Lattice Nexus 2 SERDES/PCS User Guide

Preliminary Technical Note

FPGA-TN-02395-0.80

April 2025

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults, and all associated risk is the responsibility entirely of the Buyer. The information provided herein is for informational purposes only and may contain technical inaccuracies or omissions, and may be otherwise rendered inaccurate for many reasons, and Lattice assumes no obligation to update or otherwise correct or revise this information. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. LATTICE PRODUCTS AND SERVICES ARE NOT DESIGNED, MANUFACTURED, OR TESTED FOR USE IN LIFE OR SAFETY CRITICAL SYSTEMS, HAZARDOUS ENVIRONMENTS, OR ANY OTHER ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, INCLUDING ANY APPLICATION IN WHICH THE FAILURE OF THE PRODUCT OR SERVICE COULD LEAD TO DEATH, PERSONAL INJURY, SEVERE PROPERTY DAMAGE OR ENVIRONMENTAL HARM (COLLECTIVELY, "HIGH-RISK USES"). FURTHER, BUYER MUST TAKE PRUDENT STEPS TO PROTECT AGAINST PRODUCT AND SERVICE FAILURES, INCLUDING PROVIDING APPROPRIATE REDUNDANCIES, FAIL-SAFE FEATURES, AND/OR SHUT-DOWN MECHANISMS. LATTICE EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS OF THE PRODUCTS OR SERVICES FOR HIGH-RISK USES. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Inclusive Language

This document was created consistent with Lattice Semiconductor’s inclusive language policy. In some cases, the language in underlying tools and other items may not yet have been updated. Please refer to Lattice’s inclusive language FAQ 6878 for a cross reference of terms. Note in some cases such as register names and state names it has been necessary to continue to utilize older terminology for compatibility.

Contents

Figures

Tables

Abbreviations in This Document

A list of abbreviations used in this document.

Abbreviation	Definition
AC	Alternating Current
AN	Auto Negotiation
BER	Bit Error Ratio
BIST	Built In Self-Test
CDR	Clock and Data Recovery
CE	Consumer Electronic
CIS	CMOS Image Sensor
CoaXPress	An interface to connect devices (typical cameras) to host (typical frame grabbers)
COMMA	The seven bit comma string is defined as either 7'b0011111 (comma+) or 7'b1100000 (comma-)
CTLE	Continuous Time Linear Equalizer
DC	Direct Current
DDR	Double Data Rate
DFE	Decision Feedback Equalization
DP	DisplayPort
DSP	Digital Signal Processor
eDP	Embedded DisplayPort
EMI	Electro-Magnetic Interference
EPCS	External PCS
FEC	Forward Error Correction
FFE	Feed Forward Equalizer
FIFO	First Input First Output
FIR	Finite Impulse Response
FPGA	Field-Programmable Gate Array
FS	Full Swing
FSM	Finite State Machine
HBR	High Bit Rate
HBR2	High Bit Rate 2
HBR3	High Bit Rate 3
Gen1/Gen2/Gen3	Generation 1/Generation 2/Generation 3
GMII	Gigabit Media Independent Interface
GUI	Graphic User Interface
IP	Intellectual Property
IPG	Inter-Packet Gap
ISI	Inter-Symbol Interference
JIIA	Japan Industrial Imaging Association
JTAG	Joint Test Action Group
L2	A low power state inside PCIe LTSSM
LF	Low Frequency
LMMI	Lattice Memory Mapped Interface
LTSSM	Link Training and Status State Machine
MAC	Media Access Control
MPCS	Multi-protocol Physical Coding Sublayer
NL	Number of Lane
PC	Personal Computer

Abbreviation	Definition
PCI	Peripheral Component Interconnect
PCIe / PCI-E	PCI Express
PCS	Physical Coding Sublayer
PFD	Phase Frequency Detector
PMA	Physical Media Attachment
PIPE	PHY Interface for PCI Express
PLL	Phase Locked Loop
PPM	Parts Per Million
QSGMII	Quad Serial Gigabit Media Independent Interface
RBR	Reduced Bit Rate
RC	Root Complex
RS	Reconciliation Sublayer
Rx	Receiver
RxDP	Receiver Data Positive pin
RxDN	Receiver Data Negative pin
SERDES	Serializer/Deserializer
SGMII	Serial Gigabit Media Independent Interface
SKP	SKIP order sets defined by PCI Express
SLVS-EC	Scalable Low Voltage Signaling with Embedded Clock
SSC	Spread Spectrum Clocking
Sync	Synchronous
Tx	Transmitter
UI	Unit Interval
VC0_RX	Virtual Channel 0 Receiver
VC0_TX	Virtual Channel 0 Transmitter
VESA	Video Electronics Standards Association
WAKE#	A wake up signal defined by PCIe
WIS	WAN Interface Sublayer
XAUI	10 Gigabit Ethernet Attachment Unit Interface
XGMII	10 Gigabit Media Independent Interface

1. Introduction

The Lattice Certus™-N2 device families of Lattice Nexus™ 2 platform have up to 8 channels embedded SERDES with associated Physical Coding Sublayer (PCS) logic, which supports PCI Express Gen1/2/3/4 hard IP Core, DisplayPort, JESD204B, 10G Ethernet, SLVS-EC, CPRI, and CoaXPress protocols.

Each SERDES channel of the Certus-N2 device family contains dedicated transmit and receive logic for high-speed, full-duplex serial data transfer at data rates of up to 16 Gbps. The MPPCS logic in each channel can be configured to support corresponding protocols. In addition, the protocol-based logic can be fully or partially bypassed in a few configurations to provide the flexibility of designing your own high-speed data interface. SERDES channel input can be independently AC-coupled or DC-coupled to meet the requirements of different protocols.

1.1. Using this Technical Note

This technical note provides a thorough description of the complete functionality of the embedded SERDES and associated PCS logic, including the description of:

- Architecture of the Nexus 2 SERDES
- SERDES/PCS function
- Clock and reset
- SERDES/PCS debug capabilities
- SERDES/PCS register access
- SERDES/PCS usage for different protocols
- SERDES/MPPCS block latency
- SERDES/MPPCS generation in the Lattice Radiant™ software
- The status and control registers associated with the SERDES and PCS logic, which can be accessed through the Lattice Memory Mapped Interface (LMMI)

The electrical and timing characteristics of the embedded SERDES and the package pinout information are provided in the Lattice Nexus 2 Platform – Specifications Data Sheet (FPGA-DS-02121).

The Lattice Radiant design tools support all modes of PCS. Most modes are dedicated to applications for specific industry standard data protocols listed in the Features section. Other modes are more general purpose in nature in order to let you customize your own application settings. The Radiant design tools allow you to define the mode for each Quad in the design. This technical note describes operation of the SERDES and PCS for all modes supported by the Radiant software.

2. Features

Key features of the Lattice Nexus 2 SERDES/PCS include:

- Single Channel MPPCS Functionalities:
  - From 1.25 Gbps to 16 Gbps per channel
  - Word alignment and link synchronization FSM
  - PCI Express Gen1/2/3/4 hard IP Core, DisplayPort 1.3/1.4, JESD204B, Ethernet, SLVS-EC, CPRI, and CoaXPress protocols
  - User-specified generic 8B/10B mode
  - Ethernet 64B/66B encoding and decoding
  - Per channel configuration
  - Scrambler and descrambler based on protocol specific polynomials
  - Forward Error Correction for the JESDC and Ethernet protocols
  - Clock tolerance compensation with low latency
  - PCS bypass mode to allow PHY interface directly to the FPGA fabric
- Multiple Lane Operation:
  - 4 lanes per MPPCS block. Can be reduced to 1, 2, or 3-lane operation or can be increased by combining MPPCS blocks for up to 8-lane operation
- Integrated Loopback Modes for System Debugging:
  - Two loopback modes are provided by SERDES (PMA) which is Tx to Rx serial loopback and Rx to Tx parallel loopback
- Diagnostic Tools:
  - Rx Eye Monitor, which provides integrated 2D eye-scan for each Rx channel
  - Built-in BERT pattern generator and checker
- Easy-to-Use Design Interface in the Lattice Radiant design tool:
  - Easy-to-use graphic user interface (GUI)
  - Easy-to-use soft IP cores and example projects

3. Supported Standards

The supported standards are listed in Table 3.1. Note that only the flip-chip package-based device can support standards with data rates higher than 6.25 Gbps. The supported standards vary between Lattice Nexus 2 families. For more details, refer to the Lattice Nexus 2 Platform – Overview Data Sheet (FPGA-DS-02122).

Table 3.1. Standards Supported by the SERDES/PCS

Standard	Device[2]	Data Rate (Mbps)	Number of Link Width	Encoding Style
PCI Express Gen1[1]	All Nexus 2 devices	2500	×1, ×2, ×4	8b10b
PCI Express Gen2[1]	All Nexus 2 devices	5000	×1, ×2, ×4	8b10b
PCI Express Gen3[1]	All Nexus 2 devices	8000	×1, ×2, ×4	128b130b
PCI Express Gen4[1]	All Nexus 2 devices	16000	×1, ×2, ×4	128b130b
Ethernet 1.25 G	All Nexus 2 devices	1250	×1	8b10b
Ethernet 3.125 G	All Nexus 2 devices	3125	×1, ×4	8b10b
Ethernet 5 G	All Nexus 2 devices	5000	×1	8b10b
Ethernet 6.25 G	All Nexus 2 devices	6250	×2	8b10b
Eth_10GBASE-R	All Nexus 2 devices	10312.5	×1	64b66b
DP/eDP RBR	All Nexus 2 devices	1620	×1, ×2, ×4	8b10b
DP/eDP HBR	All Nexus 2 devices	2700	×1, ×2, ×4	8b10b
DP/eDP HBR2	All Nexus 2 devices	5400	×1, ×2, ×4	8b10b
DP/eDP HBR3	All Nexus 2 devices	8100	×1, ×2, ×4	8b10b
SLVS-EC	All Nexus 2 devices	1250	×4, x8[3]	8b10b
SLVS-EC	All Nexus 2 devices	2500	×4, x8[3]	8b10b
SLVS-EC	All Nexus 2 devices	5000	×4, x8[3]	8b10b
CoaXPress CXP-3	All Nexus 2 devices	3125	×1, ×2, ×4, ×8[3]	8b10b
CoaXPress CXP-6	All Nexus 2 devices	6250	×1, ×2, ×4, ×8[3]	8b10b
CoaXPress CXP-10	All Nexus 2 devices	10000	×1, ×2, ×4, ×8[3]	8b10b
CoaXPress CXP-12	All Nexus 2 devices	12500	×1, ×2, ×4, ×8[3]	8b10b
CPRI	All Nexus 2 devices	1228.8, 2457.6, 3072, 4915, 6144	×1, ×2, ×4, ×8[3]	8b10b
CPRI	All Nexus 2 devices	8110.08, 9830.4, 10137.6, 12165.12	×1, ×2, ×4, ×8[3]	64b66b
eCPRI	All Nexus 2 devices	10312.5	×1,	64b66b
JESD204B	All Nexus 2 devices	3125, 6250	×1, ×2, ×4, ×8[3]	8b10b
JESD204B	All Nexus 2 devices	12500	×1, ×2, ×4, ×8[3]	64b66b

Notes:

1. Lattice Nexus 2 device families support a maximum of four lanes PCIe with hard IP, refer to the SERDES Architecture and PCI Express Architecture sections for more details.
2. All Nexus 2 devices cover the LN2-CT-06, LN2-CT-10, LN2-CT-16, and LN2-CT-20 devices.
3. Only LN2-CT-16 and LN2-CT-20 devices support ×8 link width.

3.1. Device Architecture

The SERDES block is arranged in Quads containing logic for full-duplex data channels. Figure 3.1, Figure 3.2, Figure 3.3, and Figure 3.4 show the arrangement of SERDES/PCS Quads on the LN2-CT20, LN2-CT16, LN2-CT10, and LN2-CT06 devices.

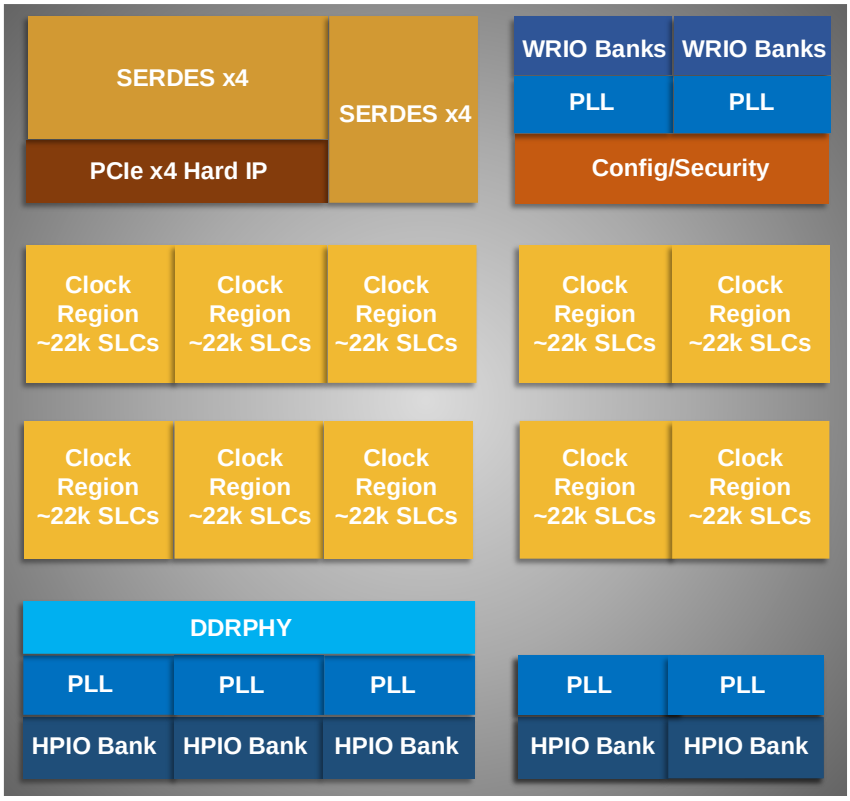

Figure 3.1. High-level Device Floorplan (LN2-CT20 Device)

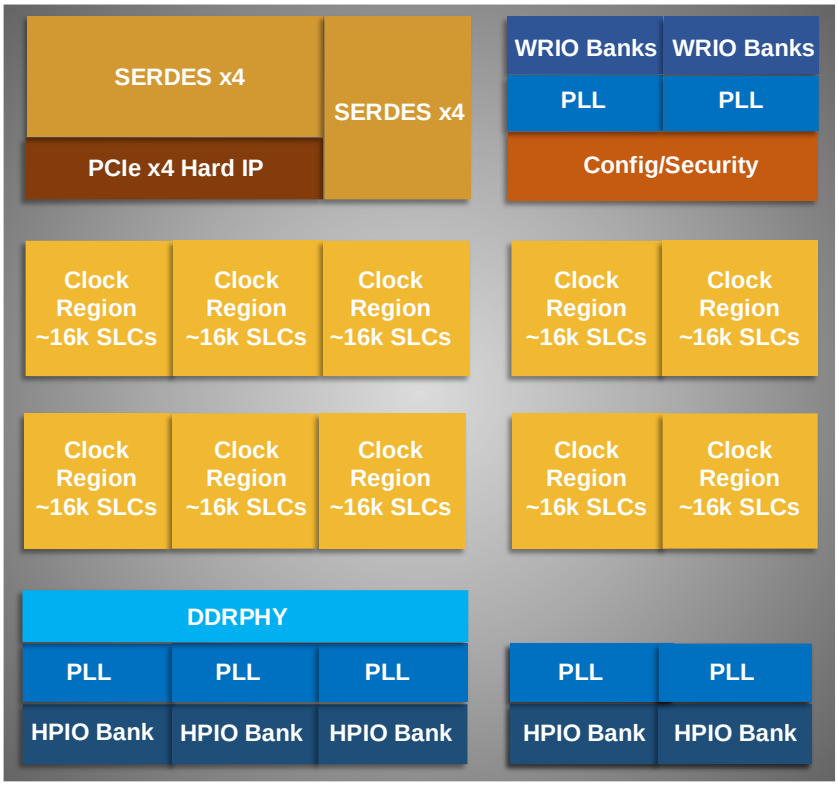

Figure 3.2. High-level Device Floorplan (LN2-CT16 Device)

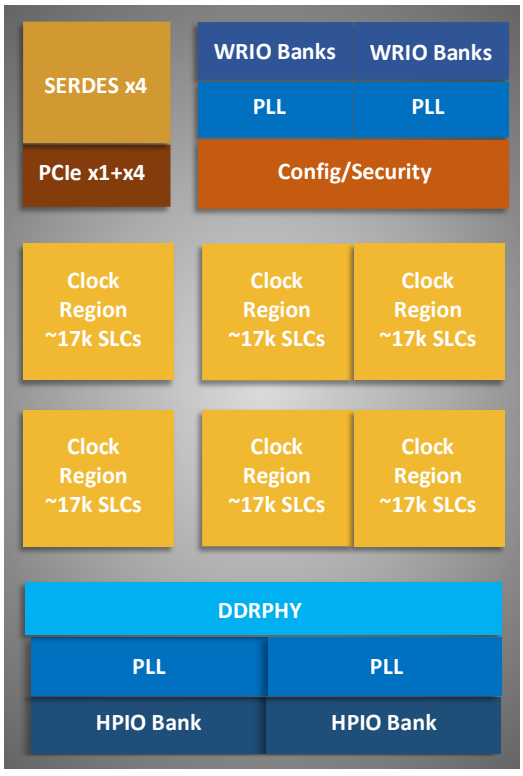

Figure 3.3. High-level Device Floorplan (LN2-CT10 Device)

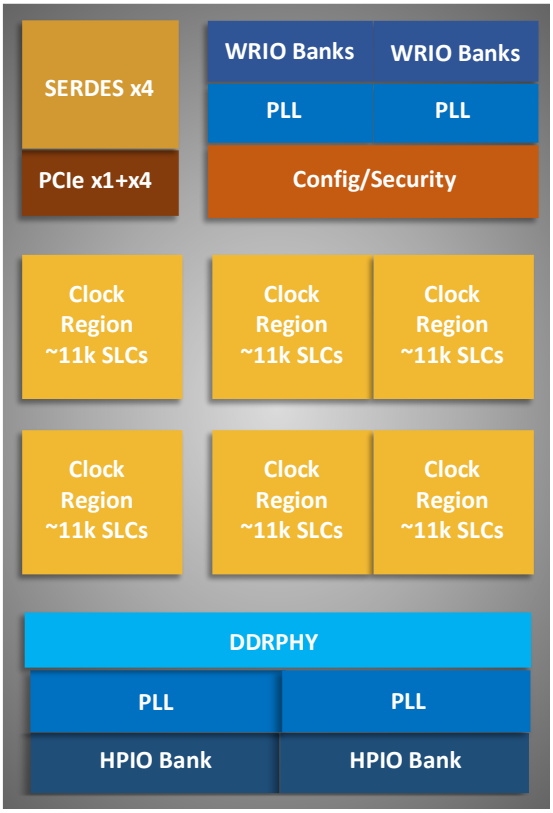

Figure 3.4. High-level Device Floorplan (LN2-CT06 Device)

Table 3.2 shows the maximum number of available SERDES channels for each Lattice Nexus 2 device. Refer to the Lattice Nexus 2 Platform – Overview Data Sheet (FPGA-DS-02122) or Lattice Nexus 2 Platform – Specifications Data Sheet (FPGA-DS-02121) for more details on the actual number of channels varying from package to package.

Table 3.2. Maximum Number of SERDES Channels per Lattice Nexus 2 Device

Package	LN2-CT20	LN2-CT16	LN2-CT10	LN2-CT06
SERDES Quads	2	2	1	1
SERDES Channels	8	8	4	4

3.2. SERDES Architecture

Each Nexus 2 SERDES Quad includes four channels. Each channel includes one PMA and MPPCS. Each PMA channel integrates CDR for Receiver and PLL for Transmitter. The Nexus 2 SERDES Quad also incorporates a raw PCS which operates as a PMA controller to manage the register access, calibration, and equalization adaptation of the PMA channels within the Quad.

The Lattice Nexus 2 device also integrates PCI Express ×4 Link Layer block. The PCI Express Link Layer block can be configured as ×1, ×2, or ×4 mode. Table 3.3 shows the maximum number of available PCI Express ×4 Link Layer blocks information for the Nexus 2 devices. Refer to the Lattice Nexus 2 Platform – Overview Data Sheet (FPGA-DS-02122) or Lattice Nexus 2 Platform – Specifications Data Sheet (FPGA-DS-02121) for more details on the actual number of channels varying from package to package.

Table 3.3. Maximum Number of PCI Express ×4 Link Layer Blocks per Lattice Nexus 2 Device

Package	LN2-CT20	LN2-CT16	LN2-CT10	LN2-CT06
PCI Express ×4 Link Layer Blocks	1	1	1	1
SERDES Quad	1	1	1	1

Figure 3.5 shows Lattice Nexus 2 device SERDES Quad architecture. The MPPCS can also be bypassed so that the SERDES module works in PCS-Bypass mode.

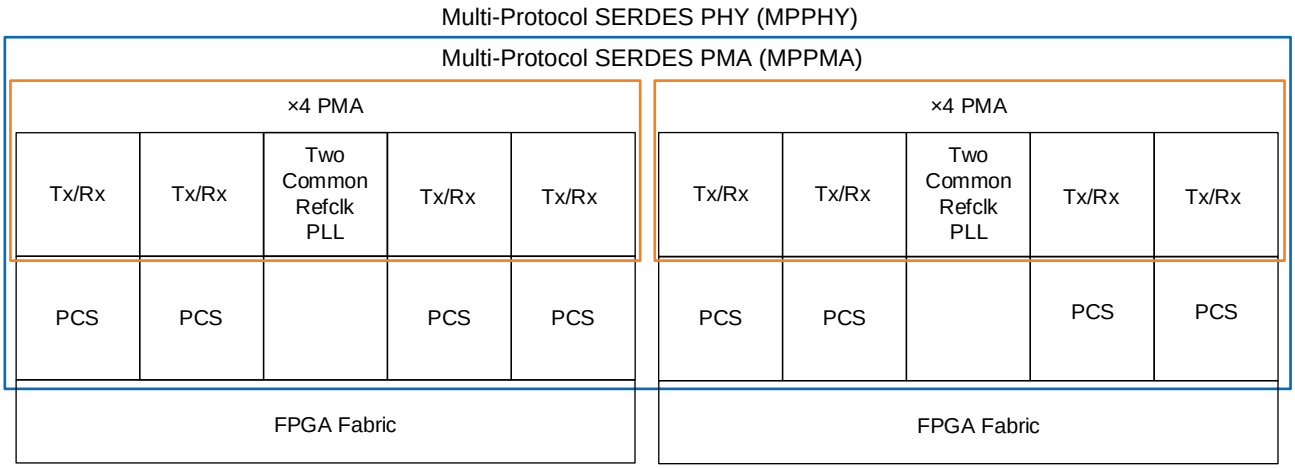

Figure 3.5. Lattice Nexus 2 SERDES Quad Architecture

3.3. PCI Express Architecture

The Lattice Nexus 2 PCI Express ×4 Link Layer block is a hardened IP, which supports PCI Express Gen1, Gen2, Gen3, and Gen4. The PCI Express ×4 Link Layer block is compatible with the PCI-SIG PCI Express Base Spec version 4.0. The PCI Express ×4 Link Layer block implements a portion of the PHY layer, Data Link Layer, and Transaction Layer. The PCI Express ×4 Link Layer block together with the MPPCS, raw PCS and PMA, and optional hardened DMA Controller constitute the complete PCI Express ×4 Hard IP.

The PCI Express Hard IP supports up to eight physical functions per link. Each of the eight functions has an independent PCI Express configurations space. The IP also supports ECC and parity data path protection.

Figure 3.6 shows the architecture of the Lattice Nexus 2 PCI Express IP. There are eight physical functions in the PCI Express ×4 Link Layer block. The core Configuration and Status Registers can be accessed using the corresponding LMMI interface per Link Layer block. PCI Express ×4 Link Layer block uses a 256-bit fabric interface at 250 MHz for four lanes.

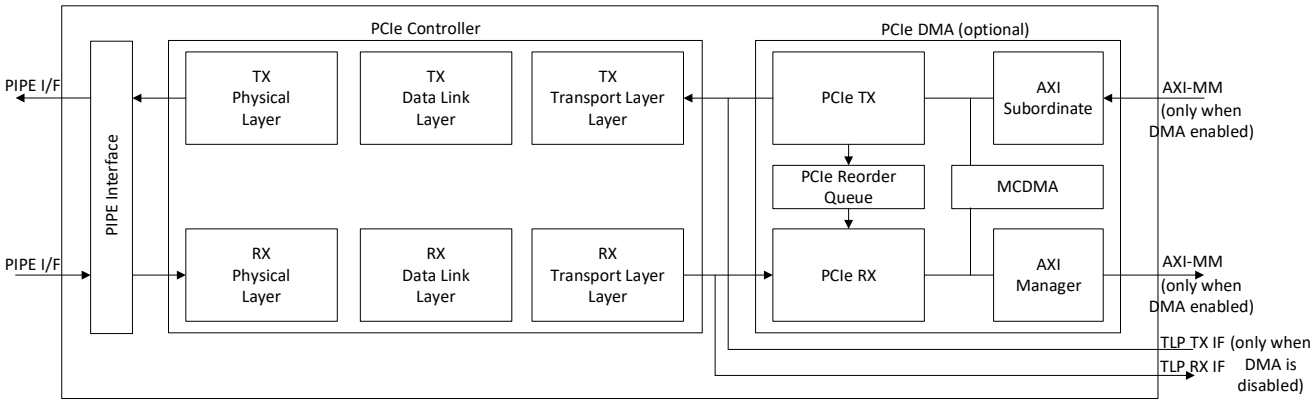

Figure 3.6. PCI Express Hard IP Architecture

PCI Express Link Layer can be configured as ×4, ×2, and ×1 modes. Table 3.4 describes the SERDES/PCS lane mapping details.

Table 3.4. PCI Express ×4 Link Layer Lane Mapping

Mode Name	Description
×4	Lane 0 till Lane 3 of SERDES Quad 0 are used
×2	Lane 0 and Lane 1 of SERDES Quad 0 are used
×1	Lane 0 of SERDES Quad 0 is used

Figure 3.7 shows the PCI Express Link Layer block functional diagram. For more detailed information on Lattice Nexus 2 PCI Express features, function descriptions, and IP usage, refer to the PCIe X4 IP Core User Guide (FPGA-IPUG-02126) (TBD).

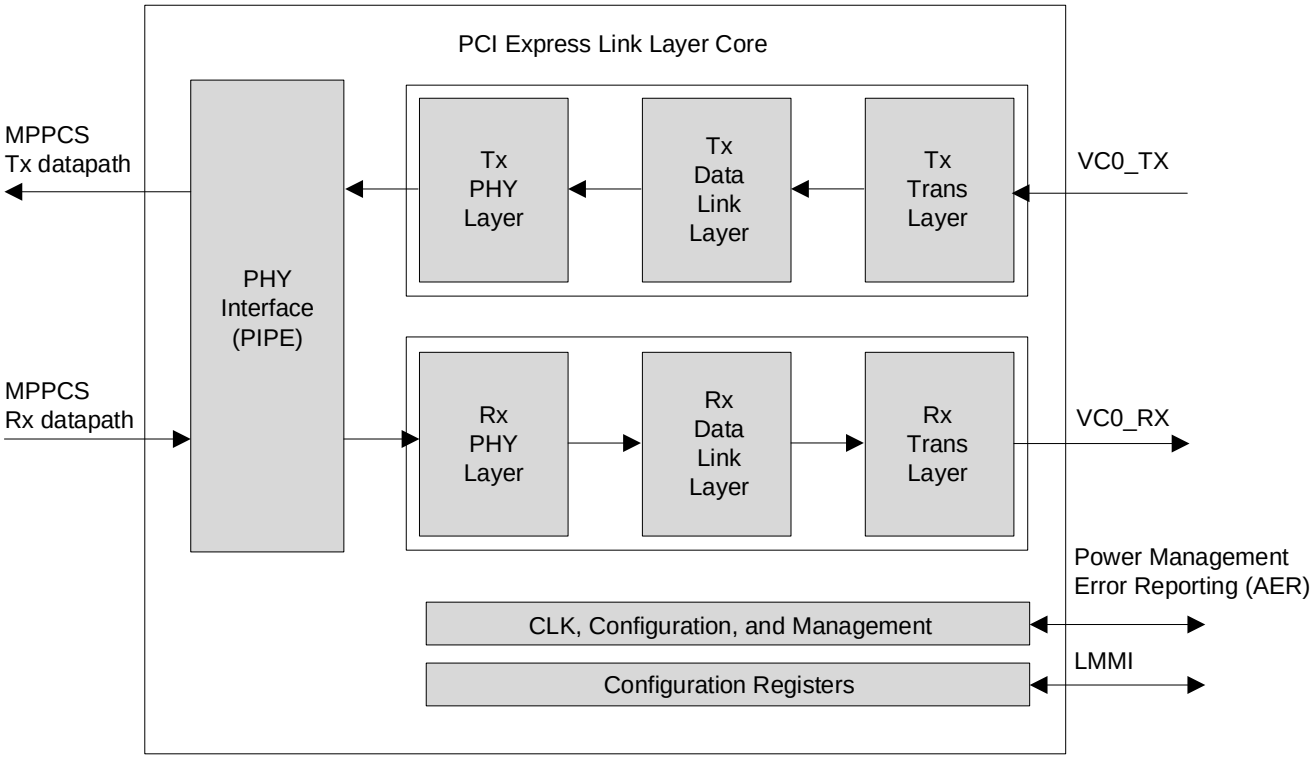

Figure 3.7. PCI Express Link Layer Functional Diagram

Each PMA Quad has four PMA channels. Every PMA Quad has two independent Tx PLL and every PMA channel has an independent CDR PLL. However, all PMA channels within the Quad share the same reference clock source. Each Quad requires its own reference clock, which can be sourced externally or from the FPGA internally. The reference clock from one PMA Quad is also routed out through the buffer and CML output driver to allow internal routing.

Figure 3.8 shows Lattice Nexus 2 device SERDES/PCS reference clock architecture. The Clock Tree block is designed for balancing the skew between different Quads based on one reference clock source, and the skew between different clock sources.

