JESD204C Intel® FPGA IP User Guide

Updated for Intel® Quartus® Prime Design Suite: **21.3**

IP Version: **1.1.0**

Online Version
Send Feedback

UG-20226

ID: 683108
Version: 2023.02.10

Contents

1. About the JESD204C Intel FPGA IP User Guide

This user guide provides the features, architecture description, steps to instantiate, and guidelines to design the JESD204C Intel® FPGA IP using Intel Stratix® 10 and Intel Agilex™ devices.

Intended Audience

This document is intended for:

- Design architect to make IP selection during system level design planning phase
- Hardware designers when integrating the IP into their system level design
- Validation engineers during system level simulation and hardware validation phase

Related Documents

The following table lists other reference documents which are related to the JESD protocol.

Table 1. Related Documents

Reference	Description
JESD204C Intel Agilex Design Example User Guide	Provides information about how to instantiate JESD204C design examples using Intel Agilex devices.
JESD204C Intel Stratix 10 Design Example User Guide	Provides information about how to instantiate JESD204C design examples using Intel Stratix 10 devices.
JESD204C Intel FPGA IP Release Notes	Lists the changes made for the JESD204C Intel FPGA IP in a particular release.
F-Tile JESD204C Intel FPGA IP User Guide	Provides information about the F-Tile JESD204C Intel FPGA IP.
F-Tile JESD204C Intel FPGA IP Design Example User Guide	Provides information about how to instantiate the F-Tile JESD204C design examples using Intel Agilex F-Tile devices.
F-Tile JESD204C Intel FPGA IP Release Notes	Lists the changes made for the F-Tile JESD204C Intel FPGA IP in a particular release.
Intel Agilex Device Data Sheet	This document describes the electrical characteristics, switching characteristics, configuration specifications, and timing for Intel Agilex devices.
Intel Stratix 10 Device Data Sheet	Provides information about the electrical characteristics, switching characteristics, configuration specifications, and timing for Intel Stratix 10 devices
E-Tile Transceiver PHY User Guide	Provides information about the E-tile Transceiver PHY.

Acronyms and Glossary

Table 2. Acronym List

Acronym	Expansion
LEMC	Local Extended Multiblock Clock
FC	Frame clock rate
ADC	Analog to Digital Converter
DAC	Digital to Analog Converter
DSP	Digital Signal Processor
TX	Transmitter
RX	Receiver
DLL	Data link layer
CSR	Control and status register
CRU	Clock and Reset Unit
ISR	Interrupt Service Routine
FIFO	First-In-First-Out
SERDES	Serializer Deserializer
ECC	Error Correcting Code
FEC	Forward Error Correction
SERR	Single Error Detection (in ECC, correctable)
DERR	Double Error Detection (in ECC, fatal)
PRBS	Pseudorandom binary sequence
MAC	Media Access Controller. MAC includes protocol sublayer, transport layer, and data link layer.
PHY	Physical Layer. PHY typically includes the physical layer, SERDES, drivers, receivers and CDR.
PCS	Physical Coding Sub-layer
PMA	Physical Medium Attachment
RBD	RX Buffer Delay
UI	Unit Interval = duration of serial bit
RBD count	RX Buffer Delay latest lane arrival
RBD offset	RX Buffer Delay release opportunity
SH	Sync header
TL	Transport layer

Table 3. Glossary List

Term	Description
Converter Device	ADC or DAC converter
Logic Device	FPGA or ASIC

continued...

Term	Description
Octet	A group of 8 bits, serving as input to 64/66 encoder and output from the decoder
Nibble	A set of 4 bits which is the base working unit of JESD204C specifications
Block	A 66-bit symbol generated by the 64/66 encoding scheme
Link Clock	The associated parallel data will be 128 bit/132 bit instead of 64 bit/66 bit. Link Clock = Lane Line Rate/132.
Frame	A set of consecutive octets in which the position of each octet can be identified by reference to a frame alignment signal.
Frame Clock	A system clock which runs at the frame's rate, that must be 1x, 2x, or 4x link clock.
Samples per frame clock	Samples per clock, the total samples in frame clock for the converter device.
LEMC	Internal clock used to align the boundary of the extended multiblocks between lanes and into the external references (SYSREF or Subclass 1).
Subclass 0	No support for deterministic latency. Data should be immediately released upon lane to lane deskew on receiver.
Subclass 1	Deterministic latency using SYSREF.
Multipoint Link	Inter-device links with 2 or more converter devices.
64B/66B Encoding	Line code that maps 64-bit data to 66 bits to form a block. The base level data structure is a block that starts with 2-bit sync header.

Table 4. Symbols

Term	Description
L	Number of lanes per converter device
M	Number of converters per device
F	Number of octets per frame on a single lane
S	Number of samples transmitted per single converter per frame cycle
N	Converter resolution
N’	Total number of bits per sample in the user data format
CS	Number of control bits per conversion sample
CF	Number of control words per frame clock period per link
HD	High Density user data format
E	Number of multiblocks in an extended multiblock

2. Overview of the JESD204C Intel FPGA IP

The JESD204C Intel FPGA IP is a high-speed point-to-point serial interface for digital-to-analog (DAC) or analog-to-digital (ADC) converters to transfer data to FPGA devices. This unidirectional serial interface runs at a maximum data rate of 28.9 Gbps. This protocol offers higher bandwidth, low I/O count and supports scalability in both number of lanes and data rates.

The JESD204C Intel FPGA IP addresses multidevice synchronization using Subclass 1 to achieve deterministic latency.

The JESD204C Intel FPGA IP supports TX-only, RX-only, and Duplex (TX and RX) mode. The Intel FPGA IP is a unidirectional protocol where interfacing to ADC utilizes the transceiver RX path and interfacing to DAC utilizes the transceiver TX path.

The Intel FPGA IP incorporates:

- Media access control (MAC)—data link layer (DLL) and transport layer (TL) blocks that control the link states.
- Physical layer (PHY)—physical coding sublayer (PCS) and physical media attachment (PMA) block.

The transport layer (TL) in the MAC controls the assembling and disassembling of the frames.

Figure 1. JESD204C Duplex Functional Block Diagram

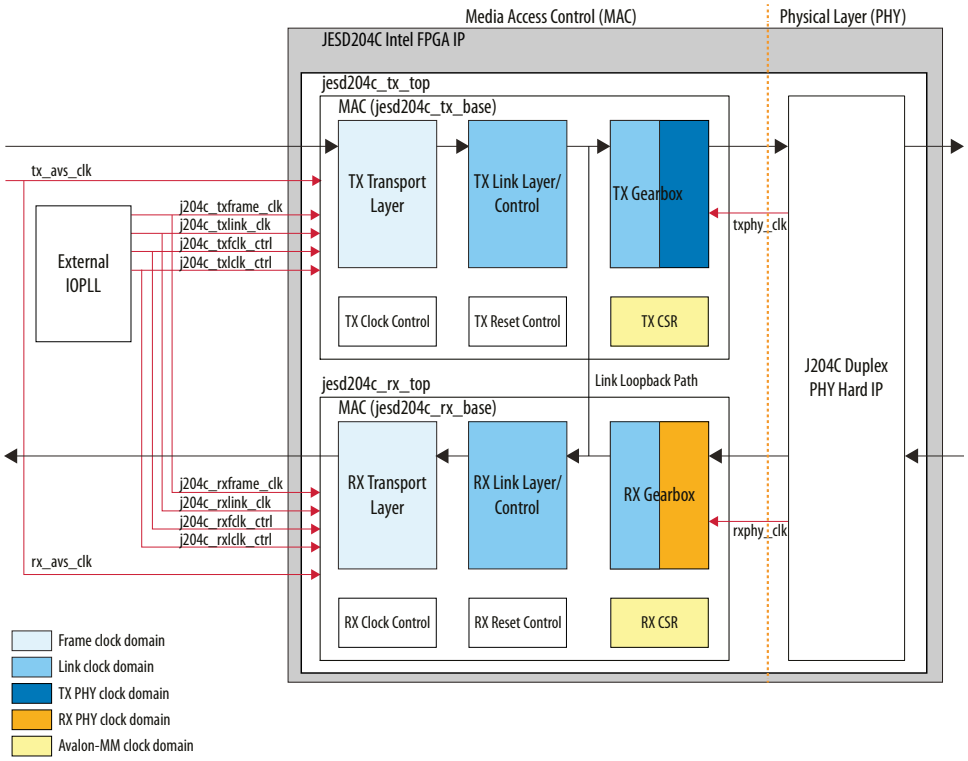


Figure 2. JESD204C TX-only Functional Block Diagram

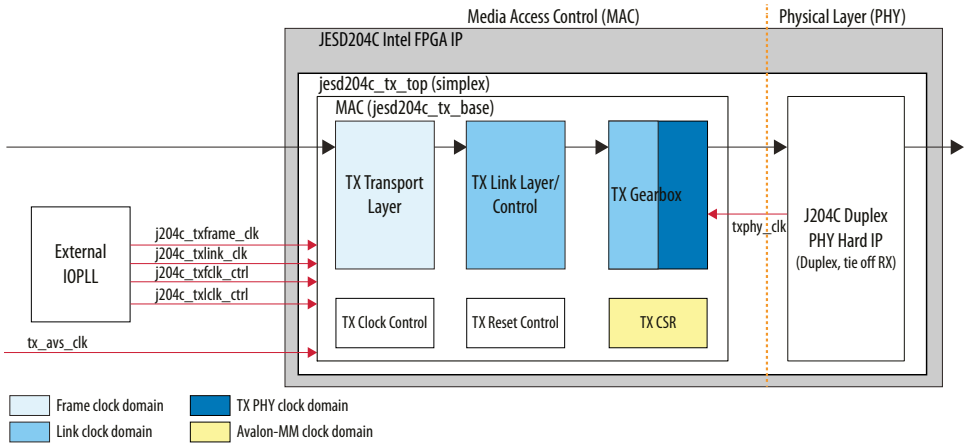
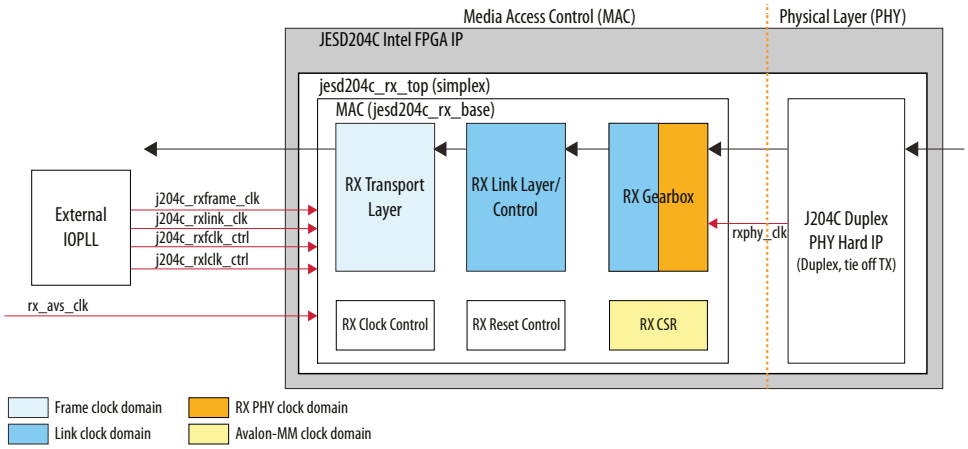


Figure 3. JESD204C RX-only Functional Block Diagram



2.1. Release Information

Intel FPGA IP versions match the Intel Quartus® Prime Design Suite software versions until v19.1. Starting in Intel Quartus Prime Design Suite software version 19.2, Intel FPGA IP has a new versioning scheme.

The Intel FPGA IP version (X.Y.Z) number can change with each Intel Quartus Prime software version. A change in:

- X indicates a major revision of the IP. If you update the Intel Quartus Prime software, you must regenerate the IP.
- Y indicates the IP includes new features. Regenerate your IP to include these new features.
- Z indicates the IP includes minor changes. Regenerate your IP to include these changes.

Table 5. JESD204C Intel FPGA IP Release Information

Item	Description
IP Version	1.1.0
Intel Quartus Prime Pro Edition Version	20.1
Release Date	2020.04.13
Ordering Code	IP-JESD204C

Related Information

JESD204C Intel FPGA IP Release Notes
Provides information about the new features and updates for each IP release.

2.2. Device Family Support

Table 6. Intel Device Family Support

Device Family	Support Level
Intel Agilex (E-tile)	Final
Intel Stratix 10 (E-tile)	Final

The following terms define device support levels for Intel FPGA IP cores:

- Advance support—the IP core is available for simulation and compilation for this device family. Timing models include initial engineering estimates of delays based on early post-layout information. The timing models are subject to change as silicon testing improves the correlation between the actual silicon and the timing models. You can use this IP core for system architecture and resource utilization studies, simulation, pinout, system latency assessments, basic timing assessments (pipeline budgeting), and I/O transfer strategy (data-path width, burst depth, I/O standards tradeoffs).
- Preliminary support—the IP core is verified with preliminary timing models for this device family. The IP core meets all functional requirements, but might still be undergoing timing analysis for the device family. It can be used in production designs with caution.
- Final support—the IP core is verified with final timing models for this device family. The IP core meets all functional and timing requirements for the device family and can be used in production designs.

2.3. JESD204C Intel FPGA IP Features

The JESD204C Intel FPGA IP is a high-speed point-to-point serial interface intellectual property (IP). The JESD204C Intel FPGA IP is the latest IP from Intel that supports the JESD204C protocol. This IP is not backwards compatible and does not support JESD204B protocol. You can use the existing the JESD204B Intel FPGA IP to support JESD204B protocol.

Table 7. Brief Information about the JESD204C Intel FPGA IP

Features	Description
Protocol Features	• Joint Electron Device Engineering Council (JEDEC) JESD204C standard 2017 • Device subclass: — Subclass 0—No deterministic latency. — Subclass 1—Uses SYSREF signal to support deterministic latency
Core Features	• Data rate of up to 28.9 Gbps for Intel Agilex and Intel Stratix 10 (E-tile) devices. • Single or multiple lanes (up to 16 lanes per link) • Local extended multiblock clock (LEMC) counter based on E=1 to 256 • Serial lane alignment and monitoring • Lane synchronization • Modular design that supports multidevice synchronization • MAC and PHY partitioning • Deterministic latency support • 64/66 encoding • Scrambling/descrambling

continued...

Features	Description
	• Avalon® streaming interface for transmit and receive datapaths • Avalon memory-mapped interface for control and status registers (CSR) • Dynamic generation of simulation testbench • Bonded and non-bonded TX PMA mode • Optional support for ECC M20K DCFIFO • Options for sync header configurations — CRC-12 — Standalone command channels
Limitations	No FEC support
Typical Application	• Wireless communication equipment • Broadcast equipment • Military equipment • Medical equipment • Test and measurement equipment
Device Family Support	• Intel Agilex and Intel Stratix 10 (E-tile) FPGA devices
Design Tools	• Platform Designer parameter editor in the Intel Quartus Prime Pro Edition software for design creation and compilation • Timing Analyzer in the Intel Quartus Prime software for timing analysis • ModelSim* - Intel FPGA Starter Edition, QuestaSim* simulator, VCS*/VCS MX, and Xcelium* Parallel simulator software for design simulation or synthesis

2.4. Presets

Intel offers presets to assist you in creating your designs.

Table 8. Available Presets

Presets	Resolution	Lane Rate (Mbps)	L	M	F	S	HD	E	CS	CF	Transceiver Reference Clock (refclk) Frequency (MHz)
Duplex	24	24333.3	2	8	12	1	0	3	0	0	368.681818
	16	16222.2	4	8	4	1	0	4	0	0	245.787878

2.5. Performance and Resource Utilization

Table 9. JESD204C Intel FPGA IP Performance

Device Family	PMA Speed Grade	FPGA Fabric Speed Grade	Enable Soft PCS (Gbps)
Intel Agilex (E-tile)	1	–1	5.0 to 28.9
	2	–2	5.0 to 28.3
		–3	5.0 to 25.6
	3	–2	5.0 to 17.4
		–3	5.0 to 17.4
Intel Stratix 10 (E-tile)	1	–1	5.0 to 28.9
		–2	5.0 to 25.6
	2	–1	5.0 to 28.3

continued...

Device Family	PMA Speed Grade	FPGA Fabric Speed Grade	Enable Soft PCS (Gbps)
		–2	5.0 to 25.6
	3	–1	5.0 to 17.4
		–2	5.0 to 17.4
		–3	5.0 to 17.4

The following table lists the estimated resource utilization data of the JESD204C IP. These results are obtained using the Intel Quartus Prime software targeting the Intel Agilex, AGFB014R24A3E3VR0 device and Intel Stratix 10, 1ST280EY3F55E3VGS1 device.

The variations for resource utilization are configured with the following parameter settings:

Table 10. Parameter Settings to Obtain the Resource Utilization Data

Parameter	Setting
JESD204C Wrapper	Base and PHY
JESD204C Subclass	1
Data Rate	17.4 Gbps
Bonding Mode	Non-bonded
Reference Clock Frequency	263.636363 MHz
Enable Scrambler (SCR)	On
Enable Error Code Correction (ECC_EN)	Off

Table 11. JESD204C IP Resource Utilization for Intel Agilex Devices

Variants	L	M	F	FCLK_MULP	WIDTH_MULP	ALM	ALUT	Logic Register	M20K
TX	4	8	6	1	8	6885	6488	9113	2
	4	8	6	2	4	7012	7080	9780	2
	4	8	4	1	4	5382	5810	7509	2
	4	8	4	2	2	6311	6876	9458	2
	2	8	6	1	8	3901	3883	5306	2
	2	8	6	2	4	4066	4247	5916	2
	8	8	3	1	16	12492	11374	15988	2
	8	8	3	2	8	12983	12577	18074	2
	3	8	4	1	2	4410	4801	6290	2
	3	8	4	2	1	5030	5617	7604	2
RX	2	8	12	1	4	5236	5780	6434	10
	2	8	12	2	2	4755	5347	5846	10
	1	2	8	1	4	2650	3189	3428	4
	1	2	8	2	2	2637	3224	3436	4

continued...

Variants	L	M	F	FCLK_MULP	WIDTH_MULP	ALM	ALUT	Logic Register	M20K
	1	4	24	1	4	3281	3881	4311	6
	1	4	24	2	2	2963	3567	3919	6
	8	1	1	1	16	13582	15237	14634	34
	8	1	1	2	8	13743	16028	15894	34
	3	2	4	1	4	5560	6244	6209	9
	3	2	4	2	2	5717	6658	6741	12

Table 12. JESD204C IP Resource Utilization for Intel Stratix 10 Devices

Variants	L	M	F	FCLK_MULP	WIDTH_MULP	ALM	ALUT	Logic Register	M20K
TX	4	8	6	1	8	6865	6474	9253	2
	4	8	6	2	4	7002	7084	10092	2
	4	8	4	1	4	5398	5829	7708	2
	4	8	4	2	2	6712	7445	10171	2
	2	8	6	1	8	3944	3881	5369	2
	2	8	6	2	4	4190	4310	6015	2
	8	8	3	1	16	12601	11494	16877	2
	8	8	3	2	8	13157	12746	18645	2
	3	8	4	1	2	4405	4827	6344	2
	3	8	4	2	1	5052	5638	7678	2
RX	2	8	12	1	4	5266	5827	6349	10
	2	8	12	2	2	4819	5420	5944	10
	1	2	8	1	4	2642	3204	3385	4
	1	2	8	2	2	2672	3214	3484	4
	1	4	24	1	4	3243	3837	4166	6
	1	4	24	2	2	3027	3585	3914	6
	8	1	1	1	16	13511	15225	14903	34
	8	1	1	2	8	13344	15812	14891	34
	3	2	4	1	4	5524	6228	6332	9
	3	2	4	2	2	5751	6673	6966	12

3. Functional Description

The JESD204C IP consists of scrambler and descrambler, transport layer, data link layer and physical layer.

The transport layer maps and packetizes the data samples into JESD204C frame data format. The transport layer operates in the parameters of M, N, S, CS and CF and maps into the parameters of F octets and L lanes. The transport layer is part of the JESD204C IP.

This IP supports line rate up to 28.9 Gbps per lane, and uses device clock which in turns generates the desired internal clocks for the transceivers and core logic. The frame clock does not need to be a physical input to the FPGA based on the *JESD204C Specification*.

To support multidevice synchronization, JESD204C IP uses Local Extended Multiblock Clock (LEMC) as a common timing reference. The IP generates the LEMC counter and uses *SYSREF* to align and reset the LEMC counter.

The IP supports Subclass 0 and Subclass 1. With Subclass 1, the IP can use the SYSREF signal and Device clock routed to achieve deterministic latency between the logic and converter devices.

