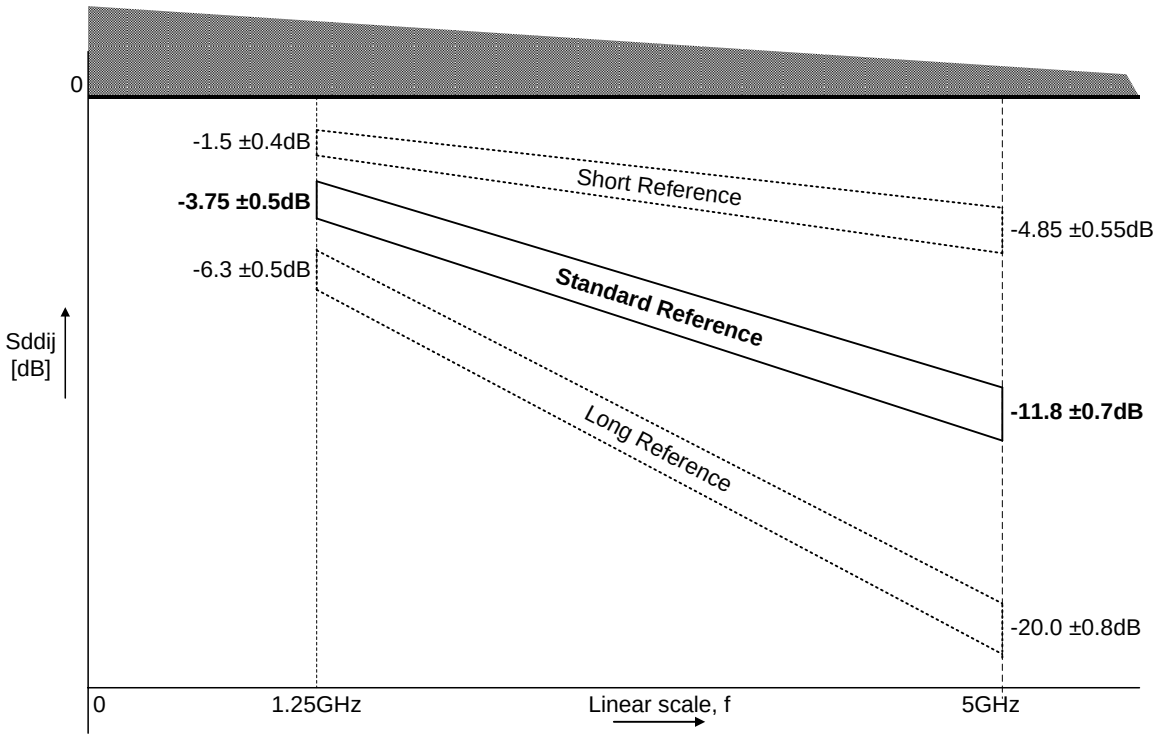
8.6 Interconnect Specifications

The Transmission-Line Signal-Routing (TLSR) is specified by means of mixed-mode 4-port S-parameter behavior templates over the frequency range. This includes the differential and common-mode, insertion and return losses, and mode-conversion limitations.

8.6.1 Differential Characteristics

8.6.1.1 Differential Insertion Loss

The differential transfer behavior (insertion loss) of the TLIS should meet the Sdd21 template shown in ***Figure 46***, where i ≠ j. This applies to all the three differential pairs AB, BC & CA. The frequency range over which the insertion loss is valid is shown in the figure.



Frequency	1.25GHz	5.0GHz
Short Reference Sddij, dB	-1.5dB ±0.4dB	-4.85dB ±0.55dB
Standard Reference Sddij, dB	**-3.75dB ±0.5dB**	**-11.8dB ±0.7dB**
Long Reference Sddij, dB	-6.3dB ±0.5dB	-20.0dB ±0.8dB

Figure 46 Template for Differential Insertion Losses

Three Reference channels (Short, Standard and Long) are defined to support a wide range of display and camera applications.

The Standard Reference Channel is the default channel configuration; transmitters and receivers should support it.

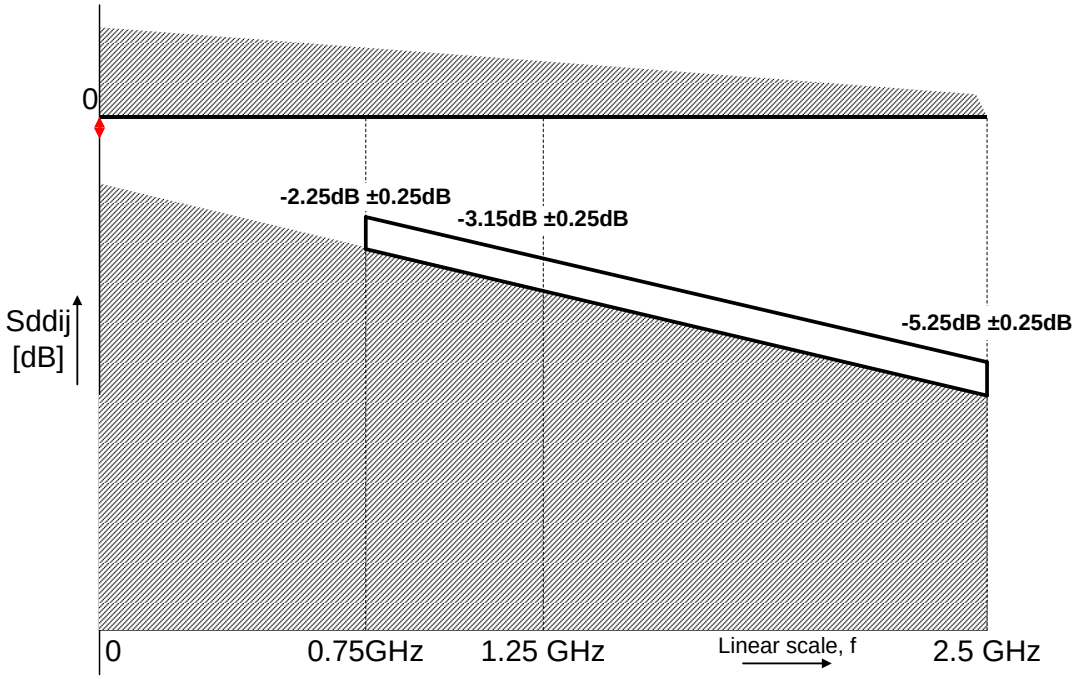
Short Reference Channel support is optional. It can be used in applications needing lower interconnect loss.

Long Reference Channel support is optional. The data rate needs to be limited when this interconnect model is used.

The Long Reference Channel is intended to model some Chip-On-Glass (COG) interconnect configurations. COG interconnect is used for display panels and has reduced cost compared to other solutions. However, it increases the total loss of the interconnect due to additional routing on the glass, bonding between the glass and flex cables, and bonding between the glass and silicon.

Specific guidance on usage of these reference channels is provided in ***Section 10.3.3***.

The channel described in MIPI C-PHY v1.0, described now as the C-PHY Legacy Channel, is illustrated in ***Figure 47***.



	0.75 GHz	1.25 GHz	2.5 GHz
Sddij, dB	-2.25 ±0.25dB	-3.15 ±0.25dB	-5.25 ±0.25dB

Figure 47 Template for Differential Insertion Losses, C-PHY Legacy Channel

8.6.1.2 Differential Reflection Loss

The differential reflection for both ports of the TLIS is specified by Sdd11 and Sdd22, and should be less than -12dB from 0 to f_{MAX}. Not meeting the differential reflection coefficients might impact interoperability and operation.

8.6.2 Common-Mode Characteristics

The common-mode insertion loss is implicitly specified by means of the differential insertion loss and the intra-Lane cross coupling. The requirements for common-mode insertion loss are therefore equal to the differential requirements.

The common-mode reflection coefficients Scc11 and Scc22 should both be below –12 dB at frequencies up to 2fh. Not meeting the common-mode reflection coefficients might impact interoperability and operation.

8.6.3 Intra-Lane Cross-Coupling

The two lines applied as a differential pair during HS transmission are also used individually for single-ended signaling during low-power mode. Therefore, the coupling between the two wires should be restricted in order to limit single-ended cross coupling. The coupling between the two wires is defined as the difference of the S-parameters Scc21 and Sdd21 or Scc12 and Sdd12. In either case, the difference should not exceed –20 dB for frequencies up to $10f_{LP,MAX}$.

8.6.4 Mode-Conversion Limits

All mixed-mode, 4-port S-parameters for differential to common-mode conversion, and vice-versa, should not exceed –29 dB for frequencies below f_{MAX}. This includes Sdc12, Scd21, Scd12, Sdc21, Scd11, Sdc11, Scd22, and Sdc22.

8.6.5 Inter-Lane Static Skew

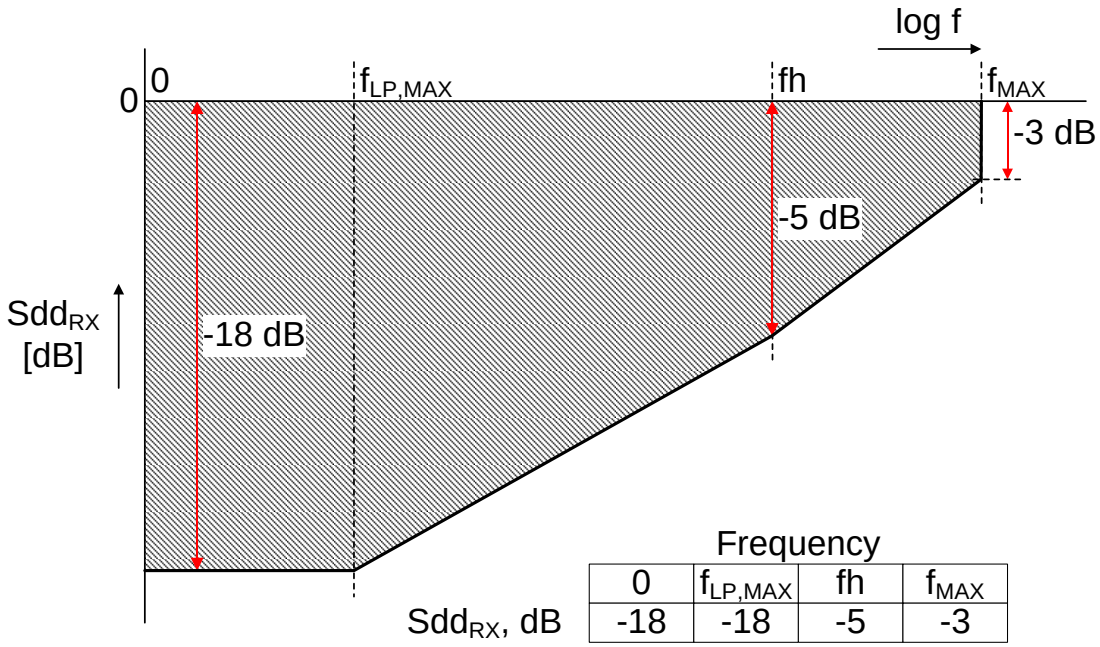
The difference in signal delay between any two Lanes should be less than 160ps for all frequencies up to, and including, fh.

8.7 Driver and Receiver Characteristics

Besides the TLIS the Lane consists of two RX-TX modules, one at each side. This paragraph specifies the reflection behavior (return loss) of these RX-TX modules in HS-mode. This applies to all the three differential pairs AB, BC & CA. The signaling characteristics of all possible functional blocks inside the RX-TX modules can be found in ***Section 9***.

8.7.1 Differential Characteristics

The differential reflection of a Lane Module in High-Speed RX mode should conform to the limits specified by the template shown in ***Figure 48***.



Frequency	0	$f_{LP,MAX}$	fh	f_{MAX}
Sdd_{RX}, dB	-18	-18	-5	-3

Figure 48 Differential Reflection Template for Lane Module Receivers

The differential reflection of a Lane Module in High-Speed TX mode should conform to the limits specified by the template shown in ***Figure 49***.

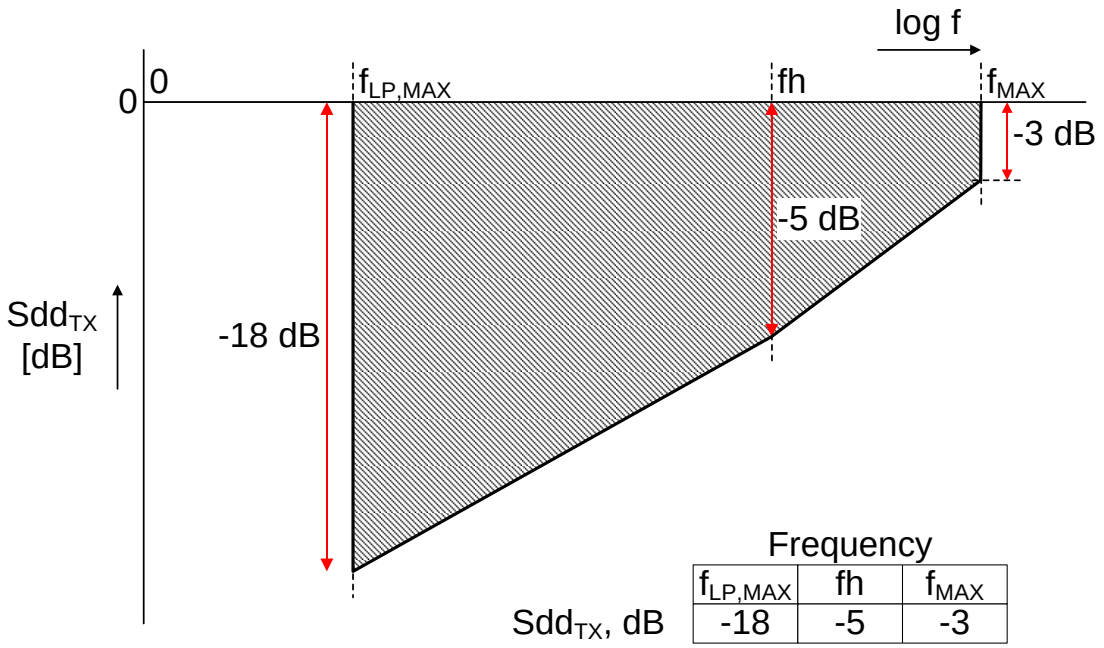


Figure 49 Differential Reflection Template for Lane Module Transmitters

8.7.2 Common-Mode Characteristics

The common-mode return loss specification is different for a High-Speed TX and RX mode, because the RX is not DC terminated to ground. The common mode reflection of a Lane Module in High-Speed TX mode should be less than -3dB from $f_{LP,MAX}$ up to f_{MAX}. The common mode reflection of a Lane Module in High-Speed RX mode should conform to the limits specified by the template shown in ***Figure 50***. Assuming a high DC common-mode impedance implies a sufficiently large capacitor at the termination center tap. The minimum value allows integration. While the common-mode termination is especially important for reduced influence of RF interferers, the RX requirement limits reflection for the most relevant frequency band.

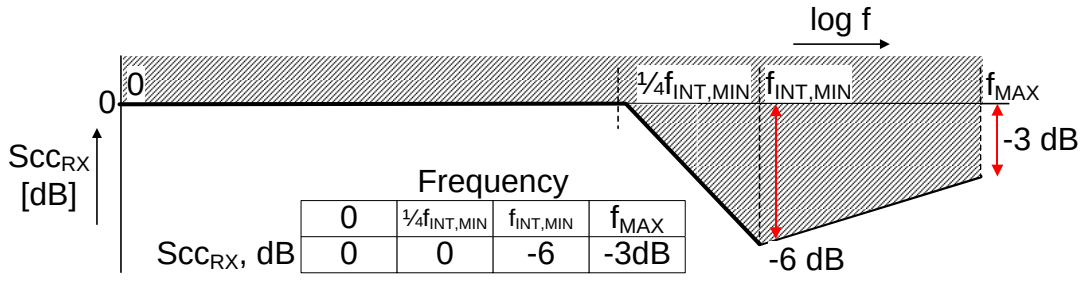


Figure 50 Template for RX Common-Mode Return Loss

8.7.3 Mode-Conversion Limits

The differential to common-mode conversion limits of RX should be -26dB up to f_{MAX}.

9 Electrical Characteristics

A PHY may contain the following electrical functions: a High-Speed Transmitter (HS-TX), a High-Speed Receiver (HS-RX), a low-power transmitter (LP-TX), a low-power receiver (LP-RX), and a low-power contention detector (LP-CD). A PHY does not need to contain all electrical functions, only the functions that are required for a particular PHY Configuration. The required functions for each configuration are specified in ***Section 5***. All electrical functions included in any PHY shall meet the specifications in this Section. ***Figure*** 51 shows the complete set of electrical functions required for a fully featured PHY transceiver.

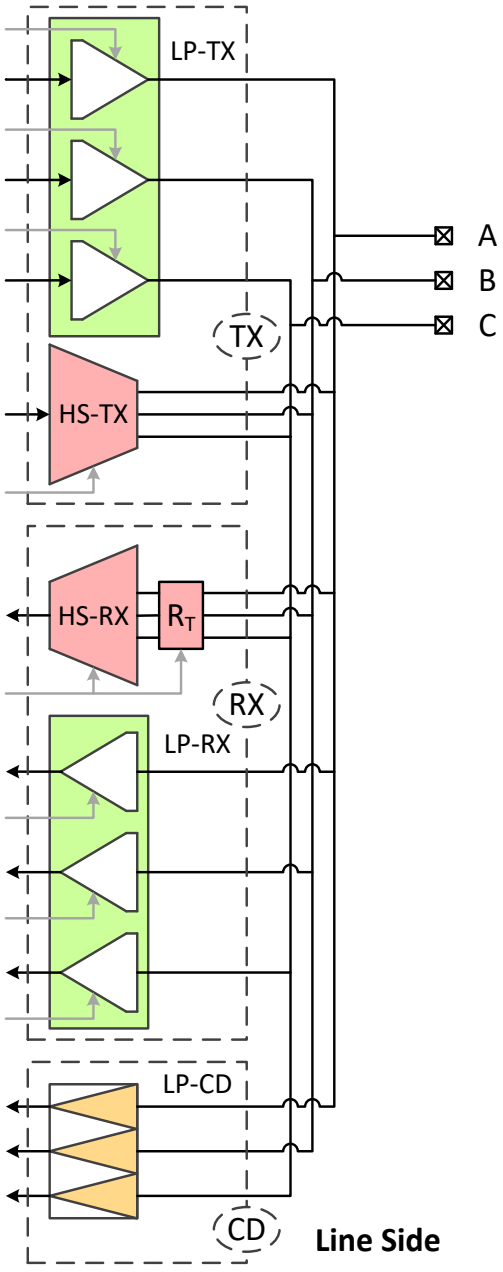


Figure 51 Electrical Functions of a Fully Featured C-PHY Transceiver

The HS transmitter and HS receiver are used for the transmission of the HS data signals. The HS transmitter and receiver utilize low-voltage C-PHY signaling for signal transmission. The HS receiver contains a switchable star termination.

The LP transmitter and LP receiver serve as a low power signaling mechanism. The LP transmitter is a push-pull driver and the LP receiver is an un-terminated, single-ended receiver.

The signal levels are different for High-Speed mode compared to single-ended low-power mode. ***Figure 52*** shows both the High-Speed and low-power signal levels on the left and right sides, respectively. The High-Speed signaling levels are below the low-power low-level input threshold such that low-power receiver always detects logic low level when High-Speed signals are present.

All absolute voltage levels are relative to the ground voltage at the transmit side.

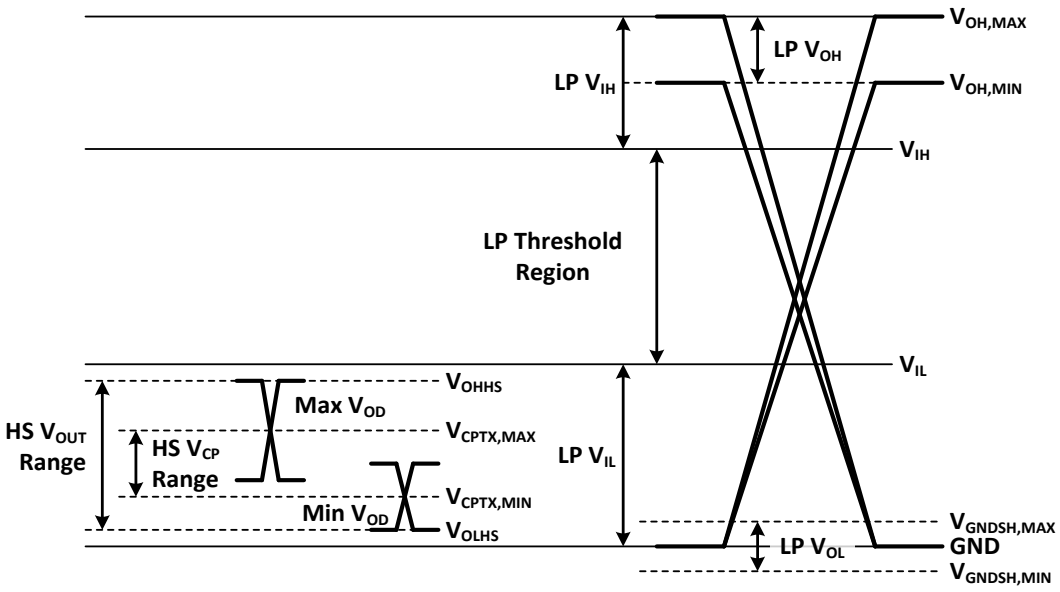


Figure 52 C-PHY Signaling Levels

A Lane switches between low-power and High-Speed mode during normal operation. Bi-directional Lanes can also switch communication direction. The change of operating mode or direction requires enabling and disabling of certain electrical functions. These enable and disable events shall not cause glitches on the lines that would result in a detection of an incorrect signal level. Therefore, all mode and direction changes shall be smooth to always ensure a proper detection of the Line signals.

9.1 Driver Characteristics

9.1.1 High-Speed Transmitter

A High-Speed C-PHY signal driven on the A, B and C pins is generated by a High-Speed output driver. ***Table*** 33 is a summary of the six possible High-Speed Wire States that can be driven on a C-PHY Lane. ***Figure 53*** shows two example implementations of a High-Speed transmitter. The “T2” Driver Type that presents a valid output impedance at the HS Mid level is the recommended implementation. At slow speeds the “T1” Driver Type may be used if the parametric requirements of ***Section 9*** and ***Section 10*** can be met.

Table 33 C-PHY High-Speed Wire States

State Code	Line Voltage Levels		
	A Line	B Line	C Line
HS_+X	HS High	HS Low	HS Mid
HS_-X	HS Low	HS High	HS Mid
HS_+Y	HS Mid	HS High	HS Low
HS_-Y	HS Mid	HS Low	HS High
HS_+Z	HS Low	HS Mid	HS High
HS_-Z	HS High	HS Mid	HS Low

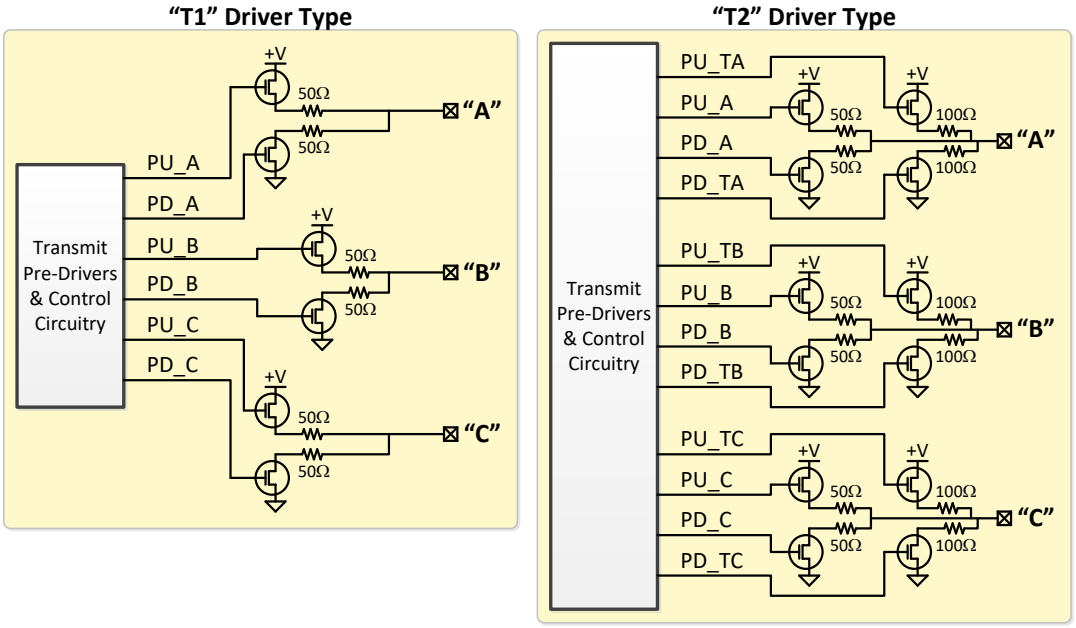


Figure 53 Example High-Speed Transmitter

The single-ended output voltages are defined V_A, V_B and V_C at the A, B and C pins, respectively. The differential output voltages V_{OD_AB}, V_{OD_BC} and V_{OD_CA} are defined as the difference of the voltages: V_A minus V_B, V_B minus V_C, and V_C minus V_A, respectively.

$$V_{OD\_AB} = V_A - V_B;\ V_{OD\_BC} = V_B - V_C;\ V_{OD\_CA} = V_C - V_A;$$

The output voltages V_A, V_B and V_C at the A, B and C pins shall not exceed the High-Speed output high voltage V_{OHHS}. V_{OLHS} is the High-Speed output, low voltage on A, B and C, and is determined by V_{OD_AB}, V_{OD_BC}, V_{OD_CA} and V_{CPTX}. The High-Speed V_{OUT} is bounded by the minimum value of V_{OLHS} and the maximum value of V_{OHHS}.

The common-point voltage V_{CPTX} is defined as the arithmetic mean value of the voltages at the A, B and C pins:

$$V_{CPTX} = \frac{V_A + V_B + V_C}{3}$$

V_{OD_AB}, V_{OD_BC} and V_{OD_CA} and V_{CPTX} are shown graphically in ***Figure 54*** for ideal High-Speed signals. ***Figure 55*** shows single-ended High-Speed signals with the possible kinds of distortion of the differential output and common-point voltages. The strong one and zero levels of V_{OD_AB}, V_{OD_BC} and V_{OD_CA}, and V_{CPTX} may be slightly different for driving any of the six possible Wire States on the Lane. The strong one and strong zero states for a given wire pair occur only in certain states, and it is the strong levels that are considered to determine ΔV_{OD}. ***Table 34*** shows which High-Speed states produce the strong levels for each wire pair.

Table 34 Strong Zero and Strong One State for Each Wire Pair

Wire Pair	Strong Zero State	Strong One State	Weak Zero States	Weak One States
AB	HS_-X	HS_+X	HS_+Y, HS_+Z	HS_-Y, HS_-Z
BC	HS_-Y	HS_+Y	HS_+X, HS_+Z	HS_-X, HS_-Z
CA	HS_-Z	HS_+Z	HS_+X, HS_+Y	HS_-X, HS_-Y

The output differential voltage mismatch, ΔV_{OD}, is defined as the difference of the maximum and minimum of: the absolute values of the differential strong one and strong zero output voltages of the three possible wire pairs. This is expressed by the following equations that consider the V_{OD} for a particular wire pair in a specific state as described in ***Table 34***:

$$V_{OD\_MAX} = max\left(V_{OD\_AB\_+X}, \left|V_{OD\_AB\_-X}\right|, V_{OD\_BC\_+Y}, \left|V_{OD\_BC\_-Y}\right|, V_{OD\_CA\_+Z}, \left|V_{OD\_CA\_-Z}\right|\right)$$

$$V_{OD\_MIN} = min\left(V_{OD\_AB\_+X}, \left|V_{OD\_AB\_-X}\right|, V_{OD\_BC\_+Y}, \left|V_{OD\_BC\_-Y}\right|, V_{OD\_CA\_+Z}, \left|V_{OD\_CA\_-Z}\right|\right)$$

$$\Delta V_{OD} = V_{OD\_MAX} - V_{OD\_MIN}$$

If $V_{CPTX(HS\_+X)}$, $V_{CPTX(HS\_-X)}$, $V_{CPTX(HS\_+Y)}$, $V_{CPTX(HS\_-Y)}$, $V_{CPTX(HS\_+Z)}$, and $V_{CPTX(HS\_-Z)}$ are the common-point voltages for static HS_+X, HS_-X, HS_+Y, HS_-Y, HS_+Z and HS_-Z states, respectively, then the common-point reference voltage is defined as:

$$V_{VCPTX,REF} =$$

$$\frac{V_{VCPTX(HS\_+X)} + V_{VCPTX(HS\_-X)} + V_{VCPTX(HS\_+Y)} + V_{VCPTX(HS\_-Y)} + V_{VCPTX(HS\_+Z)} + V_{VCPTX(HS\_-Z)}}{6}$$

The transient common-point voltage variation is defined by:

$$\Delta V_{CPTX}(t) = V_{CPTX}(t) - V_{CPTX,REF}$$

The static common-point voltage mismatch between the six High-Speed states is defined as:

$$V_{MAXCP} = max\left(V_{CPTX(HS\_+X)}, V_{CPTX(HS\_-X)}, V_{CPTX(HS\_+Y)}, V_{CPTX(HS\_-Y)}, V_{CPTX(HS\_+Z)}, V_{CPTX(HS\_-Z)}\right)$$

$$V_{MINCP} = min\left(V_{CPTX(HS\_+X)}, V_{CPTX(HS\_-X)}, V_{CPTX(HS\_+Y)}, V_{CPTX(HS\_-Y)}, V_{CPTX(HS\_+Z)}, V_{CPTX(HS\_-Z)}\right)$$

$$\Delta V_{CPTX(HS)} = \frac{V_{MAXCP} - V_{MINCP}}{2}$$

The transmitter shall send data such that the high frequency and low frequency common-point voltage variations do not exceed $\Delta V_{CPTX(HF)}$ and $\Delta V_{CPTX(LF)}$, respectively. An example test circuit for the measurement of V_{OD} and V_{CPTX} is shown in ***Figure 56***.

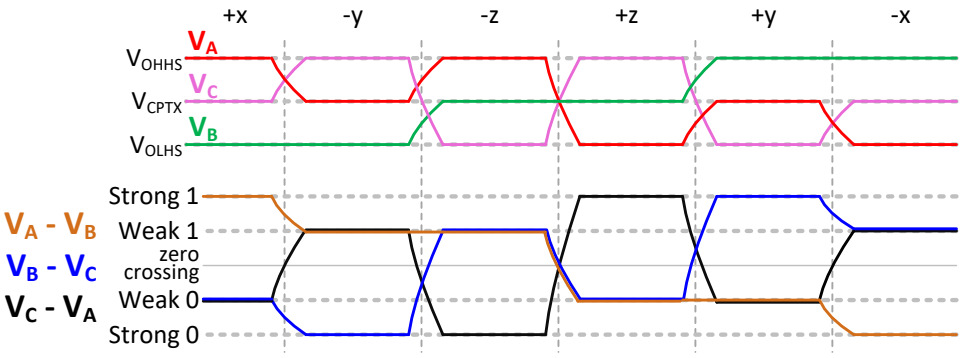
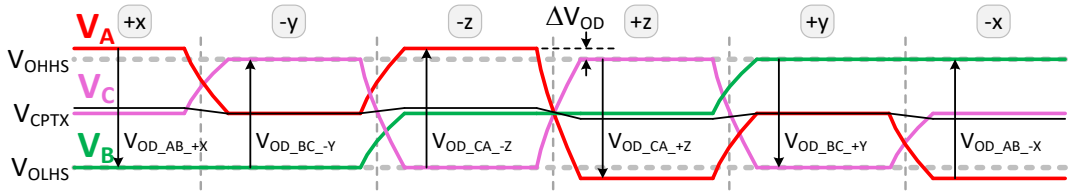
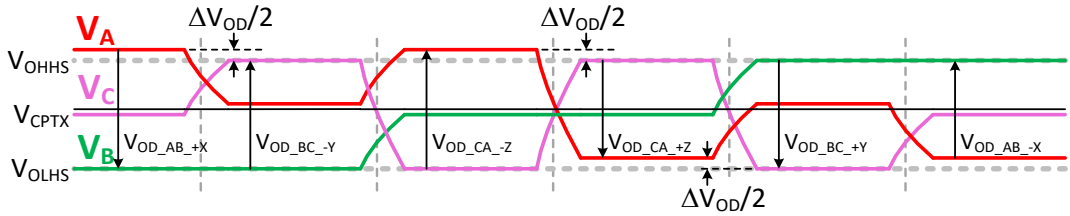


Figure 54 Ideal Single-Ended and Resulting Differential High-Speed Signals

Large V_A Amplitude (single-ended high-speed signals)



Fixed Offset V_A (single-ended high-speed signals)



Slow Rise/Fall V_A (single-ended high-speed signals)

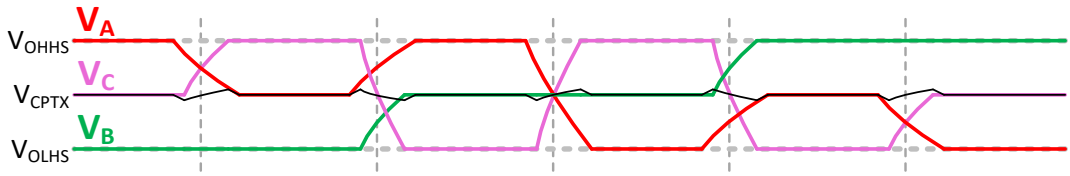


Figure 55 Possible V_{CPTX} and ΔV_{OD} Distortions of the Single-Ended HS Signals

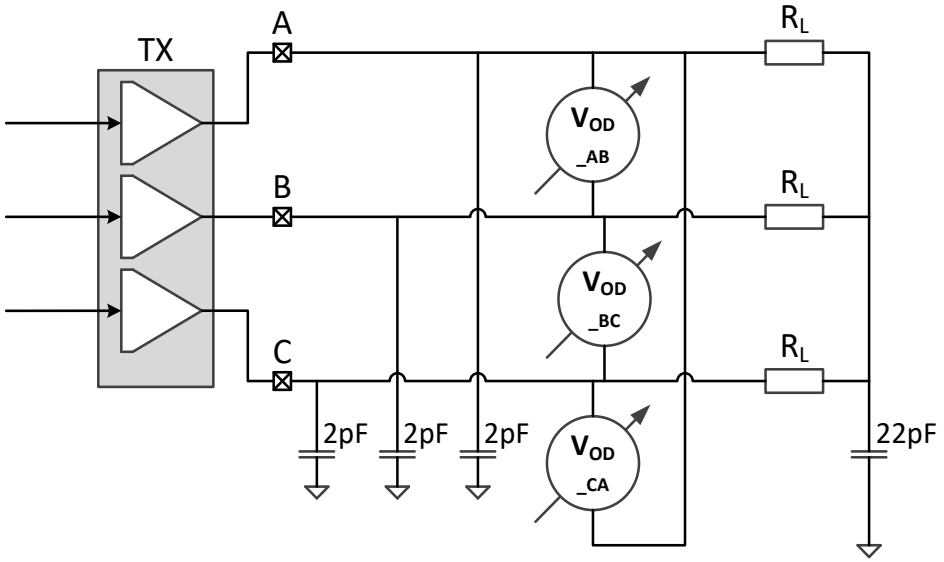


Figure 56 Example Circuit for V_{OD} and V_{CPTX} Measurements

The single-ended output impedance of the transmitter at the A, B and C pins is denoted by Z_{OS}. ΔZ_{OS} is the mismatch of the single ended output impedances at the A, B and C pins, denoted by Z_{OS_A}, Z_{OS_B} and Z_{OS_C}, respectively. This mismatch is defined as the ratio of the difference between the largest and smallest value of Z_{OS_A}, Z_{OS_B} and Z_{OS_C} and the average of those impedances:

$$\Delta Z_{OS} = 3 \cdot \frac{max(Z_{OS\_A}, Z_{OS\_B}, Z_{OS\_C}) - min(Z_{OS\_A}, Z_{OS\_B}, Z_{OS\_C})}{Z_{OS\_A} + Z_{OS\_B} + Z_{OS\_C}}$$

The output impedance Z_{OS} and the output impedance mismatch ΔZ_{OS} shall be compliant with ***Table 35*** for all six possible High-Speed Wire States and for all allowed loading conditions. It is recommended that implementations keep the output impedance during state transitions as close as possible to the steady state value. The output impedance Z_{OS} can be determined by injecting an AC current into the A, B and C pins and measuring the peak-to-peak voltage amplitude.

The specifications for transmit timing can be found in ***Section 10***. The specifications for TX common-mode return loss and the TX differential mode return loss can be found in ***Section 8***.

It is recommended that a High-Speed transmitter that is directly terminated at its pins should not generate any overshoot in order to minimize EMI.

The transmitter may support a Low-Voltage High-Speed Mode (called the LVHS Mode). The receiver specifications are not different in the LVHS Mode, and there is not an Unterminated Mode defined for the LVHS Mode. It is not intended that the transmitter will change between the LVHS Mode and normal High-Speed mode while transmitting High-Speed data. Transmitters that support the LVHS Mode will most likely use a 400mV supply rail to power the output driver stage. The LVHS Mode is only different from the normal High-Speed mode by its lower output voltage characteristics, which are described in ***Table 35*** (different common-point voltage and weak transmit differential voltage). In the normal High-Speed mode a transmitter shall comply with the V_{CPTX} and |VOD| weak parameters in ***Table 35***, and the operating symbol rate guidance is given in ***Table 58***. In the LVHS Mode a transmitter shall comply with the $V_{CPTX(LVHS)}$ and |VOD| weak (LVHS) parameters in ***Table 35***, and the operating symbol rate guidance is given in ***Table 60***. The Tx Eye Diagram requirements described in ***Section 10.3.1*** apply to normal High-Speed mode and to LVHS Mode. A transmitter may support only the Low-Voltage High-Speed Mode without supporting the normal High-Speed mode, or a transmitter may support only the normal High-Speed mode without supporting the Low-Voltage

High-Speed Mode, or a transmitter may support a mode switch so that it can operate either in the normal High-Speed mode or in the LVHS Mode.

The manufacturer shall specify in the component data sheet or other similar literature whether the LVHS Mode is implemented in the transmitter. The expected performance of the LVHS Mode is provided in ***Section 10.3.3***, but higher speed operation is possible if certain transmitter specifications are exceeded. The manufacturer may specify operating symbol rates higher or lower than the values provided in ***Section 10.3.3***.