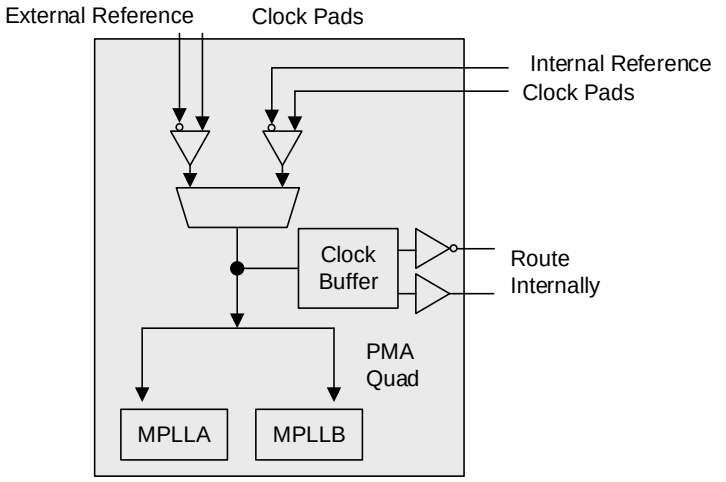

Figure 3.8. Lattice Nexus 2 SERDES Reference Clock Architecture

3.4. SERDES Block Signal Interface

For PCI Express Hard IP mode, the LMMI interface, TLP interface, and UCFG interface are accessible as shown in Figure 3.9.

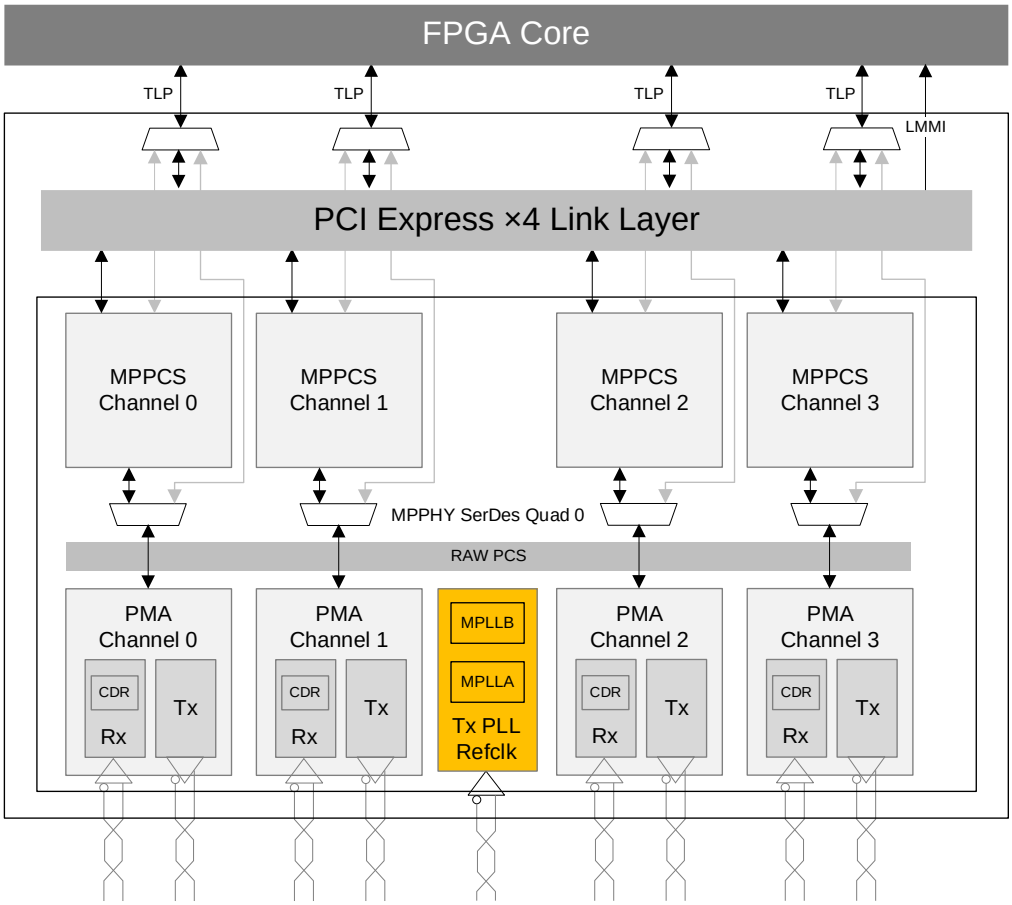

Figure 3.9. PCI Express Hard IP Mode

For use of other modes using MPPCS other than PCI Express Hard IP mode, the PCI Express ×4 Link Layer is bypassed. The LMMI interface is still accessible for each SERDES Quad as shown in Figure 3.10.

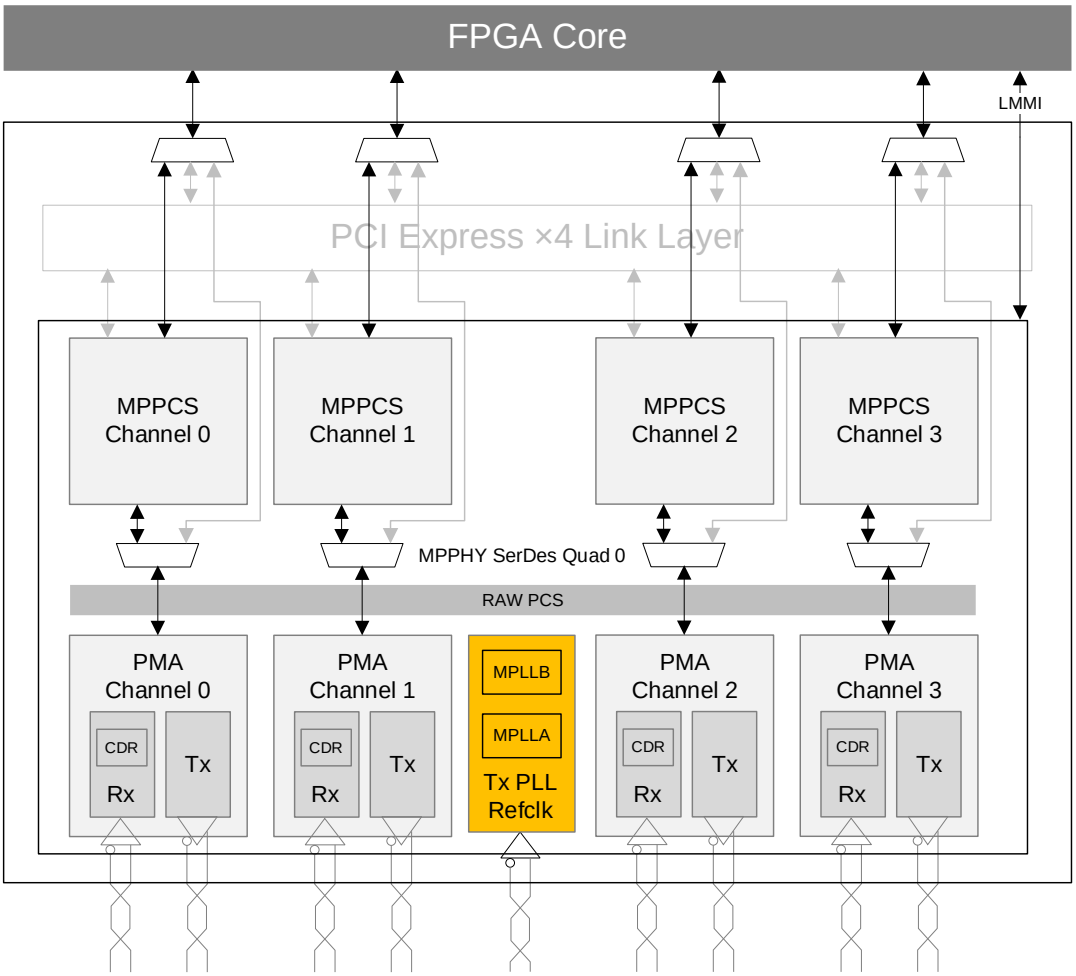

Figure 3.10. Other Modes using MPPCS bypassing PCI Express ×4 Link Layer

For MPPCS Bypass mode, the MPPCS is bypassed and PCI Express ×4 Link Layer is bypassed. The LMMI interface is still accessible for each SERDES Quad as shown in Figure 3.11.

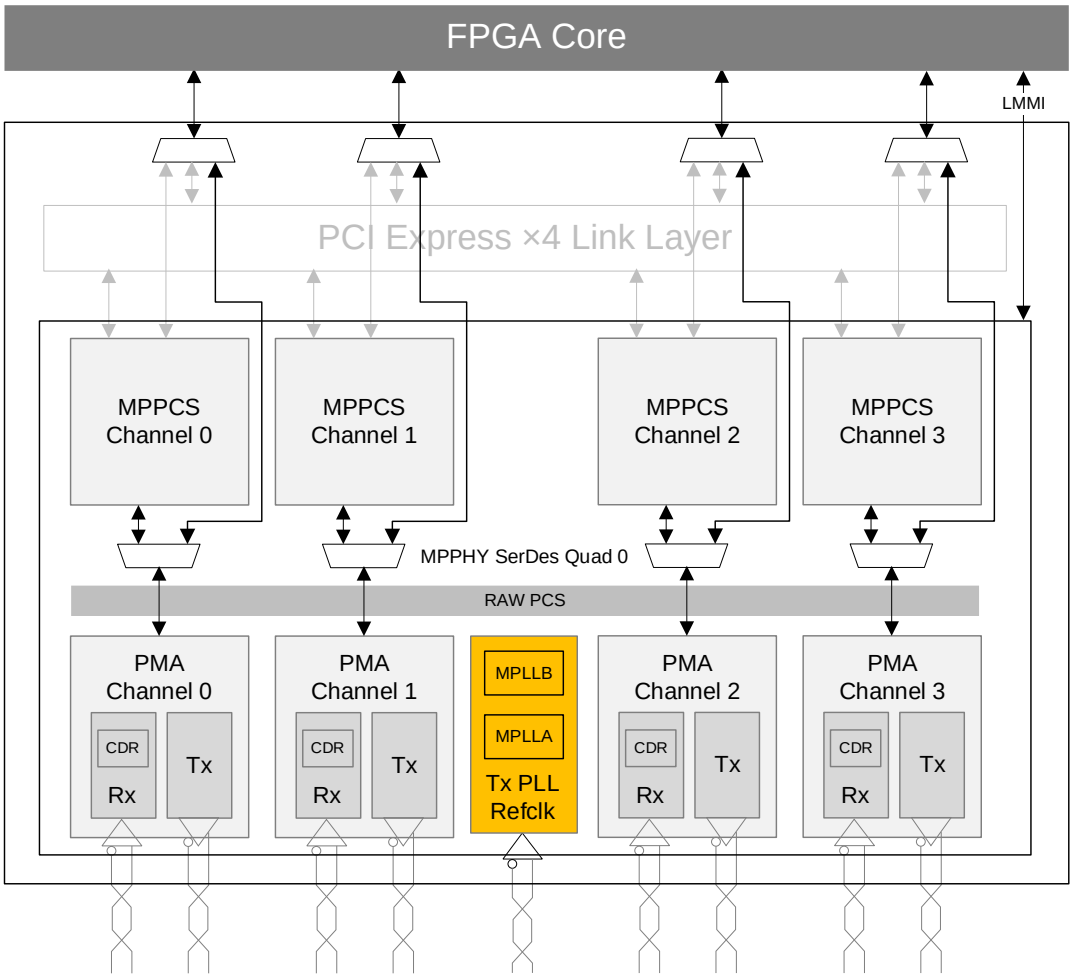

Figure 3.11. MPPCS Bypass Mode

3.5. Signal Descriptions

TBD

3.6. Control and Status Signals

Table 3.5 and Table 3.6 describe the control and status signals for 8B/10B protocols. The 8b10b protocol is supported with 4 control bits per 8 data bits. These are mapped to 2 control bits and 2 sideband bits per octet. Table 3.7 describes the control and status signals for 64B/66B and 128B/130B protocols.

Table 3.5. Tx Control and Data Signals per Octet for 8b10b Protocols

Sideband[1]	Sideband[0]	Control[1]	Control[0]	8b data	Description
0	0	0	0	tx_data	Data symbol with calculated disparity
0	0	0	1	tx_data	K-symbol with calculated disparity
0	0	1	0	tx_data	Data symbol with inverted disparity
0	0	1	1	tx_data	K-symbol with inverted disparity
0	1	0	0	tx_data	Data symbol with forced positive disparity
0	1	0	1	tx_data	K-symbol with forced positive disparity
0	1	1	0	tx_data	Data symbol with forced negative disparity
0	1	1	1	tx_data	K-symbol with forced negative disparity
1	0	tx_disp	tx_datak	tx_data	Symbol modified per disparity function such as 10GBaseR *cordisp* bit
1	1	tx_data[9]	tx_data[8]	tx_data[7:0]	10-bit symbol (encoder bypass)

Table 3.6. Rx Control and Data Signals per Octet for 8b10b Protocols

Sideband[1]	Sideband[0]	Control[1]	Control[0]	8b data	Description
0	0	0	rx_datak	rx_data	Symbol valid
0	0	1	rx_datak	rx_data	SKP added
0	1	0	rx_datak	rx_data	SKP removed
0	1	1	rx_datak	rx_data	Receiver detected
1	0	0	rx_datak	rx_data	Decode error (and possible disparity error)
1	0	1	rx_datak	rx_data	Elastic buffer overflow
1	1	0	rx_datak	rx_data	Elastic buffer underflow
1	1	1	rx_datak	rx_data	Disparity error on correctly decoded symbol
0	0	rx_data[9]	rx_data[8]	rx_data[7:0]	10-bit symbol (decoder bypass)
0	1	rx_data[9]	rx_data[8]	rx_data[7:0]	Receiver detected (decoder bypass)
1	0	rx_data[9]	rx_data[8]	rx_data[7:0]	Elastic buffer overflow (decoder bypass)
1	1	rx_data[9]	rx_data[8]	rx_data[7:0]	Elastic buffer underflow (decoder bypass)

Table 3.7. Control and Status Signals Functions (64B/66B and 128B/130B Protocols)

Sideband	Control[1]	Control[0] (Direct Mode)	Control[0] (Codec Mode)	8b data
[167:160]	[36] = block start [39:37] = reserved	[33:32] = sync header for PCIe, 802.3 and others [35:34] = extra sync header	[32] = K indicator of 1st octet	[7:0] = 1st octet
			[33] = K indicator of 2nd octet	[15:8] =2nd octet
			[34] = K indicator of 3rd octet	[23:16] =3rd octet
			[35] = K indicator of 4th octet	[31:24] =4th octet
[175:168]	[76] = block start [79:77] = reserved	[73:72] = sync header for PCIe, 802.3 and others [75:74] = extra sync header	[72] = K indicator of 1st octet	[47:40] = 1st octet
			[73] = K indicator of 2nd octet	[55: 48] =2nd octet
			[74] = K indicator of 3rd octet	[63:56] =3rd octet
			[75] = K indicator of 4th octet	[71:64] =4th octet
[183:176]	[116] = block start [119:117] = reserved	[113:112] = sync header for PCIe, 802.3 and others [115:114] = extra sync header	[112] = K indicator of 1st octet	[87:80] = 1st octet
			[113] = K indicator of 2nd octet	[95: 88] =2nd octet
			[114] = K indicator of 3rd octet	[103:96] =3rd octet
			[115] = K indicator of 4th octet	[111:104] =4th octet
[191:184]	[156] = block start [159:157] = reserved	[153:152] = sync header for PCIe, 802.3 and others [155:154] = extra sync header	[152] = K indicator of 1st octet	[127:120] = 1st octet
			[153] = K indicator of 2nd octet	[135: 128] =2nd octet
			[154] = K indicator of 3rd octet	[143:136] =3rd octet
			[155] = K indicator of 4th octet	[151:144] =4th octet

4. PHY SERDES Functional Description

Lattice Nexus 2 devices have one to two Quads of PHY SERDES. Each Quad, in turn, supports four independent full-duplex data channels. A single channel can support a data link, and each Quad can support up to four such channels. The SERDES CDR PLLs and Tx PLLs support data rates that cover a wide range of industry standard protocols.

4.1. MPPMA

Figure 4.1 shows a simplified functional block diagram of the MPPMA macro. Each of the MPPMA macros includes Tx, Rx, Support and Digital Control blocks.

The Tx block receives a transition-encoded data synchronous with a Tx clock, serializes it into a single stream of differential transmitted data and transmits to the lane. The transmitter supports multi-level output driver and multi-level transition emphasis. A FIR based Tx equalizer is implemented to support de-emphasis and pre-shoot. The Tx blocks also includes a Receiver Detection circuit for PCIe Express protocol and programmable on chip termination.

The Rx block receives differential serial data from a pair of external pads. The incoming serial data to the Rx input is attenuated by the transmission medium and a CTLE is first used to compensate for channel loss. After a CTLE, a DFE is implemented to optimize the data eye further for high-speed data. The Rx output signal is passed to the CDR circuit, which recovers the embedded clock in the data. A dedicated VCO is the source of the recovering clock. The CDR loop adjusts the VCO clock frequency and phase until the frequency matches incoming data and the phase is aligned with incoming data. The recovered clock is used to retime received data and send it to the deserializer, which produces parallel data and a parallel data clock for the relevant PCS lane.

The Rx block also includes an eye monitor to map the post-equalized eye density. Eye X and Y coordinates can be controlled, and the error density can be calculated at each coordinate.

The support blocks within the MPPMA contains two Tx PLL to generate the required Tx link clocks and high frequency internal clocks from the provided reference clock source.

The MPPMA also requires an external off-chip reference resistor for resistor tuning which is required for Tx and Rx on chip termination.

The MPPMA is digitally controlled by the raw PCS which performs the calibration, equalization adaptation, provides register access to the MPPHY, and enables the BIST features such as eye monitor and loopbacks.

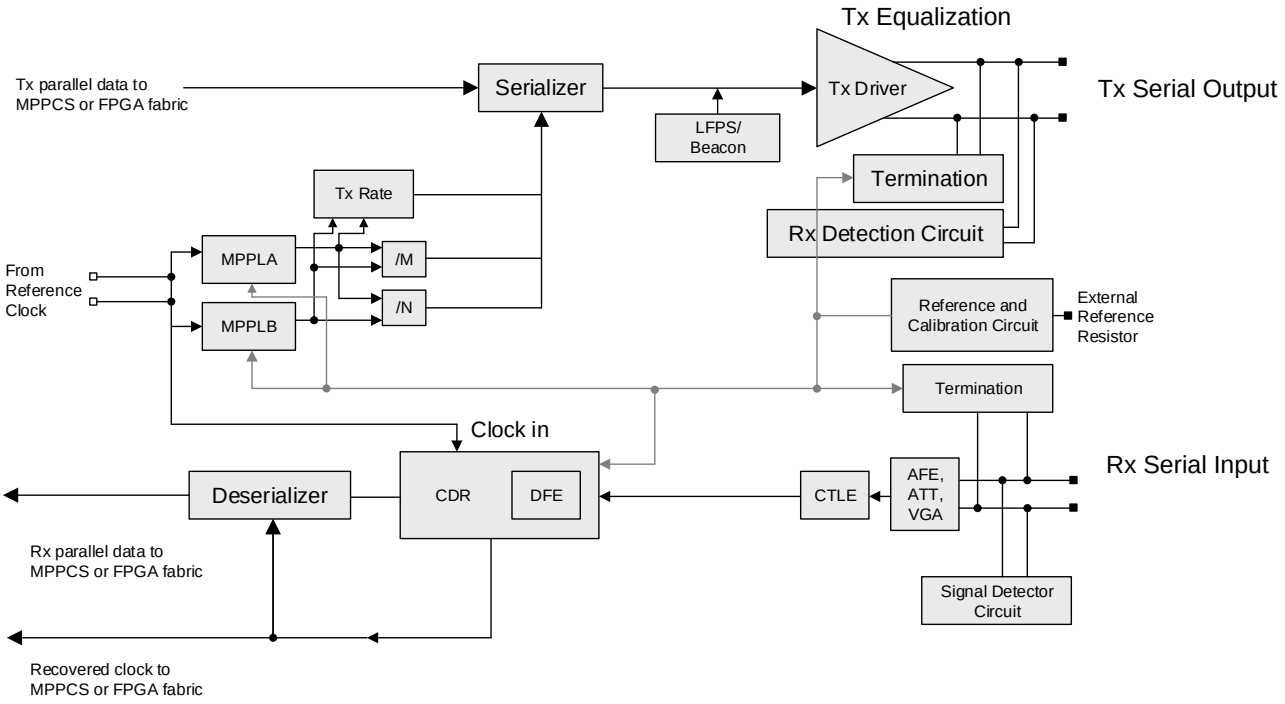

Figure 4.1. Simplified Block Diagram of the Single Lane MPPMA

4.1.1. AC Coupling

For most serial protocols, each PMA channel needs to be AC coupled externally. Suitable values for AC coupling capacitors must be used to maximize link signal quality and conform to respective standards specifications.

Take PCI Express as example, the AC coupling capacitors should be placed near Tx side, as shown in Figure 4.2.

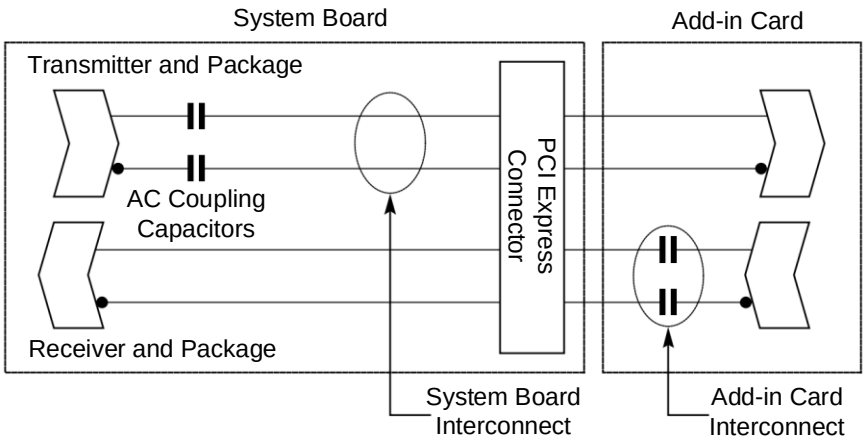

Figure 4.2. PCI Express AC Coupling Capacitors Location

4.1.2. Rx Block Diagram within MPPMA

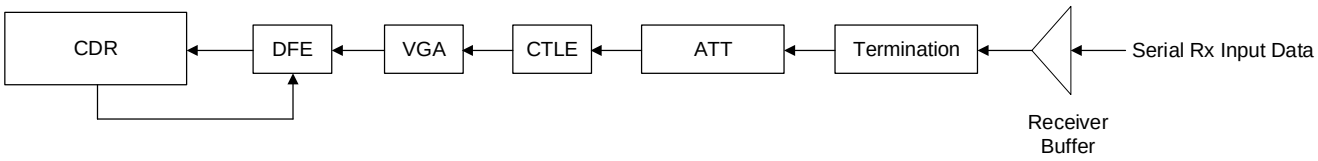

Figure 4.3. Rx Block Diagram within MPPMA

The Rx block within the MPPMA contains analogue blocks. The Receiver Buffer is terminated by the Termination block to ensure minimal impedance mismatch between the physical medium and the Rx PMA. The Termination value is calibrated by the PHY by referencing to external off chip reference resistors. The Rx data is then attenuated to acceptable levels for the equalization circuit. The equalization circuit consist of an adaptive continuous time linear equalization (CTLE), two variable gain amplifiers (VGA), and a 5-tap continuously adaptive decision feedback equalizer (DFE). The CTLE circuit boost the high frequency components of the Rx data to compensate for intersymbol interference (ISI) losses. The VGA improves the gain control range. The DFE performs non-linear equalization before the Rx data is sent to the Clock and Data Recovery (CDR) circuit and De-serializer.

4.1.3. Rx Clock and Data Recovery

The CDR circuit is used to recover the slow or parallel clock from received data. The CDR consist of a Voltage Controlled Oscillator (VCO), Phase Detector circuit, and loop filters as shown in Figure 4.4. The CDR VCO generates an Rx clock aligned with the incoming data.

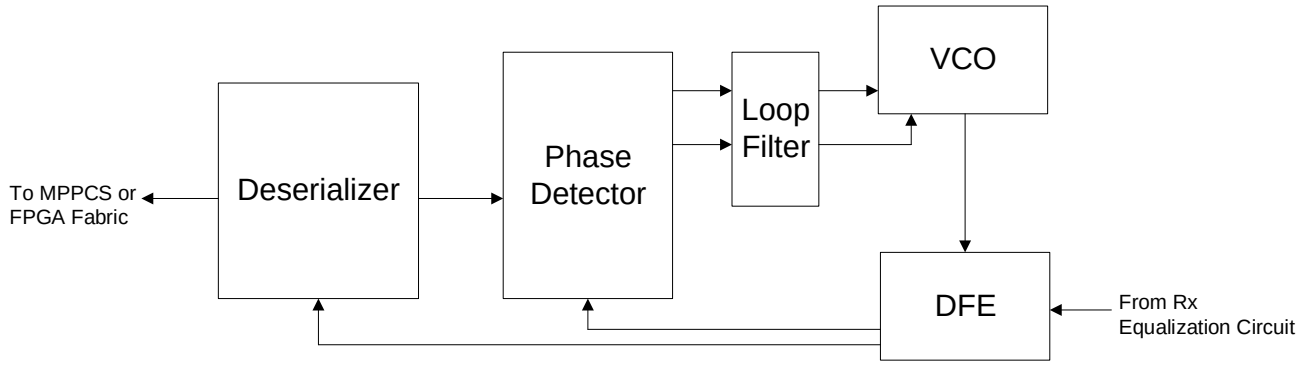

Figure 4.4. CDR Blocks

4.1.4. Rx Deserializer

The deserializer converts the serial data to parallel data to be sent to the Rx datapath of the MPPCS or FPGA fabric. The deserializer uses the recovered clock to sample the parallel data. The deserializer supports an 8, 16, 10, or 20-bit interface deferring for different data rates.

4.1.5. Rx Eye Monitor

The PHY consists of an eye monitor circuit which can provide data to produce a plot of time and voltage axis resembling an eye diagram plot. The eye monitor can be plotted with live traffic as seen after the Attenuation Circuit and after the VGA circuit.

4.1.6. Rx Rate Control

The VC0 within the CDR generates a range of high-speed clocks. The generated high-speed clocks are used by the CDR to control and align the frequency and phase of the data received by the Rx. Hence, this enables the Rx to support flexible data rate control to reliably receive incoming data with rates from 1.25 Gbps to a maximum of 16 Gbps.

4.1.7. Rx Calibration

The PHY calibrates the equalization circuit to minimize the variation or offset. The calibration is run upon startup after the PHY is brought out of reset. Part of the PHY calibrations is also set to run continuously during operation. The PHY performs a start-up calibration after the Rx is powered up. The calibration ensures offset of CTLE, VGA, attenuation circuit, DFE, and CDR blocks is managed within the operation limits.

4.2. Raw PCS

The Raw PCS block digitally manages the MPPMA. The Raw PCS implements the following:

- Power-up calibration algorithms for the PMA analog front-end, including Rx AFE offset cancellation, Rx slicer calibration, MPLL calibrations, and Rx VCO calibrations.
- Controls for the Rx equalization adaptation routines in the MPPMA for modes when adaptation is enabled.
- MPLL power-up and power-down control.
- PMA register access.
- Programmable Rx CDR unlock detector.

4.3. MPPCS Architecture

Lattice Nexus 2 Multi-Protocol PCS is designed for popular serial protocols such as PCI Express, Ethernet, SLVS-EC, CoaXpress, DP/eDP, CPRI/eCPRI, and JESD204B. The supported standards vary between Lattice Nexus 2 families. Refer to the Lattice Nexus 2 Platform – Overview Data Sheet (FPGA-DS-02122) for more details.

MPPCS can be configured as Generic 8B/10B mode or Generic 64b/66b or 128b/130b mode for user-defined serial protocols other than those listed in Table 3.1. The datapath width is programmable to 8, 10, 16, 20, 32, or 40 bits. Figure 4.5 and Figure 4.6 show the simplified block diagrams of the MPPCS within one lane width for both Tx and Rx datapaths.

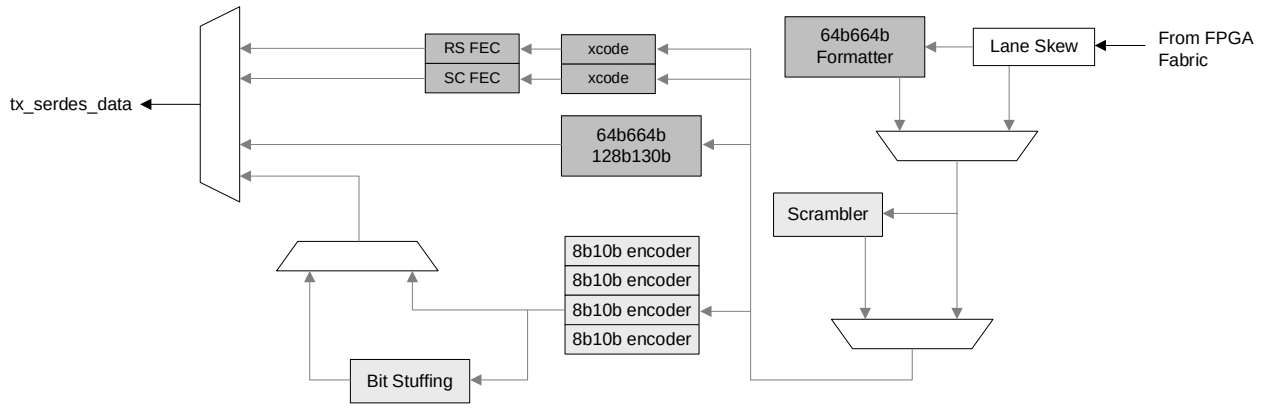

Figure 4.5. MPPCS Tx Datapath Block Diagram for One Lane

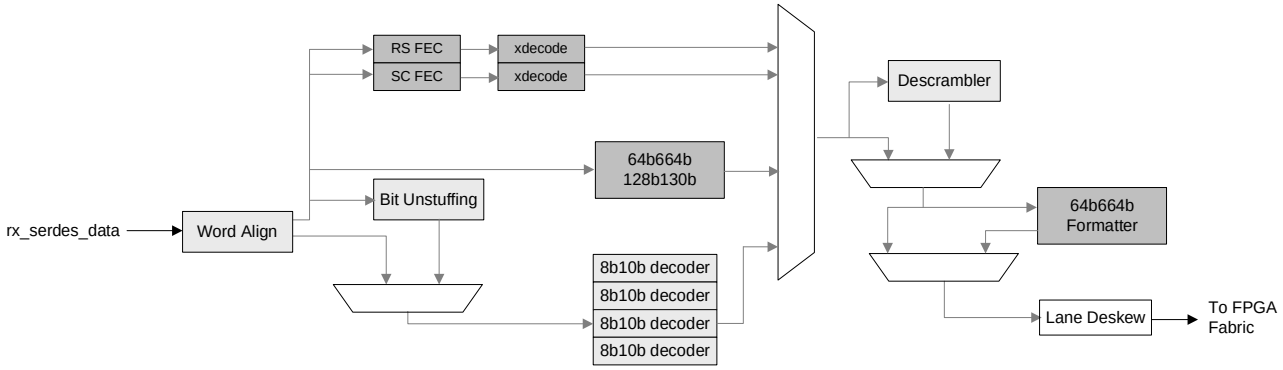

Figure 4.6. MPPCS Rx Datapath Block Diagram for One Lane

Four MPPCS channels can work independently or together to compose a multi-lane link. The internal functional block diagram of the channel is illustrated in Figure 4.5 and Figure 4.6. The Rx and Tx datapaths are almost symmetrical to each other with additional symbol alignment blocks on the Rx datapath to identify the block or word boundary.