3.1. Clocks

The JESD204C IP runs on link clock (link layer) and frame clock (transport layer). The transceiver runs in the link clock domain and the serial clock domain.

Table 13. JESD204C IP Clocks

Clock Signal	Formula	Description
TX/RX device clock `j204c_pll_refclk`	PLL selection	The device clock is the PLL reference clock to the transceiver PLL.

TX/RX link clock
`j204c_txlink_clk`
`j204c_rxlink_clk`
	Line rate/132	The timing reference for the JESD204C IP. The link clock is line rate divided by 132 because the link clock operates in a 132-bit data bus domain architecture after 64B/66B encoding.

TX/RX frame clock
`j204c_txframe_clk`
`j204c_rxframe_clk`
	(Link clock frequency*FCLK_MULP) MHz	The frame clock as per the JESD204C specification. The frame clock is always 1x, 2x, or 4x of the link clock.
		.

TX/RX Avalon-MM (AVMM) clock `j204c_tx_avs_clk`	—	The configuration clock for the JESD204C IP control and status registers through the Avalon-MM interface. This clock is asynchronous to all the other clocks. The frequency range of this clock is 75–125 MHz.

continued...

Clock Signal	Formula	Description
`j204c_rx_avs_clk`		

TX/RX PHY clock
`j204c_txphy_clk`
`j204c_rxphy_clk`
	Line rate/64	The PHY clock internally generated from the transceiver parallel clock for the TX path or the recovered clock generated from the CDR for the RX path.

Transceiver reconfig clock `j204c_reconfig_clk`	—	The transceiver reconfiguration clock. The frequency range of this clock is 100–162 MHz.

3.1.1. Device Clock

In a converter device, the sampling clock is typically the device clock. The JESD204C IP uses the device clock to generate the desired internal clocks for the transceivers and core logic.

For the JESD204C IP link in an FPGA logic device, you can select one of the options provided in the **PLL/CDR reference clock frequency** parameter in the JESD204C IP parameter editor.

In the single reference clock design, both sets of pins are driven by the same clock source. The device clock is used as the transceiver PLL reference clock and also the core PLL reference clock. In the dual reference clock design, each set of pins are driven by a different clock source. The device clock is used as the core PLL reference clock and the other reference clock (phase-locked to device clock) is used as the transceiver PLL reference clock. If you want to use the same reference clock for the transceiver and core PLLs, you must use two separate input pins for these PLL reference clocks in your design. Use a common clock source on the board to generate two separate clocks of the same frequency to drive the inputs.

The device clock frequency depends on the data rate and total number of lanes. When you generate the IP, the Intel Quartus Prime Pro Edition software provides the available reference frequency for the transceiver PLL and core PLL based on your selection.

For Subclass 1 application, ensure that the routing of the SYSREF signal and the device clock to the FPGA has matching trace lengths.

3.1.2. Frame Clock and Link Clock

The frame clock frequency always equals the link clock frequency times the frame clock frequency multiplier (FCLK_MULP):

Frame clock frequency = FCLK_MULP x Link clock frequency

You can set the frame clock frequency multiplier through the JESD204C IP parameter editor. The valid values for the multiplier are limited to 1, 2, and 4. Because of the fixed relationship between the link clock and the frame clock, the Avalon-ST data will not always be streaming.

To provide consistency across the design regardless of frame clock and sampling clock, the link clock is used as a timing reference.

The IOPLL core should provide both the frame clock and link clock from the same PLL as these two clocks are treated as synchronous in the design.

For JESD204 TX and RX IPs, `j204c_txlclk_ctrl` or `j204c_rxlclk_ctrl` provides the phase information of a link clock rising edge that aligned to a frame clock rising edge.

Similarly, `j204c_txfclk_ctrl` or `j204c_rxfclk_ctrl` provides the phase information of a frame clock rising edge that aligned to a link clock rising edge. This additional clock phase information handles the transfer between frame clock and link clock in a synchronous manner.

3.2. Local Extended Multiblock Clock

The JESD204C IP uses the Extended Multiblock Clock (LEMC) as a common timing reference to support multidevice configuration.

LEMC is an internal clock that aligns the boundaries of the extended multiblocks between lanes. In deterministic latency devices, LEMC aligns the boundaries to an external reference, for example, SYSREF. The use of LEMC is mandatory in Subclass 1 modes but optional in Subclass 0 modes.

The JESD204C IP implements LEMC as a counter that increments in link clock counts, and depends on the **Multiblocks in an extended multiblock (E)** parameter. The extended multiblock is a container of a number of multiblocks.

The E parameter depends on these two factors:

- The parameter must allow an integer of F within an extended multiblock. For example, if F=3, 32 multiblocks contain 256 octets (32x64/8). 256 octets is not divisible by F=3. So, for F=3, the minimum E is 3.
- E must be larger than the maximum possible delay variation across any two lanes of a link.

In Subclass 1 deterministic latency system, SYSREF is distributed to the devices to be aligned in the system. The SYSREF signal resets the internal LEMC clock edge when the sampled SYSREF rising edge transitions from 0 to 1.

The JESD204C IP does not use the device clock directly to sample SYSREF because of the source synchronous signaling of SYSREF with respect to the device clock sampling from the clock chip. The IP uses the link clock to sample SYSREF. The PLL that provides the link clock or frame clock must be in normal mode to phase-compensate the link clock to the device clock.

You can program a single or multiple sampling of SYSREF through the JESD204C control and status registers.

- A single sampling SYSREF does not detect SYSREF period errors.
- A continuous sampling mode detects SYSREF period errors.

In most converter device systems, disable SYSREF sampling if there are no errors, and begin link operation with a link reinitialization request.

3.2.1. LEMC Counter

JESD204C IP maintains an LEMC counter that counts from 0 to (E*32)–1 and wraps around again.

In Subclass 0 system, the LEMC counter starts at the deassertion of the link reset signal, without waiting for SYSREF detection.

In Subclass 1 deterministic latency system, all transmitters and receivers receive a common SYSREF, and the LEMC counter resets within two link clock cycles. SYSREF must be the same for the converter devices, which are grouped and required to be synchronized together.

Maximum SYSREF frequency = data rate/(66x32xE).

Table 14. Example of *SYSREF* Frequency Calculation

In this example, you can choose to perform one of the following options:

- Provide two SYSREF and a device clock; in which the ADC groups share the device clock and the two SYSREF clock (1.42 MHz and 2.84 MHz).
- Provide 1 SYSREF running at 1.4 MHz and a device clock for the two ADC groups and one DAC group because the SYSREF period in the DAC is in the multiplication of *n* integer.

Group	Configuration	*SYSREF* Frequency
ADC Group 1 (2 ADCs)	• LMF = 222 • E = 2 • Data rate = 6,000 Mbps	(6,000 MHz/(66x32x2) = 1.42 MHz
ADC Group 2 (2 ADCs)	• LMF = 811 • E = 1 • Data rate = 6,000 Mbps	(6,000 MHz/(66x32x1) = 2.84 MHz
DAC Group 3 (2 DACs)	• LMF = 222 • E = 1 • Data rate = 3,000 Mbps	(3,000 MHz/(66x32x1) = 1.42 MHz

Note: 1.42 MHz is the common maximum SYSREF frequency. You can lower the frequency to 0.71 MHz and the design still works.

3.3. CRC Encoding/Decoding

The JESD204C IP supports only CRC-12 encoding/decoding.

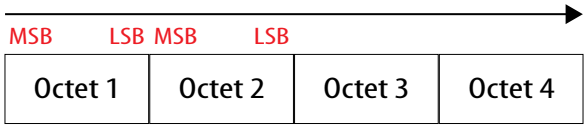
The CRC-12 encoder computes 12 parity bits using this polynomial:

$0 \times 987 = x^{12} + x^9 + x^8 + x^3 + x^2 + x^1 + 1$

3.4. Scrambler/Descrambler

Both the scrambler and descrambler are designed in a 128-bit parallel implementation and the scrambling/descrambling order starts from the first octet with MSB first.

Figure 4. Scrambling/Descrambling Order



The JESD204C TX and RX IP core support scrambling by implementing a 128-bit parallel scrambler in each lane. The scrambler and descrambler are located in the JESD204C IP MAC interfacing to the Avalon streaming interface. You can enable or disable scrambling through CSR configuration for all lanes. Mixed mode operation, where scrambling is enabled for some lanes, is not permitted.

The scrambling polynomial is:

$x^{58} + x^{39} + 1$

The descrambler can self-synchronize in 58 bits. In a typical application where the reset value of the scrambler seed is different from the converter device to FPGA logic device, the correct user data is recovered in the receiver in 1 link clock (due to the 128-bit architecture). The PRBS pattern checker on the transport layer should always disable checking of the first eight octets from the JESD204C RX IP.

4. Getting Started

4.1. Installing and Licensing Intel FPGA IP Cores

The Intel Quartus Prime software installation includes the Intel FPGA IP library. This library provides many useful IP cores for your production use without the need for an additional license. Some Intel FPGA IP cores require purchase of a separate license for production use. The Intel FPGA IP Evaluation Mode allows you to evaluate these licensed Intel FPGA IP cores in simulation and hardware, before deciding to purchase a full production IP core license. You only need to purchase a full production license for licensed Intel IP cores after you complete hardware testing and are ready to use the IP in production.

The Intel Quartus Prime software installs IP cores in the following locations by default:

Figure 5. IP Core Installation Path

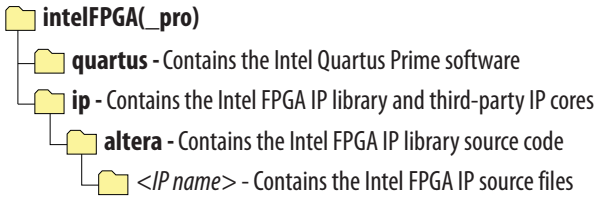


Table 15. IP Core Installation Locations

Location	Software	Platform
`<drive>:\intelFPGA_pro\ip\altera`	Intel Quartus Prime Pro Edition	Windows*
`<drive>:\intelFPGA\ip\altera`	Intel Quartus Prime Standard Edition	Windows
`<home directory>:/intelFPGA_pro/ip/altera`	Intel Quartus Prime Pro Edition	Linux*
`<home directory>:/intelFPGA/ip/altera`	Intel Quartus Prime Standard Edition	Linux

Note: The Intel Quartus Prime software does not support spaces in the installation path.

4.2. Intel FPGA IP Evaluation Mode

The free Intel FPGA IP Evaluation Mode allows you to evaluate licensed Intel FPGA IP cores in simulation and hardware before purchase. Intel FPGA IP Evaluation Mode supports the following evaluations without additional license:

- Simulate the behavior of a licensed Intel FPGA IP core in your system.
- Verify the functionality, size, and speed of the IP core quickly and easily.
- Generate time-limited device programming files for designs that include IP cores.
- Program a device with your IP core and verify your design in hardware.

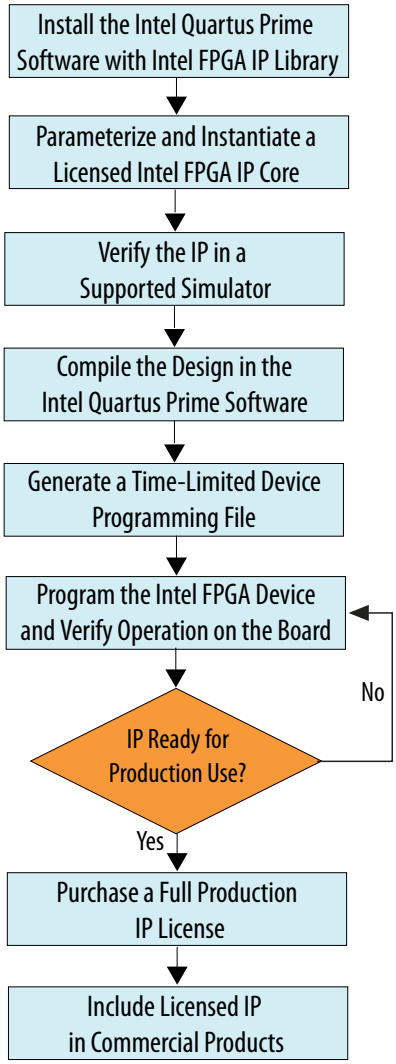
Intel FPGA IP Evaluation Mode supports the following operation modes:

- **Tethered**—Allows running the design containing the licensed Intel FPGA IP indefinitely with a connection between your board and the host computer. Tethered mode requires a serial joint test action group (JTAG) cable connected between the JTAG port on your board and the host computer, which is running the Intel Quartus Prime Programmer for the duration of the hardware evaluation period. The Programmer only requires a minimum installation of the Intel Quartus Prime software, and requires no Intel Quartus Prime license. The host computer controls the evaluation time by sending a periodic signal to the device via the JTAG port. If all licensed IP cores in the design support tethered mode, the evaluation time runs until any IP core evaluation expires. If all of the IP cores support unlimited evaluation time, the device does not time-out.
- **Untethered**—Allows running the design containing the licensed IP for a limited time. The IP core reverts to untethered mode if the device disconnects from the host computer running the Intel Quartus Prime software. The IP core also reverts to untethered mode if any other licensed IP core in the design does not support tethered mode.

When the evaluation time expires for any licensed Intel FPGA IP in the design, the design stops functioning. All IP cores that use the Intel FPGA IP Evaluation Mode time out simultaneously when any IP core in the design times out. When the evaluation time expires, you must reprogram the FPGA device before continuing hardware verification. To extend use of the IP core for production, purchase a full production license for the IP core.

You must purchase the license and generate a full production license key before you can generate an unrestricted device programming file. During Intel FPGA IP Evaluation Mode, the Compiler only generates a time-limited device programming file (<*project name*>`_time_limited.sof`) that expires at the time limit.

Figure 6. Intel FPGA IP Evaluation Mode Flow



Note: Refer to each IP core's user guide for parameterization steps and implementation details.

Intel licenses IP cores on a per-seat, perpetual basis. The license fee includes first-year maintenance and support. You must renew the maintenance contract to receive updates, bug fixes, and technical support beyond the first year. You must purchase a full production license for Intel FPGA IP cores that require a production license, before generating programming files that you may use for an unlimited time. During Intel FPGA IP Evaluation Mode, the Compiler only generates a time-limited device programming file (`<project name>_time_limited.sof`) that expires at the time limit. To obtain your production license keys, visit the Intel FPGA Self-Service Licensing Center.

The Intel FPGA Software License Agreements govern the installation and use of licensed IP cores, the Intel Quartus Prime design software, and all unlicensed IP cores.

Related Information

- Intel FPGA Licensing Support Center
- Introduction to Intel FPGA Software Installation and Licensing

4.3. IP Catalog and Parameter Editor

The IP Catalog displays the IP cores available for your project, including Intel FPGA IP and other IP that you add to the IP Catalog search path.. Use the following features of the IP Catalog to locate and customize an IP core:

- Filter IP Catalog to **Show IP for active device family** or **Show IP for all device families**. If you have no project open, select the **Device Family** in IP Catalog.
- Type in the Search field to locate any full or partial IP core name in IP Catalog.
- Right-click an IP core name in IP Catalog to display details about supported devices, to open the IP core's installation folder, and for links to IP documentation.
- Click **Search for Partner IP** to access partner IP information on the web.

The parameter editor prompts you to specify an IP variation name, optional ports, and output file generation options. The parameter editor generates a top-level Intel Quartus Prime IP file (`.ip`) for an IP variation in Intel Quartus Prime Pro Edition projects or Quartus IP file (`.qip`) for an IP variation in Intel Quartus Prime Standard Edition projects.

4.4. JESD204C IP Component Files

The following table describes the generated files and other files that may be in your project directory. The names and types of generated files specified may vary depending on whether you create your design with VHDL or Verilog HDL.

Table 16. Generated Files

Extension	Description
<*variation name*>.**v** or **.vhd**	IP core variation file, which defines a VHDL or Verilog HDL description of the custom IP. Instantiate the entity defined by this file inside of your design. Include this file when compiling your design in the Intel Quartus Prime software.
<*variation name*>**.cmp**	A VHDL component declaration file for the IP variation. Add the contents of this file to any VHDL architecture that instantiates the IP.
<*variation name*>**.sdc**	Contains timing constraints for your IP variation.
<*variation name*>**.qip** or **.ip**	Contains Intel Quartus Prime project information for your IP variation.
<*variation name*>**.tcl**	Tcl script file to run in Intel Quartus Prime software.
<*variation name*>**.sip**	Contains IP library mapping information required by the Intel Quartus Prime software. The Intel Quartus Prime software generates a **. sip** file during generation of some Intel FPGA IP cores. You must add any generated **.sip** file to your project for use by NativeLink simulation and the Intel Quartus Prime Archiver.
<*variation name*>**.spd**	Contains a list of required simulation files for your IP.

4.5. Creating a New Intel Quartus Prime Project

You can create a new Intel Quartus Prime project with the **New Project Wizard**. Creating a new project allows you to do the following:

- Specify the working directory for the project.
- Assign the project name.
- Designate the name of the top-level design entity.

1. Launch the Intel Quartus Prime software.
2. On the **File** menu, click **New Project Wizard**.
3. In the **New Project Wizard: Directory, Name, Top-Level Entity** page, specify the working directory, project name, and top-level design entity name. Click **Next**.
4. In the **New Project Wizard: Add Files** page, select the existing design files (if any) you want to include in the project. Click **Next**.
5. In the **New Project Wizard: Family & Device Settings** page, select the device family and specific device you want to target for compilation. Click **Next**.
6. In the **EDA Tool Settings** page, select the EDA tools you want to use with the Intel Quartus Prime software to develop your project.
7. Review the summary of your chosen settings in the **New Project Wizard** window, then click **Finish** to complete the Intel Quartus Prime project creation.

4.6. Parameterizing and Generating the IP

Refer to JESD204C Intel FPGA IP Parameters on page 35 for the IP parameter values and description.

1. In the IP Catalog (**Tools ➤ IP Catalog**), locate and double-click the JESD204C Intel FPGA IP.
2. Specify a top-level name for your custom IP variation. This name identifies the IP variation files in your project. If prompted, also specify the target Intel FPGA device family and output file HDL preference. Click **OK**.
3. After parameterizing the core, go to the Example Design tab and click **Generate Example Design** to create the simulation testbench. Skip to 5 if you do not want to generate the design example.
4. Set a name for your <*example_design_directory*> and click **OK** to generate supporting files and scripts.
   The testbench and scripts are located in the <*example_design_directory*>/`simulation` folder.

   The **Generate Example Design** option generates supporting files for the following entities:

   - IP core design example for simulation—refer to *Generating and Simulating the Design Example* section in the respective design example user guides.
   - IP core design example for synthesis—refer to Compiling the *JESD204C Design Example* section in the respective design example user guides.
5. Click **Finish** or **Generate HDL** to generate synthesis and other optional files matching your IP variation specifications. The parameter editor generates the top-level `.ip`, `.qip` or `.qsys` IP variation file and HDL files for synthesis and simulation.

The top-level IP variation is added to the current Intel Quartus Prime project. Click **Project ➤ Add/Remove Files in Project** to manually add a `.qip` or `.qsys` file to a project. Make appropriate pin assignments to connect ports.

Note: Some parameter options are grayed out if they are not supported in a selected configuration or it is a derived parameter.

4.7. Compiling the JESD204C IP Design

Refer to the Designing with the JESD204C Intel FPGA IP on page 25 before compiling the JESD204C IP core design.

To compile your design, click **Start Compilation** on the Processing menu in the Intel Quartus Prime software. You can use the generated `.ip` or `.qip` file to include relevant files into your project.

Related Information

Intel Quartus Prime Help
More information about compilation in Intel Quartus Prime software.

4.8. Programming an FPGA Device

After successfully compiling your design, program the targeted Intel device with the Intel Quartus Prime Programmer and verify the design in hardware. For instructions on programming the FPGA device, refer to the *Intel Quartus Prime Pro Edition User Guide: Programmer*.

5. Designing with the JESD204C Intel FPGA IP

When designing with the JESD204C Intel FPGA IP, you need to take into account certain considerations to ensure a fully-functioning design. Follow the design guidelines provided.

5.1. JESD204C TX and RX Reset Sequence

The JESD204C base core and transport layer require various resets for the IP and transceiver. All the resets in the core assert asynchronously and deassert synchronously.