1451 **Table 35 HS Transmitter DC Specifications**

Parameter	Description	Min	Nom	Max	Units	Notes
V_{CPTX}	HS transmit static common-point voltage	175	225 to 250	310	mV	1, 3
$V_{CPTX(LVHS)}$	HS transmit static common-point voltage, Low-Voltage High-Speed Mode	150	200	255	mV	1
$\lvert\Delta V_{CPTX(HS)}\rvert$	V_{CPTX} mismatch when output is in any of the six High-Speed states	–	–	9	mV	2
$\lvert V_{OD}\rvert$ strong	HS transmit differential voltage of the differential strong one and strong zero specified in ***Table 34***.	–	–	300	mV	1
$\lvert V_{OD}\rvert$ weak	HS transmit differential voltage of the differential weak one and weak zero specified in ***Table 34***.	97	–	–	mV	1
$\lvert V_{OD}\rvert$ Weak (LVHS)	HS transmit differential voltage of the differential weak one and weak zero specified in ***Table 34***, Low-Voltage High-Speed Mode.	70	–	–	mV	1
$\lvert\Delta V_{OD}\rvert$	V_{OD} mismatch between the absolute values of the differential strong one and strong zero output voltages in any of the six possible High-Speed states.	–	–	17	mV	2
V_{OHHS}	HS output high voltage	–	–	425	mV	1
Z_{OS}	Single ended output impedance	40	50	60	Ω	–
ΔZ_{OS}	Single ended output impedance mismatch	–	–	10	%	–

Note:

1. *Value when driving into load impedance, Z_{ID}, equal to 100 Ω.*
2. *A transmitter should minimize ΔV_{OD} and $\Delta V_{CPTX(HS)}$ in order to minimize radiation, and optimize signal integrity.*
3. *Typical value of V_{CPTX} should be in the specified range depending upon the supply voltage used for each implementation.*

1452 **Table 36 HS Transmitter AC Specifications**

Parameter	Description	Min	Nom	Max	Units	Notes
$\Delta V_{CPTX(HF)}$	Common-level variations above 450 MHz	–	–	15	mV_{RMS}	–
$\Delta V_{CPTX(LF)}$	Common-level variation between 50 MHz and 450 MHz	–	–	25	mV_{PEAK}	–

9.1.1.1 Tx HS Unterminated Mode

1453 To reduce the device power consumption, the system designer may choose the option to operate the Link at
1454 a reduced speed in an unterminated mode. In this mode, the transmitter has the conventional termination but
1455 the receiver termination is not enabled. This is called the HS Unterminated Mode. The HS Unterminated
1456 Mode described in this Section is optional normative. If the HS Unterminated Mode is implemented in the
1457 transmitter then it shall be implemented as defined in the C-PHY specification. The manufacturer shall
1458 specify in the component data sheet or other similar literature whether the HS Unterminated Mode is
1459 implemented in the transmitter.

1460 When operating in the HS Unterminated Mode it is highly recommended that the Transmitter use the “T2”
1461 Driver Type shown in ***Figure 53***. This is so that the HS Mid level is driven to a deterministic level when the
1462 receiver termination is not present.

There are no changes to the PHY protocols described in ***Section 5*** and ***Section 6*** for HS Unterminated Mode, except that the enabling and disabling of the receive termination is not applicable.

The transmitter and receiver shall both be operated in the same mode, e.g. both shall have HS Unterminated Mode disabled, or both shall have HS Unterminated mode enabled. A mixed configuration with HS Unterminated Mode enabled in one and not the other is not allowed.

It is possible that emissions will be higher in Unterminated Mode compared to the standard mode of operation due to signal reflections in the unterminated transmission lines. It is the responsibility of the system designer to take proper precautions to meet the appropriate regulatory requirements and to minimize self-interference with co-located wireless communications transceivers.

The transmitter circuit in ***Figure 56*** is also applicable for the parameters defined in ***Table 37***.

Table 37 HS Transmitter DC Specifications for HS Unterminated Mode

Parameter	Description	Min	Nom	Max	Units	Notes
$V_{CPTX(UT)}$	HS transmit static common-point voltage, V_{CPTX}, for Unterminated Mode	150	197	245	mV	1
$\|V_{OD(UT)}\|$ strong	HS transmit differential voltage of the differential strong one and strong zero specified in ***Table 34***, in Unterminated Mode.	–	–	425	mV	1
$\|V_{OD(UT)}\|$ weak	HS transmit differential voltage of the differential weak one and weak zero specified in ***Table 34***, in Unterminated Mode.	180	–	–	mV	1
$\|\Delta V_{OD(UT)}\|$	V_{OD} mismatch between the absolute values of the differential strong one and strong zero output voltages in any of the six possible High-Speed states.	–	–	17	mV	–
$V_{OHHS(UT)}$	HS output high voltage, in Unterminated Mode	–	–	425	mV	1
$Z_{OS(UT)}$	Single ended output impedance, in Unterminated Mode	40	50	60	Ω	–
$\Delta Z_{OS(UT)}$	Single ended output impedance mismatch	–	–	10	Ω	–

Note:

1. *Value when driving into load impedance, Z_{ID}, of at least 10 kΩ.*

Transmitter AC specifications are not defined for HS Unterminated Mode due to difficulty of measuring these parameters with an unterminated transmission line.

9.1.1.2 Advanced Tx Equalization (TxEQ Option)

To mitigate the impact of channel-induced ISI, the system designer may choose the option to enable Advanced Tx Equalization in the HS-TX. This is called the TxEQ Option. The TxEQ Option described in this Section is optional normative. If the TxEQ Option is implemented in the transmitter then it shall be implemented per requirements in the C-PHY specification. The manufacturer shall specify in the component data sheet or other similar literature whether the TxEQ Option is implemented in the transmitter. The description of Advanced Tx Equalization for the 3-phase C-PHY signal is provided in this subsection.

The TxEQ Option in this specification replaces the method described in MIPI C-PHY Version 1.1. Transmitters that comply with MIPI C-PHY Version 1.1 are compatible with receivers that comply with this version of the specification and earlier versions.

Building on the concept of the three single-ended 3-phase output levels described in the Lane State Descriptions of ***Table 9***, the Advanced Tx Equalization uses three sublevels for each of the three single-ended 3-phase output signal levels as follows:

- HS High: H0, H1, H2 (in order from lowest to highest)

- HS Mid: M1-, M0, M1+ (in order from lowest to highest)
- HS Low: L2, L1, L0 (in order from lowest to highest)

The H2 and L2 voltage levels are identical to V_{OHHS} and V_{OLHS}, respectively. Similarly, M0 voltage level is identical to V_{CPTX}. ***Figure 57*** illustrates the relative levels of these HS High, HS Mid and HS Low sublevels.

Figure 57 also defines the Advanced Tx Equalization encoding rules. The levels shown last for a duration of 1.0 UI. ***Table 38*** defines the sublevels to be used for every possible signal transition on a Line. The Starting HS Level is the single-ended level prior to the signal transition, and the Ending HS Level is the level after the signal transition.

Table 38 Advanced Tx Equalization Strong and Weak Boost

Starting HS Level	Ending HS Level	Ending HS Level Sublevel
HS High: H0 or H1 or H2	HS High	H0
	HS Mid	M1-
	HS Low	L2
HS Mid: M1- or M0 or M1+	HS High	H1
	HS Mid	M0
	HS Low	L1
HS Low: L0 or L1 or L2	HS High	H2
	HS Mid	M+
	HS Low	L0

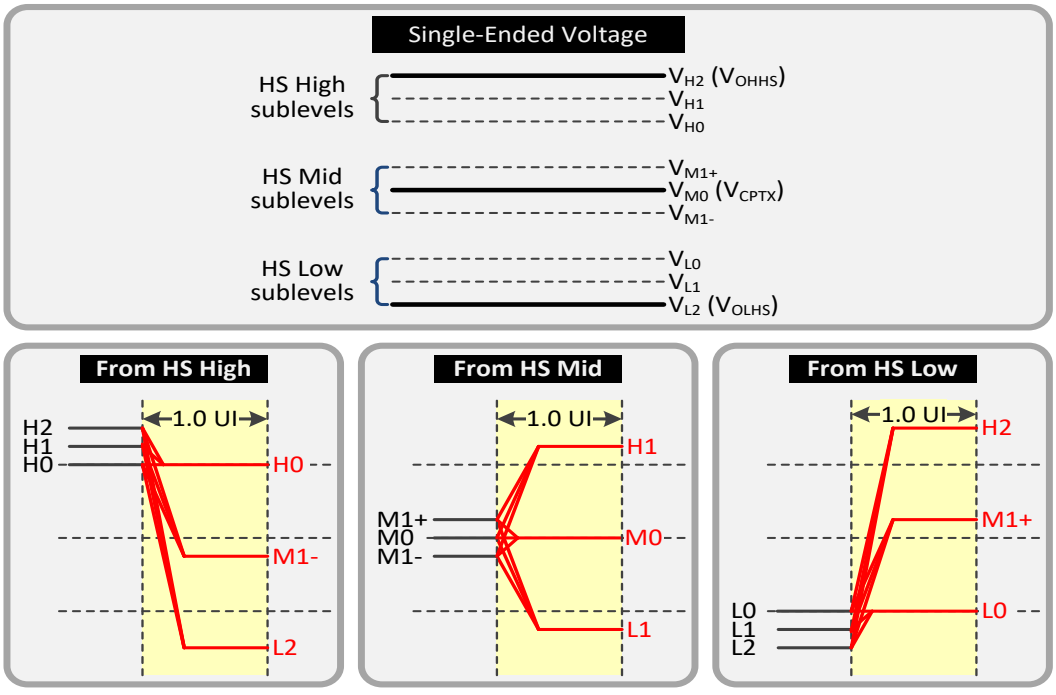


Figure 57 Example of High, Mid and Low Sublevels for Advanced Tx Equalization

Advanced Tx Equalization can be implemented in the transmitter by switching legs in the output driver as shown in ***Figure 58***. This method keeps the output impedance, Z_{OS}, always constant at 50 Ω.

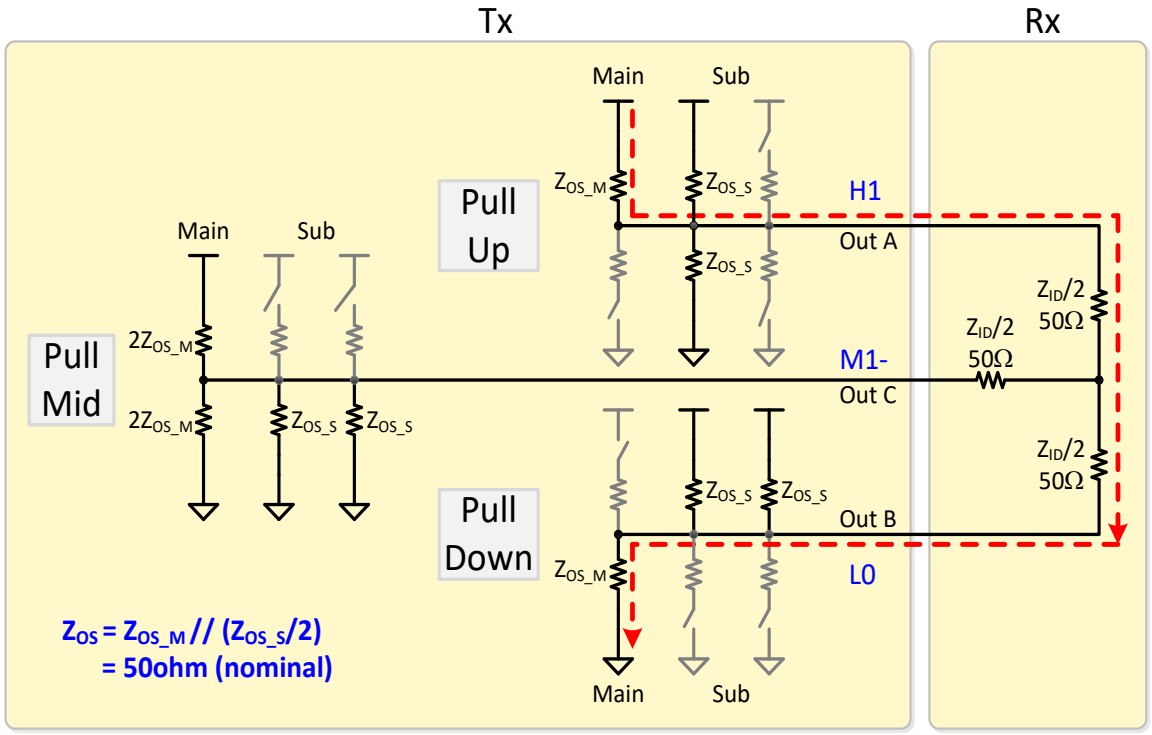


Figure 58 Example Switched Leg Implementation

Figure 59 shows an example transmit waveform with and without Advanced Tx Equalization. The top waveform shows the signal without Advanced Tx Equalization and the bottom waveform is the signal with Advanced Tx Equalization.

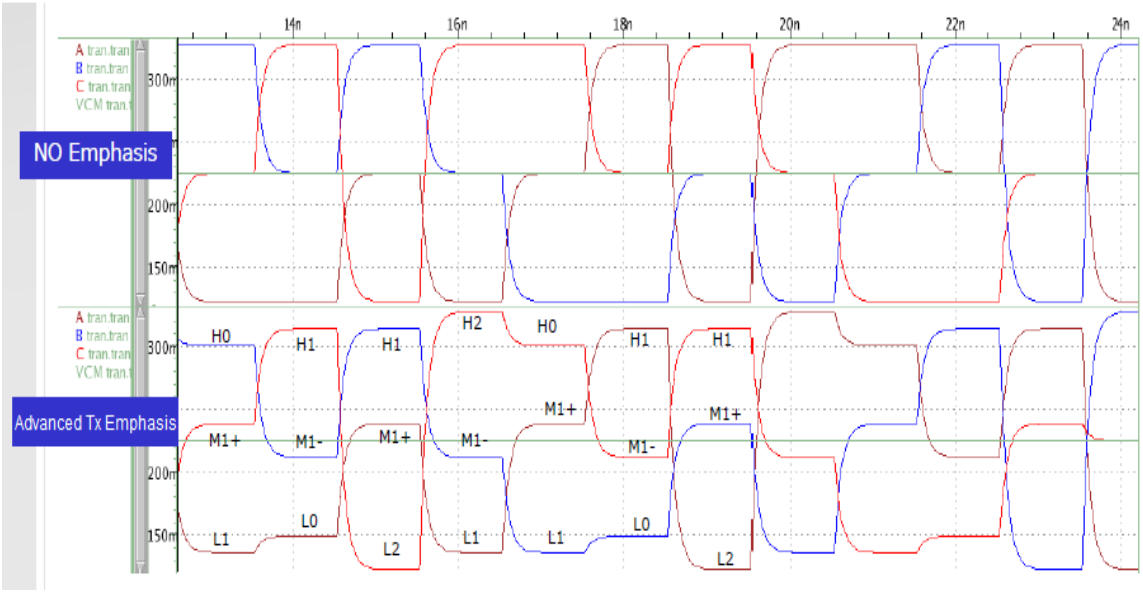


Figure 59 Example Waveforms With and Without Advanced Tx Equalization

The equalization levels are defined based on a change in the single-ended signal levels.

The magnitude of the change in the single-ended signals is defined as follows:

$$EQ_{M1+} = -20 \cdot \log\left(\frac{V_{H2} - V_{M1+}}{V_{H2} - V_{M0}}\right)$$

$$EQ_{M1-} = -20 \cdot \log\left(\frac{V_{M1-} - V_{L2}}{V_{M0} - V_{L2}}\right)$$

$$EQ_{H1} = -20 \cdot \log\left(\frac{V_{H1} - V_{M0}}{V_{H2} - V_{M0}}\right)$$

$$EQ_{H0} = -20 \cdot \log\left(\frac{V_{H0} - V_{M0}}{V_{H2} - V_{M0}}\right)$$

$$EQ_{L1} = -20 \cdot \log\left(\frac{V_{M0} - V_{L1}}{V_{M0} - V_{L2}}\right)$$

$$EQ_{L0} = -20 \cdot \log\left(\frac{V_{M0} - V_{L0}}{V_{M0} - V_{L2}}\right)$$

Table 39 Advanced Tx Equalization Sublevels

Parameter	Description	Min	Nom	Max	Units	Notes
EQ_{M1+} EQ_{M1-}	HS Mid Level Equalization	–	1.75	–	dB	1
EQ_{H1} EQ_{L1}	HS High H1 Level and the HS Low L1 Level Equalization	–	1.75	–	dB	1
EQ_{H0} EQ_{L0}	HS High H0 Level and the HS Low L0 Level Equalization	–	3.5	–	dB	1

Note:

1. *The nominal values are an example of equalization sublevels for 3.5Gsps transmission with the Standard Reference Channel. The implementer may adjust the equalization sublevels according to the symbol rate and channel characteristics being used.*

Examples above are provided as guidance to the implementer. For devices that use the TxEQ Option, the transmitter shall meet the Tx Timing Specifications specified in ***Section 10.3.1***, at the symbol rates of operation. The symbol rate capabilities are listed in ***Section 10.3.3***.

9.1.1.3 Tx ALP-Pause and ALP-Pause Wake

ALP Mode uses High-Speed transmission of codes as a replacement for Legacy LP signaling. The electrical specifications for ALP Mode are identical to the specifications for transmission of High-Speed data except that there are some additional requirements when the link is in the ALP-Pause Stop, ALP-Pause ULPS or ALP-Pause Wake states.

ALP-Pause Stop and ALP-Pause ULPS are defined by driving the A, B, and C signals to the same voltage level. For example, $V_{OD\_AB} = V_{OD\_BC} = V_{OD\_CA} = 0$. It is likely that a special receiver will be used to detect the difference in differential levels between the ALP-Pause state ($V_{OD\_AB} = 0$) and ALP-Pause Wake state ($V_{OD\_AB} = |V_{OD}|$ Strong).

Table 40 ALP-Pause Transmitter DC Specifications

Parameter	Description	Min	Nom	Max	Units	Notes
$V_{CPTX\_ALP\text{-}PAUSE}$	HS transmit static common-point voltage	0	–	310	mV	1
$\|V_{OD\_ALP\text{-}WAKE}\|$ strong	HS transmit differential voltage while in ALP Mode of the differential strong one and strong zero specified in ***Table 34***.	194	–	300	mV	1

Parameter	Description	Min	Nom	Max	Units	Notes
$\|V_{OD\_ALP-WAKE}\|$ strong (LVHS)	HS transmit differential voltage while in ALP Mode of the differential strong one and strong zero specified in ***Table 34***. Low-Voltage High-Speed Mode.	160	–	300	mV	1
$\|V_{OD\_ALP-PAUSE}\|$	ALP-Pause transmit differential voltage during the ALP-Pause Stop or ALP-Pause ULPS states	–	–	10	mV	1
V_{OHHS_ALP}	HS output high voltage during ALP-Pause Stop, ALP-Pause ULPS or ALP-Pause Wake	–	–	425	mV	1
$Z_{OS\_ALP-Wake}$	Single ended output impedance, Driven during ALP-Pause Wake	40	50	60	Ω	–
$Z_{OS\_ALP-PAUSE}$	Single ended output impedance, during ALP-Pause Stop or ALP-Pause ULPS	40	50	60	Ω	–

Note:

1. *Value when driving into load impedance, Z_{ID}, equal to 100 Ω.*

9.1.2 Low-Power Transmitter

The low-power transmitter shall be a slew-rate controlled push-pull driver. It is used for driving the Lines in all low-power operating modes. It is therefore important that the static power consumption of an LP transmitter is as low as possible. The slew-rate of signal transitions is bounded in order to keep EMI low. A Low-Power transmitter may additionally support the optional Low Voltage Low Power operation, in which the maximum output voltage is limited in comparison to the normal Low Power mode. An example of an LP transmitter is shown in ***Figure 60***.

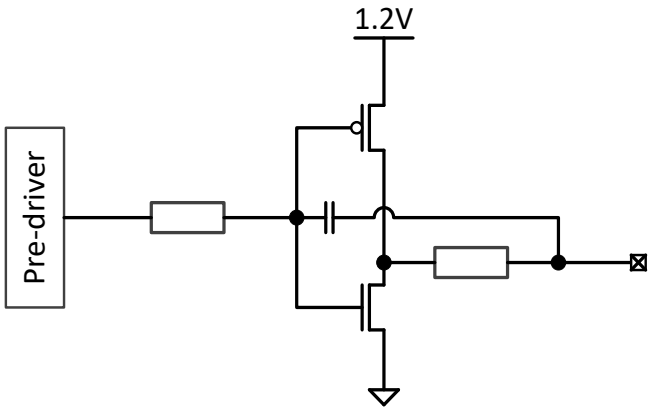


Figure 60 Example LP Transmitter

V_{OL} is the Thevenin output, low-level voltage in the LP transmit mode. This is the voltage at an unloaded pad pin in the low-level state. V_{OH} is the Thevenin output, high-level voltage in the high-level state, when the pad pin is not loaded. The LP transmitter shall not drive the pad pin potential statically beyond the maximum value of V_{OH}. The pull-up and pull-down output impedances of LP transmitters shall be as described in ***Figure 61*** and ***Figure 62***, respectively. The circuit for measuring V_{OL} and V_{OH} is shown in ***Figure 63***.

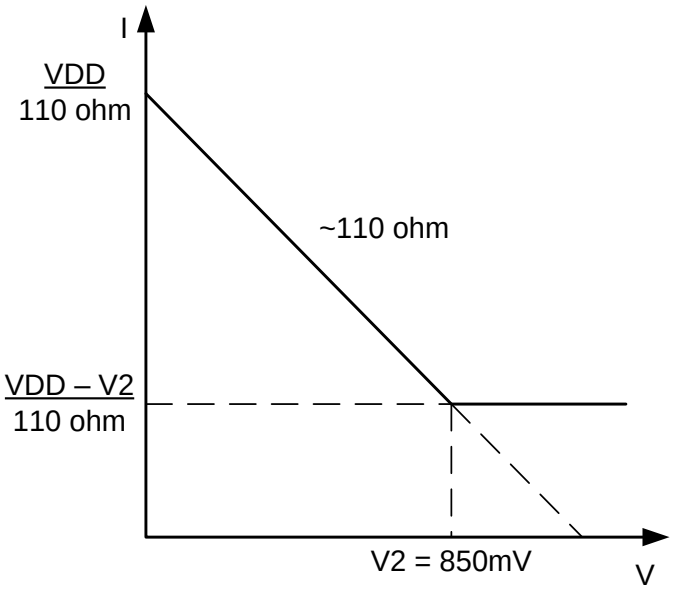


Figure 61 V-I Characteristic for LP Transmitter Driving Logic High

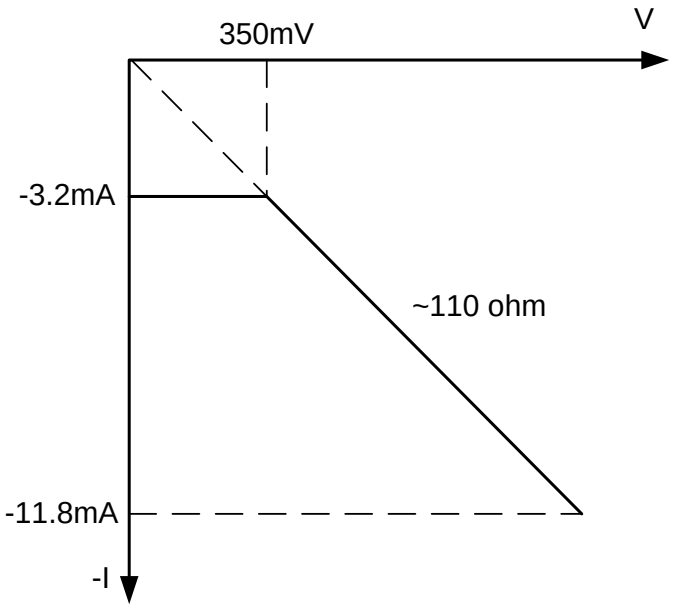


Figure 62 V-I Characteristic for LP Transmitter Driving Logic Low

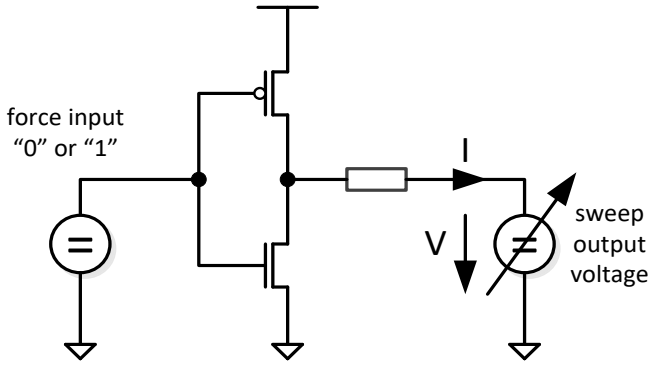


Figure 63 LP Transmitter V-I Characteristic Measurement Setup

The impedance Z_{OLP} is defined by:

$$Z_{OLP} = \left|\frac{V_{THEVENIN} - V_{PIN}}{I_{OUT}}\right|$$

The times t_{RLP} and t_{FLP} are the 15%-85% rise and fall times, respectively, of the output signal voltage, when the LP transmitter is driving a capacitive load C_{LOAD}. The 15%-85% levels are relative to the fully settled V_{OH} and V_{OL} voltages. The slew rate $\delta V/\delta t_{SR}$ is the derivative of the LP transmitter output signal voltage over time. The LP transmitter output signal transitions shall meet the maximum and minimum slew rate specifications as shown in ***Table 42***, ***Figure 64***, and ***Figure 65***. The intention of specifying a maximum slew rate value is to limit EMI.

1550 **Table 41 LP Transmitter DC Specifications**

Parameter	Description	Min	Nom	Max	Units	Notes
V_{OH}	Thevenin output high level	0.95	–	1.3	V	–
		0.95	–	1.1	V	3
V_{OL}	Thevenin output low level	-50	–	50	mV	–
Z_{OLP}	Output impedance of LP transmitter	110	–		Ω	1, 2

Note:

1. *See **Figure 61** and **Figure 62**.*
2. *Though no maximum value for Z_{OLP} is specified, the LP transmitter output impedance shall ensure the t_{RLP}/t_{FLP} specification is met.*
3. *Applicable when the Lane Module is in optional LVLP operation.*

Table 42 LP Transmitter AC Specifications

Parameter	Description		Min	Nom	Max	Units	Notes
t_{RLP}/t_{FLP}	15% - 85% rise time and fall time		–	–	25	ns	1
t_{REOT}	30% - 85% rise time and fall time		–	–	35	ns	5, 6
$t_{LP\text{-}PULSE\text{-}TX}$	Pulse width of the LP exclusive-OR clock	First LP exclusive-OR clock pulse after Stop state or last pulse before stop state	40	–	–	–	4
		All other pulses	20	–	–	–	4
$t_{LP\text{-}PER\text{-}TX}$	Period of the LP exclusive-OR clock		90	–	–	ns	–
$\delta V/\delta t_{SR}$	Slew rate @ C_{LOAD} = 0pF		–	–	500	mV/ns	1, 3, 7, 8
	Slew rate @ C_{LOAD} = 5pF		–	–	300	mV/ns	1, 3, 7, 8
	Slew rate @ C_{LOAD} = 20pF		–	–	250	mV/ns	1, 3, 7, 8
	Slew rate @ C_{LOAD} = 70pF		–	–	150	mV/ns	1, 3, 7, 8
	Slew rate @ C_{LOAD} = 0 to 70pF (Falling Edge Only)		25	–	–	mV/ns	1, 2, 3
	Slew rate @ C_{LOAD} = 0 to 70pF (Rising Edge Only)		25	–	–	mV/ns	1, 3, 9
	Slew rate @ C_{LOAD} = 0 to 70pF (Rising Edge Only)		$25 - 0.0625 \cdot (V_{O,INST} - 550)$	–	–	mV/ns	1, 3, 10, 11
C_{LOAD}	Load capacitance		0	–	70	pF	1

Note:

1. *C_{LOAD} includes the low-frequency equivalent transmission line capacitance. The capacitance of TX and RX are assumed to always be <10pF. The distributed line capacitance can be up to 50pF for a transmission line with 2ns delay.*
2. *When the output voltage is between 400 mV and 790 mV.*
3. *Measured as average across any 50 mV segment of the output signal transition.*
4. *This parameter value can be lower than t_{LPX} due to differences in rise vs. fall signal slopes and trip levels and mismatches between A, B and C LP transmitters. Any LP exclusive-OR pulse observed during HS EoT (transition from HS level to LP-111) is glitch behavior as described in **Section 9.2.2**.*
5. *The rise-time of t_{REOT} starts from the HS common-level at the moment the differential amplitude drops below 70mV, due to stopping the differential drive.*
6. *With an additional load capacitance C_{CP} between 0 and 90 pF on the termination center tap at RX side of the Lane*
7. *This value represents a corner point in a piecewise linear curve. See **Figure 64** and **Figure 65**.*
8. *When the output voltage is in the range specified by $V_{PIN(absmax)}$.*
9. *When the output voltage is between 400 mV and 550 mV.*
10. *Where $V_{O,INST}$ is the instantaneous output voltage, A, B or C, in millivolts.*
11. *When the output voltage is between 550 mV and 790 mV.*

There are minimum requirements on the duration of each LP state. To determine the duration of the LP state, the A, B and C signal lines are each compared to a common trip-level. The result of these comparisons of the A and C signals lines is then exclusive-ORed to produce a single pulse train. The output of this “exclusive-OR clock” can then be used to find the minimum pulse width output of an LP transmitter.

Using a common trip-level in the range [$V_{IL,MAX}$ + $V_{OL,MIN}$, $V_{IH,MIN}$ + $V_{OL,MAX}$], the exclusive-OR clock shall not contain pulses shorter than $t_{LP\text{-}PULSE\text{-}TX}$.

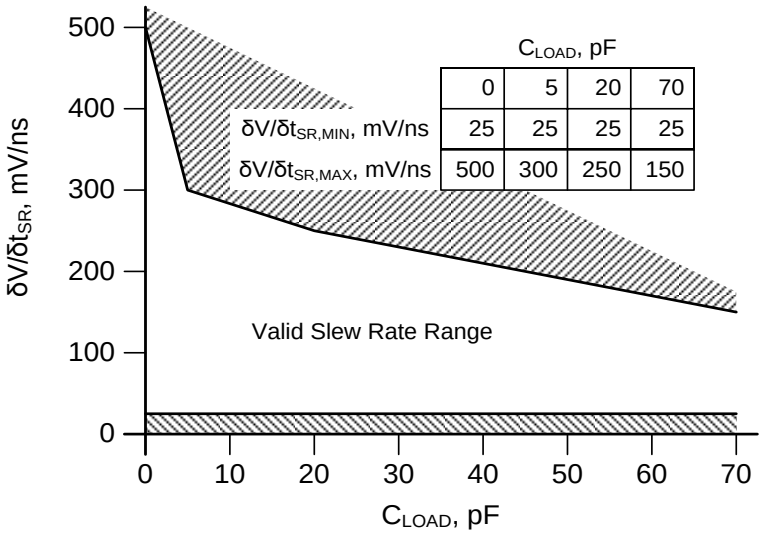


Figure 64 Slew Rate vs. C_{LOAD} (Falling Edge)

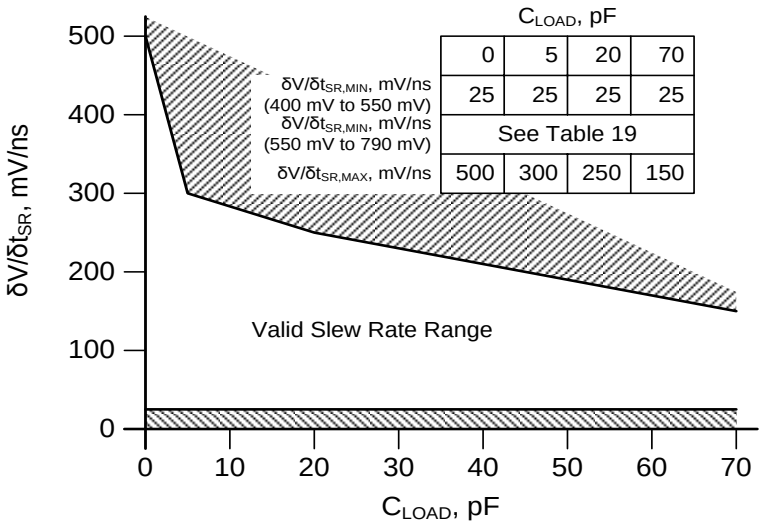


Figure 65 Slew Rate vs. C_{LOAD} (Rising Edge)

9.2 Receiver Characteristics

9.2.1 High-Speed Receiver

The HS receiver is a group of three differential line receivers. It contains three switchable parallel input terminations, $Z_{ID}/2$ between the three inputs: A, B and C. A simplified diagram of an example implementation is shown in ***Figure 66***.

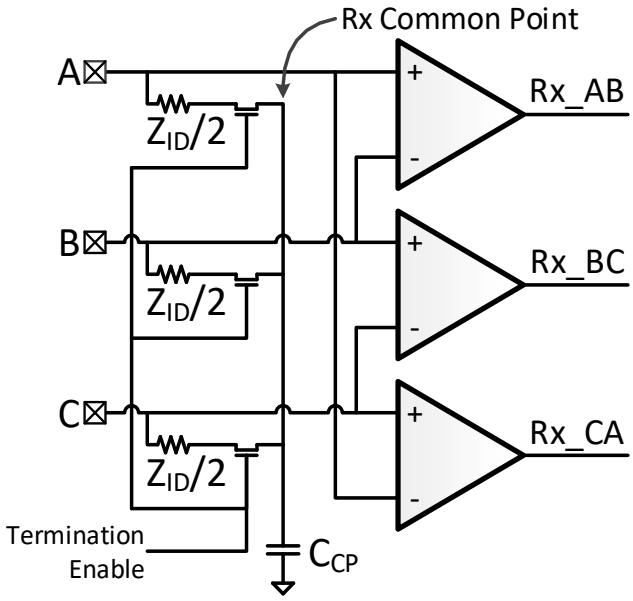


Figure 66 HS Receiver Implementation Example

The differential input high and low threshold voltages of the High-Speed receiver are denoted by V_{IDTH} and V_{IDTL}, respectively. V_{ILHS} and V_{IHHS} are the single-ended, input low and input high voltages, respectively. $V_{CPRX(DC)}$ is the differential input common-point voltage. The High-Speed receiver shall be able to detect differential signals at its A, B and C input signal pins when all three signal voltages, V_A, V_B and V_C, are within the common-point voltage range and if the voltage differences between V_A, V_B and V_C exceed either V_{IDTH} or V_{IDTL}. The High-Speed receiver shall receive High-Speed data correctly while rejecting common-point interference $\Delta V_{CPRX(HF)}$ and $\Delta V_{CPRX(LF)}$.

During operation of the High-Speed receiver, the three termination impedances $Z_{ID}/2$ are required between the A, B and C pins of the High-Speed receiver. The three $Z_{ID}/2$ terminations shall be disabled when the module is not in the High-Speed receive mode. When transitioning from low-power mode to High-Speed receive mode the termination impedances shall not be enabled until the single-ended input voltages on all of A, B and C fall below $V_{TERM-EN}$. To meet this requirement, a receiver does not need to sense the A, B and C lines to determine when to enable the line termination, rather the LP to HS transition timing can allow the line voltages to fall to the appropriate level before the line termination is enabled.

The differential input impedances of the receiver for A-B, B-C and C-A pairs are denoted by Z_{ID_AB}, Z_{ID_BC}, and Z_{ID_CA}, respectively. ΔZ_{ID} is the mismatch of the differential input impedances. This mismatch is defined as the ratio of the difference between the largest and smallest value of Z_{ID_AB}, Z_{ID_BC} and Z_{ID_CA}, and the average of those impedances:

$$\Delta Z_{ID} = 3 \cdot \frac{max(Z_{ID\_AB}, Z_{ID\_BC}, Z_{ID\_CA}) - min(Z_{ID\_AB}, Z_{ID\_BC}, Z_{ID\_CA})}{Z_{ID\_AB} + Z_{ID\_BC} + Z_{ID\_CA}}$$

The differential input impedances Z_{ID} and the differential input impedance mismatch ΔZ_{ID} shall be compliant with ***Table 43*** for all six possible High-Speed Wire States and for all allowed loading conditions. It is recommended that implementations keep the input impedance during state transitions as close as possible to the steady state value.

The RX common-mode return loss and the RX differential mode return loss are specified in ***Section 8***. C_{CP} is the common-mode AC termination, which ensures a proper termination of the receiver at higher

frequencies. For higher data rates, C_{CP} is needed at the termination center tap in order to meet the common-mode reflection requirements.

The differential input voltage signal $V_{DIF\_RX}(t)$ is defined as the voltage difference of the receiver inputs for the A-B, B-C and C-A pairs, defined as:

$$V_{DIF\_RX\_AB}(t) = V_A(t) - V_B(t);\ V_{DIF\_RX\_BC}(t) = V_B(t) - V_C(t);\ V_{DIF\_RX\_CA}(t) = V_C(t) - V_A(t);$$

$$V_{DIF\_RX,MAX} = V_{IHHS,MAX} - V_{ILHS,MIN}$$

Table 43 HS Receiver DC Specifications

Parameter	Description	Min	Nom	Max	Units	Notes
$V_{CPRX(DC)}$	Common-Point voltage HS receive mode	95	–	390	mV	1, 2
Z_{ID_AB} Z_{ID_BC} Z_{ID_CA}	Differential input impedance	80	100	120	Ω	–
ΔZ_{ID}	Differential input impedance mismatch	–	–	10	Ω	–

Note:

1. *Excluding possible additional RF interference of 100mV peak sine wave beyond 450MHz.*
2. *This table value includes a ground difference of 50mV between the transmitter and the receiver, the static common-point level tolerance and variations below 450MHz.*

Table 44 HS Receiver AC Specifications

Parameter	Description	Min	Nom	Max	Units	Notes
$\Delta V_{CPRX(HF)}$	Common-point interference beyond 450 MHz	–	–	50	mV	2
$\Delta V_{CPRX(LF)}$	Common-point interference 50MHz – 450MHz	-25	–	25	mV	1, 4
V_{IDTH}	Differential input high threshold	–	–	40	mV	–
V_{IDTL}	Differential input low threshold	-40	–	–	mV	–
V_{IHHS}	Single-ended input high voltage	–	–	535	mV	5
V_{ILHS}	Single-ended input low voltage	-40	–	–	mV	5
$V_{TERM-EN}$	Single-ended threshold for HS termination enable	–	–	450	mV	–
C_{CP}	Common-point termination	–	–	90	pF	3

Note:

1. *Excluding 'static' ground shift of 50mV.*
2. *$\Delta V_{CPRX(HF)}$ is the peak amplitude of a sine wave superimposed on the receiver inputs.*
3. *For higher bit rates, a 22pF capacitor is needed to meet the common-mode return loss specification.*
4. *Voltage difference compared to the DC average common-point potential.*
5. *Excluding possible additional RF interference of 100mV peak sine wave beyond 450MHz.*

9.2.1.1 Rx HS Unterminated Mode

To reduce the device power consumption, the system designer may choose the option operate the Link at a reduced speed in an unterminated mode. Refer to ***Section 9.1.1.1*** for a complete definition of the HS Unterminated Mode. The HS Unterminated Mode described in this Section is optional normative. If the HS Unterminated Mode is implemented in a receiver then it shall be implemented as defined in this Section. The

manufacturer shall specify in the component data sheet or other similar literature whether the Unterminated Mode Option is implemented in the receiver.