The MPPCS Tx datapath functional blocks are listed and described in Table 4.1.

Table 4.1. Functional Blocks in MPPCS Tx Datapath

Functional Block	Description
Lane Skew	This function is supported by an elastic buffer with controllable latency. This function serves as a basic FIFO for clock-domain crossing between lane-specific clocks. This function also provides the capability for intentional skew or de-skew between lanes. In addition, this function supports clock tolerance compensation. This block can be optionally bypassed.
Scrambler	This function is supported by a programmable multi-mode LFSR of the order 64. This block can be optionally bypassed.
Block Encoder	The encoder block is instantiated four times in each lane to support a lane width of up to 40 bits. Two types of encoder is supported, which is 8B10B encoder or programmable 64b/66b or 128B/130B encoder.
Bit Stuffing	This function is used to support lower data rates.
Forward Error Correction (FEC)	This function is only supported for the operations at the highest data rate. This function is supported by a programmable multi-mode LFSR of the order 32.

The MPPCS Rx datapath functional blocks are listed and described in Table 4.2.

Table 4.2. Functional Blocks in MPPCS Rx Datapath

Functional Block	Description
Word Aligner	This function is to determine the word boundary of the data stream, data is compared with a fixed pattern of every possible alignment position equal to the data width. Each data bit is compared with a corresponding pattern bit. The alignment scores a hit when every data bit at a given alignment position matches the pattern bit or the mask bit is set at a particular position. For every alignment position, a counter increments when a hit is scored. An arbiter outputs the alignment position with the highest score. You can configure your own word aligner pattern and state machine. This block can be optionally bypassed.
Bit Unstuffing	This function is used to support lower data rates.
Forward Error Correction (FEC)	Can be set using FEC Mode in the Receiver Group. It is only supported for the operations at the highest data rate. This function is supported by a programmable multi-mode LFSR of the order 32.
Block Decoder	The decoder block is instantiated four times in each lane to support a lane width of up to 40 bits. Two types of decoders are supported which is 8B10B decoder or programmable 64b/66b or 128B/130B decoder.
Descrambler	This function is supported by a programmable multi-mode LFSR of the order 64. This block can be optionally bypassed.
Lane De-Skew	This function is supported by an elastic buffer with controllable latency.

5. Clocks

5.1. MPPHY Clock Domain

There are three clock domains on the Tx and Rx datapath which are:

- tx_serdes_clk and rx_serdes_clk, which are clocks generated from the Tx PLL and CDR divided using the clock dividers.
- tx_pipe_clk and rx_pipe_clk, which are PCS block and link layer clocks.
- tx_in_clk and rx_in_clk, which are fabric interface clocks.

Figure 5.1 shows the clock domain within the MPPHY.

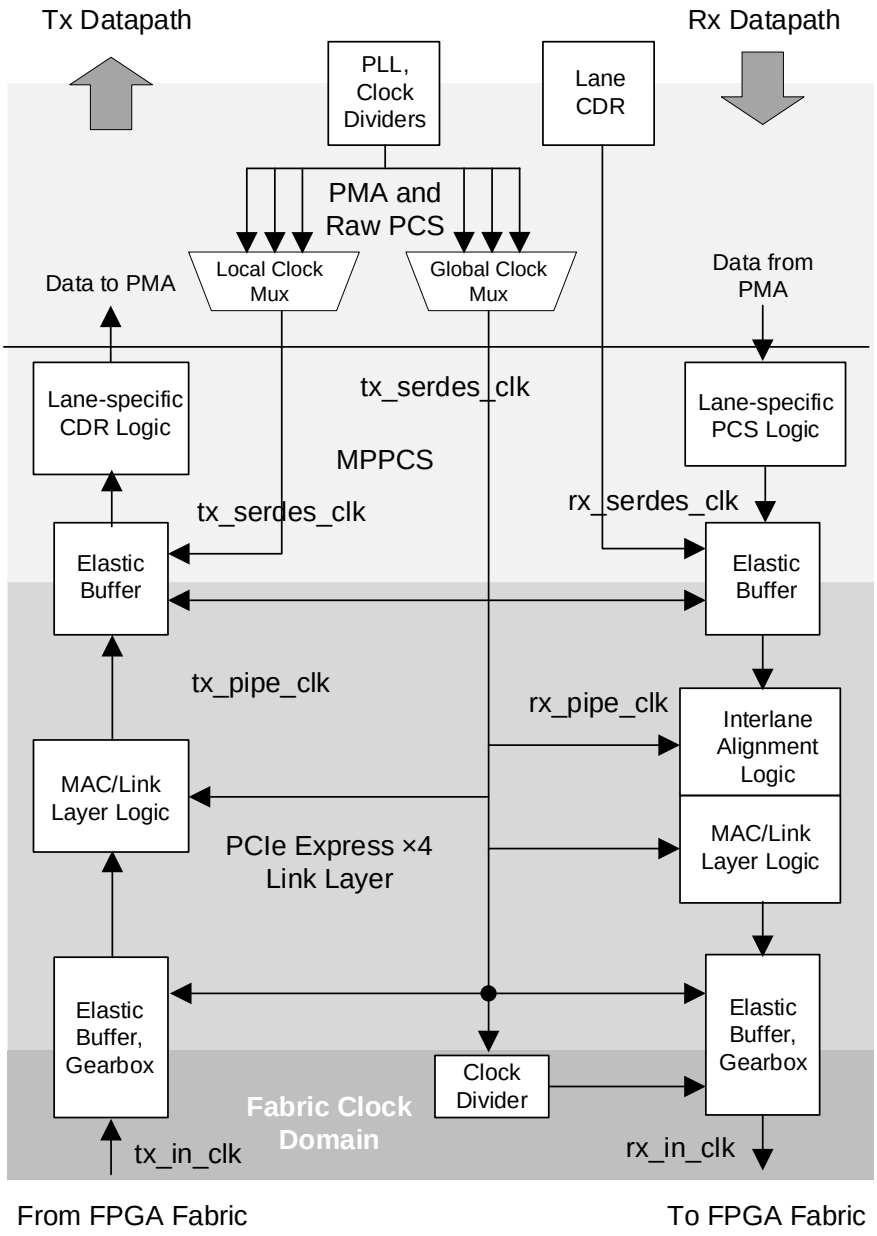

Figure 5.1. MPPHY Clock Domain

5.1.1. MPPHY Clock Scheme

This section describes the clocking scheme for high-speed operations of the MPPHY. Each Quad has two PLLS which are MPLLA and MPLLB. Each PLL can generate eight possible clocks to be used at the PMA/PCS interface. Figure 5.2 shows the clock scheme across two Quads.

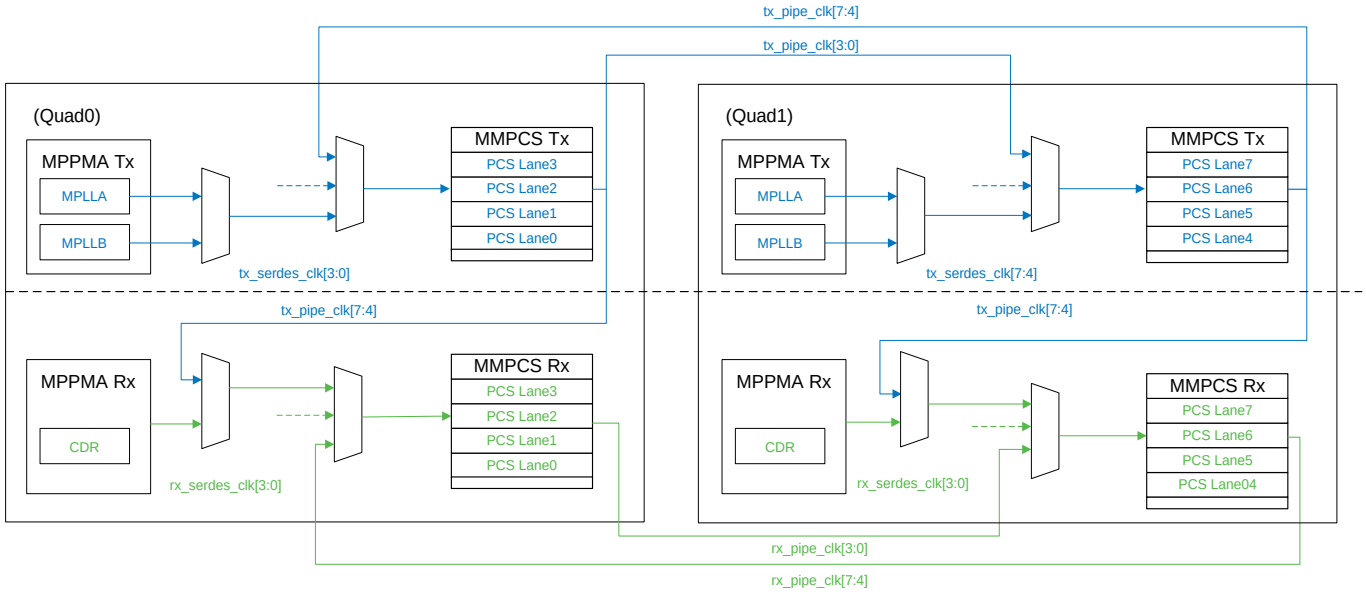

Figure 5.2. Clock Scheme across Two Quads (LN2-CT-16 and LN2-CT-20 Devices)

5.2. Clock Frequency

This section lists the recommended reference clock frequency, the PLL frequency, and encoding and divide ratio for some of the supported protocols. The PLL settings are the same for Tx PLL and Rx CDR, considering that the architecture of Tx PLL and Rx CDR is similar. The supported standards vary between Lattice Nexus 2 families. Refer to the Lattice Nexus 2 Platform – Overview Data Sheet (FPGA-DS-02122) for more details.

Table 5.1. Clock Frequency

Protocol	Data Rate (Gbps)	Encoding	Reference Clock (MHz)	PLL Freq (GHz)	Divide Ratio
PCIe Gen1	2.5	8b10b	100	5	50
PCIe Gen2	5	8b10b	100	5	50
PCIe Gen3	8	128b130b	100	8	80
PCIe Gen4	16	128b130b	100	8	80
Ethernet	1.25	8b10b	156.25	—	—
	3.125	8b10b	156.25	6.25	40
	5	8b10b	—	—	—
	6.25	8b10b	156.25	6.25	40
	10.3125	64b66b	156.25	6.375	33
JESD204B	3.125	8b10b	125	6.25	50
	6.375	8b10b	125	6.375	51
	12.25	64b66b	125	6.25	50
CPRI	1.2288	8b10b	122.88	4.9152	40
	2.4576	8b10b	122.88	4.9152	40
	3.072	8b10b	122.88	6.144	50
	4.915	8b10b	122.88	4.9152	40
	6.144	8b10b	122.88	6.144	50
	8.11008	8b10b	122.88	8.11008	66
	9.8304	8b10b	122.88	4.9152	40
	10.1376	64b66b	92.16	5.0688	55
	12.16512	64b66b	92.16	6.08256	66

6. SERDES Equalization

The interconnection between the transmitter and receiver device acts as a filter at typical baud rates and distorts the serial data signal to varying extents. Figure 6.1 shows the signal distortion for the typical backplane applications. The signal out from the transmitter side is a clean digital signal, but the signal waveform is significantly distorted at the receiver side. The frequency response function for this interconnection is a low-pass filter, considering the insertion loss increases with the increase of the frequency of signal. Signal distortion occurs because the signal baud rate is above the cut-off frequency for this low-pass filter.

As data rates get higher, the Unit Interval (UI, or bit time) becomes smaller. As a result, there is a higher possibility of one bit impacting the value of the subsequent bit across time. When a signal is held at the same voltage for several bit times, and when sending several bits in a row of the same polarity, the channel has more time to approach the target voltage. The resulting higher voltage makes it difficult to change to the opposite value within the required time when the polarity does change. AC coupling sometimes exacerbates this phenomenon, considering the charging effect of capacitance. What is more, the interconnection low-pass filter makes it become more difficult for the receiver to recognize the real value of the signal, because the high frequency component of the repeating value signal is less than the continuous flip signal. This problem of previous bits affecting subsequent bits is referred to as Inter-Symbol Interference (ISI).

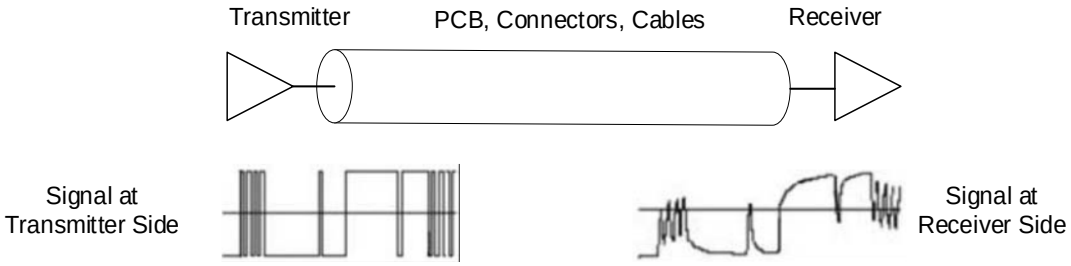

Figure 6.1. Signal Distortion for Typical Backplane Application

To solve these Signal Integrity (SI) problems, data signal should be equalized at the transmitter side. The equalization at the receiver side is also necessary sometimes even though equalization at the transmitter side is always required for high-speed digital signals. De-emphasis and pre-emphasis are common equalization technologies used by most high-speed serial applications. De-emphasis reduces the voltage for repeated bits in a data bitstream. Pre-emphasis is similar to de-emphasis, which intentionally overdrives at the first bit for repeated bits in a data bitstream. Both de-emphasis and pre-emphasis can be implemented by 3-tap Finite Impulse Response (FIR). In some documents, this FIR is named as Feed Forward Equalizer (FFE). For data rate higher than 6 Gbps, the 3-tap FIR based equalizer is usually used by a transmitter. The Continuous-Time Linear Equalization (CTLE) and Decision Feedback Equalization (DFE) are common implementations in receiver equalizer designs.

Figure 6.2 the typical backplane application with Tx equalizer. Tx equalizer distorts the signal with de-emphasis technology, such that the resulting signal at the receiver input is a clean waveform and is easier to be recognized.

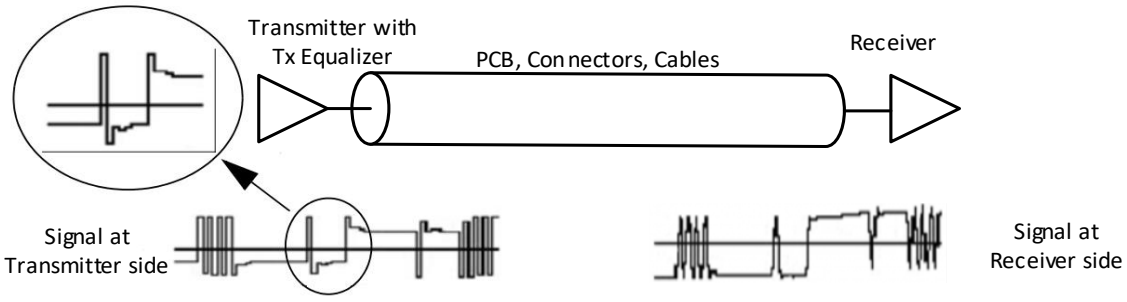

Figure 6.2. Typical Backplane Application with Tx Equalizer

Figure 6.3 shows the typical backplane application with Rx equalizer. Despite the signal distortion at the input of the receiver, Rx equalizer corrects for the distortion and produces a clean waveform. CTLE reduces the low frequency component from the received signal, which attenuates by a lower amount on the transmission line. The high frequency component of the received signal can also be amplified by CTLE sometimes. The DFE is similar to the Tx equalizer, usually implemented by multi-tap FIR. The combination of CTLE and DFE tolerates more channel loss than either equalizer alone, when they have been tuned properly.

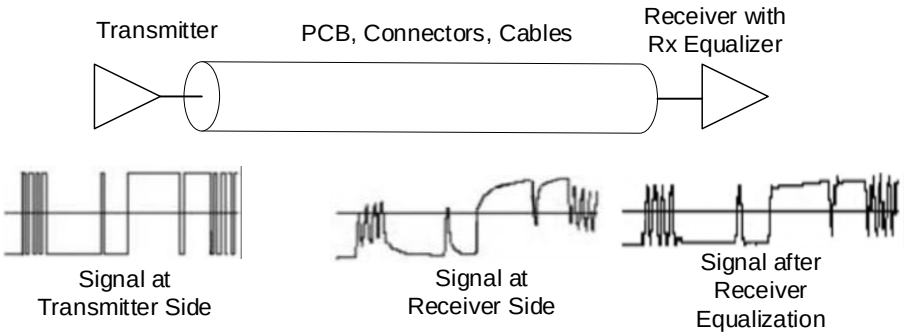

Figure 6.3. Typical Backplane Application with Rx Equalizer

Lattice Nexus 2 device implements both Tx equalizer and Rx equalizer for multiple protocols supporting. Tx equalizer is based on 3-tap FIR, while the Rx equalizer is based on CTLE and 1-tap DFE. Below sections descript the detailed usage of Lattice Nexus 2 Tx equalizer and Rx equalizer.

6.1. Tx Equalization

Figure 6.4 shows the block diagram of transmit equalizer. Both pre-cursor ratio (C_{-1}) and post-cursor ratio (C_{+1}) are negative and $|C_{-1}| + C_0 + |C_{+1}| = 1,\ |C_{-1}| + |C_{+1}| < 0.5$

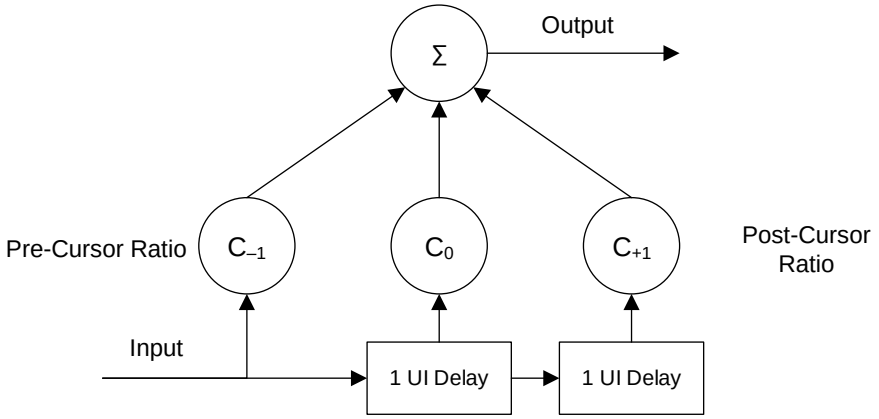

Figure 6.4. Transmit Equalizer Block Diagram

Table 6.1. Description of Tx Voltage Levels

Tx Voltage Levels	Definition	Normalized Amplitude
Va	The amplitude of the first bit of a repeated bitstream (given 00001111 pattern).	$1 + 2 \times C_{-1}$
Vb	The amplitude between the first and the last bits in a repeated bitstream.	$1 + 2 \times C_{-1} + 2 \times C_{+1}$
Vc	The amplitude of the last bit of a repeated bitstream (given 00001111 pattern).	$1 + 2 \times C_{+1}$
Vd	Full-scale amplitude (given 01010101 pattern).	1

Given a differential signal with symmetric swings above and below 0, it can be seen from Figure 6.5 that there are four possible output values in each positive or negative directions. Table 6.1 shows the positive normalized voltage levels.

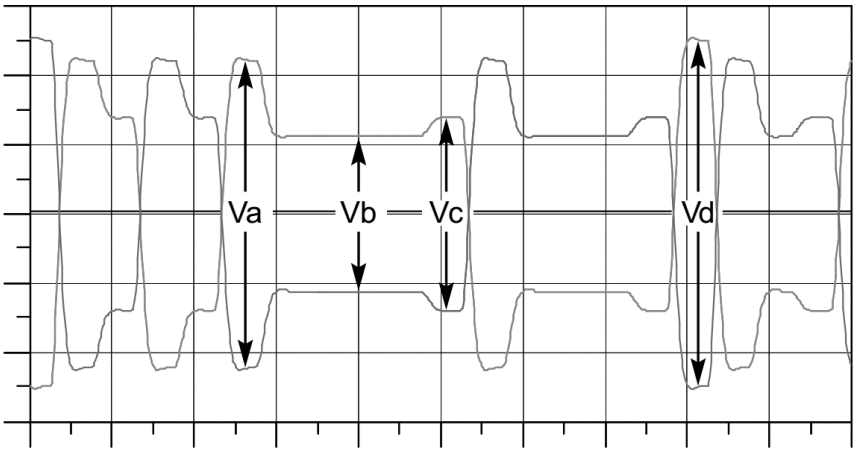

Figure 6.5. Definition of Tx Voltage Levels

The de-emphasis is defined as

$$\text{Deemphasis (dB)} = 20\log_{10}(V_b/V_a)\ \text{(dB)}.$$

Moreover, the pre-shoot is defined as

$$\text{Preshoot(dB)} = 20\log_{10}(V_c/V_b)\ \text{(dB)}.$$

Table 6.2 shows the Tx preset ratios and corresponding coefficient values defined by PCI Express Specification. For protocols that only require de-emphasis feature, the pre-cursor ratio (C_{-1}) should be fixed as zero. For example, to achieve –3.5 dB de-emphasis, post-cursor ratio (C_{+1}) should be set as –0.167 and pre-cursor ratio (C_{-1}) should be set as 0.000.

Table 6.2. PCIe Tx Preset Ratios and Corresponding Coefficient Values

Preset #	Preshoot (dB)	De-emphasis (dB)	C_{-1}	C_{+1}	Va/Vd	Vb/Vd	Vc/Vd
P4	0.0	0.0	0.000	0.000	1.000	1.000	1.000
P1	0.0	–3.5	0.000	–0.167	1.000	0.668	0.668
P0	0.0	–6.0	0.000	–0.250	1.000	0.500	0.500
P9	3.5	0.0	–0.166	0.000	0.668	0.668	1.000
P8	3.5	–3.5	–0.125	–0.125	0.750	0.500	0.750
P7	3.5	–6.0	–0.100	–0.200	0.800	0.400	0.600
P5	1.9	0.0	–0.100	0.000	0.800	0.800	1.000
P6	2.5	0.0	–0.125	0.000	0.750	0.750	1.000
P3	0.0	–2.5	0.000	–0.125	1.000	0.750	0.750
P2	0.0	–4.4	0.000	–0.200	1.000	0.600	0.600

6.2. Rx Equalization

Lattice Nexus 2 device implements programmable single pole-zero Continuous Timer Linear Equalizer (CTLE) at the receiver side, followed by Decision Feedback Equalization (DFE) and circuitry to support adaptation of the CTLE and DFE. Figure 6.6 shows the block diagram of receive equalizer. CTLE comprises an active high pass filter that has the effect of amplifying higher frequency components that have been more severely attenuated by the interconnection or attenuating the lower frequency components to a greater degree than the higher frequency components. DFE comprises samplers offset by a programmable voltage value and a decision selection circuit that has the effect of varying the sample threshold voltage based on the sampled data stream.

The CTLE and DFE can be tuned to compensate for more severe inter-symbol interference (ISI) than either circuit performs alone.

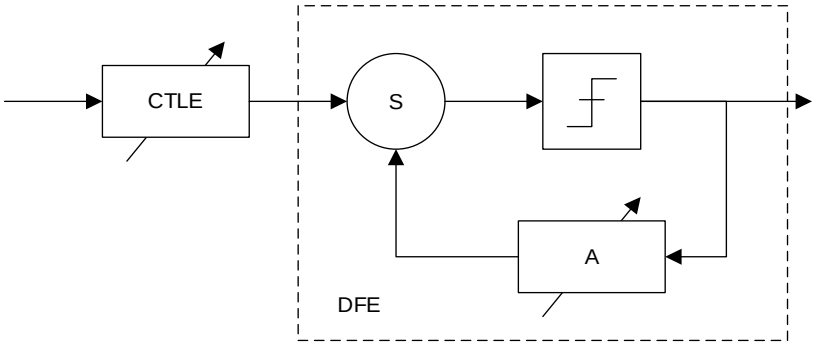

Figure 6.6. Receive Equalizer Block Diagram

The CTLE can be considered as a cascade of two frequency dependent equalizers. One primarily controls low frequency attenuation and the other primarily controls the boost at high frequency.

The gain A_V of a CTLE is defined as

$$A_V = 20 \times log_{10}(\frac{V_{OUT}}{V_{IN}})$$

7. SERDES Debug Capabilities

7.1. Loopback Mode

Lattice Nexus 2 SERDES supports three loopback modes for different test purposes:

- MPPMA Internal Tx to Rx Serial Loopback
- MPPMA Tx to Rx Parallel Loopback
- MPPMA Rx to Tx Parallel Loopback
- MPPCS Digital Loopback
- MPPCS SERDES Loopback

7.1.1. MPCS Loopback

Figure 7.1 shows the two loopback modes implemented by MPPCS. The SERDES Loopback loops back the output fabric Rx data of MPPCS back to the input fabric Tx data of MPPCS block. The Digital Loopback loops back the Tx data from MPPCS back to the input Rx data.

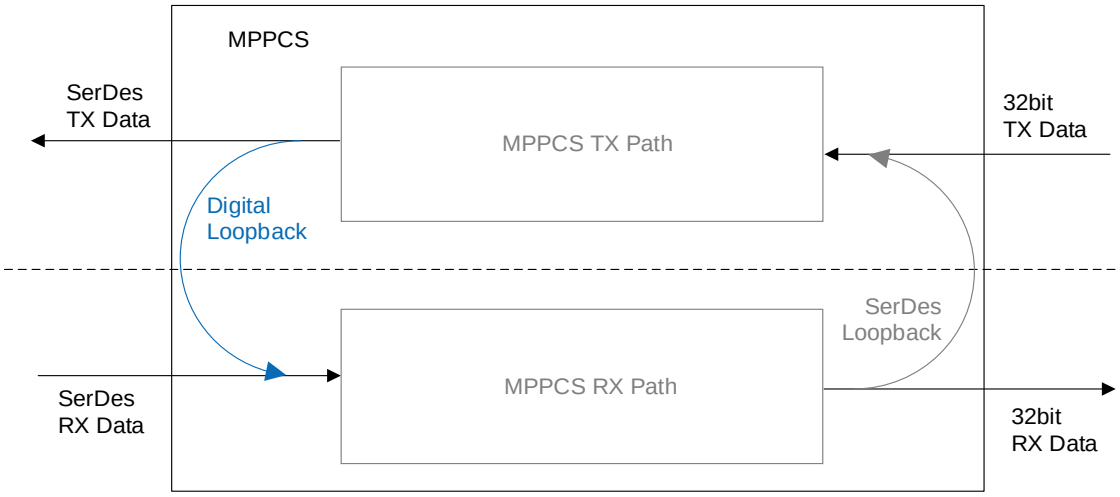

Figure 7.1. MPPCS Loopback Mode

7.1.2. MPPMA Loopback

The MPPMA block is equipped with several loopback modes for testing purposes. The Internal Tx to Rx Serial Loopback loops back the Tx serial data output into the Rx datapath inside the analog portion of the MPPMA. The Tx to Rx Parallel Loopback loops back the Tx parallel data into the Rx datapath at raw PCS or digital portion of the MPPMA. This loopback covers the entire Tx/Rx in the digital portion. The Rx to Tx Parallel Loopback loops back the Rx parallel data into Rx datapath at raw PCS or digital portion of the MPPMA. This loopback covers the entire datapath for both Tx and Rx in analog and digital. Figure 7.2 shows the block diagram MPPMA Loopback Modes.

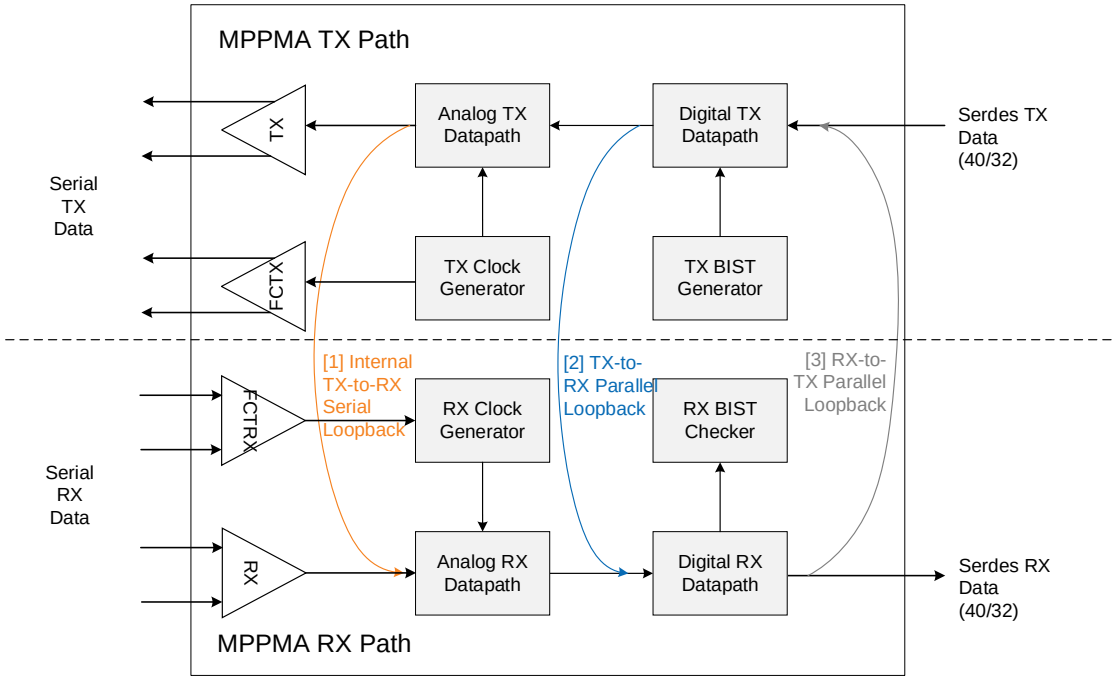

Figure 7.2. MPPMA Loopback

7.2. Loss of Lock

Both Tx PLLs and CDR PLLs have the loss-of-lock detectors for PMA debug purpose.

