

Quickly Implement JESD204B on a Xilinx FPGA

By Haijiao Fan

Introduction

[JESD204](#) is a high-speed serial interface for connecting data converters (ADCs and DACs) to logic devices. Revision B of the standard supports serial data rates up to 12.5 Gbps and ensures repeatable, deterministic latency on the JESD204 link. As the speed and resolution of converters continues to increase, the JESD204B interface has become ever more common in ADI's high-speed converters and integrated RF transceivers. In addition, flexible serializer/deserializer ([SERDES](#)) designs in FPGAs and ASICs have naturally started to replace the traditional parallel LVDS/CMOS interface to converters, and are used to implement the JESD204B physical layer. This article describes how to quickly set up a project using a Xilinx® FPGA to implement the JESD204B interface, and provides some application and debug suggestions for FPGA designers.

JESD204B Protocol Implementation Overview

The JESD204B specification defines four key layers that implement the protocol data stream, as shown in Figure 1. The transport layer maps the conversion between samples and framed, unscrambled octets. The optional scrambling layer scrambles/descrambles the octets, spreading the spectral peaks to reduce EMI. The data-link layer handles link

synchronization, setup, and maintenance, and encodes/decodes the optionally scrambled octets to/from 10-bit characters. The physical layer is responsible for transmission and reception of characters at the bit rate.

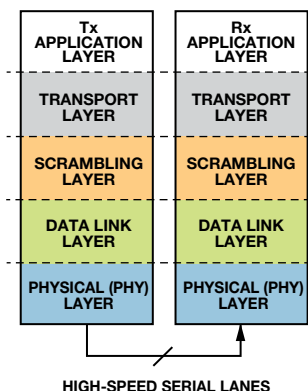


Figure 1. Key layers of JESD204B standard.

Different JESD204B IP vendors may implement the layers in different ways. Figure 2 and Figure 3 illustrate how the JESD204B transmit and receive protocols are implemented by ADI.

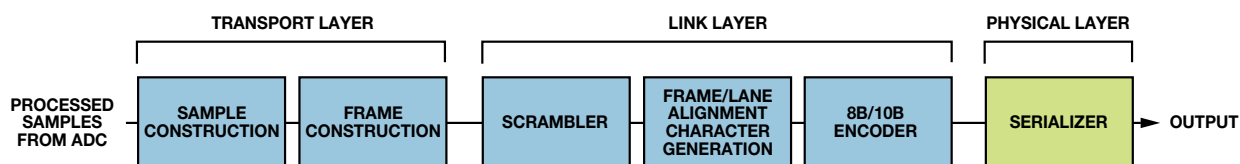


Figure 2. JESD204B transmitter implementation.

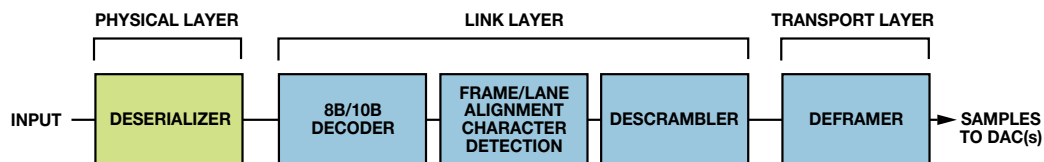


Figure 3. JESD204B receiver implementation.

The transport layer implementation depends strongly on the specific converter’s configuration and how it maps between samples and frames, so most FPGA vendors exclude it from their JESD204 IP. In addition, highly configurable, tightly integrated SERDES transceivers are integrated in FPGAs. These can be used to support all kinds of serial protocols, including PCIe, SATA, SRIO, CPRI, and JESD204B. Thus, a logic core that implements the link layer, combined with a configurable SERDES that realizes the physical layer, forms the basis for a JESD204B link. Figure 4 and Figure 5 show block diagrams of a JESD204B transmitter and receiver on a Xilinx FPGA. The transmitter/receiver lanes implement the scramble and link layers; the 8B/10B encoder/decoder and the physical layer are implemented in the GTX/GTX/GTH gigabit transceivers.