Refer to ***Section 9.1.1.1*** for additional guidance regarding the implementation of Unterminated Mode.

Any description of enabling or disabling the receiver termination, Z_{ID}, in ***Section 9.2.1*** is not applicable to the HS Unterminated Mode because the termination is always disabled.

The receiver circuit in ***Figure 66*** is also applicable for the parameters defined in ***Table 45***.

Table 45 HS Receiver DC Specifications for HS Unterminated Mode

Parameter	Description	Min	Nom	Max	Units	Notes
$V_{CPRX(DC)(UT)}$	Common-Point voltage HS receive mode	95	–	325	mV	1, 2
$Z_{ID\_AB(UT)}$ $Z_{ID\_BC(UT)}$ $Z_{ID\_CA(UT)}$	Differential input impedance	10	–	–	kΩ	–

Note:

1. *Excluding possible additional RF interference of 100mV peak sine wave beyond 450MHz.*
2. *This table value includes a ground difference of 50mV between the transmitter and the receiver, the static common-point level tolerance and variations below 450MHz.*

The High-Speed receiver AC specifications for the HS Unterminated Mode are specified in ***Table 44***. The Common-point termination capacitance, CCP, does not apply to the HS Unterminated Mode.

9.2.1.2 Rx ALP-Pause and ALP-Pause Wake

ALP-Pause and ALP-Pause Wake are detected differently in the receiver compared to the standard High-Speed receivers. This section contains electrical specifications for reception of the ALP-Pause and ALP-Pause Wake signaling. It is sufficient to detect the differential level solely between the A and B signals for detection of the ALP-Pause and ALP-Pause Wake states as described in ***Table 10***.

Table 46 HS Receiver DC Specifications for ALP-Pause and ALP-Pause Wake

Parameter	Description	Min	Nom	Max	Units	Notes
$V_{CPRX\_ALP(DC)}$	Common-Point voltage HS receive mode during ALP-Pause and ALP-Pause Wake	0	–	360	mV	1, 2
$Z_{ID\_AB\_ALP}$	Differential input impedance	80	100	120	Ω	–

Note:

1. *Excluding possible additional RF interference of 100mV peak sine wave beyond 450MHz.*
2. *This table value includes a ground difference of 50mV between the transmitter and the receiver, the static common-point level tolerance and variations below 450MHz.*

Table 47 HS Receiver AC Specifications for ALP-Pause and ALP-Pause Wake

Parameter	Description	Min	Nom	Max	Units	Notes
V_{IDTH_ALP}	Differential input high threshold for ALP-Pause and ALP-Pause Wake	–	–	120	mV	–
V_{IDTL_ALP}	Differential input low threshold for ALP-Pause and ALP-Pause Wake	40	–	–	mV	–
V_{IHHS_ALP}	Single-ended input high voltage for ALP-Pause and ALP-Pause Wake	–	–	535	mV	1
V_{ILHS_ALP}	Single-ended input low voltage for ALP-Pause and ALP-Pause Wake	-40	–	–	mV	1

Note:

1. *Excluding possible additional RF interference of 100mV peak sine wave beyond 450MHz.*

9.2.2 Low-Power Receiver

The low-power receiver is an un-terminated, single-ended receiver circuit. The low-power receiver is used to detect the low-power state on each pin. For high robustness, the LP receiver shall filter out noise pulses and RF interference. It is recommended the implementer optimize the LP receiver design for low power.

The input low-level voltage, V_{IL}, is the voltage at which the receiver is required to detect a low state in the input signal. A lower input voltage, $V_{IL\text{-}ULPS}$, may be used when the receiver is in the Ultra-Low Power State. V_{IL} is larger than the maximum single-ended Line voltage during HS transmission. Therefore, a LP receiver shall detect low during HS signaling.

The input high-level voltage, V_{IH}, is the voltage at which the receiver is required to detect a high state in the input signal. In order to reduce noise sensitivity on the received signal, an LP receiver shall incorporate hysteresis. The hysteresis voltage is defined as V_{HYST}.

The LP receiver shall reject any input signal smaller than e_{SPIKE}. Signal pulses wider than $T_{MIN\text{-}RX}$ shall propagate through the LP receiver.

Furthermore, the LP receivers shall be tolerant of super-positioned RF interference on top of the wanted Line signals. This implies an input signal filter. The LP receiver shall meet all specifications for interference with peak amplitude V_{INT} and frequency f_{INT}. The interference shall not cause glitches or incorrect operation during signal transitions.

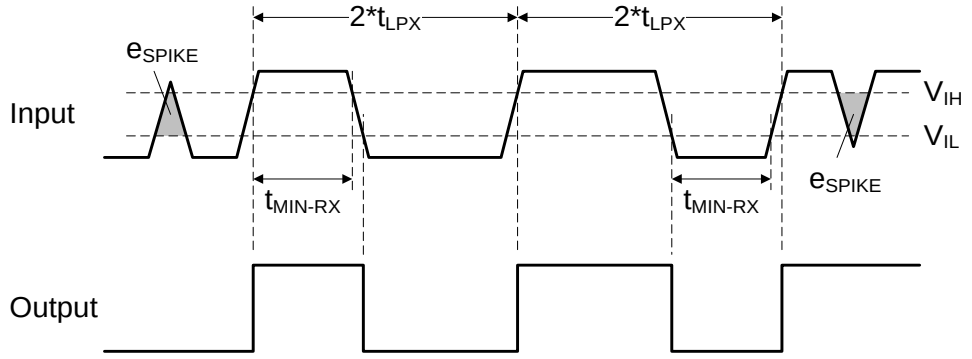


Figure 67 Input Glitch Rejection of Low-Power Receivers

Table 48 LP Receiver DC Specifications

Parameter	Description	Min	Nom	Max	Units	Notes
V_{IH}	Logic 1 input voltage	740	–	–	mV	–
V_{IL}	Logic 0 input voltage, not in ULP State	–	–	550	mV	–
$V_{IL\text{-}ULPS}$	Logic 0 input voltage, ULP State	–	–	300	mV	–
V_{HYST}	Input hysteresis	25	–	–	mV	–

Table 49 LP Receiver AC Specifications

Parameter	Description	Min	Nom	Max	Units	Notes
e_{SPIKE}	Input pulse rejection	–	–	300	V·ps	1, 2, 3
$T_{MIN\text{-}RX}$	Minimum pulse width response	20	–	–	ns	4
V_{INT}	Peak interference amplitude	–	–	200	mV	–
f_{INT}	Interference frequency	450	–	–	MHz	–

Note:

1. *Time-voltage integration of a spike above V_{IL} when being in LP-0 state or below V_{IH} when being in LP-1 state. e_{SPIKE} generation will ensure the spike is crossing both $V_{IL,MAX}$ and $V_{IH,MIN}$ levels.*
2. *An impulse less than this will not change the receiver state.*
3. *In addition to the required glitch rejection, implementers shall ensure rejection of known RF-interferers.*
4. *An input pulse greater than this shall toggle the output.*

9.3 Line Contention Detection

The low-power receiver and a separate contention detector (LP-CD) shall be used in a Bi-directional Lane to monitor the Line voltage on each low-power signal. This is required to detect Line contention as described in ***Section 7.1***. The low-power receiver shall be used to detect an LP high fault when the LP transmitter is driving high and the pin voltage is less than V_{IL}. Refer to ***Table 48***. The LP-CD shall be used to detect an LP low fault when the LP transmitter is driving low and the pin voltage is greater than V_{IHCD}. Refer to ***Table 50***. An LP low fault shall not be detected when the pin voltage is less than V_{ILCD}.

The general operation of a contention detector shall be similar to that of an LP receiver with lower threshold voltages. Although the DC specifications differ, the AC specifications of the LP-CD are defined to match those of the LP receiver and the LP-CD shall meet the specifications listed in ***Table 49*** except for T_{MIN-RX}. The LP-CD shall sufficiently filter the input signal to avoid false triggering on short events.

The LP-CD threshold voltages (V_{ILCD}, V_{IHCD}) are shown along with the normal signaling voltages in ***Figure 68***.

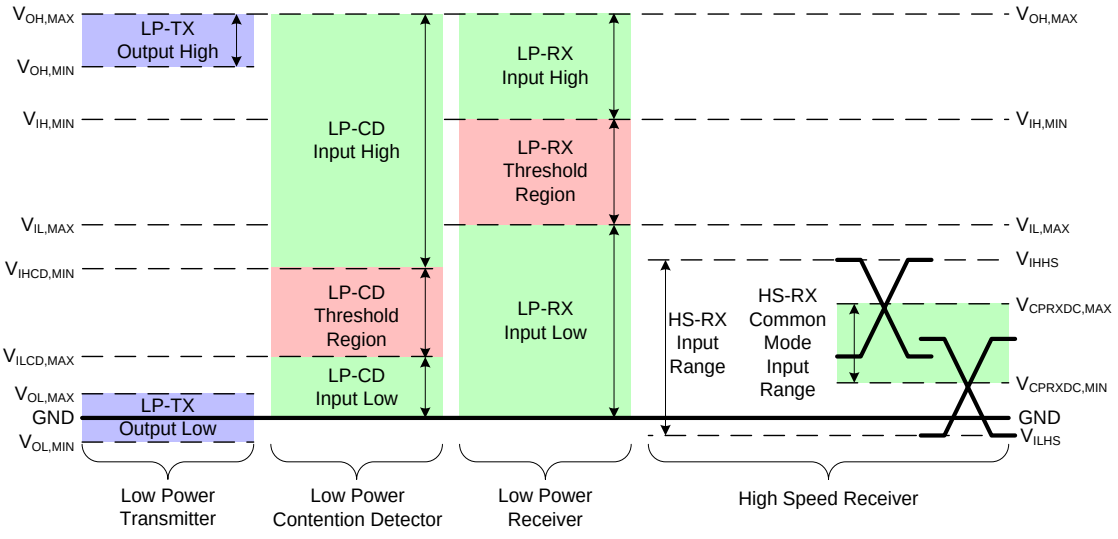


Figure 68 Signaling and Contention Voltage Levels

Table 50 Contention Detector (LP-CD) DC Specifications

Parameter	Description	Min	Nom	Max	Units	Notes
V_{IHCD}	Logic 1 contention threshold	450	–	–	mV	–
V_{ILCD}	Logic 0 contention threshold	–	–	200	mV	–

9.4 Input Characteristics

No structure within the PHY may be damaged, when a DC signal, which is within the signal voltage range V_{PIN}, is applied to a pad pin for an indefinite period of time. $V_{PIN(absmax)}$ is the maximum transient output voltage at the transmitter pin. The transmitter output voltage shall not exceed $V_{PIN,MAX}$ for a period greater

than $t_{VPIN(absmax)}$. When the PHY is in the low-power receive mode the pad pin leakage current shall be I_{LEAK}, when the pad signal voltage is within the signal voltage range of V_{PIN}. When a PHY is operated in the optional LVLP operating range, the pad pin leakage current shall be within the range defined by I_{LEAK} for pad signal voltages in the range of V_{PIN_LVLP}. The specification of I_{LEAK} assures interoperability of any PHY in the LP mode by restricting the maximum load current of an LP transmitter. An example test circuit for leakage current measurement is shown in ***Figure 69***.

The ground supply voltage shift between a Master and a Slave shall be less than V_{GNDSH}.

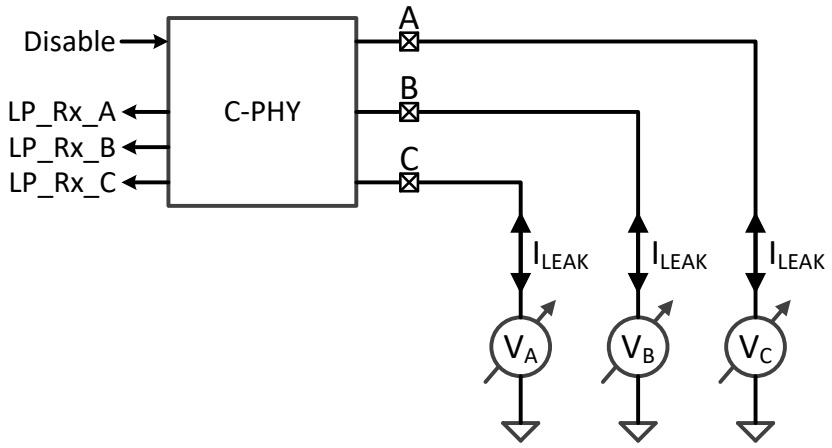


Figure 69 Pin Leakage Measurement Example Circuit

Table 51 Pin Characteristic Specifications

Parameter	Description	Min	Nom	Max	Units	Notes
V_{PIN}	Pin signal voltage range	-50	–	1350	mV	–
V_{PIN_LVLP}	Pin signal voltage range in LVLP operation	-50	–	1150	mV	–
I_{LEAK}	Pin leakage current	-100	–	100	µA	1
V_{GNDSH}	Ground shift	-50	–	50	mV	–
$V_{PIN(absmax)}$	Transient pin voltage level	-0.15	–	1.45	V	2, 3
$t_{VPIN(absmax)}$	Maximum transient time above $V_{PIN(max)}$ or below $V_{PIN(min)}$	–	–	20	ns	–

Note:

1. *When the Lane Module is in LP receive mode and the pad voltage is in the signal voltage range V_{PIN} or V_{PIN_LVLP} for LP mode or LVLP operation, respectively. I_{LEAK} should be well within the limits in LVLP operation.*
2. *The voltage overshoot and undershoot beyond the V_{PIN} range is only allowed for a duration of $t_{VPIN(absmax)}$ after any LP-0 to LP-1 transition or vice versa. For all other situations it must stay within the V_{PIN} range.*
3. *This value includes ground shift.*

10 High-Speed Signal Timing

This Section specifies the required timing of the High-Speed signaling interface independent of the electrical characteristics of the signal. C-PHY is based on 3-Phase symbol encoding technology where the symbol timing information is encoded in the data sent in each Lane. There is at least one transition in the received signal at each UI boundary.

Data transmission may occur at any rate greater than the minimum specified data bit rate.

Figure 70 shows an example PHY Configuration including the compliance measurement planes for the specified timing requirements. Note that the effect of signal degradation inside each package due to parasitic effects is included in the timing budget for the transmitter and receiver and is not included in the interconnect degradation budget. See ***Section 8*** for details.

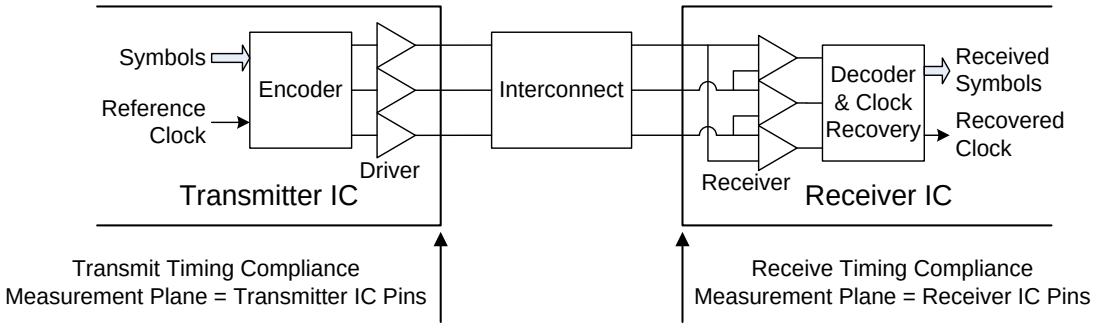


Figure 70 Conceptual C-PHY Lane Timing Compliance Measurement Planes

10.1 High-Speed UI Timing

The transmitter sends High-Speed data timing to the receiver by encoding the symbol clock timing in the transmitted symbol stream. Symbol encoding to Wire States ensures that a transition occurs in the High-Speed data at every symbol boundary. The receiver recovers the clock for data sampling using these guaranteed transitions in the symbol stream. An example of the single-ended V_A, V_B, and V_C voltages that change at every UI interval, as well as the differential received voltages V_A-V_B, V_B-V_C, and V_C-V_A, is shown in ***Figure 71***.

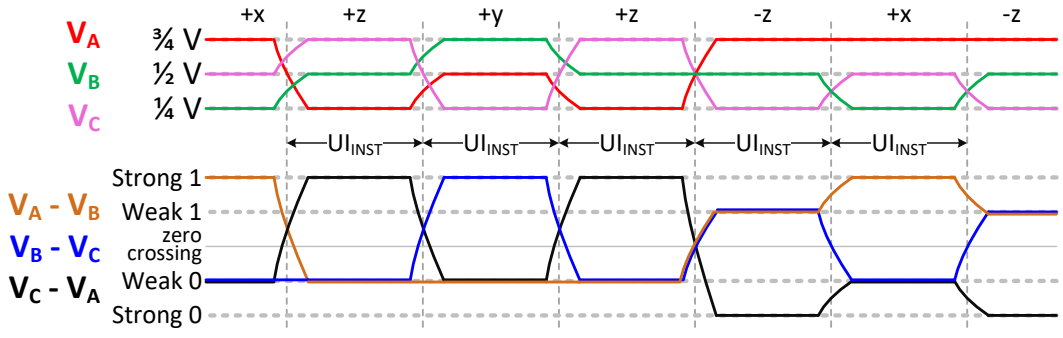


Figure 71 Example of Wire State Transitions at Symbol (UI) Boundaries

Receiver circuitry that recovers clock and samples data should respond immediately to transitions in the received data stream. Therefore, implementations may use frequency spreading modulation on the clock to reduce EMI.

Table 52 Unit Interval (UI) Specification

Parameter	Description	Min	Nom	Max	Units	Notes
UI_{INST}	UI instantaneous	–	–	12.5	ns	1, 2
ΔUI	UI variation	-10%	–	10%	UI	3
		-5%	–	5%	UI	4

Note:

1. *This value corresponds to a minimum 80 Msps data rate.*
2. *The minimum UI shall not be violated for any single bit period. The allowed instantaneous UI variation can cause instantaneous data rate variations. Therefore, receivers should be able to accommodate these instantaneous variations of the UI interval.*
3. *When UI ≥ 1ns, within a single burst.*
4. *When UI < 1ns, within a single burst.*

The allowed instantaneous UI variation can cause large, instantaneous data rate variations. Therefore, it is recommended that devices accommodate these instantaneous variations using some method, such as with elastic storage or by designing the data sink to be tolerant of UI variations.

10.2 High-Speed Data Eye Pattern and Transmission Timing

An eye pattern is a useful tool to specify the C-PHY timing characteristics. The C-PHY eye pattern described below is slightly different than a conventional eye pattern. Differences compared to conventional eye patterns are due to multiple levels seen at the receiving end, and due to specific behaviors of the clock recovery and data capture circuits that are likely to be implemented.

One or more of the differential receiver outputs will change at each UI boundary due to the symbol encoding rules. When multiple receiver outputs change they are often staggered in time due to slight differences in rise and fall times between the three signals of the Lane and due to slight differences in signal propagation times between the combinations of signal pairs received (e.g. A-B, B-C, and C-A). This concept is illustrated in detail in ***Figure 72***, which shows the five types of transitions that can appear in the eye pattern. ***Figure 72*** illustrates the concept that transitions of all three pair combinations can occur at slightly different times near each UI boundary due to the noted characteristics of the C-PHY drivers and receivers; and that there can be one, two or three zero-crossings at each UI boundary. Time $t_{\Delta J}$ in ***Figure 72***, highlights the time difference of the zero-crossings between the first and last signal pair transition.

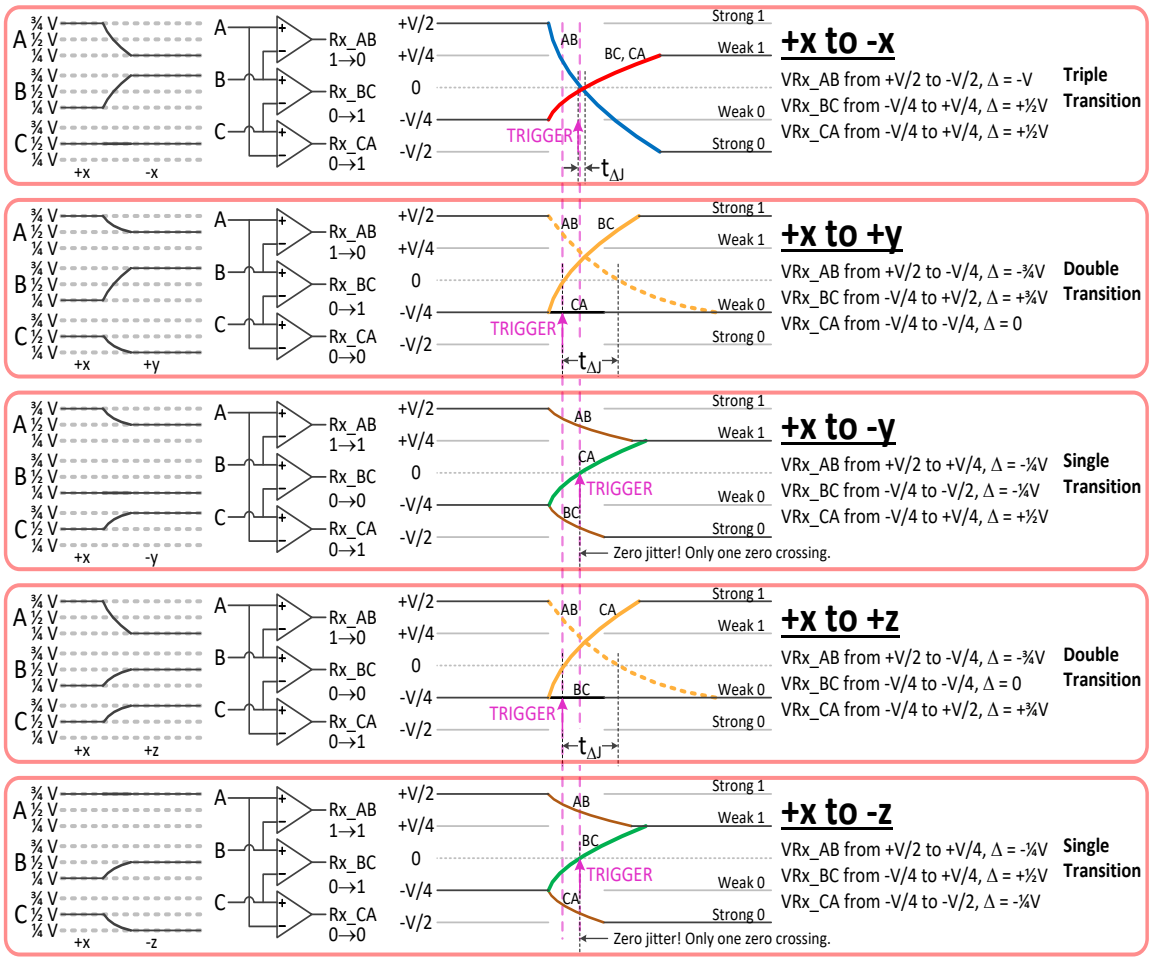


Figure 72 Illustration of all Possible Transitions from the +x State

The eye pattern shown in ***Figure 73*** has four received signal levels that are the result of three transmitted single-ended levels (¼ V, ½ V, ¾ V) of the driver circuit in the C-PHY transmitter. Combinations of the three single-ended levels from the drivers on the three signals of a Lane cause a strong and weak 1 and 0 to appear across the three differential receiver inputs (3 ways to receive 2 signals at a time out of a total of 3 signals). Only the center of the eye between the weak 0 and weak 1 are considered by the receivers. The eye pattern shall be drawn by overlapping the three waveforms of all three pairs of signals, which are: A minus B, B minus C, and C minus A. The eye pattern is drawn in this manner because all three pairs of signals are used simultaneously when the clock is recovered and data is captured at the C-PHY receiver.

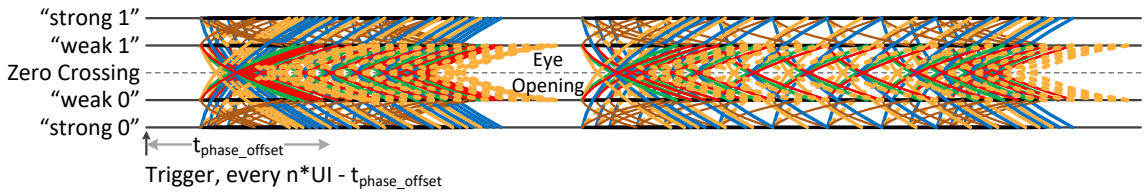


Figure 73 Eye Pattern Example, "Conventional" Trigger

As mentioned previously in ***Section 10.1***, the receiver recovers the clock for data sampling by using the guaranteed transitions at each UI boundary. Since the receiver can make use of this characteristic, it is important to know the events at the inputs of the differential receivers leading up to the transitions that occur

at the UI boundary. Events leading up to the first transition are obscured when the eye is viewed in the conventional manner as shown in ***Figure 73***. The C-PHY eye pattern in ***Figure 74*** is a triggered eye, meaning that the right side of the eye is aligned at a trigger point. The trigger is the first zero crossing of any of the three differential waveforms (A minus B, B minus C, and C minus A) that occur at each UI boundary. This trigger point is also shown in the individual waveforms shown in ***Figure 72*** for each of the of the transition types. For UI boundaries that have more than one transition of the differential waveforms, the subsequent transitions in the triggered eye are drawn at their proper position relative to the first transition. (For example: compare the relative position of the solid orange transition with the dashed orange transition in ***Figure 74***, and note how these two transitions are consistent with the same orange transitions in ***Figure 72***.) All of the first zero crossings at each UI boundary are aligned at the trigger point. Similarly, the transitions that occurred during the prior UI boundary are drawn at their proper position relative to the trigger point. The eye mask of the triggered eye diagram represents the worst case that will be observed at a C-PHY receiver that responds to the first zero crossing.

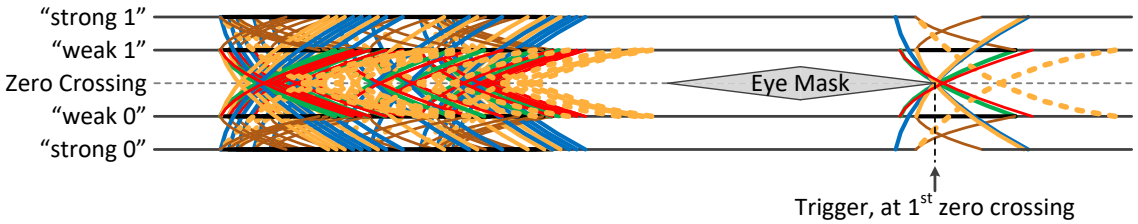


Figure 74 C-PHY Eye Pattern Example, Triggered Eye

The right-most point of the eye mask is aligned with the first zero crossing trigger point so it is consistent with sampling the received data just prior to the trigger point.

As mentioned above, the first zero crossing at each UI boundary (the trigger point) is associated with the sampling of the Wire State transmitted prior to that UI boundary. ***Figure 72*** shows that this first transition is caused by the following types of Wire State transitions: weak-to-weak, weak-to-strong, and possibly a strong-to-strong (in the triple transition case, +x to -x in ***Figure 72***). The difference of the first transition arrival time at one UI boundary relative to the first transition at the previous UI boundary affects the time period between sampling of two successive Wire States (receiver outputs). The peak-to-peak deviation of this zero-crossing time (the trigger point) is illustrated by the two pink dashed lines that span across all five waveforms in ***Figure 72***. Sampling clock jitter is also affected by cycle-to-cycle transmit clock jitter, receiver input offset voltage, and receiver duty-cycle distortion. The jitter caused by the relative difference in zero-crossing time due to the signal slew rate for each transition type is what is illustrated by the pink dashed lines in ***Figure 72***.

10.2.1 UI Jitter

The unit interval duration will vary slightly from one unit interval to the next due to the types of transitions that are produced by the symbol encoding, jitter amplification and ISI from the channel, and clock jitter at the transmitter. The UI Jitter and minimum UI concepts are described here because it is useful to know an upper bound of the amount of unit interval variation that can naturally occur. This is particularly helpful for test development so that signal quality and eye diagram tests can be designed so they do not overstress the receiver function by presenting an instantaneous unit interval time to the receiver that is shorter than what can normally occur. Knowing the magnitude of the UI jitter is also useful to the designer of the receiver function.

The variation of the unit interval, or UI Jitter, is the jitter of the instantaneous UI observed at the output of the channel, UI_{CHAN}. This is measured with the transmitter connected to one of the reference channels defined in ***Section 8***. UI_{CHAN} is defined as the difference in time between consecutive first transitions, or trigger points, at the UI boundary. The instantaneous UI after the channel for a specific UI cycle, n, is called $UI_{CHAN(n)}$.

$$UI_{CHAN(n)} = t_{TRIGGER\_POINT(n)} - t_{TRIGGER\_POINT(n-1)}$$

The average unit interval, $UI_{AVERAGE}$, is the reciprocal of the symbol rate:

$$UI_{AVERAGE} = \frac{1}{Symbol\_Rate}$$

The long-term average of UI_{CHAN} is also equal to $UI_{AVERAGE}$:

$$for\ a\ very\ large\ N\text{:}\ \frac{\sum_{n=1}^{N} UI_{CHAN(n)}}{N} = UI_{AVERAGE}$$

The cycle jitter during a particular UI, $UI\_Jitter_{CHAN(n)}$, is the difference between the average and instantaneous unit interval times:

$$UI\_Jitter_{CHAN(n)} = \left|UI_{AVERAGE} - UI_{CHAN(n)}\right|$$

Certain sequences of symbol transitions cause $UI_{CHAN(n)}$ to be shorter than $UI_{AVERAGE}$. Examples of these are shown in ***Figure 75***. The first example is a triple-transition followed by a double-transition and the second example is a single-transition followed by a double-transition. Transitions that occur in the opposite order of those shown in ***Figure 75*** can cause $UI_{CHAN(n)}$ to be longer than $UI_{AVERAGE}$.

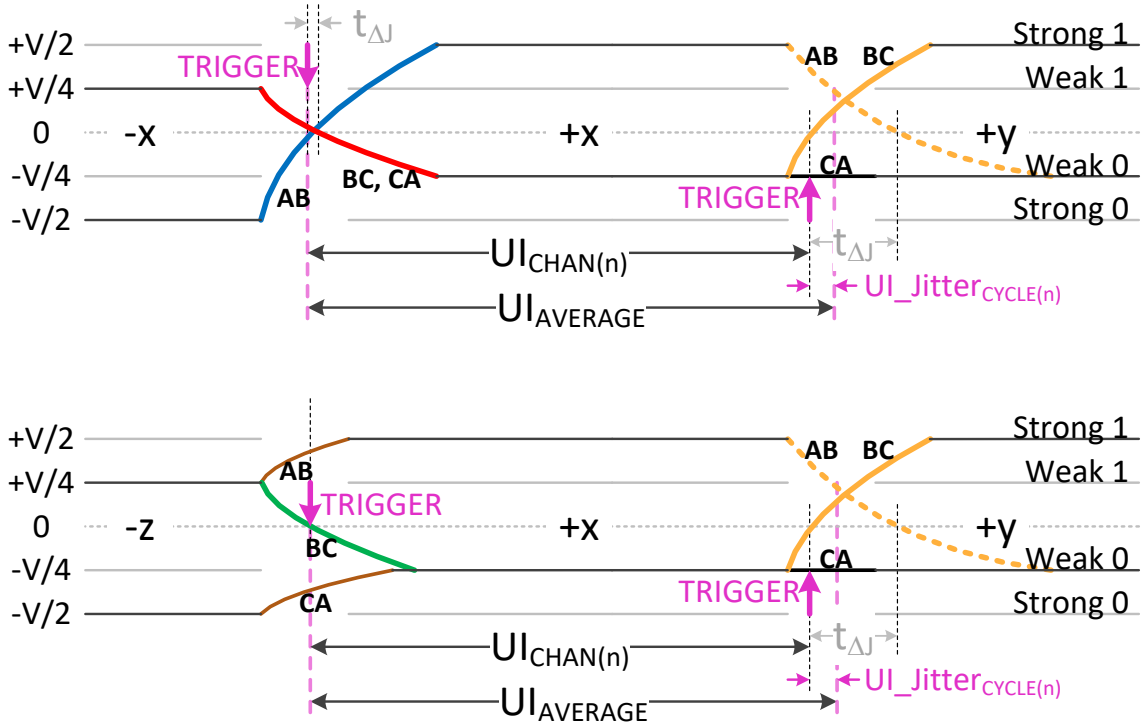


Figure 75 UI Jitter Examples, Short UI

Figure 76 shows different examples of UI jitter and its impact on the recovered clock (RCLK) in the receiver. Three cases are shown in the figure, one each for: $UI_{CHAN(n)} < UI_{AVERAGE}$, $UI_{CHAN(n)} = UI_{AVERAGE}$, and $UI_{CHAN(n)} > UI_{AVERAGE}$. The RCLK rising edge is caused by the first transition at each symbol boundary, which is the time indicated by the "Trigger" in ***Figure 76***. UI jitter causes RCLK jitter of the same amount.

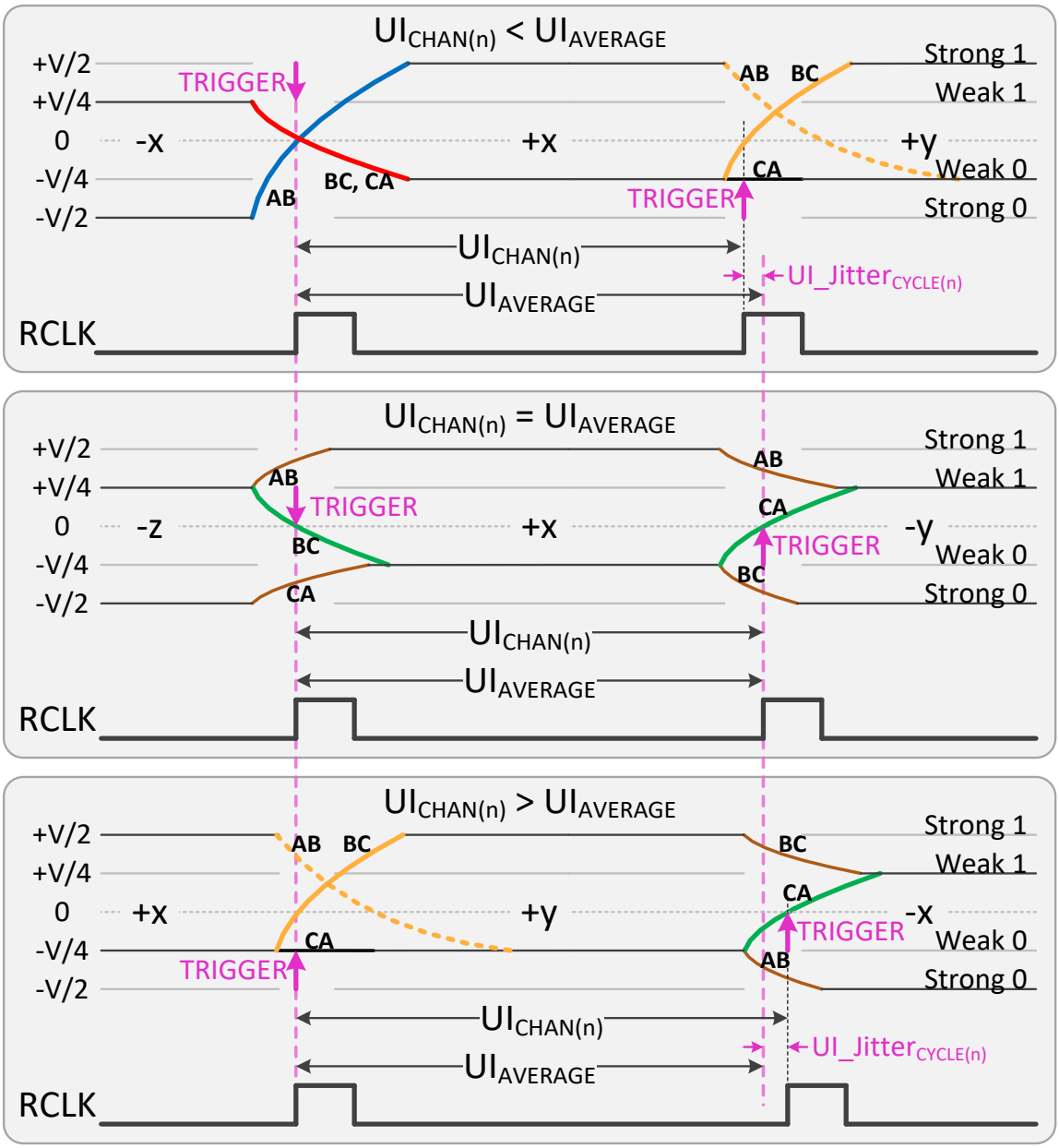


Figure 76 UI Jitter Examples, Causes of Different Instantaneous UI Durations

The peak UI jitter, UI_Jitter_{PEAK}, is expressed in units of the average unit interval time, and is defined as the maximum of $UI\_Jitter_{CHAN(n)}$ over many symbols, as follows, for a large N:

$$UI\_Jitter_{PEAK} = \frac{\max_{1 \le n \le N} \left| UI\_Jitter_{CHAN(n)} \right|}{UI_{AVERAGE}} = \frac{\max_{1 \le n \le N} \left| UI_{AVERAGE} - UI_{CHAN(n)} \right|}{UI_{AVERAGE}}$$

For the channel conditions and symbol rates described in this document, UI_Jitter$_{PEAK}$ for the transmitter and receiver is specified in ***Sections 10.3.1*** and ***10.3.3***, respectively. Note: the special case for symbol rates less than 1 Gsps is so UI_Jitter$_{PEAK}$ is compatible with the Transmitter Eye Diagram specification for symbol rates less than 1Gsps. This UI_Jitter$_{PEAK}$ behavior has been observed using a random distribution of symbol transitions, such as would be seen with a PRBS pattern. However, any distribution of complex wire state transitions is expected to have the same behavior. UI_Jitter$_{PEAK}$ is measured after the Transmitter output passes through the reference channels described in Section 8, Interconnect and Lane Configuration, with the termination at the output of the reference channel (Z_{ID_AB}, Z_{ID_BC} and Z_{ID_CA}) equal to 100 Ω.

Figure 77 shows an example of UI_Jitter$_{PEAK}$ with the triggered eye waveform.