Table 17. JESD204C IP Resets

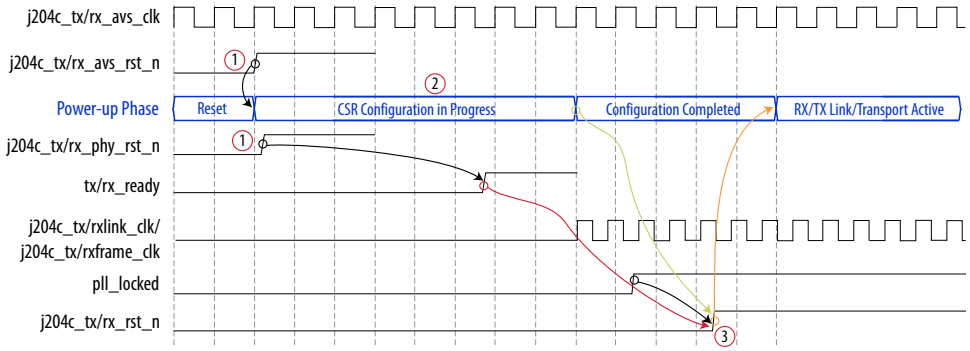
Reset Signal	Clock Domain	Description

TX/RX Link and Frame Reset
`j204c_tx_rst_n`
`j204c_rx_rst_n`
	TX/RX link clock
	TX/RX frame clock
		• You can deassert the link and frame reset after the configuration phase completes.
		• After this reset deasserts, the JESD204C IP is in operation mode.

TX/RX PHY Reset
`j204c_tx_phy_rst_n`
`j204c_rx_phy_rst_n`
	Transceiver Native PHY clock	• The transceiver requires this reset to reset the PMA and PCS blocks.
		• Intel recommends that you assert the link and frame reset when this reset asserts.

TX/RX AVS Reset
`j204c_tx_avs_rst_n`
`j204c_rx_avs_rst_n`
	TX/RX Avalon memory-mapped reset for CSR (`j204c_tx_avs_clk`/ `j204c_rx_avs_clk`)	• This reset is for the Avalon memory-mapped slave interface, which consists of the Configuration and Status Register (CSR) blocks.
		• This reset must deassert first before the JESD204C IP link reset and frame reset deassert.
		• After this reset deasserts, configuration phase starts. You can program the CSR register values if a non-default value is required.
		• Intel recommends that you assert the link and frame reset when this reset asserts.

Figure 7. JESD204C TX/RX Reset Sequence



The descriptions below correspond to the Figure 7 on page 25:

1. The sequence begins when the TX/RX AVS reset and TX/RX PHY reset deassert.
2. During the configuration phase, you can configure the run-time CSR parameters. The number of clock cycles does not matter provided that `j204c_tx_rst_n`/`j204c_rx_rst_n` remains asserted.
3. `j204c_tx_rst_n`/`j204c_rx_rst_n` only deasserts after configuration phase, and when the PLL is locked and the transceiver is ready. Internally, in the JESD204C IP, `j204c_tx_rst_n`/`j204c_rx_rst_n` synchronizes to the respective clock domains. You can assert `j204c_tx_rst_n`/`j204c_rx_rst_n` at any time after its initial deassertion, but when you deassert, make sure the configuration phase is over, the PLL is locked, and the transceiver is ready.

5.2. Configuration Phase

Before the hardware reset deasserts, if you want to make any changes to your JESD204C IP configuration, you have to make the changes during the configuration phase.

The configuration phase is the only right phase to change the configuration because all configuration registers are quasi-static in nature and stable before the IP comes out of reset. The known exception to this rule is the SYSREF control registers.

If you want to make a change in the link configuration, such as disable interrupts, during mid-operation, you must always do a link re-initialization.

5.3. Link Reinitialization

The JESD204C IP implements a simple synchronous clear to all data and control logics during link reinitialization.

Link reinitialization occurs in two ways:

- You manually trigger link reinitialization by setting the `link_reinit` bit. The hardware clears the `link_reinit` and `reinit_in_prog` bits when link reinitialization completes.
- The hardware automatically triggers link reinitialization because of errors. You have full control, through the `tx_err` and `rx_err` registers, to set the specific type of errors to trigger link reinitialization automatically. The hardware clears the `reinit_in_prog` bit when link reinitialization completes.

Note: Link reinitialization does not initiate SYSREF re-detection. Use the `sysref_singledet` bit to re-detect SYSREF edge. Link reinitialization affects only the transport layer and link layer; the CSR, transceiver, and the PHY-related logics are not affected.

5.4. SYSREF Sampling

You can choose to enable continuous SYSREF sampling or a single SYSREF detection.

The software logic programs the clock cleaner to the SPI to enable SYSREF generation. To resample SYSREF , the software logic writes to the SYSREF_CTRL registers to enable either a continuous sampling or a single detection. If both bits are enabled, continuous sampling takes precedence.

You may want to disable SYSREF sampling after some time. Disabling SYSREF sampling also disables the continuous sampling mode, and subsequently programs the clock cleaner to disable SYSREF output.

Consider one of the following recommendations to configure the SYSREF resampling flow:

1. Set the `sysref_singledet`, `sysref_alwayson`, and `sysref_lemc_err_en_reinit` register bits to 1.
   - If a new SYSREF edge is detected, the JESD204C IP clears the `sysref_singledet` bit and automatically starts a link reinitialization.
   - All the Avalon streaming interface signals return to default state.
   - The LEMC block resets to reflect the newly detected SYSREF edge.
   - All the Avalon streaming interface signals are active again based on a new LEMC data.
   - If LEMC error interrupt is enabled, the JESD204C needs to service and clear the error.
2. Set the `sysref_singledet` bit to 1 and `sysref_alwayson` bit to 0.
   - If a new SYSREF edge is detected, the JESD204C IP clears the `sysref_singledet` bit, but no automatic start of link reinitialization.
   - All the Avalon streaming interface signals remain active.
   - The LEMC block resets to reflect the newly detected SYSREF edge.
     - In the TX IP, the egress sync header (SH) adjusts as LEMC undergoes realignment to the new SYSREF edge. The corresponding RX (that receives the adjusted SH) may subject to SH-related errors.
     - In the RX IP, the ingress Avalon streaming data does not get affected because the change of LEMC does not impact the already streaming data. However, the link loses its deterministic latency characteristic. To restore the deterministic latency behavior, a link reinitialization is required.
   - You must set the `link_reinit` bit to 1 after `sysref_singledet` clears to start a link reinitialization.
   - All the Avalon streaming interface signals return to default state, and get reactivated based on a new LEMC data.

To better handle potential race conditions between the assertion of the `link_reinit` bit (through user-specified or auto reinitialization) and the assertion of the `sysref_singledet` bit, the IP imposes the following behavior:

- For TX core: If the the `link_reinit` bit asserts), the Avalon streaming interface deactivates. After the link reinitialization is complete, the Avalon streaming interface gets activated only when the `sysref_singledet` deasserts.
- For RX core: If the the `link_reinit` bit asserts, the Avalon streaming interface deactivates. After the link reinitialization is complete, the Avalon streaming interface gets activated only when the `sysref_singledet` deasserts, and SH (`j204c_rx_sh_lock`) and EMB (`j204c_rx_emb_lock`) have achieved lock.

5.5. Interrupt and Error Handling

The JESD204C IP interrupts the processor when there are errors or reinitialization requests in the design. The interrupts are peripheral and level sensitive.

The IP holds a level-sensitive interrupt signal asserted until the peripheral deasserts the interrupt signal. When the level-sensitive interrupt is high, the state of the interrupt in the Interrupt Controller is pending or active pending. If the peripheral deasserts the interrupt signal for any reason, the Interrupt Controller removes the pending state from the interrupt.

Every error condition in the JESD204C IP latches on the error status and keeps the interrupt signal asserted until the error is serviced and the ISR writes a 1 to clear the error status.

When interrupt is asserted and fulfills the Interrupt Controller configuration (for example, priority, interrupt IDs), the processor jumps to the Interrupt Service Routine (ISR) to execute the routine.

The ISR must service the requirements of the JESD204C IP by reading the error status and then clearing the interrupt, so that the JESD204C IP could deassert the interrupt. This is particularly important for level-sensitive interrupts, where ISR must ensure that the interrupt is deasserted at the Interrupt Controller input before proceeding to the next step. Typically, this is called the top half ISR handler.

The bottom half ISR handler may require a chain of events.

5.5.1. Interrupt Configuration for TX and RX

When configuring the control and status registers, you must determine which error types to promote as interrupts and which ones to disable.

To determine the error types as interrupts or otherwise, configure the JESD204C TX `Error Enable` and `RX Error Enable` registers at offset 0x64.

By default, the IP promotes all errors as interrupt enable except the ECC correctable error for the RX. The following examples depict errors that you can exclude as interrupts:

- The JESD204C TX core detects data bubble in the Avalon streaming interface. If your system design has no data bubble, and there is a continuous data stream from the upstream device, you can disable interrupt for this error type.
- If your system design do not keep track of correctable error (CE) occurrences, you may disable the intruder.

  Note: You may want to enable CE interrupt for high-end server systems, to keep track of the CE events as a predictor for future events. Predicting future events enables the IP to execute some form of preventive maintenance or part placement to prevent the likelihood of dreaded uncorrectable errors and system panics.

5.5.2. Interrupt Top Half ISR Handler

The top half ISR handler reads the error status (JESD204C TX and RX error status at offset 0x60) and stores the error bits meant for bottom half handling.

The ISR writes a 1 to the corresponding error bits to clear the status. The JESD204C IP deasserts the interrupt. Then, the ISR checks the pending interrupt to ensure that the IP deasserts the interrupt.

Note: The ISR should not write all ones to the register to clear because this may clear incoming errors for different error types.

If the interrupt is not cleared, then the ISR checks the status, and stores the new error types, and OR it with the previous error status. Then, once again the ISR repeats the clearing operation and checks for pending interrupts.

Note: The error types are not grouped as correctable errors, uncorrectable errors (non fatal), and uncorrectable errors (fatal). Intel recommends that you (system designer) identify the error types and bucket them for software error handling routines.

5.5.3. Interrupt Bottom Half ISR Handler

The bottom half ISR handler mainly controls a chain of events to the SPI controller device or clock chips, and the JESD204C IP.

The following tables describe the ISR handler recommendations for different TX and RX error types.

Table 18. TX Interrupt Handler Recommendations

Error Type	ISR Handler Operation
`tx_sysref_lemc_err`	• This error occurs when the SYSREF signal continuously resets the LEMC counter at an unexpected time, which violates the LEMC period. • The processor needs to check the setting for all LEMC counters configured on the JESD204C IP converter device and clock chip.
`tx_dll_data_invalid_err`	• This error occurs when data bubble appears on the Avalon streaming interface in the JESD204C TX base core. • The processor needs to probe the upstream device the find out why it has stalled and take further action. *Note:* By design, you should calculate the data throughput to ensure that there are no data bubbles in the design. This additional protection is to minimize errors in the system.
`tx_frame_data_invalid_err`	• This error occurs when data bubble appears on the Avalon streaming interface in the JESD204C TX transport layer. • The processor needs to probe the upstream device the find out why it has stalled and take further action. *Note:* By design, you should calculate the data throughput to ensure that there are no data bubbles in the design. This additional protection is to minimize errors in the system.
`cmd_invalid_err`	• This error occurs when data bubble appears on the Avalon streaming command interface in the JESD204C TX link layer. • The processor needs to probe the upstream device the find out why it has stalled and take further action. *Note:* By design, you should calculate the data throughput to ensure that there are no data bubbles in the design. This additional protection is to minimize errors in the system.
`tx_ready_err`	• This error is flagged when the transceiver indicates that it is down. • The processor needs to probe the transceiver controls.
`tx_pcfifo_full_err`	• This error is flagged when at least 1 lane of the transceiver phase compensation FIFO indicates full. • The processor needs to probe the correct clocks and data-width settings.
`tx_gb_underflow_err`	• This error is flagged when at least 1 instance of the TX gearbox FIFO is having underflow. • The processor ensures that the system design provides the correct `j204c_txlink_clk` frequency of (data rate)/132 to the IP.
`tx_gb_overflow_err`	• This error is flagged when at least 1 instance of the TX gearbox FIFO is having overflow. • The processor ensures that the system design provides the correct `j204c_txlink_clk` frequency of (data rate)/132 to the IP.

Table 19. RX Interrupt Handler Recommendations

Error Type	ISR Handler Operation
`rx_sysref_lemc_err`	• This error occurs when the SYSREF signal continuously resets the LEMC counter at an unexpected time, which violates the LEMC period. • The processor needs to check the setting for all LEMC counters configured on the JESD204C IP converter device and clock chip. • By default, the link automatically reinitializes when this signal is asserted.
`rx_dll_data_ready_err`	• This error occurs when the Avalon streaming interface in the JESD204C RX base core does not have backpressure and yet, the upstream device indicates that it is unable to take in data. • The processor needs to probe the upstream device to find out why it has stalled and take further action. *Note:* By design, you should calculate the data throughput to ensure that there are no data bubbles in the design. This additional protection is to minimize errors in the system.
`rx_frame_data_ready_err`	• This error occurs when the Avalon streaming interface in the RX transport layer does not have backpressure and yet, the upstream device indicates that it is unable to take in data. • The processor needs to probe the upstream device to find out why it has stalled and take further action. *Note:* By design, you should calculate the data throughput to ensure that there are no data bubbles in the design. This additional protection is to minimize errors in the system.
`rx_cmd_ready_err`	• This error occurs when the Avalon streaming command interface in the RX link layer does not have backpressure and yet, the upstream device indicates that it is unable to take in data. • The processor needs to probe the upstream device to find out why it has stalled and take further action. *Note:* By design, you should calculate the data throughput to ensure that there are no data bubbles in the design. This additional protection is to minimize errors in the system.
`rx_cdr_locked_err`	• This error occurs because of incoming data stream that causes RX PMA to unlock its CDR. • The processor may not be able to recover from such error. PMA debug is required.
`rx_pcfifo_empty_err`	• This error is flagged when at least 1 lane of the transceiver phase compensation FIFO indicates empty. • The processor may not be able to recover from such error. Transceiver debug is required.
`rx_pcfifo_full_err`	• This error is flagged when at least 1 lane of the transceiver phase compensation FIFO indicates full. • The processor may not be able to recover from such error. Transceiver debug is required.
`rx_lane_deskew_err`	• This is a system error. • You need to investigate total skew and the E parameter settings.
`rx_invalid_sync_header`	• This error occurs when the IP receives “00” or “11” in the expected SH location. • Refer to the *JESD204C Specifications* for action required for hardware.
`rx_invalid_eomb`	• This errors occurs when “00001” sequence in the pilot signal is not received at an expected location in the sync word. • Refer to the *JESD204C Specifications* for action required for hardware.

continued...

Error Type	ISR Handler Operation
rx_invalid_eoemb	• This error occurs when the EoEMB identifier in the pilot signal has an unexpected value. • Refer to the *JESD204C Specifications* for action required for hardware.
rx_cmd_par_err	• The error is flagged when the final parity bit in the command channel data for a given Sync Word does not match the calculated parity for the received command channel bits. • Refer to the *JESD204C Specifications* for action required for hardware.
rx_crc_err	• This error occurs when the receive CRC generator has calculated a parity which does not match the parity received in the Sync Word. • The hardware tries to reinitialize but if the error persists, the processor is required to find out the cause of CRC error.
rx_gb_underflow_err	• This error is flagged when at least 1 instance of the RX gearbox FIFO is having underflow. • This underflow error occurs when the frequency ratio between j204c_rxlink_clk and j204_rxphy_clk is larger than 16:33. In this scenario, the j204c_rxlink_clk clock is unexpectedly faster than the j204c_rxphy_clk clock. • When this error occurs, the system needs to reset. • To prevent this error from occurring, the frequency ratio of j204c_rxlink_clk and j204_rxphy_clk must be 16:33.
rx_gb_overflow_err	• This error is flagged when at least 1 instance of the RX gearbox is having overflow. • This overflow error occurs when the frequency ratio between j204c_rxlink_clk and j204_rxphy_clk is smaller than 16:33. In this scenario, the j204c_rxlink_clk clock is unexpectedly slower than the j204c_rxphy_clk clock. • When this error occurs, the system needs to reset. • To prevent this error from occurring, the frequency ratio of j204c_rxlink_clk and j204_rxphy_clk must be 16:33.
rx_sh_unlock_err	• This error indicates that SH alignment is lost. • The hardware always tries to reinitialize for this error.
rx_emb_unlock_err	• This error indicates that EMB alignment is lost. • The hardware always tries to reinitialize for this error.
rx_eb_full_err	• This error could be caused by the same reason that causes deskew error. • You need to investigate total skew and the E parameter settings.
rx_ecc_corrected_err	• This error keeps track of the EC detected. Intel recommends you to enable this error so that the processor can keep track of the EC detected. • The processor can enter error detection and correction routine.
rx_ecc_fatal_err	• ECC fatal error that indicates bad data has been sent to the upstream device. The severity of this error depends on the system. • For example, ultrasound applications may still cope with uncorrectable ECC errors. However, systems where data packets should never be lost, the system may reset the core.

5.6. Deterministic Latency

To achieve optimal performance for deterministic latency, Intel recommends that you follow the guidelines provided.

The *JESD204C Specifications* states the following requirements to achieve optimal performance for deterministic latency:

- Length of extended multiblock size must be larger than the maximum possible delay variation across any link.
- Value of RX buffer delay (RBD) in terms of link cycles must be larger than possible delay across any link.

These two requirements ensure that the RBD is large enough to guarantee the TX data reaches the RX buffers before the RX elastic buffer is released. The IP releases the RX elastic buffer at the assertion of the SYSREF signal. You could set the IP to release the RX elastic buffer earlier to reduce latency.

The JESD204C TX and RX cores run on a link clock with 128-bit data width, and support a configurable E parameter. These settings enable the IP to tune the RBD release in the link clock domain instead of the frame clock domain. The effective frame clock period changes depending on the F parameter.

For the JESD204C IP, Subclass 1 modes support deterministic latency. Use the following guidelines for Subclass 1 deterministic latency tuning.

- The *JESD204C Specifications* only describes the tuning of the RBD release on the left side of the *SYSREF*. The JESD204C RX core allows the tuning for RBD release on both left and right sides of the *SYSREF* tick, as long the tuning does not violate the multiframe buffer.
- Different ADC/DAC vendors have different variations. The SYSREF offset depends on how precise the system is set up and minimized.
- In a multipoint link, all IP cores within that multipoint link must use the same SYSREF signal to ensure that the LEMC counters in each core are aligned.
- The converter device and the FPGA devices must always sample the SYSREF signal before deterministic latency can be achieved. If there is race condition, do a link reinitialization so that all transactions are based on the LEMC counters sampled with SYSREF instead of the free-running LEMC counters in both the devices.
- Upon the detection of the SYSREF edge, the JESD204C TX core transmits the SH data when the next LEMC counter is 0. Subsequently, the core indicates the end of an extended multiblock (EoEMB) after E number of block has been sent.
- The JESD204C RX core implements the RX elastic buffer (per lane) that is large enough to store all multiblocks. The RX elastic buffer is 512 deep with 2 multiblocks. This buffer size allows the tolerance of lane skew between the earliest possible data arrival to the latest lane arrival to the release opportunity. Release opportunity should never be set before the earliest arrival data.
- The release opportunity in JESD204C specification indicates the range that covers the full size of RX elastic buffer or at least one LEMC cycle, which ever is smaller. For JESD204C RX core, the release opportunity is either LEMC or RBD offset (whichever is earlier). Due to the limit of elastic buffer size, JESD204C RX core does not tolerate the condition where ((E*16) – `rbd_count_early`) and `rbd_offset` has the delta of >512.

- Latency incurred by TX and RX should be repeatable. Upon resets, there is also latency variation in the RX SERDES contributed from the phase compensation FIFOs, gearbox, and word aligner.
- By choosing a RBD release which can tolerate the cumulative latency variation from the transmit path to the receiver path, there will always be a fixed number of latency from transmit to release path. This creates the deterministic latency.
- RBD count reflects on which LEMC count the latest arrival lane is. RBD offset is a user-defined value to indicate on which LEMC count the RBD will be released. All lanes will be aligned when RBD is released.
- RBD count may vary slightly upon multiple resets. The worst possible value is 2 link clock counts in a single direction. RBD count reflects on which LEMC count the latest lane arrived, thus it will always be any legal value from 0 to E – 1.
- RBD offset is a user-defined register. Legal value for specific point of tuning has to be 0 to (E*16) – 1. However, if you set any value larger than (E*16) – 1, this will be interpreted as immediate RBD release which is equivalent to the RBD count on the latest lane arrival. Alternatively, the control and status registers provide an additional bit which can be set to indicate RBD immediate release.
- If there are multiple lanes, setting RBD offset as RBD count minus 1 is illegal and causes LEMC align error. This setting violates the internal buffer.
- For example, in a system where E=1; legal value of RBD count will be from 0 to 15. During the first reset, if the RBD count reported is 8, do not set RBD offset 2 link clock counts before and after RBD count. This is because in multiple reset scenario, you do not know if the RBD count will vary forwards or backwards. For the actual count, you need to wrap around the count. For the actual point of release:
  - ((E*16) – RBD offset) value which is larger than RBD count means RBD release on the right side of the LEMC tick.
  - ((E*16) – RBD offset) value which is smaller than RBD count means that RBD release on the left side of the LEMC tick.
- After identifying the correct RBD offset value to set, set this value to all the links in the multipoint link.

