A JESD204B-Compliant Architecture for Remote and Deterministic-Latency Operation

Raffaele Giordano, Vincenzo Izzo, Sabrina Perrella, and Alberto Aloisio

***Abstract*— High-speed analog-to-digital converters (ADCs) are key components in a huge variety of systems, including trigger and data acquisition (TDAQ) systems of nuclear and subnuclear physics experiments. Over the last decades, the sample rate and dynamic range of high-speed ADCs underwent a continuous growth, and it required the development of suitable interface protocols, such as the new JESD204B serial interface protocol. In this paper, we present an original JESD204B-compliant architecture we designed, which is able to operate an ADC in a remote fashion. Our design includes a deterministic-latency high-speed serial link, which is the only connection between the local and remote logic of the architecture and which preserves the deterministic timing features of the protocol. By means of our solution, it is possible to read data out of several converters, even remote to each other, and keep them operating synchronously. Our link also supports forward error correction (FEC) capabilities, in the view of the operation in radiation areas (e.g., on-detector in TDAQ systems). We describe an implementation of our concept in a latest generation field programmable gate array (Xilinx Kintex-7 325T) for reading data from a high-speed JESD204B-compliant ADC. We present measurements of the jitter of JESD204B timing-critical signals forwarded over the link and of latency determinism of the FEC-protected link.**

***Index Terms*— Analog-to-digital converter (ADC), application-specific integrated circuit (ASIC), field programmable gate array (FPGA), JESD204B.**

I. Introduction

HIGH-SPEED analog-to-digital converters (ADCs) are key components in a huge variety of systems, such as wireless infrastructure transceivers, software defined radios, radar, secure communications, medical imaging systems, and trigger and data acquisition (TDAQ) systems of nuclear and subnuclear physics experiments. In fact, the usage of high-speed ADCs for digitizing analog pulses produced by the front-end electronics [1]–[4] opens the way to a fully digital processing, which can be implemented by means of application-specific integrated circuits (ASICs) or field programmable gate arrays (FPGAs). The Joint Electron Device Engineering Council has proposed a new, serial interface protocol (JESD204B [5]) optimized for analog-to-digital and digital-to-analog converters. The JESD204B standard supports data rates of up to 12.5 Gb/s per serial lane and foresees dedicated features to guarantee a deterministic timing of the conversion and to support the synchronization of multiple converters in the same system. The portfolio of JESD204B-compliant converters is in constant growth and FPGA vendors and third parties offer intellectual properties for handling the protocol complexity [6]–[9]. The timing predictability of the protocol is of great interest for TDAQ systems, where it is often required to operate the whole apparatus synchronously in order to preserve critical trigger information and timing-related data. It is important to note that the JESD204B standard is designed for local operation, i.e., the data producer and consumer chips are meant to be on the same board or anyway at the distances of the order of few centimeters, while TDAQ systems may require the converter to be remote (e.g., on-detector) with respect to the logic receiving the data (e.g., off-detector).

In this paper, we present an original JESD204B-compliant architecture we designed, which is able to operate an ADC in a remote fashion. Our design includes a deterministic-latency high-speed serial link, which is the only connection between the local and remote logic of the architecture, and preserves the deterministic timing features of the protocol. By means of our solution, it is possible to read data out of several converters, even remote to each other, and keep them operating synchronously. Our link also supports forward error correction (FEC) capabilities, in the view of the operation in radiation areas (e.g., on-detector in TDAQ systems).

This paper is organized as follows. In Section II, we briefly summarize the aspects of the JESD204B protocol relevant to this paper. In Section III, we discuss the link architecture we designed. In Section IV, we focus on an implementation of our concept in a Xilinx Kintex-7 FPGA and a demonstrator built around a JESD204B-compliant high-speed Texas Instruments ADC. In Section V, we present measurements of the timing jitter of JESD204B timing-critical signals forwarded over the link and of latency determinism of the FEC-protected link.

II. JESD204B Protocol

The JESD204 standard has been released in 2006 with support for a single serial lane at data rates up to 3.125 Gb/s. The standard has been revised in 2008 (JESD204A) adding the support for multiple lanes and further in 2011 (JESD204B) increasing the data rates up to 12.5 Gb/s and introducing support for deterministic latency in data transfers.

Manuscript received July 11, 2016; revised November 16, 2016; accepted January 15, 2017. Date of publication January 20, 2017; date of current version June 26, 2017. Accesso Aperto MIUR. This work was part of the ROAL Project (Grant RBSI14JOUV) supported by the 2014 Scientific Independence of Young Researchers (SIR) program of the Italian Ministry of Education, University and Research (MIUR). *(Corresponding author: Raffaele Giordano.)*

R. Giordano, S. Perrella, and A. Aloisio are with Università degli Studi di Napoli Federico II, I-80126 Napoli, Italy, and also with INFN Sezione di Napoli, I-80126 Napoli, Italy (e-mail: rgiordano@na.infn.it).