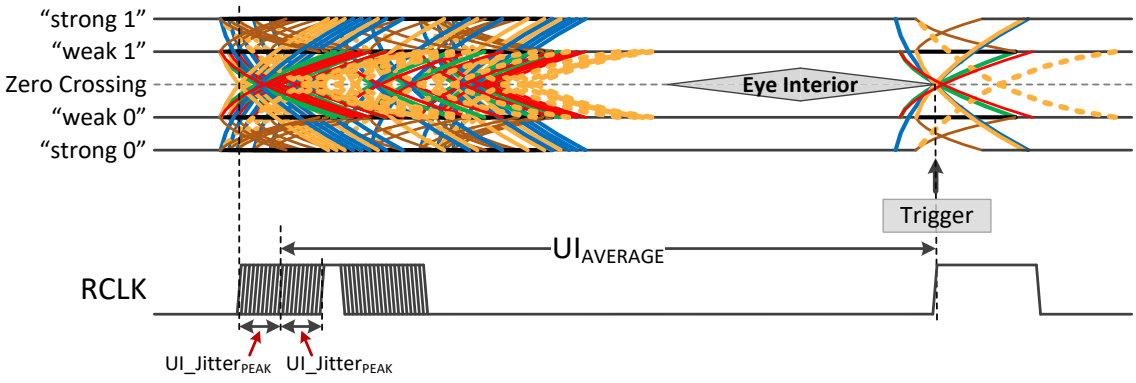


Figure 77 UI_Jitter$_{PEAK}$ Example, with Triggered Eye Waveform

10.2.2 Sources of Data Jitter and Recovered Clock Jitter

It is useful to the receiver designer to understand the sources of the data jitter described in ***Section 10.2*** and the UI jitter described in ***Section 10.2.1***. An example is provided so this can be visualized more easily. ***Figure*** 78 shows the data jitter and RCLK jitter effects on the triggered eye. The clock transition mask minimum and maximum are also shown.

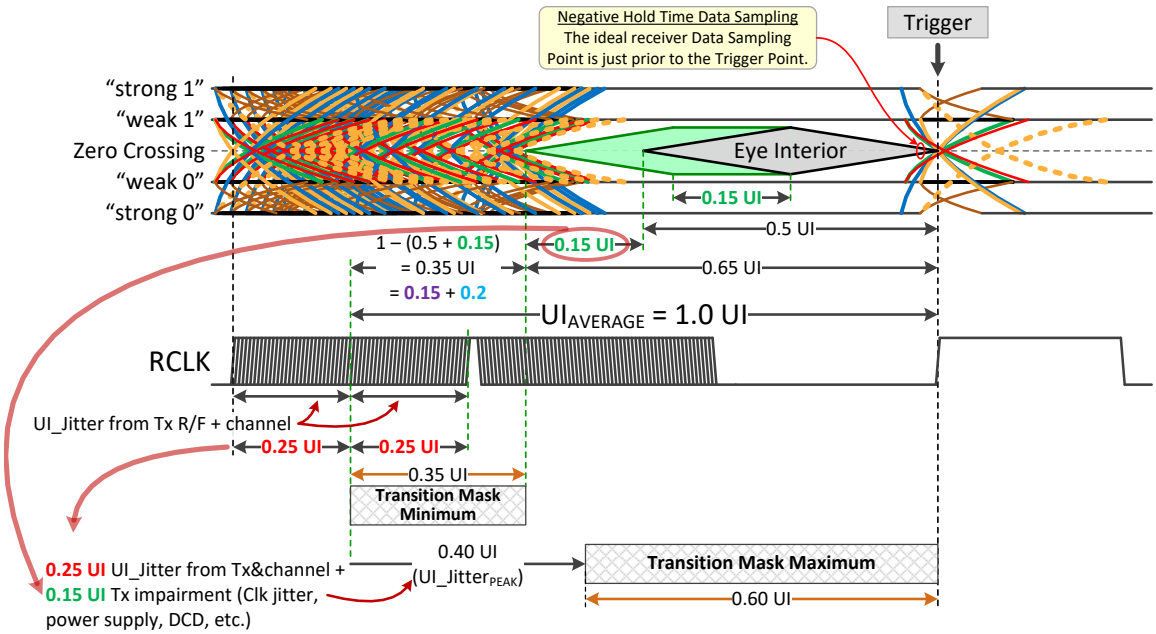


Figure 78 Data and RCLK Jitter Timing Diagram (Informative)

The primary purpose of receiver calibration is to more accurately configure the transition mask in the receiver clock recovery circuit. The minimum and maximum values of the transition mask are illustrated in ***Figure*** 78. It is better to set the transition mask as far toward the maximum as possible without exceeding the maximum. This is particularly true in test equipment where extreme conditions might be observed.

10.3 Timing Specifications

The timing requirements specified in this Section shall be met for the signal levels specified in ***Section 9***, with the channel specified in ***Section 8***, while transmitting a pseudo-random data pattern having data transition density similar to the PRBS data patterns described in ***Section 12***. The C-PHY Receiver Eye Diagram shown in ***Figure 81*** defines the receiver eye measurement parameters. The measurement points for the transmitter and for the receiver are specified in ***Figure 70***.

The timing specifications are based on allocations of the total unit interval as described in ***Table 53***.

Table 53 Timing Budget and C_{PAD} Assumptions (Informative)

Symbol Rate	Transmit	C_{PAD_TX}	TLIS	Receive	C_{PAD_RX}
> 2.5Gsps	0.3 UI	≤ 2pF	0.2 UI	0.5 UI	≤ 2pF
≥ 1.0Gsps & ≤ 2.5Gsps	0.3 UI	≤ 3pF	0.2 UI	0.5 UI	≤ 2pF

10.3.1 Tx Timing Specifications

The transmit signal level requirements are specified in ***Section 9.1.1***.

The inter-Lane skew between Lanes that are used together as a group of Lanes by a higher layer protocol shall be ±3.5 UI maximum at the output of the transmitter. Inter-Lane skew for the interconnect is described in ***Section 8.6.5***.

The Transmitter Eye Diagram requirements are specified below in ***Table 54***, which use ***Figure 79*** as a reference. The Eye Diagram for the transmitter is specified after the Transmitter output passes through the reference channels described in ***Section 8***, with the termination at the output of the reference channel (Z_{ID_AB},

Z_{ID_BC} and Z_{ID_CA}) equal to 100 Ω. The reference channel includes only insertion loss as defined in the TLIS specification in ***Section 8.6.1.1***. The Eye Interior part of the Transmitter Eye Diagram may float horizontally in the triggered eye to ensure compliance.

The Tx Eye Diagram specification described below shall be used to evaluate the Transmitter performance. ***Section 10.3.3*** shows the maximum operating rates for each feature and channel type combination.

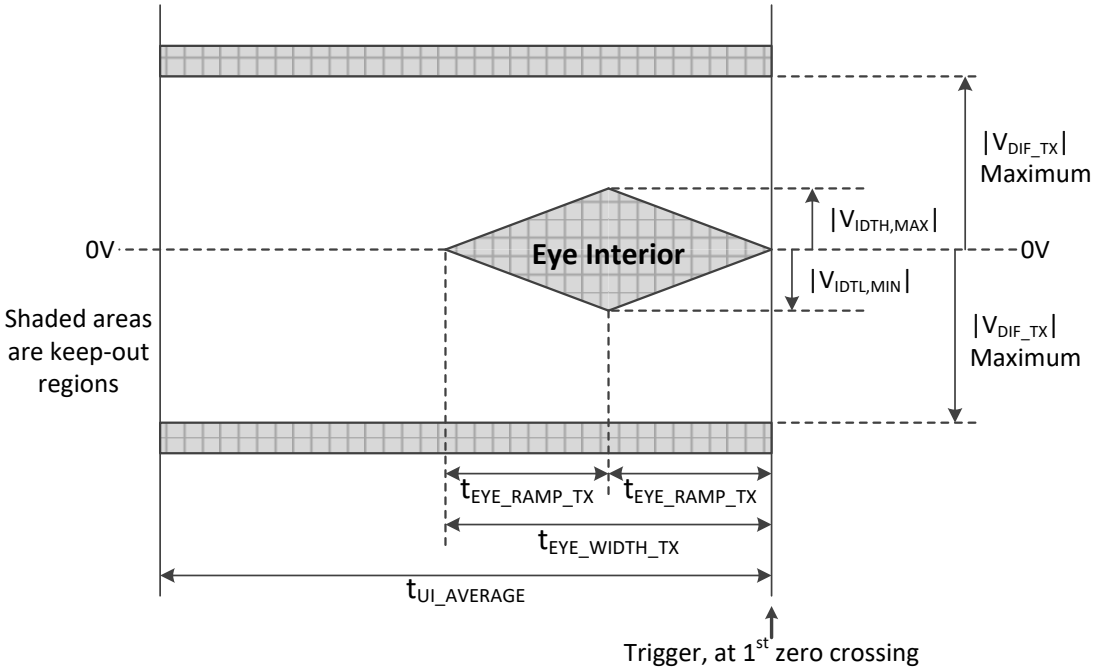


Figure 79 C-PHY Transmitter Eye Diagram

$$V_{DIF\_TX} = V_{IHHS,MAX} - V_{ILHS,MIN}$$

Table 54 Transmitter Timing Specifications

Parameter	Description	Min	Nom	Max	Units	Notes
$t_{EYE\_RAMP\_TX}$	Eye ramp time at the reference channel output	0.25	–	–	UI	–
$t_{EYE\_WIDTH\_TX}$	Eye width at the reference channel output	0.5	–	–	UI	–
$t_{UI\_AVERAGE}$	UI average	–	UI_{INST}	–	–	–
$UI_{CHAN(n)\_TX}$	Minimum Tx UI at the reference channel output	0.6	–	–	UI	–
$UI\_Jitter_{PEAK\_TX}$	Maximum Tx UI Jitter at the reference channel output	–	–	0.4	UI	–

The Low-Rate Transmitter Eye Diagram, ***Figure 80***, is a hexagonal-shaped eye diagram that shall be used at rates below 1Gsps. It has a variable width Eye Interior that is used to limit the range of jitter at the receiver when operating at low symbol rates. The t_{MID_TX} parameter increases at by the same amount as the $t_{UI\_AVERAGE}$ increases above the value at 1Gsps. This causes the eye opening to be a larger fraction of the total UI at lower symbol rates and does not allow the transition region to grow as a function of $t_{UI\_AVERAGE}$ at symbol rates below 1Gsps. The Eye Interior part of the Transmitter Hexagonal Eye Diagram may float horizontally in the triggered eye to ensure compliance.

It is anticipated that the transmitter will be tested using either the C-PHY Transmitter Eye Diagram in ***Figure*** 79 or the C-PHY Low-Rate Transmitter Hexagonal Eye Diagram in ***Figure 80*** based on the maximum operating speed of the device, but not both. It is anticipated that the use of Rx Calibration is unlikely for operation at rates below 1 Gsps.

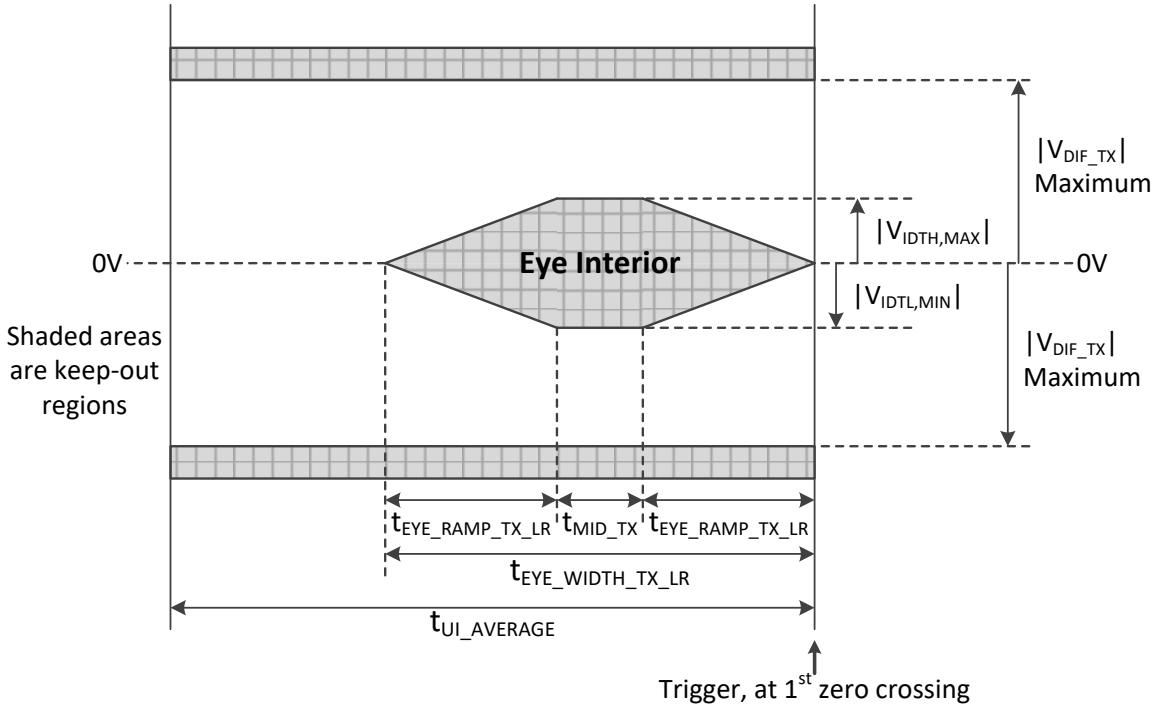


Figure 80 C-PHY Low-Rate Transmitter Hexagonal Eye Diagram

Table 55 Low Symbol Rate Transmitter Timing Specifications

Parameter	Description	Min	Nom	Max	Units	Notes
$t_{EYE\_RAMP\_TX\_LR}$	Transmit Eye ramp time at the reference channel output	–	–	250	ps	–
$t_{EYE\_WIDTH\_TX\_LR}$	Transmit Eye width at the reference channel output	–	–	Note 1	–	1
t_{MID_TX}	Width of the horizontal mid-section of the hexagonal Eye Interior	–	–	Note 2	–	2
$t_{UI\_AVERAGE}$	UI average	–	UI_{INST}	–	–	–
$UI_{CHAN(n)\_TX\_LR}$	Minimum Tx UI, Low Rate, at the reference channel output	Note 3	–	–	ps	3
$UI\_Jitter_{PEAK\_TX\_LR}$	Maximum Tx UI Jitter, Low Rate, at the reference channel output	–	–	400	ps	–

Note:

1. $t_{EYE\_WIDTH\_TX\_LR} = 2 \cdot t_{EYE\_RAMP\_TX\_LR} + t_{MID\_TX} = t_{MID\_TX} + 500ps$
2. $t_{MID\_TX} = t_{UI\_AVERAGE} – 1000ps$
3. $UI_{CHAN(n)\_TX\_LR(MIN)} = t_{MID\_TX} + 600\ ps$

10.3.2 Rx Timing Specifications

The receiver eye diagram requirements are specified below in ***Table 56***, which use ***Figure 81*** as a reference. The Eye Interior part of the Receiver Eye Diagram may float horizontally in the triggered eye to ensure compliance.

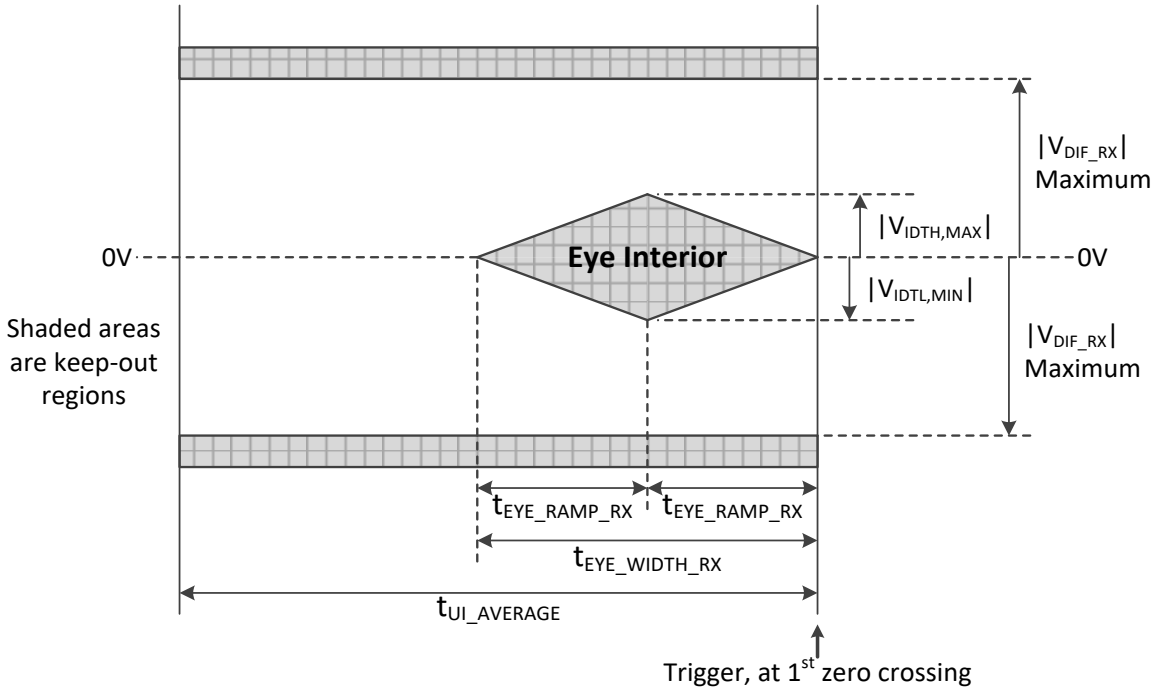


Figure 81 C-PHY Receiver Eye Diagram

$$V_{DIF\_RX} = V_{IHHS,MAX} - V_{ILHS,MIN}$$

Table 56 Receiver Timing Specifications

Parameter	Description	Min	Nom	Max	Units	Notes
$t_{EYE\_RAMP\_RX}$	Eye ramp time at the receiver	0.25	–	–	UI	–
$t_{EYE\_WIDTH\_RX}$	Eye width at the receiver	0.5	–	–	UI	–
$t_{UI\_AVERAGE}$	UI average	–	UI_{INST}	–	–	–
$UI_{CHAN(n)\_RX}$	Minimum Rx UI at the receiver	0.6	–	–	UI	–
$UI\_Jitter_{PEAK\_RX}$	Maximum Rx UI Jitter at the receiver	–	–	0.4	UI	–

The Low-Rate Receiver Eye Diagram, ***Figure 82***, is a hexagonal-shaped eye diagram that shall be used at rates below 1Gsps. It has a variable width Eye Interior that is used to limit the range of jitter at the receiver when operating at low symbol rates. The t_{MID_RX} parameter increases at by the same amount as the $t_{UI\_AVERAGE}$ increases above the value at 1Gsps. This causes the eye opening to be a larger fraction of the total UI at lower symbol rates and does not allow the transition region to grow as a function of $t_{UI\_AVERAGE}$ at symbol rates below 1Gsps. The Eye Interior part of the Receiver Hexagonal Eye Diagram may float horizontally in the triggered eye to ensure compliance.

It is anticipated that the receiver will be tested using either the C-PHY Receiver Eye Diagram in ***Figure 81*** or the C-PHY Low-Rate Receiver Hexagonal Eye Diagram in ***Figure 82*** based on the maximum operating speed of the device, but not both. It is anticipated that the use of receive calibration is unlikely for operation at symbol rates below 1 Gsps.

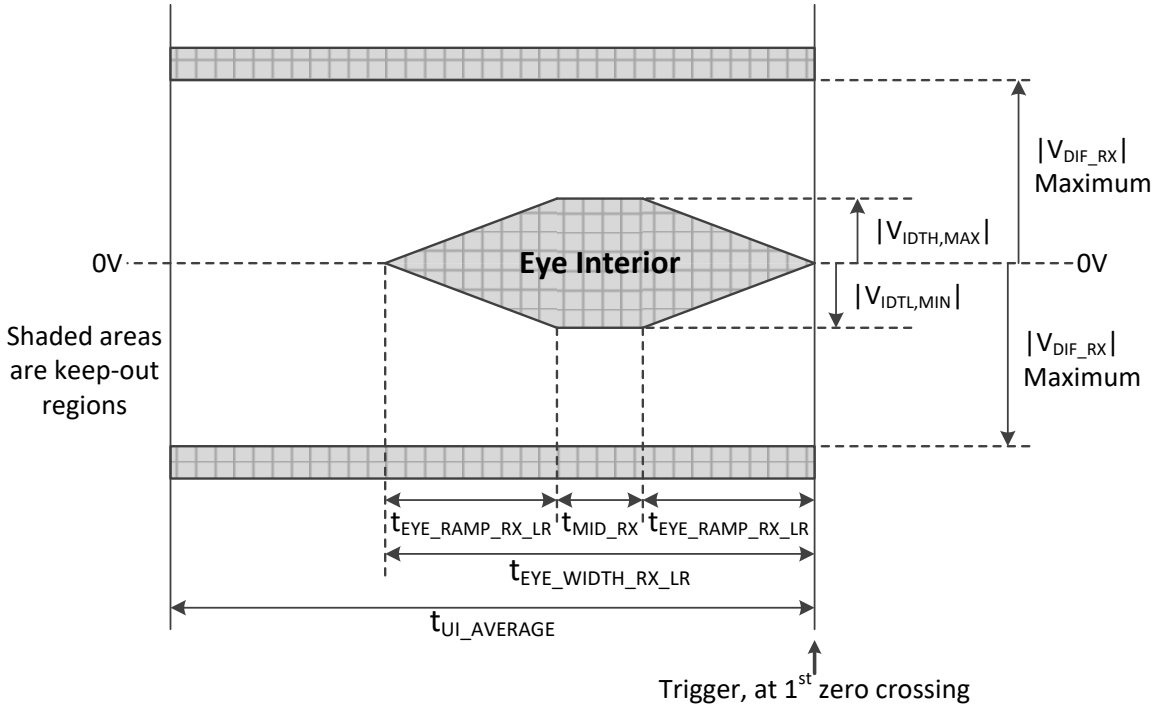


Figure 82 C-PHY Low-Rate Receiver Hexagonal Eye Diagram

Table 57 Low Symbol Rate Receiver Timing Specifications

Parameter	Description	Min	Nom	Max	Units	Notes
$t_{EYE\_RAMP\_RX\_LR}$	Eye ramp time at the receiver	–	–	250	ps	–
$t_{EYE\_WIDTH\_RX\_LR}$	Eye width at the receiver	–	–	Note 1	–	1
t_{MID_RX}	Width of the horizontal mid-section of the hexagonal Eye Interior	–	–	Note 2	–	2
$t_{UI\_AVERAGE}$	UI average	–	UI_{INST}	–	–	–
$UI_{CHAN(n)\_RX\_LR}$	Minimum Rx UI, Low Rate, at the receiver	Note 3	–	–	ps	3
$UI\_Jitter_{PEAK\_RX\_LR}$	Maximum Rx UI Jitter, Low Rate, at the receiver	–	–	400	ps	–

Note:

1. $t_{EYE\_WIDTH\_RX\_LR} = 2 \cdot t_{EYE\_RAMP\_RX\_LR} + t_{MID\_RX} = t_{MID\_RX} + 500ps$
2. $t_{MID\_RX} = t_{UI\_AVERAGE} – 1000ps$
3. $UI_{CHAN(n)\_RX\_LR(MIN)} = t_{MID\_RX} + 600\ ps$

10.3.3 Channel Rate Guidance as a Function of Interconnect and Feature Set

Table 58, ***Table 59***, and ***Table 60*** summarize the anticipated symbol rate capability of the system based on the features defined in the C-PHY Specification and the type of channel defined in ***Section 8***. Advanced Tx Equalization, referred to as the TxEQ Option in the tables below, is described in ***Section 9.1.1.2***.

A manufacturer's device data sheet may specify maximum operating speeds that are less than the values stated in ***Table 58***, ***Table 59***, and ***Table 60***. For example: a device may support the TxEQ Option but have an inherent internal maximum speed limitation. Use of the TxEQ Option would allow the device to operate over high-loss channels, but the maximum speed over the lower loss channels (e.g. Short Reference Channel) may be less than the values stated in ***Table 58***.

Table 58 C-PHY System Capability, Normal High-Speed Mode

Channel	TxEQ Option disabled	TxEQ Option enabled
Short Reference Channel (***Section 8***) [1]	2.5 Gsps	4.5 Gsps
Standard Reference Channel (***Section 8***)	1.7 Gsps	3.5 Gsps
Long Reference Channel (***Section 8***)	1.3 Gsps	2.3 Gsps

Note:

1. *The achievable symbol rate using the C-PHY v1.0 Legacy Channel is the same as the symbol rate using the Short Reference Channel.*

Guidance for the HS Unterminated Mode is provided in ***Table 59***. It is allowed to operate the HS Unterminated Mode with the TxEQ Option enabled but no specific guidance is provided.

Table 59 C-PHY System Capability, HS Unterminated Mode

Channel	TxEQ Option disabled	TxEQ Option enabled
Short Reference Channel (***Section 8***)	0.9 Gsps	Note 1
Standard Reference Channel (***Section 8***)	0.9 Gsps	Note 1
Long Reference Channel (***Section 8***)	0.8 Gsps	Note 1

Note:

1. *Operation in this condition is allowed, but no specific guidance is provided.*

Guidance for the Low-Voltage High-Speed Mode (LVHS Mode) is provided in ***Table 60***.

Table 60 C-PHY System Capability, Low-Voltage High-Speed Mode (LVHS Mode)

Channel	TxEQ Option disabled	TxEQ Option enabled
Short Reference Channel (***Section 8***)	2.3 Gsps	Note 1
Standard Reference Channel (***Section 8***)	1.4 Gsps	Note 1
Long Reference Channel (***Section 8***)	0.8 Gsps	Note 1

Note:

1. *Operation in this condition is allowed, but no specific guidance is provided.*

10.4 Reverse High-Speed Data Transmission Timing

High-Speed reverse data transmission is not supported.

11 Regulatory Requirements

All C-PHY based devices should be designed to meet the applicable regulatory requirements.

12 Built-In Test Circuitry (Informative)

12.1 Introduction

Standardized built-in test circuitry in the C-PHY Lane function simplifies production testing, verification, interoperability testing and even self-test of the mobile device that uses the C-PHY. Compatibility of the built-in test circuitry benefits both test equipment and device makers. The test circuit specification defines precise characteristics and behavior of the built-in test circuits, and also includes a register definition for control of the PHY circuit operating and test modes and for observability of important PHY circuit operating conditions.

It is recommended to include the built-in test circuitry and associated control and status registers per the method described in ***Section 12***.

12.2 Register Concept

The Lane Configuration and Status Registers can be accessible through any register or memory space that is associated with or related to the C-PHY function. There is no need to use any specific physical interface to gain access to the register or memory space. The only general characteristics to ensure compatibility between test equipment and devices being tested are that each register location be at least 8 bits wide, that the register space be both readable and writeable, and that there are a sufficient number of available address locations to accommodate the product of the per-Lane register count times the number of Lanes being used plus the number of global registers that are necessary.

12.2.1 Allocation of Register Addresses

Following are specific characteristics of registers that are associated with each C-PHY Lane:

1. There is a group of Lane Configuration and Status Registers associated with each C-PHY Lane Tx and Rx function.
2. The group of registers for the Tx Lane function is separate from a group of registers for the Rx Lane function. Each C-PHY Lane transmitter function is associated with one group of Tx Lane Configuration and Status Registers, and each C-PHY Lane receiver function is associated with one group of Rx Lane Configuration and Status Registers.
3. The individual registers within a group of Lane Configuration and Status Registers that correspond to a specific Lane are defined to exist in a contiguous block of addresses starting at a Tx Lane Base Address (Tx_Lane_n_Base) or a Rx Lane Base Address (Rx_Lane_n_Base). The offset of each Lane Configuration and Status Register relative to the Tx Lane Base Address or Rx Lane Base Address is defined for each register definition.
4. The exact physical addresses of each Tx Lane Base Address and Rx Lane Base Address are flexible. They are defined by the device manufacturer.
5. There can be gaps in the address space between the highest register address within any group of Lane Configuration and Status Registers and Lane Base Address of any other group of Lane Configuration and Status Registers depending on the assignment of physical addresses to base addresses. Device manufacturers may choose to have gaps between Lane Configuration and Status Register groups to be able to fit groups of registers within available addresses in the register address space. (There may be gaps in the address space between each group of Lane Configuration and Status Registers.)
6. Having a unique Tx Lane Base Address or Rx Lane Base Address for each C-PHY Lane in a device allows the attributes and status of each Lane to be controlled and read individually.

Following are specific characteristics of registers that apply globally across all C-PHY Lanes, or to a defined group of C-PHY Lanes, in a device:

1. There is a set of Tx Global Configuration and Status Registers that applies globally to all C-PHY Tx Lanes in a device (Lanes that have C-PHY transmit capability) or to a defined group of Tx C-PHY Lanes in a device.
2. There is a set of Rx Global Configuration and Status Registers that applies globally to all Rx Lanes in a device (Lanes that have C-PHY receive capability) or to a defined group of Rx C-PHY Lanes in a device.
3. The individual registers within the block of Global Configuration and Status Registers are defined to exist in a contiguous block of addresses beginning from the Tx Global Registers Base Address (Tx_Global_Registers_Base) or Rx Global Registers Base Address (Rx_Global_Registers_Base). The offset of each Global Configuration and Status Register relative to the Tx Global Registers Base Address or Rx Global Registers Base Address is defined for each register definition.
4. The exact physical addresses of the Tx Global Registers Base Address and Rx Global Registers Base Address are flexible. They are defined by the device manufacturer.
5. There can be gaps in the address space between the highest register address within the Tx Global Configuration and Status Registers or Rx Global Configuration and Status Registers and any Tx Lane Base Address or Rx Lane Base Address depending on the assignment of physical addresses to base addresses. Device manufacturers may choose to have gaps between register groups to be able to fit groups of registers within available addresses in the register address space.

A pictorial example of this method is illustrated in ***Figure 83***. This method of register definition enables compatibility between test equipment or test fixtures and devices under test, where only the device-specific base addresses need to be programmed into the testers. Compatibility is ensured by consistent use of the register addressing definition and functional behavior of every bit in the register space defined in this Section.

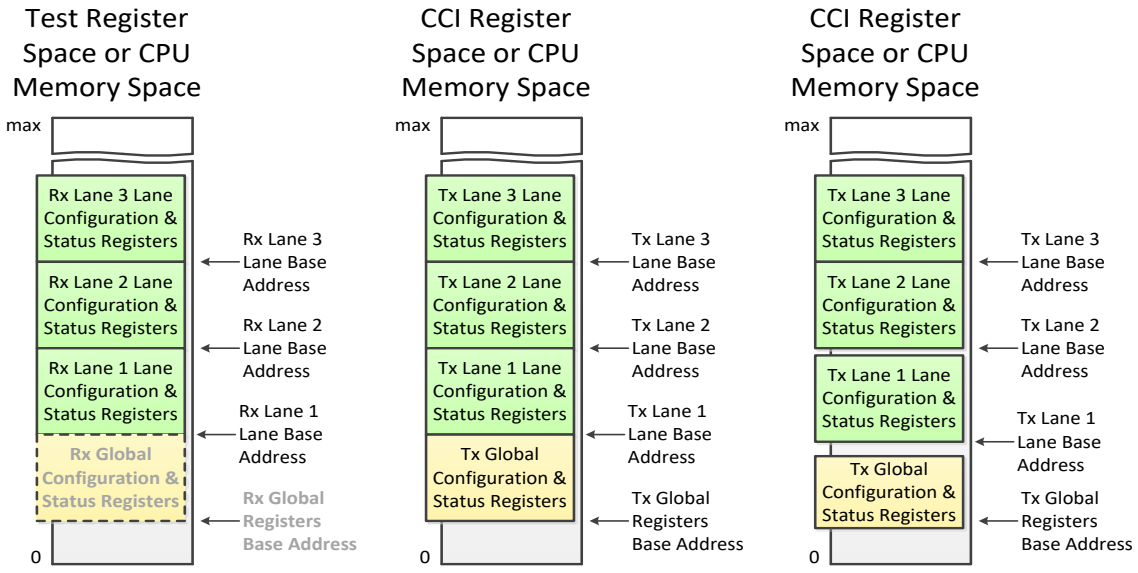


Figure 83 Configuration and Status Register mapping

12.2.2 Example of Register Access via CCI

Examples of suitable interfaces to access the C-PHY register space are: CSI-2 CCI in an image sensor, “flipped” CCI interface or AHB/APB in an application processor, or debug serial bus (JTAG or other) in a display driver IC. The C-PHY Global and Lane Configuration and Status Registers in the image sensor can be accessed easily via the CCI interface. These configuration and status registers in the image sensor can be written and read via CCI without any changes to the intended operating mode of CCI. It is only necessary to allocate space for the registers within the CCI register space. The registers in the application processor may

be accessed in a number of ways. One method is via a form of CCI, but a special test mode can be enabled where the CCI Master in the applications processor is disabled and a CCI Slave, or even an I2C Slave, is enabled instead. This CCI Slave or I2C Slave function is connected to the same SDA and SCL pins as the CCI Master, but the Slave is only enabled for test mode. The applications processor code will never simultaneously enable both the CCI Master and the Slave (either CCI or I2C). For normal system operation, the CCI Master is enabled and the CCI or I2C Slave is disabled. For test mode, the CCI or I2C Slave is enabled and the CCI Master is disabled. The recommended locations of CCI Master and CCI or I2C Slave in each mode, are shown in ***Figure 84***. Other command delivery options such as command bridging through an external device are shown as well.

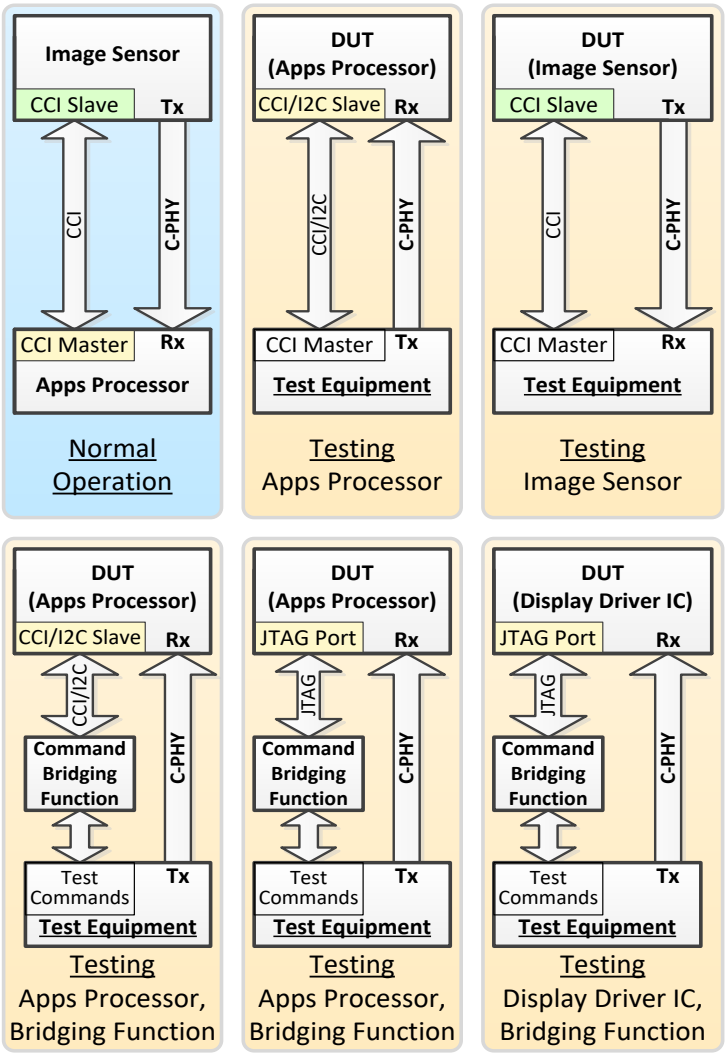


Figure 84 Use of CCI for Normal Operation and Test

The Rx Global and Lane Configuration and Status Registers can be accessed in a similar manner via the CCI Slave (for test mode) in the applications processor. It is also possible for the Rx Global and Lane Configuration and Status Registers to be mapped into the CPU memory space instead of using a special CCI Slave for test mode. The specific method of register access in the applications processor is an implementation choice. However, implementation of the specific function and address mapping of these registers defined in this Section is recommended.

12.2.3 Register Definitions

A high-level view of the global and Lane-specific test circuits is shown in ***Figure 85***. The following sections describe the global and Lane test functions for both Tx and Rx. The test circuits can be controlled and observed through registers such as those accessible via the CSI-2 CCI interface. The four primary groups of registers and the abbreviations used in the register names are as follows:

- TLRn – Tx Lane Register n, where n is the number of the Lane starting at 1. There is one set of Tx Lane Configuration and Status Registers per Lane. In a system having 6 Lanes, TLRn could be TLR1, TLR2, TLR3, TLR4, TLR5, or TLR6. n can be larger than 6 in chips supporting multiple camera or display ports. n is limited only by the size of the address space.
- RLRn – Rx Lane Register n, where n is the number of the Lane starting at 1. There is one set of Rx Lane Registers per Lane. In a system having 6 Lanes, RLRn could be RLR1, RLR2, RLR3, RLR4, RLR5, or RLR6. n can be larger than 6 in chips supporting multiple camera or display ports. n is limited only by the size of the address space.
- TGR – Tx Global Configuration and Status Registers, a set of read and write functions that apply to all Tx Lanes in a device, or to a defined group of Tx Lanes in a device.
- RGR – Rx Global Configuration and Status Registers, a set of read and write functions that apply to all Rx Lanes in a device, or to a defined group of Rx Lanes in a device.