6. JESD204C Intel FPGA IP Parameters

Table 20. JESD204C Intel FPGA IP Parameters

Parameter	Value	Description
Main Tab		
Device family	Intel Agilex Intel Stratix 10 (E-tile)	Supports Intel Agilex and Intel Stratix 10 E-tile devices.
JESD204C wrapper	• Base Only • PHY Only • Both Base and PHY	Select the JESD204C wrapper. • Base Only—generates the data link layer and transport layer. • PHY Only—generates the transceiver PHY layer only (hard PCS). • Both Base and PHY—generates both the base (data link layer and transport layer) and the transceiver PHY layer (hard PCS).
Data path	• Simplex Receiver • Simplex Transmitter • Duplex	Select the operation modes. This selection enables or disables the receiver and transmitter supporting logic. • Simplex Receiver—instantiates the receiver to interface to the ADC. • Simplex Transmitter—instantiates the transmitter to interface to the DAC. • Duplex—instantiates the receiver and transmitter to interface to both the ADC and DAC.
JESD204C Subclass	• 0 • 1	Select the JESD204C subclass modes. • 0—Set subclass 0 • 1—Set subclass 1
Data rate	5.0–28.9 Gbps	Set the lane rate for each lane. The maximum rate is 28.9 Gbps. Refer to Performance and Resource Utilization on page 11 for more information.
Transceiver type	E-tile	Default option is E-tile.
Bonding mode	• Bonded • Non-bonded	Set the bonding modes. • Bonded—Select this option to minimize inter-lanes skew for the transmitter datapath. • Non-bonded—Select this option to disable inter-lanes skew control for the transmitter datapath.
PLL/CDR reference clock frequency	*Variable*	Set the transceiver reference clock frequency for PLL or CDR. The frequency range available for you to choose depends on the data rate.
Enable dynamic reconfiguration	On	This option enables dynamic data rate change.
Enable Native PHY Debug Master Endpoint (NPDME)	On, Off	This option enables the Transceiver Native PHY IP core to include an embedded Native PHY debug master endpoint. This endpoint connects internally to the Avalon-MM slave interface of the Transceiver Native PHY and can access the reconfiguration space of the transceiver. It can perform certain test and debug functions through JTAG using System Console.

continued...

Parameter	Value	Description
Enable capability registers	On, Off	This option enables capability registers, which provides high level information about the transceiver channel's configuration.
Set user-defined IP identifier	0–255	Set a user-defined numeric identifier that can be read from the identifier offset when the capability registers are enabled.
Enable control and status registers	On, Off	This option enables soft registers for reading status signals and writing control signals on the PHY interface through the embedded debug.

JESD204C Configurations Tab		
Lanes per converter device (L)	1–16	Set the number of lanes per converter device.
Converters per device (M)	1–32	Set the number of converters per converter device.
Octets per frame (F)	1–256	The number of octets per frame is derived from F= M*N'*S/(8*L).
Converter resolution (N)	1–32	Set the number of conversion bits per converter.
Transmitted bits per sample (N')	4–32	Set the number of transmitted bits per sample (JESD204 word size, which is in nibble group). *Note:* If parameter CF equals to 0 (no control word), parameter N' must be larger than or equal to sum of parameter N and parameter CS (N' ≥ N + CS). Otherwise, parameter N' must be larger than or equal to parameter N (N'≥N).
Samples per converter per frame (S)	1–32	Set the number of transmitted samples per converter per frame.
Multiblocks in an extended multiblock (E)	1–32	Set the number of multiblock within an extended multiblock.
Control bits (CS)	0–3	Set the number of control bits per conversion sample.
Control words (CF)	0–31	Set the number of control words per frame clock period per link.
High-density user data format (HD)	0–1	Turn on this option to set the data format. This parameter controls whether a sample may be divided over more lanes. • On: High Density format • Off: Data should not cross the lane boundary
Sync header configuration (SH_CONFIG)	• CRC-12 • Standalone command channels	Sets the SH encoding configuration. • CRC-12: For the sync word to contain error detection information. • Standalone command channel: For the sync word to contain transmit commands and status information.
Frame clock frequency multiplier (FCLK_MULP)	1, 2, 4	Select the frame clock frequency multiplier. • 1: Frame clock and link clock are at the same frequency. • 2: Frame clock frequency is two times the link clock frequency. • 4: Frame clock frequency is four times the link clock frequency. *Note:* When the frame clock frequency multiplier is 2, Intel recommends that you use the following data rates with the stipulated FPGA fabric speed grades if you encounter timing closure difficulties for Intel Stratix 10 devices. • –1 = up to 21.5 Gbps • –2 = up to 19.5 Gbps • –3 = up to 17.4 Gbps
Frame data width multiplier (WIDTH_MULP)	1, 2, 4, 8, 16	Select the data width multiplier between the application layer and transport layer.

continued...

		Note: The multiplier value is auto-calculated based on the M, N, S, and F configurations. Select the smallest data width multiplier value on the list. Other data width multiplier values are not allowed.
Enable TX data pipestage (Transmitter)	0, 1, 2	Select the number of pipeline stages in TX datapath for timing improvement. Setting values of 1 or 2 usually requires additional resources. • 0: Do not insert any pipeline stage. Minimum latency. • 1: Insert 1 pipeline stage. • 2: Insert 2 pipeline stages. *Note:* For high data rates, Intel recommends that you insert 2 pipeline stages for better timing.
Use MLAB DCFIFO in TX Gearbox (Transmitter)	On, Off	Select the type of FIFO used in the TX gearbox. By default, the gearbox uses M20K FIFO. Enable this parameter to use MLAB FIFO.
TX LEMC offset (Transmitter)	0–255	TX LEMC offset from *SYSREF*. Default is 0.
EMB error threshold (Receiver)	1–8	EMB error threshold to unlock EMB FSM back to initialization state. Default is 8.
SH error threshold (Receiver)	1–16	Sync header error threshold to unlock SH FSM back to initialization state. Default is 16.
RX LEMC offset (Receiver)	0–255	RX LEMC offset from *SYSREF*. Default is 0.
RBD offset (Receiver)	0–511	Elastic buffer released point (reference to LEMC) for Subclass 1 usage. Default is 0. One full LEMC, N number means (LEMC – N) cycles to release data in elastic buffer when deskew alignment is achieved.
Enable RX data pipestage (Receiver)	On, Off	Turn on to add pipeline stage in RX datapath for timing improvement. Enabling this option usually requires additional resources.
Use MLAB DCFIFO in RX gearbox (Receiver)	On, Off	Select the type of FIFO used in the RX gearbox. By default, gearbox uses M20K FIFO. Enable this parameter to use MLAB FIFO to achieve better timing and performance.
Enable ECC in M20K DCFIFO (Receiver)	On, Off	Turn on to enable ECC feature if M20K is used as FIFO.
Lane polarity attribute	• Optimize away • Writable	Select whether you want the lane polarity attribute to be read-only (RO) or read and write (RW). • Optimize away: Select this option to enable the attribute to be read-only. • Writable: Select this option to enable the attribute to be read and write. Applies only for RX.
Enable lane polarity detection (Receiver)	16'h0–16'hFFFF	Specify the bit representing the polarity enable status of each lane. For example, LSB represents lane 0, LSB+1 represents lane 1, MSB represents lane 15, and so on. This value depends on the number of lanes you specify.
Polarity inversion (Receiver)	16'h0–16'hFFFF	Specify the bit representing the polarity inversion status of each lane. For example, LSB represents lane 0, LSB+1 represents lane 1, MSB represents lane 15, and so on. This value depends on the number of lanes you specify.
Single lane mode (Receiver)	On, Off	Turn on only when you set the **Sync header configuration** parameter to **Standalone command channel**.
Multilink mode (Receiver)	On, Off	Turn on this parameter when you want to implement synchronization between multiple JESD204C RX IP instances. When you turn on this parameter, the `j204c_rx_dev_emblock_align` and `j204c_rx_alldev_emblock_align` signals are present.

		The IP uses the `j204c_rx_dev_emblock_align` and `j204c_rx_alldev_emblock_align` signals together with the `j204c_rx_dev_lane_align` and `j204c_rx_alldev_lane_align` signals to achieve multidevice synchronization. Refer to Receiver Signals on page 47 for more information about these signals.

Configurations and Status Registers Tab		
CSR optimization	On, Off	Turn on to optimize the usage of the registers, including the Avalon-MM interfaces.

Note: For more information about the **PMA Adaptation** parameters, refer to the *PMA Adaptation* section in the *E-tile Transceiver PHY User Guide*.

Related Information

PMA Adaptation chapter in the E-tile Transceiver PHY User Guide
Provides more information about the PMA Adaption parameters.

7. Interface Signals

The JESD204C Intel FPGA IP uses the signals from the following interfaces.

Table 21. JESD204C Intel FPGA IP Interfaces

Interface	Description
JESD204C MAC to and from the PHY interface	• The IP allows you to generate PHY only, MAC only, or MAC and PHY configurations. • The PHY only generation is to provide a clean interface between the MAC and the PHY, where these signals are useful for debugging link and PHY issues. • The PHY mode has less number of PLLs in the transceiver, because the transmit channels will be bonded when generated together thus requiring less PLLs. Channel bonding also reduces the lane to lane skew on the transmit path.
Avalon memory-mapped interface	• The IP uses the Avalon memory-mapped interface for reading and writing on the JESD204C IP slave component in a memory-mapped system. • The Avalon memory-mapped slave interface allows upstream devices to access internal control and status registers. • The Avalon memory-mapped slave is referred to as Management interface. • The Avalon memory-mapped slave interface is designed as an asynchronous domain to the JESD204C Link clock and Frame clock domains. • If you want to keep the Avalon memory-mapped slave interface as a synchronous domain to the JESD204C Link clock domain, you may do so provided that the domain is within the minimum and maximum frequency specified for `j204_tx_avs_clk` or `j204_rx_avs_clk`.
Avalon streaming interface	• The IP uses the following types of Avalon streaming interface signals: — Avalon streaming data interface, which operates in `txframe_clk` and `rxframe_clk` domains. — Avalon streaming control sample interface, which operates in `txframe_clk` and `rxframe_clk` domains. — Avalon streaming command interface, which operates in `txlink_clk` and `rxlink_clk` domains.

Note: You should terminate any unused signals.

7.1. Transmitter Signals

Table 22. Top-level Transmitter IP Core Signals

Signal	Width	Direction	Description
JESD204C TX MAC Clocks and Resets			
`j204c_pll_refclk`	1	Input	TX PLL reference clock for the transceiver.
`j204c_txlink_clk`	1	Input	This clock is equal to the TX data rate divided by 132. Generated from the same PLL as `txframe_clk`.

continued...

Signal	Width	Direction	Description
j204c_txlclk_ctrl	1	Input	Generated from the same PLL as txlink_clk and txframe_clk. This clock acts as a phase information for txlink_clk to handle CDC between txlink_clk and txframe_clk.
j204c_txframe_clk	1	Input	Synchronous with txlink_clk. Frequency is equal, 2x, or 4x txlink_clk, based on the selected option for the frame clock frequency multiplier parameter. Generated from the same PLL as txlink_clk. .
j204c_txfclk_ctrl	1	Input	Generated from the same PLL as txlink_clk and txframe_clk. This clock acts as a phase information for txframe_clk to handle CDC between txlink_clk and txframe_clk.
j204c_tx_avs_clk	1	Input	Avalon memory-mapped interface clock.
j204c_reconfig_clk	1	Input	Transceiver reconfiguration clock. In duplex mode, both TX and RX share the same reconfig pins.
j204c_tx_rst_n	1	Input	Active-low asynchronous reset signal for MAC LL and TL.
j204c_tx_phy_rst_n	1	Input	Active-low asynchronous reset signal for PHY.
j204c_tx_avs_rst_n	1	Input	Active-low asynchronous reset signal for TX Avalon memory-mapped interface.
j204c_reconfig_reset	1	Input	Active-high reset signal for transceiver reconfiguration. In duplex mode, both TX and RX share the same reconfig pins.

Signal	Width	Direction	Description
Transceiver Interface			
tx_serial_data	L	Output	Differential high speed serial output data. The clock is embedded in the serial data stream.
tx_serial_data_n	L	Output	Differential high speed serial output data. The clock is embedded in the serial data stream.
tx_ready	L	Output	Indicates that the transceiver TX (per lane) is ready.
tx_pma_ready	L	Output	Indicates that the transceiver TX PMA (per lane) is ready.
j204c_reconfig_read	1	Input	During duplex mode, both TX and RX share the same reconfig pins.
j204c_reconfig_write	1	Input	During duplex mode, both TX and RX share the same reconfig pins.
j204c_reconfig_address	ceil (log2(L)) +19	Input	During duplex mode, both TX and RX share the same reconfig pins. The lower 19 bits specify the address, the upper bits (log2(L)) specify the channel. If L=1, total address bit is always 19 bits.
j204c_reconfig_readdata	8	Output	During duplex mode, both TX and RX share the same reconfig pins.
j204c_reconfig_writedata	8	Output	During duplex mode, both TX and RX share the same reconfig pins.
j204c_reconfig_waitrequest	1	Output	Wait request signal. During duplex mode, both TX and RX share the same reconfig pins.

Signal	Width	Direction	Description
JESD204C TX MAC Avalon Memory-Mapped Interface			
j204c_tx_avs_chipselect	1	Input	When this signal is present, the slave port ignores all Avalon memory-mapped signals unless this signal is asserted. This signal must be used in combination with read or write. If the Avalon memory-mapped bus does not support chip select, you are recommended to tie this port to 1.
j204c_tx_avs_address	10	Input	For Avalon memory-mapped slave, each slave access is based on byte-based offset. For example, address = 0 selects the first four bytes of the slave register and the address = 4 selects the next four bytes of the slave register space.
j204c_tx_avs_writedata	32	Input	32-bit data for write transfers. The width of this signal and the j204c_tx_avs_readdata[31:0] signal must be the same if both signals are present.
j204c_tx_avs_read	1	Input	This signal is asserted to indicate a read transfer. This is an active high signal and requires the j204c_tx_avs_readdata[31:0] signal to be in use.
j204c_tx_avs_write	1	Input	This signal is asserted to indicate a write transfer. This is an active high signal and requires the j204c_tx_avs_writedata[31:0] signal to be in use.
j204c_tx_avs_readdata	32	Output	32-bit data driven from the Avalon memory-mapped slave to master in response to a read transfer.
j204c_tx_avs_waitrequest	1	Output	This signal is asserted by the Avalon memory-mapped slave to indicate that it is unable to respond to a read or write request. The JESD204C Intel FPGA IP ties this signal to 0 to return the data in the access cycle.

Signal	Width	Direction	Description
JESD204C TX MAC Avalon Streaming Interface (Data Channel)			
j204c_tx_avst_data	M*S*WIDTH_MULP*N	Input	The minimum data width = M*S*N. Indicates the converter samples that will be processed by TL. The data format is big endian. If L=1 and M*S*WIDTH_MULP*N=128, the first octet is located at bit[127:120], second octet at bit[119:112], and the last octet at bit[7:0]. If more than one lane is instantiated, lane 0 data is always located in the upper M*S*WIDTH_MULP*N bit data lane, followed by the next lane, with the first octet position for lane 0 is at MSB.
j204c_tx_avst_control	M*S*WIDTH_MULP*CS	Input	Control bits to be inserted as part of CS parameter.
j204c_tx_avst_valid	1	Input	Indicates whether the data from the application layer is valid or invalid. The Avalon streaming sink interface in the TX core cannot be backpressured and assumes that the data is always valid on every cycle when the j204c_tx_avst_ready signal is asserted.

continued...

Signal	Width	Direction	Description
			• 0—data is invalid • 1—data is valid
j204c_tx_avst_ready	1	Output	Indicates that the Avalon streaming sink interface in the TX core is ready to accept data. The Avalon streaming sink interface asserts this signal on the JESD204C transport state of USER_DATA phase. The ready latency is 0.
j204c_tx_frame_ready	1	Output	Indicates that the link layer is ready to accept data. The link layer asserts this signal on a predetermined time before the assertion of the j204c_tx_avst_ready signal.

Signal	Width	Direction	Description
JESD204C TX MAC Command (Command Channel)			
j204c_tx_cmd_data	L*n	Input	Indicates a 6/18-bit user command (per lane) at txlink_clk clock rate. The data format is big endian. If more than one lane is instantiated, lane 0 data is always located at the upper 18 bits or 6 bits of data. Lane L is located at bit[17:0] or bit[5:0], with the first command bit position for lane L at bit[17] or bit[5]. *Note:* n=6 for CRC-12 operation and n =18 for standalone command channel
j204c_tx_cmd_valid	1	Input	Indicates whether the command from the application layer is valid or invalid. The Avalon streaming sink interface in the TX core cannot be backpressured and assumes that data is always valid on every cycle when the j204c_tx_cmd_ready signal is asserted. • 0—data is invalid • 1—data is valid
j204c_tx_cmd_ready	1	Output	Indicates that the Avalon streaming sink interface in the TX core is ready to accept command. The Avalon streaming sink interface asserts this signal on the JESD204C link/transport state of USER_DATA phase. The ready latency is 0.

Signal	Width	Direction	Description
JESD204C Interface			
j204c_tx_sysref	1	Input	SYSREF signal for JESD204C Subclass 1 implementation. For Subclass 0 mode, tie-off this signal to 0.
j204c_tx_somb	1	Output	Start of multiblock.
j204c_tx_soemb	1	Output	Start of extended multiblock.

Signal	Width	Direction	Description
JESD204C TX MAC CSR			
j204c_tx_csr_l	4	Output	Indicates the number of active lanes for the link. The transport layer can use this signal as a compile-time parameter.

continued...

Signal	Width	Direction	Description
j204c_tx_csr_f	8	Output	Indicates the number of octets per frame. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_m	8	Output	Indicates the number of converters for the link. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_cs	2	Output	Indicates the number of control bits per sample. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_n	5	Output	Indicates the converter resolution. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_np	5	Output	Indicates the total number of bits per sample. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_s	5	Output	Indicates the number of samples per converter per frame cycle. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_hd	1	Output	Indicates the high density data format. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_cf	5	Output	Indicates the number of control words per frame clock period per link. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_e	8	Output	LEMC period
j204c_tx_csr_testmode	4	Output	0000: No test mode 0001: Scrambler disabled Other values are reserved.

Signal	Width	Direction	Description
JESD204C TX MAC Out-of-band (OOB)			
j204c_tx_int	1	Output	Interrupt pin for the JESD204C Intel FPGA IP. Interrupt is asserted when any error or synchronization request is detected. Configure the tx_err_enable register to set the type of error that can trigger an interrupt.
j204c_tx2rx_lbdata	L*132	Output	Output as 132-bit width data before the TX gearbox to connect to the RX core (same signal name) for 2-block loopback function. If L>0, LSB of this bus is mapped to lane 0. MSB is mapped to lane L–1.

Table 23. Top-level Transmitter Base Core Signals

Signal	Width	Direction	Description
JESD204C TX MAC Clocks and Resets			
j204c_txlink_clk	1	Input	This clock is equal to the TX data rate divided by 132. Generated from the same PLL as txframe_clk.
j204c_txlclk_ctrl	1	Input	Generated from the same PLL as txlink_clk and txframe_clk. This clock acts as a phase information for txlink_clk to handle CDC between txlink_clk and txframe_clk.

continued...

Signal	Width	Direction	Description
j204c_txframe_clk	1	Input	Synchronous with txlink_clk. Frequency is equal, 2x, or 4x txlink_clk, based on the selected option for the frame clock frequency multiplier parameter. Generated from the same PLL as txlink_clk. .
j204c_txfclk_ctrl	1	Input	Generated from the same PLL as txlink_clk and txframe_clk. This clock acts as a phase information for txframe_clk to handle CDC between txlink_clk and txframe_clk.
j204c_tx_avs_clk	1	Input	Avalon memory-mapped interface clock.
j204c_txphy_clk	1	Input	This clock is equal to the TX data rate divided by 64. Asynchronous with frame or link clock.
j204c_tx_rst_n	1	Input	Active-low asynchronous reset signal for MAC LL and TL.
j204c_tx_phy_rst_n	1	Input	Active-low asynchronous reset signal for PHY.
j204c_tx_avs_rst_n	1	Input	Active-low asynchronous reset signal for TX Avalon memory-mapped interface.