V. Izzo is with INFN Sezione di Napoli, I-80126 Napoli, Italy.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNS.2017.2655569

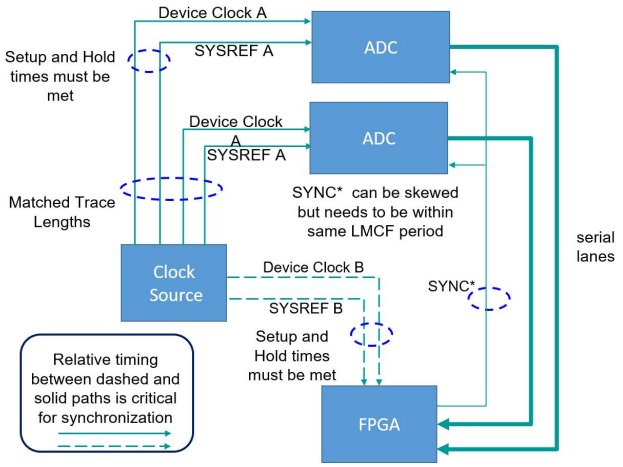

Fig. 1. Simplified block diagram of the JESD204B Subclass 1 synchronization scheme.

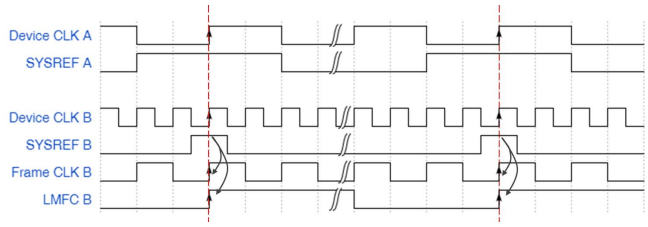

Fig. 2. Timing diagram of SYSREF signals pertaining to two device clocks at different frequencies.

The JESD204B standard includes three subclasses (0, 1, and 2). The architecture discussed in this paper supports JESD204B subclass 1, which achieves deterministic latency on serial lanes by means of auxiliary LVDS signals. Let us consider a simple JESD204B subclass 1 including two transmitters (e.g., ADCs) and one receiver (e.g., FPGA) (Fig. 1). In a JESD204B-compliant device, conversion data are sequenced in words called frames, whose bit size is device-dependent (e.g., a 16-b ADC might output 16- or 8-b frames). A sequence of K consecutive frames is called a multiframe. The number of frames per multiframe is usually configurable. A dedicated source distributes clocks to transmitters and receivers plus the SYSREF signal, which tags a specific edge (Fig. 2). The marked edge is used to generate the so-called frame clock and local multiframe clock (LMFC) in each JESD204B device. The JESD204B protocol is based on the 8b10b line coding, and it uses five control characters (/K/, /F/, /A/, /R/, and /Q/) for various functions in data links (Table I). The SYNC* signal is driven by receivers, and it flags to transmitters to send special 8b10b alignment symbols (/K/=K28.5). The /K/ symbols are then used by the receiver in order to correctly determine the byte boundary in the serial stream.

In order to establish a JESD204B link and start transferring user data, the transmitter and receiver devices have to go through two preliminary states, the code group synchronization (CGS) and the initial lane alignment sequence (ILAS) (Fig. 3). During CGS, the receiver asserts SYNC*, it waits for four subsequent /K/ symbols, it finds the correct byte boundary

TABLE I
MAIN 8B10B CONTROL CHARACTERS USED IN THE JESD204B PROTOCOL

Abbreviation	8b10b symbol	8 bit value	10 bit value (RD=-1)	10 bit value (RD=+1)	JESD204B function
/R/	K28.0	000 11100	001111 0100	110000 1011	Begin of multi-frame
/A/	K28.3	011 11100	001111 0011	110000 1100	Lane alignment
/Q/	K28.4	100 11100	001111 0010	110000 1101	Start of link configuration data
/K/	K28.5	101 11100	001111 1010	110000 0101	Groups synchronization
/F/	K28.7	111 11100	001111 1000	110000 0111	Frame alignment

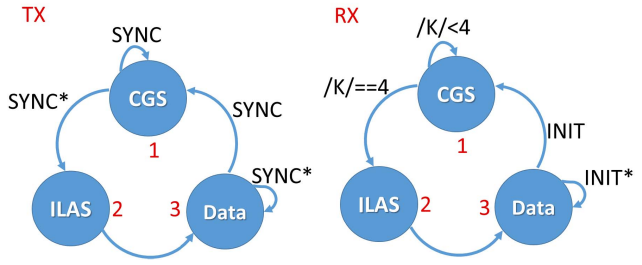

Fig. 3. JESD204B state diagrams for the transmitter (left) and for the receiver (right).

in the serial stream, it deasserts SYNC*, and it enters the ILAS state. The deassertion of SYNC* also triggers the transmitter to enter the ILAS state at the next LMFC cycle. In this state, the transmitter sends a sequence of frames containing link parameters (number of lanes, number of frames per multiframe packet, and so on). The sequence always begins with a K28.0 character (/R/), it terminates with a K28.3 character (/A/), and it consists of four multiframes. While receiving the alignment sequence, the receiver stores it into a first-in–first-out (FIFO) buffer until all the lanes have received the last frame. A timing skew between the lanes is absorbed by FIFOs on each lane and data are read out from all the FIFOs at the same LMFC edge. This in turn relaxes the requirement of balanced-delay routes for the different serial lanes of the receiver. After the ILAS, the transmitter starts transferring user data (e.g., ADC samples). It is important to note that, if the least significant octet of the nth frame (D_n) is unchanged with respect to the previous clock cycles (e.g., $D_n = D_{n-1}$), the transmitter replaces it with an 8b10b alignment character (/A/ if D_n is in the last frame of a multiframe and /F/ = K28.7 otherwise). When an /F/ (or /A/) character is decoded by the receiver, it checks that the character is at the end of a frame (or a multiframe) and replaces it with the corresponding octet from the previous frame (D_{n-1}). On the other hand, an /F/ (or /A/) character received at a position different than the end of a frame (or a multiframe) allows the receiver to detect alignment errors and to request a realignment procedure to the transmitter by asserting the SYNC* signal.