If the Tx PLL loses lock, the loss-of-lock for the PLL is asserted and remains asserted until the PLL reacquires lock state. When the PLL loses lock, it is likely to be caused by a reference clock problem.

For CDR PLLs, there are two steps to get the lock state:

- Frequency lock or lock with reference to reference clock, is a frequency lock operation whereby CDR PLL locks to the reference clock through the Phase Frequency Detector (PFD) operation. Sampling clock at the receiver is not aligned to the center of the data eye during this step.
- Phase lock or lock with reference to input bitstream, is a phase lock operation whereby CDR PLL acquires phase and small frequency deviation lock to the bitstream. Sampling clock at the receiver is aligned to the center of the data eye after this step. It is imperative that the bitstream be valid upon entering phase lock. There are two further steps for the phase lock:
  - Coarse phase lock, which has higher range of frequency acquisition (±5000 ppm of static frequency difference). This step is always a transient step before embarking on to fine phase lock.
  - Fine phase lock, which has a lower range of frequency acquisition (±300 ppm static frequency difference). Note that fine phase lock can acquire to spread spectrum modulated signals (dynamic frequency difference) when the slow-moving frequency of the input intersects the current frequency of CDR PLL. Thereafter, fine phase lock continues to track spread spectrum modulation barring abrupt frequency jumps requiring reacquisition in coarse phase lock.

7.3. Eye Monitor

Eye monitor is an on-chip scope to visualize post-equalization signal quality at SERDES/PMA Rx side.

8. SERDES Register Access

8.1. Register Access Bus

Each SERDES quad has an independent register access bus – the Lattice Memory Mapped Interface (LMMI). It has the following features:

- 16-bit width of write and read data.
- 16-bit address (offset).
- Support single byte read and write.
- Support burst read and write.
- Read-back data validity indication.
- Ready signal to support wait state.

When the LMMI controller is ready to request a transaction, it asserts lmmi_request_i and drives the request data (lmmi_wr_rdn_i, lmmi_offset_i, and lmmi_wdata_i) at the same time and starts sampling lmmi_ready_o on the positive edge clock. The LMMI controller holds the lmmi_request_i and the requested data constantly until the lmmi_ready_o is asserted. When both lmmi_request_i and lmmi_ready_o are asserted on the same positive clock edge, the LMMI target latches the requested data and starts transaction. Once the requested data has been latched, the LMMI controller can immediately start the next transaction request as its own discretion. If the LMMI target is ready to start another transaction in the next clock cycle, it holds lmmi_ready_o asserted. Otherwise, the LMMI target deasserts lmmi_ready_o until it is ready to accept a new transaction. Figure 8.1, Figure 8.2, and Figure 8.3 show the examples for different transactions.

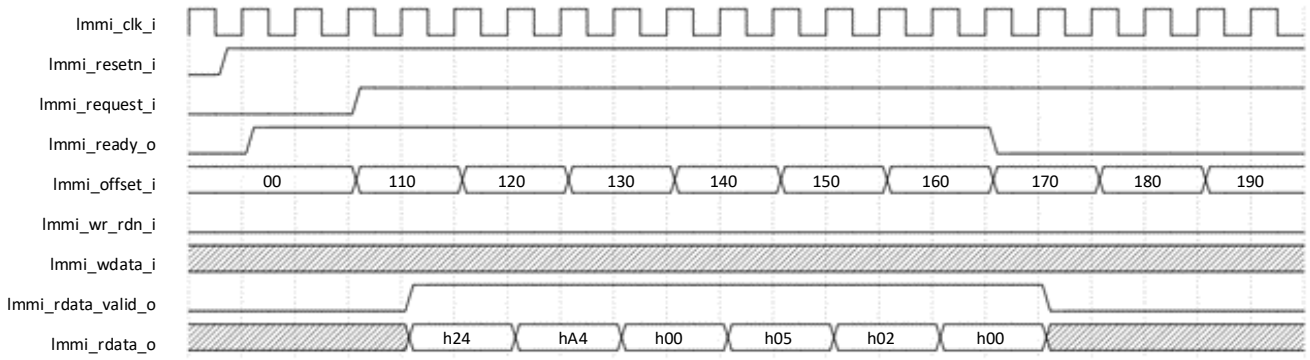

Figure 8.1. Burst Read Transaction

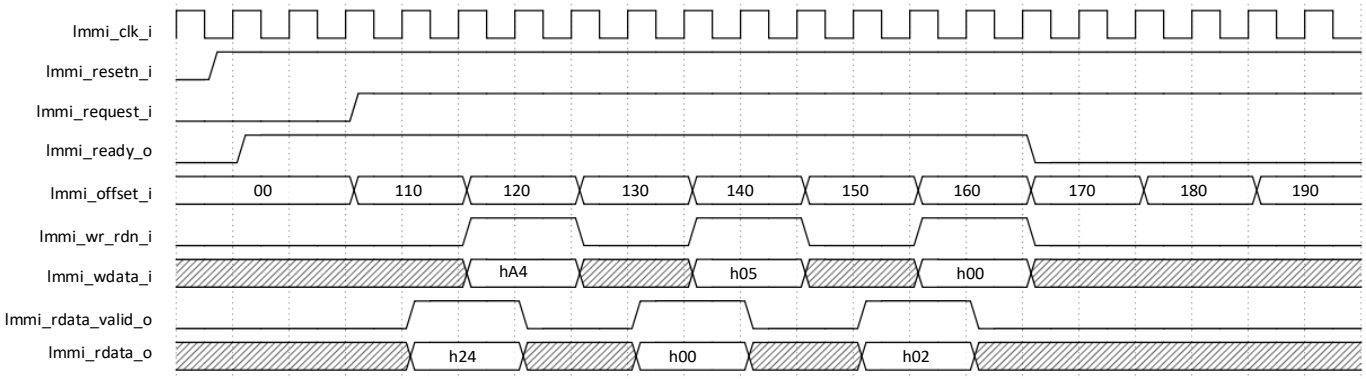

Figure 8.2. Back-to-Back Read and Write Transaction

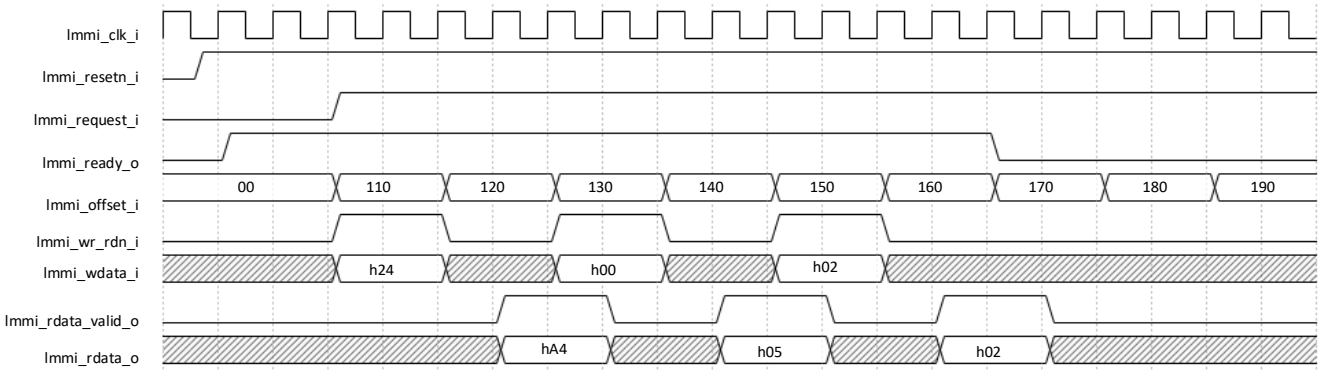

Figure 8.3. Back-to-Back Write and Read Transaction

9. Register Descriptions

9.1. Overview

Table 9.1 lists the address map in Word size, and Table 9.2 provides the access type definitions that are used throughout this document. Table 9.3 and Table 9.4 summarize available registers. A detailed information of each registers is described in tables in subsequent sections.

For all MPCS registers:

- Use Word offset as address to read/write the 16-bit MPCS register
- (Word 0ffset × 2) is the Byte address for data [7:0]
- (Word 0ffset × 2) + 1 is the Byte address for data [15:8]

Table 9.1. MPP Register Address Map

MPP Details	LMMI Address (Word Size)	
	Start Address	End Address
Lane 0 Control	0x5000	0x50FF
Lane 1 Control	0x5100	0x51FF
Lane 2 Control	0x5200	0x52FF
Lane 3 Control	0x5300	0x53FF

Table 9.2. Access Type Definitions

Access Type	Behavior on Read Access	Behavior on Write Access
RO	Returns register value.	Ignores write access.
WO	Returns *0*.	Updates register value.
RW	Returns register value.	Updates register value.
RW1C	Returns register value.	Writing *1'b1* on a register bit clears the bit to *1'b0*. Writing *1'b0* on a register bit is ignored.
CR	Clears on read.	Clears register value on read.
WOP	Returns *0*.	Write Only Pulse.

Table 9.3. MPCS Registers Summary (Offset Address is in Byte)

LMMI Word Offset	LMMI Byte Offset	Register Name	Access Type	Description
8'h0	8'h0	reg0	RW	MPCS_CONTROL
8'h08	8'h010	reg10	RW	MPCS TX Path Register
8'h09	8'h012	reg12	RW	MPCS RX Path Register0
8'h09	8'h13	reg13	RO	MPCS RX Path Register1
8'h0A	8'h14	reg14	RW	8B/10B Control Register0
8'h0A	8'h15	reg15	RW	8B/10B Control Register1
8'h0B	8'h16	reg16	RW	Word Alignment Control Register
8'h0C	8'h18	reg18	RW	Primary Word Alignment Pattern Register0
8'h0C	8'h19	reg19	RW	Primary Word Alignment Pattern Register1
8'h0D	8'h1A	reg1A	RW	Primary Word Alignment Pattern Register MSB
8'h0E	8'h1C	reg1C	RW	Secondary Word Alignment Pattern Register0
8'h0E	8'h1D	reg1D	RW	Secondary Word Alignment Pattern Register1
8'h0F	8'h1E	reg1E	RW	Secondary Word Alignment Pattern Register MSB
8'h10	8'h20	reg20	RW	Word Alignment Pattern Mask Code Register0
8'h10	8'h21	reg21	RW	Word Alignment Pattern Mask Code Register1

LMMI Word Offset	LMMI Byte Offset	Register Name	Access Type	Description
8'h11	8'h22	reg22	RW	Word Alignment Pattern Mask Code Register MSB
8'h12	8'h24	reg24	RW	Sync_Det FSM Configuration Register0
8'h12	8'h25	reg25	RW	Sync_Det FSM Configuration Register1
8'h13	8'h26	reg26	RW	Sync_Det FSM Configuration Register2
8'h13	8'h27	reg27	RW	Sync_Det FSM Configuration Register3
8'h14	8'h28	reg28	RO	Number of Bit Slipped During Word Alignment Register
8'h15	8'h2A	reg2A	RW	Primary Sync Det Pattern Register0
8'h15	8'h2B	reg2B	RW	Primary Sync Det Pattern Register1
8'h16	8'h2C	reg2C	RW	Primary Sync Det Pattern Register2
8'h16	8'h2D	reg2D	RW	Primary Sync Det Pattern Register3
8'h17	8'h2E	reg2E	RW	Primary Sync Det Pattern Register MSB
8'h18	8'h30	reg30	RW	Secondary Sync Det Pattern Register0
8'h18	8'h31	reg31	RW	Secondary Sync Det Pattern Register1
8'h19	8'h32	reg32	RW	Secondary Sync Det Pattern Register2
8'h19	8'h33	reg33	RW	Secondary Sync Det Pattern Register3
8'h1A	8'h34	reg34	RW	Secondary Sync Det Pattern Register MSB
8'h1B	8'h36	reg36	RW	Sync_Det Pattern Mask Code Register0
8'h1B	8'h37	reg37	RW	Sync_Det Pattern Mask Code Register1
8'h1C	8'h38	reg38	RW	Sync_Det Pattern Mask Code Register2
8'h1C	8'h39	reg39	RW	Sync_Det Pattern Mask Code Register3
8'h1D	8'h3A	reg3A	RW	Sync_Det Pattern Mask Code Register MSB
8'h1E	8'h3C	reg3C	RW	Lane Alignment Control Register
8'h1E	8'h3D	reg3D	RW	Maximum Lane-to-Lane Skew Register
8'h1F	8'h3E	reg3E	RW	Primary Lane Alignment Pattern Register0
8'h1F	8'h3F	reg3F	RW	Primary Lane Alignment Pattern Register1
8'h20	8'h40	reg40	RW	Primary Lane Alignment Pattern Register2
8'h20	8'h41	reg41	RW	Primary Lane Alignment Pattern Register3
8'h21	8'h42	reg42	RW	Primary Lane Alignment Pattern Register MSB
8'h22	8'h44	reg44	RW	Secondary Lane Alignment Pattern Register0
8'h22	8'h45	reg45	RW	Secondary Lane Alignment Pattern Register1
8'h23	8'h46	reg46	RW	Secondary Lane Alignment Pattern Register2
8'h23	8'h47	reg47	RW	Secondary Lane Alignment Pattern Register3
8'h24	8'h48	reg48	RW	Secondary Lane Alignment Pattern Register MSB
8'h25	8'h4A	reg4A	RO	Lane Aligner Byte Shift Status Register
8'h26	8'h4C	reg4C	RW	Lane Alignment Pattern Mask Code Register
8'h27	8'h4E	reg4E	RW	Clock Frequency Compensation Control Register
8'h27	8'h4F	reg4F	RW	SKIP Pattern Insertion/Deletion Control Register
8'h28	8'h50	reg50	RW	Elastic FIFO Water Line Register0
8'h28	8'h51	reg51	RW	Elastic FIFO Water Line Register1
8'h29	8'h52	reg52	RW	Primary SKIP Pattern Register0
8'h29	8'h53	reg53	RW	Primary SKIP Pattern Register1
8'h2A	8'h54	reg54	RW	Primary SKIP Pattern Register2
8'h2A	8'h55	reg55	RW	Primary SKIP Pattern Register3
8'h2B	8'h56	reg56	RW	Primary SKIP Pattern Register MSB
8'h2C	8'h58	reg58	RW	Secondary SKIP Pattern Register0
8'h2C	8'h59	reg59	RW	Secondary SKIP Pattern Register1

LMMI Word Offset	LMMI Byte Offset	Register Name	Access Type	Description
8'h2D	8'h5A	reg5A	RW	Secondary SKIP Pattern Register2
8'h2D	8'h5B	reg5B	RW	Secondary SKIP Pattern Register3
8'h2E	8'h5C	reg5C	RW	Secondary SKIP Pattern Register MSB
8'h2F	8'h5E	reg5E	RW	SKIP Pattern Mask Code Register
8'h30	8'h60	reg60	RW	64B/66B PCS TX Path Control Register
8'h31	8'h62	reg62	RW	64B/66B PCS TX Gearbox FIFO Setting Register HIGH
8'h31	8'h63	reg63	RW	64B/66B PCS TX Gearbox FIFO Setting Register LOW
8'h32	8'h64	reg64	RW	64B/66B PCS RSFEC TX Gearbox FIFO Setting Register HIGH
8'h32	8'h65	reg65	RW	64B/66B PCS RSFEC TX Gearbox FIFO Setting Register LOW
8'h33	8'h66	reg66	RW	64B/66B PCS RX Path Control Register0
8'h33	8'h67	reg67	RW	64B/66B PCS RX Path Control Register1
8'h34	8'h68	reg68	RW	64B/66B PCS RX RSFEC AM period Register0
8'h34	8'h69	reg69	RW	64B/66B PCS RX RSFEC AM period Register1
8'h35	8'h6A	reg6A	RW	64B/66B PCS RX RSFEC AM period Register2
8'h36	8'h6C	reg6C	RW	64B/66B PCS CTC Water Line Register HIGH
8'h36	8'h6D	reg6D	RW	64B/66B PCS CTC Water Line Register LOW
8'h37	8'h6E	reg6E	RW	64B/66B PCS RX Gearbox FIFO Setting Register HIGH
8'h37	8'h6F	reg6F	RW	64B/66B PCS RX Gearbox FIFO Setting Register LOW
8'h38	8'h70	reg70	RW	64B/66B PCS RSFEC RX Gearbox FIFO Setting Register HIGH
8'h38	8'h71	reg71	RW	64B/66B PCS RSFEC RX Gearbox FIFO Setting Register LOW
8'h39	8'h72	reg72	RO	64B/66B PCS RX Path Status Register
8'h3A	8'h74	reg74	RO	64B/66B PCS Block Align Shift Register
8'h3B	8'h76	reg76	RO	64B/66B PCS RSFEC Align Shift Register0
8'h3B	8'h77	reg77	RO	64B/66B PCS RSFEC Align Shift Register1
8'h3C	8'h78	reg78	RO	64B/66B PCS FCFEC Align Shift Register0
8'h3C	8'h79	reg79	RO	64B/66B PCS FCFEC Align Shift Register1
8'h3D	8'h7A	reg7A	RW	10GBASE-R Error Counter Clear Register
8'h3E	8'h7C	reg7C	RO	10GBASE-R Block Error Counter Register0
8'h3E	8'h7D	reg7D	RO	10GBASE-R Block Error Counter Register1
8'h3F	8'h7E	reg7E	RO	10GBASE-R Block Error Counter Register2
8'h3F	8'h7F	reg7F	RO	10GBASE-R Block Error Counter Register3
8'h40	8'h80	reg80	RO	10GBASE-R Block Error Counter Register4
8'h40	8'h81	reg81	RO	10GBASE-R Block Error Counter Register5
8'h41	8'h82	reg82	RO	10GBASE-R Block Error Counter Register6
8'h41	8'h83	reg83	RO	10GBASE-R Block Error Counter Register7
8'h42	8'h84	reg84	RW	10GBASE-R Test Pattern Seed A Register0
8'h42	8'h85	reg85	RW	10GBASE-R Test Pattern Seed A Register1
8'h43	8'h86	reg86	RW	10GBASE-R Test Pattern Seed A Register2
8'h43	8'h87	reg87	RW	10GBASE-R Test Pattern Seed A Register3
8'h44	8'h88	reg88	RW	10GBASE-R Test Pattern Seed A Register4
8'h44	8'h89	reg89	RW	10GBASE-R Test Pattern Seed A Register5
8'h45	8'h8A	reg8A	RW	10GBASE-R Test Pattern Seed A Register6
8'h45	8'h8B	reg8B	RW	10GBASE-R Test Pattern Seed A Register7
8'h46	8'h8C	reg8C	RW	10GBASE-R Test Pattern Seed B Register0
8'h46	8'h8D	reg8D	RW	10GBASE-R Test Pattern Seed B Register1
8'h47	8'h8E	reg8E	RW	10GBASE-R Test Pattern Seed B Register2

LMMI Word Offset	LMMI Byte Offset	Register Name	Access Type	Description
8'h47	8'h8F	reg8F	RW	10GBASE-R Test Pattern Seed B Register3
8'h48	8'h90	reg90	RW	10GBASE-R Test Pattern Seed B Register4
8'h48	8'h91	reg91	RW	10GBASE-R Test Pattern Seed B Register5
8'h49	8'h92	reg92	RW	10GBASE-R Test Pattern Seed B Register6
8'h49	8'h93	reg93	RW	10GBASE-R Test Pattern Seed B Register7
8'h4A	8'h94	reg94	RW	10GBASE-R Test Pattern Control Register0
8'h4A	8'h95	reg95	RW	10GBASE-R Test Pattern Control Register1
8'h4B	8'h96	reg96	RO	10GBASE-R Test Pattern Error Control Register0
8'h4B	8'h97	reg97	RO	10GBASE-R Test Pattern Error Control Register1
8'h4C	8'h98	reg98	RW	Lane Alignment Debug Control Register0
8'h4C	8'h99	reg99	RW	Lane Alignment Debug Control Register1
8'h4D	8'h9A	reg9A	RO	Lane Alignment Debug Status Register0
8'h4D	8'h9B	reg9B	RO	Lane Alignment Debug Status Register1
8'h4E	8'h9C	reg9C	RO	Lane Alignment Debug Status Register2
8'h4E	8'h9D	reg9D	RO	Lane Alignment Debug Status Register3
8'h4F	8'h9E	reg9E	RW	Clock Frequency Compensation Debug Control Register
8'h4F	8'h9F	reg9F	RW	Clock Frequency Compensation Debug Control Register 2
8'h50	8'hA0	regA0	RW	Loop-back Mode Control Register
8'h51	8'hA2	regA2	RW	FIFO Debug Control Register0
8'h51	8'hA3	regA3	RW	FIFO Debug Control Register1
8'h52	8'hA4	regA4	RO	FIFO Debug Status Register0
8'h52	8'hA5	regA5	RO	FIFO Debug Status Register1
8'h53	8'hA6	regA6	RW	MPPCS BIST Control Register0
8'h53	8'hA7	regA7	RW	MPPCS BIST Control Register1
8'h54	8'hA8	regA8	RW	User Defined BIST Constant 1 Register0
8'h54	8'hA9	regA9	RW	User Defined BIST Constant 1 Register1
8'h55	8'hAA	regAA	RW	User Defined BIST Constant 1 Register MSB
8'h56	8'hAC	regAC	RW	User Defined BIST Constant 2 Register0
8'h56	8'hAD	regAD	RW	User Defined BIST Constant 2 Register1
8'h57	8'hAE	regAE	RW	User Defined BIST Constant 2 Register MSB
8'h58	8'hB0	regB0	RO	BIST Status Register0
8'h58	8'hB1	regB1	RO	BIST Status Register1
8'h59	8'hB2	regB2	RO	PMA Direct Status Register
8'h5A	8'hB4	regB4	RW	TX DL Control Register1
8'h5B	8'hB6	regB6	RW	TX DL Control Register2
8'h5B	8'hB7	regB7	RW	TX DL Control Register3
8'h5C	8'hB8	regB8	RW	TX DL Status Register1
8'h5C	8'hB9	regB9	RO	TX_DL Status Register2
8'h5D	8'hBA	regBA	RO	TX DL Status Register3
8'h60	8'hC0	regC0	RW	RX DL Control Register
8'h61	8'hC2	regC2	RW	RX DL Control Register2
8'h61	8'hC3	regC3	RW	RX DL Control Register3
8'h62	8'hC4	regC4	RO	RX DL Status Register1
8'h62	8'hC5	regC5	RO	RX DL Status Register2
8'h63	8'hC6	regC6	RO	RX DL Status Register3
8'h63	8'hC7	regC7	RO	RX DL Status Register4

LMMI Word Offset	LMMI Byte Offset	Register Name	Access Type	Description
8'h68	8'hD0	regD0	RW	10GBASE-R RSFEC TX Error Injection Control Register0
8'h69	8'hD2	regD2	RW	10GBASE-R RSFEC TX Error Injection Control Register1
8'h69	8'hD3	regD3	RW	10GBASE-R RSFEC TX Error Injection Control Register2
8'h6A	8'hD4	regD4	RW	10GBASE-R RSFEC TX Error Injection Control Register3
8'h6A	8'hD5	regD5	RW	10GBASE-R RSFEC TX Error Injection Control Register4
8'h6B	8'hD6	regD6	RW	10GBASE-R RSFEC TX Error Injection Control Register5
8'h6B	8'hD7	regD7	RW	10GBASE-R RSFEC TX Error Injection Control Register6
8'h6C	8'hD8	regD8	RW	10GBASE-R RSFEC TX Error Injection Control Register7
8'h6D	8'hDA	regDA	RW	10GBASE-R RSFEC TX Error Injection Control Register8
8'h6D	8'hDB	regDB	RW	10GBASE-R RSFEC TX Error Injection Control Register9
8'h6E	8'hDC	regDC	RW	10GBASE-R RSFEC TX Error Injection Control Register10
8'h6E	8'hDD	regDD	RW	10GBASE-R RSFEC TX Error Injection Control Register11
8'h6F	8'hDE	regDE	RW	10GBASE-R RSFEC TX Error Injection Control Register12

Table 9.4. PMA Registers Summary (Offset Address is in Byte)

LMMI Word Offset	LMMI Byte Offset	Register Name	Access Type	Description
8'hA2	8'h144	reg144	RW	PHY_LANE_RX_REQUEST_REQACK0
8'hA2	8'h145	reg145	RO	PHY_LANE_RX_REQUEST_REQACK1
8'hA3	8'h146	reg146	RW	PHY_LANE_RX_ADAPTATION_REQACK0
8'hA3	8'h147	reg147	RO	PHY_LANE_RX_ADAPTATION_REQACK1
8'hAA	8'h154	reg154	RW	PHY_LANE_RX_ADAPTATION
8'hAC	8'h158	reg158	RW	PHY_LANE_LOOPBACK
8'hAE	8'h15C	reg15C	RW	PHY_LANE_TX_REQUEST_REQACK0
8'hAE	8'h15D	reg15D	RO	PHY_LANE_TX_REQUEST_REQACK1
8'hB4	8'h168	reg168	RO	PHY_LANE_RX_ADAPT_STATUS0
8'hB4	8'h169	reg169	RO	PHY_LANE_RX_ADAPT_STATUS1

9.2. MPCS Control Register (reg0)

Table 9.5. MPCS Control Register (reg0) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	tx_pcs_mode	00: Invalid 01: 8b/10b data path 10: 64b/66b data path 11: PMA only mode	RW	2	2'b00
3:2	rx_pcs_mode	00: Invalid 01: 8b/10b data path 10: 64b/66b data path 11: PMA only mode	RW	2	2'b00
5:4	tx_serdes_arc_mode	The mode TX serdes architecture datapath is operating at 00: Disabled 01: XGMII mode (tx_pcs_mode must set to 2'b10) 10: FCFEC mode (tx_pcs_mode must set to 2'b10)	RW	2	2'b00

Field	Name	Description	Access	Width	Reset
		11: PMA mode (tx_pcs_mode must set to 2'b10) The tx_pcs_mode is only meaningful with tx_serdes_arc_mode disabled.			
7:6	rx_serdes_arc_mode	The mode RX serdes architecture datapath is operating at 00: Disabled 01: XGMII mode (rx_pcs_mode must set to 2'b10) 10: FCFEC mode (rx_pcs_mode must set to 2'b10) 11: PMA mode (rx_pcs_mode must set to 2'b10) The rx_pcs_mode is only meaningful with rx_serdes_arc_mode disabled.	RW	2	2'b00

9.3. MPCS TX Path Register (reg10)

Table 9.6. MPCS TX Path Register (reg10) Descriptions

Field	Name	Description	Access	Width	Reset
0	enc_8b10b_dis	1: Disables 8b10b encoding 0: Enables 8b10b encoding	RW	1	1'b0
1	tx_dbus_20b	1: Internal data bus is 20-bit width 0: Internal data bus is 10-bit width This bit is not applicable for 64B/66B PCS.	RW	1	1'b0
2	tx_bond_mask	1: Excludes this channel from bonded channel group 0: Do not exclude this channel	RW	1	1'b0
3	tx_8b10b_swizzle_en	1: Enables bit swizzling 0: Disables bit swizzling	RW	1	1'b0
4	tx_symbol_swap_en	1: Enables byte swap when 20b data bus is enabled 0: Disables byte swap	RW	1	1'b0
7:5	rsvd	Reserved	—	3	3'b000

9.4. MPCS RX Path Register0 (reg12)

Table 9.7. MPCS RX Path Register0 (reg12) Descriptions

Field	Name	Description	Access	Width	Reset
0	dec_8b10b_dis	1: Disables 8b10b decoding 0: Enables 8b10b decoding	RW	1	1'b0
1	rx_dbus_20b	1: Internal data bus is 20-bit width 0: Internal data bus is 10-bit width This bit is not applicable for 64B/66B PCS.	RW	1	1'b0
2	rx_bond_mask	1: Excludes this channel from bonded channel group 0: Do not exclude this channel	RW	1	1'b0
3	rx_8b10b_swizzle_en	1: Enables bit swizzling	RW	1	1'b0

Field	Name	Description	Access	Width	Reset
		0: Disables bit swizzling			
4	rx_symbol_swap_en	1: Enables byte swap when 20b data bus is enabled 0: Disables byte swap	RW	1	1'b0
5	din_com_align	1: Enables the feature that always puts the COMMA byte to LSByte (Byte_0) of the data bus 0: Do not enable this feature This feature can be optionally enabled. Once enabled, the value of register bit *din_byte_shift* is used to signal the number of byte data shift during the COMMA alignment operation.	RW	1	1'b0
7:6	rsvd	Reserved	—	2	2'b00

9.5. MPCS RX Path Register1 (reg13)

Table 9.8. MPCS RX Path Register1 (reg13) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	din_byte_shift	11: There is three byte data shift caused by COMMA byte alignment operation 10: There is two byte data shift caused by COMMA byte alignment operation 01: There is one byte data shift caused by COMMA byte alignment operation 00: There is no byte shift	RO	2	2'b00
4:2	cmn_fifo_8b10b_dl_shift	Number of byte shifting that happens in the common FIFO. The value could go up to 7 for the gear ratio of 4 and 20b data width.	RO	3	3'b000
7:5	rsvd	Reserved	—	3	3'b000

9.6. 8B/10B Control Register0 (reg14)

Table 9.9. 8B/10B Control Register0 (reg14) Descriptions

Field	Name	Description	Access	Width	Reset
0	enc_8b10b_interleave	1: Enables interleaving mode in 8B/10B encoder 0: Disables interleaving mode in 8B/10B encoder	RW	1	1'b0
7:1	rsvd	Reserved	—	7	7'b0000000

9.7. 8B/10B Control Register1 (reg15)

Table 9.10. 8B/10B Control Register1 (reg15) Descriptions

Field	Name	Description	Access	Width	Reset
0	dec_8b10b_interleave	1: Enables interleaving mode in 8B/10B decoder 0: Disables interleaving mode in 8B/10B decoder	RW	1	1'b0
1	dec_8b10b_bypass_disperr_8b	1: Forwards 8b decoded data when disparity error happen 0: Outputs FE error character when disparity error happen	RW	1	1'b0
2	dec_8b10b_bypass_disperr_errcode_8b	1: Forwards 8b decoded data when disparity error or error code happen	RW	1	1'b0

Field	Name	Description	Access	Width	Reset
		0: Outputs FE error character when disparity error or error code happen			
3	dec_8b10b_bypass_disperr_errcode_10b	1: Forwards 10b encoded data when disparity error or error code happen 0: Outputs FE error character when disparity error or error code happen	RW	1	1'b0
7:4	rsvd	Reserved	—	4	4'b0000