Signal	Width	Direction	Description
JESD204C TX MAC Avalon Memory-Mapped Interface			
j204c_tx_avs_chipselect	1	Input	When this signal is present, the slave port ignores all Avalon memory-mapped signals unless this signal is asserted. This signal must be used in combination with read or write. If the Avalon memory-mapped bus does not support chip select, you are recommended to tie this port to 1.
j204c_tx_avs_address	10	Input	For Avalon memory-mapped slave, each slave access is based on byte-based offset. For example, address = 0 selects the first four bytes of the slave register and the address = 4 selects the next four bytes of the slave register space.
j204c_tx_avs_writedata	32	Input	32-bit data for write transfers. The width of this signal and the j204c_tx_avs_readdata[31:0] signal must be the same if both signals are present.
j204c_tx_avs_read	1	Input	This signal is asserted to indicate a read transfer. This is an active high signal and requires the j204c_tx_avs_readdata[31:0] signal to be in use.
j204c_tx_avs_write	1	Input	This signal is asserted to indicate a write transfer. This is an active high signal and requires the j204c_tx_avs_writedata[31:0] signal to be in use.
j204c_tx_avs_readdata	32	Output	32-bit data driven from the Avalon memory-mapped slave to master in response to a read transfer.
j204c_tx_avs_waitrequest	1	Output	This signal is asserted by the Avalon memory-mapped slave to indicate that it is unable to respond to a read or write request. The JESD204C Intel FPGA IP ties this signal to 0 to return the data in the access cycle.

Signal	Width	Direction	Description
JESD204C TX MAC Avalon Streaming Interface (Data Channel)			
j204c_tx_avst_data	M*S*WIDTH_MULP*N	Input	The minimum data width = M*S*N. Indicates the converter samples that will be processed by TL. The data format is big endian. If L=1 and M*S*WIDTH_MULP*N=128, the first octet is located at bit[127:120], second octet at bit[119:112], and the last octet at bit[7:0]. If more than one lane is instantiated, lane 0 data is always located in the upper M*S*WIDTH_MULP*N bit data lane, followed by the next lane, with the first octet position for lane 0 is at MSB.
j204c_tx_avst_control	M*S*WIDTH_MULP*CS	Input	Control bits to be inserted as part of CS parameter.
j204c_tx_avst_valid	1	Input	Indicates whether the data from the application layer is valid or invalid. The Avalon streaming sink interface in the TX core cannot be backpressured and assumes that the data is always valid on every cycle when the j204c_tx_avst_ready signal is asserted. • 0—data is invalid • 1—data is valid
j204c_tx_avst_ready	1	Output	Indicates that the Avalon streaming sink interface in the TX core is ready to accept data. The Avalon streaming sink interface asserts this signal on the JESD204C transport state of USER_DATA phase. The ready latency is 0.
j204c_tx_frame_ready	1	Output	Indicates that the link layer is ready to accept data. The link layer asserts this signal on a predetermined time before the assertion of the j204c_tx_avst_ready signal.

Signal	Width	Direction	Description
JESD204C TX MAC Command (Command Channel)			
j204c_tx_cmd_data	L*n	Input	Indicates a 6/18-bit user command (per lane) at txlink_clk clock rate. The data format is big endian. If more than one lane is instantiated, lane 0 data is always located at the upper 18 bits or 6 bits of data. Lane L is located at bit[17:0] or bit[5:0], with the first command bit position for lane L at bit[17] or bit[5]. *Note:* n=6 for CRC-12 operation and n =18 for standalone command channel
j204c_tx_cmd_valid	1	Input	Indicates whether the command from the application layer is valid or invalid. The Avalon streaming sink interface in the TX core cannot be backpressured and assumes that data is always valid on every cycle when the j204c_tx_cmd_ready signal is asserted. • 0—data is invalid • 1—data is valid
j204c_tx_cmd_ready	1	Output	Indicates that the Avalon streaming sink interface in the TX core is ready to accept command. The Avalon streaming sink interface asserts this signal on the JESD204C link/transport state of USER_DATA phase. The ready latency is 0.

Signal	Width	Direction	Description
JESD204C Interface			
j204c_tx_sysref	1	Input	SYSREF signal for JESD204C Subclass 1 implementation. For Subclass 0 mode, tie-off this signal to 0.
j204c_tx_somb	1	Output	Start of multiblock.
j204c_tx_soemb	1	Output	Start of extended multiblock.

Signal	Width	Direction	Description
JESD204C TX MAC CSR			
j204c_tx_csr_l	4	Output	Indicates the number of active lanes for the link. The transport layer can use this signal as a compile-time parameter.
j204c_tx_csr_f	8	Output	Indicates the number of octets per frame. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_m	8	Output	Indicates the number of converters for the link. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_cs	2	Output	Indicates the number of control bits per sample. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_n	5	Output	Indicates the converter resolution. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_np	5	Output	Indicates the total number of bits per sample. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_s	5	Output	Indicates the number of samples per converter per frame cycle. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_hd	1	Output	Indicates the high density data format. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_cf	5	Output	Indicates the number of control words per frame clock period per link. The transport layer uses this signal as a compile-time parameter.
j204c_tx_csr_e	8	Output	LEMC period
j204c_tx_csr_testmode	4	Output	0000: No test mode 0001: Scrambler disabled Other values are reserved.

Signal	Width	Direction	Description
JESD204C TX MAC Out-of-band (OOB)			
j204c_tx_int	1	Output	Interrupt pin for the JESD204C Intel FPGA IP. Interrupt is asserted when any error or synchronization request is detected. Configure the tx_err_enable register to set the type of error that can trigger an interrupt.

continued...

Signal	Width	Direction	Description
j204c_tx2rx_lbdata	L*132	Output	Output as 132-bit width data before the TX gearbox to connect to the RX core (same signal name) for 2-block loopback function. If L>0, LSB of this bus is mapped to lane 0. MSB is mapped to lane L–1.
txphy_data	64*L	Output	TX PHY parallel data.
tx_fifo_full	L	Input	Indicates the TX core interface FIFO is full.

Note: For information about the transceiver PHY signals, refer to the *Port Information* section in the *E-tile Transceiver PHY User Guide*.

Related Information

Port Information chapter in the E-tile Transceiver PHY User Guide
Provides more information about the TX and RX PHY signals.

7.2. Receiver Signals

Table 24. Top-level Receiver IP Core Signals

Signal	Width	Direction	Description
JESD204C RX Clocks and Resets			
j204c_pll_refclk	1	Input	Transceiver reference clock signal.
j204c_rxlink_clk	1	Input	This clock is equal to the RX data rate divided by 132. Generated from the same PLL as rxframe_clk.
j204c_rxlclk_ctrl	1	Input	Generated from the same PLL as rxlink_clk and rxframe_clk. This clock acts as a phase information for rxlink_clk to handle CDC between rxlink_clk and rxframe_clk.
j204c_rxframe_clk	1	Input	Synchronous with rxlink_clk. Frequency is equal, 2x, or 4x rxlink_clk. Generated from the same PLL as rxlink_clk.
j204c_rxfclk_ctrl	1	Input	Generated from the same PLL as rxlink_clk and rxframe_clk. This clock acts as a phase information for rxframe_clk to handle CDC between rxlink_clk and rxframe_clk.
j204c_rx_avs_clk	1	Input	Avalon memory-mapped interface clock.
j204c_reconfig_clk	1	Input	Transceiver reconfiguration clock. During duplex mode, both TX and RX share the same reconfig pins.
j204c_rx_rst_n	1	Input	Active-low asynchronous reset signal for MAC LL and TL.
j204c_rx_phy_rst_n	1	Input	Active-low asynchronous reset signal for PHY.
j204c_rx_avs_rst_n	1	Input	Active-low asynchronous reset signal for RX Avalon memory-mapped interface.
j204c_reconfig_reset	1	Input	Active-high reset signal for transceiver reconfiguration. During duplex mode, both TX and RX share the same reconfig pins.

Signal	Width	Direction	Description
Transceiver Interface			
rx_serial_data	L	Input	Differential high speed serial input data. The clock is recovered from the serial data stream.
rx_serial_data_n	L	Input	Differential high speed serial input data. The clock is recovered from the serial data stream. You do not need to connect this signal at the top-level pinout for proper compilation.
rx_ready	L	Output	Indicates that the transceiver RX (per lane) is ready.
rx_pma_ready	L	Output	Indicates that the transceiver RX PMA (per lane) is ready.
j204c_reconfig_read	1	Input	During duplex mode, both TX and RX share the same reconfig pins.
j204c_reconfig_write	1	Input	During duplex mode, both TX and RX share the same reconfig pins.
j204c_reconfig_address	ceil (log2(L)) +19	Input	During duplex mode, both TX and RX share the same reconfig pins. The lower 19 bits specify the address, the upper bits (log2(L)) specify the channel. If L=1, total address bit is always 19 bits.
j204c_reconfig_readdata	8	Output	During duplex mode, both TX and RX share the same reconfig pins.
j204c_reconfig_writedata	8	Input	During duplex mode, both TX and RX share the same reconfig pins.
j204c_reconfig_waitrequest	1	Output	Wait request signal. During duplex mode, both TX and RX share the same reconfig pins.

Signal	Width	Direction	Description
JESD204C RX MAC Avalon Memory-Mapped Interface			
j204c_rx_avs_chipselect	1	Input	When this signal is present, the slave port ignores all Avalon memory-mapped signals unless this signal is asserted. This signal must be used in combination with read or write. If the Avalon memory-mapped bus does not support chip select, you are recommended to tie this port to 1.
j204c_rx_avs_address	10	Input	For Avalon memory-mapped slave, each slave access is based on byte-based offset. For example, address = 0 selects the first four bytes of the slave register and the address = 4 selects the next four bytes of the slave register space.
j204c_rx_avs_writedata	32	Input	32-bit data for write transfers.
j204c_rx_avs_read	1	Input	This signal is asserted to indicate a read transfer. This is an active high signal and requires the j204c_rx_avs_readdata[31:0] signal to be in use.

continued...

Signal	Width	Direction	Description
j204c_rx_avs_write	1	Input	This signal is asserted to indicate a write transfer. This is an active high signal and requires the j204c_rx_avs_writedata[31:0] signal to be in use.
j204c_rx_avs_readdata	32	Output	32-bit data driven from the Avalon memory-mapped slave to master in response to a read transfer.
j204c_rx_avs_waitrequest	1	Output	This signal is asserted by the Avalon memory-mapped slave to indicate that it is unable to respond to a read or write request. The JESD204C Intel FPGA IP ties this signal to 0 to return the data in the access cycle.

Signal	Width	Direction	Description
JESD204C RX MAC Avalon Streaming Interface (Data Channel)			
j204c_rx_avst_data	M*S*WIDTH_MULP*N	Output	The minimum data width = M*S*N. Indicates the converter samples that will be processed by TL. The data format is big endian. If L=1 and M*S*WIDTH_MULP*N=128, the first octet is located at bit[127:120], second octet at bit[119:112], and the last octet at bit[7:0]. If more than one lane is instantiated, lane 0 data is always located in the upper and M*S*WIDTH_MULP*N bit data lane, followed by the next lane, with the first octet position for lane 0 is at MSB.
j204c_rx_avst_control	M*S*WIDTH_MULP*CS	Output	Control bits that were inserted as part of CS parameter.
j204c_rx_avst_valid	1	Output	Indicates whether the data to the application layer is valid or invalid. The Avalon streaming sink interface in the RX core cannot be backpressured and assumes that the data is always valid on every cycle when the j204c_rx_avst_ready signal is asserted. • 0—data is invalid • 1—data is valid
j204c_rx_avst_ready	1	Input	Indicates that the Avalon streaming sink interface in the application layer is ready to accept data. The Avalon streaming sink interface asserts this signal on the JESD204C transport state of USER_DATA phase. The ready latency is 0.
j204c_rx_crc_err	L	Output	Indicates when CRC error is detected on previous multiblock.

Signal	Width	Direction	Description
JESD204C RX MAC Command (Command Channel)			
j204c_rx_cmd_data	L*n	Output	Indicates a 6/18-bit user command (per lane) at rxlink_clk clock rate. The data format is big endian. If more than one lane is instantiated, lane 0 data is always located at the upper 18 bits or 6 bits of data. Lane L is located at bit[17:0] or bit[5:0], with the first command bit position for lane L at bit[17] or bit[5].

continued...

Signal	Width	Direction	Description
			Note: n=6 for CRC-12 operation and n =18 for standalone command channel
j204c_rx_cmd_valid	1	Output	Indicates whether the command from the link layer is valid or invalid when the j204c_rx_cmd_ready signal is asserted. • 0—data is invalid • 1—data is valid
j204c_rx_cmd_ready	1	Input	Indicates that the transport or application layer is ready to accept command. The application layer interface asserts this signal on the JESD204C link/ transport state of USER_DATA phase. The ready latency is 0.
j204c_rx_cmd_par_err	L or 1	Output	Indicates when parity error is detected. • Width is 1 if you enable single lane mode. • Width is L if you disable single lane mode.

Signal	Width	Direction	Description
JESD204C Interface			
j204c_rx_sysref	1	Input	SYSREF signal for JESD204C Subclass 1 implementation. For Subclass 0 mode, tie-off this signal to 0.
j204c_rx_somb	1	Output	Indicates the start of multiblock.
j204c_rx_soemb	1	Output	Indicates the start of extended multiblock.
j204c_rx_sh_lock	1	Output	Indicates sync header lock.
j204c_rx_emb_lock	1	Output	Indicates EMB lock.
j204c_rx_dev_emblock_align	1	Output	Indicates that all EMB blocks of all the lanes in a JESD204C IP instance are aligned. *Note:* Applicable only when you turn on the **Multilink mode** parameter.
j204c_rx_dev_lane_align	1	Output	Indicates that all lanes in a JESD204C IP instance are aligned.
j204c_rx_alldev_emblock_align	1	Input	For multilink synchronization, input the j204c_rx_dev_emblock_align signals from all the JESD204C IP instances to an AND gate and connect the AND gate output to this signal. *Note:* Applicable only when you turn on the **Multilink mode** parameter.
j204c_rx_alldev_lane_align	1	Input	For multilink synchronization, input the j204c_rx_dev_lane_align signals from all the JESD204C IP instances to an AND gate and connect the AND gate output to this signal. For single device, connect the j204c_rx_dev_lane_align signal back to this signal.

Signal	Width	Direction	Description
JESD204C RX MAC CSR			
j204c_rx_csr_l	4	Output	Indicates the number of active lanes for the link. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_f	8	Output	Indicates the number of octets per frame. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_m	8	Output	Indicates the number of converters for the link. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_cs	2	Output	Indicates the number of control bits per sample. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_n	5	Output	Indicates the converter resolution. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_np	5	Output	Indicates the total number of bits per sample. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_s	5	Output	Indicates the number of samples per converter per frame cycle. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_hd	1	Output	Indicates the high density data format. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_cf	5	Output	Indicates the number of control words per frame clock period per link. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_e	8	Output	LEMC period. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_testmode	4	Output	0000: No test mode. 00x1: Descrambler disabled. 001x: 2-block loopback mode enabled. Other values are reserved.

Signal	Width	Direction	Description
JESD204C RX MAC Out-of-band (OOB)			
j204c_rx_int	1	Output	Interrupt pin for the JESD204C Intel FPGA IP. Interrupt is asserted when any error is detected. Configure the rx_err_enable register to set the type of error that can trigger an interrupt.
j204c_tx2rx_lbdata	L*132	Input	Multiplexed with the RX gearbox output when 2-block loopback mode is enabled through bit-10 rx_2b_lben (offset 0x0) to connect to the TX core in the duplex setup (same signal name). If L>0, LSB of this bus is mapped to lane 0. MSB is mapped to lane L-1. This signal only exists in simplex mode. When the IP is configured as duplex, the parallel loopback path is connected from TX to RX internally.

Table 25. Top-level Receiver Base Core Signals

Signal	Width	Direction	Description
JESD204C RX Clocks and Resets			
j204c_rxlink_clk	1	Input	This clock is equal to the RX data rate divided by 132. Generated from the same PLL as rxframe_clk.
j204c_rxlclk_ctrl	1	Input	Generated from the same PLL as rxlink_clk and rxframe_clk. This clock acts as a phase information for rxlink_clk to handle CDC between rxlink_clk and rxframe_clk.
j204c_rxframe_clk	1	Input	Synchronous with rxlink_clk. Frequency is equal, 2x, or 4x rxlink_clk. Generated from the same PLL as rxlink_clk.
j204c_rxfclk_ctrl	1	Input	Generated from the same PLL as rxlink_clk and rxframe_clk. This clock acts as a phase information for rxframe_clk to handle CDC between rxlink_clk and rxframe_clk.
j204c_rx_avs_clk	1	Input	Avalon memory-mapped interface clock.
j204c_rx_rst_n	1	Input	Active-low asynchronous reset signal for MAC LL and TL.
j204c_rx_phy_rst_n	1	Input	Active-low asynchronous reset signal for PHY.
j204c_rx_avs_rst_n	1	Input	Active-low asynchronous reset signal for RX Avalon memory-mapped interface.

Signal	Width	Direction	Description
JESD204C RX MAC Avalon Memory-Mapped Interface			
j204c_rx_avs_chipselect	1	Input	When this signal is present, the slave port ignores all Avalon memory-mapped signals unless this signal is asserted. This signal must be used in combination with read or write. If the Avalon memory-mapped bus does not support chip select, you are recommended to tie this port to 1.
j204c_rx_avs_address	10	Input	For Avalon memory-mapped slave, each slave access is based on byte-based offset. For example, address = 0 selects the first four bytes of the slave register and the address = 4 selects the next four bytes of the slave register space.
j204c_rx_avs_writedata	32	Input	32-bit data for write transfers.
j204c_rx_avs_read	1	Input	This signal is asserted to indicate a read transfer. This is an active high signal and requires the j204c_rx_avs_readdata[31:0] signal to be in use.

continued...

Signal	Width	Direction	Description
j204c_rx_avs_write	1	Input	This signal is asserted to indicate a write transfer. This is an active high signal and requires the j204c_rx_avs_writedata[31:0] signal to be in use.
j204c_rx_avs_readdata	32	Output	32-bit data driven from the Avalon memory-mapped slave to master in response to a read transfer.
j204c_rx_avs_waitrequest	1	Output	This signal is asserted by the Avalon memory-mapped slave to indicate that it is unable to respond to a read or write request. The JESD204C Intel FPGA IP ties this signal to 0 to return the data in the access cycle.

Signal	Width	Direction	Description
JESD204C RX MAC Avalon Streaming Interface (Data Channel)			
j204c_rx_avst_data	M*S*WIDTH_MULP*N	Output	The minimum data width = M*S*N. Indicates the converter samples that will be processed by TL. The data format is big endian. If L=1 and M*S*WIDTH_MULP*N=128, the first octet is located at bit[127:120], second octet at bit[119:112], and the last octet at bit[7:0]. If more than one lane is instantiated, lane 0 data is always located in the upper and M*S*WIDTH_MULP*N bit data lane, followed by the next lane, with the first octet position for lane 0 is at MSB.
j204c_rx_avst_control	M*S*WIDTH_MULP*CS	Output	Control bits that were inserted as part of CS parameter.
j204c_rx_avst_valid	1	Output	Indicates whether the data to the application layer is valid or invalid. The Avalon streaming sink interface in the RX core cannot be backpressured and assumes that the data is always valid on every cycle when the j204c_rx_avst_ready signal is asserted. • 0—data is invalid • 1—data is valid
j204c_rx_avst_ready	1	Input	Indicates that the Avalon streaming sink interface in the application layer is ready to accept data. The Avalon streaming sink interface asserts this signal on the JESD204C transport state of USER_DATA phase. The ready latency is 0.
j204c_rx_crc_err	L	Output	Indicates when CRC error is detected on previous multiblock.

Signal	Width	Direction	Description
JESD204C RX MAC Command (Command Channel)			
j204c_rx_cmd_data	L*n	Output	Indicates a 6/18-bit user command (per lane) at rxlink_clk clock rate. The data format is big endian. If more than one lane is instantiated, lane 0 data is always located at the upper 18 bits or 6 bits of data. Lane L is located at bit[17:0] or bit[5:0], with the first command bit position for lane L at bit[17] or bit[5].

continued...

Signal	Width	Direction	Description
			Note: n=6 for CRC-12 operation and n =18 for standalone command channel
j204c_rx_cmd_valid	1	Output	Indicates whether the command from the link layer is valid or invalid when the j204c_rx_cmd_ready signal is asserted. • 0—data is invalid • 1—data is valid
j204c_rx_cmd_ready	1	Input	Indicates that the transport or application layer is ready to accept command. The application layer interface asserts this signal on the JESD204C link/ transport state of USER_DATA phase. The ready latency is 0.
j204c_rx_cmd_par_err	L or 1	Output	Indicates when parity error is detected. • Width is 1 if you enable single lane mode. • Width is L if you disable single lane mode.