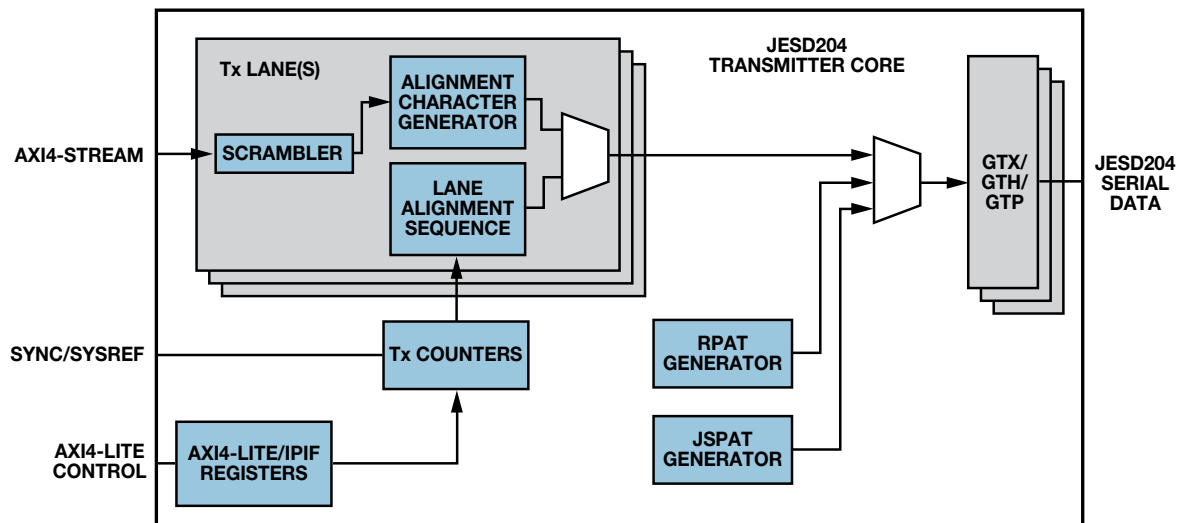


Figure 4. JESD204B transmitter implementation using a Xilinx FPGA.

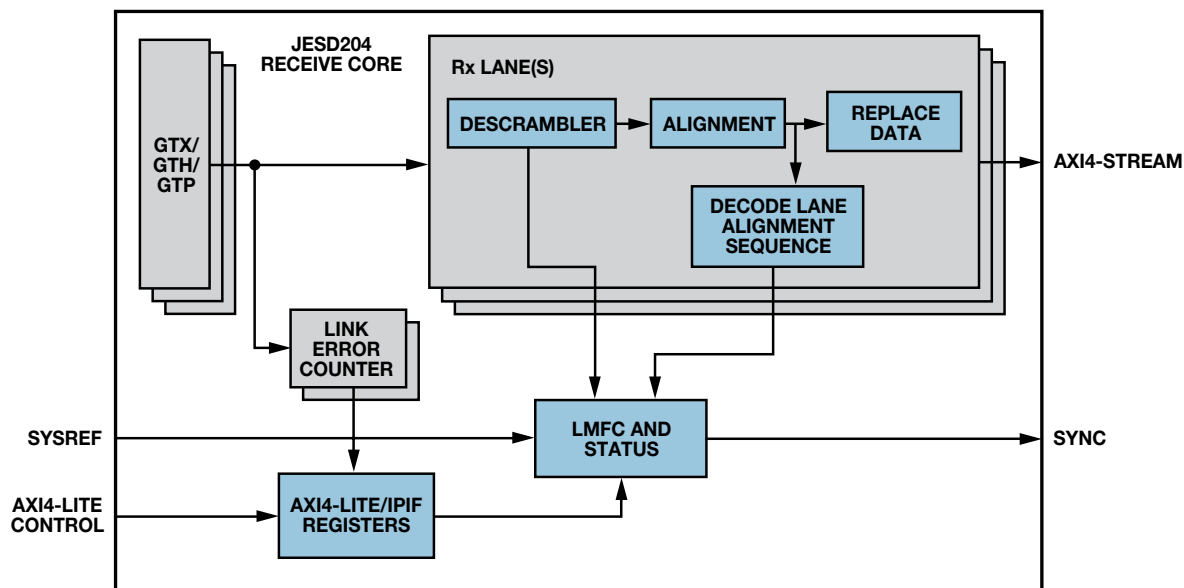


Figure 5. JESD204B receiver implementation using a Xilinx FPGA.