Unfortunately, the distribution of clocks and SYSREF signals from a common clock source limits the application of JESD204B links to distances of few tens of centimeters, e.g., it

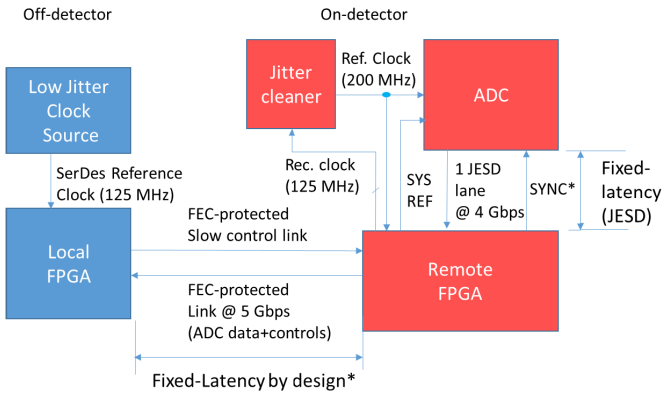

Fig. 4. Simplified block diagram of the remote JESD204B-compliant architecture.

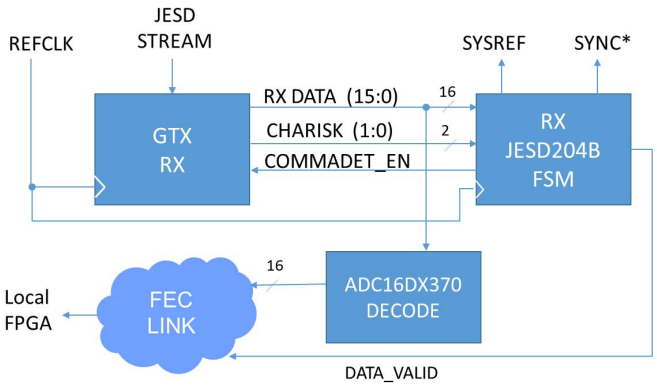

Fig. 5. Simplified block diagram of the developed JESD204B interface.

is expected that the transmitter and receiver are local to each other, ideally on the same board.

In Section III, we introduce the architecture we developed for reading out a JESD204B-compliant ADC remotely.

III. Remote JESD204B-Compliant Read-Out Architecture

We have designed our architecture (Fig. 4) in the view of applications in trigger and data acquisition systems of high-energy physics experiments. It consists of a remote part, which includes the ADC and it is supposed to be installed on detector, and a local part, which is supposed to be off-detector, e.g., in a counting room. Both parts are built around an FPGA, and a bidirectional serial link running at 5 Gb/s connects the two FPGAs. The link is protected by means of FEC in order to minimize possible transmission errors related to radiation on detector. The local FPGA uses the link to send slow control commands to the remote FPGA, which forwards the clock recovered from the serial stream to provide a reference to a jitter cleaner. The cleaner generates two clean clocks at 200 MHz and distributes them as JESD204B device clocks to the ADC and back to the remote FPGA. The remote FPGA implements a simple JESD204B interface (a single lane at 4 Gb/s), extracts ADC samples, and serializes them over the FEC-protected link toward the local FPGA, which deserializes the data and further processes them. The latency of the data transfer between the ADC and the remote FPGA is kept fixed thanks to the JESD204B protocol, while the FEC-protected link implements a fixed-latency architecture. Therefore, the whole data path operates with fixed latency.

In Sections III-A and III-B, we describe in more detail the JESD204B interface and the FEC-protected link.

A. JESD204B Interface

Our JESD204B interface (Fig. 5) consists of a high-speed deserializer of the FPGA device [GTX receiver (GTX-RX)], a finite-state machine according to the JESD204B receiver state diagram, and a data decoder. The GTX-RX deserializes and decodes the 8b10b-encoded serial stream into 16-b frames [RXDATA(15:0)] plus two flag bits [CHARISK(1:0)] clocked at 200 MHz. The CHARISK(0/1) bit flags whether the least/most significant octet in the RXDATA bus is a data or control character. The FSM generates the SYSREF for the ADC and it monitors the RXDATA and CHARISK busses. The comma detection and alignment circuit in the GTX-RX is configured to detect and align on the K28.5 symbol, which contains an 8b10b comma sequence, and it is transmitted by the ADC during CGS. The FSM disables the comma detector after CGS, in order to prevent fake realignments of the link on other commas received out of place. After completing the ILA phase, the FSM starts performing character replacement and asserts the DATA_VALID signal. Since 16-b ADC samples are divided into two octets and they are encoded by the ADC prior to JESD transmission, a dedicated logic decodes the data to obtain samples, which are then routed together with the DATA_VALID signal to the FEC-protected link.

B. FEC-Protected Link

The line code proposed for the FEC-protected link is designed in order to provide the following:

1) a dc-balanced serial stream for transmission on ac-coupled physical layers;
2) low-coding overhead (<40%);
3) an error correcting code for mitigating radiation single event effects;
4) low-latency (<20 ns) coding and decoding for applications in trigger systems.