Signal	Width	Direction	Description
JESD204C Interface			
j204c_rx_sysref	1	Input	SYSREF signal for JESD204C Subclass 1 implementation. For Subclass 0 mode, tie-off this signal to 0.
j204c_rx_somb	1	Output	Indicates the start of multiblock.
j204c_rx_soemb	1	Output	Indicates the start of extended multiblock.
j204c_rx_sh_lock	1	Output	Indicates sync header lock.
j204c_rx_emb_lock	1	Output	Indicates EMB lock.
j204c_rx_dev_lane_align	1	Output	Indicates that all lanes for this device are aligned.
j204c_rx_alldev_lane_align	1	Input	For multidevice synchronization, input the j204c_rx_dev_lane_align signals from all the devices to an AND gate and connect the AND gate output to this signal. For single device, connect the j204c_rx_dev_lane_align signal back to this signal.

Signal	Width	Direction	Description
JESD204 RX MAC CSR			
j204c_rx_csr_l	4	Output	Indicates the number of active lanes for the link. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_f	8	Output	Indicates the number of octets per frame. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_m	8	Output	Indicates the number of converters for the link. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_cs	2	Output	Indicates the number of control bits per sample. The transport layer uses this signal as a compile-time parameter.

continued...

Signal	Width	Direction	Description
j204c_rx_csr_n	5	Output	Indicates the converter resolution. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_np	5	Output	Indicates the total number of bits per sample. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_s	5	Output	Indicates the number of samples per converter per frame cycle. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_hd	1	Output	Indicates the high density data format. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_cf	5	Output	Indicates the number of control words per frame clock period per link. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_e	8	Output	LEMC period. The transport layer uses this signal as a compile-time parameter.
j204c_rx_csr_testmode	4	Output	0000: No test mode. 00x1: Descrambler disabled. 001x: 2-block loopback mode enabled. Other values are reserved.

Signal	Width	Direction	Description
JESD204C RX MAC Out-of-band (OOB)			
j204c_rx_int	1	Output	Interrupt pin for the JESD204C Intel FPGA IP. Interrupt is asserted when any error is detected. Configure the rx_err_enable register to set the type of error that can trigger an interrupt.
j204c_tx2rx_lbdata	L*132	Input	Multiplexed with the RX gearbox output when 2-block loopback mode is enabled through bit-10 rx_2b_lben (offset 0x0) to connect to the TX core in the duplex setup (same signal name). If L>0, LSB of this bus is mapped to lane 0. MSB is mapped to lane L-1. This signal only exists in simplex mode. When the IP is configured as duplex, the parallel loopback path is connected from TX to RX internally.

Note: For information about the transceiver PHY signals, refer to the *Port Information* section in the *E-tile Transceiver PHY User Guide*.

Related Information

Port Information chapter in the E-tile Transceiver PHY User Guide
Provides more information about the TX and RX PHY signals.

8. Control and Status Registers

The control and status registers refer to byte addressing as seen by the software, and as implemented by hardware. All registers that are Read-Writable must be protected to comply with Security Development Lifecycle (SDL) practices. You are required to perform the register access protection.

Table 26. Register Access Type and Definition

This table describes the register access type for Intel FPGA IPs.

Access Type	Definition
RO	Software read-only (no effect on write). The value is hard-tied internally to either '0' or '1' and does not vary.
RO/V	Software read-only (no effect on write). The value may vary.
RC	• Software reads and returns the current bit value, then the bit self-clears to 0. • Software read also causes the bit value to be cleared to 0.
RW	• Software reads and returns the current bit value. • Software writes and sets the bit to the desired value.
RW1C	• Software reads and returns the current bit value. • Software writes 0 and has no effect. • Software writes 1 and clears the bit to 0 if the bit has been set to 1 by hardware. • Hardware sets the bit to 1. • Software clear has higher priority than hardware set.
RW1S	• Software reads and returns the current bit value. • Software writes 0 and has no effect. • Software writes 1 and sets the bit to 1. • Hardware clears the bit to 0 if the bit has been set to 1 by software. • Software set has higher priority than hardware clear.

8.1. Transmitter Registers

Table 27. lane_ctrl_common

Common lane control and assignment. The common lane control applies to all lanes in the link.

Offset: 0x0

Note: For bits that are compile-time specific, you must recompile to change the reset value.

Bit	Name	Description	Attribute	Reset
31:2	Reserved	Reserved	RV	0x0
1	scr_disable	Setting this bit disables TX scrambler	RW	Compile-time specific
0	bit_reversal	This is a compile-time option that you need to set before IP generation. 0 = LSB-first serialization. 1 = MSB-first serialization. *Note:* The JESD204C converter device may support either MSB-first serialization or LSB-first serialization. When bit_reversal = 1, the word aligner reverses TX parallel data bits before transmitting it to the PMA for serialization. For example; in 64-bit mode => D[63:0] is rewired to D[0:63]	RO	Compile-time specific

Table 28. lane_ctrl_0

Lane control and assignment for Lane 0.

Offset: 0x4

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 29. lane_ctrl_1

Lane control and assignment for Lane 1.

Offset: 0x8

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 30. lane_ctrl_2

Lane control and assignment for Lane 2.

Offset: 0xC

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 31. lane_ctrl_3

Lane control and assignment for Lane 3.

Offset: 0x10

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 32. lane_ctrl_4

Lane control and assignment for Lane 4.

Offset: 0x14

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 33. lane_ctrl_5

Lane control and assignment for Lane 5.

Offset: 0x18

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 34. lane_ctrl_6

Lane control and assignment for Lane 6.

Offset: 0x1C

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 35. lane_ctrl_7

Lane control and assignment for Lane 7.

Offset: 0x20

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 36. lane_ctrl_8

Lane control and assignment for Lane 8.

Offset: 0x24

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 37. lane_ctrl_9

Lane control and assignment for Lane 9.

Offset: 0x28

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 38. lane_ctrl_10

Lane control and assignment for Lane 10.

Offset: 0x2C

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 39. lane_ctrl_11

Lane control and assignment for Lane 11.

Offset: 0x30

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 40. lane_ctrl_12

Lane control and assignment for Lane 12.

Offset: 0x34

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 41. lane_ctrl_13

Lane control and assignment for Lane 13.

Offset: 0x38

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 42. lane_ctrl_14

Lane control and assignment for Lane 14.

Offset: 0x3C

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 43. lane_ctrl_15

Lane control and assignment for Lane 15.

Offset: 0x40

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 44. tl_ctrl

Transport layer control.

Offset: 0x50

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 45. sysref_ctrl

SYSREF control.

Offset: 0x54

Note: For bits that are compile-time specific, you must recompile to change the reset value.

Bit	Name	Description	Attribute	Reset
31:16	Reserved	Reserved	RV	0x0
15:8	`lemc_offset`	Upon the detection of the rising edge of SYSREF in continuous mode or single detect mode, the LEMC counter will be reset to the value set in `lemc_offset`. LEMC counter operates in link clock domain, therefore the legal value for the counter is from 0 to (E*16)-1. • In the event that (E*16)-1 > 255, the design has no capability to adjust the LEMC for offset greater than 255. • If (E*16-1) < 255, and an out-of-range value is set, the LEMC offset will be internally reset to 0. *Note:* By default, the rising edge of SYSREF resets the LEMC counter to 0. However, if the system design has a large phase offset between the SYSREF sampled by the converter device and the FPGA, you can virtually shift the SYSREF edges by changing the LEMC offset reset value using this register.	RW	Compile-time specific
7:3	Reserved	Reserved	RV	0x0
2	`sysref_singledet`	This register enables LEMC realignment with a single sample of the rising edge of SYSREF. The bit is auto-cleared by the hardware once SYSREF is sampled. If you require SYSREF to be sampled again (due to link reset or reinitialization), you must set this bit again. This register also has another critical function. The JESD204C IP will never send EoEMB unless at least a SYSREF edge is sampled. This is to prevent race condition between SYSREF being sampled at RX (converter device) and the deterministic timing of EoEMB transmission. • 0 = Any rising edge of SYSREF will not reset the LEMC counter. • 1 = Resets the LEMC counter on the first rising edge of SYSREF and then clears this bit. (Default) *Note:* Intel recommends that you use `sysref_singledet` with `sysref_alwayson` even if you want	RW1S	0x1

continued...

Bit	Name	Description	Attribute	Reset
		to do SYSREF continuous detection mode. This is because this register is able to indicate whether SYSREF was ever sampled. This register also prevents race condition as mentioned above. Using only SYSREF single detect mode will not be able to detect incorrect SYSREF period.		
1	`sysref_alwayson`	This register enables LEMC realignment at every rising edge of SYSREF. LEMC counter resets when every SYSREF transition from 0 to 1 is detected. • 0 = Any rising edge of SYSREF will not reset the LEMC counter. • 1 = Continuously resets LEMC counter at every SYSREF rising edge. *Note:* When this bit is set, the SYSREF period will be checked that it never violates internal extended multiblock period and this period can only be n-integer multiplied of (E*32). If the SYSREF period is different from the local extended multiblock period, the IP asserts the `sysref_lemc_err` (0x60) register and triggers an interrupt. If you want to change the SYSREF period, this bit should be set to 0 first. After SYSREF clock has stabilized, this bit is set to 1 to sample the rising edges of the new SYSREF.	RW	0x0
0	`link_reinit`	The JESD204C IP reinitializes the TX link by resetting all internal pipestages and status, but not including SYSREF detection information. This bit automatically clears once link reinitialization is entered by hardware. • 0 = No link reinitialization request (Default) • 1 = Reinitialize the link.	RW1S	0x0

Table 46. tx_err

This register logs errors detected in the FPGA IP. Each set bit in the register will generate an interrupt, if enabled by corresponding bits in the TX Error Enable register (*tx_err_enable (0x64)*). After servicing the interrupt, the software must clear the appropriate serviced interrupt status bit and ensure that no other interrupts are pending.

Offset: 0x60

Bit	Name	Description	Attribute	Reset
31:9	Reserved	Reserved	RV	0x0
8	`tx_gb_overflow_err`	Assert when overflow happens on any of the lane's TX gearbox.	RW1C	0x0
7	`tx_gb_underflow_err`	Assert when underflow happens on any of the lane's TX gearbox.	RW1C	0x0

continued...

Bit	Name	Description	Attribute	Reset
6	Reserved	Reserved	RV	0x0
5	pcfifo_full_err	Detected 1 or more lanes of Phase Compensation FIFO is full unexpectedly when JESD204C link is running. *Note:* User MUST reset JESD204C link if this bit is triggered. The transceiver channel, and the JESD204C IP core link reset must be applied.	RW1C	0x0
4	tx_ready_err	Detected 1 or more lanes of tx_ready (from the transceiver) drop when the JESD204C link is running.	RW1C	0x0
3	cmd_invalid_err	This error bit is applicable only if Command Channel is used in the JESD204C link. This error bit will be asserted if the upstream component deassert the j204c_tx_cmd_valid signal while Link Layer is requesting for command (via j204c_tx_cmd_ready).	RW1C	0x0
2	frame_data_invalid_err	This error bit is applicable only if you use Intel FPGA transport layer in your design. This error bit will be asserted if the upstream component deasserts j204c_tx_avst_valid signal at the Intel FPGA transport layer Avalon-ST bus. The transport layer expects the upstream device in the system will always send the valid data with zero latency when j204c_tx_avst_ready is asserted by the transport layer.	RW1C	0x0
1	dll_data_invalid_err	This error bit will be asserted if the link layer TX detects data invalid on the Avalon-ST bus when data is requested. By design, the JESD204C TX link layer expects the upstream device (JESD204C transport layer) will always send the valid data with zero latency when ready is asserted.	RW1C	0x0
0	sysref_lemc_err	When the sysref_ctrl (0x54) sysref_alwayson register is set to 1, the LEMC counter will check whether SYSREF period matches the LEMC counter where it is n-integer multiplier of the (E*32). If SYSREF period does not match the LEMC period, this bit will be asserted.	RW1C	0x0

Table 47. tx_err_en

This register enables the error types that will generate interrupt. Setting 0 to the register bits will disable the specific error type from generating interrupt.

Offset: 0x64

Bit	Name	Description	Attribute	Reset
31:9	Reserved	Reserved	RV	0x0
8	tx_gb_overflow_err_en	TX gearbox overflow error interrupt enable	RW	0x1
7	tx_gb_underflow_err_en	TX gearbox underflow error interrupt enable	RW	0x1
6	Reserved	Reserved	RV	0x0
5	pcfifo_full_err_en	PCFIFO full error interrupt enable	RW	0x1
4	tx_ready_err_en	Transceiver TX Ready error interrupt enable	RW	0x1
3	cmd_invalid_err_en	Command invalid error interrupt enable	RW	0x0
2	frame_data_invalid_err_en	Frame data invalid error interrupt enable	RW	0x0
1	dll_data_invalid_err_en	Link data invalid error interrupt enable	RW	0x0
0	sysref_lemc_err_en	*SYSREF* LEMC error interrupt enable	RW	0x1

Table 48. tx_err_link_reinit

This register enables the error types that will generate link reinitialization. Setting 0 to the register bits will disable the specific error type from link reinitialization.

Offset: 0x68

Bit	Name	Description	Attribute	Reset
31:9	Reserved	Reserved	RV	0x0
8	tx_gb_overflow_err_en_reinit	TX gearbox overflow error reinitialization enable.	RW	0x0
7	tx_gb_underflow_err_en_reinit	TX gearbox underflow error reinitialization enable.	RW	0x0
6	Reserved	Reserved	RV	0x0
5	pcfifo_full_err_en_reinit	PCFIFO full error reinitialization enable. *Note:* Link reinitialization sequence does not cover the transceiver reinitialization steps, hence such error will not be recovered via link reinit.	RW	0x0
4	tx_ready_err_en_reinit	Transceiver TX ready error reinitialization enable.	RW	0x0

continued...

Bit	Name	Description	Attribute	Reset
		Note: Link reinitialization sequence does not cover transceiver reinitialization steps, hence such error will not be recovered via link reinitialization.		
3	cmd_invalid_err_en_reinit	Command invalid error reinitialization enable	RW	0x0
2	frame_data_invalid_err_en_reinit	Frame data invalid error reinitialization enable	RW	0x0
1	dll_data_invalid_err_en_reinit	Link data invalid error reinitialization enable	RW	0x0
0	sysref_lemc_err_en_reinit	*SYSREF* LEMC error reinitialization enable	RW	0x0

Table 49. tx_status0

Monitor ports of internal signals and counter which will be useful for debugging.

Offset: 0x80

Note: For bits that are compile-time specific, you must recompile to change the reset value.

Bit	Name	Description	Attribute	Reset
31:12	Reserved	Reserved	RV	0x0
11	sysref_det_pending	Indicate that SYSREF is yet to be detected. You need to set the sysref_singledet bit to enable link initialization.	ROV	0x0
10	reinit_in_prog	Indicates that auto or manual link reinitialization is in progress.	ROV	0x0
9:2	lemc_period	Represent E: the number of multiblock in an extended multiblock.	RO	Compile-time specific
1:0	sh_config	Sync header encoding configuration b00: CRC-12 b01: Standalone command channel b10: Reserved b11: Reserved	RO	Compile-time specific

Table 50. tx_status1

Monitor ports of internal signals and counter which will be useful for debugging.

Offset: 0x84

Bit	Name	Description	Attribute	Reset
31:16	Reserved	Reserved	RV	0x0
15	lane15_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 15	ROV	0x0
14	lane14_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 14	ROV	0x0

continued...

Bit	Name	Description	Attribute	Reset
13	lane13_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 13	ROV	0x0
12	lane12_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 12	ROV	0x0
11	lane11_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 11	ROV	0x0
10	lane10_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 10	ROV	0x0
9	lane9_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 9	ROV	0x0
8	lane8_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 8	ROV	0x0
7	lane7_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 7	ROV	0x0
6	lane6_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 6	ROV	0x0
5	lane5_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 5	ROV	0x0
4	lane4_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 4	ROV	0x0
3	lane3_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 3	ROV	0x0
2	lane2_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 2	ROV	0x0
1	lane1_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 1	ROV	0x0
0	lane0_tx_pcfifo_full	TX phase compensation FIFO status full flag for Lane 0	ROV	0x0

Table 51. tx_status2

Monitor ports of internal signals and counter which will be useful for debugging.

Offset: 0x88

Bit	Name	Description	Attribute	Reset
31:16	Reserved	Reserved	RV	0x0
15	lane15_tx_xcvr_ready	TX transceiver ready status flag for Lane 15	ROV	0x0
14	lane14_tx_xcvr_ready	TX transceiver ready status flag for Lane 14	ROV	0x0
13	lane13_tx_xcvr_ready	TX transceiver ready status flag for Lane 13	ROV	0x0
12	lane12_tx_xcvr_ready	TX transceiver ready status flag for Lane 12	ROV	0x0
11	lane11_tx_xcvr_ready	TX transceiver ready status flag for Lane 11	ROV	0x0

continued...

Bit	Name	Description	Attribute	Reset
10	lane10_tx_xcvr_ready	TX transceiver ready status flag for Lane 10	ROV	0x0
9	lane9_tx_xcvr_ready	TX transceiver ready status flag for Lane 9	ROV	0x0
8	lane8_tx_xcvr_ready	TX transceiver ready status flag for Lane 8	ROV	0x0
7	lane7_tx_xcvr_ready	TX transceiver ready status flag for Lane 7	ROV	0x0
6	lane6_tx_xcvr_ready	TX transceiver ready status flag for Lane 6	ROV	0x0
5	lane5_tx_xcvr_ready	TX transceiver ready status flag for Lane 5	ROV	0x0
4	lane4_tx_xcvr_ready	TX transceiver ready status flag for Lane 4	ROV	0x0
3	lane3_tx_xcvr_ready	TX transceiver ready status flag for Lane 3	ROV	0x0
2	lane2_tx_xcvr_ready	TX transceiver ready status flag for Lane 2	ROV	0x0
1	lane1_tx_xcvr_ready	TX transceiver ready status flag for Lane 1	ROV	0x0
0	lane0_tx_xcvr_ready	TX transceiver ready status flag for Lane 0	ROV	0x0

Table 52. tx_converter_param1

Link and transport control configuration per converter parameters.

Offset: 0xC0

Note: For bits that are compile-time specific, you must recompile to change the reset value.

Bit	Name	Description	Attribute	Reset
31:30	CS	Number of control bits per converter sample. 1-based value. For example, 0=0 bit, 1=1 bit.	RO	Compile-time specific
29	HD	High Density format.	RO	Compile-time specific
28:24	N	Number of data bits per converter sample. 0-based value. For example, 0=1 bit, 1=2 bits. Note that CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7.	RO	Compile-time specific
23:16	M	Number of converter per device. 0-based value. For example, 0=1 converter, 1=2 converters. *Note:* CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7.	RO	Compile-time specific

continued...

Bit	Name	Description	Attribute	Reset
15:8	F	*Note:* CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7.Number of octets per frame. 0-based value. For example, 0=1 octet, 1=2 octets.	RO	Compile-time specific
7:4	Reserved	Reserved	RV	0x0
3:0	L	Number of lanes per link. 0-based value. For example, 0=1 lane, 1=2 lanes. *Note:* CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7.	RO	Compile-time specific

Table 53. tx_converter_param2

Link and transport control configuration per converter parameters.

Offset: 0xC4

Note: For bits that are compile-time specific, you must recompile to change the reset value.

Bit	Name	Description	Attribute	Reset
31:24	E	Number of multiblock within an extended multiblock. 0-based value. For example, 0=1 multiblock to form extended multiblock, 1=2 multiblock to form an extended multiblock. If (256 Mod F)=1, E must be greater than 1. (The register value should be greater than 0). *Note:* CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7.	RO	Compile-time specific
23:21	Reserved	Reserved	RV	0x0
20:16	CF	Number of control words per frame clock per link. 1-based value. I.e 0=0 word, 1=1 word.	RO	Compile-time specific
15:13	Reserved	Reserved	RV	0x0
12:8	S	Number of samples per converter frame cycle. 0-based value. For example, 0=1 sample, 1=2 samples. *Note:* CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7.	RO	Compile-time specific
7:5	subclass_ver	Device Subclass Version • b000: Subclass 0 • b001: Subclass 1	RO	Compile-time specific
4:0	NP	Number of data bits+control bits+tail bits per converter sample. 0-based value. For example, 0=1 bit, 1=2 bits.	RO	Compile-time specific

continued...

Bit	Name	Description	Attribute	Reset
		Note: CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7.		

8.2. Receiver Registers

Table 54. lane_ctrl_common

Common lane control and assignment. The common lane control applies to all lanes in the link.