9.8. Word Alignment Control Register (reg16)

Table 9.11. Word Alignment Control Register (reg16) Descriptions

Field	Name	Description	Access	Width	Reset
0	rsvd	Reserved	—	1	1'b0
1	syncdet_fsm_dis	0: Uses *sync_det* FSM to control the automatic word alignment 1: Do not use *sync_det* FSM to control the automatic word alignment	RW	1	1'b0
2	wa_ptn_20b	0: Word alignment pattern is 10-bit width; only bit 9 to 0 of the 20-bit pattern and mask code are used. 1: Word alignment pattern is 20-bit width	RW	1	1'b0
3	sec_waptn_dis	0: Enables secondary word alignment pattern matching 1: Disables secondary word alignment pattern matching	RW	1	1'b0
4	align_2byte_dis	0: In 2-byte internal data bus width mode, always put the LSByte of the word alignment pattern to bit 9 to 0 of the data bus. 1: Do not need to put the LSByte of the word alignment pattern to bit 9 to 0 of data bus; the pattern could appear at either bit 9 to 0 or bit 19 to 10.	RW	1	1'b0
5	wa_dis	0: Do not disable word alignment module 1: Disables word alignment module and bypass the input data; when disabled, the *sync_det* FSM will not operate as well.	RW	1	1'b0
7:6	rsvd	Reserved	—	2	2'b00

9.9. Primary Word Alignment Pattern Register0 (reg18)

Table 9.12. Primary Word Alignment Pattern Register0 (reg18) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_wa_ptn__7_0	The 20-bit primary word alignment pattern. In 10-bit width mode, only bit 9 to 0 are applied.	RW	8	8'b01111100

9.10. Primary Word Alignment Pattern Register1 (reg19)

Table 9.13. Primary Word Alignment Pattern Register1 (reg19) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_wa_ptn__17_10	The 20-bit primary word alignment pattern. In 10-bit width mode, only bit 9 to 0 are applied.	RW	8	8'b00000000

9.11. Primary Word Alignment Pattern Register MSB (reg1A)

Table 9.14. Primary Word Alignment Pattern Register MSB (reg1A) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	pri_wa_ptn__9_8	The 20-bit primary word alignment pattern. In 10-bit width mode, only bit 9 to 0 are applied.	RW	2	2'b01
3:2	pri_wa_ptn__19_18	The 20-bit primary word alignment pattern. In 10-bit width mode, only bit 9 to 0 are applied.	RW	2	2'b00
7:4	rsvd	Reserved	—	4	4'b0000

9.12. Secondary Word Alignment Pattern Register0 (reg1C)

Table 9.15. Secondary Word Alignment Pattern Register0 (reg1C) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_wa_ptn__7_0	The 20-bit secondary word alignment pattern. In 10-bit width mode, only bit 9 to 0 are applied.	RW	8	8'b10000011

9.13. Secondary Word Alignment Pattern Register1 (reg1D)

Table 9.16. Secondary Word Alignment Pattern Register1 (reg1D) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_wa_ptn__17_10	The 20-bit secondary word alignment pattern. In 10-bit width mode, only bit 9 to 0 are applied.	RW	8	8'b00000000

9.14. Secondary Word Alignment Pattern Register MSB (reg1E)

Table 9.17. Secondary Word Alignment Pattern Register MSB (reg1E) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	sec_wa_ptn__9_8	The 20-bit secondary word alignment pattern. In 10-bit width mode, only bit 9 to 0 are applied.	RW	2	2'b10
3:2	sec_wa_ptn__19_18	The 20-bit secondary word alignment pattern. In 10-bit width mode, only bit 9 to 0 are applied.	RW	2	2'b00
7:4	rsvd	Reserved	—	4	4'b0000

9.15. Word Alignment Pattern Mask Code Register0 (reg20)

Table 9.18. Word Alignment Pattern Mask Code Register0 (reg20) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	wa_mask_code__7_0	The 20-bit word alignment pattern mask code. In 10-bit width mode, only bit 9 to 0 are applied. 1: The corresponding bit of word alignment pattern is ignored during alignment pattern matching 0: The corresponding bit of word alignment pattern is not ignored during alignment pattern matching	RW	8	8'b00000000

9.16. Word Alignment Pattern Mask Code Register1 (reg21)

Table 9.19. Word Alignment Pattern Mask Code Register1 (reg21) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	wa_mask_code__17_10	The 20-bit word alignment pattern mask code. In 10-bit width mode, only bit 9 to 0 are applied. 1: The corresponding bit of word alignment pattern is ignored during alignment pattern matching 0: The corresponding bit of word alignment pattern is not ignored during alignment pattern matching	RW	8	8'b00000000

9.17. Word Alignment Pattern Mask Code Register MSB (reg22)

Table 9.20. Word Alignment Pattern Mask Code Register MSB (reg22) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	wa_mask_code__9_8	The 20-bit word alignment pattern mask code. In 10-bit width mode, only bit 9 to 0 are applied. 1: The corresponding bit of word alignment pattern is ignored during alignment pattern matching 0: The corresponding bit of word alignment pattern is not ignored during alignment pattern matching	RW	2	2'b00
3:2	wa_mask_code__19_18	The 20-bit word alignment pattern mask code. In 10-bit width mode, only bit 9 to 0 are applied. 1: The corresponding bit of word alignment pattern is ignored during alignment pattern matching 0: The corresponding bit of word alignment pattern is not ignored during alignment pattern matching	RW	2	2'b00
7:4	rsvd	Reserved	—	4	4'b0000

9.18. Sync_Det FSM Configuration Register0 (reg24)

Table 9.21. Sync_Det FSM Configuration Register0 (reg24) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	num_val_sync	The number of valid synchronization code groups or ordered sets that *sync_det* FSM must receive to achieve synchronization state.	RW	8	8'b00000011

9.19. Sync_Det FSM Configuration Register1 (reg25)

Table 9.22. Sync_Det FSM Configuration Register1 (reg25) Descriptions

Field	Name	Description	Access	Width	Reset
5:0	num_bad_code	The number of bad code groups received by *sync_det* FSM to conclude the loss of synchronization.	RW	6	6'b000100
7:6	rsvd	Reserved	—	2	2'b00

9.20. Sync_Det FSM Configuration Register2 (reg26)

Table 9.23. Sync_Det FSM Configuration Register2 (reg26) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	num_good_code	The number of continuous good code groups received by *sync_det* FSM to reduce the error count by one.	RW	8	8'b00000100

9.21. Sync_Det FSM Configuration Register3 (reg27)

Table 9.24. Sync_Det FSM Configuration Register3 (reg27) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	sync_ptn_len	The length of sync_det pattern. 2'b00: the length is 1 2'b01: the length is 2 2'b1x: the length is 4	RW	2	2'b01
2	sync_ptn_align	1: The first byte of sync_det pattern must appear on N-byte boundary, where N is the sync_det pattern length defined by sync_ptn_len register. Once a sync_det pattern is detected, the sync_det FSM will apply this criterion to the validity check of subsequent incoming data. 0: Do not enable this sync_det pattern alignment check	RW	1	1'b1
3	sync_ptn_10b	1: The sync_det pattern is 10b code 0: The sync_det pattern is 8b code	RW	1	1'b0
4	sec_sync_ptn_dis	1: Disables secondary sync_det pattern and do not use it for matching 0: Uses the secondary sync_det pattern for matching	RW	1	1'b0
7:5	rsvd	Reserved	—	3	3'b000

9.22. Number of Bit Slipped During Word Alignment Register (reg28)

Table 9.25. Number of Bit Slipped During Word Alignment Register (reg28) Descriptions

Field	Name	Description	Access	Width	Reset
4:0	num_bit_slipped	0: There is no bit slip 1: 1 bit is slipped 2: 2 bits are slipped … 19: 19 bits are slipped (19 is the maximum number)	RO	5	5'b00000
7:5	rsvd	Reserved	—	3	3'b000

9.23. Primary Sync Det Pattern Register0 (reg2A)

Table 9.26. Primary Sync Det Pattern Register0 (reg2A) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_sdptn_byte0__7_0	Bit 7 to 0 of primary sync_det pattern byte_0	RW	8	8'b10111100

9.24. Primary Sync Det Pattern Register1 (reg2B)

Table 9.27. Primary Sync Det Pattern Register1 (reg2B) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_sdptn_byte1__7_0	Bit 7 to 0 of primary sync_det pattern byte_1	RW	8	8'b01010000

9.25. Primary Sync Det Pattern Register2 (reg2C)

Table 9.28. Primary Sync Det Pattern Register2 (reg2C) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_sdptn_byte2__7_0	Bit 7 to 0 of primary sync_det pattern byte_2	RW	8	8'b00000000

9.26. Primary Sync Det Pattern Register3 (reg2D)

Table 9.29. Primary Sync Det Pattern Register3 (reg2D) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_sdptn_byte3__7_0	Bit 7 to 0 of primary sync_det pattern byte_3	RW	8	8'b00000000

9.27. Primary Sync Det Pattern Register MSB (reg2E)

Table 9.30. Primary Sync Det Pattern Register MSB (reg2E) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	pri_sdptn_byte0__9_8	Bit 9 to 8 of primary sync_det pattern byte_0	RW	2	2'b01
3:2	pri_sdptn_byte1__9_8	Bit 9 to 8 of primary sync_det pattern byte_1	RW	2	2'b00
5:4	pri_sdptn_byte2__9_8	Bit 9 to 8 of primary sync_det pattern byte_2	RW	2	2'b00
7:6	pri_sdptn_byte3__9_8	Bit 9 to 8 of primary sync_det pattern byte_3	RW	2	2'b00

9.28. Secondary Sync Det Pattern Register0 (reg30)

Table 9.31. Secondary Sync Det Pattern Register0 (reg30) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_sdptn_byte0__7_0	Bit 7 to 0 of secondary sync_det pattern byte_0	RW	8	8'b10111100

9.29. Secondary Sync Det Pattern Register1 (reg31)

Table 9.32. Secondary Sync Det Pattern Register1 (reg31) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_sdptn_byte1__7_0	Bit 7 to 0 of secondary sync_det pattern byte_1	RW	8	8'b11000101

9.30. Secondary Sync Det Pattern Register2 (reg32)

Table 9.33. Secondary Sync Det Pattern Register2 (reg32) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_sdptn_byte2__7_0	Bit 7 to 0 of secondary sync_det pattern byte_2	RW	8	8'b00000000

9.31. Secondary Sync Det Pattern Register3 (reg33)

Table 9.34. Secondary Sync Det Pattern Register3 (reg33) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_sdptn_byte3__7_0	Bit 7 to 0 of secondary sync_det pattern byte_3	RW	8	8'b00000000

9.32. Secondary Sync Det Pattern Register MSB (reg34)

Table 9.35. Secondary Sync Det Pattern Register MSB (reg34) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	sec_sdptn_byte0__9_8	Bit 9 to 8 of secondary sync_det pattern byte_0	RW	2	2'b01
3:2	sec_sdptn_byte1__9_8	Bit 9 to 8 of secondary sync_det pattern byte_1	RW	2	2'b00
5:4	sec_sdptn_byte2__9_8	Bit 9 to 8 of secondary sync_det pattern byte_2	RW	2	2'b00
7:6	sec_sdptn_byte3__9_8	Bit 9 to 8 of secondary sync_det pattern byte_3	RW	2	2'b00

9.33. Sync_Det Pattern Mask Code Register0 (reg36)

Table 9.36. Sync_Det Pattern Mask Code Register0 (reg36) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sdptn_mask_byte0__7_0	Bit 7 to 0 of sync_det pattern mask code byte_0	RW	8	8'b00000000

9.34. Sync_Det Pattern Mask Code Register1 (reg37)

Table 9.37. Sync_Det Pattern Mask Code Register1 (reg37) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sdptn_mask_byte1__7_0	Bit 7 to 0 of sync_det pattern mask code byte_1	RW	8	8'b00000000

9.35. Sync_Det Pattern Mask Code Register2 (reg38)

Table 9.38. Sync_Det Pattern Mask Code Register2 (reg38) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sdptn_mask_byte2__7_0	Bit 7 to 0 of sync_det pattern mask code byte_2	RW	8	8'b00000000

9.36. Sync_Det Pattern Mask Code Register3 (reg39)

Table 9.39. Sync_Det Pattern Mask Code Register3 (reg39) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sdptn_mask_byte3__7_0	Bit 7 to 0 of sync_det pattern mask code byte_3	RW	8	8'b00000000

9.37. Sync_Det Pattern Mask Code Register MSB (reg3A)

Table 9.40. Sync_Det Pattern Mask Code Register MSB (reg3A) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	sdptn_mask_byte0__9_8	Bit 9 to 8 of sync_det pattern mask code byte_0	RW	2	2'b10
3:2	sdptn_mask_byte1__9_8	Bit 9 to 8 of sync_det pattern mask code byte_1	RW	2	2'b10
5:4	sdptn_mask_byte2__9_8	Bit 9 to 8 of sync_det pattern mask code byte_2	RW	2	2'b00
7:6	sdptn_mask_byte3__9_8	Bit 9 to 8 of sync_det pattern mask code byte_3	RW	2	2'b00

9.38. Lane Alignment Control Register (reg3C)

Table 9.41. Lane Alignment Control Register (reg3C) Descriptions

Field	Name	Description	Access	Width	Reset
0	lalign_en	0: Disables lane alignment 1: Enables lane alignment	RW	1	1'b0
1	lalign_10b	0: The input data is in 8b code mode 1: The input data is in 10b code mode	RW	1	1'b0
3:2	lalign_ptn_len	The lane alignment pattern length in byte. 2'b00: 1-byte 2'b01: 2-byte 2'b1x: 4-byte	RW	2	2'b00
4	sec_laptn_en	0: Disables secondary lane alignment pattern matching 1: Enables secondary lane alignment pattern matching	RW	1	1'b0
7:5	rsvd	Reserved	—	3	3'b000

9.39. Maximum Lane-to-Lane Skew Register (reg3D)

Table 9.42. Maximum Lane-to-Lane Skew Register (reg3D) Descriptions

Field	Name	Description	Access	Width	Reset
3:0	max_lskew	The maximum lane-to-lane skew in byte. 4'd0: The maximum skew that can be handled by hardware 4'd1: 1-byte skew 4'd2: 2-byte skew … 4'd10: 10-byte skew Other codes: The maximum skew that can be handled by hardware The maximum lane-to-lane skew that can be handled by hardware is 10-byte. This domain must be correctly configured by software. Hardware uses this domain without any checking.	RW	4	4'b0000

Field	Name	Description	Access	Width	Reset
		Note that: • If the lane number is less than or equal to 4, the *max_ lskew* should be configured as *expected max skew + 1*. • If the lane number is more than 4, the *max_lskew* should be configured as *expected max skew + 2*. • Example: If the max skew specified by a 4-lane protocol is 5, this register should be configured as *6*.			
7:4	rsvd	Reserved	—	4	4'b0000

9.40. Primary Lane Alignment Pattern Register0 (reg3E)

Table 9.43. Primary Lane Alignment Pattern Register0 (reg3E) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_laptn_byte0__7_0	Bit 7 to 0 of primary lane alignment pattern byte_0	RW	8	8'b01111100

9.41. Primary Lane Alignment Pattern Register1 (reg3F)

Table 9.44. Primary Lane Alignment Pattern Register1 (reg3F) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_laptn_byte1__7_0	Bit 7 to 0 of primary lane alignment pattern byte_1	RW	8	8'b00000000

9.42. Primary Lane Alignment Pattern Register2 (reg40)

Table 9.45. Primary Lane Alignment Pattern Register2 (reg40) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_laptn_byte2__7_0	Bit 7 to 0 of primary lane alignment pattern byte_2	RW	8	8'b00000000

9.43. Primary Lane Alignment Pattern Register3 (reg41)

Table 9.46. Primary Lane Alignment Pattern Register3 (reg41) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_laptn_byte3__7_0	Bit 7 to 0 of primary lane alignment pattern byte_3	RW	8	8'b00000000

9.44. Primary Lane Alignment Pattern Register MSB (reg42)

Table 9.47. Primary Lane Alignment Pattern Register MSB (reg42) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	pri_laptn_byte0__9_8	Bit 9 to 8 of primary lane alignment pattern byte_0	RW	2	2'b01
3:2	pri_laptn_byte1__9_8	Bit 9 to 8 of primary lane alignment pattern byte_1	RW	2	2'b00
5:4	pri_laptn_byte2__9_8	Bit 9 to 8 of primary lane alignment pattern byte_2	RW	2	2'b00
7:6	pri_laptn_byte3__9_8	Bit 9 to 8 of primary lane alignment pattern byte_3	RW	2	2'b00

9.45. Secondary Lane Alignment Pattern Register0 (reg44)

Table 9.48. Secondary Lane Alignment Pattern Register0 (reg44) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_laptn_byte0__7_0	Bit 7 to 0 of secondary lane alignment pattern byte_0	RW	8	8'b01111100

9.46. Secondary Lane Alignment Pattern Register1 (reg45)

Table 9.49. Secondary Lane Alignment Pattern Register1 (reg45) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_laptn_byte1__7_0	Bit 7 to 0 of secondary lane alignment pattern byte_1	RW	8	8'b00000000

9.47. Secondary Lane Alignment Pattern Register2 (reg46)

Table 9.50. Secondary Lane Alignment Pattern Register2 (reg46) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_laptn_byte2__7_0	Bit 7 to 0 of secondary lane alignment pattern byte_2	RW	8	8'b00000000

9.48. Secondary Lane Alignment Pattern Register3 (reg47)

Table 9.51. Secondary Lane Alignment Pattern Register3 (reg47) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_laptn_byte3__7_0	Bit 7 to 0 of secondary lane alignment pattern byte_3	RW	8	8'b00000000

9.49. Secondary Lane Alignment Pattern Register MSB (reg48)

Table 9.52. Secondary Lane Alignment Pattern Register MSB (reg48) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	sec_laptn_byte0__9_8	Bit 9 to 8 of secondary lane alignment pattern byte_0	RW	2	2'b01
3:2	sec_laptn_byte1__9_8	Bit 9 to 8 of secondary lane alignment pattern byte_1	RW	2	2'b00
5:4	sec_laptn_byte2__9_8	Bit 9 to 8 of secondary lane alignment pattern byte_2	RW	2	2'b00
7:6	sec_laptn_byte3__9_8	Bit 9 to 8 of secondary lane alignment pattern byte_3	RW	2	2'b00

9.50. Lane Aligner Byte Shift Status Register (reg4A)

Table 9.53. Lane Aligner Byte Shift Status Register (reg4A) Descriptions

Field	Name	Description	Access	Width	Reset
0	lalign_byte_shift	One byte shifted for lane alignment. 1: One byte shifted 0: No byte shifted	RO	1	1'b0
7:1	rsvd	Reserved	—	7	7'b0000000

9.51. Lane Alignment Pattern Mask Code Register (reg4C)

Table 9.54. Lane Alignment Pattern Mask Code Register (reg4C) Descriptions

Field	Name	Description	Access	Width	Reset
3:0	lalign_mask_code	Bit 3 to 0 of lane alignment pattern mask code. 1: The corresponding byte of lane alignment pattern is ignored during alignment pattern matching 0: The corresponding byte of lane alignment pattern is not ignored during alignment pattern matching	RW	4	4'b0000
7:4	rsvd	Reserved	—	4	4'b0000

9.52. Clock Frequency Compensation Control Register (reg4E)

Table 9.55. Clock Frequency Compensation Control Register (reg4E) Descriptions

Field	Name	Description	Access	Width	Reset
0	clk_comp_en	1: Enables clock frequency compensation 0: Disables clock frequency compensation Combined with the *ctc_fifo_en* bit, the *clock frequency compensation* module can work in different modes.	RW	1	1'b0
1	ctc_fifo_en	0: Disables CTC FIFO 1: Enables CTC FIFO	RW	1	1'b0
2	clk_comp_10b	0: The input data is in 8b code mode 1: The input data is in 10b code mode	RW	1	1'b0
4:3	skip_ptn_len	The SKIP pattern length in byte. 2'b00: 1-byte 2'b01: 2-byte 2'b1x: 4-byte	RW	2	2'b00
5	sec_skip_en	0 : Disables secondary SKIP pattern matching 1 : Enables secondary SKIP pattern matching	RW	1	1'b0
7:6	rsvd	Reserved	—	2	2'b00

9.53. SKIP Pattern Insertion/Deletion Control Register (reg4F)

Table 9.56. SKIP Pattern Insertion/Deletion Control Register (reg4F) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	min_ipg_cnt	This register is used to guarantee the minimum number of bytes between packets (minimum Inter Packet Gap) after SKIP deletion. 2'b00: No constraint on SKIP deletion 2'b01: At least one SKIP pattern is kept in IPG 2'b10: At least two SKIP patterns are kept in IPG 2'b11: At least three SKIP patterns are kept in IPG The length in byte of SKIP pattern is defined in the *skip_ptn_len* domain of register *Clock Frequency Compensation Control*.	RW	2	2'b00
6:2	ctc_repeat_wait	This register is used to control the clock correction frequency. It defines the minimum number of clock cycles between two clock correction events (including SKIP deletion and insertion).	RW	5	5'b00000

Field	Name	Description	Access	Width	Reset
		0: No constraint on SKIP deletion and insertion x (x = 1 to 31): The next time of SKIP deletion or insertion must be x clock cycles away from the current SKIP deletion or insertion			
7	rsvd	Reserved	—	1	1'b0

9.54. Elastic FIFO Water Line Register0 (reg50)

Table 9.57. Elastic FIFO Water Line Register0 (reg50) Descriptions

Field	Name	Description	Access	Width	Reset
4:0	high_water_line	Clock Compensation FIFO high water line. 10b data width: • mean is 5'b10000 • overrun when hit 5'b11111 20b data width: • mean is 5'b01000 • overrun when hit 5'b01111	RW	5	5'b10110
7:5	rsvd	Reserved	—	3	3'b000

9.55. Elastic FIFO Water Line Register1 (reg51)

Table 9.58. Elastic FIFO Water Line Register1 (reg51) Descriptions

Field	Name	Description	Access	Width	Reset
4:0	low_water_line	Clock Compensation FIFO low water line. 10b data width: • mean is 5'b10000 • underrun when hit 5'b00000 20b data width: • mean is 5'b01000 • underrun when hit 5'b00000	RW	5	5'b01010
7:5	rsvd	Reserved	—	3	3'b000

9.56. Primary SKIP Pattern Register0 (reg52)

Table 9.59. Primary SKIP Pattern Register0 (reg52) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_skip_byte0__7_0	Bit 7 to 0 of primary SKIP pattern byte_0	RW	8	8'b01111100

9.57. Primary SKIP Pattern Register1 (reg53)

Table 9.60. Primary SKIP Pattern Register1 (reg53) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_skip_byte1__7_0	Bit 7 to 0 of primary SKIP pattern byte_1	RW	8	8'b00000000

9.58. Primary SKIP Pattern Register2 (reg54)

Table 9.61. Primary SKIP Pattern Register2 (reg54) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_skip_byte2__7_0	Bit 7 to 0 of primary SKIP pattern byte_2	RW	8	8'b00000000

9.59. Primary SKIP Pattern Register3 (reg55)

Table 9.62. Primary SKIP Pattern Register3 (reg55) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	pri_skip_byte3__7_0	Bit 7 to 0 of primary SKIP pattern byte_3	RW	8	8'b00000000

9.60. Primary SKIP Pattern Register MSB (reg56)

Table 9.63. Primary SKIP Pattern Register MSB (reg56) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	pri_skip_byte0__9_8	Bit 9 to 8 of primary SKIP pattern byte_0	RW	2	2'b01
3:2	pri_skip_byte1__9_8	Bit 9 to 8 of primary SKIP pattern byte_1	RW	2	2'b00
5:4	pri_skip_byte2__9_8	Bit 9 to 8 of primary SKIP pattern byte_2	RW	2	2'b00
7:6	pri_skip_byte3__9_8	Bit 9 to 8 of primary SKIP pattern byte_3	RW	2	2'b00

9.61. Secondary SKIP Pattern Register0 (reg58)

Table 9.64. Secondary SKIP Pattern Register0 (reg58) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_skip_byte0__7_0	Bit 7 to 0 of secondary SKIP pattern byte_0	RW	8	8'b01111100

9.62. Secondary SKIP Pattern Register1 (reg59)

Table 9.65. Secondary SKIP Pattern Register1 (reg59) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_skip_byte1__7_0	Bit 7 to 0 of secondary SKIP pattern byte_1	RW	8	8'b00000000

9.63. Secondary SKIP Pattern Register2 (reg5A)

Table 9.66. Secondary SKIP Pattern Register2 (reg5A) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_skip_byte2__7_0	Bit 7 to 0 of secondary SKIP pattern byte_2	RW	8	8'b00000000

9.64. Secondary SKIP Pattern Register3 (reg5B)

Table 9.67. Secondary SKIP Pattern Register3 (reg5B) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	sec_skip_byte3__7_0	Bit 7 to 0 of secondary SKIP pattern byte_3	RW	8	8'b00000000

9.65. Secondary SKIP Pattern Register MSB (reg5C)

Table 9.68. Secondary SKIP Pattern Register MSB (reg5C) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	sec_skip_byte0__9_8	Bit 9 to 8 of secondary SKIP pattern byte_0	RW	2	2'b01
3:2	sec_skip_byte1__9_8	Bit 9 to 8 of secondary SKIP pattern byte_1	RW	2	2'b00
5:4	sec_skip_byte2__9_8	Bit 9 to 8 of secondary SKIP pattern byte_2	RW	2	2'b00
7:6	sec_skip_byte3__9_8	Bit 9 to 8 of secondary SKIP pattern byte_3	RW	2	2'b00

9.66. SKIP Pattern Mask Code Register (reg5E)

Table 9.69. SKIP Pattern Mask Code Register (reg5E) Descriptions

Field	Name	Description	Access	Width	Reset
3:0	skp_mask_code	Bit 3 to 0 of SKIP pattern mask code. 1: The corresponding byte of SKIP pattern is ignored during SKIP pattern matching 0: The corresponding byte of SKIP pattern is not ignored during SKIP pattern matching	RW	4	4'b0000
7:4	rsvd	Reserved	—	4	4'b0000

9.67. 64B/66B PCS TX Path Control Register (reg60)

Table 9.70. 64B/66B PCS TX Path Control Register (reg60) Descriptions

Field	Name	Description	Access	Width	Reset
0	scr_64b66b_dis	1: Disables scrambler 0: Enables scrambler	RW	1	1'b0
1	enc_64b66b_dis	1: Disables 64B/66B encoder 0: Enables 64B/66B encoder	RW	1	1'b0
2	tx_64b66b_swizzle_en	Bit swizzling for the data to PMA in Tx direction. 0: Disables bit swizzling 1: Enables bit swizzling	RW	1	1'b0
4:3	tx_fec_mode	Tx datapath FEC mode.	RW	2	2'b00

Field	Name	Description	Access	Width	Reset
		00: No FEC 01: Ethernet RSFEC 10: CPRI RSFEC 11: FCFEC			
6:5	tx_gbx_mode	Tx Gearbox mode. 00: 66b to 32b 01: 66b to 16b 10: 66b to 40b 11: Reserved	RW	2	2'b00
7	rsvd	Reserved	—	1	1'b0

9.68. 64B/66B PCS TX Gearbox FIFO Setting Register HIGH (reg62)

Table 9.71. 64B/66B PCS TX Gearbox FIFO Setting Register HIGH (reg62) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	tx_gbx_high	The difference between two pointers to assert almost full flag for Tx gearbox.	RW	8	8'b00000100

9.69. 64B/66B PCS TX Gearbox FIFO Setting Register LOW (reg63)

Table 9.72. 64B/66B PCS TX Gearbox FIFO Setting Register LOW (reg63) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	tx_gbx_low	The difference between two pointers to assert almost empty flag for Tx gearbox.	RW	8	8'b00000100

9.70. 64B/66B PCS RSFEC TX Gearbox FIFO Setting Register HIGH (reg64)

Table 9.73. 64B/66B PCS RSFEC TX Gearbox FIFO Setting Register HIGH (reg64) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_gbx_high	The difference between two pointers to assert almost full flag for RSFEC Tx gearbox.	RW	8	8'b00000100

9.71. 64B/66B PCS RSFEC TX Gearbox FIFO Setting Register LOW (reg65)

Table 9.74. 64B/66B PCS RSFEC TX Gearbox FIFO Setting Register LOW (reg65) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_gbx_low	The difference between two pointers to assert almost empty flag for RSFEC Tx gearbox.	RW	8	8'b00010000

9.72. 64B/66B PCS RX Path Control Register0 (reg66)

Table 9.75. 64B/66B PCS RX Path Control Register0 (reg66) Descriptions

Field	Name	Description	Access	Width	Reset
0	descr_64b66b_dis	1: Disables 64B/66B descrambler 0: Enables 64B/66B descrambler	RW	1	1'b0
1	dec_64b66b_dis	1: Disables 64B/66B decoder 0: Enables 64B/66B decoder	RW	1	1'b0
2	ctc_64b66b_dis	1: Disables clock frequency compensation 0: Enables clock frequency compensation	RW	1	1'b0
3	balign_64b66b_dis	1: Disables block aligner 0: Enables block aligner	RW	1	1'b0
4	del_os_cont	1: Allow continuously delete Sequence ordered_set when doing clock frequency compensation 0: Only delete one Sequence ordered_set each time when doing clock frequency compensation	RW	1	1'b0
5	data_rate	Data rate of 64B/66B data path. 0: 10Gbps 1: 25Gbps	RW	1	1'b0
6	rx_64b66b_swizzle_en	Bit swizzling for the data from PMA in Rx direction. 0: disable bit swizzling 1: enable bit swizzling	RW	1	1'b0
7	rsvd	Reserved	—	1	1'b0

9.73. 64B/66B PCS RX Path Control Register1 (reg67)

Table 9.76. 64B/66B PCS RX Path Control Register1 (reg67) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	rx_gbx_mode	Rx Gearbox mode. 00: 32b to 66b 01: 16b to 66b 10: 40b to 66b 11: Reserved	RW	2	2'b00
3:2	rx_fec_mode	Rx datapath FEC mode. 00: No FEC 01: Ethernet RSFEC 10: CPRI RSFEC 11: FCFEC	RW	2	2'b00
5:4	am_align_mode	Operating mode of AM aligner. The operation is depending on rx_fec_mode and ctc_64b66b_dis configuration. Ethernet RS-FEC: 00: No alignment. Data and AM signals bypass the block. 01: When CTC is disabled, always swap 1 cycle of data right after AM to right before AM and thus delay AM by 1 clock cycle. When CTC is enabled, whenever AM is shifted to upper nibble, always swap it back to lower nibble. 10: Always align AM to the lower word when common FIFO is operating in gear ratio of 2.	RW	2	2'b00