The first two points are achieved by means of a scrambler, which requires no extra bandwidth, and it is implemented as a linear feedback shift register (LFSR) [10]. Error correction is performed by means of a Reed–Solomon (RS) code [11], as this family of codes is capable to correct burst errors, which might occur due to radiation-induced failures of opto-electronics. RS encoding and decoding latency depends on the code complexity [12], which in turn depends on its correction capability [13]. In order to minimize the link latency and the complexity of the RS decoder, we chose to limit the correction capability to one symbol per RS word.

The components of the transmitter and the receiver of the FEC link are shown in Figs. 6 and 7, respectively. The functional blocks of the transmitter are a 16-b scrambler followed

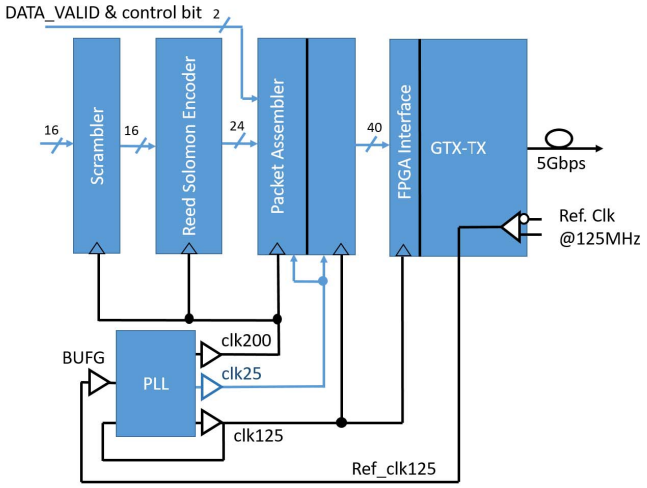

Fig. 6. Simplified block diagram of the FEC-protected link transmitter at the local FPGA.

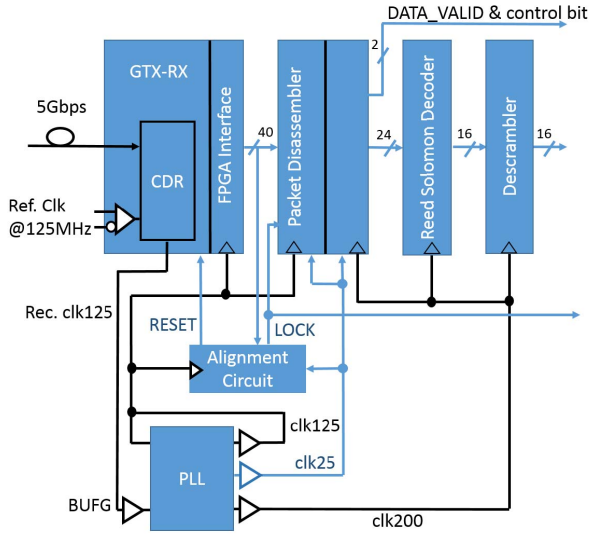

Fig. 7. Simplified block diagram of the FEC-protected link receiver.

by a shortened RS(6,4) [14] encoder, a packet assembler, and a GTX transmitter (GTX-TX). At the receiver, the inverse operations are performed by a GTX-RX, a packet disassembler, a shortened RS(6,4) decoder, and a 16-b descrambler. In addition, a packet alignment circuit is used in order to correctly find and align to the packet boundary in the serial stream. At the transmitter (receiver) end of the link, a PLL generates a 200-MHz clock for scrambling (descrambling) and RS coding (decoding) and a 125-MHz clock for synchronizing data into (out from) the GTX. Both at the local FEC-link transmitter and receiver, the 200- and 125-MHz clocks are derived from the same 125-MHz clock and they align their edges every 40 ns (eight clock cycles at 200 MHz or five at 125 MHz), which is the amount of time needed for assembling (or disassembling) a packet. The PLLs generate an additional 25-MHz clock, with a fixed-phase relationship to the common edge of the 125- and 200-MHz clocks but with an appropriate phase to be captured as data. A simple logic in the packet assembler (disassambler) detects the rising

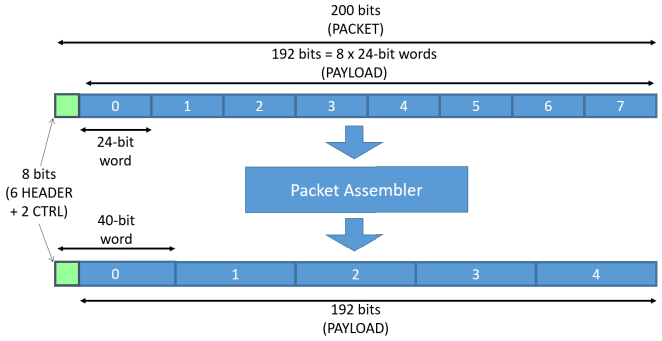

Fig. 8. Structure of a packet.

edge of the 25-MHz clock and generates two 25-MHz marker signals. One of the markers tags the edge in the 125-MHz clock domain, while the other tags it in the 200-MHz clock domain. The packet assembler (or disassembler) synchronizes internal operations to the markers. In fact, they are produced synchronously to the packet least significant word, which at the remote receiver is recognized by the alignment logic. The latency for a full packet to cross the packet assembler (or disassembler) is therefore constant at 40 ns.