Offset: 0x0

Note: For bits that are compile-time specific, you must recompile to change the reset value.

Bit	Name	Description	Attribute	Reset
31:14	Reserved	Reserved	RV	0x0
13:11	Reserved	Reserved	RV	0x0
10	rx_2b_lben	Enables the 132-bit interface loopback from TX. Instead of taking RX gearbox data, TX loopback data is multiplexed in for subsequent RX operation.	RW	0x0
9:6	rx_thresh_sh_err	The number of consecutive erroneous sequences required to force the algorithm back to initial SH_INIT. 0-based value. 0=threshold of 1. `d15= threshold of 16.	RW	Compile-time specific
5:3	rx_thresh_emb_err	The number of consecutive erroneous sequences required to force the algorithm back to initial EMB_INIT. 0-based value. 0=threshold of 1. `d7= threshold of 8.	RW	Compile-time specific
2	Reserved	Reserved	RV	0x0
1	scr_disable	Setting this bit disables RX descrambler.	RW	Compile-time specific
0	bit_reversal	This is a compile-time option which needs to be set before IP generation. • 0 = LSB-first serialization. • 1 = MSB-first serialization. *Note:* JESD204C converter device may support either MSB-first serialization or LSB-first serialization. When bit_reversal = 1, the word aligner reverses RX parallel data bits upon receiving the PMA deserialized data. For example; in 64-bit mode => D[63:0] is rewired to D[0:63]	RO	Compile-time specific

Table 55. lane_ctrl_0

Lane control and assignment for Lane 0.

Offset: 0x4

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 56. lane_ctrl_1

Lane control and assignment for Lane 1.

Offset: 0x8

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 57. lane_ctrl_2

Lane control and assignment for Lane 2.

Offset: 0xC

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 58. lane_ctrl_3

Lane control and assignment for Lane 3.

Offset: 0x10

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 59. lane_ctrl_4

Lane control and assignment for Lane 4.

Offset: 0x14

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 60. lane_ctrl_5

Lane control and assignment for Lane 5.

Offset: 0x18

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 61. lane_ctrl_6

Lane control and assignment for Lane 6.

Offset: 0x1C

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 62. lane_ctrl_7

Lane control and assignment for Lane 7.

Offset: 0x20

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 63. lane_ctrl_8

Lane control and assignment for Lane 8.

Offset: 0x24

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 64. lane_ctrl_9

Lane control and assignment for Lane 9.

Offset: 0x28

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 65. lane_ctrl_10

Lane control and assignment for Lane 10.

Offset: 0x2C

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 66. lane_ctrl_11

Lane control and assignment for Lane 11.

Offset: 0x30

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 67. lane_ctrl_12

Lane control and assignment for Lane 12.

Offset: 0x34

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 68. lane_ctrl_13

Lane control and assignment for Lane 13.

Offset: 0x38

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 69. lane_ctrl_14

Lane control and assignment for Lane 14.

Offset: 0x3C

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 70. lane_ctrl_15

Lane control and assignment for Lane 15.

Offset: 0x40

Bit	Name	Description	Attribute	Reset
31:1	Reserved	Reserved	RV	0x0
0	lane_polarity_en	Set 1 to enable lane polarity detection. When set, the RX interface detects and inverts the polarity of the RX data. If the CSR_OPT=1 or POL_EN_ATR=0, this register is RO. Otherwise, it is RW.	RW/RO	POL_ENx

Table 71. tl_ctrl

Transport layer control.

Offset: 0x50

Bit	Name	Description	Attribute	Reset
31:0	Reserved	Reserved	RV	0x0

Table 72. sysref_ctrl

SYSREF control.

Offset: 0x54

Note: For bits that are compile-time specific, you must recompile to change the reset value.

Bit	Name	Description	Attribute	Reset
31:26	Reserved	Reserved	RV	0x0
25	`force_rbd_release`	Setting this bit will force RBD elastic buffer to be released immediately when the latest arrival lane arrived in the system. It indirectly forces `rbd_offset == rx_status0` (0x80) `rbd_count`. This register overrides `rbd_offset`.	RW	Compile-time specific
24:16	`rbd_offset`	RX Buffer Delay (RBD) offset. RX elastic buffer will align the data from multiple lanes of the link and release the buffer at the LEMC boundary (`rbd_offset` = 0). This register provides flexibility for an early RBD release opportunity. Legal value of RBD offset is from (E*16-1) down to 0 as it is aligned in number of link clocks. If `rbd_offset` is set out of the legal value, the RBD elastic buffer will be immediately released.	RW	Compile-time specific
15:8	`lemc_offset`	Upon the detection of the rising edge of SYSREF in continuous mode or single detect mode, the LEMC counter will be reset to the value set in `lemc_offset`. LEMC counter operates in the link clock domain, therefore the legal value for the counter is from 0 to (E*16)-1. • In the event that (E*16)-1 > 255, the design has no capability to adjust the LEMC for offset greater than 255. • If (E*16-1) < 255, and an out-of-range value is set, the LEMC offset will be internally reset to 0. By default, the rising edge of SYSREF resets the LEMC counter to 0. However, if the system design has a large phase offset between the SYSREF sampled by the converter device and the FPGA, you can virtually shift the SYSREF edges by changing the LEMC offset reset value using this register.	RW	Compile-time specific

continued...

Bit	Name	Description	Attribute	Reset
7:3	Reserved	Reserved	RV	0x0
2	sysref_singledet	This register enables LEMC realignment with a single sample of the rising edge of SYSREF. The bit is auto-cleared by hardware once SYSREF is sampled. If the user requires SYSREF to be sampled again (due to link reset or reinitialization), you must set this bit again. This register also has another critical function: JESD204C IP will never send EoEMB unless at least a SYSREF edge is sampled. This is to prevent race condition between SYSREF being sampled at TX (logic device) and the deterministic timing of EoEMB transmission. • 0 = Any rising edge of SYSREF will not reset the LEMC counter. • SYSREF and then clears this bit. (Default) Intel recommends to use 1 = Resets the LEMC counter on the first rising edge of sysref_singledet with sysref_alwayson even if you want to do SYSREF continuous detection mode. This is because this register is able to indicate whether SYSREF was ever sampled. This register also prevents race condition as mentioned above. Using only SYSREF single detect mode will not be able to detect incorrect SYSREF period.	RW1S	0x1
1	sysref_alwayson	This register enables LEMC realignment at every rising edge of SYSREF. LEMC counter is reset when every SYSREF transition from 0 to 1 is detected. 0 = Any rising edge of SYSREF will not reset the LEMC counter. 1 = Continuously resets LEMC counter at every SYSREF rising edge. When this bit is set, the SYSREF period will be checked to make sure it never violates internal extended multiblock period and this period can only be n-integer multiplied of (E*32). *Note:* When this bit is set, the SYSREF period will be checked to make sure it never violates internal extended multiblock period and this period can only be n-integer multiplied of (E*32). If the SYSREF period s different from the local extended multiblock period, the sysref_lemc_err (0x60) register will be asserted and an interrupt will be triggered.	RW	0x0

continued...

Bit	Name	Description	Attribute	Reset
		If you want to change the SYSREF period, this bit should be set to 0 first. After SYSREF clock has stabilized, this bit is set to 1 to sample the rising edges of the new SYSREF.		
0	link_reinit	The JESD204C IP reinitializes the RX link by resetting all internal pipestages and status, but not including SYSREF detection information. (This bit will automatically be cleared once link reinitialization is entered by hardware). • 0 = No link reinit request (Default) • 1 = Reinitialize the link.	RW1S	0x0

Table 73. rx_err

This register logs errors detected in the FPGA IP. Each set bit in the register will generate interrupt, if enabled by corresponding bits in the RX Error Enable register(*rx_err_enable (0x64)*). After servicing the interrupt, software must clear the appropriate serviced interrupt status bit and ensure that no other interrupts are pending.

Offset: 0x60

Bit	Name	Description	Attribute	Reset
31:23	Reserved	Reserved	RV	0x0
22	ecc_fatal_err	Assert when ECC fatal error occurs. This reflects a double bit error detected and uncorrected.	RW1C	0x0
21	ecc_corrected_err	Assert when ECC error has been corrected. This reflects a single bit error detected and corrected.	RW1C	0x0
20	eb_full_err	Assert when any of the RX elastic buffer detected an overflow condition.	RW1C	0x0
19	emb_unlock_err	Assert when any of the EMB alignment logic detected an "unlock" due to error count> error threshold, e.g. EMB_UNLOCK=1.	RW1C	0x0
18	sh_unlock_err	Assert when any of sync header alignment logic detected an "unlock" due to error count> error threshold, e.g. SH_UNLOCK=1.	RW1C	0x0
17	rx_gb_overflow_err	Assert when overflow happens on any of the lane's RX gearbox.	RW1C	0x0
16	rx_gb_underflow_err	Assert when underflow happens on any of the lane's RX gearbox.	RW1C	0x0
15	Reserved	Placed holder for "Uncorrectable FEC error"	RV	0x0
14	crc_err	The RX CRC generator has calculated a parity which does not match the parity received in the sync word	RW1C	0x0
13	Reserved	Place holder for "Smaller than expected payload in command channel". To move this detection to the application layer.	RV	0x0

continued...

Bit	Name	Description	Attribute	Reset
12	Reserved	Place holder for "Invalid command channel header". To move this detection to the application layer.	RV	0x0
11	cmd_par_err	The final parity bit in the command channel data for a given sync word does not match the calculated parity for the received command channel bits.	RW1C	0x0
10	invalid_eoemb	The EoEMB identifier in the pilot signal has an unexpected value.	RW1C	0x0
9	invalid_eomb	The "00001" sequence in the pilot signal is not received at an expected location in the sync word.	RW1C	0x0
8	invalid_sync_header	"11" or "00" received in expected sync header location	RW1C	0x0
7	lane_deskew_err	Asserted when lane to lane deskew exceeds the LEMC boundary. This error will trigger when rbd_offset is not correctly programmed or the lane to lane skew within the device or across multidevice has exceeded the LEMC boundary. EoEMB for all lanes should be within one LEMC boundary. Refer to Deterministic Latency on page 33 for more information.	RW1C	0x0
6	pcfifo_empty_err	Detected 1 or more lanes of Phase Compensation FIFO is empty unexpectedly when JESD204C link is running. *Note:* You MUST reset the JESD204C link if this bit is triggered. The transceiver channel, and the JESD204C IP link reset must be applied.	RW1C	0x0
5	pcfifo_full_err	Detected 1 or more lanes of Phase Compensation FIFO is full unexpectedly when JESD204C link is running. *Note:* You MUST reset the JESD204C link if this bit is triggered. The transceiver channel, and the JESD204C IP link reset must be applied.	RW1C	0x0
4	cdr_locked_err	Detected 1 or more lanes of CDR locked lose lock when JESD204C link is running.	RW1C	0x0
3	cmd_ready_err	This error bit is applicable only if command channel is used in JESD204C link. This error bit will be asserted if the upstream component deasserts j204c_rx_cmd_ready signal while link layer is sending command (via j204c_rx_cmd_valid).	RW1C	0x0
2	frame_data_ready_err	This error bit will be asserted if the RX detects data ready by the upstream component is 0 on the Avalon-ST bus when data is valid. The transport layer	RW1C	0x0

continued...

Bit	Name	Description	Attribute	Reset
		expects the upstream device in the system (Avalon-ST sink component) will always be ready to receive the valid data from the transport layer. *Note:* If this error detection is not required, the user can tie off the data ready signal from the upstream to 1, `j204_rx_avst_ready` in the transport layer.		
1	`dll_data_ready_err`	This error bit will be asserted if the RX detects data ready by the upstream component is 0 on the Avalon-ST bus when data is valid. By design, the JESD204C RX IP core expects the upstream device (JESD204C transport layer/application layer) will always be ready to receive the valid data from JESD204C RX IP. *Note:* If this error detection is not required, the user can tie off the Avalon-ST the `j204_rx_avst_ready` signal to 1.	RW1C	0x0
0	`sysref_lemc_err`	When the `sysref_alwayson` (0x54) register is set to 1, the LEMC counter checks whether the SYSREF period matches the LEMC counter where it is n-integer multiplier of the (E*32). If the SYSREF period does not match the LEMC period, the IP asserts this bit.	RW1C	0x0

Table 74. rx_err_en

This register enables the error types that will generate interrupt. Setting 0 to the register bits will disable the specific error type from generating interrupt.

Offset: 0x64

Bit	Name	Description	Attribute	Reset
31:23	Reserved	Reserved	RV	0x0
22	`ecc_fatal_err_en`	ECC fatal error interrupt enable	RW	0x1
21	`ecc_corrected_err_en`	ECC corrected error interrupt enable	RW	0x0
20	`eb_full_err_en`	Elastic buffer full error interrupt enable	RW	0x1
19	`emb_unlock_err_en`	EMB alignment unlock error interrupt enable	RW	0x1
18	`sh_unlock_err_en`	Sync header alignment unlock error interrupt enable	RW	0x1
17	`rx_gb_overflow_err_en`	Gearbox overflow error interrupt enable	RW	0x1
16	`rx_gb_underflow_err_en`	Gearbox underflow error interrupt enable	RW	0x1

continued...

Bit	Name	Description	Attribute	Reset
15	Reserved	Reserved	RV	0x0
14	crc_err_en	CRC error interrupt enable	RW	0x1
13	Reserved	Reserved	RV	0x0
12	Reserved	Reserved	RV	0x0
11	cmd_par_err_en	Command Parity error interrupt enable	RW	0x1
10	invalid_eoemb_en	Invalid EoEMB error interrupt enable	RW	0x1
9	invalid_eomb_en	Invalid EoMB error interrupt enable	RW	0x1
8	invalid_sync_header_en	Invalid sync header error interrupt enable	RW	0x1
7	lane_deskew_err_en	Lane deskew error interrupt enable	RW	0x1
6	pcfifo_empty_err_en	PCFIFO empty error interrupt enable	RW	0x1
5	pcfifo_full_err_en	PCFIFO Full error interrupt enable	RW	0x1
4	cdr_locked_err_en	CDR lost lock error interrupt enable	RW	0x1
3	cmd_ready_err_en	Command data ready error interrupt enable	RW	0x0
2	frame_data_ready_err_en	Frame data ready error interrupt enable	RW	0x0
1	dll_data_ready_err_en	Link data ready error interrupt enable	RW	0x0
0	sysref_lemc_err_en	SYSREF LEMC error interrupt enable	RW	0x1

Table 75. rx_err_link_reinit

This register enables the error types that will generate link reinitialization. Setting 0 to the register bits will disable the specific error type from link reinitialization.

Offset: 0x68

Bit	Name	Description	Attribute	Reset
31:23	Reserved	Reserved	RV	0x0
22	ecc_fatal_err_en_reinit	ECC fatal error reinitialization enable	RW	0x0
21	ecc_corrected_err_en_reinit	ECC corrected error reinitialization enable	RW	0x0
20	eb_full_err_en_reinit	Elastic buffer full error reinitialization enable	RW	0x0
19	Reserved	Reserved	RV	0x0
18	Reserved	Reserved	RV	0x0

continued...

Bit	Name	Description	Attribute	Reset
17	rx_gb_overflow_err_en_reinit	Gearbox overflow error reinitialization enable	RW	0x0
16	rx_gb_underflow_err_en_reinit	Gearbox underflow error reinitialization enable	RW	0x0
15	Reserved	Reserved	RV	0x0
14	crc_err_en_reinit	CRC error reinitialization enable	RW	0x0
13	Reserved	Reserved	RV	0x0
12	Reserved	Reserved	RV	0x0
11	cmd_par_err_en_reinit	Command Parity error reinitialization enable	RW	0x0
10	invalid_eoemb_en_reinit	Invalid EoEMB error reinitialization enable	RW	0x0
9	invalid_eomb_en_reinit	Invalid EoMB error reinitialization enable	RW	0x0
8	invalid_sync_header_en_reinit	Invalid sync header error reinitialization enable	RW	0x0
7	lane_deskew_err_en_reinit	Lane deskew error reinitialization enable	RW	0x0
6	pcfifo_empty_err_en_reinit	PCFIFO empty error reinitialization enable. Note: Link reinitialization sequence does not cover transceiver reinitialization steps, hence such error will not be recovered via link reinitialization.	RW	0x0
5	pcfifo_full_err_en_reinit	PCFIFO Full error reinitialization enable. Note: Link reinitialization sequence does not cover transceiver reinitialization steps, hence such error will not be recovered via link reinitialization.	RW	0x0
4	cdr_locked_err_en_reinit	CDR lost lock error reinitialization enable. Note: Link reinitialization sequence does not cover transceiver reinitialization steps, hence such error will not be recovered via link reinitialization.	RW	0x0
3	cmd_ready_err_en_reinit	Command data ready error reinitialization enable	RW	0x0

continued...

Bit	Name	Description	Attribute	Reset
2	frame_data_ready_err_en_reinit	Frame data ready error reinitialization enable	RW	0x0
1	dll_data_ready_err_en_reinit	Link data ready error reinitialization enable	RW	0x0
0	sysref_lemc_err_en_reinit	SYSREF LEMC error reinitialization enable	RW	0x0

Table 76. rx_status0

Monitor ports of internal signals and counter which will be useful for debugging.

Offset: 0x80

Note: For bits that are compile-time specific, you must recompile to change the reset value.

Bit	Name	Description	Attribute	Reset
31:30	Reserved	Reserved	RV	0x0
29	sysref_det_pending	Indicate that SYSREF is yet to be detected. You need to set sysref_singledet to enable link initialization.	ROV	0x0
28	reinit_in_prog	Indicates that auto or manual link reinitialization is in progress.	ROV	0x0
27:19	rbd_count_early	• When rbd_count_early = 0, this indicates that the earliest lane arrives within the link at the LEMC boundary. • When rbd_count_early = 1, this indicates that the earliest lane arrives within the link at 1 link clock cycle after the LEMC boundary.	ROV	0x0
18:10	rbd_count	Legal value reported from this register is 0 to 512. When rbd_count = 0, this indicates that the latest lane arrives within the link at the LEMC boundary. When rbd_count = 1, this indicates that the latest lane arrives within the link at 1 link clock cycle after the LEMC boundary. *Note:* When the latest lane arrival in the link is too close to the LEMC boundary, Intel recommends you set the RBD release opportunity (rbd_offset) at least 2 link clocks away from rbd_count to accommodate for worst-case power cycle variation. Refer to Deterministic Latency on page 33 for more information.	ROV	0x0
9:2	lemc_period	Represent E: the number of multiblock in an extended multiblock	RO	Compile-time specific
1:0	sh_config	b00: CRC-12 b01: Standalone command channel	RO	Compile-time specific

continued...

Bit	Name	Description	Attribute	Reset
		b10: Reserved (CRC-3) b11: Reserved (FEC)		

Table 77. rx_status1

Monitor ports of internal signals and counter which will be useful for debugging.

Offset: 0x84

Bit	Name	Description	Attribute	Reset
31	lane15_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 15	ROV	0x0
30	lane14_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 14	ROV	0x0
29	lane13_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 13	ROV	0x0
28	lane12_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 12	ROV	0x0
27	lane11_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 11	ROV	0x0
26	lane10_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 10	ROV	0x0
25	lane9_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 9	ROV	0x0
24	lane8_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 8	ROV	0x0
23	lane7_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 7	ROV	0x0
22	lane6_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 6	ROV	0x0
21	lane5_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 5	ROV	0x0
20	lane4_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 4	ROV	0x0
19	lane3_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 3	ROV	0x0
18	lane2_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 2	ROV	0x0
17	lane1_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 1	ROV	0x0
16	lane0_rx_pcfifo_empty	RX phase compensation FIFO status empty flag for Lane 0	ROV	0x0
15	lane15_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 15	ROV	0x0
14	lane14_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 14	ROV	0x0

continued...

Bit	Name	Description	Attribute	Reset
13	lane13_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 13	ROV	0x0
12	lane12_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 12	ROV	0x0
11	lane11_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 11	ROV	0x0
10	lane10_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 10	ROV	0x0
9	lane9_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 9	ROV	0x0
8	lane8_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 8	ROV	0x0
7	lane7_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 7	ROV	0x0
6	lane6_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 6	ROV	0x0
5	lane5_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 5	ROV	0x0
4	lane4_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 4	ROV	0x0
3	lane3_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 3	ROV	0x0
2	lane2_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 2	ROV	0x0
1	lane1_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 1	ROV	0x0
0	lane0_rx_pcfifo_full	RX phase compensation FIFO status full flag for Lane 0	ROV	0x0

Table 78. rx_status2

Monitor ports of internal signals and counter which will be useful for debugging.