Field	Name	Description	Access	Width	Reset
		11: Implements both 2'b01 and 2'b10 operations above. Non Ethernet RS-FEC: 00: Bypassed 01: Bypassed 10: When CTC is disabled, always align AM to the lower word when common FIFO is operating in gear ratio of 2. When CTC is enabled, no alignment happen. 11: Invalid			
6	rsfec_rx_bei	RSFEC Rx error monitoring. 0: Error monitoring is turned off 1: Error monitoring is turned on When the number of RSFEC symbol errors in a block of 8192 consecutive codewords exceeds a threshold (417), then synchronization header errors will be generated towards the PCS layer for a period of 60 ms to 75 ms.	RW	1	1'b0
7	rsvd	Reserved	—	1	1'b0

9.74. 64B/66B PCS RX RSFEC AM Period Register0 (reg68)

Table 9.77. 64B/66B PCS RX RSFEC AM Period Register0 (reg68) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	am_period__7_0	RSFEC alignment marker interval. Calculated in the following manner: (5280 × n / 40) – 1; where n is the number of RSFEC codeword. • Hardware: 18'h20FFF (1024 RSFEC codeword) • Simulation: 18'h20F (4 RSFEC codeword) Rx RSFEC AM period (bit 0 till bit 7)	RW	8	8'b11111111

9.75. 64B/66B PCS RX RSFEC AM Period Register1 (reg69)

Table 9.78. 64B/66B PCS RX RSFEC AM Period Register1 (reg69) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	am_period__15_8	RSFEC alignment marker interval. Calculated in the following manner: (5280 × n / 40) – 1; where n is the number of RSFEC codeword. • Hardware: 18'h20FFF (1024 RSFEC codeword) • Simulation: 18'h20F (4 RSFEC codeword) Rx RSFEC AM period (bit 8 till bit 15)	RW	8	8'b00001111

9.76. 64B/66B PCS RX RSFEC AM Period Register2 (reg6A)

Table 9.79. 64B/66B PCS RX RSFEC AM Period Register2 (reg6A) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	am_period__17_16	RSFEC alignment marker interval. Calculated in the following manner: (5280 × n / 40) – 1; where n is the number of RSFEC codeword. • Hardware: 18'h20FFF (1024 RSFEC codeword) • Simulation: 18'h20F (4 RSFEC codeword) Rx RSFEC AM period (bit 16 till bit 17)	RW	2	2'b10
7:2	rsvd	Reserved	—	6	6'b000000

9.77. 64B/66B PCS RX FCFEC AM Period Register (reg6B)

Table 9.80. 64B/66B PCS RX FCFEC AM Period Register (reg6B) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	fcfec_am_period	The FCFEC alignment marker interval which is used for DL measurement. The interval must match the pulse interval configured in the DL pulse gen. The recommended value is as shown below: 10/25G or soft 40G PCS: 9 (equivalent to the interval of 10x2112 bit time) E100 hard 40G PCS: 7 (equivalent to the interval of 8x2112 bit time)	RW	8	8'b00001001

9.78. 64B/66B PCS CTC Water Line Register HIGH (reg6C)

Table 9.81. 64B/66B PCS CTC Water Line Register HIGH (reg6C) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	ctc_64b66b_high	The difference between two pointers to assert almost full flag for clock frequency compensation.	RW	8	8'b00000100

9.79. 64B/66B PCS CTC Water Line Register LOW (reg6D)

Table 9.82. 64B/66B PCS CTC Water Line Register LOW (reg6D) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	ctc_64b66b_low	The difference between two pointers to assert almost empty flag for clock frequency compensation.	RW	8	8'b00000100

9.80. 64B/66B PCS RX Gearbox FIFO Setting Register HIGH (reg6E)

Table 9.83. 64B/66B PCS RX Gearbox FIFO Setting Register HIGH (reg6E) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rx_gbx_high	The difference between two pointers to assert almost full flag for Rx gearbox FIFO.	RW	8	8'b00000100

9.81. 64B/66B PCS RX Gearbox FIFO Setting Register LOW (reg6F)

Table 9.84. 64B/66B PCS RX Gearbox FIFO Setting Register LOW (reg6F) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rx_gbx_low	The difference between two pointers to assert almost empty flag for Rx gearbox FIFO.	RW	8	8'b00001000

9.82. 64B/66B PCS RSFEC RX Gearbox FIFO Setting Register HIGH (reg70)

Table 9.85. 64B/66B PCS RSFEC RX Gearbox FIFO Setting Register HIGH (reg70) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_rx_gbx_high	The difference between two pointers to assert almost full flag for RSFEC Rx gearbox.	RW	8	8'b00000100

9.83. 64B/66B PCS RSFEC RX Gearbox FIFO Setting Register LOW (reg71)

Table 9.86. 64B/66B PCS RSFEC RX Gearbox FIFO Setting Register LOW (reg71) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_rx_gbx_low	The difference between two pointers to assert almost empty flag for RSFEC Rx gearbox.	RW	8	8'b00000100

9.84. 64B/66B PCS RX Path Status Register (reg72)

Table 9.87. 64B/66B PCS RX Path Status Register (reg72) Descriptions

Field	Name	Description	Access	Width	Reset
0	rx_blk_lock	The high level of this register indicates the block lock is achieved.	RO	1	1'b0
1	rx_hi_ber	The high level of this register indicates High BER is detected.	RO	1	1'b0
2	rx_fec_lock	The high level of this register indicates the FEC (RSFEC or FCFEC) lock is achieved.	RO	1	1'b0
3	rsfec_hi_ser	High symbol error rate. • Applicable only for 64b66b Rx path operates in RSFEC mode and rsfec_rx_bei is set to 1'b1. • Set for a period of 60 ms to 75ms when the number of symbol errors in a block of 8192 RSFEC codewords exceeds 417.	RO	1	1'b0
7:4	rsvd	Reserved	—	4	4'b0000

9.85. 64B/66B PCS Block Align Shift Register (reg74)

Table 9.88. 64B/66B PCS Block Align Shift Register (reg74) Descriptions

Field	Name	Description	Access	Width	Reset
6:0	balign_64b66b_shift	The bit shifting of block alignment.	RO	7	7'b0000000
7	rsvd	Reserved	—	1	1'b0

9.86. 64B/66B PCS RSFEC Align Shift Register0 (reg76)

Table 9.89. 64B/66B PCS RSFEC Align Shift Register0 (reg76) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_align_shift__7_0	The bit shifting of RSFEC alignment (bit 0 till bit 7).	RO	8	8'b00000000

9.87. 64B/66B PCS RSFEC Align Shift Register1 (reg77)

Table 9.90. 64B/66B PCS RSFEC Align Shift Register1 (reg77) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_align_shift__15_8	The bit shifting of RSFEC alignment (bit 8 till bit 15).	RO	8	8'b00000000

9.88. 64B/66B PCS FCFEC Align Shift Register0 (reg78)

Table 9.91. 64B/66B PCS FCFEC Align Shift Register0 (reg78) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	fcfec_align_shift__7_0	The bit shifting of FCFEC alignment (bit 0 till bit 7).	RO	8	8'b00000000

9.89. 64B/66B PCS FCFEC Align Shift Register1 (reg79)

Table 9.92. 64B/66B PCS FCFEC Align Shift Register1 (reg79) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	fcfec_align_shift__15_8	The bit shifting of FCFEC alignment (bit 8 till bit 15).	RO	8	8'b00000000

9.90. 10GBASE-R Error Counter Clear Register (reg7A)

Table 9.93. 10GBASE-R Error Counter Clear Register (reg7A) Descriptions

Field	Name	Description	Access	Width	Reset
0	ber_count_clr_int	A write pulse generated to indicate the clear of ber_count.	WOP	1	1'b0
1	errored_block_count_clr_int	A write pulse generated to indicate the clear of errored_block_count.	WOP	1	1'b0
2	rsfec_corrected_sym_count_clr_int	A write pulse generated to indicate the clear of RSFEC rsfec_corrected_sym_count.	WOP	1	1'b0
3	rsfec_corrected_error_count_clr_int	A write pulse generated to indicate the clear of RSFEC rsfec_corrected_error_count.	WOP	1	1'b0
4	rsfec_uncorrected_error_count_clr_int	A write pulse generated to indicate the clear of RSFEC rsfec_uncorrected_error_count.	WOP	1	1'b0
5	fcfec_corrected_error_count_clr_int	A write pulse generated to indicate the clear of FCFEC fcfec_corrected_error_count.	WOP	1	1'b0
6	fcfec_uncorrected_error_count_clr_int	A write pulse generated to indicate the clear of FCFEC fcfec_uncorrected_error_count.	WOP	1	1'b0
7	tp_error_cnt_clr_int	A write pulse generated to indicate the clear of tp_error_cnt.	WOP	1	1'b0

9.91. 10GBASE-R Block Error Counter Register0 (reg7C)

Table 9.94. 10GBASE-R Block Error Counter Register0 (reg7C) Descriptions

Field	Name	Description	Access	Width	Reset
5:0	ber_count	The BER counter is a 6-bit count as defined by the ber_count variable in 49.2.14.2 of IEEE 802.3 standard for 10GBASE-R. These bits are held at all ones in the case of overflow.	RO	6	6'b000000

Field	Name	Description	Access	Width	Reset
7:6	rsvd	Reserved	—	2	2'b00

9.92. 10GBASE-R Block Error Counter Register1 (reg7D)

Table 9.95. 10GBASE-R Block Error Counter Register1 (reg7D) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	errored_block_count	The errored blocks counter is an 8-bit count defined by the errored_block_count counter specified in 49.2.14.2 of IEEE 802.3 standard for 10GBASE-R. These bits are held at all ones in the case of overflow.	RO	8	8'b00000000

9.93. 10GBASE-R Block Error Counter Register2 (reg7E)

Table 9.96. 10GBASE-R Block Error Counter Register2 (reg7E) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_corrected_sym_count__7_0	RSFEC corrected sym count is an 16-bit counter that is incremented with RSFEC corrected symbols. These bits are held at all ones in the case of overflow.	RO	8	8'b00000000

9.94. 10GBASE-R Block Error Counter Register3 (reg7F)

Table 9.97. 10GBASE-R Block Error Counter Register3 (reg7F) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_corrected_sym_count__15_8	RSFEC corrected sym count is an 16-bit counter that is incremented with RSFEC corrected symbols. These bits are held at all ones in the case of overflow.	RO	8	8'b00000000

9.95. 10GBASE-R Block Error Counter Register4 (reg80)

Table 9.98. 10GBASE-R Block Error Counter Register4 (reg80) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_corrected_error_count	RSFEC corrected error count is an 8-bit counter that is incremented with RSFEC corrected error event. These bits are held at all ones in the case of overflow.	RO	8	8'b00000000

9.96. 10GBASE-R Block Error Counter Register5 (reg81)

Table 9.99. 10GBASE-R Block Error Counter Register5 (reg81) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_uncorrected_error_count	RSFEC uncorrected error count is an 8-bit counter that is incremented with RSFEC uncorrected error event. These bits are held at all ones in the case of overflow.	RO	8	8'b00000000

9.97. 10GBASE-R Block Error Counter Register6 (reg82)

Table 9.100. 10GBASE-R Block Error Counter Register6 (reg82) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	fcfec_corrected_error_count	FCFEC corrected error count is an 8-bit counter that is incremented with FCFEC corrected error event. These bits are held at all ones in the case of overflow.	RO	8	8'b00000000

9.98. 10GBASE-R Block Error Counter Register7 (reg83)

Table 9.101. 10GBASE-R Block Error Counter Register7 (reg83) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	fcfec_uncorrected_error_count	FCFEC uncorrected error count is an 8-bit counter that is incremented with FCFEC uncorrected error event. These bits are held at all ones in the case of overflow.	RO	8	8'b00000000

9.99. 10GBASE-R Test Pattern Seed A Register0 (reg84)

Table 9.102. 10GBASE-R Test Pattern Seed A Register0 (reg84) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_a__7_0	This register defines the 10GBASE-R PCS test pattern seed A.	RW	8	8'b00000000

9.100.10GBASE-R Test Pattern Seed A Register1 (reg85)

Table 9.103. 10GBASE-R Test Pattern Seed A Register1 (reg85) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_a__15_8	This register defines the 10GBASE-R PCS test pattern seed A.	RW	8	8'b00000000

9.101. 10GBASE-R Test Pattern Seed A Register2 (reg86)

Table 9.104. 10GBASE-R Test Pattern Seed A Register2 (reg86) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_a__23_16	This register defines the 10GBASE-R PCS test pattern seed A.	RW	8	8'b00000000

9.102. 10GBASE-R Test Pattern Seed A Register3 (reg87)

Table 9.105. 10GBASE-R Test Pattern Seed A Register3 (reg87) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_a__31_24	This register defines the 10GBASE-R PCS test pattern seed A.	RW	8	8'b00000000

9.103. 10GBASE-R Test Pattern Seed A Register4 (reg88)

Table 9.106. 10GBASE-R Test Pattern Seed A Register4 (reg88) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_a__39_32	This register defines the 10GBASE-R PCS test pattern seed A.	RW	8	8'b00000000

9.104. 10GBASE-R Test Pattern Seed A Register5 (reg89)

Table 9.107. 10GBASE-R Test Pattern Seed A Register5 (reg89) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_a__47_40	This register defines the 10GBASE-R PCS test pattern seed A.	RW	8	8'b00000000

9.105. 10GBASE-R Test Pattern Seed A Register6 (reg8A)

Table 9.108. 10GBASE-R Test Pattern Seed A Register6 (reg8A) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_a__55_48	This register defines the 10GBASE-R PCS test pattern seed A.	RW	8	8'b00000000

9.106. 10GBASE-R Test Pattern Seed A Register7 (reg8B)

Table 9.109. 10GBASE-R Test Pattern Seed A Register7 (reg8B) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	prtp_seed_a__57_56	This register defines the 10GBASE-R PCS test pattern seed A.	RW	2	2'b00
7:2	rsvd	Reserved	—	6	6'b000000

9.107. 10GBASE-R Test Pattern Seed B Register0 (reg8C)

Table 9.110. 10GBASE-R Test Pattern Seed B Register0 (reg8C) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_b__7_0	This register defines the 10GBASE-R PCS test pattern seed B.	RW	8	8'b00000000

9.108. 10GBASE-R Test Pattern Seed B Register1 (reg8D)

Table 9.111. 10GBASE-R Test Pattern Seed B Register1 (reg8D) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_b__15_8	This register defines the 10GBASE-R PCS test pattern seed B.	RW	8	8'b00000000

9.109. 10GBASE-R Test Pattern Seed B Register2 (reg8E)

Table 9.112. 10GBASE-R Test Pattern Seed B Register2 (reg8E) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_b__23_16	This register defines the 10GBASE-R PCS test pattern seed B.	RW	8	8'b00000000

9.110. 10GBASE-R Test Pattern Seed B Register3 (reg8F)

Table 9.113. 10GBASE-R Test Pattern Seed B Register3 (reg8F) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_b__31_24	This register defines the 10GBASE-PCS test pattern seed B.	RW	8	8'b00000000

9.111. 10GBASE-R Test Pattern Seed B Register4 (reg90)

Table 9.114. 10GBASE-R Test Pattern Seed B Register4 (reg90) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_b__39_32	This register defines the 10GBASE-R PCS test pattern seed B.	RW	8	8'b00000000

9.112. 10GBASE-R Test Pattern Seed B Register5 (reg91)

Table 9.115. 10GBASE-R Test Pattern Seed B Register5 (reg91) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_b__47_40	This register defines the 10GBASE-R PCS test pattern seed B.	RW	8	8'b00000000

9.113. 10GBASE-R Test Pattern Seed B Register6 (reg92)

Table 9.116. 10GBASE-R Test Pattern Seed B Register6 (reg92) Description

Field	Name	Description	Access	Width	Reset
7:0	prtp_seed_b__55_48	This register defines the 10GBASE-R PCS test pattern seed B.	RW	8	8'b00000000

9.114. 10GBASE-R Test Pattern Seed B Register7 (reg93)

Table 9.117. 10GBASE-R Test Pattern Seed B Register7 (reg93) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	prtp_seed_b__57_56	This register defines the 10GBASE-R PCS test pattern seed B.	RW	2	2'b00
7:2	rsvd	Reserved	—	6	6'b000000

9.115. 10GBASE-R Test Pattern Control Register0 (reg94)

Table 9.118. 10GBASE-R Test Pattern Control Register0 (reg94) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	tx_prtp_data_sel	Select the data pattern for pseudo-random test pattern. 11: Reserved 10: IDLE data pattern 01: Zeros data pattern 00: LF (Local Fault Ordered-set) data pattern	RW	2	2'b00
2	tx_prtp_test_sel	Select either square wave or pseudo random as test pattern. 1: Square wave test pattern 0: Pseudo random test pattern	RW	1	1'b0

Field	Name	Description	Access	Width	Reset
3	tx_prtp_en	1: Enables transmit test pattern 0: Disables transmit test pattern	RW	1	1'b0
6:4	tx_sw_pattern	Defines the square wave repeating pattern of n ones followed by n zeros where n is configurable between 4 and 11. 0: n = 4 1: n = 6 2: n = 8 3: n = 11 4, 5, 6, 7: reserved	RW	3	3'b000
7	rsvd	Reserved	—	1	1'b0

9.116. 10GBASE-R Test Pattern Control Register1 (reg95)

Table 9.119. 10GBASE-R Test Pattern Control Register1 (reg95) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	rx_prtp_data_sel	Use this register to select the data pattern for pseudo-random test pattern comparing. 11: Reserved 10: IDLE data pattern 01: Zeros data pattern 00: LF data pattern	RW	2	2'b00
2	rx_prtp_en	1: Enables receive test-pattern checking 0: Disables receive test-pattern checking	RW	1	1'b0
7:3	rsvd	Reserved	—	5	5'b00000

9.117. 10GBASE-R Test Pattern Error Control Register0 (reg96)

Table 9.120. 10GBASE-R Test Pattern Error Control Register0 (reg96) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	tp_error_cnt__7_0	The counter contains the number of errors received during a pattern test. These bits are held at all ones in the case of overflow. The test-pattern methodology is described in section 49.2.12 of the *IEEE 802.3* standard. This counter counts either block errors or bit errors dependent on the test mode.	RO	8	8'b00000000

9.118. 10GBASE-R Test Pattern Error Control Register1 (reg97)

Table 9.121. 10GBASE-R Test Pattern Error Control Register1 (reg97) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	tp_error_cnt__15_8	The counter contains the number of errors received during a pattern test. These bits are held at all ones in the case of overflow. The test-pattern methodology is described in section 49.2.12 of the *IEEE 802.3* standard. This counter counts either block errors or bit errors dependent on the test mode.	RO	8	8'b00000000

9.119. Lane Alignment Debug Control Register0 (reg98)

Table 9.122. Lane Alignment Debug Control Register0 (reg98) Descriptions

Field	Name	Description	Access	Width	Reset
0	lock_fifo_pointers	0: FIFO write and read pointers work normally 1: FIFO write and read pointers are locked for debugging	RW	1	1'b0
1	clear_enable	0 : Recovers status register 1 : Clears status register	RW	1	1'b0
7:2	rsvd	Reserved	—	6	6'b000000

9.120. Lane Alignment Debug Control Register1 (reg99)

Table 9.123. Lane Alignment Debug Control Register1 (reg99) Descriptions

Field	Name	Description	Access	Width	Reset
0	lalign_dcfifo_overflow_clr_int	0: Do not clear overflow status 1: Clear lane deskew dc FIFO overfow status	WOP	1	1'b0
1	lalign_dcfifo_underflow_clr_int	0: Do not clear underrflow status 1: Clear lane deskew dc FIFO underfow status	WOP	1	1'b0
7:2	rsvd	Reserved	—	6	6'b000000

9.121. Lane Alignment Debug Status Register0 (reg9A)

Table 9.124. Lane Alignment Debug Status Register0 (reg9A) Descriptions

Field	Name	Description	Access	Width	Reset
0	fifo_overflow_status	0: Overflow did not happen 1: Overflow happened	RO	1	1'b0
1	rrdy_triggered	0: rrdy has not been triggered 1: rrdy has been triggered	RO	1	1'b0
2	rden_triggered	0: rden has not been triggered 1: rden has been triggered	RO	1	1'b0
3	max_lskew_triggered	0 : max lskew has not been triggered 1 : max lskew has been triggered	RO	1	1'b0
5:4	state_status	rx_lane_deskew state	RO	2	
6	lalign_dcfifo_overflow	0: lane deskew DC FIFO overflow did not happen 1: lane deskew DC FIFO overflow happened	RO	1	1'b0
7	lalign_dcfifo_underflow	0: lane deskew DC FIFO underflow did not happen 1: lane deskew DC FIFO underflow happened	RO	1	1'b0

9.122. Lane Alignment Debug Status Register1 (reg9B)

Table 9.125. Lane Alignment Debug Status Register1 (reg9B) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	fifo_wr_addr	FIFO write address	RO	8	8'b00000000

9.123. Lane Alignment Debug Status Register2 (reg9C)

Table 9.126. Lane Alignment Debug Status Register2 (reg9C) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	fifo_rd_addr	FIFO read address	RO	8	8'b00000000

9.124. Lane Alignment Debug Status Register3 (reg9D)

Table 9.127. Lane Alignment Debug Status Register3 (reg9D) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	max_lskew_cnt	Count the number of cycles spent in the detected state.	RO	8	8'b00000000

9.125. Clock Frequency Compensation Debug Control Register (reg9E)

Table 9.128. Clock Frequency Compensation Debug Control Register (reg9E) Descriptions

Field	Name	Description	Access	Width	Reset
0	enable_msb_comma	Used in Clock Frequency Compensation debug.	RW	1	1'b1
7:1	rsvd	Reserved	—	7	7'b0000000

9.126. Clock Frequency Compensation Debug Control Register 2 (reg9F)

Table 9.129. Clock Frequency Compensation Debug Control Register 2 (reg9F) Descriptions

Field	Name	Description	Access	Width	Reset
0	ctc_overflow_clr_int	0: Do not clear ctc fifo overflow 1: Clear ctc fifo overflow	WOP	1	1'b0
1	ctc_underflow_clr_int	0: Do not clear ctc fifo underflow 1: Clear ctc fifo underflow	WOP	1	1'b0
7:2	rsvd	Reserved	—	6	6'b000000

9.127. Loop-back Mode Control Register (regA0)

Table 9.130. Loop-back Mode Control Register (regA0) Descriptions

Field	Name	Description	Access	Width	Reset
0	near_lpbk_en	In 8B/10B PCS mode: 1: Enables near end parallel loopback 0: Do not enable parallel loopback In 64B66B PCS mode: 1: Enables loopback A 0: Do not enable loopback A	RW	1	1'b0
1	far_lpbk_en	In 8B/10B PCS mode: 1: Enables far end parallel loopback 0: Do not enable far end parallel loopback In 64B/66B PCS mode: 1: Enables loopback C	RW	1	1'b0

Field	Name	Description	Access	Width	Reset
		0: Do not enable loopback C			
7:2	rsvd	Reserved	—	6	6'b000000

9.128. FIFO Debug Control Register0 (regA2)

Table 9.131. FIFO Debug Control Register0 (regA2) Descriptions

Field	Name	Description	Access	Width	Reset
0	ctc_fifo_overflow_clr_int	A write pulse generated to indicate the clear of ctc_fifo_overflow.	WOP	1	1'b0
1	ctc_fifo_underflow_clr_int	A write pulse generated to indicate the clear of ctc_fifo_underflow.	WOP	1	1'b0
2	tx_gbx_fifo_overflow_clr_int	A write pulse generated to indicate the clear of tx_gbx_fifo_overflow.	WOP	1	1'b0
3	tx_gbx_fifo_underflow_clr_int	A write pulse generated to indicate the clear of tx_gbx_fifo_underflow.	WOP	1	1'b0
4	rx_gbx_fifo_overflow_clr_int	A write pulse generated to indicate the clear of rx_gbx_fifo_overflow.	WOP	1	1'b0
5	rx_gbx_fifo_underflow_clr_int	A write pulse generated to indicate the clear of ctc_fifo_overflow.	WOP	1	1'b0
6	rsfec_tx_gbx_fifo_overflow_clr_int	A write pulse generated to indicate the clear of rsfec_tx_gbx_fifo_overflow.	WOP	1	1'b0
7	rsfec_tx_gbx_fifo_underflow_clr_int	A write pulse generated to indicate the clear of rsfec_tx_gbx_fifo_underflow.	WOP	1	1'b0

9.129. FIFO Debug Control Register1 (regA3)

Table 9.132. FIFO Debug Control Register1 (regA3) Descriptions

Field	Name	Description	Access	Width	Reset
0	rsfec_rx_gbx_fifo_overflow_clr_int	A write pulse generated to indicate the clear of rsfec_rx_gbx_fifo_overflow.	WOP	1	1'b0
1	rsfec_rx_gbx_fifo_underflow_clr_int	A write pulse generated to indicate the clear of rsfec_rx_gbx_fifo_underflow.	WOP	1	1'b0
7:2	rsvd	Reserved	—	6	6'b000000

9.130. FIFO Debug Status Register0 (regA4)

Table 9.133. FIFO Debug Status Register0 (regA4) Descriptions

Field	Name	Description	Access	Width	Reset
0	ctc_fifo_overflow	CTC FIFO overflow	RO	1	1'b0
1	ctc_fifo_underflow	CTC FIFO underflow	RO	1	1'b0
2	tx_gbx_fifo_overflow	Tx gearbox FIFO overflow	RO	1	1'b0
3	tx_gbx_fifo_underflow	Tx gearbox FIFO underflow	RO	1	1'b0
4	rx_gbx_fifo_overflow	Rx gearbox FIFO overflow	RO	1	1'b0
5	rx_gbx_fifo_underflow	Rx gearbox FIFO underflow	RO	1	1'b0
6	rsfec_tx_gbx_fifo_overflow	RSFEC Tx gearbox FIFO overflow	RO	1	1'b0
7	rsfec_tx_gbx_fifo_underflow	RSFEC Tx gearbox FIFO underflow	RO	1	1'b0

9.131. FIFO Debug Status Register1 (regA5)

Table 9.134. FIFO Debug Status Register1 (regA5) Descriptions

Field	Name	Description	Access	Width	Reset
0	rsfec_rx_gbx_fifo_overflow	RSFEC Rx gearbox FIFO overflow	RO	1	1'b0
1	rsfec_rx_gbx_fifo_underflow	RSFEC Rx gearbox FIFO underflow	RO	1	1'b0
7:2	rsvd	Reserved	—	6	6'b000000

9.132. MPPCS BIST Control Register0 (regA6)

Table 9.135. MPPCS BIST Control Register0 (regA6) Descriptions

Field	Name	Description	Access	Width	Reset
0	bist_en	1: Enables MPCS BIST 0: Do not enable MPCS BIST	RW	1	1'b0
1	bist_mode	1: Continuous BIST Mode 0: Timed BIST mode	RW	1	1'b0
2	bist_bus8bit_sel	1: 8-bit data BIST 0: 10-bit data BIST	RW	1	1'b0
3	bist_bypass_tx_gate	0: Start to send BIST data after finding the head 1: Force to send BIST data whether the head is found or not	RW	1	1'b0
6:4	bist_ptn_sel	BIST pattern selection. 000: PRBS7 001: PRBS11 010: PRBS23 011: PRBS31 100: PRBS15 101: Constant_Value 1 110: Constant_Value 2 111: Alternate Constant Values 1 and 2	RW	3	3'b000
7	rsvd	Reserved	—	1	1'b0

9.133. MPPCS BIST Control Register1 (regA7)

Table 9.136. MPPCS BIST Control Register1 (regA7) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	bist_sync_head_req	BIST sync header counter selection. 00: 5 01: 8 10: 14 11: 24	RW	2	2'b00
3:2	bist_time_sel	BIST time selection. 00: 5e+8 cycles 01: 5e+9 cycles 10: 5e+6 cycles	RW	2	2'b00

Field	Name	Description	Access	Width	Reset
		11: 1e+5 cycles			
5:4	bist_res_sel	BIST acceptable resolution selection. 00: = no error 01: < 2 errors 10: < 16 errors 11: < 128 errors	RW	2	2'b00
7:6	rsvd	Reserved	—	2	2'b00

9.134. User Defined BIST Constant 1 Register0 (regA8)

Table 9.137. User Defined BIST Constant 1 Register0 (regA8) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	udbc1__7_0	Lower 8 bits of the 10b User Defined BIST Constant value pattern 1.	RW	8	8'b00000000

9.135. User Defined BIST Constant 1 Register1 (regA9)

Table 9.138. User Defined BIST Constant 1 Register1 (regA9) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	udbc1__17_10	Lower 8 bits of the 10b User Defined BIST Constant value pattern 1. Used only in 2-byte internal bus width mode.	RW	8	8'b00000000

9.136. User Defined BIST Constant 1 Register MSB (regAA)

Table 9.139. User Defined BIST Constant 1 Register MSB (regAA) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	udbc1__9_8	Upper 2 bits of the 10b User Defined BIST Constant value pattern 1.	RW	2	2'b00
3:2	udbc1__19_18	Upper 2 bits of the 10b User Defined BIST Constant value pattern 1. Used only in 2-byte internal bus width mode.	RW	2	2'b00
7:4	rsvd	Reserved	—	4	4'b0000

9.137. User Defined BIST Constant 2 Register0 (regAC)

Table 9.140. User Defined BIST Constant 2 Register0 (regAC) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	udbc2__7_0	Lower 8 bits of the 10b User Defined BIST Constant value pattern 2.	RW	8	8'b00000000

9.138. User Defined BIST Constant 2 Register1 (regAD)

Table 9.141. User Defined BIST Constant 2 Register1 (regAD) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	udbc2__17_10	Lower 8 bits of the 10b User Defined BIST Constant value pattern 2. Used only in 2-byte internal bus width mode.	RW	8	8'b00000000

9.139. User Defined BIST Constant 2 Register MSB (regAE)

Table 9.142. User Defined BIST Constant 2 Register MSB (regAE) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	udbc2__9_8	Upper 2 bits of the 10b User Defined BIST Constant value pattern 2.	RW	2	2'b00
3:2	udbc2__19_18	Upper 2 bits of the 10b User Defined BIST Constant value pattern 2. Used only in 2-byte internal bus width mode.	RW	2	2'b00
7:4	rsvd	Reserved	—	4	4'b0000

9.140. BIST Status Register0 (regB0)

Table 9.143. BIST Status Register0 (regB0) Descriptions

Field	Name	Description	Access	Width	Reset
0	bist_time_out	1: BIST is timed out but it is still in *sync header detection* stage 0: BIST is ongoing or not started	RO	1	1'b0
1	bist_done	1: BIST timer expires 0: BIST is ongoing or not started	RO	1	1'b0
2	bist_ok	1: no error for bist_res_sel=0, 2 errors for bist_res_sel=1, <16 errors for bist_res_sel=2 <128 errors for bist_res_sel=3 0: BIST is timed out or not started	RO	1	1'b0
7:3	bist_err_cnt__4_0	BIST error number	RO	5	5'b00000