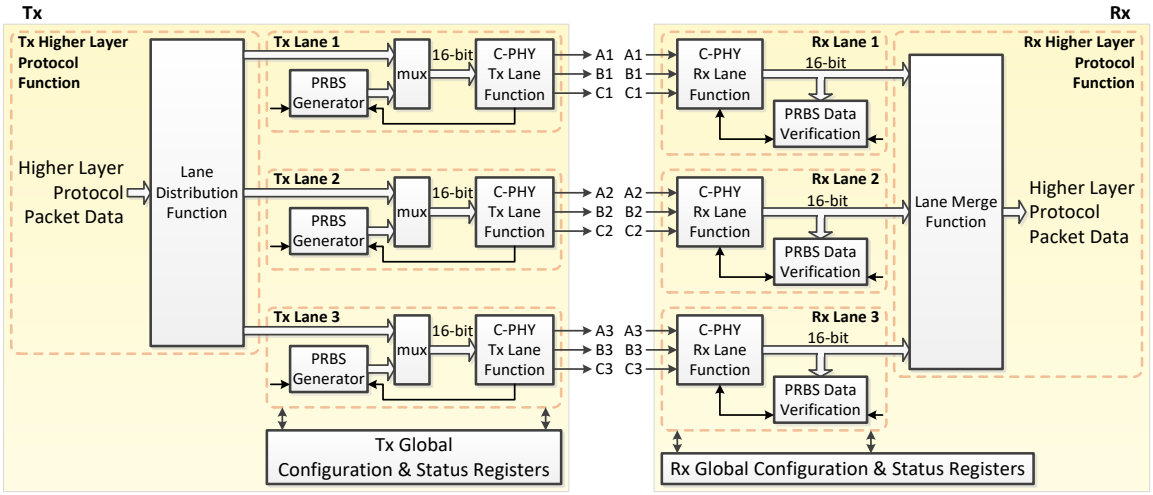
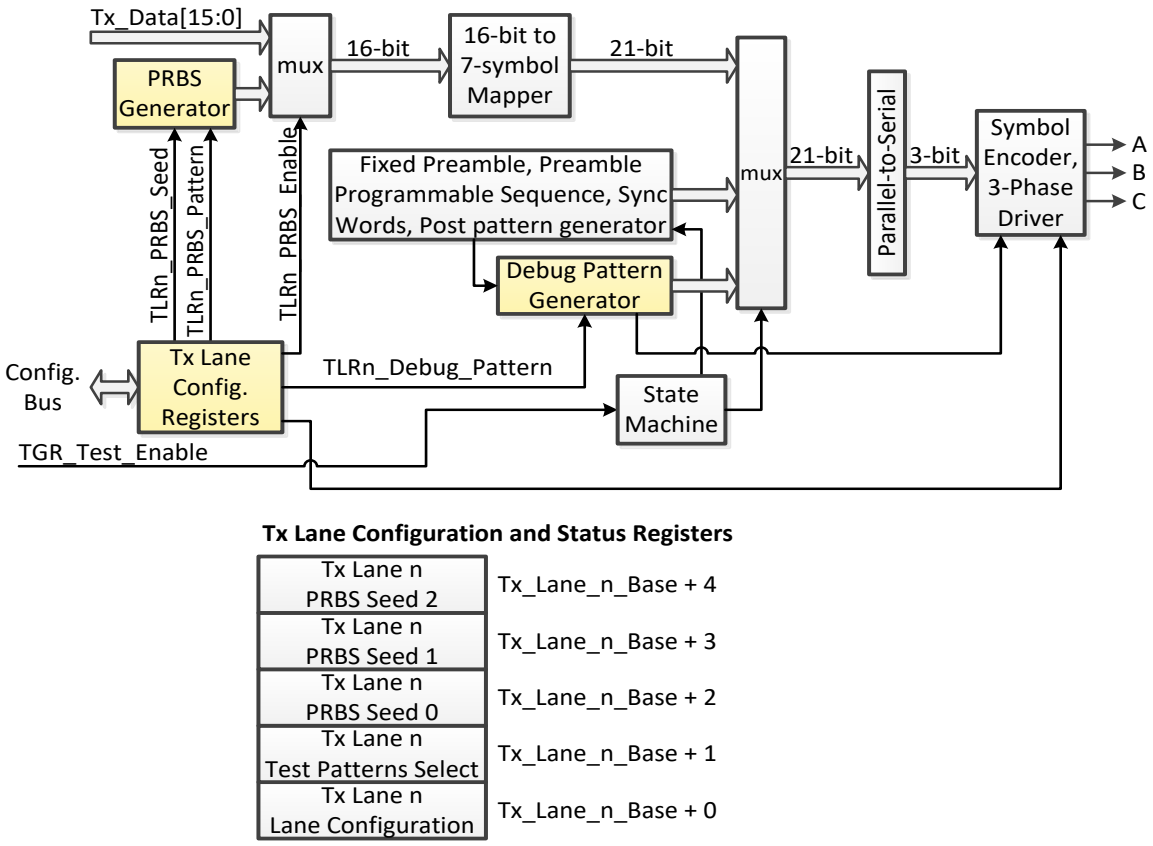


Figure 85 High-Level Tx and Rx, Global and Lane Functions

12.3 Tx Lane Test Circuitry

Figure 86 is a high-level block diagram of a single Tx Lane circuit. Test circuitry is indicated by yellow fill in specific blocks. There is one set of Tx Lane Configuration and Status Registers per Lane, so a 3-Lane system has three sets of Tx Lane Configuration and Status Registers. The definition of each register is described in more detail later in this Section.



Tx Lane Configuration and Status Registers

Tx Lane n PRBS Seed 2	Tx_Lane_n_Base + 4
Tx Lane n PRBS Seed 1	Tx_Lane_n_Base + 3
Tx Lane n PRBS Seed 0	Tx_Lane_n_Base + 2
Tx Lane n Test Patterns Select	Tx_Lane_n_Base + 1
Tx Lane n Lane Configuration	Tx_Lane_n_Base + 0

Figure 86 Transmit Lane Block Diagram with Test Circuitry

The Tx Lane Configuration and Status Registers have the following definitions:

12.3.1 TLRn_Lane_Configuration

write-only, Address: Tx_Lane_n_Base + 0

The Tx Lane n Lane Configuration register is used to configure parameters that are specific to the function of the Lane.

[7:0] – reserved for future use.	–

12.3.2 TLRn_Test_Patterns_Select

write-only, Address: Tx_Lane_n_Base + 1

The Tx Lane n Test Patterns Select register provides the means to choose a specific test pattern to be output by a transmit Lane function.

[7:5] – TLRn_PRBS_Pattern	=0 – select 16-bit Tx_Data[15:0] from Lane Distribution Function (normal operation) =1 to 3 – reserved for future use =4 – select PRBS9 =5 – select PRBS11 =6 – select PRBS18 =7 – reserved for future use
[4] – Reserved for future use	–

[3:0] – TLRn_Debug_Pattern	=0 – select output of 16-to7 Mapper (normal operation) or PRBS pattern, as selected by TLRn_PRBS_Pattern =1 – debug pattern is a sequence of 14 symbols defined by the same Tx Global Registers that define the Programmable Sequence of the Preamble =2 – debug pattern is a sequence of Wire States that are defined by the same Tx Global Registers that define the Programmable Sequence of the Preamble =3 to 15 – reserved for future use

Since the mux to select the TLRn_Debug_Pattern is "downstream" from the mux that selects the PRBS pattern, the TLRn_Debug_Pattern selection takes precedence over the TLRn_PRBS_Pattern setting. When the TLRn_Debug_Pattern selection is equal to 1 then the debug pattern is defined by the Tx Global Registers: TGR_Preamble_Prog_Sequence_0,1 [Tx_Global_Registers_Base_Address + 3] through TGR_Preamble_Prog_Sequence_12,13 [Tx_Global_Registers_Base_Address + 9]. These are the same registers that define the programmable sequence portion of the preamble. The 14-symbol debug pattern is repeated in the transmitted High-Speed data following the sync word as shown in ***Figure 87***.

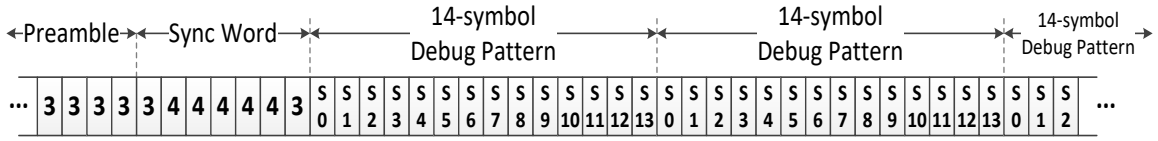


Figure 87 Repeating 14-Symbol Debug Pattern in High-Speed Data

If the TLRn_Debug_Pattern field is equal to 1 then the debug pattern is a sequence of 14 symbols defined by the same Tx Global Registers that define the Programmable Sequence of the Preamble. The symbols are defined using the same 3-bit Flip Rotation Polarity format that is described in ***Table 4***.

If the TLRn_Debug_Pattern field is equal to 2 then the debug pattern is a sequence of 14 Wire States defined by the same Tx Global Registers that define the Programmable Sequence of the Preamble. The Wire States are defined using the 3-bit format described in the table below. When TLRn_Debug_Pattern is set to 2 then the programmable sequence values are defined as Wire States rather than symbol values. The bit numbers in the column headings of the table correspond to bit numbers in the Transmit Global Registers TGR_Preamble_Prog_Sequence_0,1 through TGR_Preamble_Prog_Sequence_12,13.

	TGR_Preamble_ Prog_Sequence…[5]	**TGR_Preamble_ Prog_Sequence…[4]**	**TGR_Preamble_ Prog_Sequence…[3]**
Wire State	**TGR_Preamble_ Prog_Sequence…[2]**	**TGR_Preamble_ Prog_Sequence…[1]**	**TGR_Preamble_ Prog_Sequence…[0]**
+x	1	0	0
-x	0	1	1
+y	0	1	0
-y	1	0	1
+z	0	0	1
-z	1	1	0

Note that the first Wire State of the programmable sequence transmitted following the last bit of the Sync Word might happen to be the same Wire State transmitted during the last unit interval of the Sync Word. If this happens then there will be no Wire State transition at that unit interval boundary to generate a symbol clock pulse in a receiver. This is acceptable behavior because the purpose of the programmable Wire State debug pattern is to evaluate electrical and timing characteristics of the driver.

12.3.3 TLRn_PRBS_Seed_0

write-only, Address: Tx_Lane_n_Base + 2

The Tx Lane n PRBS Seed 0 register is an 8-bit value used to initialize the least significant 8 bits of the 18-bit Seed for the Tx Lane PRBS register. ***Figure 88*** shows the method to initialize the Transmit Lane PRBS register using the fragments of the Seed value.

[7:0] – seed value for Transmit PRBS register Q[8:1]	TLRn_PRBS_Seed_0[7] → Q[8]; through TLRn_PRBS_Seed_0[0] → Q[1];

12.3.4 TLRn_PRBS_Seed_1

write-only, Address: Tx_Lane_n_Base + 3

The Tx Lane n PRBS Seed 1 register is an 8-bit value used to initialize the next least significant 8 bits of the 18-bit Seed for the Tx Lane PRBS register. ***Figure 88*** shows the method to initialize the Transmit Lane PRBS register using the fragments of the Seed value.

[7:0] – seed value for Transmit PRBS register Q[16:9]	TLRn_PRBS_Seed_1[7] → Q[16]; through TLRn_PRBS_Seed_1[0] → Q[9];

12.3.5 TLRn_PRBS_Seed_2

write-only, Address: Tx_Lane_n_Base + 4

The Tx Lane n PRBS Seed 2 register is a 2-bit value used to initialize the most significant 2 bits of the 18-bit Seed for the Tx Lane PRBS register. ***Figure 88*** shows the method to initialize the Transmit Lane PRBS register using the fragments of the Seed value.

[7:2] – Reserved for future use	–
[1:0] – seed value for Transmit PRBS register Q[18:17]	TLRn_PRBS_Seed_2[1] → Q[18]; through TLRn_PRBS_Seed_2[0] → Q[17];

12.3.6 Tx Lane PRBS Register Operation

The Transmit Lane PRBS Register Q[16:1] is the source of data input to TxD[15:0] of the Mapper when the Lane transmit function is transmitting one of the three PRBS patterns as defined by the TLRn_Test_Patterns_Select register. The Transmit Lane PRBS Register function is defined in ***Figure 88***. The Transmit Lane PRBS register is initialized using the seed values TLRn_PRBS_Seed_0, TLRn_PRBS_Seed_1 and TLRn_PRBS_Seed_2, as shown in ***Figure 88***. The first word transmitted from the PRBS generator is equal to the seed value: [TLRn_PRBS_Seed_1[7:0], TLRn_PRBS_Seed_0[7:0]]. This initial 16-bit value from the PRBS register is transmitted immediately following transmission of the first Sync Word after the low-power to High-Speed mode transition. The Transmit Lane PRBS Register is shifted 16 bit positions after each 16-bit word is output to minimize correlation from one data value to the next. This way no bits are re-used in successive samples.

Example Seed values and data sequences for the chosen PRBS mode are as follows:

PRBS9 – Seed = 0x789a; TLRn_PRBS_Seed_0[7:0] = 0x9a; TLRn_PRBS_Seed_1[7:0] = 0x78; TLRn_PRBS_Seed_2[7:0] value does not matter;
Transmit data sequence: 0x789a, 0x9980, 0xc651, 0xa5fd, 0x163a, 0xcb3c, 0x7dd0…

PRBS11 – Seed = 0x789a; TLRn_PRBS_Seed_0[7:0] = 0x9a; TLRn_PRBS_Seed_1[7:0] = 0x78; TLRn_PRBS_Seed_2[7:0] value does not matter;
Transmit data sequence: 0x789a, 0x5e64, 0xfee0, 0xac43, 0xa9a1, 0xe4ce, 0xfea0…

PRBS18 – Seed = 0x2789a; TLRn_PRBS_Seed_0[7:0] = 0x9a; TLRn_PRBS_Seed_1[7:0] = 0x78; TLRn_PRBS_Seed_2[7:0] = 0x02;
Transmit data sequence: 0x789a, 0x8d77, 0x0dbc, 0x74e1, 0x8108, 0x414a, 0x3915…

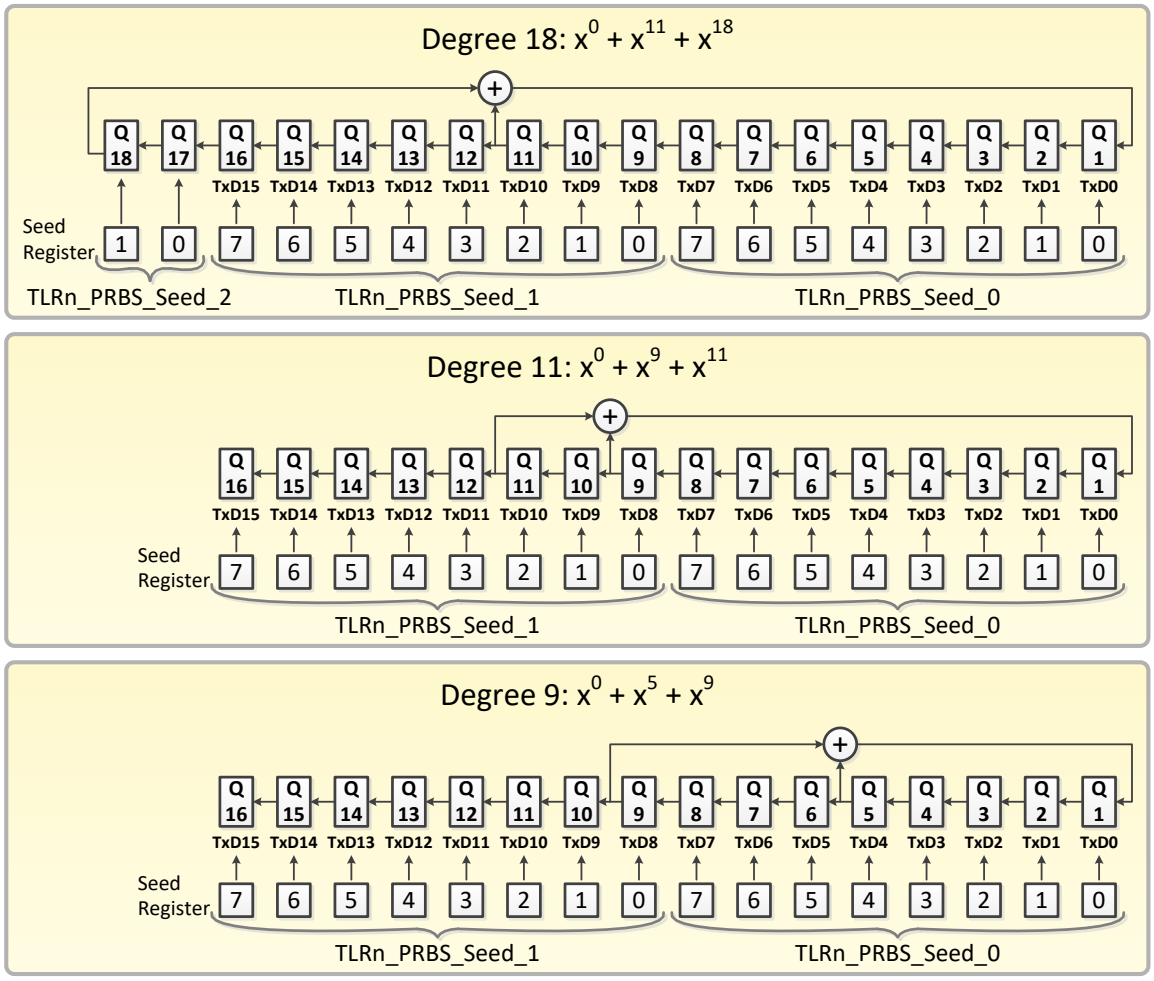
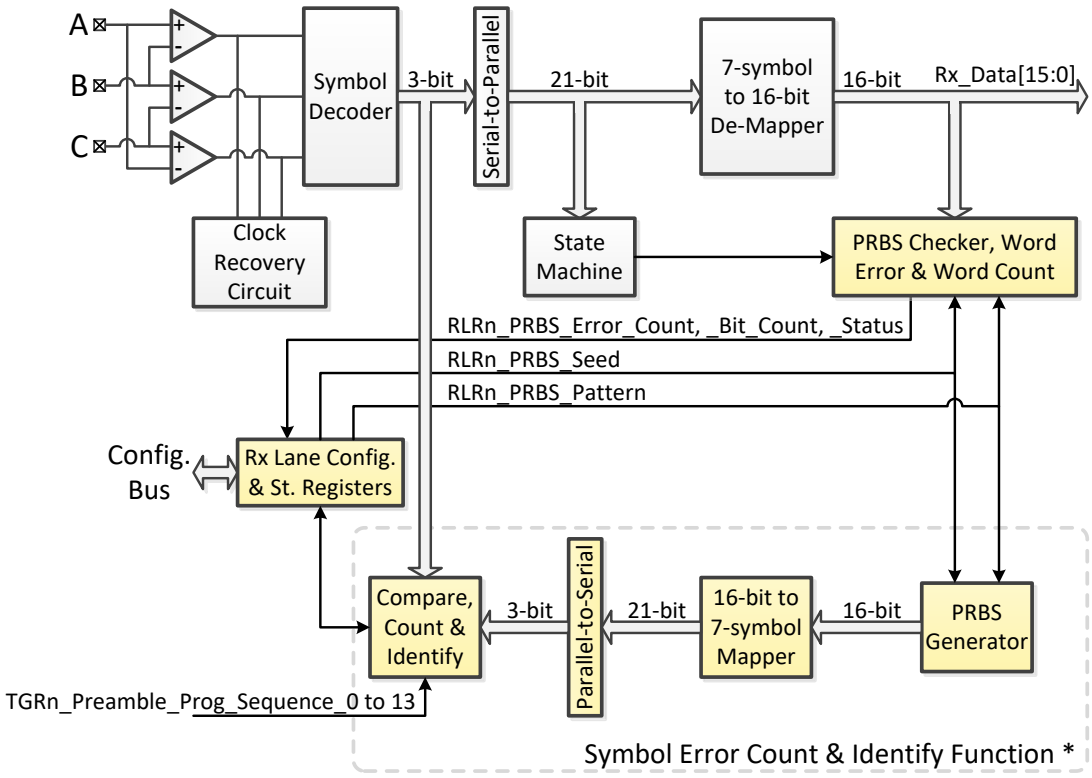


Figure 88 Tx Lane PRBS Register Function and Seed Value Initialization

12.4 Rx Lane Test Circuitry

Figure 89 shows a high-level block diagram of the Receive Lane with test circuitry.



Rx Lane Configuration and Status Registers

Rx Lane n 1st Sym Error Location 5 *	Rx_Lane_n_Base + 19	Rx Lane n Word Count 2	Rx_Lane_n_Base + 9
Rx Lane n 1st Sym Error Location 4 *	Rx_Lane_n_Base + 18	Rx Lane n Word Count 1	Rx_Lane_n_Base + 8
Rx Lane n 1st Sym Error Location 3 *	Rx_Lane_n_Base + 17	Rx Lane n Word Count 0	Rx_Lane_n_Base + 7
Rx Lane n 1st Sym Error Location 2 *	Rx_Lane_n_Base + 16	Rx Lane n Word Error Count	Rx_Lane_n_Base + 6
Rx Lane n 1st Sym Error Location 1 *	Rx_Lane_n_Base + 15	Rx Lane n PRBS Seed 2	Rx_Lane_n_Base + 5
Rx Lane n 1st Sym Error Location 0 *	Rx_Lane_n_Base + 14	Rx Lane n PRBS Seed 1	Rx_Lane_n_Base + 4
Rx Lane n Symbol Error Count *	Rx_Lane_n_Base + 13	Rx Lane n PRBS Seed 0	Rx_Lane_n_Base + 3
Rx Lane n Word Count 5	Rx_Lane_n_Base + 12	Rx Lane n Status	Rx_Lane_n_Base + 2
Rx Lane n Word Count 4	Rx_Lane_n_Base + 11	Rx Lane n PRBS Pattern	Rx_Lane_n_Base + 1
Rx Lane n Word Count 3	Rx_Lane_n_Base + 10	Rx Lane n Configuration	Rx_Lane_n_Base + 0

* a Symbol Error Count Function, remove if not needed

Figure 89 Receive Lane Block Diagram with Test Circuitry

The Rx Lane Configuration and Status Registers have the following definitions:

12.4.1 RLRn_Lane_Configuration

write-only, Address: Rx_Lane_n_Base + 0

The Rx Lane n Configuration register currently has no assigned bits in the register to affect the Lane function.

[7:0] – reserved for future use	–

12.4.2 RLRn_Test_Pattern_Select

write-only, Address: Rx_Lane_n_Base + 1

The Rx Lane n Test Pattern register provides the means to choose a specific PRBS pattern to be used by the PRBS Checker. The register contents also specify which pattern is to be used by the symbol error counting function, if the symbol error count feature is implemented.

[7:5] – RLRn_PRBS_Pattern_Select	=0 – disable error detection and counting =1 to 3 – reserved for future use =4 – select PRBS9 =5 – select PRBS11 =6 – select PRBS18 =7 – reserved for future use
[4] – reserved for future use	–
[3:0] – RLRn_Symbol_Error_Count_Function	=0 – the symbol error count function, if implemented, is controlled by bits [7:5]. The symbol error count compares received symbols with the selected PRBS sequence sent through the mapper; or if bits [7:5] are all zero then the symbol error count function is disabled. =1 – the symbol error count function, if implemented, compares received symbols with the user-defined debug pattern, which is defined by the same Tx Global Registers that define the Programmable Sequence of the Preamble. For a receive-only device, the receiver would need to implement the seven Tx Global Registers that define the programmable sequence, even if that device does not contain a Tx C-PHY function. =2 to 15 – reserved for future use

It is not necessary to enable or disable the error counting function. There is no harm in counting errors all of the time, even when actual packet data is being received. The system software will know to ignore the error counter value at that time. The enable/disable setting may be useful to prevent activity in the receiver error measurement system to slightly reduce power consumption when the capability to count errors is not required.

12.4.3 RLRn_Rx_Lane_Status

read-only, Address: Rx_Lane_n_Base + 2

The Rx Lane n Status register contains status indicators relating to important events that occur when a packet is received. All bits in the register are reset when the Lane detects the transition from LP-111 to LP-001. The Lane status logic keeps track of each event described in the RLRn_Rx_Lane_Status register. The register contents may be valid prior to the LP-000 to LP-111 transition. The register contents contain the actual status following the LP-000 to LP-111 transition.

[7] – LP-001 to LP-000 transition was detected.	1 – The transition was detected. 0 – The transition was not detected.
[6] – Preamble Programmable Sequence status (optional)	0 if not used. The specific function of this bit is determined by the device manufacturer.

[5] –Sync Word was detected.	1 – The Sync Word was detected. 0 – The Sync Word was not detected.
[4] – Post sequence was detected	1 – Post was detected. 0 – Post was not detected.
[3:0] – reserved for future use	–

12.4.4 RLRn_PRBS_Seed_0

write-only, Address: Rx_Lane_n_Base + 3

The Rx Lane n PRBS Seed 0 register is an 8-bit value used to initialize the least significant 8 bits of the 18-bit Seed for the Receive Lane PRBS register. ***Figure 90*** shows the method to initialize the PRBS register using the fragments of the Seed value.

[7:0] – seed value for Receive PRBS register Q[8:1]	RLRn_PRBS_Seed_0[7] → Q[8]; through RLRn_PRBS_Seed_0[0] → Q[1];

12.4.5 RLRn_PRBS_Seed_1

write-only, Address: Rx_Lane_n_Base + 4

The Rx Lane n PRBS Seed 1 register is an 8-bit value used to initialize the next least significant 8 bits of the 18-bit Seed for the Receive Lane PRBS register. ***Figure 90*** shows the method to initialize the PRBS register using the fragments of the Seed value.

[7:0] – seed value for Receive PRBS register Q[16:9]	RLRn_PRBS_Seed_1[7] → Q[16]; through RLRn_PRBS_Seed_1[0] → Q[9];

12.4.6 RLRn_PRBS_Seed_2

write-only, Address: Rx_Lane_n_Base + 5

The Rx Lane n PRBS Seed 2 register is a 2-bit value used to initialize the most significant 2 bits of the 18-bit Seed for the Receive Lane PRBS register. ***Figure 90*** shows the method to initialize the PRBS register using the fragments of the Seed value.

[7:2] – Reserved for future use	
[1:0] – seed value for Receive PRBS register Q[18:17]	RLRn_PRBS_Seed_2[1] → Q[18]; through RLRn_PRBS_Seed_2[0] → Q[17];

12.4.7 Rx Lane PRBS Register Operation

The Receive Lane PRBS Register Q[16:1] is the reference data that is compared with RxD[15:0] from the De-Mapper when the Lane receive function is operating in one of the three PRBS test modes as defined by the RLRn_PRBS_Pattern register. The Receive Lane PRBS Register function is defined in ***Figure 90***. The Receive Lane PRBS Register is initialized using the seed values RLRn_PRBS_Seed_0, RLRn_PRBS_Seed_1 and RLRn_PRBS_Seed_2, as shown in ***Figure 90***. The first reference word from the Receive Lane PRBS Register is equal to the seed value: [RLRn_PRBS_Seed_1[7:0], RLRn_PRBS_Seed_0[7:0]]. This first word from the Receive Lane PRBS Register is compared with the first word of received data immediately following the first Sync Word after the low-power to High-Speed mode transition. The Receive Lane PRBS Register is shifted 16 bit positions after each 16-bit word of reference data is output to correspond exactly with the data that was transmitted.

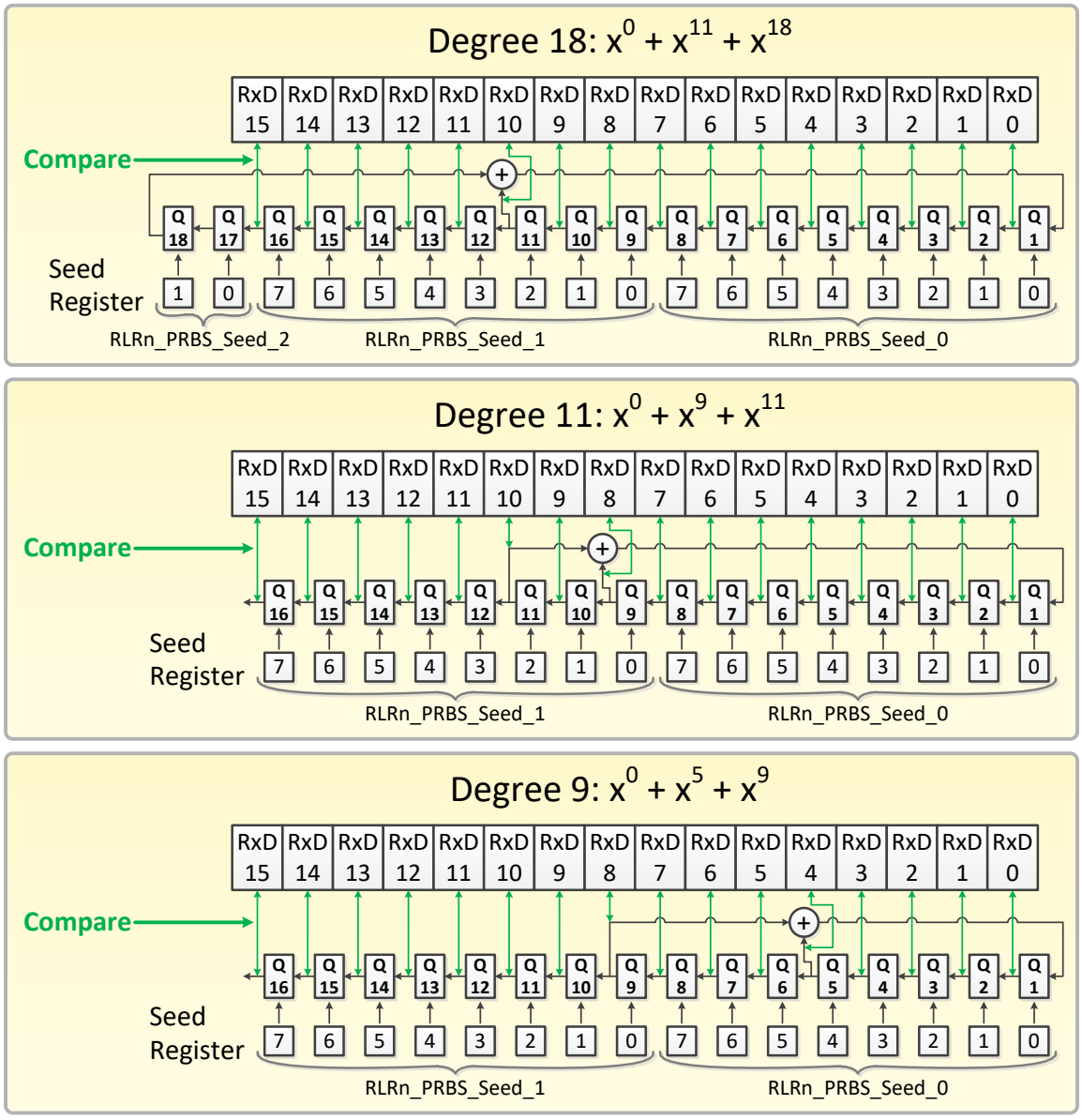


Figure 90 Rx Lane PRBS Register Function and Seed Value Initialization

12.4.8 Rx Lane Word Error Count and Word Count Functionality

The Rx Lane Word Error Count is a count of the number of word errors that were detected in the corresponding Lane. If there is one or more bit errors in the comparison of RxD[15:0] versus Q[16:1] of the Receive Lane PRBS Register then the error count is incremented by one. (Note that if there are multiple errors in the comparison of one received data word, then the Rx Lane Word Error Count is incremented by only one count.) The Rx Lane Word Error Count saturates at 255, so if more than 255 errors are detected then the count stops at 255 and will not roll over to zero. The word error count is reset to zero at the transition from low-power to High-Speed mode (the LP-001 to LP-000 transition), and any detected word errors are counted beginning at the first De-Mapper output word of the Packet Data field following the Sync Word field.

The Rx Lane Word Count described below is a 48-bit integer value. The Rx Lane Word Count is recorded by the receiver because the duration of the High-Speed data is controlled by an enable/disable bit in the

Tx_Global_Configuration_Register. Knowing the word count makes it possible to accurately compute the word error rate or symbol error rate. The Rx Lane Word Count word count is sufficiently large to perform an error rate test at 2.5Gsps for slightly more than 9 days. A single word error over this maximum test interval corresponds to a symbol error rate of about $5 \cdot 10^{-16}$. If there are any symbol errors that cause missing symbol clocks then the word count will be reduced by 1/7th of a word for each such occurrence, and the error count will most likely saturate in that instance, so the calculated error rate would not be meaningful. It is anticipated that even with such a large word counter, the symbol error rates and word error rates will not be measurable unless the signal amplitude or channel conditions are degraded beyond the required limits specified in this document. If symbol errors are counted to determine the symbol error rate, then it is necessary to multiply the word count by 7 to know the symbol count for the symbol error rate calculation:

$$symbol\_error\_rate = symbol\_error\_count/(word\_count \cdot 7)$$

All 48 bits of the Rx Lane Word Count are reset to zero on the transition from low-power mode to High-Speed mode (on the transition from LP-001 to LP-000). The Rx Lane Word Count can be read after the transition back to low-power mode (following the transition from LP-000 to LP-111). If the Rx Lane Word Count is read during the error rate measurement test in High-Speed mode then the count can contain an invalid result, depending on the implementation of the word count read circuit.

12.4.9 RLRn_Word_Error_Count

read-only, Address: Rx_Lane_n_Base + 6

The Rx Lane n Word Error Count is a count of the number of word errors that were detected in the corresponding Lane.

[7:0] – RLRn_Word_Error_Count	Rx Lane Word Error Count

12.4.10 RLRn_Word_Count_0

read-only, Address: Rx_Lane_n_Base + 7

Rx Lane n Word Count 0 consists of bits 7 through 0 of the 48-bit Rx Lane Word Count value.

[7:0] – RLRn_Word_Count_0	Rx Lane Word Count[7:0]

12.4.11 RLRn_Word_Count_1

read-only, Address: Rx_Lane_n_Base + 8

Rx Lane n Word Count 1 consists of bits 15 through 8 of the 48-bit Rx Lane Word Count value.

[7:0] – RLRn_Word_Count_1	Rx Lane Word Count[15:8]

12.4.12 RLRn_Word_Count_2

read-only, Address: Rx_Lane_n_Base + 9

Rx Lane n Word Count 2 consists of bits 23 through 16 of the 48-bit Rx Lane Word Count value.

[7:0] – RLRn_Word_Count_2	Rx Lane Word Count[23:16]

12.4.13 RLRn_Word_Count_3

read-only, Address: Rx_Lane_n_Base + 10

Rx Lane n Word Count 3 consists of bits 31 through 24 of the 48-bit Rx Lane Word Count value.

[7:0] – RLRn_Word_Count_3	Rx Lane Word Count[31:24]

12.4.14 RLRn_Word_Count_4

read-only, Address: Rx_Lane_n_Base + 11

Rx Lane n Word Count 4 consists of bits 39 through 32 of the 48-bit Rx Lane Word Count value.

[7:0] – RLRn_Word_Count_4	Rx Lane Word Count[39:32]

12.4.15 RLRn_Word_Count_5

read-only, Address: Rx_Lane_n_Base + 12

Rx Lane n Word Count 5 consists of bits 47 through 40 of the 48-bit Rx Lane Word Count value.

[7:0] – RLRn_Word_Count_5	Rx Lane Word Count[47:40]

12.4.16 Symbol Error Count and Symbol Error Location Functionality

The symbol error count and symbol error location capabilities might have less value than the other built-in-test capabilities; it may be sufficient to implement only the data generation and word error counting capabilities described above.

To measure the symbol error count it is necessary to duplicate the data generation, Mapping and Encoding functions of the transmitter to create a copy of the transmitted symbol stream. Then the received symbol stream can be compared to the regenerated symbol stream and any differences are counted and can be read via the RLRn_Symbol_Error_Count register. Any errors that result in a slip of the symbol clock will likely cause the symbol error count to saturate immediately following the clock slip event. It is anticipated that this is an extremely unlikely event when the Link is operated under the required conditions, but the possibility of this failure is noted so the system designer is aware of it.

The Receive Lane Symbol Error Count is a count of the number of symbol errors that were detected in the corresponding Lane. The Receive Lane Symbol Error Count saturates at 255, so if more than 255 errors are detected then the count stops at 255 and will not roll-over to zero. The Receive Lane Symbol Error Count is reset to zero at the transition from low-power to High-Speed mode, and errors are counted beginning at the first symbol of the Packet Data field following the first Sync Word after the low-power to High-Speed mode transition.

The Receive Lane 1st Symbol Error Location values are a debug capability that allows the symbol position of the first error in the Packet Data Field to be identified. This is the value of a rather long counter (a 48-bit counter) that identifies the symbol offset of the location of the first symbol error in the Packet Data field.

12.4.17 RLRn_Sym_Error_Count

read-only, Address: Rx_Lane_n_Base + 13

Receive Lane Symbol Error Count is a count of the number of symbol errors that were detected in Lane n.

[7:0] – RLRn_Sym_Error_Count	Receive Lane Symbol Error Count

12.4.18 RLRn_1st_Sym_Err_Loc_0

read-only, Address: Rx_Lane_n_Base + 14

RLRn_1st_Sym_Err_Loc_0 consists of bits 7 through 0 of the 48-bit Receive Lane 1st Symbol Error Location value.

[7:0] – RLRn_1st_Sym_Err_Loc_0	Receive Lane 1st Symbol Error Location[7:0]

12.4.19 RLRn_1st_Sym_Err_Loc_1

read-only, Address: Rx_Lane_n_Base + 15

RLRn_1st_Sym_Err_Loc_1 consists of bits 15 through 8 of the 48-bit Receive Lane 1st Symbol Error Location value.

[7:0] – RLRn_1st_Sym_Err_Loc_1	Receive Lane 1st Symbol Error Location[15:8]

12.4.20 RLRn_1st_Sym_Err_Loc_2

read-only, Address: Rx_Lane_n_Base + 16

RLRn_1st_Sym_Err_Loc_2 consists of bits 23 through 16 of the 48-bit Receive Lane 1st Symbol Error Location value.

[7:0] – RLRn_1st_Sym_Err_Loc_2	Receive Lane 1st Symbol Error Location[23:16]

12.4.21 RLRn_1st_Sym_Err_Loc_3

read-only, Address: Rx_Lane_n_Base + 17

RLRn_1st_Sym_Err_Loc_3 consists of bits 31 through 24 of the 48-bit Receive Lane 1st Symbol Error Location value.

[7:0] – RLRn_1st_Sym_Err_Loc_3	Receive Lane 1st Symbol Error Location[31:24]

12.4.22 RLRn_1st_Sym_Err_Loc_4

read-only, Address: Rx_Lane_n_Base + 18

RLRn_1st_Sym_Err_Loc_4 consists of bits 39 through 32 of the 48-bit Receive Lane 1st Symbol Error Location value.

[7:0] – RLRn_1st_Sym_Err_Loc_4	Receive Lane 1st Symbol Error Location[39:32]

12.4.23 RLRn_1st_Sym_Err_Loc_5

read-only, Address: Rx_Lane_n_Base + 19

RLRn_1st_Sym_Err_Loc_5 consists of bits 47 through 40 of the 48-bit Receive Lane 1st Symbol Error Location value.

[7:0] – RLRn_1st_Sym_Err_Loc_5	Receive Lane 1st Symbol Error Location[47:40]

12.5 Tx Global Configuration and Status Registers

The Tx Global Configuration and Status Registers have the following definitions:

12.5.1 TGR_Global_Configuration

write-only, Address: Tx_Global_Registers_Base + 0

TGR_Global_Configuration is a register to configure parameters and operate controls that apply to all C-PHY Lanes.

[7:1] – reserved for future use.	–
[0] – TGR burst enable/disable	TGR burst enable/disable, starts or stops the High-Speed test burst. =0 – Disable sending High-Speed test data =1 – Enable sending High-Speed test data

12.5.2 Burst Enable/Disable Functionality

When the TGR burst enable/disable bit transitions from 0 to 1, the Lane state machines should transmit LP-001, then LP-000, then Preamble, then Sync Word, then test data is sent in the Packet Data field either from the Tx Lane PRBS Generator or Debug Pattern Generator. The specific data transmitted depends on the values written to the TLRn_Test_Patterns_Select register. The Packet Data field contains only the selected test data and has no high layer protocol packet structure. The selected test data will be sent continuously, as long as the burst enable/disable bit is set to “1”. There is no limit to the length of the Packet Data field when

test data is being transmitted. The C-PHY Lane circuit begins in the LP-111 state to respond to the 0-to-1 transition of the TGR burst enable/disable bit. If the C-PHY Lane circuit is not in the LP-111 state during the 0-to-1 transition event then the C-PHY Lane will ignore the state of the TGR burst enable/disable bit until it returns to the 0 state and has a subsequent 0-to-1 transition when the Lane is in the LP-111 state.