At the transmitter side, the 16-b ADC words are transferred through the scrambler. The feedback polynomial $(x^{16}+x^{14}+x^{13}+x+1)$ chosen for the scrambler corresponds to a maximal-length LFSR. We have chosen a parallel architecture for the scrambler in order to reduce the latency of the block and a 16-b word size in order to match the ADC data width. The RS encoder is based on the x^5+x+1 generator polynomial. It processes 16-b words as four 4-b symbols and it adds eight parity bits, i.e., two 4-b parity symbols. Therefore, during decoding, it is possible to identify and correct up to four consecutive incorrect bits (a 4-b symbol) in each 16-b word, resulting in an overall error correction capability of 25%. The 24-b words generated by the RS encoder do not fit the GTX parallel input port width. Moreover, the addition of a header is required for aligning to the packet boundary. For these reasons, we decided to assemble 24-b words into packets and mux them into a sequence of 40-b words, which is instead a supported word size. A packet (Fig. 8) consists of 200 b, i.e., five 40-b words or eight 24-b plus a 6-b, dc-balanced header ("101010"), the DATA_VALID flag, and an optional control bit. Once the packet is assembled, 40-b words are clocked at 125 MHz into the GTX-TX, which serializes them at 5.0 Gb/s.

We remark that the clocking scheme of the transmitter is slightly different for the remote and local nodes. In fact, at the local node, the reference clock for the GTX-TX is generated by a local oscillator at 125 MHz, while at the remote node, the reference is a 200-MHz clock provided by the JESD204B-compliant jitter cleaner. The input for the jitter cleaner is the 125-MHz GTX-RX recovered clock.

The serial stream is deserialized by the GTX-RX, which runs with the elastic buffer active for resolving internal clock domain phase differences. For the RS decoder and descrambler to operate properly, the packet boundary in the serial stream has to be determined. In order to stabilize the latency through the receiver, we disabled the GTX-RX internal "comma detect

TABLE II
LATENCY OF THE LINK SUBBLOCKS

	# of clock cycles	Clock Period (ns)	Block Latency (ns)
Transmitter			
Scrambler	1	5	5
RS Encoder	1	5	5
Packet Generator	8	5	40
GTX-TX	5.5	8	44
Total Transmitter			94
Receiver			
GTX-RX	17.5	8	140
Packet Decoder	5	8	40
RS Decoder	4	5	20
Descrambler	2	5	10
Total Receiver			210
Total Tx+Rx			304

TABLE III
LATENCY OF THE GTX-TX AND GTX-RX SUBCOMPONENTS IN THE ADOPTED CONFIGURATION

	# of clock cycles	Clock Period (ns)	Block Latency (ns)
GTX-TX			
FPGA Interface	2	8	16
Elastic Buffer (skipped)	1	8	8
PMA Interface	1	8	8
PMA Latency	61.5	0.2	12.3
Total GTX-TX			44.3
GTX-RX			
PMA Latency	178.5	0.2	35.7
Elastic Buffer (used)	10	8	80
Comma Detector (skipped)	1	8	8
FPGA Interface	2	8	16
Total GTX-RX			139.7

TABLE IV
RESOURCE OCCUPATION OF THE SINGLE-LANE JESD204B INTERFACE

Logic Resources	Used/Available	Percentage
FFs	3,901/407,600	1%
LUTs	3,263/203,800	2%
BUFGs	1/32	3%
BUFGDS_GTEs	1/8	12%
GTXs	1/16	6%

TABLE V
RESOURCE OCCUPATION OF THE FEC-PROTECTED LINK (TRANSMITTER PLUS RECEIVER)

Logic Resources	Used/Available	Percentage
FFs	1,799/407,600	1%
LUTs	5,082/203,800	2%
BUFGs	7/32	22%
BUFGDS_GTEs	1/8	12%
GTXs	1/16	6%

and alignment" circuit, and we implemented an alternative alignment circuit in the fabric based on the pure roulette approach as described in [15] and [16]. In fact, the logic resets the GTX-RX, until it finds the header aligned to the least significant bit of the packet and the GTX-RX latency remains constant after a power cycle or a reset of the link. No clock or data shifting is performed at the GTX level.

The alignment logic requires the identification of ten valid headers out of the last 24 before asserting the lock flag. Due to the action of the scrambler, it is very unlikely for the scrambled data to be equal to header bits a repeated number of times. On the other hand, four invalid headers out of 24 are interpreted as a loss of lock and cause the deassertion of the lock flag. After the alignment procedure has been completed, the packet disassembler is enabled. This circuit receives 40-b words at 125 MHz from the GTX-RX and it outputs RS-coded 24-b words at 200 MHz to the RS decoder plus the DATA_VALID flag and control bit. The last block of the FEC link is the descrambler, which returns 16-b ADC words. The descrambler architecture is symmetrical to the scrambler's one, and it implements the same LFSR with a reversed bit order. In order to reconstruct the original bit-stream, the registers in the scrambler and descrambler have to be synchronized. In our scheme, we used a self-synchronized scrambling; therefore, no dedicated synchronization header is necessary. The descrambler automatically synchronizes after one clock cycle, needed for the received data to fill the register in the LFSR. Table II reports the latency for each block of the link, while Table III provides details on the latency of GTX subblocks in the adopted configuration.