9.141. BIST Status Register1 (regB1)

Table 9.144. BIST Status Register1 (regB1) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	bist_err_cnt__12_5	BIST error number	RO	8	8'b00000000

9.142. PMA Direct Status Register B2 (regB2)

Table 9.145. PMA Direct Status Register B2 (regB2) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	pma_dir_cmn_fifo_dl_shift	Number of byte shifting happen in the common FIFO. The value could go up to 3 for the gear ratio of 4.	RO	2	2'b00
7:2	rsvd	Reserved	—	6	6'b000000

9.143. TX DL Control Register1 (regB4)

Table 9.146. TX DL Control Register1 (regB4) Descriptions

Field	Name	Description	Access	Width	Reset
0	tx_dl_en	Enable bit for DL logic. Can be used to restart DL measurement process by	RW	1	1'b0

Field	Name	Description	Access	Width	Reset
		writing a ‘b0 and followed by a ‘b1. 1: Enables delay measurement 0: Disables delay measurement			
3:1	tx_sample_size	Number of samples used for delay averaging. 3’b000: 16 samples 3’b001: 32 samples 3’b010: 64 samples 3’b011: 128 samples 3’b100: 256 samples 3’b101: 512 samples 3’b110: 1024 samples 3’b111: 2048 samples	RW	3	3'b010
5:4	tx_pulse_sel	Selection for which pulse to be used to stop the delay averager counter. In Tx direction, stop pulse is async AM from PMA; while in Rx direction, stop pulse is AM bit from the output of common FIFO. 2’b00: 1st received stop pulse after counter start. 2’b01: 2nd received stop pulse after counter start. 2’b10: 3rd received stop pulse after counter start. 2’b11: 4th received stop pulse after counter start.	RW	2	2'b00
6	tx_dbg_bypass_am	Debug feature which is used to bypass internal generated Tx AM pulse. You need to provide AM from fabric when this debug is enabled. 0: Disables debug feature 1: Enables debug feature Can be used for Tx gearbox deskew in 64/66 bonding mode.	RW	1	1'b0
7	rsvd	Reserved	—	1	1'b0

9.144. TX DL Control Register2 (regB6)

Table 9.147. TX DL Control Register2 (regB6) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	tx_dl_interval__7_0	Bit [7:0] of the interval it took for delay measurement to collect 1 sample. It is in the unit of data path clock cycles.	RW	8	8'b01000000

9.145. TX DL Control Register3 (regB7)

Table 9.148. TX DL Control Register3 (regB7) Descriptions

Field	Name	Description	Access	Width	Reset
3:0	tx_dl_interval__11_8	Bit [11:8] of the interval it took for delay measurement to collect 1 sample. It is in the unit of data path clock cycles.	RW	4	4'b0001
7:4	tx_dl_width	The width of the generated pulse in the unit of clock cycles.	RW	4	4'b1111

9.146. TX DL Status Register1 (regB8)

Table 9.149. TX DL Status Register1 (regB8) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	tx_dl_data__7_0	Bit[7:0] of delay value in the unit of dl_clk cycles. dl_data[7:0]: fractional clock cycles dl_data[17:8]: clock cycles	RO	8	8'b00000000

9.147. TX_DL Status Register2 (regB9)

Table 9.150. TX_DL Status Register2 (regB9) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	tx_dl_data__15_8	Bit[15:8] of delay value in the unit of dl_clk cycles. dl_data[7:0]: fractional clock cycles dl_data[17:8]: clock cycles	RO	8	8'b00000000

9.148. TX DL Status Register3 (regBA)

Table 9.151. TX DL Status Register3 (regBA) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	tx_dl_data__17_16	Bit[17:16] of delay value in the unit of dl_clk cycles. dl_data[7:0]: fractional clock cycles dl_data[17:8]: clock cycles	RO	2	2'b00
2	tx_dl_valid	Valid signal to indicate sr_dl_data carry a valid delay value	RO	1	1'b0
7:3	rsvd	Reserved	—	5	5'b00000

9.149. RX DL Control Register (regC0)

Table 9.152. RX DL Control Register (regC0) Descriptions

Field	Name	Description	Access	Width	Reset
0	rx_dl_en	Enable bit for DL logic. Can be used to restart DL measurement process by writing a ‘b0 and followed by a ‘b1. 1: Enables delay measurement 0: Disables delay measurement	RW	1	1'b0
3:1	rx_sample_size	Number of samples used for delay averaging. 3’b000: 16 samples 3’b001: 32 samples 3’b010: 64 samples 3’b011: 128 samples 3’b100: 256 samples 3’b101: 512 samples 3’b110: 1024 samples 3’b111: 2048 samples	RW	3	3'b010

Field	Name	Description	Access	Width	Reset
5:4	rx_pulse_sel	Selection for which pulse to be used to stop the delay averager counter. In Tx direction, stop pulse is async AM from PMA; while in Rx direction, stop pulse is AM bit from the output of common FIFO. 2'b00: First received stop pulse after counter start 2'b01: Second received stop pulse after counter start 2'b10: Third received stop pulse after counter start 2'b11: Fourth received stop pulse after counter start	RW	2	2'b00
7:6	rsvd	Reserved	—	2	2'b00

9.150. RX DL Control Register2 (regC2)

Table 9.153. RX DL Control Register2 (regC2) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rx_dl_interval__7_0	Bit [7:0] of the interval it took for delay measurement to collect 1 sample. It is in the unit of data path clock cycles.	RW	8	8'b10010100

9.151. RX DL Control Register3 (regC3)

Table 9.154. RX DL Control Register3 (regC3) Descriptions

Field	Name	Description	Access	Width	Reset
3:0	rx_dl_interval__11_8	Bit [11:8] of the interval it took for delay measurement to collect 1 sample. It is in the unit of data path clock cycles.	RW	4	4'b0010
7:4	rsvd (**all bits High**)	The width of the generated pulse in the unit of clock cycles.	RW	4	4'b1111

Note: Warning on bit[7:4]: Engineering Mode: Defaulted to 4'b1111. Always use 4'b1111. Any value other than 4'b1111 for bit[7:4] leads to undesired behavior.

9.152. RX DL Status Register1 (regC4)

Table 9.155. RX DL Status Register1 (regC4) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rx_dl_data__7_0	Bit[7:0] of delay value in the unit of dl_clk cycles. dl_data[7:0]: fractional clock cycles dl_data[17:8]: clock cycles	RO	8	8'b00000000

9.153. RX DL Status Register2 (regC5)

Table 9.156. RX DL Status Register2 (regC5) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rx_dl_data__15_8	Bit[15:8] of delay value in the unit of dl_clk cycles. dl_data[7:0]: fractional clock cycles dl_data[17:8]: clock cycles	RO	8	8'b00000000

9.154. RX DL Status Register3 (regC6)

Table 9.157. RX DL Status Register3 (regC6) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	rx_dl_data__17_16	Bit[17:16] of delay value in the unit of dl_clk cycles. dl_data[7:0]: fractional clock cycles dl_data[17:8]: clock cycles	RO	2	2'b00
2	rx_dl_valid	Valid signal to indicate sr_dl_data carries a valid delay value.	RO	1	1'b0
7:3	rsvd	Reserved	—	5	5'b00000

9.155. RX DL Status Register4 (regC7)

Table 9.158. RX DL Status Register4 (regC7) Descriptions

Field	Name	Description	Access	Width	Reset
5:0	rx_shift_data	UI shifting that need to be compensated on measured delay value.	RO	6	6'b000000
6	rx_shift_valid	Valid signal to indicate shift_data carries a valid shift value.	RO	1	1'b0
7	rsvd	Reserved	—	1	1'b0

9.156. 10GBASE-R RSFEC TX Error Injection Control Register0 (regD0)

Table 9.159. 10GBASE-R RSFEC TX Error Injection Control Register0 (regD0) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_cnt_limit	RSFEC error injection reference counter counts from 8'h0 till the specified limit and restart at 8'h0.	RW	8	8'b10000011

9.157. 10GBASE-R RSFEC TX Error Injection Control Register1 (regD2)

Table 9.160. 10GBASE-R RSFEC TX Error Injection Control Register1 (regD2) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_mask_cnt_0	RSFEC error injection reference counter value used to apply RSFEC Tx error injection mask 0.	RW	8	8'b00000000

9.158. 10GBASE-R RSFEC TX Error Injection Control Register2 (regD3)

Table 9.161. 10GBASE-R RSFEC TX Error Injection Control Register2 (regD3) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_mask_cnt_1	RSFEC error injection reference counter value used to apply RSFEC Tx error injection mask 1.	RW	8	8'b00000001

9.159. 10GBASE-R RSFEC TX Error Injection Control Register3 (regD4)

Table 9.162. 10GBASE-R RSFEC TX Error Injection Control Register3 (regD4) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_mask_0__7_0	RSFEC error injection mask 0. It is used to corrupt the	RW	8	8'b00000000

Field	Name	Description	Access	Width	Reset
		encoded RSFEC data by inverting it before it is sent to Tx PMA. Bit value 1 means corrupt the corresponding encoded RSFEC data.			

9.160. 10GBASE-R RSFEC TX Error Injection Control Register4 (regD5)

Table 9.163. 10GBASE-R RSFEC TX Error Injection Control Register4 (regD5) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_mask_0__15_8	RSFEC error injection mask 0. It is used to corrupt the encoded RSFEC data by inverting it before it is sent to Tx PMA. Bit value 1 means corrupt the corresponding encoded RSFEC data.	RW	8	8'b00000000

9.161. 10GBASE-R RSFEC TX Error Injection Control Register5 (regD6)

Table 9.164. 10GBASE-R RSFEC TX Error Injection Control Register5 (regD6) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_mask_0__23_16	RSFEC error injection mask 0. It is used to corrupt the encoded RSFEC data by inverting it before it is sent to Tx PMA. Bit value 1 means corrupt the corresponding encoded RSFEC data.	RW	8	8'b00000000

9.162. 10GBASE-R RSFEC TX Error Injection Control Register6 (regD7)

Table 9.165. 10GBASE-R RSFEC TX Error Injection Control Register6 (regD7) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_mask_0__31_24	RSFEC error injection mask 0. It is used to corrupt the encoded RSFEC data by inverting it before it is sent to Tx PMA. Bit value 1 means corrupt the corresponding encoded RSFEC data.	RW	8	8'b00000000

9.163. 10GBASE-R RSFEC TX Error Injection Control Register7 (regD8)

Table 9.166. 10GBASE-R RSFEC TX Error Injection Control Register7 (regD8) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_mask_0__39_32	RSFEC error injection mask 0. It is used to corrupt the encoded RSFEC data by inverting it before it is sent to Tx PMA. Bit value 1 means corrupt the corresponding encoded RSFEC data.	RW	8	8'b00000000

9.164. 10GBASE-R RSFEC TX Error Injection Control Register8 (regDA)

Table 9.167. 10GBASE-R RSFEC TX Error Injection Control Register8 (regDA) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_mask_1__7_0	RSFEC error injection mask 1. It is used to corrupt the	RW	8	8'b00000000

Field	Name	Description	Access	Width	Reset
		encoded RSFEC data by inverting it before it is sent to Tx PMA. Bit value 1 means corrupt the corresponding encoded RSFEC data.			

9.165. 10GBASE-R RSFEC TX Error Injection Control Register9 (regDB)

Table 9.168. 10GBASE-R RSFEC TX Error Injection Control Register9 (regDB) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_mask_1__15_8	RSFEC error injection mask 1. It is used to corrupt the encoded RSFEC data by inverting it before it is sent to Tx PMA. Bit value 1 means corrupt the corresponding encoded RSFEC data.	RW	8	8'b00000000

9.166. 10GBASE-R RSFEC TX Error Injection Control Register10 (regDC)

Table 9.169. 10GBASE-R RSFEC TX Error Injection Control Register10 (regDC) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_mask_1__23_16	RSFEC error injection mask 1. It is used to corrupt the encoded RSFEC data by inverting it before it is sent to Tx PMA. Bit value 1 means corrupt the corresponding encoded RSFEC data.	RW	8	8'b00000000

9.167. 10GBASE-R RSFEC TX Error Injection Control Register11 (regDD)

Table 9.170. 10GBASE-R RSFEC TX Error Injection Control Register11 (regDD) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_mask_1__31_24	RSFEC error injection mask 1. It is used to corrupt the encoded RSFEC data by inverting it before it is sent to Tx PMA. Bit value 1 means corrupt the corresponding encoded RSFEC data.	RW	8	8'b00000000

9.168. 10GBASE-R RSFEC TX Error Injection Control Register12 (regDE)

Table 9.171. 10GBASE-R RSFEC TX Error Injection Control Register12 (regDE) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	rsfec_tx_err_inj_mask_1__39_32	RSFEC error injection mask 1. It is used to corrupt the encoded RSFEC data by inverting it before it is sent to Tx PMA. Bit value 1 means corrupt the corresponding encoded RSFEC data.	RW	8	8'b00000000

9.169. PHY_LANE_RX_REQUEST_REQACK0 (reg144)

Table 9.172. PHY_LANE_RX_REQUEST_REQACK0 (reg144) Descriptions

Field	Name	Description	Access	Width	Reset
0	phy_rx_req	Receiver request.	RW	1	1'b0

Field	Name	Description	Access	Width	Reset
		This signal indicates a new receiver setting request. This signal must stay high until the phy_rx_ack output is asserted. After phy_rx_ack is asserted, phy_rx_req must be de-asserted. The PHY captures the following input signals when phy_rx_req is asserted: • phy_rx_pstate • phy_rx_lpd • phy_rx_rate • phy_rx_width • phy_rx_dfe_bypass • phy_rx_eq_att_lvl • phy_rx_eq_afe_rate • phy_rx_eq_vga1_gain • phy_rx_eq_vga2_gain • phy_rx_eq_ctle_pole • phy_rx_eq_ctle_boost • phy_rx_eq_dfe_tap1 • phy_rx_ref_ld_val • phy_rx_vco_ld_val • phy_rx_adapt_sel • phy_rx_cdr_vco_config • phy_rx_delta_iq • phy_rx_margin_iq • phy_rx_misc • phy_rx_dcc_ctrl_range • phy_rx_recal_force_en • phy_rx_recal_skip_en • phy_rx_recal_bank_sel These signals must be stable before assertion of phy_rx_req and remain stable until phy_rx_req is de-asserted. Any combination of D300 changes in these signals is valid for a single request, and the completion of all changes are indicated by the assertion of phy_rx_ack.			
7:1	rsvd	Reserved	—	7	7'b0000000

9.170. PHY_LANE_RX_REQUEST_REQACK1 (reg145)

Table 9.173. PHY_LANE_RX_REQUEST_REQACK1 (reg145) Descriptions

Field	Name	Description	Access	Width	Reset
0	phy_rx_ack	Receiver acknowledge. Assertion of this signal indicates the requested receiver setting is complete. This signal stays asserted until the phy_rx_req input is de-asserted. Once asserted, phy_rx_ack is de-asserted when phy_rx_req is de-asserted.	RO	1	1'b1
7:1	rsvd	Reserved	—	7	7'b0000000

9.171. PHY_LANE_RX_ADAPTATION_REQACK0 (reg146)

Table 9.174. PHY_LANE_RX_ADAPTATION_REQACK0 (reg146) Descriptions

Field	Name	Description	Access	Width	Reset
0	phy_rx_adapt_req	Receiver adaptation request. This input indicates a request for the receiver to adapt to incoming data. This signal must remain asserted until phy_rx_adapt_ack is asserted, after which phy_rx_adapt_req can be de-asserted. Aborting phy_rx_adapt_req by negating before phy_rx_adapt_ack is asserted or by asserting phy_rx_reset may leave the receiver with a less than optimal settings.	RW	1	1'b0
7:1	rsvd	Reserved	—	7	7'b0000000

9.172. PHY_LANE_RX_ADAPTATION_REQACK1 (reg147)

Table 9.175. PHY_LANE_RX_ADAPTATION_REQACK1 (reg147) Descriptions

Field	Name	Description	Access	Width	Reset
0	phy_rx_adapt_ack	Receiver adaptation acknowledge. Assertion of this output indicates completion of the receiver adaptation sequence and that values on phy_rx_adapt_fom, phy_rx_txmain_dir, phy_rx_txpre_dir, and phy_rx_txpost_dir are valid. This signal remains asserted until phy_rx_adapt_req is de-asserted, after which phy_rx_adapt_ack is de-asserted.	RO	1	1'b1
7:1	rsvd	Reserved	—	7	7'b0000000

9.173. PHY_LANE_RX_ADAPTATION (reg154)

Table 9.176. PHY_LANE_RX_ADAPTATION (reg154) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	phy_rx_adapt_mode	Rx Adaptation mode select. Selects an adaptation mode. An adaptation mode defines an adaptation algorithm with varying degrees of length and accuracy, which defines the length of the capture window, number of iterations, and update gains. For phy_rx_adapt_mode=0, setting=PCIe4/ESM0/ESM1. For phy_rx_adapt_mode=1, setting=PCIe3/ESM0/ESM1. For phy_rx_adapt_mode=2, setting=SHORT. For phy_rx_adapt_mode=3, setting=LONG Note that: • phy_rx_adapt_mode = 1 must be used for ESM0/ESM1 rates from 8 Gbps to 12.5 Gbps. • phy_rx_adapt_mode = 0 must be used for ESM0/ESM1 rates greater than 12.5 Gbps. This input must be set according to the values provided in the *PHY Usage and Configuration*.	RW	2	2'b00
2	phy_rx_adapt_in_prog	Receiver adaptation in progress. This input indicates that an adaptation is in progress. This signal must be asserted prior to the phy_rx_reset assertion of the first adaptation iteration. It must be de-asserted after the phy_rx_adapt_ack of the final adaptation iteration.	RW	1	1'b0

Field	Name	Description	Access	Width	Reset
3	phy_rx_adapt_cont	Receiver adaptation continuous operation. This input must be asserted if Rx continuous adaptation is required. If this signal is de-asserted, the receiver adaptation stops when the adaptation acknowledge (phy_rx_adapt_ack) is asserted; otherwise, the receiver continues to adapt. Note that this signal can be kept asserted. Continuous adaptation will only be performed in rates which support adaptation (that is, phy_rx_dfe_bypass - 1'b1) and that have completed startup adaptation.	RW	1	1'b0
4	phy_rx_offcan_cont	Receiver offset cancellation continuous operation. If continuous receiver offset cancellation is required, then assert this signal. If this signal is de-asserted, then offset cancellation runs only when the receiver exits the P2 power state.	RW	1	1'b0
7:5	rsvd	Reserved	—	3	3'b000

9.174. PHY_LANE_LOOPBACK (reg158)

Table 9.177. PHY_LANE_LOOPBACK (reg158) Descriptions

Field	Name	Description	Access	Width	Reset
0	phy_lane_rx2tx_par_lb_en	Parallel (Rx-to-Tx) loopback enable. When this signal is asserted, recovered parallel data from the receiver is looped back to the transmit serializer.	RW	1	1'b0
3:1	rsvd	Reserved	—	3	
4	phy_lane_tx2rx_ser_lb_en	Analog serial (Tx-to-Rx) loopback enable. When this signal is asserted, data from the Tx pre-driver is looped back to the Rx Analog Front End (AFE); signal detection (SIGDET) is bypassed and based on the phy_tx_data_en input so that phy_rx_sigdet_lf = phy_tx_data_en.	RW	1	1'b0
7:5	rsvd	Reserved	—	3	3'b000

9.175. PHY_LANE_TX_REQUEST_REQACK0 (reg15C)

Table 9.178. PHY_LANE_TX_REQUEST_REQACK0 (reg15C) Descriptions

Field	Name	Description	Access	Width	Reset
0	phy_tx_req	Transmitter request. This signal indicates a new transmitter setting request. This signal must be asserted only when phy_tx_ack is low and must stay high until the phy_tx_ack output is asserted. After phy_tx_ack is asserted, phy_tx_req must be de-asserted.	RW	1	1'b0
1	phy_tx_detrx_req	Transmitter Rx-Detection request. This signal indicates an Rx-detection request. When phy_tx_ack is asserted then, the detection result on phy_tx_detrx_result is valid. Note that: • This signal must stay high until the phy_tx_ack output is asserted, then phy_tx_detrx_req must be de-asserted. • Rx detection is only valid when the Tx is in the P1 power state.	RW	1	1'b0
7:2	rsvd	Reserved	—	6	6'b000000

9.176. PHY_LANE_TX_REQUEST_REQACK1 (reg15D)

Table 9.179. PHY_LANE_TX_REQUEST_REQACK1 (reg15D) Descriptions

Field	Name	Description	Access	Width	Reset
0	phy_tx_ack	Transmitter acknowledge. The assertion of this signal indicates the requested transmitter setting (phy_tx_req) is complete or the requested Rx detection operation is complete. This signal stays asserted until the request input (phy_tx_req or phy_tx_detrx_req) is de-asserted, after which phy_tx_ack is de-asserted.	RO	1	1'b1
7:1	rsvd	Reserved	—	7	7'b0000000

9.177. PHY_LANE_RX_ADAPT_STATUS0 (reg168)

Table 9.180. PHY_LANE_RX_ADAPT_STATUS0 (reg168) Descriptions

Field	Name	Description	Access	Width	Reset
7:0	phy_rx_adapt_fom	Receiver figure of merit. An output indicating the quality of the received data eye. A higher value of phy_rx_adapt_fom indicates better link equalization. 8'd0 indicating worst equalization setting and 8'd255 indicating the best equalization setting. This output is valid only when phy_rx_adapt_ack is asserted.	RO	8	8'b00000000

9.178. PHY_LANE_RX_ADAPT_STATUS1 (reg169)

Table 9.181. PHY_LANE_RX_ADAPT_STATUS1 (reg169) Descriptions

Field	Name	Description	Access	Width	Reset
1:0	phy_rx_txpre_dir	Direction change for the far-end Tx main, pre- and post-equalization coefficients. The encoding for each signal is as follows: 2'b00: HOLD 2'b01: INC 2'b10: DEC 2'b11: Reserved. These outputs are valid only when phy_rx_adapt_ack is asserted. The PHY does not generate INC/DEC commands for the phy_rx_txmain_dir[1:0]. The INC/DEC definition for phy_rx_txpre_dir[1:0] and phy_rx_txpost_dir[1:0] may be opposite to the definition provided in certain protocol standards.	RO	2	2'b00
3:2	phy_rx_txmain_dir	Direction change for the far-end Tx main, pre- and post-equalization coefficients. The encoding for each signal is as follows: 2'b00: HOLD 2'b01: INC 2'b10: DEC 2'b11: Reserved. These outputs are valid only when phy_rx_adapt_ack is asserted. The PHY does not generate INC/DEC commands for the phy_rx_txmain_dir[1:0]. The	RO	2	2'b00

Field	Name	Description	Access	Width	Reset
		INC/DEC definition for phy_rx_txpre_dir[1:0] and phy_rx_txpost_dir[1:0] may be opposite to the definition provided in certain protocol standards.			
5:4	phy_rx_txpost_dir	Direction change for the far-end Tx main, pre- and post-equalization coefficients. The encoding for each signal is as follows: 2'b00: HOLD 2'b01: INC 2'b10: DEC 2'b11: Reserved. These outputs are valid only when phy_rx_adapt_ack is asserted. The PHY does not generate INC/DEC commands for the phy_rx_txmain_dir[1:0]. The INC/DEC definition for phy_rx_txpre_dir[1:0] and phy_rx_txpost_dir[1:0] may be opposite to the definition provided in certain protocol standards.	RO	2	2'b00
6	phy_tx_detrx_result	Transmitter Rx-Detection Result. This signal is valid when phy_tx_ack is asserted following a phy_tx_detrx_req request. 1'b0: Receiver not detected. 1'b1: Receiver detected.	RO	1	1'b0
7	rsvd	Reserved	—	1	1'b0

10. Protocol Mode

10.1. PCI Express Mode

PCI Express is a high performance, fully scalable, well-defined standard for a wide variety of computing and communications platforms. Being a packet based serial technology, PCI Express greatly reduces the number of required pins and simplifies board routing and manufacturing. PCI Express is a point-to-point technology, as opposed to the multi-drop bus in PCI. Each PCI Express device has the advantage of full duplex communication with its link partner to greatly increase overall system bandwidth.

In Lattice Nexus 2, only Quad0 integrates the PCIe Link Layer hard IP, which works with PCI Express PCS, PMA Controller and PMA. The channel cannot be configured as MPCS mode or EPCS mode when this channel is configured as PCI Express mode, considering all these modes share the PMA Controller and PMA channels.

PCIe Base Specification requires external AC-coupling. The recommended Capacitance are shown in Table 10.1. The detected state is the initial state after PCIe reset or power up. In this state, a device electrically detects if a receiver device is present at the far end of the link by observing the rate of change. And the device compares the time the line voltage to charge-up and the expected time. If a receiver device is attached, the charge time is much longer. The value and location of this AC-coupling capacitor affects the charge time during PCIe detect state. That is why the recommended capacitance range is so narrow.

Table 10.1. PCI Express Recommended AC Capacitance

PCIe Generation	Minimum	Typical	Maximum
Gen1	75 nF	—	265 nF
Gen2	75 nF	—	265 nF
Gen3	176 nF	—	265 nF
Gen4	176 nF	—	265 nF

PCI Express Base Specification defines a beacon signal inside the Link Training and Status State Machine (LTSSM). While a PCIe link is in L2 power state, its main power source and clock are turned off. An auxiliary voltage (V_{aux}) keeps a small part of the device working, including the wake-up logic. To signal a wake-up event, a downstream device can drive the beacon signal upstream to start the L2 exit sequence. A switch or bridge receiving a beacon signal on its downstream port must forward notification upstream by sending the beacon signal on its upstream port or by asserting the WAKE# pin.

However, Lattice Nexus 2 SERDES does not support the auxiliary power required by PCIe LTSSM L2 state. It is meaningless to support PCIe beacon signal or WAKE# signal generation and detection for unsupported LTSSM L2 state.

For more detailed information on Lattice Nexus 2 PCI Express features, function descriptions, and IP usage, refer to the PCIe X4 IP Core User Guide (FPGA-IPUG-02126).

10.2. Ethernet Mode

10.2.1. 1000BASE-X (GigE) Mode

The 1000BASE-X (or Gigabit Ethernet, GigE) mode of Lattice Nexus 2 SERDES/PCS fully supports from the serial I/O to the GMII/SGMII interface of the IEEE 802.3 1000BASE-X Gigabit Ethernet standard.

In 1000BASE-X mode, Lattice Nexus 2 SERDES/PCS supports the clock compensation and auto-negotiation features. The auto-negotiation control logic should be implemented in FPGA fabric to work with Lattice Nexus 2 SERDES/PCS.

Idle pattern insertion is required for clock compensation and auto-negotiation. The mpcs_anxmit_i signal should be asserted by auto-negotiation control logic to indicate the current state is GigE auto-negotiation state. In this state, the /C1/ and /C2/ ordered sets are replaced by /I2/ ordered sets periodically at Rx path, so that the Elastic Buffer gets the opportunity to perform clock compensation functionality. While auto-negotiating, the link partner transmits /C1/ and /C2/ ordered sets continuously.

The cordisp bit of mpcs_tx_data is used on the transmit side of the PCS to ensure that an Inter-Packet Gap (IPG) begins in the negative disparity state. Note that at the end of an Ethernet frame, the current disparity state of the transmitter can be either positive or negative, depending on the size and data content of the Ethernet frame.

However, from the FPGA fabric side of the PCS, the current disparity state of the PCS transmitter is unknown. This is where the cordisp bit signal comes into play. If the cordisp bit signal is asserted for one clock cycle upon entering an IPG, it forces the Tx path PCS to insert a /I1/ ordered set into the transmit data stream when the current disparity is positive. If the current disparity is negative, then no change is made to the transmit data stream.

From the FPGA fabric side of the PCS, the IPG is typically characterized by the continuous transmission of the /I2/ ordered set.

Note that in the PCS channel, /I2/ ordered set means the current disparity is to be preserved, and /I1/ ordered set means the current disparity state should be flipped. Therefore, it is possible to ensure that the IPG begins in a negative disparity state. If the disparity state before the IPG is negative, then a continuous stream of /I2/ ordered sets are transmitted during the IPG. If the disparity state before the IPG is positive, then followed by a continuous stream of /I2/ ordered sets, a single /I1/ ordered set is transmitted.

At the FPGA fabric side of the PCS, the IPG is always driven into the PCS with /I2/ ordered sets. The cordisp bit signal is asserted for one clock cycle, when the IPG first begins. If necessary, the PCS converts this /I2/ ordered set into a /I1/ ordered set. For the remainder of the IPG, /I2/ ordered sets should be driven into the PCS and the cordisp bit signal should remain de-asserted.

For example, if a continuous stream of 512 bytes of Ethernet frames and 512 bytes of /I/ ordered sets are set, the following can be observed:

- During the first IPG, all negative disparity /I2/ ordered sets are seen.
- During the next IPG, the period begins with positive disparity /I1/ ordered set. Then all the remaining ordered sets are negative disparity /I2/ ordered sets.
- During the next IPG, all negative disparity /I2/ ordered sets are seen.
- During the next IPG, all period begins with positive disparity /I1/ ordered set. Then all remaining ordered sets are negative disparity /I2/ ordered sets.

Table 10.2 shows the definition of the configuration ordered sets and IDLE ordered sets for Ethernet 1000BASE-X (GigE).

Table 10.2. GigE Configuration and IDLE Ordered Sets Definition

Configuration/IDLE	Code	Ordered Sets	Number of Code	Encoding
Configuration /C/	/C1/	Configuration 1	4	/K28.5//D21.5//Config_Reg/
	/C2/	Configuration 2	4	/K28.5//D2.2//Config_Reg/
IDLE /I/	/I1/	IDLE 1	2	/28.5//D5.6/
	/I2/	IDLE 2	2	/28.5//D16.2/

There is no specific requirement for GigE on AC-coupling or DC-coupling implementation. Lattice Nexus 2 SERDES/PCS supports both AC-coupling and DC-coupling link. A 100 nF AC-coupling capacitor is recommended to be used in an AC coupled link.

10.2.2. XAUI Mode

The XAUI mode of Lattice Nexus 2 SERDES/PCS is intended for the optional interface specified by IEEE802.3 between 10G Ethernet MAC and PHY. In XAUI mode, Lattice Nexus 2 SERDES/PCS is configured as four 2.5 Gbps lanes based on 8B/10B PCS. With Lattice XAUI IP core, the XAUI mode of Lattice Nexus 2 SERDES/PCS fully supports from serial I/O to the XGMII interface.