When the TGR burst enable/disable bit transitions from 1 to 0 while sending test data, the Lane state machine causes the Post Sequence (4,4,4,4,4,4,4) to be sent and repeated by the number of times defined in TGR_Post_Length, and then the termination is disabled and the signals return to the LP-111 state. ***Figure 91*** illustrates the functionality resulting from TGR burst enable/disable changing state.

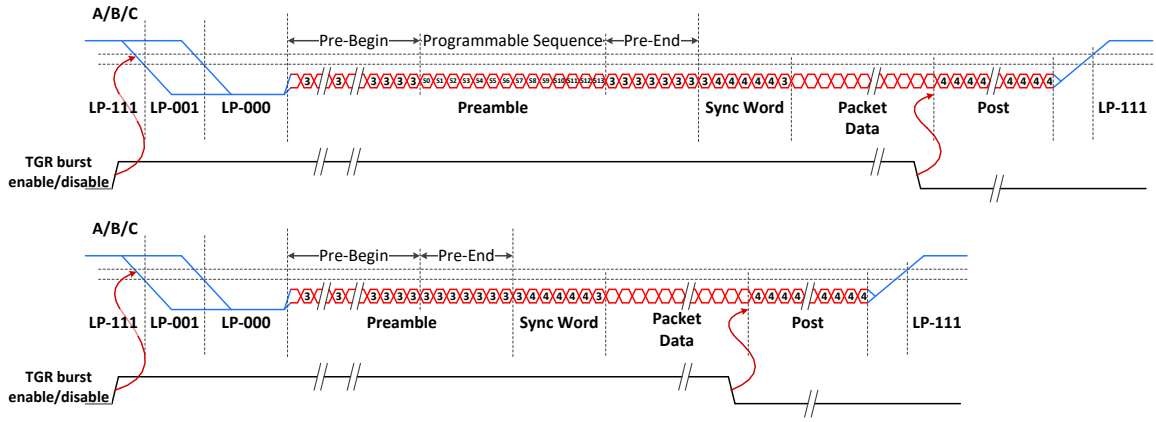


Figure 91 Example Showing Cause/Effect of TGR burst enable/disable

The Lane receiver test circuitry contains a word count field that can be read from RLRn_Word_Count_0 through RLRn_Word_Count_5. From this, it is possible to compute the word error rate, or symbol error rate by also reading the word error count and symbol error count values.

Note that there are not specific global registers defined for selection of the PRBS polynomial or PRBS seed value. This allows the PRBS polynomial and seed to be chosen on a per-Lane basis. When multiple Lanes are tested simultaneously, it is anticipated that the most frequent use will be to select the same PRBS polynomial in all Lanes and use a different seed value in each Lane so that the transmitted data pattern in each Lane is independent of the others.

12.5.3 TGR_Preamble_Length

write-only, Address: Tx_Global_Registers_Base + 1

TGR_Preamble_Length specifies the length of the Preamble and provides a means to enable or disable the Programmable Sequence in the Preamble.

[7] – enable/disable the Preamble Programmable Sequence	Enable or disable the Preamble Programmable Sequence, refer to ***Figure 92*** =0 – Disable the Preamble Programmable Sequence, the lower waveforms of ***Figure 92*** =1 – Enable the Preamble Programmable Sequence, the upper waveforms of ***Figure 92*** The default value is 0, which disables the Preamble Programmable Sequence on system reset.
[6] – reserved for future use.	–
[5:0] – Begin_Preamble_Length	The number of symbols in the PreBegin section of the preamble is: (Begin_Preamble_Length + 1) · 7

	The default value is 0x3f or 63, which sets the length of the PreBegin part of the Preamble to 64 Words on system reset.

The PreBegin part of the Preamble may range from 1 to 64 Words, or 7 to 448 symbols.

12.5.4 TGR_Post_Length

write-only, Address: Tx_Global_Registers_Base + 2

TGR_Post_Length specifies the length of the Post field.

[7:5] – reserved for future use.	–
[4:0] – Post_Length	The number of symbols in the Post field is: (Post_Length + 1) · 7 The default value is 0x1f or 31, which sets the length of the PreBegin part of the Preamble to 32 Words on system reset.

The Post field may range from 1 to 32 Words, or 7 to 224 symbols.

12.5.5 TGR_Preamble_Prog_Sequence_0,1

write-only, Address: Tx_Global_Registers_Base + 3

TGR_Preamble_Prog_Sequence_0,1 specifies the values of symbols 0 and 1 in the Programmable Sequence portion of the Preamble.

[7:6] – reserved for future use.	–
[5:3] – Symbol 1 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.
[2:0] – Symbol 0 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.

12.5.6 TGR_Preamble_Prog_Sequence_2,3

write-only, Address: Tx_Global_Registers_Base + 4

TGR_Preamble_Prog_Sequence_2,3 specifies the values of symbols 2 and 3 in the Programmable Sequence portion of the Preamble.

[7:6] – reserved for future use.	–
[5:3] – Symbol 3 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.
[2:0] – Symbol 2 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.

12.5.7 TGR_Preamble_Prog_Sequence_4,5

write-only, Address: Tx_Global_Registers_Base + 5

TGR_Preamble_Prog_Sequence_4,5 specifies the values of symbols 4 and 5 in the Programmable Sequence portion of the Preamble.

[7:6] – reserved for future use.	–

[5:3] – Symbol 5 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.
[2:0] – Symbol 4 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.

12.5.8 TGR_Preamble_Prog_Sequence_6,7

write-only, Address: Tx_Global_Registers_Base + 6

TGR_Preamble_Prog_Sequence_6,7 specifies the values of symbols 6 and 7 in the Programmable Sequence portion of the Preamble.

[7:6] – reserved for future use.	–
[5:3] – Symbol 7 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.
[2:0] – Symbol 6 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.

12.5.9 TGR_Preamble_Prog_Sequence_8,9

write-only, Address: Tx_Global_Registers_Base + 7

TGR_Preamble_Prog_Sequence_8,9 specifies the values of symbols 8 and 9 in the Programmable Sequence portion of the Preamble.

[7:6] – reserved for future use.	–
[5:3] – Symbol 9 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.
[2:0] – Symbol 8 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.

12.5.10 TGR_Preamble_Prog_Sequence_10,11

write-only, Address: Tx_Global_Registers_Base + 8

TGR_Preamble_Prog_Sequence_10,11 specifies the values of symbols 10 and 11 in the Programmable Sequence portion of the Preamble.

[7:6] – reserved for future use.	–
[5:3] – Symbol 11 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.
[2:0] – Symbol 10 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.

12.5.11 TGR_Preamble_Prog_Sequence_12,13

write-only, Address: Tx_Global_Registers_Base + 9

TGR_Preamble_Prog_Sequence_12,13 specifies the values of symbols 12 and 13 in the Programmable Sequence portion of the Preamble.

[7:6] – reserved for future use.	–
[5:3] – Symbol 13 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.
[2:0] – Symbol 12 of the Preamble Programmable Sequence	=0 to 4 – symbol value =5 to 7 – invalid value, not a valid symbol value. The default value is 3 on system reset.

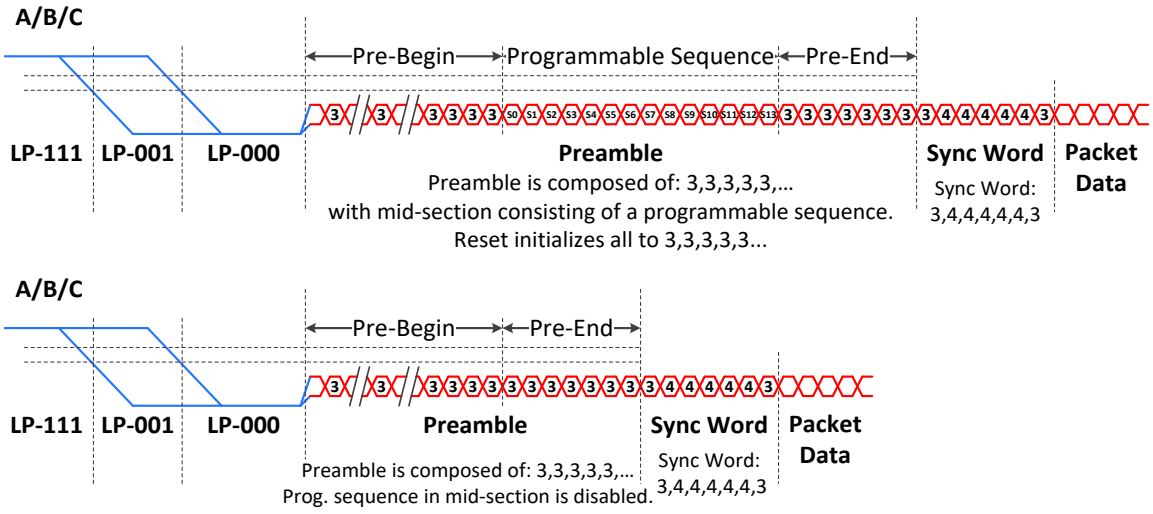


Figure 92 Preamble Programmable Sequence, Showing Bit Order, and Enabled/Disabled

12.6 Rx Global Configuration and Status Registers

There are no anticipated needs for Rx Global Configuration and Status Registers.

Annex A Logical PHY-Protocol Interface Description (Informative)

The PHY-Protocol Interface (PPI) is used to make a connection between the PHY Lane Modules and the higher protocol layers of a communication stack. The interface described here is intended to be generic and application independent.

This annex is informative only. Conformance to the C-PHY specification does not depend on any portion of the PPI defined herein. Because of that, this Section avoids normative language and does not use words like "shall" and "should." Instead, present tense language has been used to describe the PPI, utilizing words like "is" and "does." The reader may find it helpful to consider this annex to be a description of an example implementation, rather than a specification.

This PPI is optimized for controlling a C-PHY and transmitting and receiving parallel data. The interface described here is defined as an on-chip connection, and does not attempt to minimize signal count or define timing parameters or voltage levels for the PPI signals.

A.1 Signal Description

Table 61 defines the signals used in the PPI. For a PHY with multiple Lanes, a set of PPI signals is used for each Lane. Each signal has been assigned into one of six categories: High-Speed transmit signals, High-Speed receive signals, Escape Mode transmit signals, Escape Mode receive signals, control signals, and error signals. Bi-Directional High-Speed Lanes with support for Bi-Directional Escape Mode include nearly all of the signals listed in the table. Unidirectional Lanes include only a subset of the signals. The direction of each signal is listed as "I" or "O". Signals with the direction "I" are PHY inputs, driven from the protocol layer. Signals with the direction "O" are PHY outputs, driven to the protocol layer. For this logical interface, most clocks are described as being generated outside the PHY, although any specific PHY may implement the clock circuit differently.

The "Categories" column in ***Table 61*** indicates for which Lane Module types each signal applies. The category names are described in ***Table 2*** and are summarized here for convenience. Each category is described using a four-letter acronym, defined as [Side, HS-capabilities, Escape-Forward, Escape-Reverse]. The first letter, Side, can be M (Master) or S (Slave). The second letter, High-Speed capabilities, can be F (forward data) or R (Reverse and Forward data). The third and fourth letters indicate Escape Mode capability in the Forward and Reverse Directions, respectively. The third letter can be A (All) or E (Events – Triggers and ULPS only), while the fourth letter can be A (All, including LPDT), E (Events, triggers and ULPS only), Y (Any but not None: so A or E) or N (None). Any of the four identification letters can be replaced by an X, to indicate that each of the available options is appropriate.

The signal description includes options for the designer to choose a data path width to simplify the task of timing closure between the C-PHY and high-level protocol logic. It is not necessary for the PPI data path width of the transmit function in one IC to match the PPI data path width of the receive function in another IC. The C-PHY has the ability to transmit and receive any integer number of words greater than zero, regardless of the width of the PPI Tx and Rx data paths. A set of data valid signals accompany each set of data transferred over the PPI to indicate which words contain valid data to transmit or which words contain data that was actually received from the channel.

2218 **Table 61 PPI Signals**

Symbol	Dir	Categories	Description
High-Speed Transmit Signals			
TxSymbolClkHS	I	MXXX SRXX	Lane High-Speed Transmit Symbol Clock. This clock provides the timing used to transmit High-Speed symbol data over the Lane interconnect. All Lanes may use the same TxSymbolClkHS clock signal, or any subset of Lanes down to a single Lane may use a TxSymbolClkHS. Lanes may use different TxSymbolClkHS clocks as long as the inter-Lane skew requirement is met.
TxWordClkHS	O	MXXX SRXX	High-Speed Transmit Word Clock. This is used to synchronize PPI signals in the High-Speed transmit clock domain. It is recommended that all transmitting Lane Modules share one TxWordClkHS signal. The frequency of TxWordClkHS is dependent upon the width of the High-Speed Transmit Data, as follows: • 16-bit width, TxDataHS[15:0], the High-Speed Transmit Word Clock is exactly 1/7 the High-Speed symbol rate. • 32-bit width, TxDataHS[31:0], the High-Speed Transmit Word Clock is exactly 1/14 the High-Speed symbol rate.
TxDataWidthHS[1:0]	I	MXXX SRXX	High-Speed Transmit Data Width Select. Selects the bus width of TxDataHS: • TxDataWidthHS[1:0] = 00: not used, reserved. • TxDataWidthHS[1:0] = 01: 16-bit, TxDataHS[15:0] • TxDataWidthHS[1:0] = 10: 32-bit, TxDataHS[31:0] • TxDataWidthHS[1:0] = 11: not used, reserved. An implementation may support any data width - one fixed width, or subset of widths or all widths defined above.
TxDataHS[15:0], or TxDataHS[31:0]	I	MXXX SRXX	High-Speed Transmit Data. High-Speed data to be transmitted. The signal connected to TxDataHS[0] is associated with bit 0 of the C-PHY Mapping function. If a 32-bit transmit data path is used then TxDataHS[16] is also associated with bit 0 of the C-PHY Mapping function, and if TxWordValidHS[1:0] = 11, then the seven symbols associated with TxDataHS[31:16] are transmitted after the seven symbols associated with TxDataHS[15:0]. Data is captured on rising edges of TxWordClkHS. The following signals are defined for the High-Speed Transmit Data based on the width of the transmit data path: • 16-bit width – TxDataHS[15:0] • 32-bit width – TxDataHS[31:0] An implementation may support any data width - one fixed width, or subset of widths or all widths defined above.

Symbol	Dir	Categories	Description
TxWordValidHS[0], or TxWordValidHS[1:0]	I	MXXX SRXX	High-Speed Transmit Word Data Valid. When the High-Speed Transmit Data width is greater than 16 bits it is necessary to indicate which 16-bit segments contain valid transmit data to be able to transmit any number of words. The following Transmit Sync Word signals are defined based on the width of the transmit data path: • 16-bit width – TxWordValidHS[0] • 32-bit width – TxWordValidHS[1:0] The following Transmit Word Data Valid signals indicate which bits of the TxDataHS data bus contain valid data to transmit as follows: • TxWordValidHS[0] –TxDataHS[15:0] contains valid data to be transmitted • TxWordValidHS[1] – TxDataHS[31:16] contains valid data to be transmitted
TxSendSyncHS[0], or TxSendSyncHS[1:0] (note 1)	I	MXXX SRXX	High-Speed Command to Transmit Sync Word. The protocol adapter attached to the C-PHY may need to transmit Sync Words to separate multiple copies of a packet header. This command signal has the same timing as High-Speed Transmit Data on the PPI, but when a TxSendSyncHS signal is active on a given TxWordClkHS cycle then the corresponding 16 bits of High-Speed Transmit Data is ignored and the C-PHY transmits a Sync Word in place of the corresponding High-Speed Data Word for any Word Clock cycle where a TxSendSyncHS signal is active. The following Transmit Sync Word signals are defined based on the width of the transmit data path: • 16-bit width – TxSendSyncHS[0] • 32-bit width – TxSendSyncHS[1:0] The following Transmit Sync Word signals cause a Sync Word to be transmitted as follows: • TxSendSyncHS[0] – transmit a Sync Word in place of the C-PHY mapped version of TxDataHS[15:0] • TxSendSyncHS[1] – transmit a Sync Word in place of the C-PHY mapped version of TxDataHS[31:16]
TxSyncTypeHS0[2:0], and TxSyncTypeHS1[2:0]	I	MXXX SRXX	High-Speed Command to choose the Sync Type. When TxSendSyncHS[0] is asserted then TxSyncTypeHS0[2:0] selects which type of Sync Word is transmitted. Similarly, When TxSendSyncHS[1] is asserted then TxSyncTypeHS1[2:0] selects which type of Sync Word is transmitted. The mapping of Sync Type to the actual Sync Word value is specified in ***Section 6.4.4.2***. The default Sync Word, 3444443, is Sync Type 3 and is selected using the binary value 011. The following Sync Type Command signals are defined based on the width of the receive data path: • 16-bit width – TxSyncTypeHS0[2:0] • 32-bit width – TxSyncTypeHS0[2:0] and TxSyncTypeHS1[2:0]

Symbol	Dir	Categories	Description
TxRequestHS	I	MXXX SRXX	High-Speed Transmit Request and Data Valid. A low-to-high transition on TxRequestHS causes the Lane Module to initiate a Start-of-Transmission sequence. A high-to-low transition on TxRequest causes the Lane Module to initiate an End-of-Transmission sequence. This active high signal also indicates that the protocol layer is driving valid data on TxDataHS to be transmitted. The Lane Module accepts the data when both TxRequestHS and TxReadyHS are active on the same rising TxWordClkHS clock edge. The protocol layer always provides valid transmit data when TxRequestHS is active. Once asserted, TxRequestHS remains high until the data has been accepted, as indicated by TxReadyHS. TxRequestHS is only asserted while TxRequestEsc is low.
TxReadyHS	O	MXXX SRXX	High-Speed Transmit Ready. This active high signal indicates that TxDataHS is accepted by the Lane Module to be serially transmitted. TxReadyHS is valid on rising edges of TxWordClkHS.
High-Speed ALP Transmit Signals			
TxSendALPHS[0], or TxSendALPHS[1:0] (note 1)	I	MXXX SRXX	High-Speed Command to Transmit ALP Code. The protocol adapter attached to the C-PHY may need to transmit ALP Codes. This command signal has the same timing as High-Speed Transmit Data on the PPI, but when a TxSendALPHS signal is active on a given TxWordClkHS cycle then the corresponding 16 bits of High-Speed Transmit Data is ignored and the C-PHY transmits an ALP Code in place of the corresponding High-Speed Data Word for any Word Clock cycle where a TxSendALPHS signal is active. The following Transmit ALP Command signals are defined based on the width of the transmit data path: • 16-bit width – TxSendALPHS[0] • 32-bit width – TxSendALPHS[1:0] The following Transmit ALP Command signals cause an ALP Code to be transmitted as follows: • TxSendALPHS[0] – transmit an ALP Code in place of the C-PHY mapped version of TxDataHS[15:0] • TxSendALPHS[1] – transmit an ALP Code in place of the C-PHY mapped version of TxDataHS[31:16]

Symbol	Dir	Categories	Description
TxALPCodeHS0[3:0], TxALPCodeHS1[3:0]	I	MXXX SRXX	High-Speed Transmit ALP Code. When TxSendALPHS[0] is asserted then TxALPCodeHS0[3:0] selects which ALP Code is transmitted. Similarly, When TxSendALPHS[1] is asserted then TxALPCodeHS1[3:0] selects which ALP Code is transmitted. The mapping of the 4-bit TxALPCodeHS0[3:0] and TxALPCodeHS1[3:0] values to the transmitted ALP Codes is specified in ***Table*** 16.The following Transmit ALP Code signals are defined based on the width of the transmit data path: • 16-bit width – TxALPCodeHS0[3:0] • 32-bit width – TxALPCodeHS0[3:0] and TxALPCodeHS1[3:0]
TxALPNibble0[3:0], TxALPNibble1[3:0]	I	MXXX SRXX	High-Speed ALP Nibble Data. When TxSendALPHS[0] is asserted and TxALPCodeHS0[3:0] = 0b1010 to select the LPDT Nibble Code then TxALPNibble0[3:0] specifies the values of the first and last symbols of the LPDT Nibble Code. Similarly, when TxSendALPHS[1] is asserted and TxALPCodeHS1[3:0] = 0b1010 to select the LPDT Nibble Code then TxALPNibble1[3:0] specifies the values of the first and last symbols of the LPDT Nibble Code. The mapping of the 4-bit Nibble Code to the S_1 and S_0 symbols is specified in ***Table 16*** and ***Section 6.4.5.4***. The following ALP Nibble Code signals are defined based on the width of the transmit data path: • 16-bit width – TxALPNibble0[3:0] • 32-bit width – TxALPNibble0[3:0] and TxALPNibble1[3:0]
High-Speed Receive Signals			
RxWordClkHS	O	MRXX SXXX	High-Speed Receive Word Clock. This is used to synchronize signals in the High-Speed receive clock domain. The RxWordClkHS is generated by dividing the recovered High-Speed clock. The frequency of RxWordClkHS is dependent upon the width of the High-Speed Receive Data, as follows: • 16-bit width, RxDataHS[15:0], the High-Speed Receive Word Clock is exactly 1/7 the High-Speed received symbol rate. • 32-bit width, RxDataHS[31:0], the High-Speed Receive Word Clock is exactly 1/14 the High-Speed received symbol rate.

Symbol	Dir	Categories	Description
RxDataWidthHS[1:0]	I	MRXX SXXX	High-Speed Receive Data Width Select. Selects the bus width of RxDataHS: • RxDataWidthHS[1:0] = 00: not used, reserved. • RxDataWidthHS[1:0] = 01: 16-bit, RxDataHS[15:0] • RxDataWidthHS[1:0] = 10: 32-bit, RxDataHS[31:0] • RxDataWidthHS[1:0] = 11: not used, reserved. An implementation may support any data width - one fixed width, or subset of widths or all widths defined above.
RxDataHS[15:0] , or RxDataHS[31:0]	O	MRXX SXXX	High-Speed Receive Data. High-Speed data received by the Lane Module. The signal connected to RxDataHS[0] is associated with bit 0 of the C-PHY De-Mapping function. If a 32-bit receive data path is used then RxDataHS[16] is also associated with bit 0 of the C-PHY Mapping function, and if RxValidHS[1:0] = 11, then the seven symbols associated with RxDataHS[31:16] were received after the seven symbols associated with RxDataHS[15:0]. Data is transferred on rising edges of RxWordClkHS. The following signals are defined for the High-Speed Receive Data based on the width of the receive data path: • 16-bit width – RxDataHS[15:0] • 32-bit width – RxDataHS[31:0] An implementation may support any data width - one fixed width, or subset of widths or all widths defined above.
RxInvalidCodeHS[0], or RxInvalidCodeHS[1:0]	O	MRXX SXXX	High-Speed Invalid Code Word Detection. A High-Speed status signal that indicates the present Receive High-Speed Data was produced by a group of seven symbols that were not a valid code word. This is an indication that the data word being output has a low confidence of being correct. RxInvalidCodeHS signals are not active when either the Sync Word, the Post Pattern, or any of the ALP Codes are presented to the De-Mapper. The following High-Speed Invalid Code Word Detection signals are defined based on the width of the receive data path: • 16-bit width – RxInvalidCodeHS[0] • 32-bit width – RxInvalidCodeHS[1:0] The following High-Speed Invalid Code Word signals indicate invalid code words were used when DeMapping the received data as follows: • RxInvalidCodeHS[0] – invalid code word received and used for DeMapping of RxDataHS[15:0] • RxInvalidCodeHS[1] – invalid code word received and used for DeMapping of RxDataHS[31:16]

Symbol	Dir	Categories	Description
RxValidHS[0], or RxValidHS[1:0]	O	MRXX SXXX	High-Speed Receive Data Valid. This active high signal indicates that the Lane Module is driving data to the protocol layer on the RxDataHS output. There is no “RxReadyHS” signal, and the protocol layer is expected to capture RxDataHS on every rising edge of RxWordClkHS where any RxValidHS bit is asserted. There is no provision for the protocol layer to slow down (“throttle”) the receive data. The following High-Speed Receive Data Valid signals are defined based on the width of the receive data path: • 16-bit width – RxValidHS[0] • 32-bit width – RxValidHS[1:0] The following High-Speed Receive Data Valid signals indicate which bits of the RxDataHS data bus contain valid data as follows: • RxValidHS[0] – RxDataHS[15:0] contains valid data that was received from the channel • RxValidHS[1] – RxDataHS[31:16] contains valid data that was received from the channel
RxActiveHS	O	MRXX SXXX	High-Speed Reception Active. This active high signal indicates that the Lane Module is actively receiving a High-Speed transmission from the Lane Interconnect.
RxSyncHS[0], or RxSyncHS[1:0]	O	MRXX SXXX	Receiver Synchronization Observed. These active high signals indicate that the Lane Module has detected the 7-symbol sync word in the received data. In a typical High-Speed transmission, RxSyncHS[0] is high for one cycle of RxWordClkHS at the beginning of a High-Speed transmission when RxActiveHS is first asserted, and appropriate RxSyncHS signals are asserted prior to redundant packet headers that may appear in the data burst. The following Receiver Synchronization Observed signals are defined based on the width of the receive data path: • 16-bit width – RxSyncHS[0] • 32-bit width – RxSyncHS[1:0] The following Receiver Synchronization Observed signals indicate where a Sync Word was received as follows: • RxSyncHS[0] – a Sync Word was received and RxDataHS[15:0] does not contain valid received data • RxSyncHS[1] – a Sync Word was received and RxDataHS[31:16] does not contain valid received data

Symbol	Dir	Categories	Description
RxSyncTypeHS0[2:0], and RxSyncTypeHS1[2:0]	O	MRXX SXXX	High-Speed signal indicates the Sync Type. When RxSyncHS[0] is asserted then RxSyncTypeHS0[2:0] indicates which type of Sync Word was received. Similarly, When RxSyncHS[1] is asserted then RxSyncTypeHS1[2:0] indicates which type of Sync Word was received. The mapping of Sync Type to the actual Sync Word value is specified in ***Section 6.4.4.2***. The default Sync Word, 3444443, is Sync Type 3 indicated by the binary value 011. The following Sync Type Indicator signals are defined based on the width of the receive data path: • 16-bit width – RxSyncTypeHS0[2:0] • 32-bit width – RxSyncTypeHS0[2:0] and RxSyncTypeHS1[2:0]
High-Speed ALP Receive Signals			
RxALPValidHS[0], or RxALPValidHS[1:0]	O	MRXX SXXX	Receiver ALP Code Observed. These active high signals indicate that the Lane Module has detected the 7-symbol ALP Code in the received data. In a typical High-Speed transmission, RxALPValidHS[0] is high for one cycle of RxWordClkHS whenever a valid ALP Code is received in a High-Speed burst following the normal preamble or following the calibration preamble. The following Receiver ALP Code Observed signals are defined based on the width of the receive data path: • 16-bit width – RxALPValidHS[0] • 32-bit width – RxALPValidHS[1:0] The following Receiver ALP Code Observed signals indicate where an ALP Code was received as follows: • RxALPValidHS[0] – an ALP Code was received and RxDataHS[15:0] does not contain valid received data • RxALPValidHS[1] – an ALP Code was received and RxDataHS[31:16] does not contain valid received data
RxALPCode0[3:0], RxALPCode1[3:0]	O	MRXX SXXX	Receiver ALP Code Value. When RxALPValidHS[0] is asserted then RxALPCode0[3:0] indicates which ALP Code was received. Similarly, When RxALPValidHS[1] is asserted then RxALPCode1[3:0] indicates which ALP Code was received. The mapping of the 4-bit ALP Code value to the actual ALP Code is defined in ***Table 16*** with additional descriptions in ***Section 6.4.5***. The following Receiver ALP Code Value signals are defined based on the width of the receive data path: • 16-bit width – RxALPCode0[3:0] • 32-bit width – RxALPCode0[3:0] and RxALPCode1[3:0]

Symbol	Dir	Categories	Description
RxALPNibble0[3:0], RxALPNibble1[3:0]	O	MRXX SXXX	Receiver ALP LPDT Nibble Data. When RxALPValidHS[0] is asserted and RxALPCode0[3:0] is equal to 0b1010 (which corresponds to the LPDT Nibble Code) then RxALPNibble0[3:0] contains four bits of valid LPDT data. Similarly, When RxALPValidHS[1] is asserted and RxALPCode1[3:0] is equal to 0b1010 (which corresponds to the LPDT Nibble Code) then RxALPNibble1[3:0] contains four bits of valid LPDT data. The definition of the format of the LPDT nibble data within the ALP code is defined in ***Table 16*** and ***Section 6.4.5.4***. The following Receiver ALP LPDT Nibble Data signals are defined based on the width of the receive data path: • 16-bit width – RxALPNibble0[3:0] • 32-bit width – RxALPNibble0[3:0] and RxALPNibble1[3:0]
Escape Mode Transmit Signals			
TxClkEsc	I	MXXX SXXY	Escape Mode Transmit Clock. This clock is directly used to generate escape sequences. The period of this clock determines the phase times for low-power signals as defined in ***Section 6.6.2***. It is therefore constrained by the normative part of the C-PHY specification. See ***Section 9***. Note that this clock is used to synchronize TurnRequest and is included for any module that supports Bi-directional High-Speed operation, even if that module does not support transmit or Bi-directional Escape Mode.
TxRequestEsc	I	MXXX SXXY	Escape Mode Transmit Request. This active high signal, asserted together with exactly one of TxLpdtEsc, TxUlpsEsc, or one bit of TxTriggerEsc, is used to request entry into Escape Mode. Once in Escape Mode, the Lane stays in Escape Mode until TxRequestEsc is de-asserted. TxRequestEsc is only asserted by the protocol layer while TxRequestHS is low.
TxLpdtEsc	I	MXAX SXXA	Escape Mode Transmit Low-Power Data. This active high signal is asserted with TxRequestEsc to cause the Lane Module to enter low-power data transmission mode. The Lane Module remains in this mode until TxRequestEsc is de-asserted. TxUlpsEsc and all bits of TxTriggerEsc are low when TxLpdtEsc is asserted.

Symbol	Dir	Categories	Description
TxUlpsExit	I	MXXX SXXY	Transmit ULP Exit Sequence. This active high signal is asserted when ULP state is active and the protocol layer is ready to leave ULP state. The PHY leaves ULP state and begins driving Mark-1 after TxUlpsExit is asserted. The PHY later drives the Stop state (LP-111) when TxRequestEsc is de-asserted. TxUlpsExit is synchronous to TxClkEsc. This signal is ignored when the Lane is not in the ULP State.
TxUlpsEsc	I	MXXX SXXY	Escape Mode Transmit Ultra-Low Power State. This active high signal is asserted with TxRequestEsc to cause the Lane Module to enter the Ultra-Low Power State. The Lane Module remains in this mode until TxRequestEsc is de-asserted. TxLpdtEsc and all bits of TxTriggerEsc are low when TxUlpsEsc is asserted.
TxTriggerEsc[3:0]	I	MXXX SXXY	Escape Mode Transmit Trigger 0-3. One of these active high signals is asserted with TxRequestEsc to cause the associated Trigger to be sent across the Lane Interconnect. In the receiving Lane Module, the same bit of RxTriggerEsc is then asserted and remains asserted until the Lane Interconnect returns to Stop state, which happens when TxRequestEsc is de-asserted at the transmitter. Only one bit of TxTriggerEsc is asserted at any time, and only when TxLpdtEsc and TxUlpsEsc are both low. TxTriggerEsc[0] corresponds to Reset-Trigger. TxTriggerEsc[1] corresponds to Unknown-3 Trigger. TxTriggerEsc[2] corresponds to Unknown-4 Trigger. TxTriggerEsc[3] corresponds to Unknown-5 Trigger.
TxDataEsc[7:0]	I	MXAX SXXA	Escape Mode Transmit Data. This is the eight bit Escape Mode data to be transmitted in low-power data transmission mode. The signal connected to TxDataEsc[0] is transmitted first. Data is captured on rising edges of TxClkEsc.
TxValidEsc	I	MXAX SXXA	Escape Mode Transmit Data Valid. This active high signal indicates that the protocol layer is driving valid data on TxDataEsc to be transmitted. The Lane Module accepts the data when TxRequestEsc, TxValidEsc and TxReadyEsc are all active on the same rising TxClkEsc clock edge.
TxReadyEsc	O	MXAX SXXA	Escape Mode Transmit Ready. This active high signal indicates that TxDataEsc is accepted by the Lane Module to be serially transmitted. TxReadyEsc is valid on rising edges of TxClkEsc.

Symbol	Dir	Categories	Description
Escape Mode Receive Signals			
RxClkEsc	O	MXXY SXXX	Escape Mode Receive Clock. This signal is used to transfer received data to the protocol layer during Escape Mode. This “clock” is generated from the two low-power signals in the Lane Interconnect. Because of the asynchronous nature of Escape Mode data transmission, this “clock” may not be periodic.
RxLpdtEsc	O	MXXA SXAX	Escape Low-Power Data Receive mode. This active high signal is asserted to indicate that the Lane Module is in low-power data receive mode. While in this mode, received data bytes are driven onto the RxDataEsc output when RxValidEsc is active. The Lane Module remains in this mode with RxLpdtEsc asserted until a Stop state is detected on the Lane Interconnect.
RxUlpsEsc	O	MXXY SXXX	Escape Ultra-Low Power (Receive) mode. This active high signal is asserted to indicate that the Lane Module has entered the Ultra-Low Power State. The Lane Module remains in this mode with RxUlpsEsc asserted until a Stop state is detected on the Lane Interconnect.
RxTriggerEsc[3:0]	O	MXXY SXXX	Escape Mode Receive Trigger 0-3. These active high signals indicate that a trigger event has been received. The asserted RxTriggerEsc signal remains active until a Stop state is detected on the Lane Interconnect. RxTriggerEsc[0] corresponds to Reset-Trigger. RxTriggerEsc[1] corresponds to Unknown-3 Trigger. RxTriggerEsc[2] corresponds to Unknown-4 Trigger. RxTriggerEsc[3] corresponds to Unknown-5 Trigger.
RxDataEsc[7:0]	O	MXXA SXAX	Escape Mode Receive Data. This is the eight-bit Escape Mode low-power data received by the Lane Module. The signal connected to RxDataEsc[0] was received first. Data is transferred on rising edges of RxClkEsc.
RxValidEsc	O	MXXA SXAX	Escape Mode Receive Data Valid. This active high signal indicates that the Lane Module is driving valid data to the protocol layer on the RxDataEsc output. There is no “RxReadyEsc” signal, and the protocol layer is expected to capture RxDataEsc on every rising edge of RxClkEsc where RxValidEsc is asserted. There is no provision for the protocol layer to slow down (“throttle”) the receive data.

Symbol	Dir	Categories	Description
Control Signals			
TurnRequest	I	XRXX XFXY	Turn Around Request. This active high signal is used to indicate that the protocol layer desires to turn the Lane around, allowing the other side to begin transmission. TurnRequest is valid on rising edges of TxClkEsc. TurnRequest is only meaningful for a Lane Module that is currently the transmitter (Direction=0). If the Lane Module is in receive mode (Direction=1), this signal is ignored.
Direction	O	XRXX XFXY	Transmit/Receive Direction. This signal is used to indicate the current direction of the Lane Interconnect. When Direction=0, the Lane is in transmit mode (0=Output). When Direction=1, the Lane is in receive mode (1=Input).
TurnDisable	I	XRXX XFXY	Disable Turn-around. This signal is used to prevent a (Bi-directional) Lane from going into transmit mode – even if it observes a turn-around request on the Lane Interconnect. This is useful to prevent a potential “lock-up” situation when a Unidirectional Lane Module is connected to a Bi-directional Lane Module.
ForceRxmode	I	MRXX MXXY SXXX	Force Lane Module Into Receive mode / Wait for Stop state. This signal allows the protocol layer to initialize a Lane Module, or force a Bi-Directional Lane Module, into receive mode. This signal is used during initialization or to resolve a contention situation. When this signal is high, the Lane Module immediately transitions into receive control mode and waits for a Stop state to appear on the Lane Interconnect. When used for initialization, this signal is released, i.e. driven low, only when the Dp & Dn inputs are in Stop state for time t_{INIT}, or longer.
ForceTxStopmode	I	MXXX SRXX SXXY	Force Lane Module Into Transmit mode / Generate Stop state. This signal allows the protocol layer to force a Lane Module into transmit mode and Stop state during initialization or following an error situation, e.g. expired time out. When this signal is high, the Lane Module immediately transitions into transmit mode and the module state machine is forced into the Stop state.
Stopstate	O	XXXX	Lane is in Stop state. This active high signal indicates that the Lane Module, regardless of whether the Lane Module is a transmitter or a receiver, is currently in Stop state. Note that this signal is asynchronous to any clock in the PPI interface. Also, the protocol layer may use this signal to indirectly determine if the PHY line levels are in the LP-111 state.

Symbol	Dir	Categories	Description
Enable	I	XXXX	Enable Lane Module. This active high signal forces the Lane Module out of “shutdown”. All line drivers, receivers, terminators, and contention detectors are turned off when Enable is low. Furthermore, while Enable is low, all other PPI inputs are ignored and all PPI outputs are driven to the default inactive state. Enable is a level sensitive signal and does not depend on any clock.
UlpsActiveNot	O	XXXX	ULP State (not) Active. This active low signal is asserted to indicate that the Lane is in ULP state. For a transmitter, this signal is asserted sometime after TxUlpsEsc and TxRequestEsc are asserted. The transmitting PHY continues to supply TxClkEsc until UlpsActiveNot is asserted. In order to leave ULP state, the transmitter first drives TxUlpsExit high, then waits for UlpsActiveNot to become high (inactive). At that point, the transmitting PHY is active and has started transmitting a Mark-1 on the lines. The protocol layer waits for time t_{wakeup} and then drives TxRequestEsc inactive to return the Lane to Stop state. For a receiver, this signal indicates that the Lane is in ULP state. At the beginning of ULP state, UlpsActiveNot is asserted together with RxUlpsEsc. At the end of the ULP state, this signal becomes inactive to indicate that the Mark-1 state has been observed. Later, after a period of time t_{wakeup}, the RxUlpsEsc signal is de-asserted.
ALPWakeState[2:0]	I	MXXX SRXX	ALP-Pause Wake Wire State A group of control signals that select the wire state to be driven on the Lane during the ALP-Pause Wake state. The default is to drive the “+x” state but it may be necessary to drive one of the other five possible states to resolve wire permutations in the Lane Interconnect, in case A, B and C at the Master are not connected to A. B and C, respectively, at the Slave. The binary values on ALPWakeState[2:0] choose the wire state driven during ALP-Pause Wake as follows: • 000 – “+x” driven during ALP-Pause Wake (default) • 001 – “-x” driven during ALP-Pause Wake • 010 – “+y” driven during ALP-Pause Wake • 011 – “-y” driven during ALP-Pause Wake • 100 – “+z” driven during ALP-Pause Wake • 101 – “-z” driven during ALP-Pause Wake

Symbol	Dir	Categories	Description
Error Signals			
ErrSotHS	O	MRXX SXXX	Start-of-Transmission (SoT) Error. If the High-Speed SoT Sync Word is corrupted, by having the least-significant symbol of the 3444443 pattern not equal to 3, this active high signal is asserted for one cycle of RxWordClkHS. This is considered to be a “soft error” in the Sync Word and confidence in the payload data is reduced. This error applies only to the reception of the first Sync Word in a burst that follows the normal preamble or calibration preamble. This first Sync Word is transmitted by the C-PHY and is always intended to be the symbol sequence: 3444443.
ErrSotSyncHS	O	MRXX SXXX	Start-of-Transmission Synchronization Error. If the High-Speed SoT Sync Word is corrupted in a way that proper synchronization cannot be expected, this active high signal is asserted for one cycle of RxWordClkHS. (Considered optional, this is an extremely low-probability event.)
ErrEsc	O	MXXY SXXX	Escape Entry Error. If an unrecognized escape entry command is received, this active high signal is asserted and remains asserted until the next change in line state.
ErrSyncEsc	O	MXXA SXAX	Low-Power Data Transmission Synchronization Error. If the number of bits received during a low-power data transmission is not a multiple of eight when the transmission ends, this active high signal is asserted and remains asserted until the next change in Line state.
ErrControl	O	MXXY SXXX	Control Error. This active high signal is asserted when an incorrect line state sequence is detected. For example, if a Turn-around request or Escape Mode request is immediately followed by a Stop state instead of the required Bridge state, this signal is asserted and remains asserted until the next change in Line state.
ErrContentionLP0	O	MXXX SXXY	LP0 Contention Error. This active high signal is asserted when the Lane Module detects a contention situation on a Line while trying to drive the Line low.
ErrContentionLP1	O	MXXX SXXY	LP1 Contention Error. This active high signal is asserted when the Lane Module detects a contention situation on a Line while trying to drive the Line high.