In the described architecture, the ADC produces 16-b samples at 200 MHz. The latency can be kept fixed assuming all the frequencies in the design can be scaled accordingly. For instance for a 370-MHz sampling rate, the JESD link would run at 7.4 Gb/s; the reference and the GTX user clocks at 231.25 MHz; and the FEC link at 9.25 Gb/s.

IV. IMPLEMENTATION

We implemented the remote and local ends of our architecture in Xilinx Kintex-7 [17] 325T FPGA, which is a medium-sized device. Tables IV and V report separately the resource occupation for the JESD interface and the FEC-link. The overall logic footprint is tiny, equal, or below 2% in terms of lookup tables and flip-flops for both the JESD-interface and the FEC-link. A small Kintex-7 FPGA, such as the 70T, would have plenty of resources to implement the on-detector logic.

The demonstrator architecture we implemented includes a single FEC-link between the remote and local node; however, in a real application, the local (off-detector) node would aggregate several links from remote boards. In fact, the processing may depend on data acquired from several remote (on-detector) boards (e.g., for trigger algorithms). The actual limit to the local FPGA size is given by the number of available transceivers (16 for the 325T). The available global clock buffers might look like another limiting factor, as each FEC-link uses seven out of the 32 available; however, it is important to note that also regional clock buffers, of which there are 40 in the device, could be used.

V. TEST RESULTS

Our test setup (Fig. 9) includes two Xilinx KC705 demo boards, acting as the local and remote boards, interconnected by means of 50-Ω coaxial cable pairs providing the physical layer for the FEC-link. A Texas Instruments

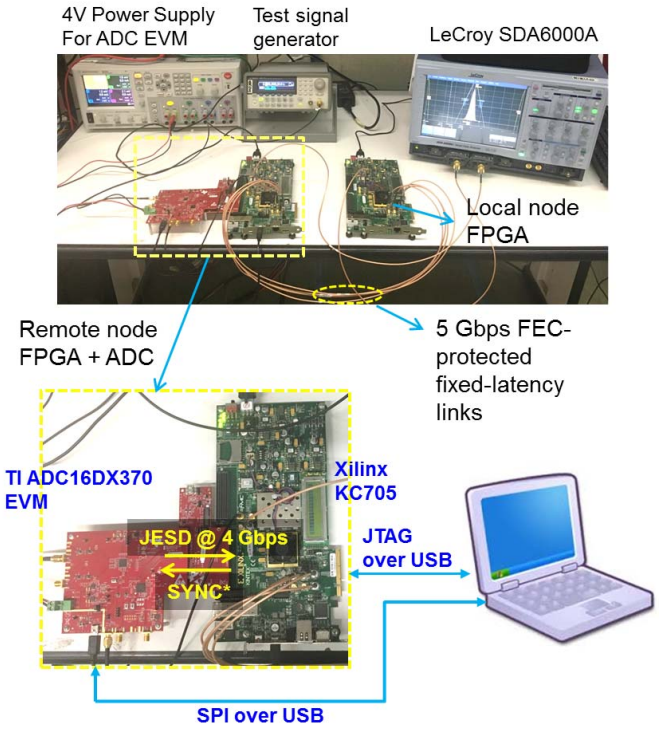

Fig. 9. Photo of the test setup for functional tests, jitter, and latency measurements.

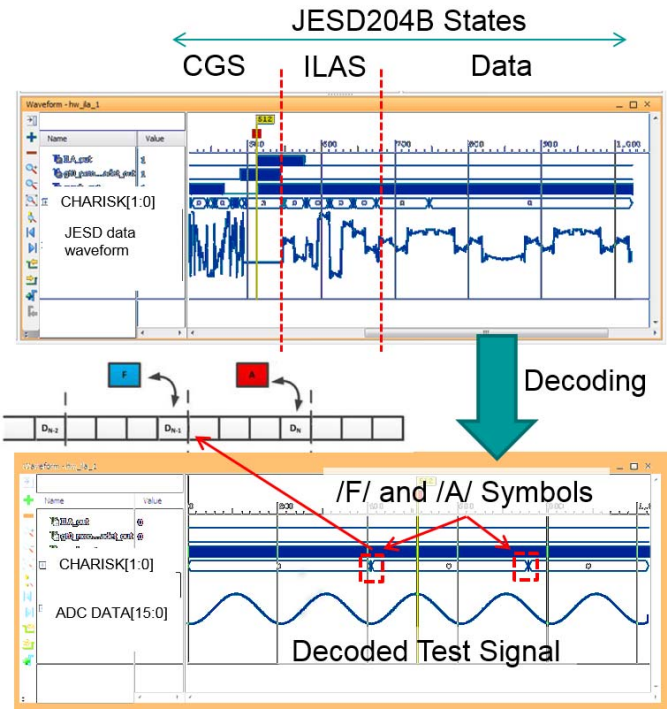

Fig. 10. Screenshots of Xilinx Hardware Manager showing the JESD204B deserialized data (top) and the test signal digitized by the ADC (bottom).