In XAUI mode, the transmit state machine inside 8B/10B PCS performs translation of XGMII idles to proper ‖A‖, ‖K‖, and ‖R‖ ordered sets according to the IEEE802.3 specifications. Table 10.3 shows the definition of the IDLE ordered sets for Ethernet XAUI.

Table 10.3. XAUI IDLE Ordered Sets Definition

Code	Ordered Sets	Number of Code	Encoding
‖A‖	Sync Column	4	/28.5//28.5//28.5//28.5/
‖K‖	Skip Column	4	/28.0//28.0//28.0//28.0/
‖R‖	Align Column	4	/28.3//28.3//28.3//28.3/

The XAUI receiver need be AC-coupled to the XAUI, and a 100 nF AC-coupling capacitor is recommended to be used in an AC-coupled link.

10.2.3. SGMII Mode and QSGMII Mode

The Serial Gigabit Media Independent Interface (SGMII) designed by Cisco is to convey network data, and port speed between a 10/100/1000 PHY and a MAC with significantly fewer signal pins than required for GMII. The SGMII interface can operate in both half and full duplex and at all ports speeds. The Quad Serial Gigabit Media Independent Interface (QSGMII) is designed to convey four ports of network data between Ethernet PHY and MAC like SGMII.

The SGMII mode and QSGMII mode of Lattice Nexus 2 SERDES/PCS fully supports from serial I/O to GMII interface. In SGMII mode, SERDES/PCS block is configured as single lane at 1.25 Gbps, based on 8B/10B PCS. In QSGMII mode, SERDES/PCS block is configured as single lane at 5 Gbps, based on 8B/10B PCS.

SGMII and QSGMII supports both AC and DC coupled operation. Lattice Nexus 2 SERDES/PCS supports both AC-coupling and DC-coupling link. A 100nF AC-coupling capacitor is recommended to be used in an AC coupled link.

10.2.4. 10GBASE-R Mode

The 10GBASE-R mode of Lattice Nexus 2 SERDES/PCS is intended for the connection between XGMII and 10 Gigabit serial I/O, in 10 Gigabit Ethernet applications specified by IEEE802.3. In 10GBASE-R mode, Lattice Nexus 2 SERDES/PCS is configured as single lane at 10.3125 Gbps, based on 64B/66B PCS.

The 64B/66B PCS encodes eight data octets or control characters, and 2-bit synchronization header into a block. Blocks containing control characters also contain a block type field. Data octets are labeled from D_0 to D_7. Control characters other than ordered sets (/O/), start control character (/S/) and termination control character (/T/) are labeled from C_0 to C_7. The control character for ordered sets are labeled as O_0 or O_4, as it is only valid on the first octet of the XGMII. The control character for start is labeled as S_0 or S_4 for the same reason. The control character for terminate is labeled from T_0 to T_7.

The first two bits of a block are the synchronization header (sync header). Blocks are either data blocks or control blocks. The sync header is 2’b01 for data blocks and 2’b10 for control blocks. Thus, there is always a transition between the first two bits of a block. The remainder of the block contains the payload. The payload is scrambled, and the sync header bypasses the scrambler. Therefore, the sync header is the only position in the block that always contains a transition. This feature of the code is used to obtain block synchronization.

Data blocks contain eight data characters. Control blocks begin with an 8-bit block type field that indicates the format of the remainder of the block. To control blocks containing a Start or Terminate character, the character is implied in the block type field. Other control characters are encoded in a 7-bit control code or a 4-bit /O/ Code. Each control block contains eight characters. Table 10.4 shows the format of the 64B/66B blocks.

Table 10.4. 64B/66B Blocks Formats

Input Data	64B/66B Block								
Data Block Format	**Sync**	**Block Payload**							
$D_0D_1D_2D_3D_4D_5D_6D_7$	01	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
Control Block Format	**Sync**	**Type**	**Payload**						
$C_0C_1C_2C_3C_4C_5C_6C_7$	10	8'h1E	C_0	C_1	C_2	C_3	C_4	C_5	C_6	C_7
$C_0C_1C_2C_3O_4D_5D_6D_7$	10	8'h2D	C_0	C_1	C_2	C_3	O_4	D_5	D_6	D_7
$C_0C_1C_2C_3S_4D_5D_6D_7$	10	8'h33	C_0	C_1	C_2	C_3		D_5	D_6	D_7
$O_0D_1D_2D_3S_4D_5D_6D_7$	10	8'h66	D_1	D_2	D_3	O_0		D_5	D_6	D_7
$O_0D_1D_2D_3O_4D_5D_6D_7$	10	8'h55	D_1	D_2	D_3	O_0	O_4	D_5	D_6	D_7
$S_0D_1D_2D_3O_4D_5D_6D_7$	10	8'h78	D_1	D_2	D_3	D_4	D_5	D_6	D_7
$O_0D_1D_2D_3C_4C_5C_6C_7$	10	8'h4B	D_1	D_2	D_3	O_0	C_4	C_5	C_6	C_7
$T_0C_1C_2C_3C_4C_5C_6C_7$	10	8'h87		C_1	C_2	C_3	C_4	C_5	C_6	C_7
$D_0T_1C_2C_3C_4C_5C_6C_7$	10	8'h99	D_0		C_2	C_3	C_4	C_5	C_6	C_7
$D_0D_1T_2C_3C_4C_5C_6C_7$	10	8'hAA	D_0	D_1		C_3	C_4	C_5	C_6	C_7
$D_0D_1D_2T_3C_4C_5C_6C_7$	10	8'hB4	D_0	D_1	D_2		C_4	C_5	C_6	C_7
$D_0D_1D_2D_3T_4C_5C_6C_7$	10	8'hCC	D_0	D_1	D_2	D_3		C_5	C_6	C_7
$D_0D_1D_2D_3D_4T_5C_6C_7$	10	8'hD2	D_0	D_1	D_2	D_3	D_4		C_6	C_7
$D_0D_1D_2D_3D_4D_5T_6C_7$	10	8'hE1	D_0	D_1	D_2	D_3	D_4	D_5		C_7
$D_0D_1D_2D_3D_4D_5D_6T_7$	10	8'hFF	D_0	D_1	D_2	D_3	D_4	D_5	D_6	

Lattice Nexus 2 SERDES/PCS 10GBASE-R mode does not support the Auto-Negotiation (AN), Training and Forward Error Correction (FEC) features required by the 10GBASE-KR backplane links. The optional WAN Interface Sublayer (WIS) part of the 10GBASE-R standard is not implemented by Lattice Nexus 2 SERDES/PCS.

10.3. SLVS-EC Mode

The Scalable Low Voltage Signaling with Embedded Clock (SLVS-EC) is a high-speed serial interface technology developed by Sony for the next generation high resolution CMOS Image Sensor (CIS). The SLVS-EC interface provides a unidirectional wide band pixel data transfer from CIS to a Digital Signal Processor (DSP) or other digital devices.

Lattice Nexus 2 SERDES/PCS supports up to eight lanes and up to 5 Gbps baud rate SLVS-EC receiver applications. Table 10.5 shows the baud grades supported by Lattice Nexus 2 SERDES/PCS. In SLVS-EC mode, SERDES/PCS block is configured as specific link rate within Baud Grade 1, 2 or 3, based on 8B/10B PCS.

Table 10.5. SLVS-EC Baud Rate

Baud Grade 1	Baud Grade 2	Baud Grade 3
1152 ~ 1250 Mbps	2304 ~ 2500 Mbps	4608 ~ 5000 Mbps

SLVS-EC, a kind of low-swing serial protocol, the minimum receiver input differential swing is 40 mV for baud grade 3. The interconnection between transmitter and receiver should be DC coupled, considering the differential swing is quite small. Table 10.6 shows the general parameters of receiver characteristics for SLVS-EC.

Table 10.6. General Parameters of Receiver Characteristics for SLVS-EC

Baud Rate	Parameters	Min	Typical	Max
Baud Grade 1 and 2	Receiver Input Differential Zero-Peak Voltage	65 mV	—	285 mV
	Receiver Input Common Mode Voltage	25 mV	—	300 mV
	Receiver Input Differential Resistance	80 Ω	—	110 Ω
Baud Grade 3	Receiver Input Differential Zero-Peak Voltage	40 mV	—	285 mV
	Receiver Input Common Mode Voltage	25 mV	—	300 mV
	Receiver Input Differential Resistance	80 Ω	—	110 Ω

10.4. DisplayPort Mode

DisplayPort (DP) is an industry standard to accommodate the growing broad adoption of digital display technology within the Personal Computer (PC) and Consumer Electronic (CE) industries. It consolidates internal and external connection methods to reduce device complexity, supports necessary features for key cross industry applications, and provides performance scalability to enable the next generation of displays featuring higher color depths, refresh rates, and display resolutions.

The DisplayPort mode of Lattice Nexus 2 SERDES/PCS supports up to four different DisplayPort link rate: RBR, HBR, HBR2, and HBR3. In DisplayPort mode, SERDES/PCS block is configured as specific link rate, based on 8B/10B PCS.

VESA DisplayPort Specification requires external AC-coupling. The recommended Capacitance are shown Table 10.7. All DisplayPort Main-Link lanes need be AC-coupled, and the AC-coupling capacitors need be placed on the transmitter side. The AC-coupling capacitors may also be placed on the receiver side for better signal performance.

Table 10.7. DisplayPort Recommend AC Capacitance

Link Rate	Minimum	Typical	Maximum
All	75 nF	—	265 nF

10.5. CoaXPress Mode

CoaXPress is defined by Japan Industrial Imaging Association (JIIA) to connect cameras and frame grabbers. It combines the simplicity of coaxial cable with state-of-art high speed data technology, allowing up to 12.5 Gbps data rate per cable, plus device control and power in the same cable.

CoaXPress is a point-to-point scalable interface that consists of one controller connection and optional extension connections. Each connection is associated with a coaxial cable. Each connection provides:

- high speed serial downstream connection at up to 12.5 Gbps;
- low speed serial upstream connection at up to 41.6 Mbps (or 20.83 Mbps).

In other words, both high speed serial downstream connection and low speed serial upstream connection share the same coaxial cable.

The CoaXPress mode of Lattice Nexus 2 SERDES/PCS supports up to five different CoaXPress link rate: CXP-1, CXP-2, CXP-3, CXP-5, and CXP-6. In CoaXPress mode, SERDES/PCS is configured as specific link rate, based on 8B/10B PCS.

11. Other Design Considerations

11.1. Spread Spectrum Clocking Support

Spread Spectrum Clocking (SSC) is a technique used in electronics design to intentionally modulate the ideal position of the clock edge such that the resulting signal spectrum is spread around the ideal frequency of the clock. In timing circuits, this technique has the advantage of reducing Electromagnetic Interference (EMI) associated with the fundamental frequency of the signal. The amount of EMI a system is allowed to generate is set by various regulatory bodies to ensure systems not interfacing with one another. System spectrum clocking is often used to help meet the regulated EMI requirements. A spread spectrum signal has the disadvantage of having much higher jitter than the un-modulated signal.

Spread spectrum clocking is commonly used for microprocessor clocks, PCI Express reference clocks to reduce EMI. Lattice Nexus 2 SERDES/PCS supports the following optional SSC features defined by PCI Express Base Specification:

- The reference clock can be modulated by +0% to -0.5% from nominal (5000 ppm), referred to as down spreading.
- The modulation rate must be between 30 kHz and 33 kHz.

In PCI Express applications, the Root Complex (RC) is responsible for spreading the reference clock. The Endpoint uses the same clock to pass back the spectrum through the Transmitter. What should be noted is that the reference clock from RC needs to be used in PCI Express Endpoint applications based on Lattice Nexus 2 SERDES/PCS, when the SSC feature has been enabled. Otherwise, reference clock source from local OSC can be used for Lattice Nexus 2 SERDES/PCS. However, using a local OSC as reference clock is not recommended on account of the system compatibility.

The Tx PLL and CDR PLL inside PMA have no capability to generate SSC clock but can be compatible with SSC clock.

Appendix A. 8B/10B Symbol Coding

Table A.1 shows the 8B/10B encoding for data characters. Table A.2 shows the 8B/10B encoding for control characters.

Table A.1. 8B/10B Data Symbol Codes

Data Byte Name	Data Byte Value (Hex)	Data Byte Value (Bin)	Coded Data Value (RD-)	Coded Data Value (RD+)
D0.0	00	000 00000	100111 0100	011000 1011
D1.0	01	000 00001	011101 0100	100010 1011
D2.0	02	000 00010	101101 0100	010010 1011
D3.0	03	000 00011	110001 1011	110001 0100
D4.0	04	000 00100	110101 0100	001010 1011
D5.0	05	000 00101	101001 1011	101001 0100
D6.0	06	000 00110	011001 1011	011001 0100
D7.0	07	000 00111	111000 1011	000111 0100
D8.0	08	000 01000	111001 0100	000110 1011
D9.0	09	000 01001	100101 1011	100101 0100
D10.0	0A	000 01010	010101 1011	010101 0100
D11.0	0B	000 01011	110100 1011	110100 0100
D12.0	0C	000 01100	001101 1011	001101 0100
D13.0	0D	000 01101	101100 1011	101100 0100
D14.0	0E	000 01110	011100 1011	011100 0100
D15.0	0F	000 01111	010111 0100	101000 1011
D16.0	10	000 10000	011011 0100	100100 1011
D17.0	11	000 10001	100011 1011	100011 0100
D18.0	12	000 10010	010011 1011	010011 0100
D19.0	13	000 10011	110010 1011	110010 0100
D20.0	14	000 10100	001011 1011	001011 0100
D21.0	15	000 10101	101010 1011	101010 0100
D22.0	16	000 10110	011010 1011	011010 0100
D23.0	17	000 10111	111010 0100	000101 1011
D24.0	18	000 11000	110011 0100	001100 1011
D25.0	19	000 11001	100110 1011	100110 0100
D26.0	1A	000 11010	010110 1011	010110 0100
D27.0	1B	000 11011	110110 0100	001001 1011
D28.0	1C	000 11100	001110 1011	001110 0100
D29.0	1D	000 11101	101110 0100	010001 1011
D30.0	1E	000 11110	011110 0100	100001 1011
D31.0	1F	000 11111	101011 0100	010100 1011
D0.1	20	001 00000	100111 1001	011000 1001
D1.1	21	001 00001	011101 1001	100010 1001
D2.1	22	001 00010	101101 1001	010010 1001
D3.1	23	001 00011	110001 1001	110001 1001
D4.1	24	001 00100	110101 1001	001010 1001
D5.1	25	001 00101	101001 1001	101001 1001
D6.1	26	001 00110	011001 1001	011001 1001
D7.1	27	001 00111	111000 1001	000111 1001
D8.1	28	001 01000	111001 1001	000110 1001
D9.1	29	001 01001	100101 1001	100101 1001
D10.1	2A	001 01010	010101 1001	010101 1001

Data Byte Name	Data Byte Value (Hex)	Data Byte Value (Bin)	Coded Data Value (RD-)	Coded Data Value (RD+)
D11.1	2B	001 01011	110100 1001	110100 1001
D12.1	2C	001 01100	001101 1001	001101 1001
D13.1	2D	001 01101	101100 1001	101100 1001
D14.1	2E	001 01110	011100 1001	011100 1001
D15.1	2F	001 01111	010111 1001	101000 1001
D16.1	30	001 10000	011011 1001	100100 1001
D17.1	31	001 10001	100011 1001	100011 1001
D18.1	32	001 10010	010011 1001	010011 1001
D19.1	33	001 10011	110010 1001	110010 1001
D20.1	34	001 10100	001011 1001	001011 1001
D21.1	35	001 10101	101010 1001	101010 1001
D22.1	36	001 10110	011010 1001	011010 1001
D23.1	37	001 10111	111010 1001	000101 1001
D24.1	38	001 11000	110011 1001	001100 1001
D25.1	39	001 11001	100110 1001	100110 1001
D26.1	3A	001 11010	010110 1001	010110 1001
D27.1	3B	001 11011	110110 1001	001001 1001
D28.1	3C	001 11100	001110 1001	001110 1001
D29.1	3D	001 11101	101110 1001	010001 1001
D30.1	3E	001 11110	011110 1001	100001 1001
D31.1	3F	001 11111	101011 1001	010100 1001
D0.2	40	010 00000	100111 0101	011000 0101
D1.2	41	010 00001	011101 0101	100010 0101
D2.2	42	010 00010	101101 0101	010010 0101
D3.2	43	010 00011	110001 0101	110001 0101
D4.2	44	010 00100	110101 0101	001010 0101
D5.2	45	010 00101	101001 0101	101001 0101
D6.2	46	010 00110	011001 0101	011001 0101
D7.2	47	010 00111	111000 0101	000111 0101
D8.2	48	010 01000	111001 0101	000110 0101
D9.2	49	010 01001	100101 0101	100101 0101
D10.2	4A	010 01010	010101 0101	010101 0101
D11.2	4B	010 01011	110100 0101	110100 0101
D12.2	4C	010 01100	001101 0101	001101 0101
D13.2	4D	010 01101	101100 0101	101100 0101
D14.2	4E	010 01110	011100 0101	011100 0101
D15.2	4F	010 01111	010111 0101	101000 0101
D16.2	50	010 10000	011011 0101	100100 0101
D17.2	51	010 10001	100011 0101	100011 0101
D18.2	52	010 10010	010011 0101	010011 0101
D19.2	53	010 10011	110010 0101	110010 0101
D20.2	54	010 10100	001011 0101	001011 0101
D21.2	55	010 10101	101010 0101	101010 0101
D22.2	56	010 10110	011010 0101	011010 0101
D23.2	57	010 10111	111010 0101	000101 0101
D24.2	58	010 11000	110011 0101	001100 0101
D25.2	59	010 11001	100110 0101	100110 0101

Data Byte Name	Data Byte Value (Hex)	Data Byte Value (Bin)	Coded Data Value (RD-)	Coded Data Value (RD+)
D26.2	5A	010 11010	010110 0101	010110 0101
D27.2	5B	010 11011	110110 0101	001001 0101
D28.2	5C	010 11100	001110 0101	001110 0101
D29.2	5D	010 11101	101110 0101	010001 0101
D30.2	5E	010 11110	011110 0101	100001 0101
D31.2	5F	010 11111	101011 0101	010100 0101
D0.3	60	011 00000	100111 0011	011000 1100
D1.3	61	011 00001	011101 0011	100010 1100
D2.3	62	011 00010	101101 0011	010010 1100
D3.3	63	011 00011	110001 1100	110001 0011
D4.3	64	011 00100	110101 0011	001010 1100
D5.3	65	011 00101	101001 1100	101001 0011
D6.3	66	011 00110	011001 1100	011001 0011
D7.3	67	011 00111	111000 1100	000111 0011
D8.3	68	011 01000	111001 0011	000110 1100
D9.3	69	011 01001	100101 1100	100101 0011
D10.3	6A	011 01010	010101 1100	010101 0011
D11.3	6B	011 01011	110100 1100	110100 0011
D12.3	6C	011 01100	001101 1100	001101 0011
D13.3	6D	011 01101	101100 1100	101100 0011
D14.3	6E	011 01110	011100 1100	011100 0011
D15.3	6F	011 01111	010111 0011	101000 1100
D16.3	70	011 10000	011011 0011	100100 1100
D17.3	71	011 10001	100011 1100	100011 0011
D18.3	72	011 10010	010011 1100	010011 0011
D19.3	73	011 10011	110010 1100	110010 0011
D20.3	74	011 10100	001011 1100	001011 0011
D21.3	75	011 10101	101010 1100	101010 0011
D22.3	76	011 10110	011010 1100	011010 0011
D23.3	77	011 10111	111010 0011	000101 1100
D24.3	78	011 11000	110011 0011	001100 1100
D25.3	79	011 11001	100110 1100	100110 0011
D26.3	7A	011 11010	010110 1100	010110 0011
D27.3	7B	011 11011	110110 0011	001001 1100
D28.3	7C	011 11100	001110 1100	001110 0011
D29.3	7D	011 11101	101110 0011	010001 1100
D30.3	7E	011 11110	011110 0011	100001 1100
D31.3	7F	011 11111	101011 0011	010100 1100
D0.4	80	100 00000	100111 0010	011000 1101
D1.4	81	100 00001	011101 0010	100010 1101
D2.4	82	100 00010	101101 0010	010010 1101
D3.4	83	100 00011	110001 1101	110001 0010
D4.4	84	100 00100	110101 0010	001010 1101
D5.4	85	100 00101	101001 1101	101001 0010
D6.4	86	100 00110	011001 1101	011001 0010
D7.4	87	100 00111	111000 1101	000111 0010
D8.4	88	100 01000	111001 0010	000110 1101

Data Byte Name	Data Byte Value (Hex)	Data Byte Value (Bin)	Coded Data Value (RD-)	Coded Data Value (RD+)
D9.4	89	100 01001	100101 1101	100101 0010
D10.4	8A	100 01010	010101 1101	010101 0010
D11.4	8B	100 01011	110100 1101	110100 0010
D12.4	8C	100 01100	001101 1101	001101 0010
D13.4	8D	100 01101	101100 1101	101100 0010
D14.4	8E	100 01110	011100 1101	011100 0010
D15.4	8F	100 01111	010111 0010	101000 1101
D16.4	90	100 10000	011011 0010	100100 1101
D17.4	91	100 10001	100011 1101	100011 0010
D18.4	92	100 10010	010011 1101	010011 0010
D19.4	93	100 10011	110010 1101	110010 0010
D20.4	94	100 10100	001011 1101	001011 0010
D21.4	95	100 10101	101010 1101	101010 0010
D22.4	96	100 10110	011010 1101	011010 0010
D23.4	97	100 10111	111010 0010	000101 1101
D24.4	98	100 11000	110011 0010	001100 1101
D25.4	99	100 11001	100110 1101	100110 0010
D26.4	9A	100 11010	010110 1101	010110 0010
D27.4	9B	100 11011	110110 0010	001001 1101
D28.4	9C	100 11100	001110 1101	001110 0010
D29.4	9D	100 11101	101110 0010	010001 1101
D30.4	9E	100 11110	011110 0010	100001 1101
D31.4	9F	100 11111	101011 0010	010100 1101
D0.5	A0	101 00000	100111 1010	011000 1010
D1.5	A1	101 00001	011101 1010	100010 1010
D2.5	A2	101 00010	101101 1010	010010 1010
D3.5	A3	101 00011	110001 1010	110001 1010
D4.5	A4	101 00100	110101 1010	001010 1010
D5.5	A5	101 00101	101001 1010	101001 1010
D6.5	A6	101 00110	011001 1010	011001 1010
D7.5	A7	101 00111	111000 1010	000111 1010
D8.5	A8	101 01000	111001 1010	000110 1010
D9.5	A9	101 01001	100101 1010	100101 1010
D10.5	AA	101 01010	010101 1010	010101 1010
D11.5	AB	101 01011	110100 1010	110100 1010
D12.5	AC	101 01100	001101 1010	001101 1010
D13.5	AD	101 01101	101100 1010	101100 1010
D14.5	AE	101 01110	011100 1010	011100 1010
D15.5	AF	101 01111	010111 1010	101000 1010
D16.5	B0	101 10000	011011 1010	100100 1010
D17.5	B1	101 10001	100011 1010	100011 1010
D18.5	B2	101 10010	010011 1010	010011 1010
D19.5	B3	101 10011	110010 1010	110010 1010
D20.5	B4	101 10100	001011 1010	001011 1010
D21.5	B5	101 10101	101010 1010	101010 1010
D22.5	B6	101 10110	011010 1010	011010 1010
D23.5	B7	101 10111	111010 1010	000101 1010

Data Byte Name	Data Byte Value (Hex)	Data Byte Value (Bin)	Coded Data Value (RD-)	Coded Data Value (RD+)
D24.5	B8	101 11000	110011 1010	001100 1010
D25.5	B9	101 11001	100110 1010	100110 1010
D26.5	BA	101 11010	010110 1010	010110 1010
D27.5	BB	101 11011	110110 1010	001001 1010
D28.5	BC	101 11100	001110 1010	001110 1010
D29.5	BD	101 11101	101110 1010	010001 1010
D30.5	BE	101 11110	011110 1010	100001 1010
D31.5	BF	101 11111	101011 1010	010100 1010
D0.6	C0	110 00000	100111 0110	011000 0110
D1.6	C1	110 00001	011101 0110	100010 0110
D2.6	C2	110 00010	101101 0110	010010 0110
D3.6	C3	110 00011	110001 0110	110001 0110
D4.6	C4	110 00100	110101 0110	001010 0110
D5.6	C5	110 00101	101001 0110	101001 0110
D6.6	C6	110 00110	011001 0110	011001 0110
D7.6	C7	110 00111	111000 0110	000111 0110
D8.6	C8	110 01000	111001 0110	000110 0110
D9.6	C9	110 01001	100101 0110	100101 0110
D10.6	CA	110 01010	010101 0110	010101 0110
D11.6	CB	110 01011	110100 0110	110100 0110
D12.6	CC	110 01100	001101 0110	001101 0110
D13.6	CD	110 01101	101100 0110	101100 0110
D14.6	CE	110 01110	011100 0110	011100 0110
D15.6	CF	110 01111	010111 0110	101000 0110
D16.6	D0	110 10000	011011 0110	100100 0110
D17.6	D1	110 10001	100011 0110	100011 0110
D18.6	D2	110 10010	010011 0110	010011 0110
D19.6	D3	110 10011	110010 0110	110010 0110
D20.6	D4	110 10100	001011 0110	001011 0110
D21.6	D5	110 10101	101010 0110	101010 0110
D22.6	D6	110 10110	011010 0110	011010 0110
D23.6	D7	110 10111	111010 0110	000101 0110
D24.6	D8	110 11000	110011 0110	001100 0110
D25.6	D9	110 11001	100110 0110	100110 0110
D26.6	DA	110 11010	010110 0110	010110 0110
D27.6	DB	110 11011	110110 0110	001001 0110
D28.6	DC	110 11100	001110 0110	001110 0110
D29.6	DD	110 11101	101110 0110	010001 0110
D30.6	DE	110 11110	011110 0110	100001 0110
D31.6	DF	110 11111	101011 0110	010100 0110
D0.7	E0	111 00000	100111 0001	011000 1110
D1.7	E1	111 00001	011101 0001	100010 1110
D2.7	E2	111 00010	101101 0001	010010 1110
D3.7	E3	111 00011	110001 1110	110001 0001
D4.7	E4	111 00100	110101 0001	001010 1110
D5.7	E5	111 00101	101001 1110	101001 0001
D6.7	E6	111 00110	011001 1110	011001 0001

Data Byte Name	Data Byte Value (Hex)	Data Byte Value (Bin)	Coded Data Value (RD-)	Coded Data Value (RD+)
D7.7	E7	111 00111	111000 1110	000111 0001
D8.7	E8	111 01000	111001 0001	000110 1110
D9.7	E9	111 01001	100101 1110	100101 0001
D10.7	EA	111 01010	010101 1110	010101 0001
D11.7	EB	111 01011	110100 1110	110100 1000
D12.7	EC	111 01100	001101 1110	001101 0001
D13.7	ED	111 01101	101100 1110	101100 1000
D14.7	EE	111 01110	011100 1110	011100 1000
D15.7	EF	111 01111	010111 0001	101000 1110
D16.7	F0	111 10000	011011 0001	100100 1110
D17.7	F1	111 10001	100011 0111	100011 0001
D18.7	F2	111 10010	010011 0111	010011 0001
D19.7	F3	111 10011	110010 1110	110010 0001
D20.7	F4	111 10100	001011 0111	001011 0001
D21.7	F5	111 10101	101010 1110	101010 0001
D22.7	F6	111 10110	011010 1110	011010 0001
D23.7	F7	111 10111	111010 0001	000101 1110
D24.7	F8	111 11000	110011 0001	001100 1110
D25.7	F9	111 11001	100110 1110	100110 0001
D26.7	FA	111 11010	010110 1110	010110 0001
D27.7	FB	111 11011	110110 0001	001001 1110
D28.7	FC	111 11100	001110 1110	001110 0001
D29.7	FD	111 11101	101110 0001	010001 1110
D30.7	FE	111 11110	011110 0001	100001 1110
D31.7	FF	111 11111	101011 0001	010100 1110

Table A.2. 8B/10B Control Symbol Codes

Data Byte Name	Data Byte Value (Hex)	Data Byte Value (Bin)	Coded Data Value (RD-)	Coded Data Value (RD+)
K28.0	1C	000 11100	001111 0100	110000 1011
K28.1[1]	3C	001 11100	001111 1001	110000 0110
K28.2	5C	010 11100	001111 0101	110000 1010
K28.3	7C	011 11100	001111 0011	110000 1100
K28.4	9C	100 11100	001111 0010	110000 1101
K28.5[1]	BC	101 11100	001111 1010	110000 0101
K28.6	DC	110 11100	001111 0110	110000 1001
K28.7[1]	FC	111 11100	001111 1000	110000 0111
K23.7	F7	111 10111	111010 1000	000101 0111
K27.7	FB	111 11011	110110 1000	001001 0111
K29.7	FD	111 11101	101110 1000	010001 0111
K30.7	FE	111 11110	011110 1000	100001 0111

Note:

1. Contains a COMMA.

References

- Lattice Nexus 2 Platform – Overview Data Sheet (FPGA-DS-02122)
- Lattice Nexus 2 Platform – Specifications Data Sheet (FPGA-DS-02121)
- Certus-N2 web page

For more information on the Nexus 2-related IP, reference designs, and board documents, refer to the following web pages:

- IP Cores and Reference Designs for Nexus 2 Devices web page
- Kits, Boards, and Demonstrations for Nexus 2 Devices web page

A variety of technical notes for the Lattice Nexus 2 Devices are available.

- Lattice Nexus 2 Embedded Memory User Guide (FPGA-TN-02366)
- Lattice Nexus 2 Multi-Boot User Guide (FPGA-TN-02385)
- Lattice Nexus 2 sysCONFIG User Guide (FPGA-TN-02370)
- Lattice Nexus 2 sysDSP User Guide (FPGA-TN-02362)
- Lattice Nexus 2 sysI/O User Guide (FPGA-TN-02365)
- Sub-LVDS Signaling Using Lattice Devices (FPGA-TN-02028)
- Electrical Recommendations for Lattice SERDES (FPGA-TN-02077)
- High-Speed PCB Design Considerations (FPGA-TN-02178)
- PCIe X4 IP Core User Guide (FPGA-IPUG-02126)

Other references:

- Lattice Radiant Software web page
- Lattice Insights web page for Lattice Semiconductor training courses and learning plans
- Clarity Designer Manual
- ECO Editor Manual
- High Speed SERDES Devices and Applications
- PCI Express Base Specification Rev5.0 v1.0
- PCI Express Technology Comprehensive Guide to Generations 1.x, 2.x, and 3.0

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/Support/AnswerDatabase.

Revision History

Revision 0.80, April 2025

Section	Change Summary
All	Preliminary release.

www.latticesemi.com