JESD204B Design Example Using a Xilinx FPGA

The latest [Xilinx JESD204 IP](#) core is delivered and encrypted as a black box via the [Vivado® Design Suite](#). Xilinx also provides a Verilog example design using the [Advanced eXtensible Interface](#) (AXI), but this example project is overdesigned for most applications. Users typically have their own configuration interfaces and do not need to integrate an extra AXI for JESD204B logic. Figure 6 shows a simplified JESD204 design, which is intended to help FPGA users understand the structure of JESD204 and quickly start their own FPGA-based JESD204 project.

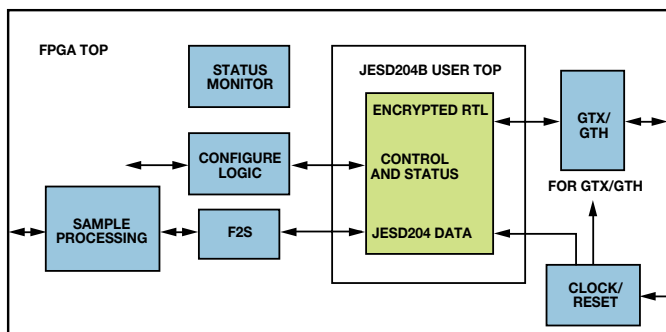


Figure 6. JESD204B design example.

The encrypted [register-transfer language](#) (RTL) block—the JESD204 logic IP core generated by Vivado—is equivalent to the transmitter and receiver modules shown in Figure 4 and Figure 5. The encrypted interface definition can be found in the Xilinx example design files. Then the encrypted RTL is wrapped into the JESD204B user top. Control, configuration, status, and JESD data interfaces from the encrypted RTL go through the wrapper to connect with user logic and the GTX/GTH transceiver data. Symbol alignment configuration for GTX/GTH is optimized and updated to make the transceivers work more robustly.

Dedicated pads should be used for the GTX/GTH reference clock to the SERDES transceivers. Careful attention must be paid to the global clock design for the FPGA logic, including clocks for the internal PLL, parallel interface, JESD204 logic core, and user-specific processing logic. In addition, the master system reference (SYSREF) input for the JESD204B logic core (Subclass 1) must be captured accurately to guarantee the deterministic delay of the JESD204 link.

The reset sequence for the GTX/GTH transceivers and the JESD204 core is crucial for reliable JESD link initialization, so the JESD204 core should be in reset state until the internal PLL in the GTX/GTH transceiver is locked and the GTX/GTH has been reset.

A frame-to-samples (F2S) module is needed to implement the transport layer of JESD204, which maps samples to or from frames according to the specific JESD204B configuration. The samples are then processed by application-specific logic. An auxiliary module monitors the JESD204 logic and physical layer (PHY) status for system debug.

Symbol Alignment in the Xilinx SERDES Transceiver

In the SERDES receiver, serial data must be aligned to symbol boundaries before it can be used as parallel data. To align the data, the transmitter sends a recognizable sequence, usually called a comma. The receiver searches for a comma in the incoming serial data stream and moves it to a symbol boundary once found. This enables the received parallel words to match the transmitted parallel words. The comma is usually a K, which is a special character in 8B/10B table used for [control symbols](#). For JESD204B applications, the transmitter will send a stream of $K = K28.5$ symbols for code group synchronization (CGS). The FPGA can therefore use $K28.5$ as a comma to align symbol boundaries and users can specify whether a comma match consists of either a comma plus ([running disparity](#) is plus) or a comma minus (running disparity is minus), or both. The JESD204B default setting for GTX/GTH comma detection allows either a comma plus or a comma minus to align the comma.

In some applications, the default comma setting may result in symbol realignment, or alignment to the wrong symbol boundary. This can cause messy 8B/10B decoding errors and broken JESD204B links. Combined comma plus and minus is more robust, forcing the comma align block to search for two commas in a row, detecting a comma only when the received data has a comma plus or minus followed by a comma minus or plus with no extra bits in between. This helps to maintain symbol boundaries and link stability when the line rate is high or the system has excessive noise.