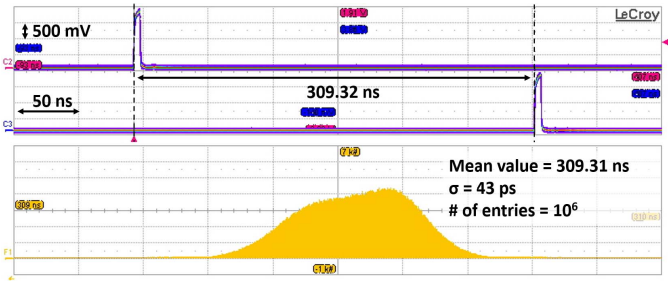

Fig. 11. Oscilloscope screenshot showing the measured latency for the FEC-protected link. Latency mean value = 309.31 ns, standard deviation = 43 ps, and number of entries 10^6.

ADC16DX370EVM demo board was plugged on the high-pin-count FPGA mezzanine card connector of the remote KC705. The ADC16DX370EVM hosts an ADC16DX370 [18] high-speed ADC and an LMK04828 [19] ultralow noise jitter cleaner. Both the devices are designed for JESD204B applications. A generator provides the needed 4-V supply voltage for the ADC16DX370EVM board. A signal generator drives one of the ADC inputs with a sine wave (1 MHz, 500 Vpp, 0 VDC) as a test signal. At each system reset, a personal computer running a dedicated TI software configures the ADC and the jitter cleaner by means of SPI bus over a USB connection. The same PC is connected to the remote KC705 board via JTAG, and it allows us to control and debug JESD-related features and to display acquired waveforms via the Xilinx Hardware Manager software (Fig. 10).

In addition to verifying the functionality of the system, we measured the rms timing jitter of the following:

1) the reference clock for the transmitter at the local node (J_{ref});
2) the recovered clock from the FEC-link at the remote node (J_{rec});
3) the cleaned clock from the jitter cleaner (J_{clean}), which drives the reference clocks for the ADC16DX370 and the GTX-RX in the JESD204B link.

We used a LeCroy SDA6000A [20] oscilloscope running a dedicated jitter analysis software, which allowed us to estimate the rms jitter by means of a best fit with a dual-Dirac model [21]. The measured jitter values were $J_{\text{ref}} = 15$ ps, $J_{\text{rec}} = 15$ ps, and $J_{\text{clean}} = 12$ ps.

In general, phase noise at the input impacts the performance of a jitter cleaner, since part of it falls below the closed-loop cutoff frequency and cannot be filtered by the PLL. The LMK04828 data-sheet reports integrated rms jitters of the order of 100 fs, which, however, are taken in ideal conditions. During the device characterization, the input was driven by an ultraclean reference VCXO [22], with 40-fs typical phase jitter (integrated from 12 kHz to 20 MHz). In addition to that, normally in this kind of measurements [23], the device under test has a dedicated low-noise power supply [24], since noise on the supply voltage also translates to an increased output jitter. However, in our tests, the cleaner still brings the jitter at a level lower than the one at the transmitter's reference clock.

The impact on the ADC signal-to-noise ratio (SNR) depends on the jitter at the device clock input and on the input frequency spectrum. Assuming an infinite resolution ADC, the theoretical limit for the rms SNR for a full-scale sine-wave input of frequency f, sampled with a T_j timing jitter, is [25]

$$\text{SNR (dBFS)} = -20\log_{10}(2\pi f T_j). \quad (1)$$

For a full-scale 1-MHz sine wave and the measured 12-ps rms jitter, the SNR results to be 82.5 dBFS, which is beyond the theoretical SNR performance limit of the ADC16DX370 reported by the data-sheet (72 dBFS), therefore no degradation can be attributed to the reference clock jitter. An SNR degradation of -3 dBFS would be experienced for a 5-MHz sine wave.

We also measured the latency through the FEC-protected link, and we verified that after more than 10^4 resets, the latency trough the FEC link remains constant at a level below one unit interval (Fig. 11). The total latency was measured to be 309.31 ± 0.04 ns, which, taking into account a 5-ns cable delay, it is consistent with the one expected by the documentation and postroute simulations.

VI. Conclusion

We developed a simple, low logic-footprint, JESD204B-compliant receiver core supporting a single lane. We tested it at 4 Gb/s, receiving data from a 200 MS/s ADC running a 16-b conversion. We developed an FEC-protected, deterministic-latency 5-Gb/s link for transferring digitized data toward a remote data consumer. The whole data transfer path features deterministic latency thanks to the JESD204B protocol and the fixed-latency architecture of the FEC-protected link. Our results show that the rms jitter on JESD204B device clocks forwarded over the FEC-protected link can be significantly reduced with respect to the reference clock at the transmitter end. Moreover, the latency through the FEC-protected link has been measured to be stable within one unit interval.

Acknowledgment

The authors would like to thank M. Salvatore from the University of Naples "Federico II"- Dipartimento di Ingegneria Chimica, dei Materiali e della Produzione Industriale, for her help with figures.

References

[1] S. Ritt, R. Dinapoli, and U. Hartmann, "Application of the DRS chip for fast waveform digitizing," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 623, no. 1, pp. 486–488, Nov. 2010.

[2] S. Masuda *et al.*, "Development of the photomultiplier tube readout system for the first large-sized telescope of the Cherenkov telescope array," in *Proc. 34th ICRC*, The Hague, The Netherlands, Jul./Aug. 2015. [Online]. Available: https://pos.sissa.it/archive/conferences/236/1003/ICRC2015_1003.pdf

[3] E. Oberla *et al.*, "A 4-channel waveform sampling ASIC in 0.13 μm CMOS for front-end readout of large-area micro-channel plate detectors," in *Proc. TIPP*, Chicago, MI, USA, 2012, pp. 1690–1698.