Offset: 0x88

Bit	Name	Description	Attribute	Reset
31	lane15_rx_cdr_locked	RX CDR lock status flag for Lane 15	ROV	0x0
30	lane14_rx_cdr_locked	RX CDR lock status flag for Lane 14	ROV	0x0
29	lane13_rx_cdr_locked	RX CDR lock status flag for Lane 13	ROV	0x0
28	lane12_rx_cdr_locked	RX CDR lock status flag for Lane 12	ROV	0x0
27	lane11_rx_cdr_locked	RX CDR lock status flag for Lane 11	ROV	0x0

continued...

Bit	Name	Description	Attribute	Reset
26	lane10_rx_cdr_locked	RX CDR lock status flag for Lane 10	ROV	0x0
25	lane9_rx_cdr_locked	RX CDR lock status flag for Lane 9	ROV	0x0
24	lane8_rx_cdr_locked	RX CDR lock status flag for Lane 8	ROV	0x0
23	lane7_rx_cdr_locked	RX CDR lock status flag for Lane 7	ROV	0x0
22	lane6_rx_cdr_locked	RX CDR lock status flag for Lane 6	ROV	0x0
21	lane5_rx_cdr_locked	RX CDR lock status flag for Lane 5	ROV	0x0
20	lane4_rx_cdr_locked	RX CDR lock status flag for Lane 4	ROV	0x0
19	lane3_rx_cdr_locked	RX CDR lock status flag for Lane 3	ROV	0x0
18	lane2_rx_cdr_locked	RX CDR lock status flag for Lane 2	ROV	0x0
17	lane1_rx_cdr_locked	RX CDR lock status flag for Lane 1	ROV	0x0
16	lane0_rx_cdr_locked	RX CDR lock status flag for Lane 0	ROV	0x0
15	lane15_rx_xcvr_ready	RX transceiver ready status flag for Lane 15	ROV	0x0
14	lane14_rx_xcvr_ready	RX transceiver ready status flag for Lane 14	ROV	0x0
13	lane13_rx_xcvr_ready	RX transceiver ready status flag for Lane 13	ROV	0x0
12	lane12_rx_xcvr_ready	RX transceiver ready status flag for Lane 12	ROV	0x0
11	lane11_rx_xcvr_ready	RX transceiver ready status flag for Lane 11	ROV	0x0
10	lane10_rx_xcvr_ready	RX transceiver ready status flag for Lane 10	ROV	0x0
9	lane9_rx_xcvr_ready	RX transceiver ready status flag for Lane 9	ROV	0x0
8	lane8_rx_xcvr_ready	RX transceiver ready status flag for Lane 8	ROV	0x0
7	lane7_rx_xcvr_ready	RX transceiver ready status flag for Lane 7	ROV	0x0
6	lane6_rx_xcvr_ready	RX transceiver ready status flag for Lane 6	ROV	0x0
5	lane5_rx_xcvr_ready	RX transceiver ready status flag for Lane 5	ROV	0x0

continued...

Bit	Name	Description	Attribute	Reset
4	lane4_rx_xcvr_ready	RX transceiver ready status flag for Lane 4	ROV	0x0
3	lane3_rx_xcvr_ready	RX transceiver ready status flag for Lane 3	ROV	0x0
2	lane2_rx_xcvr_ready	RX transceiver ready status flag for Lane 2	ROV	0x0
1	lane1_rx_xcvr_ready	RX transceiver ready status flag for Lane 1	ROV	0x0
0	lane0_rx_xcvr_ready	RX transceiver ready status flag for Lane 0	ROV	0x0

Table 79. rx_status3

Monitor ports of internal signals and counter which will be useful for debugging.

Offset: 0x8C

Bit	Name	Description	Attribute	Reset
31	lane15_rx_gb_empty	RX gearbox empty status flag for Lane 15	ROV	0x0
30	lane14_rx_gb_empty	RX gearbox empty status flag for Lane 14	ROV	0x0
29	lane13_rx_gb_empty	RX gearbox empty status flag for Lane 13	ROV	0x0
28	lane12_rx_gb_empty	RX gearbox empty status flag for Lane 12	ROV	0x0
27	lane11_rx_gb_empty	RX gearbox empty status flag for Lane 11	ROV	0x0
26	lane10_rx_gb_empty	RX gearbox empty status flag for Lane 10	ROV	0x0
25	lane9_rx_gb_empty	RX gearbox empty status flag for Lane 9	ROV	0x0
24	lane8_rx_gb_empty	RX gearbox empty status flag for Lane 8	ROV	0x0
23	lane7_rx_gb_empty	RX gearbox empty status flag for Lane 7	ROV	0x0
22	lane6_rx_gb_empty	RX gearbox empty status flag for Lane 6	ROV	0x0
21	lane5_rx_gb_empty	RX gearbox empty status flag for Lane 5	ROV	0x0
20	lane4_rx_gb_empty	RX gearbox empty status flag for Lane 4	ROV	0x0
19	lane3_rx_gb_empty	RX gearbox empty status flag for Lane 3	ROV	0x0
18	lane2_rx_gb_empty	RX gearbox empty status flag for Lane 2	ROV	0x0

continued...

Bit	Name	Description	Attribute	Reset
17	lane1_rx_gb_empty	RX gearbox empty status flag for Lane 1	ROV	0x0
16	lane0_rx_gb_empty	RX gearbox empty status flag for Lane 0	ROV	0x0
15	lane15_rx_gb_full	RX gearbox full status flag for Lane 15	ROV	0x0
14	lane14_rx_gb_full	RX gearbox full status flag for Lane 14	ROV	0x0
13	lane13_rx_gb_full	RX gearbox full status flag for Lane 13	ROV	0x0
12	lane12_rx_gb_full	RX gearbox full status flag for Lane 12	ROV	0x0
11	lane11_rx_gb_full	RX gearbox full status flag for Lane 11	ROV	0x0
10	lane10_rx_gb_full	RX gearbox full status flag for Lane 10	ROV	0x0
9	lane9_rx_gb_full	RX gearbox full status flag for Lane 9	ROV	0x0
8	lane8_rx_gb_full	RX gearbox full status flag for Lane 8	ROV	0x0
7	lane7_rx_gb_full	RX gearbox full status flag for Lane 7	ROV	0x0
6	lane6_rx_gb_full	RX gearbox full status flag for Lane 6	ROV	0x0
5	lane5_rx_gb_full	RX gearbox full status flag for Lane 5	ROV	0x0
4	lane4_rx_gb_full	RX gearbox full status flag for Lane 4	ROV	0x0
3	lane3_rx_gb_full	RX gearbox full status flag for Lane 3	ROV	0x0
2	lane2_rx_gb_full	RX gearbox full status flag for Lane 2	ROV	0x0
1	lane1_rx_gb_full	RX gearbox full status flag for Lane 1	ROV	0x0
0	lane0_rx_gb_full	RX gearbox full status flag for Lane 0	ROV	0x0

Table 80. rx_status4

Monitor ports of internal signals and counter which will be useful for debugging.

Offset: 0x90

Bit	Name	Description	Attribute	Reset
31:16	Reserved	Reserved	RV	0x0
15	lane15_sh_lock	RX sync header alignment lock status flag for Lane 15	ROV	0x0

continued...

Bit	Name	Description	Attribute	Reset
14	lane14_sh_lock	RX sync header alignment lock status flag for Lane 14	ROV	0x0
13	lane13_sh_lock	RX Sync Header alignment lock status flag for Lane 13	ROV	0x0
12	lane12_sh_lock	RX sync header alignment lock status flag for Lane 12	ROV	0x0
11	lane11_sh_lock	RX sync header alignment lock status flag for Lane 11	ROV	0x0
10	lane10_sh_lock	RX sync header alignment lock status flag for Lane 10	ROV	0x0
9	lane9_sh_lock	RX sync header alignment lock status flag for Lane 9	ROV	0x0
8	lane8_sh_lock	RX sync header alignment lock status flag for Lane 8	ROV	0x0
7	lane7_sh_lock	RX sync header alignment lock status flag for Lane 7	ROV	0x0
6	lane6_sh_lock	RX sync header alignment lock status flag for Lane 6	ROV	0x0
5	lane5_sh_lock	RX sync header alignment lock status flag for Lane 5	ROV	0x0
4	lane4_sh_lock	RX sync header alignment lock status flag for Lane 4	ROV	0x0
3	lane3_sh_lock	RX sync header alignment lock status flag for Lane 3	ROV	0x0
2	lane2_sh_lock	RX sync header alignment lock status flag for Lane 2	ROV	0x0
1	lane1_sh_lock	RX sync header alignment lock status flag for Lane 1	ROV	0x0
0	lane0_sh_lock	RX sync header alignment lock status flag for Lane 0	ROV	0x0

Table 81. rx_status5

Monitor ports of internal signals and counter which will be useful for debugging.

Offset: 0x94

Bit	Name	Description	Attribute	Reset
31:16	Reserved	Reserved	RV	0x0
15	lane15_emb_lock	RX EMB alignment lock status flag for Lane 15	ROV	0x0
14	lane14_emb_lock	RX EMB alignment lock status flag for Lane 14	ROV	0x0
13	lane13_emb_lock	RX EMB alignment lock status flag for Lane 13	ROV	0x0
12	lane12_emb_lock	RX EMB alignment lock status flag for Lane 12	ROV	0x0

continued...

Bit	Name	Description	Attribute	Reset
11	lane11_emb_lock	RX EMB alignment lock status flag for Lane 11	ROV	0x0
10	lane10_emb_lock	RX EMB alignment lock status flag for Lane 10	ROV	0x0
9	lane9_emb_lock	RX EMB alignment lock status flag for Lane 9	ROV	0x0
8	lane8_emb_lock	RX EMB alignment lock status flag for Lane 8	ROV	0x0
7	lane7_emb_lock	RX EMB alignment lock status flag for Lane 7	ROV	0x0
6	lane6_emb_lock	RX EMB alignment lock status flag for Lane 6	ROV	0x0
5	lane5_emb_lock	RX EMB alignment lock status flag for Lane 5	ROV	0x0
4	lane4_emb_lock	RX EMB alignment lock status flag for Lane 4	ROV	0x0
3	lane3_emb_lock	RX EMB alignment lock status flag for Lane 3	ROV	0x0
2	lane2_emb_lock	RX EMB alignment lock status flag for Lane 2	ROV	0x0
1	lane1_emb_lock	RX EMB alignment lock status flag for Lane 1	ROV	0x0
0	lane0_emb_lock	RX EMB alignment lock status flag for Lane 0	ROV	0x0

Table 82. rx_status6

Monitor ports of internal signals and counter which will be useful for debugging.

Offset: 0x98

Bit	Name	Description	Attribute	Reset
31:16	Reserved	Reserved	RV	0x0
15	lane15_rx_eb_full	RX Elastic buffer full status flag for Lane 15	ROV	0x0
14	lane14_rx_eb_full	RX Elastic buffer full status flag for Lane 14	ROV	0x0
13	lane13_rx_eb_full	RX Elastic buffer full status flag for Lane 13	ROV	0x0
12	lane12_rx_eb_full	RX Elastic buffer full status flag for Lane 12	ROV	0x0
11	lane11_rx_eb_full	RX Elastic buffer full status flag for Lane 11	ROV	0x0
10	lane10_rx_eb_full	RX Elastic buffer full status flag for Lane 10	ROV	0x0
9	lane9_rx_eb_full	RX Elastic buffer full status flag for Lane 9	ROV	0x0

continued...

Bit	Name	Description	Attribute	Reset
8	lane8_rx_eb_full	RX Elastic buffer full status flag for Lane 8	ROV	0x0
7	lane7_rx_eb_full	RX Elastic buffer full status flag for Lane 7	ROV	0x0
6	lane6_rx_eb_full	RX Elastic buffer full status flag for Lane 6	ROV	0x0
5	lane5_rx_eb_full	RX Elastic buffer full status flag for Lane 5	ROV	0x0
4	lane4_rx_eb_full	RX Elastic buffer full status flag for Lane 4	ROV	0x0
3	lane3_rx_eb_full	RX Elastic buffer full status flag for Lane 3	ROV	0x0
2	lane2_rx_eb_full	RX Elastic buffer full status flag for Lane 2	ROV	0x0
1	lane1_rx_eb_full	RX Elastic buffer full status flag for Lane 1	ROV	0x0
0	lane0_rx_eb_full	RX Elastic buffer full status flag for Lane 0	ROV	0x0

Table 83. rx_status7

Monitor ports of internal signals and counter which will be useful for debugging.

Offset: 0x9C

Bit	Name	Description	Attribute	Reset
31:16	Reserved	Reserved	RV	0x0
15	lane15_rx_polarity	RX polarity inversion status flag for Lane 15	ROV	0x0
14	lane14_rx_polarity	RX polarity inversion status flag for Lane 14	ROV	0x0
13	lane13_rx_polarity	RX polarity inversion status flag for Lane 13	ROV	0x0
12	lane12_rx_polarity	RX polarity inversion status flag for Lane 12	ROV	0x0
11	lane11_rx_polarity	RX polarity inversion status flag for Lane 11	ROV	0x0
10	lane10_rx_polarity	RX polarity inversion status flag for Lane 10	ROV	0x0
9	lane9_rx_polarity	RX polarity inversion status flag for Lane 9	ROV	0x0
8	lane8_rx_polarity	RX polarity inversion status flag for Lane 8	ROV	0x0
7	lane7_rx_polarity	RX polarity inversion status flag for Lane 7	ROV	0x0
6	lane6_rx_polarity	RX polarity inversion status flag for Lane 6	ROV	0x0

continued...

Bit	Name	Description	Attribute	Reset
5	lane5_rx_polarity	RX polarity inversion status flag for Lane 5	ROV	0x0
4	lane4_rx_polarity	RX polarity inversion status flag for Lane 4	ROV	0x0
3	lane3_rx_polarity	RX polarity inversion status flag for Lane 3	ROV	0x0
2	lane2_rx_polarity	RX polarity inversion status flag for Lane 2	ROV	0x0
1	lane1_rx_polarity	RX polarity inversion status flag for Lane 1	ROV	0x0
0	lane0_rx_polarity	RX polarity inversion status flag for Lane 0	ROV	0x0

Table 84. rx_converter_param1

Link and transport control configuration per converter parameters.

Offset: 0xC0

Note: For bits that are compile-time specific, you must recompile to change the reset value.

Bit	Name	Description	Attribute	Reset
31:30	CS	Number of control bits per converter sample. 1-based value. For example, 0=0 bit, 1=1 bit.	RO	Compile-time specific
29	HD	High Density format.	RO	Compile-time specific
28:24	N	Number of data bits per converter sample. 0-based value. For example, 0=0 bit, 1=2 bits. *Note:* CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7.	RO	Compile-time specific
23:16	M	Number of converters per device. 0-based value. For example, 0=1 converter, 1=2 converters. *Note:* CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7.	RO	Compile-time specific
15:8	F	Number of octets per frame per lane. 0-based value. For example, 0=1 octet, 1=2 octets. *Note:* CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7.	RO	Compile-time specific
7:4	Reserved	Reserved	RV	0x0
3:0	L	Number of lanes per link. 0-based value. For example, 0=1 lane, 1=2 lanes.	RO	Compile-time specific

continued...

Bit	Name	Description	Attribute	Reset
		Note: CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7.		

Table 85. rx_converter_param2

Link and Transport control configuration per converter parameters.

Offset: 0xC4

Note: For bits that are compile-time specific, you must recompile to change the reset value.

Bit	Name	Description	Attribute	Reset
31:24	E	Number of multiblock within an extended multiblock. 0-based value.For example, 0=1 multiblock to form extended multiblock, 1=2 multiblock to form an extended multiblock. If (256 Mod F) =1, E must be greater than 1. (The register value should be greater than 0). *Note:* CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7	RO	Compile-time specific
23:21	Reserved	Reserved	RV	0x0
20:16	CF	Number of control words per frame clock per link. 1-based value. For example, 0=0 word, 1=1 word.	RO	Compile-time specific
15:13	Reserved	Reserved	RO	0x0
12:8	S	Number of samples per converter frame cycle. 0-based value. For example, 0=1 sample, 1=2 samples. *Note:* CSR indexing is different from the parameter indexing. If parameter=`d8, this register field will be `d7	RO	Compile-time specific
7:5	subclass_ver	Device Subclass Version • b000: Subclass 0 • b001: Subclass 1	RO	Compile-time specific
4:0	NP	Number of data bits+control bits+tail bits per converter sample. 0-based value. For example, 0=1 bit, 1=2 bits.	RO	Compile-time specific

9. JESD204C Intel FPGA IP User Guide Archives

For the latest and previous versions of this user guide, refer to JESD204C Intel FPGA IP User Guide. If an IP or software version is not listed, the user guide for the previous IP or software version applies.

IP versions are the same as the Intel Quartus Prime Design Suite software versions up to v19.1. From Intel Quartus Prime Design Suite software version 19.2 or later, IP cores have a new IP versioning scheme.

10. Document Revision History for the JESD204C Intel FPGA IP User Guide

Document Version	Intel Quartus Prime Version	IP Version	Changes
2023.02.10	21.3	1.1.0	• Fixed the links in Table 1: *Related Documents*. • Updated the *JESD204C Intel FPGA IP User Guide Archives* section.
2021.10.22	21.3	1.1.0	• Updated the description for *Control and Status Registers* clarify that registers that are Read-Writable must be protected to comply with Security Development Lifecycle (SDL) practices. • Updated information about bit[31:16] in Table: *rx_status4*. • Updated Table: *Brief Information about the JESD204C Intel FPGA IP* to include support for QuestaSim simulator. • Updated the description for **Frame clock frequency multiplier (FCLK_MULP)** in Table: *JESD204C Intel FPGA IP Parameters*.
2021.06.16	21.2	1.1.0	• Removed NCSim support from the list of design tools in Table: *Brief Information about the JESD204C Intel FPGA IP*. • Updated the description for TX/RX device clock in Table: *JESD204C IP Clocks*. • Added information about the reference clocks of the transceiver and core PLLs in the *Device Clock* section.
2021.03.12	20.1	1.1.0	Updated Table: *rx_err_link_reinit* to correct the bit 0 name from `syncref_lemc_err_en_reinit` to `sysref_lemc_err_en_reinit`.
2020.10.05	20.1	1.1.0	Corrected the description for sampling SYSREF in the *Local Extended Multiblock Clock* section. The IP uses the link clock to sample SYSREF, not the frame clock.
2020.05.04	20.1	1.1.0	• Added *Device Family Support* section. • Updated the supported data rate for the different fabric speed grades for Intel Agilex and Intel Stratix 10 devices in the *Performance and Resource Utilization* section. • Added information about the new **Multilink mode** parameter in the *JESD204C Intel FPGA IP Parameters* section.

continued...

Document Version	Intel Quartus Prime Version	IP Version	Changes
			• Added a note for the **Frame data width multiplier (WIDTH_MULP)** parameter in the *JESD204C Intel FPGA IP Parameters* section. Select the smallest data width multiplier value on the list. Other data width multiplier values are not allowed. • Edited the range of values supported for the Control bits (CS) parameter. • Added information about the following two new signals for multilink mode in the *Receiver Signals* section: — `j204c_rx_dev_emblock_align` — `j204c_rx_alldev_emblock_align`
2019.12.16	19.4	1.1.0	• Updated the supported maximum data rate to 28.9 Gbps (for Intel Agilex devices) in the *Overview of the JESD204C Intel FPGA IP*, *JESD204C Intel FPGA IP Features*, and *Functional Description* sections. • Updated the resource utilization data for Intel Stratix 10 and Intel Agilex devices, and the supported maximum data rate to 28.9 Gbps for Intel Stratix 10 and Intel Agilex devices in the *Performance and Resource Utilization* section. • Updated the maximum data rate value option to 28.9 Gbps for the **Data Rate** parameter in the *JESD204C Intel FPGA IP Parameters* section. • Added a note to set specific data rates when the frame clock multiplier is 2 in the description for the **Frame clock frequency multiplier (FCLK_MULP)** parameter in the *JESD204C Intel FPGA IP Parameters* section. • Added a note to insert 2 pipeline stages for high data rates in the description for the parameter in the *JESD204C Intel FPGA IP Parameters* section. • Removed the *Validated VariantsEnable TX pipestage (Transmitter)* table.
2019.10.23	19.3	1.0.0	• Added advance support for Intel Agilex devices. • Updated the related document links and the acronyms, glossary, and symbols lists in the *About the JESD204C Intel FPGA IPEnable TX pipestage User Guide* section. • Updated the *JESD204C Intel FPGA IP Features* section with maximum data rate information for Intel Agilex devices. • Updated the *Performance and Resource Utilization* section with Intel Agilex devices information. • Edited the maximum SYSREF frequency calculation in the *LEMC Counter* section for clarity.
2019.07.05	19.2	1.0.0	Initial release.