Design Consideration for JESD204 Projects on FPGA

A synchronous, active low SYNC signal from the JESD204 receiver to the transmitter indicates the state of synchronization. Link reinitialization during normal operation will cause messy samples data, so the link status must be monitored in real time. In particular, a continuous low on SYNC means the receiver cannot identify at least four consecutive $K28.5$ symbols in the received data stream. If this occurs, check the transmitter/receiver SERDES configuration or make sure that the transmitter is sending $K28.5$. A continuous high on SYNC means the link has established and maintained stability. When SYNC goes from high to low and back to high, the duration of the low state should be counted. If it is longer than five frames plus nine octets, the receiver has detected a large error and sends a request to reinitialize the JESD204 link. If the duration is equal to two frame clocks, the receiver has detected a small error, but does not trigger a link reinitialization. This function can significantly ease system debug and further link monitoring, so users should include it in their designs.

8B/10B decoding errors can lead to JESD204B link reinitialization, but they are not the only cause, so user designs should have the ability to count decoding errors of each lane to determine the cause of link resynchronization. Also, the SERDES link quality can be determined in real time by the 8B/10B decoding error status.

[Pseudorandom bit sequences](#) (PRBS) provide a useful resource for measuring signal quality and jitter tolerance in high-speed links. The SERDES transceiver in most FPGAs has a built-in PRBS generator and checker, so no extra FPGA resources are needed. Thus, do not forget to instantiate this function, which should be used when the bit-error rate (BER) or eye diagram is evaluated.

A buffer is always used in SERDES transceivers to change the internal clock domain. A bad clock design for the transmitter and receiver or the wrong clock and data recovery (CDR) setting can cause buffer overflow or underflow. Some link errors may occur in this case, so monitoring the buffer status is meaningful. An interrupt record for buffer overflow and underflow is useful for system debug, so other internal buffers that are not allowed to underflow or overflow in user logic should also be monitored.

Conclusion

This article showed how to quickly implement a JESD204 block on a Xilinx FPGA, but the method can be applied to other FPGAs as well. First, understand the function and interfaces of the JESD204 logic core and transceiver provided by the FPGA vendor, then instantiate them and wrap them into your logic. Second, globally design the FPGA clock tree and reset sequence for your entire project. Third, carefully define the interfaces between the JESD204 logic core, user logic, and transceivers. Finally, add necessary debug resources. Following these steps will help you to achieve a quick and successful design for your JESD204 interface.

References

[JESD204B Survival Guide](#)

[JESD204 Serial Interface JEDEC Standard for Data Converters](#)

[High-Speed Analog-to-Digital Converters](#)

[High-Speed Digital-to-Analog Converters](#)

[Integrated Transceivers, Transmitters, and Receivers](#)

[Alliances and FPGA Reference Designs](#)

[Demystifying the JESD204B High-Speed Data Converter-to-FPGA Interface](#)

Beavers, Ian. [“Demystifying Deterministic Latency Within JESD204B Converters.”](#) *Electronic Design*, 2/25/2014.

Beavers, Ian. [“Prototyping Systems: JESD204B Converters and FPGAs.”](#) *Electronic Design*, 1/23/2014.

Beavers, Ian and Jeffrey Ugalde. [“Designing JESD204B Converter Systems for Low BER, Part 1.”](#) *EDN*, 10/22/2014.

Beavers, Ian and Jeffrey Ugalde. [“Designing JESD204B Converter Systems for Low BER, Part 2.”](#) *EDN*, 10/28/2014.

Harris, Jonathan. [“Understanding Layers in the JESD204B Specification: A High-Speed ADC Perspective, Part 1.”](#) *EDN*, 9/24/2014.

Harris, Jonathan. [“Understanding Layers in the JESD204B Specification: A High-Speed ADC Perspective, Part 2.”](#) *EDN*, 10/2/2014.

Jones, Del. [“JESD204B Subclasses \(Part 1\): Intro and Deterministic Latency.”](#) *EDN*, 6/18/2014.

Jones, Del. [“JESD204B Subclasses \(Part 2\): Subclass 1 vs. 2, System Considerations.”](#) *EDN*, 6/25/2014.



Haijiao Fan [haijiao.fan@analog.com] is an applications engineer at Analog Devices in Beijing, China, focusing on JESD204 protocol evaluation and integrated RF transceivers application and support. He received his B.S.E.E. in 2003 and M.S.E.E. in 2006 from Northwestern Polytechnical University, China. Prior to joining ADI in July 2012, Haijiao worked as an FPGA and system engineer for over six years.



Haijiao Fan