[4] D. Breton *et al.*, "High resolution photon timing with MCP-PMTs: A comparison of a commercial constant fraction discriminator (CFD) with the ASIC-based waveform digitizers TARGET and WaveCatcher," in *Proc. IEEE NSS MIC*, Oct./Nov. 2010, pp. 856–864, doi: 10.1109/NSSMIC.2010.5873883.

[5] *Serial Interface for Data Converters*, JEDEC Standard JESD204B.01, 2012.

[6] *JESD204B IP Core*, Comcores ApS, Hoersholm, DK, USA. [Online]. Available: http://www.xilinx.com/products/intellectual-property/1-5so323.html

[7] *JESD204B IP Core User Guide*, Altera Corp., San Jose, CA, USA, accessed on Oct. 31, 2016. [Online]. Available: https://www.altera.com/en_US/pdfs/literature/ug/ug_jesd204b.pdf

[8] *JESD RX/TX IP CORE*, KuanTek Electron. Inf. Technol. Ltd., Başiskele, Turkey. [Online]. Available: http://www.kuantek.com.tr/ipcekirdek/KNTK-IP-JESD204B-TX_RX-100-PB.pd

[9] *JEDEC JESD204B IP Core*, Desig&Reuse S.A, Grenoble, France. [Online]. Available: http://www.design-reuse.com/sip/jesd204b-controller-ip-28162/

[10] A. Klein, "Linear feedback shift registers," in *Stream Ciphers*. Berlin, Germany: Springer, 2013, pp. 17–58.

[11] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields," *J. Soc. Ind. Appl. Math.*, vol. 8, no. 2, pp. 300–304, Jun. 1960.

[12] S.-L. Shieh, S.-G. Lee, and W.-H. Sheen, "A low-latency decoder for punctured/shortened Reed–Solomon codes," in *Proc. IEEE 16th PIMRC*, Sep. 2005, pp. 2547–2551.

[13] G. Balakrishnan, M. Yang, Y. Jiang, and Y. Kim, "Performance analysis of error control codes for wireless sensor networks," in *Proc. ITNG*, Las Vegas, NV, USA, Apr. 2007, pp. 876–879, doi: 10.1109/ITNG.2007.149.

[14] S. Lin and D. J. Costello, "Cyclic codes," in *Error Control Coding: Fundamentals and Applications*. New York, NY, USA: Pearson Education International, 2004, pp. 85–123.

[15] R. Giordano and A. Aloisio, "Fixed-latency, multi-gigabit serial links with Xilinx FPGAs," *IEEE Trans. Nucl. Sci.*, vol. 58, no. 1, pp. 194–201, Feb. 2011, doi: 10.1109/TNS.2010.2101083.

[16] R. Giordano and A. Aloisio, "Protocol-independent, fixed-latency links with FPGA-embedded SerDeses," *J. Instrum.*, vol. 7, p. P05004, May 2012.

[17] *Kintex-7 FPGAs Data Sheet: DC and AC Switching Characteristics*, Xilinx Inc., San Jose, CA, USA, 2015. [Online]. Available: http://www.xilinx.com/support/documentation/data_sheets/ds182_Kintex_7_Data_Sheet.pdf

[18] *ADC16DX370 Dual 16-Bit 370 MSPS ADC With 7.4 Gb/s JESD204B Outputs*, Texas Instruments Inc., Dallas, TX, USA, 2014.

[19] *LMK0482x Ultra Low-Noise JESD204B Compliant Clock Jitter Cleaner With Dual Loop PLLs*, Texas Instruments Inc., Dallas, TX, USA, 2015.

[20] *Serial Data Analyzer Operator's Manual*, LeCroy Corporation, Chestnut Ridge, NY, USA, accessed on Dec. 2007. [Online]. Available: http://cdn.teledynelecroy.com/files/pdf/lecroy_sda6000_datasheet.pdf

[21] R. Stephens, "Jitter analysis: The dual-Dirac model, RJ/DJ, and *Q*-scale," Agilent Technol., Santa Clara, CA, USA, White Paper, Dec. 2004.

[22] *CVHD-950 VCXO Ultra-Low Phase Noise Oscillators*, Crystek Corp., Fort Myers, FL, USA, accessed on 16, 2017. [Online]. Available: http://www.crystek.com/crystal/spec-sheets/vcxo/CVHD-950.pdf

[23] M. Balasubramanian, "CDCE62005 phase noise performance and jitter cleaning ability," Texas Instrum. Inc., Dallas, TX, USA, Appl. Rep. SCAA095, 2008. [Online]. Available: http://www.ti.com/lit/an/scaa095/scaa095.pdf

[24] *6620A Series Multiple-Output, 40–105 W DC Power Supplies, GPIB*, Keysight Technol., Santa Rosa, CA, USA, 2014. [Online]. Available: http://literature.cdn.keysight.com/litweb/pdf/5990-9302EN.pdf

[25] W. Kester, "Aperture time, aperture jitter, aperture delay time—Removing the confusion," Anal. Devices Inc., Norwood, MA, USA, Tech. Rep, 2009. [Online]. Available: https://www.researchgate.net/profile/Degang_Chen/publication/273339582_A_Comparative_Study_of_State-of-the-Art_High-Performance_Spectral_Test_Methods/links/5665a55f08ae15e74634a522.pdf