Note:

1. *Only one of TxSendSyncHS[0] or TxSendALPHS[0] may be asserted at a time. Similarly, regarding the control signals for the upper word when a 32-bit data path is used, only one of TxSendSyncHS[1] or TxSendALPHS[1] may be asserted at a time.*

2219 ***Table 62*** summarizes the signals that are affected by the choice of the transmit data path width.

Table 62 Tx HS PPI Signals, Impact of Data Path Width

	16-bit	32-bit
Tx HS Word Clock Rate	1/7 the HS Symbol Rate	1/14 the HS Symbol Rate
Tx HS Data Path	TxDataHS[15:0]	TxDataHS[31:0]
HS Transmit Word Valid	TxWordValidHS[0] ⇒TxDataHS[15:0]	TxWordValidHS[0] ⇒TxDataHS[15:0]; TxWordValidHS[1] ⇒TxDataHS[31:15]
Transmit Sync Word	TxSendSyncHS[0] ⇒TxDataHS[15:0]	TxSendSyncHS[0] ⇒TxDataHS[15:0]; TxSendSyncHS[1] ⇒TxDataHS[31:15]

Table 63 summarizes the signals that are affected by the choice of the transmit data path width.

Table 63 Rx HS PPI Signals, Impact of Data Path Width

	16-bit	32-bit
Rx HS Word Clock Rate	1/7 the HS Symbol Rate	1/14 the HS Symbol Rate
Rx HS Data Path	RxDataHS[15:0]	RxDataHS[31:0]
HS Receive Word Valid	RxValidHS[0] ⇒RxDataHS[15:0]	RxValidHS[0] ⇒RxDataHS[15:0]; RxValidHS[1] ⇒RxDataHS[31:15]
Receiver Sync Word Observed	RxSyncHS[0] ⇒RxDataHS[15:0]	RxSyncHS[0] ⇒RxDataHS[15:0]; RxSyncHS[1] ⇒RxDataHS[31:15]
Rx HS Invalid Code Word Detection	RxInvalidCodeHS[0] ⇒RxDataHS[15:0]	RxInvalidCodeHS[0] ⇒RxDataHS[15:0]; RxInvalidCodeHS[1] ⇒RxDataHS[31:15]

A.2 High-Speed Transmission from the Master or Slave

Figure 93 shows an example of a High-Speed transmission from the Master or Slave. While TxRequestHS is low, the Lane Module ignores the value of TxDataHS. To begin transmission, the protocol layer drives TxDataHS with the first word of data and asserts TxRequestHS. This data word is accepted by the PHY on the first rising edge of TxWordClkHS with TxReadyHS also asserted. At this point, the protocol logic drives the next data word onto TxDataHS. After every rising clock cycle with TxReadyHS active, the protocol layer supplies a new valid data word or ends the transmission. After the last data word has been transferred to the Lane Module, TxRequestHS is driven low to cause the Lane Module to stop the transmission and enter the Stop state. The minimum number of words transmitted could be as small as one.

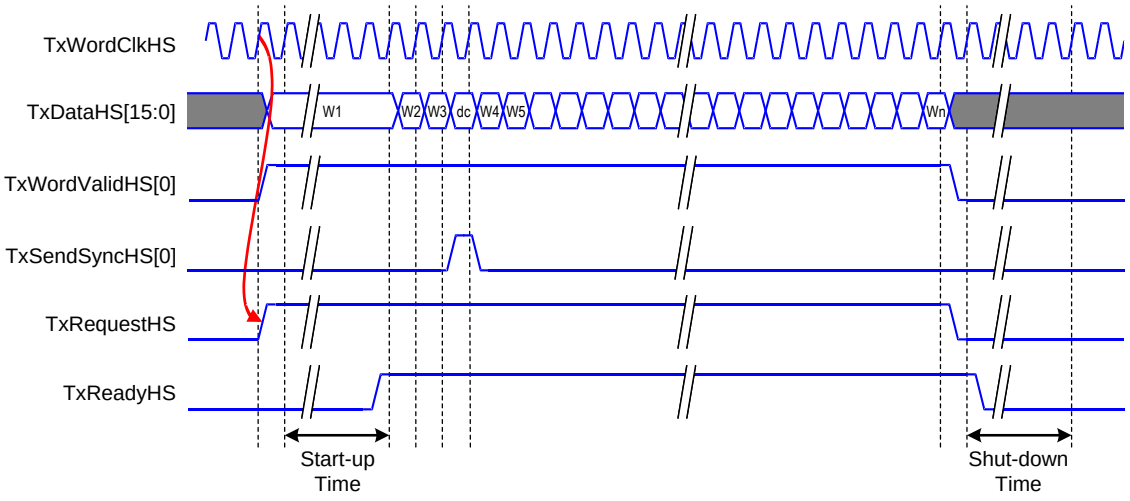


Figure 93 Example High-Speed Transmission from the Master or Slave, 16-bit PPI

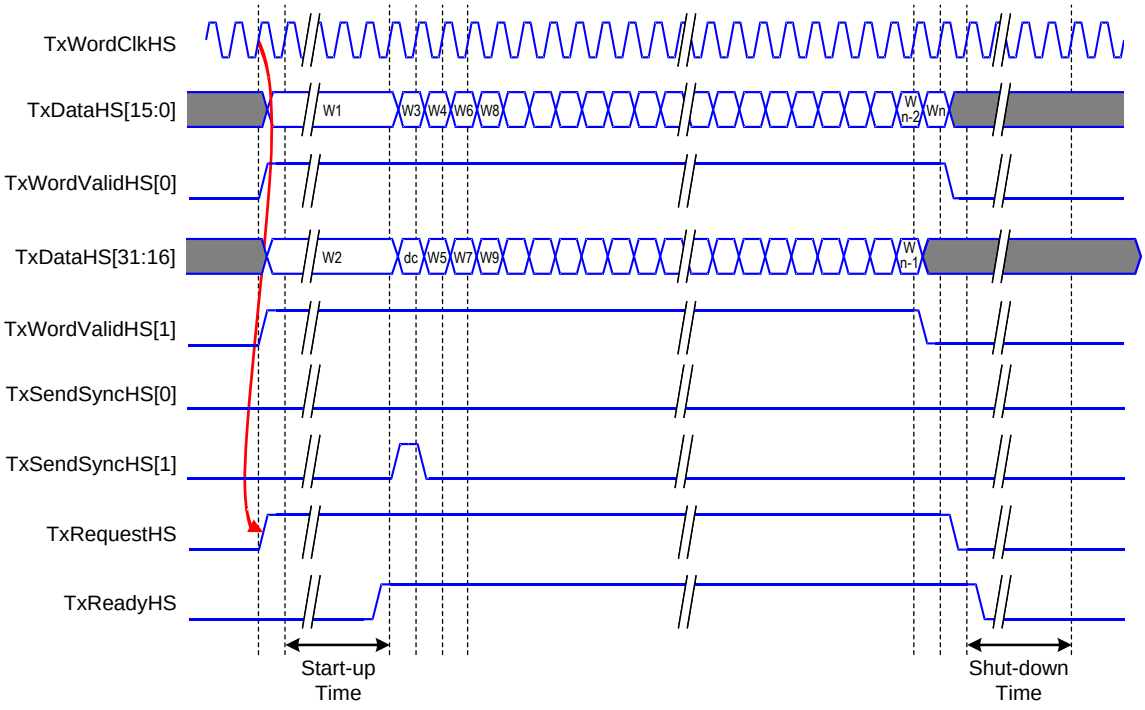


Figure 94 Example High-Speed Transmission from the Master or Slave, 32-bit PPI

A.3 High-Speed Reception at the Slave or Master

Figure 95 shows an example of a High-Speed reception at the Slave or Master. The RxActiveHS signal indicates that High-Speed data reception is occurring. Normal High-Speed data reception starts with a pulse on RxSyncHS followed by valid received data on subsequent cycles of RxWordClkHS. Note that the protocol layer is prepared to receive all of the data. There is no method for the receiving protocol layer to pause or slow data reception.

If EoT Processing is performed inside the PHY, the RxActiveHS and RxValidHS signals transition low following the last valid data word, Wn. Refer to ***Figure 95***.

If EoT processing is not performed in the PHY, one or more additional words are presented after the last valid data word. The first of these additional words, shown as word “C” in ***Figure 95***, is all ones or all zeros. Subsequent words may or may not be present, and can have any value. For a PHY that does not perform EoT processing, the RxActiveHS and RxValidHS signals transition low simultaneously sometime after word “C” is received. Once these signals have transitioned low, they remain low until the next High-Speed data reception begins.

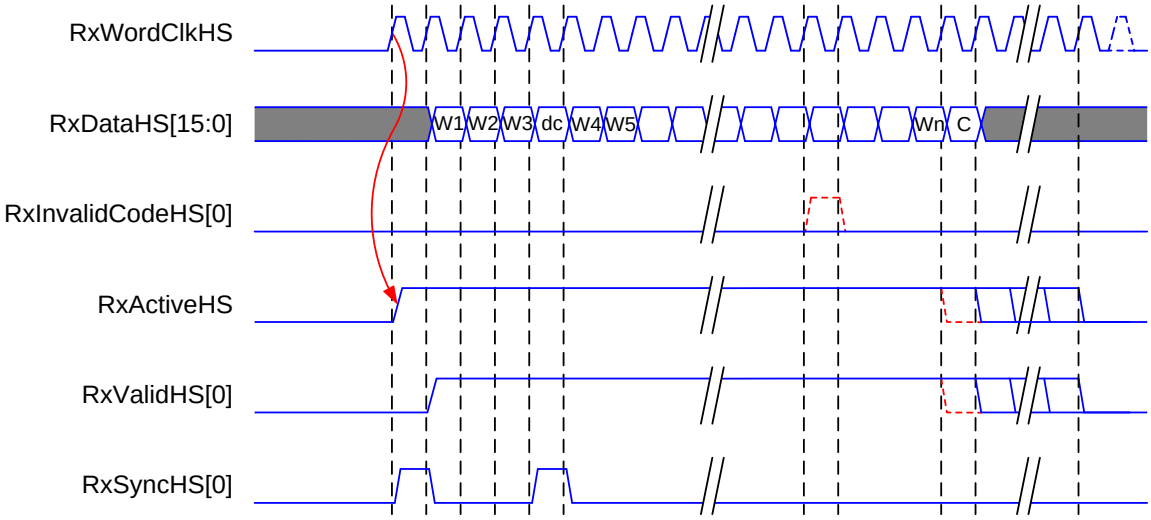


Figure 95 Example High-Speed Reception at the Slave or Master, 16-bit PPI

For D-PHY PPI compatibility, it is recommended that RxWordClkHS toggle continuously while the High-Speed signal is present. An option shown in the waveform above is to prevent RxWordClk from toggling while RxActiveHS is inactive.

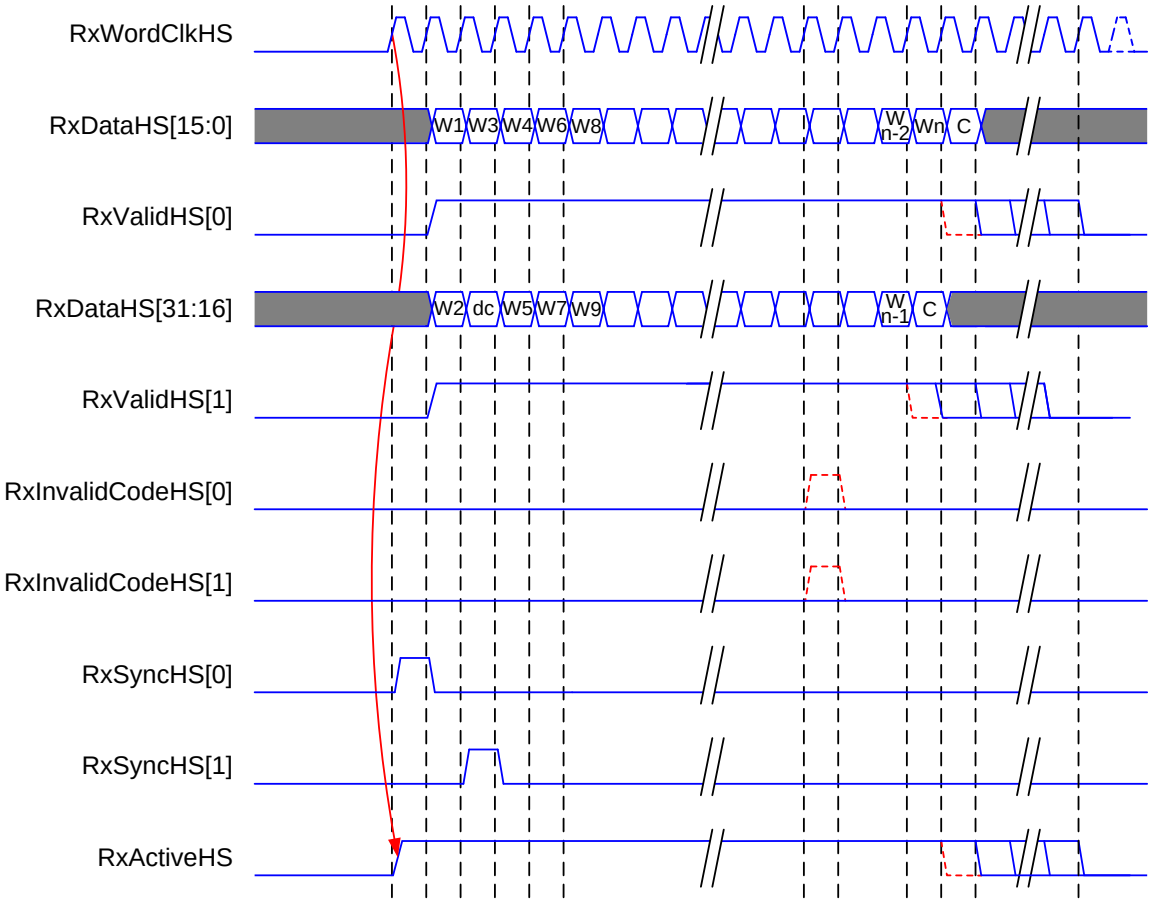


Figure 96 Example High-Speed Reception at the Slave or Master, 32-bit PPI

A.4 High-Speed Transmission from the Slave

High-Speed transmission from the Slave occurs during a High-Speed reverse data transfer. The transmission speed at the Slave (i.e., in the Reverse Direction) is not required to have any specific relationship to the speed of the Forward Direction Link transmission. Timing is embedded within the C-PHY 3-Phase signal (and not sourced from another place within the PHY), so the PPI timing and signal characteristics for the Slave transmitter are identical to ***Figure 93*** or ***Figure 94*** which describe transmission from the Master.

A.5 High-Speed Reception at the Master

High-Speed reception at the Master occurs during a High-Speed reverse data transfer. The reception speed at the Master (i.e., in the Reverse Direction) is not required to have any specific relationship to the speed of the Forward Direction Link transmission. Timing is embedded within the C-PHY 3-Phase signal (and not sourced from another place within the PHY), so the PPI timing and signal characteristics for the Master receiver are identical to ***Figure 95*** or ***Figure 96*** which describe reception at the Slave.

A.6 Low-Power Data Transmission

Furthermore, while the High-Speed interface signal TxRequestHS serves as both a transmit request and a data valid signal, on the low-power interface two separate signals are used. The protocol layer directs the Lane to enter low-power data transmission Escape Mode by asserting TxRequestEsc with TxLpdtEsc high. The low-power transmit data is transferred on the TxDataEsc lines when TxValidEsc and TxReadyEsc are both active at a rising edge of TxClkEsc. The byte is transmitted in the time after the TxDataEsc is accepted

by the Lane Module (TxValidEsc = TxReadyEsc = high) and therefore the TxClkEsc continues running for some minimum time after the last byte is transmitted. The protocol layer knows the byte transmission is finished when TxReadyEsc is asserted. After the last byte has been transmitted, the protocol layer de-asserts TxRequestEsc to end the low-power data transmission. This causes TxReadyEsc to return low, after which the TxClkEsc clock is no longer needed. Whenever TxRequestEsc transitions from high-to-low, it always remains in the low state for a minimum of two TxClkEsc clock cycles. ***Figure 97*** shows an example low-power data transmission operation.

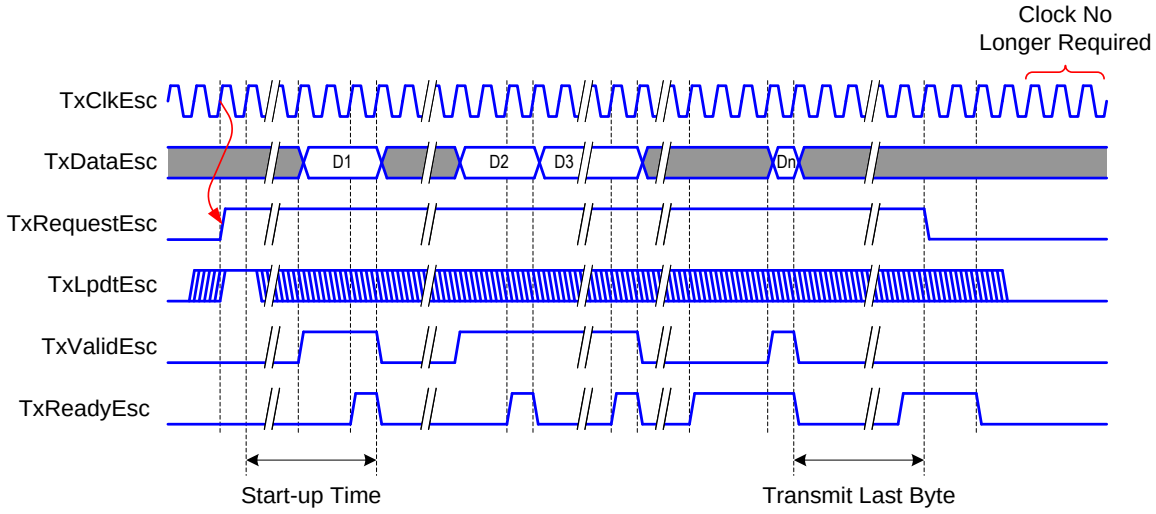


Figure 97 Low-Power Data Transmission

A.7 Low-Power Data Reception

Figure 98 shows an example low-power data reception. In this example, a low-power escape "clock" is generated from the Lane Interconnect by the logical exclusive-OR of the "A" and "C" signals. This "clock" is used within the Lane Module to capture the transmitted data. In this example, the "clock" is also used to generate RxClkEsc.

The signal RxLpdtEsc is asserted when the escape entry command is detected and stays high until the Lane returns to the Stop state, indicating that the transmission has finished. It is important to note that because of the asynchronous nature of Escape Mode transmission, the RxClkEsc signal can stop at any time in either the high or low state. This is most likely to happen just after a byte has been received, but it could happen at other times as well.

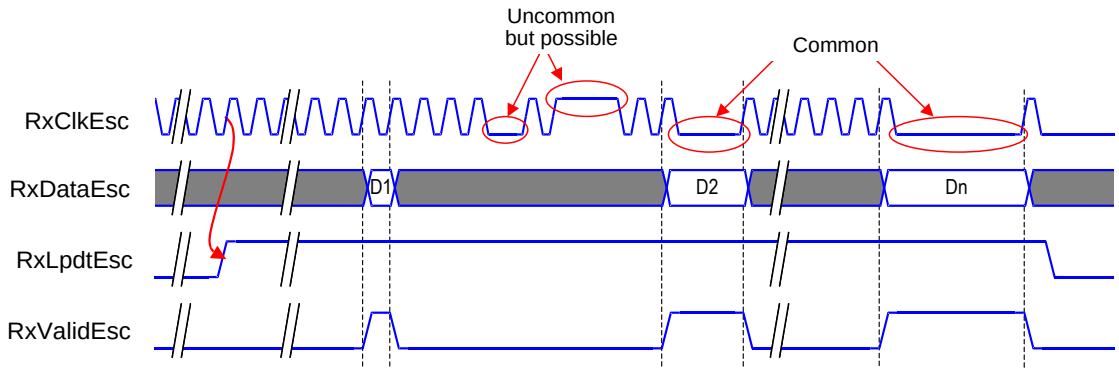


Figure 98 Example Low-Power Data Reception

A.8 Turn-Around

If the Master side and Slave side Lane Modules are both Bi-directional, it is possible to turn around the Link for High-Speed and/or Escape Mode signaling. ***Section 6.4.4.2*** explains how it is determined which side is allowed to transmit by passing a "token" back and forth. That is, the side currently transmitting passes the token to the receiving side. If the receiving side acknowledges the turn-around request, as indicated by driving the appropriate line state, the direction is switched.

Figure 99 shows an example of two turn-around events. At the beginning, the local side is the transmitter, as shown by Direction=0. When the protocol layer on this side wishes to turn the Lane around (i.e. give the token to the other side), it asserts TurnRequest for at least one cycle of TxClkEsc. This initiates the turn-around procedure. The remote side acknowledges the turn-around request by driving the appropriate states on the lines. When this happens, the local Direction signal changes from transmit (0) to receive (1).

Later in the example of ***Figure 62***, the remote side initiates a turn-around request, passing the token back to the local side. When this happens, the local Direction signal changes back to transmit (0). Note that there is no prescribed way for a receiver to request access to the Link. The current transmitter is in control of the Link direction and decides when to turn the Link around, passing control to the receiver.

If the remote side does not acknowledge the turn-around request, the Direction signal does not change.

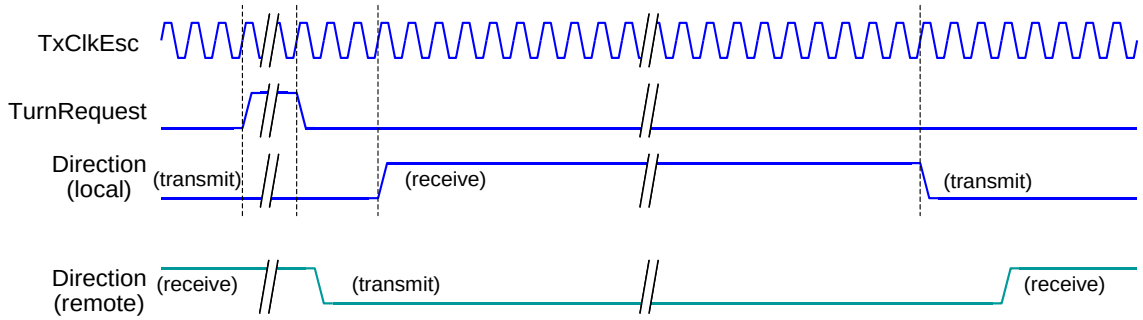


Figure 99 Example Turn-Around Actions Transmit-to-Receive and Back to Transmit

A.9 (Not Used)

Note:

This Section is null for the C-PHY Specification. The Section heading has been retained in order to synchronize Section numbering with the D-PHY Specification [MIPI01].

A.10 Optical Link Support

A.10.1 System Setup

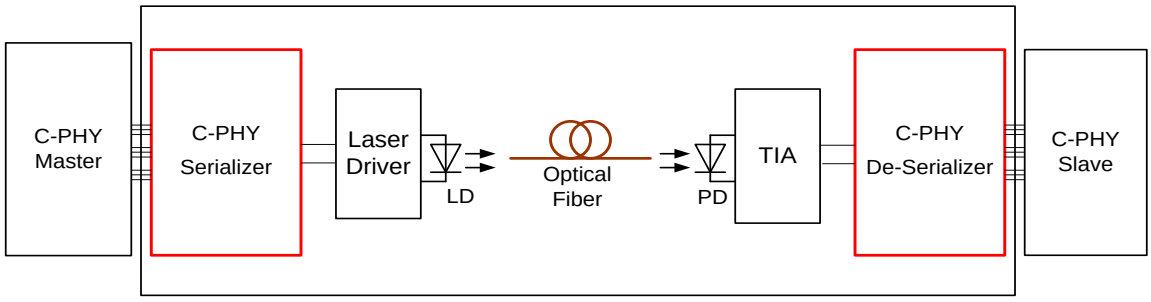


Figure 100 Typical System Setup with Optical Interconnect

Figure 100 shows a typical setup for a C-PHY system using an optical Link. The C-PHY Master provides the Master data Lanes, then the serializer multiplexes the data content of N data Lanes into a single bit stream with embedded clock. The HS data edges of one data Lane are used as a reference for the clock multiplying unit in the serializer. The single bit stream is then converted from an electrical to an optical signal by means of a laser driver and a laser diode (LD) connected to it.

The optical signal transmitted through the optical fiber is converted back to an electrical signal by means of a photo diode (PD) and a transimpedance amplifier (TIA). The de-serializer synchronizes to the clock embedded in the serial data stream and de-multiplexes the data content of N data Lanes. The output of the de-serializer is composed of a set of N C-PHY compliant data Lanes which replicates the C-PHY signal input to the serializer. This way the added optical Link provides a transparent interface between a C-PHY Master and a C-PHY Slave device.

A.10.2 Serializer and De-Serializer Block Diagrams

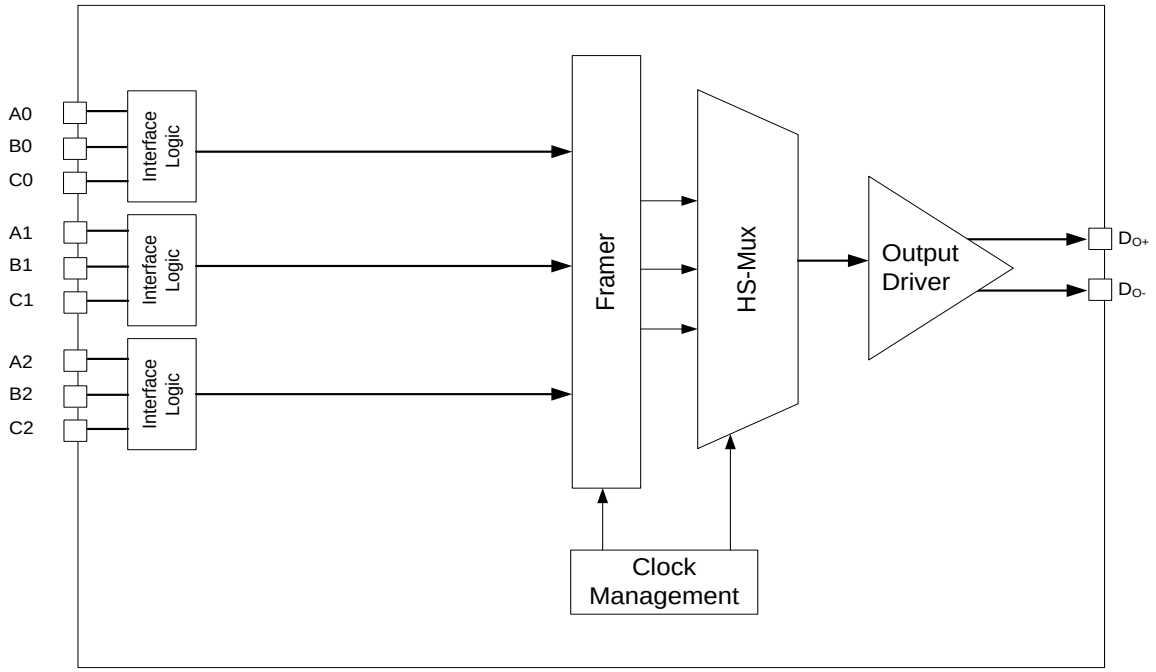


Figure 101 Block Diagram of Typical Serializer

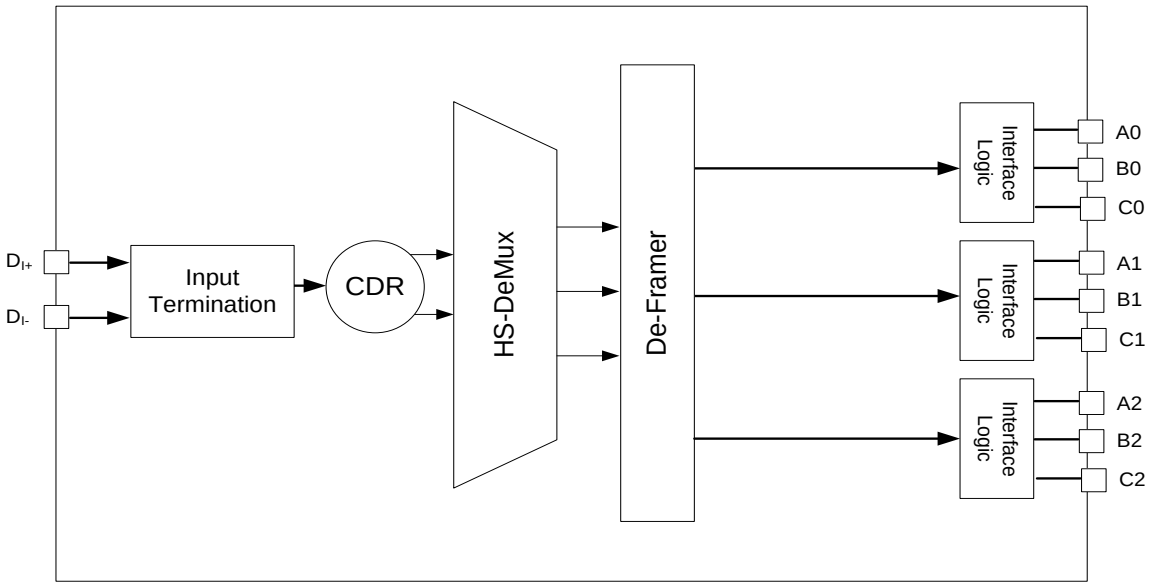


Figure 102 Block Diagram of Typical De-Serializer

Figure 101 and ***Figure 102*** show typical block diagrams for serializers and de-serializers used to implement the C-PHY optical Link.

A.10.3 Timing Constraints

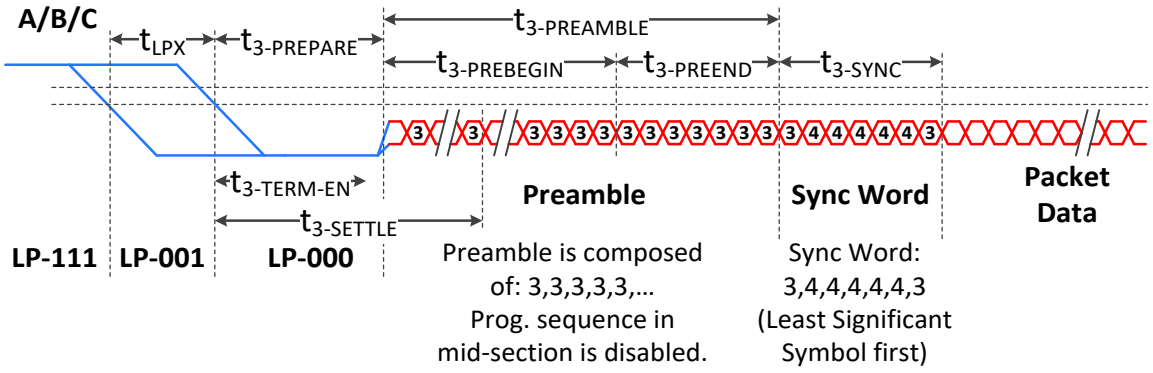


Figure 103 Delay Between Start of HS Data and HS Packet Data Transmission Without Optical Link

In a purely electrical C-PHY interconnect, the timing relationship between the start of HS data transmission and start of HS Packet Data transmission is a delay composed of the sum of $t_{3\text{-PREBEGIN}} + t_{3\text{-PREEND}} + t_{3\text{-SYNC}}$, as show in ***Figure 103*** (or ***Figure 23***, ***Section 6.4.4*** of [MIPI02]).

However, if an optical Link is added as shown in ***Figure 100***, the serializer clock multiplying unit (typically a PLL) and de-serializer clock and data recovery (CDR) need synchronization times that exceed the aforementioned timing delay.

In order to provide enough timing headroom for the optical Link to establish synchronization, an additional wait time $t_{WAIT\text{-}OPTICAL}$ shall be inserted before any HS Request is transmitted. This is explained in ***Figure*** 104.

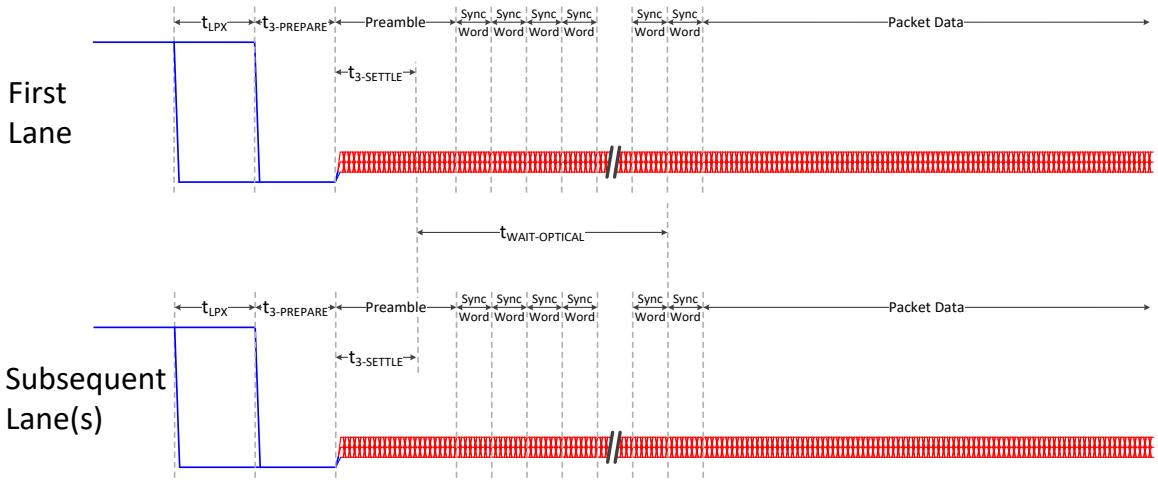


Figure 104 Delay of HS Data Transmission with Optical Link, All Lanes Transitioning into HS Mode Simultaneously

Figure 104 shows the additional wait time $t_{WAIT-OPTICAL}$ inserted between the end of $t_{3-SETTLE}$ of either a single Lane, or of multiple Lanes transitioning into HS mode simultaneously and the beginning of the sync word that precedes the packet data stream. The additional wait time $t_{WAIT-OPTICAL}$ ensures that the optical Link is fully synchronized by the time valid High-Speed data is sent on any data Lane.

If an insufficient $t_{WAIT-OPTICAL}$ is inserted, the optical Link will not be able to correctly transmit the upcoming HS data burst. The result is a loss of HS data.

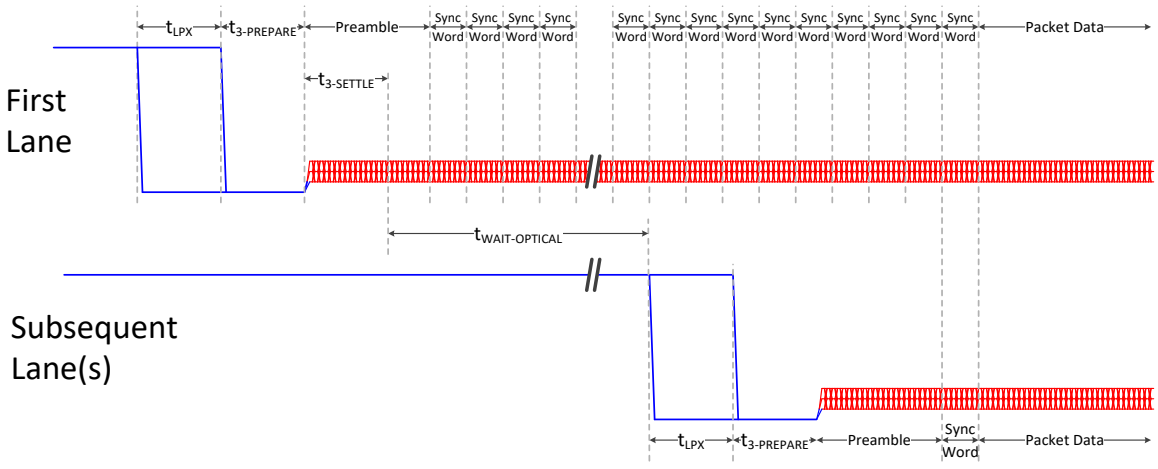


Figure 105 Delay of HS Data Transmission with Optical Link, Lanes Starting at Different Times

Figure 105 shows the additional wait time $t_{WAIT-OPTICAL}$ inserted between the end of $t_{3-SETTLE}$ of a first Lane transitioning into HS mode and the beginning of t_{LPX} of any other data Lane scheduled to switch from STOP state to HS data mode. The additional wait time $t_{WAIT-OPTICAL}$ ensures that the optical Link is fully synchronized by the time the next data Lane switches from STOP state to HS data mode.

If an insufficient $t_{WAIT-OPTICAL}$ is inserted, the optical Link will not be able to correctly transmit the beginning of the upcoming next HS data burst. The result is a loss of state information and loss of HS data.

In this use case, the first Lane continuously runs in High-Speed mode after the first transition into High-Speed mode. This provides a continuous clock reference for the optical Link.

A.10.4 System Constraints

A.10.4.1 Bus Turnaround

Due to the optical Link's inherently Unidirectional nature, bus Turnaround (BTA) may not be supported with an optical Link.

A.10.4.2 Equalization and Calibration

Equalization and calibration may be supported by the optical Link manufacturer. This must be stated in the corresponding datasheet of the optical Link. If these features are included in the optical Link, the electrical inputs of the optical Link shall follow the C-PHY specification for a C-PHY RX, and the electrical outputs of the optical Link shall follow the specification for a C-PHY TX for these features. System integrators must take care to ensure compliance during implementation.

A.10.4.3 $t_{WAIT-OPTICAL}$

Table 64 specifies $t_{WAIT-OPTICAL}$, the parameter for additional wait time for synchronization of the optical Link.

Table 64 Timing with Optical Link

Parameter	Description	Min	Units
$t_{WAIT-OPTICAL}$	Additional wait time for synchronization of the optical Link	150,000	UI (Lane data symbol time)

Annex B Interconnect Design Guidelines (Informative)

B.1 Practical Distances

The maximum Lane flight time is defined at two nanoseconds. Assuming less than 100ps wiring delay within the RX-TX modules each, the physical distance that can be bridged with external interconnect is around 54cm/√ε. For most practical PCB and flex materials this corresponds to maximum distances around 25-30 cm.

B.2 RF Frequency Bands: Interference

The most common concern is the case where emissions from the interface are in the same frequency band as the wireless signal, and the emissions act as a jammer that degrades reception of the intended signal at the wireless receiver. The path of this interference is illustrated in ***Figure 106***.

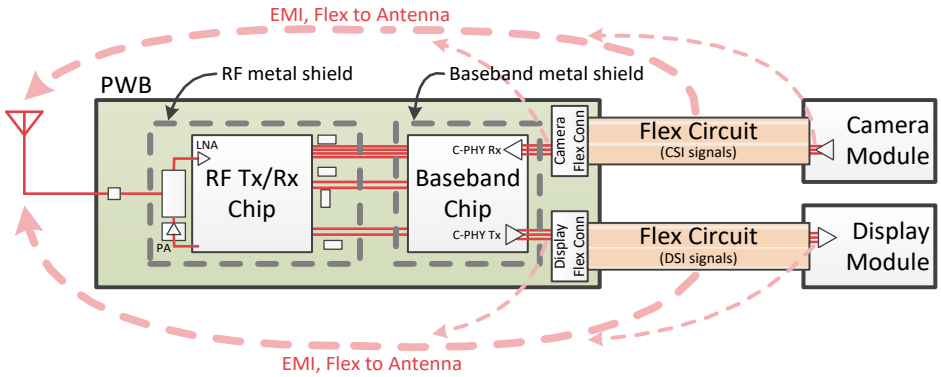


Figure 106 Radio Interference from Serial Interface Connections

The reverse of this path is also a possible concern, where a signal transmitted by the wireless transceiver is coupled to the serial interface signals in such a manner that it cannot be rejected by the common mode rejection capability of the serial interface receiver. It is also possible that the signal transmitted by the wireless transceiver is coupled in a manner that it produces an unintended differential mode signal at the serial interface receiver. Specific concerns are described below.

B.2.1 Specific Recommendations Regarding EMI and EMC

In terms of sensitivity:

- RX (downlink) is more of concern than TX (but TX might have to be considered for EMC)
- GNSS systems are almost always at sensitivity level (it is rare to get close to a satellite!) => they are the priority for EMI design
- In order of sensitivity: GNSS > Cellular > Connectivity
- Wide band systems (WCDMA/LTE/WiFi) are less sensitive to clock spurs than narrow band systems (GSM/Bluetooth/FM)

EMI design of MIPI interfaces

- Clock forwarded system (D-PHY) should avoid clock and its harmonics to fall in GNSS system bands then when feasible it should also avoid GSM bands (2, 3, 5, 8) and Bluetooth (although with frequency hopping Bluetooth is relatively robust), WCDMA, LTE and WiFi bands are less of a concern
- For others (C-PHY/M-PHY) when frequency is low enough it should try to have a null in GNSS bands

- Noise whitening techniques are essential for data Lanes to avoid energy to peak in undesired bands (note though that even for CSI and DSI that data is pretty random)

Implication for current specs

- Most bands being covered by first lobe of interface, slew control as very little benefit for interface > 1.5Gbps/GSs/Lane
- Above G1 M-PHY has no real need to keep fixed frequencies for gears

2391 **Table 65 Cellular Bands Used by Mobile Devices**

Band	Downlink MS RX (EMI)		Uplink MS TX (EMC)		Duplex	Common Name / Region
	F_{low}	F_{high}	F_{low}	F_{high}		
Band 1	2110.0	2170.0	1920.0	1980.0	FDD	IMT / EU-Asia
Band 2	1930.0	1990.0	1850.0	1910.0	FDD	PCS / US
Band 3	1805.0	1880.0	1710.0	1785.0	FDD	DCS /EU-Asia
Band 4	2110.0	2155.0	1710.0	1755.0	FDD	AWS-1 / US
Band 5	869.0	894.0	824.0	849.0	FDD	CLR / US
Band 6	875.0	885.0	830.0	840.0	FDD	UMTS 800 / Japan
Band 7	2620.0	2690.0	2500.0	2570.0	FDD	IMT / EU
Band 8	925.0	960.0	880.0	915.0	FDD	E-GSM / EU-Asia
Band 9	1844.9	1879.9	1749.9	1784.9	FDD	UMTS 1700 / Japan DCS
Band 10	2110.0	2170.0	1710.0	1770.0	FDD	Extended AWS / US
Band 11	1475.9	1495.9	1427.9	1447.9	FDD	Lower PDC / Japan
Band 12	729.0	746.0	699.0	716.0	FDD	Lower SMH blocks A,B,C / US
Band 13	746.0	756.0	777.0	787.0	FDD	Upper SMH block C / US
Band 14	758.0	768.0	788.0	798.0	FDD	Upper SMH block D / US
Band 15	2600.0	2620.0	1900.0	1920.0	FDD	Reserved
Band 16	2585.0	2600.0	2010.0	2025.0	FDD	Reserved
Band 17	734.0	746.0	704.0	716.0	FDD	Lower SMH blocks B,C / US
Band 18	860.0	875.0	815.0	830.0	FDD	lower 800 / Japan
Band 19	875.0	890.0	830.0	845.0	FDD	upper 800 / Japan
Band 20	791.0	821.0	832.0	862.0	FDD	Digital Dividend / EU
Band 21	1495.9	1510.9	1447.9	1462.9	FDD	Upper PDC / Japan
Band 22	3510.0	3590.0	3410.0	3490.0	FDD	–
Band 23	2180.0	2200.0	2000.0	2020.0	FDD	S-Band
Band 24	1525.0	1559.0	1626.5	1660.5	FDD	L-Band
Band 25	1930.0	1995.0	1850.0	1915.0	FDD	Extended PCS / US
Band 26	859.0	894.0	814.0	849.0	FDD	Extended CLR / US
Band 27	852.0	869.0	807.0	824.0	FDD	SMR
Band 28	758.0	803.0	703.0	748.0	FDD	/ APAC
Band 29	716.0	728.0	n/a	n/a	FDD	Lower SMH blocks D,E / US
Band 30	2350.0	2360.0	2305.0	2315.0	FDD	WCS blocks A,B / US
Band 31	462.5	467.5	452.5	457.5	FDD	–
AWS-2	1995.0	2000.0	1915.0	1920.0	FDD	AWS-2 / US
AWS-3	2155.0	2180.0	1755.0	1780.0	FDD	AWS-3 / US
Band iDEN	851.0	869.0	806.0	824.0	FDD	iDEN / US
Band 33	1900.0	1920.0	1900.0	1920.0	TDD	IMT / China
Band 34	2010.0	2025.0	2010.0	2025.0	TDD	IMT / China

Band	Downlink MS RX (EMI)		Uplink MS TX (EMC)		Duplex	Common Name / Region
	F_{low}	F_{high}	F_{low}	F_{high}		
Band 35	1850.0	1910.0	1850.0	1910.0	TDD	PCS (Uplink) / US
Band 36	1930.0	1990.0	1930.0	1990.0	TDD	PCS (Downlink) / US
Band 37	1910.0	1930.0	1910.0	1930.0	TDD	PCS (Duplex spacing) / US
Band 38	2570.0	2620.0	2570.0	2620.0	TDD	IMT / EU-Asia
Band 39	1880.0	1920.0	1880.0	1920.0	TDD	/ China
Band 40	2300.0	2400.0	2300.0	2400.0	TDD	CM / China
Band 41	2496.0	2696.0	2496.0	2696.0	TDD	BRS / EBS
Band 42	3400.0	3600.0	3400.0	3600.0	TDD	–
Band 43	3600.0	3800.0	3600.0	3800.0	TDD	–
Band 44	703.0	803.0	703.0	803.0	TDD	APAC

Note:

All frequencies in MHz

Table 66 GNSS and Connectivity Bands Used by Mobile Devices

System	Band	Downlink MS RX (EMI)		Uplink MS TX (EMC)		Duplex	Common Name / Region
		F_{low}	F_{high}	F_{low}	F_{high}		
GNSS	GPS L1	1574.4	1576.4	na	na	na	/ WW
GNSS	Glonass	1597.5	1606.5	na	na	na	/ WW
GNSS	Compass	1560.1	1562.1	na	na	na	/ Asia -> WW
GNSS	Galileo	1573.4	1577.5	na	na	na	in deployment
Connectivity	Bluetooth	2400.0	2483.0	2400.0	2483.0	TDD	ISM / WW
Connectivity	802.11b/g/n/ac	2400.0	2483.0	2400.0	2483.0	TDD	ISM / WW
Connectivity	802.11a/ac	4915.0	5825.0	4915.0	5825.0	TDD	/ WW
Connectivity	802.11.ad	57000.0	66000.0	57000.0	66000.0	TDD	ISM / WW
Connectivity	802.11.af	54.0	790.0	54.0	790.0	TDD	White space WiFi
RFID	NFC	13.6	13.6	13.6	13.6	TDD	/ WW
Audio Broadcast	FM	78.0	108.0	78.0	108.0	TDD	/ WW
Audio Broadcast	FM	65.8	74.0	65.8	74.0	TDD	/ Russia
Audio Broadcast	FM	76.0	90.0	76.0	90.0	TDD	/ Japan
Audio Broadcast	DAB-VHF3	174.0	240.0	na	na	na	/ EU
Audio Broadcast	DAB-L	1452.0	1492.0	na	na	na	/ US
Video Broadcast	CMMB	470.0	860.0	na	na	na	/ China

Note:

All frequencies in MHz

It is important to identify the lowest interference frequency with significant impact, as this sets '$f_{INT,MIN}$'. For this specification, $f_{INT,MIN}$ is decided to be 450 MHz, because it is identified as the lowest frequency of interest in the tables above.

B.3 Transmission Line Design

In most cases the transmission lines will either be designed as striplines and/or micro-striplines. The coupling between neighboring lines within a pair is small if the distance between them is >2x the dielectric thickness. For the separation of multiple pairs it is highly recommended to interleave the pairs with a ground or supply line in order to reduce coupling.

B.4 Reference Layer

In order to achieve good signal integrity and low EMI it is recommended that either a ground plane or a ground signal is in close proximity of any signal line.

B.5 Printed-Circuit Board

For boards with a large number of conductor layers the dielectric spacing between layers may become so small that it would be hard to meet the characteristic impedance requirements. In those cases a micro-stripline in the top or bottom layers may be a better solution. Hybrids consisting of a combination of micro-stripline and stripline are also viable solutions. A short segment of micro-stripline might be used near the driving or receiving IC where trace routing may be more intense. Then a short distance away a stripline may be the best option. Hybrid combinations of three-wire Lanes have been evaluated with good results.

B.6 Flex Circuits

Either two conductor layers or a reasonable connected cover layer makes it much easier to meet the specifications.

B.7 Series Resistance

The DC series resistance of the interconnect should be less than 5 Ω in order to meet the specifications. It is strongly recommended to keep the resistance in the ground connection below 0.2 Ω. Furthermore, it is recommended that the DC ground shift be less than 50mV, which may require an even lower value if a large current is flowing through this ground. The lower this ground series resistance value can be made, the better it is for reliability and robustness.

B.8 Connectors

Connectors usually cause some impedance discontinuity. It is important to carefully minimize these discontinuities by design, especially with respect to the through-connection of the reference layer. Although connectors are typically rather small in size, the wrong choice can mess-up signals completely. Please note that the contact resistance of connectors is part of the total series resistance budget and should therefore be sufficiently low.

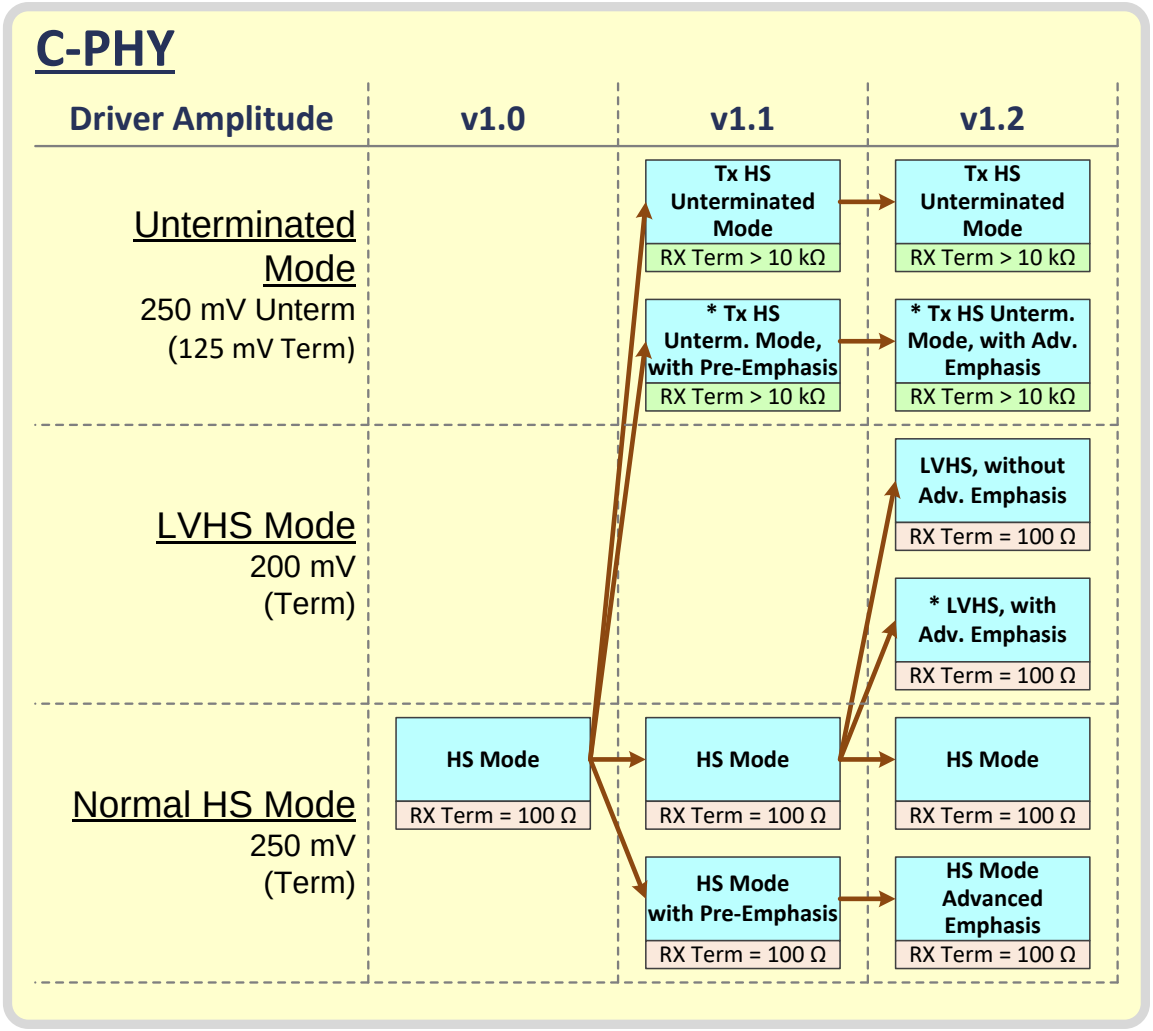
Annex C Implementation Guidelines (Informative)

C.1 Guidance Regarding High-Speed Mode Options

This section provides some clarity regarding the different High-Speed signaling options that are available for the transmitter and receiver. It may be helpful to view in a single chart which options are possible and which options in the transmitter are compatible with options in the receiver. Examples of options include:

- Driver strength (driver VDD): Normal Mode, LVHS Mode, Unterminated Mode.
- Tx equalization: no emphasis/de-emphasis, v1.1 emphasis, v1.2 advanced emphasis.
- Receiver: operating with or without a 100 Ω termination.

Figure 107 shows the evolution of High-Speed mode signaling options.



* LVHS Mode or Tx HS Unterminated Mode with v1.1 Pre-Emphasis or v1.2 Advanced Emphasis is allowed but symbol rate guidance is not provided in the specification.

Figure 107 Evolution of HS Options

Table 67 shows the compatibility of different transmit and receive High-Speed mode options.

Table 67 Tx and Rx High-Speed Options Matrix

		Rx Options			
		Rx HS Mode	Rx HS Mode With Receiver Calibration	Receiver Unterminated Mode	Receiver Unterminated With Receiver Calibration
Tx Options	Tx HS Mode	✓	✓		
	Tx HS Unterminated Mode			✓	✓
	Tx Pre-Emphasis, TxEQ Option (introduced in v1.1)	✓	✓	note 1	note 1
	Tx Advanced Emphasis, TxEQ (introduced in v1.2)	✓	✓	note 1	note 1
	LVHS Mode	✓	✓		
	LVHS Mode with Tx Emphasis (any form)	✓	✓		

Note:

1. *Tx Pre-Emphasis (introduced in v1.1) or Tx Advanced Emphasis (introduced in v1.2) may be used in conjunction with the Receiver Unterminated Mode if the Tx HS Unterminated Mode is also simultaneously selected.*

C.2 Receiver Pairwise Common Mode Level Guidance

This section provides guidance regarding the pairwise common mode voltage present at the input of each of the three differential receivers in a Lane. The electrical specifications in ***Section 9.2.1*** specify DC common-point level variation at the receiver based on (A+B+C)/3.

Each of the three differential receivers are exposed to a common mode level that changes from one Unit Interval to the next, which is a normal characteristic of 3-Phase signal encoding. The three receivers will operate at three different common mode levels at one time instant as shown in the ***Figure 108***. The range of this variation is approximately (HS Low + HS Mid)/2 to (HS Mid + HS High)/2. The common mode level at each of the three differential receivers is $(V_A+V_B)/2$, $(V_B+V_C)/2$, $(V_C+V_A)/2$.

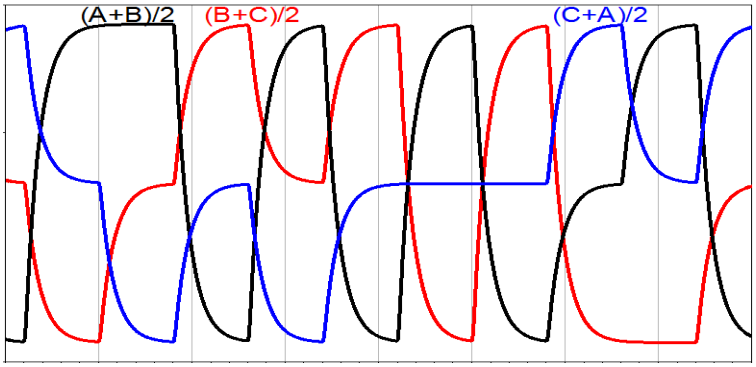


Figure 108 Common Mode Waveform for Each of Three Differential Receivers

Additional characteristics (both AC and DC) that affect the common mode range of the differential receivers are:

- The DC common-point range of the transmitter, V_{CPTX}
- Static ground shift between the transmitter and receiver, V_{GNDSH}
- High frequency and low frequency common-point interference between the transmitter and receiver, $\Delta V_{CPRX(HF)}$ and $\Delta V_{CPRX(LF)}$

Figure 109 shows an example of the high frequency and low frequency common-point interference between the transmitter and receiver. A likely source of this type of interference is a conducted signal that is a result of currents through parasitic inductance in the ground path between the transmitter and receiver. The transitions in this conducted signal can be uncorrelated with the high speed signaling between the transmitter and receiver. The use of a C-PHY common mode filter can reduce the amplitude of an AC common mode signal variation at the receiver.

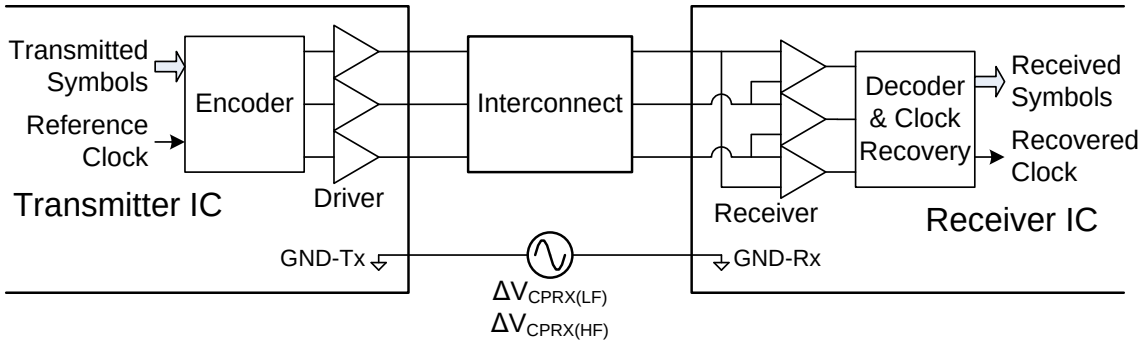


Figure 109 Common-point Signal Difference Between Tx and Rx

C.2.1 Pairwise Common Mode without AC Common Mode Noise

The worst-case pairwise common mode variation based on the 3-Phase signaling without AC common mode noise can be derived from the electrical specifications in ***Section 9.2.1***. The variation is computed using the receiver common-point range $V_{CPRX(DC)}$, which is 95 mV to 390 mV, and minimum and maximum single-ended voltages seen at the receiver V_{ILHS} and V_{IHHS}, which are -40 mV and 535 mV, respectively. The receiver common-point voltage range $V_{CPRX(DC)}$ is ±80 mV wider than the transmitter common-point voltage range V_{CPTX} to allow for AC common mode noise $V_{CPRX(LF)}$ and $V_{CPRX(HF)}$, and DC ground shift between the transmitter and receiver.

Figure 110 shows an example of the pairwise common mode signal at the inputs of each of the three differential receivers for the largest signal amplitude without AC common mode noise at the receiver. The largest pairwise common mode variation occurs at $V_{CPRX(DC)(min)}$ when |VOD| strong = $2•(V_{CPRX(DC)(min)} - V_{ILHS(min)})$, and at $V_{CPRX(DC)(max)}$ when |VOD| strong = $2•(V_{IHHS(max)} - V_{CPRX(DC)(max)})$.

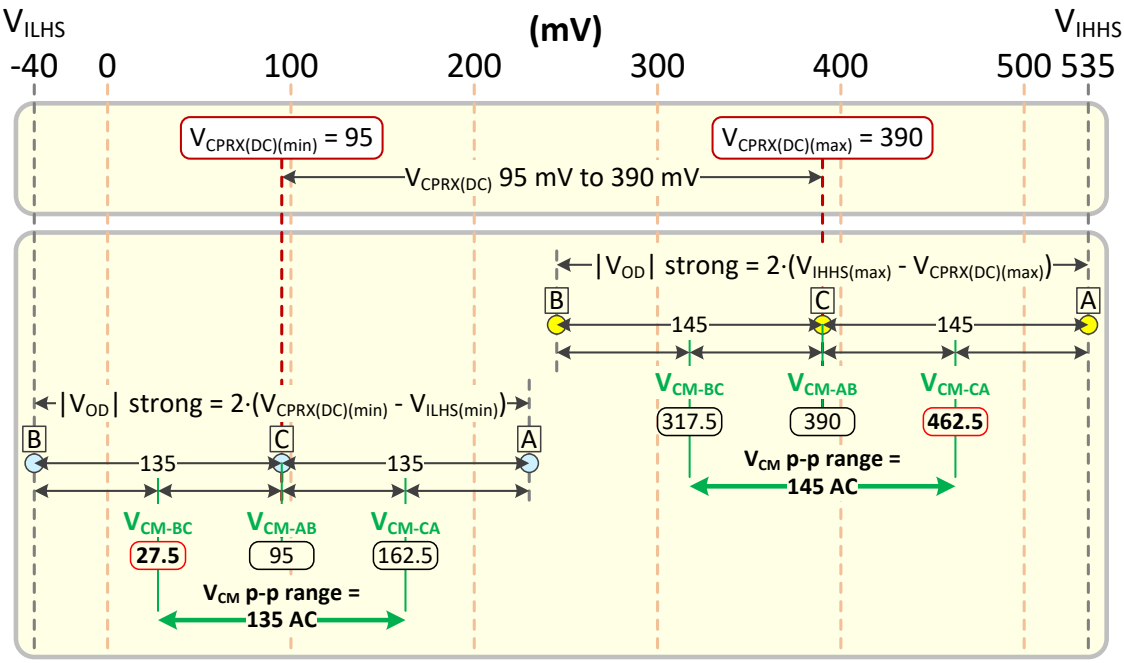


Figure 110 Pairwise Common Mode without AC Common Mode Noise

C.2.2 Pairwise Common Mode with Low Frequency AC Common Mode Noise

Figure 111 shows an example of the pairwise common mode signal at the inputs of each of the three differential receivers for the largest signal amplitude with low frequency AC common mode noise in the range of 50 MHz to 450 MHz at the receiver. The levels are based on the V_{CPTX} range of 175 mV to 310 mV, and $\Delta V_{CPRX(LF)}$ at 25 mV peak. The levels are also based on a ground shift of -40 mV at the lower end of the range and +50 mV at the upper end of the range. The ground shift is less than 50 mV in some cases to keep from violating the V_{ILHS} or V_{IHHS} specifications while $\Delta V_{CPRX(LF)}$ is set to the maximum allowed value. With these conditions, the maximum AC peak-to-peak common mode variation for each differential receiver is 200 mV, as shown in ***Figure 111***.

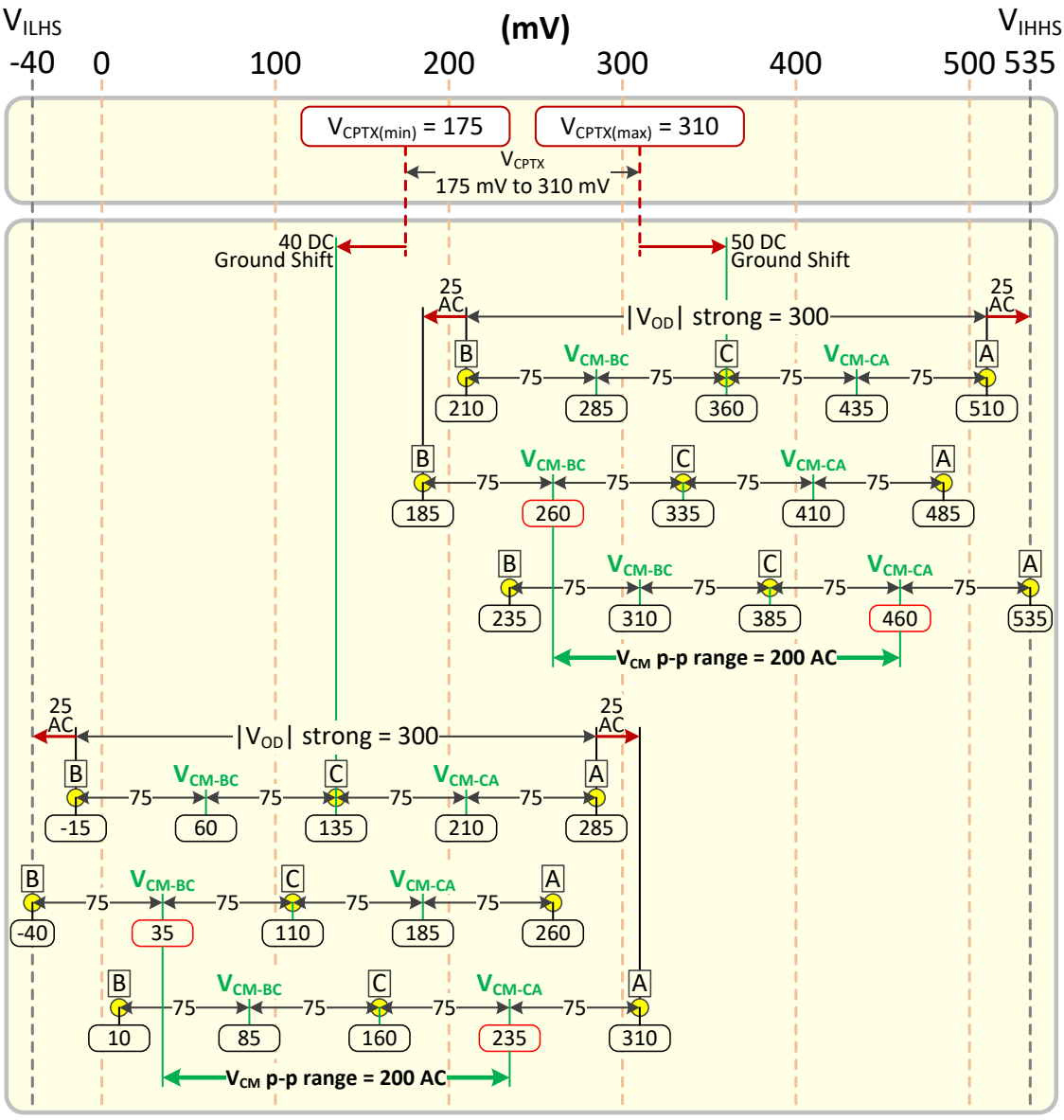
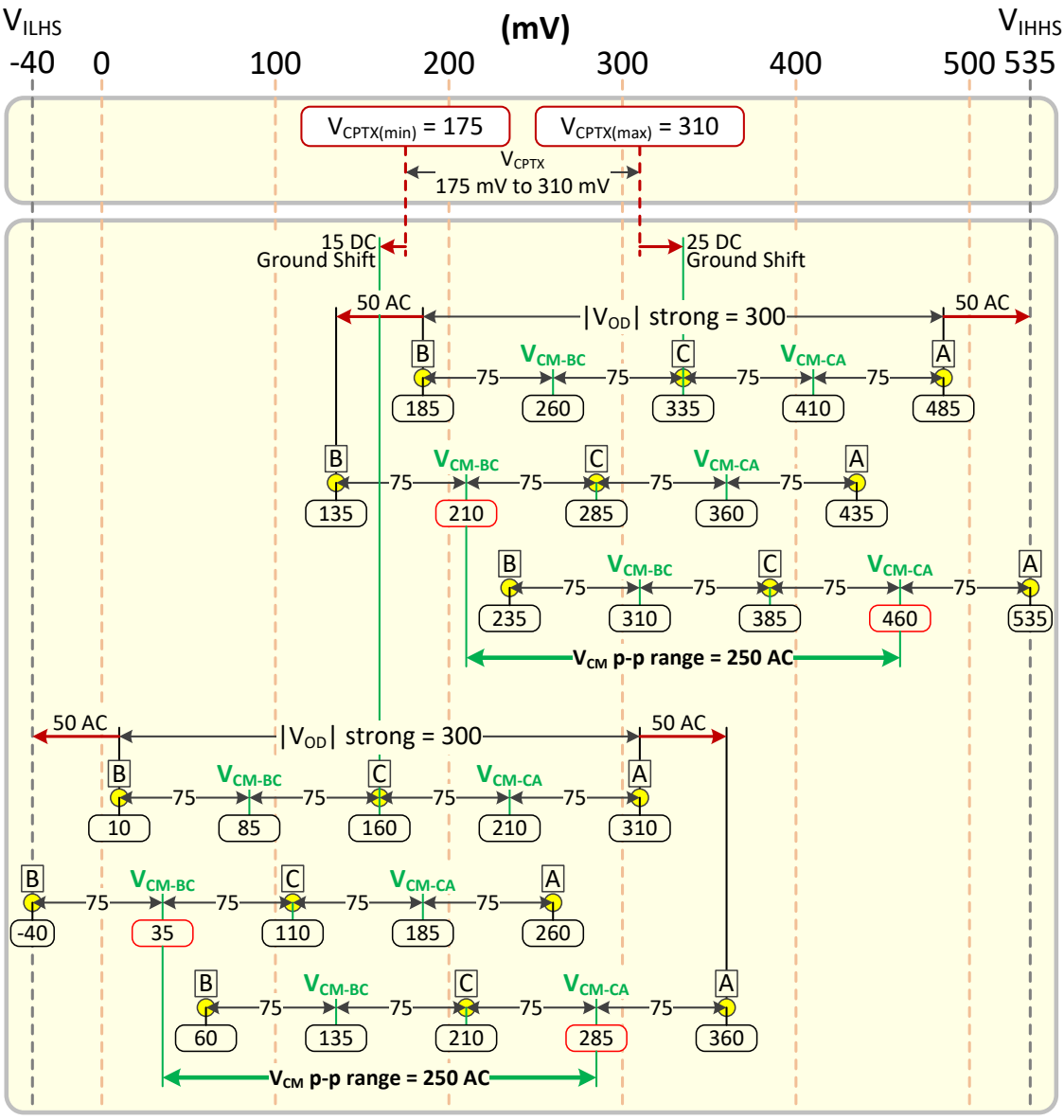


Figure 111 Pairwise Common Mode with Low Frequency Common Mode Noise

C.2.3 Pairwise Common Mode with High Frequency AC Common Mode Noise

Figure 112 shows an example of the pairwise common mode signal at the inputs of each of the three differential receivers for the largest signal amplitude with high frequency AC common mode noise above 450 MHz at the receiver. The levels are based on the V_{CPTX} range of 175 mV to 310 mV, and $\Delta V_{CPRX(HF)}$ at 50 mV peak. The levels are also based on a ground shift of -15 mV at the lower end of the range and +25 mV at the upper end of the range. The ground shift is less than 50 mV in some cases to keep from violating the V_{ILHS} or V_{IHHS} specifications while $\Delta V_{CPRX(HF)}$ is set to the maximum allowed value. With these conditions, the maximum AC peak-to-peak common mode variation for each differential receiver is 250 mV, as shown in ***Figure 112***.



2479

Figure 112 Pairwise Common Mode with High Frequency Common Mode Noise

Participants

The list below includes those persons who participated in the Working Group that developed this Specification and who consented to appear on this list.

Ahmed Aboulella, Mixel, Inc.

Yahya Akram, Mixel, Inc.

Mario Ackers, Toshiba Corporation

Thomas Blon, Silicon Line GmbH

Aditya Burra, Tektronix, Inc.

Seok-Keun Choi, Keysight Technologies Inc.

Keyur Diwan, Tektronix, Inc

Mahmoud El-Banna, Mixel, Inc.

Luke Fay, Sony Corporation

Ralf Gaisbauer, Toshiba Corporation

Mohamed Hafed, Introspect Technology

Hiroaki Hayashi, Sony Corporation

David Herrig, NVIDIA

Henrik Icking, Intel Corporation

Tom Kopet, ON Semiconductor

Marcin Kowalewski, Synopsys, Inc.

C. K. Lee, Qualcomm Incorporated

Thomas Marik, BitifEye Digital Test Solutions GmbH

Darshan Mehta, Tektronix, Inc.

Raj Kumar Nagpal, Synopsys, Inc.

Duane Quiet, Intel Corporation

Parthasarathy Raju, Tektronix, Inc.

P. E. Ramesh, Tektronix, Inc.

Jim Rippie, MIPI Alliance, Inc. (staff)

Ravi Rudraraju, Intel Corporation

Victor Sanchez-Rico, BitifEye Digital Test Solutions GmbH

Roland Scherzinger, Keysight Technologies Inc.

Bill Simms, NVIDIA

Dong Hyun Song, SK Hynix

Tatsuya Sugioka, Sony Corporation

Rick Wietfeldt, Qualcomm Incorporated

George Wiley, Qualcomm Incorporated

Charles Wu, OmniVision Technologies, Inc.

Karthikeyan Rajendran, Tektronix, Inc.

Suryakant Kumar, Tektronix, Inc.

The list below includes those persons who participated in the Working Group that developed the v1.0 and v1.1 Specification and who consented to appear on this list.

Ahmed Aboulella, Mixel, Inc.

Mario Ackers, Toshiba Corporation

Bhupendra Ahuja, NVIDIA

Andrew Baldman, University of New Hampshire InterOperability Lab (UNH-IOL)

Dominique Brunel, STMicroelectronics

Mara Carvalho, Synopsys, Inc.

Min-Jie Chong, Agilent Technologies Inc.

Doug Day, Toshiba Corporation

Kirill Dimitrov, SanDisk Corporation

Keyur Diwan, Tektronix, Inc.

Ken Drottar, Intel Corporation

Mahmoud El-Bana, Mixel, Inc.

Michael Fleischer-Reumann, Agilent Technologies Inc.

Ralf Gaisbauer, Toshiba Corporation

Mohamed Hafed, Introspect Technology

Will Harris, Advanced Micro Devices, Inc.

Hiroaki Hayashi, Sony Corporation

Henrik Icking, Intel Corporation

Tom Kopet, Aptina Imaging Corporation

Marcin Kowalewski, Synopsys, Inc.

Amit Laknaur, NVIDIA

C. K. Lee, Qualcomm Incorporated

Jason Lee, NVIDIA

Delbert Liao, MediaTek Inc.

Kenneth Ma, Broadcom Corporation

Jiri Macku, Silicon Line GmbH

Thomas Marik, BitifEye Digital Test Solutions GmbH

Raj Kumar Nagpal, STMicroelectronics

Kinshuk Parekh, Cadence Design Systems, Inc.

Joao Pereira, Synopsys, Inc.

Duane Quiet, Intel Corporation

John Raitz, The Moving Pixel Company

Parthasarathy Raju, Tektronix, Inc.

P. E. Ramesh, Tektronix, Inc.

Jim Rippie, MIPI Alliance, Inc. (staff)

Ravi Rudraraju, Intel Corporation

Joseph Schachner, LeCroy Corporation

Omer Schori, Cadence Design Systems, Inc.

Jos Sebastian, GDA Technologies

Sho Sengoku, Qualcomm Incorporated

Sergio Silva, Synopsys, Inc.

Scott Silver, The Moving Pixel Company

Bill Simms, NVIDIA

Dong Hyun Song, SK Hynix

Tatsuya Sugioka, Sony Corporation

Haran Thanigasalam, Intel Corporation

Aravind Vijayakumar, Cadence Design Systems, Inc.

Manuel Weber, Toshiba Corporation

Rick Wietfeldt, Qualcomm Incorporated

George Wiley, Qualcomm Incorporated

Charles Wu, OmniVision Technologies, Inc.