

An FPGA-based Emulation Platform for Evaluation of Time-Interleaved ADC Calibration Systems

Raúl M. Sanchez[†], Benjamín T. Reyes[‡], Ariel L. Pola[†], and Mario R. Hueda[‡]

[†] Fundación Fulgor - Romagosa 518 - Córdoba (X5016GQM) - Argentina

[‡] Laboratorio de Comunicaciones Digitales - IDIT - UNC - CONICET

Av. Vélez Sarsfield 1611 - Córdoba (X5016GCA) - Argentina

Emails: {breyes, mhueda}@efn.uncor.edu , arielpola@gmail.com

Abstract—This work describes a 1 Gb/s digital communication system implemented on an FPGA-based platform to investigate mixed-signal calibration techniques of time-interleaved analog-to-digital converters (TI-ADCs). Design of multi-gigabit TI-ADCs is of great interest for next generation digital communication systems such as optical coherent networks. In these applications, mismatches of the sampling time, gain, offset, and frequency response among the interleaves of a TI-ADC limit the performance of the converter unless they are compensated. Typically, long computer simulation run time is required to evaluate the performance of mixed-signal calibration algorithms. We show that the FPGA-based system described in this paper drastically reduces the emulation time by more than hundreds of magnitude orders. The proposed FPGA framework includes: (i) a diagnostic and control unit built upon an embedded processor NIOSII, (ii) DSP blocks to implement the transmitter and the receiver, and (iii) a Gaussian number generator to emulate the noise channel component. Experimental results with a 2 GS/s 6-bit CMOS TI-ADC demonstrate the excellent capability of the implemented FPGA-based emulator to evaluate the performance of a mixed-signal calibration algorithm.

Index Terms—DSP, digital communication, feed-forward equalizer, FPGA, optical fiber, time-interleaved ADC, parallel processing.

I. INTRODUCTION

The new generation of fiber optical transceivers is based on complex digital signal processing (DSP) to compensate channel impairments such as chromatic dispersion or polarization-mode dispersion [1]. A typical dual-polarization (DP) quadrature phase shift keying (QPSK) coherent optical DSP receiver architecture is depicted in Fig. 1 (see [1] for details). In this architecture, the analog input signals consist of four channels, corresponding to the in-phase and quadrature components of the two polarizations. The signals are amplified and converted to digital domain in the analog front-end (AFE) by using four high-speed time-interleaved analog-to-digital converters (TI-ADCs). Mismatches of the sampling time, gain, offset, and frequency response among the interleaves of a TI-ADC limit the performance of the transceiver unless they are compensated. In particular, mixed-signal calibration techniques have been proposed for coherent optical transceivers [2].

Typically, long computer simulation run time is required to evaluate the performance of the different algorithms used

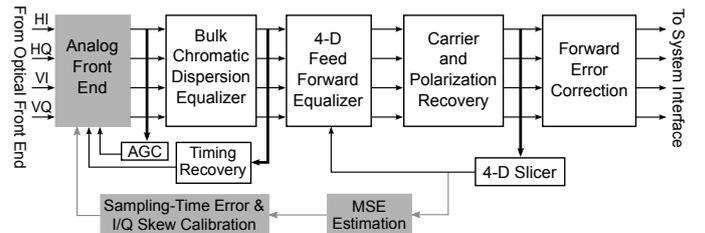


Figure 1. Digital signal processor architecture for a DP-QPSK coherent receiver.

in these receivers. Hardware emulation platforms can speed-up the testing and validation process. Field programmable logic array (FPGA) chips are particularly suitable for emulation/implementation of digital communication systems. For example, the performance of forward error correction (FEC) blocks is commonly evaluated by FPGA emulation as a result of the large amount of symbols required (e.g., $\sim 10^{15}$) to provide reliable estimation of the bit-error-rate (BER) [3]. However, FPGA emulation is not limited to FEC blocks, and it could be useful for any other kind of algorithms. This is the case of the algorithm proposed in [2] for sampling-time error calibration in TI-ADC. This is a mixed-signal calibration technique based on mean squared error (MSE) minimization to adjust the clock phases of parallel ADCs (see blocks in gray in Fig. 1). Since this algorithm is based on a relative *slow* averaging of MSE at slicer, the performance verification would require a large amount of symbols processing. For example, in order to perform a phase adjustment step in one ADC channel, the simulator processes around 10^7 symbols [2]. If this value is extrapolated for a calibration sequence to 1000 iterations, $\sim 10^{10}$ symbols would be required. This quantity of symbols will take only a few milliseconds in a 100 Giga-Samples/seconds (GS/s) optical receiver. Nevertheless, the computation time demanded for a simulation can scale from weeks to months if we test various signal cases. In this context, an FPGA platform to emulate a communication system is emerging as the best option for an experimental demonstration of the techniques proposed in [2].

In this work we describe the implementation of an emulator of digital communication systems based on FPGA. The system includes a simplified wireline transceiver DSP running over

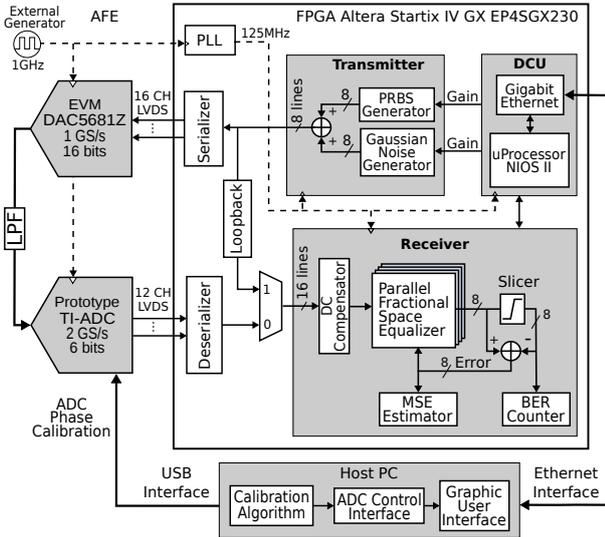


Figure 2. Simplified block diagram of the transceiver.

a high performance FPGA. The scheme is completed using an AFE based on a prototype high-speed data converter specially designed for verification of calibration techniques [4]. The transceiver is capable of processing sampling data at 2 GS/s and 1 Giga-bit/second (Gb/s) symbol rate. The emulation platform and the TI-ADC can be controlled and configured from an external interface so that sampling phase calibration techniques can be implemented. Furthermore, the system and the TI-ADC signals can be logged in real time from a computer. Experimental results demonstrate the drastic reduction of the emulation time for evaluating the performance of a mixed-signal calibration algorithm.

The rest of the paper is organized as follows. Section II describes the emulation platform architecture. Section III shows the experimental results of the system, while conclusions are drawn in Section IV.

II. FUNCTIONAL ARCHITECTURE AND IMPLEMENTATION DETAILS

Fig. 2 presents the basic diagram of emulated transceiver proposed. The system is composed by AFE, DSP and physical channel (electrical filter). Digital blocks (i.e. transmitter/receiver) and an embedded processor are implemented in the FPGA. The transceiver operates at a nominal symbol rate of 1 Gb/s ($T = 1 \text{ ns}$) and can emulate different scenarios of signal-to-noise ratio (SNR) (e.g. from 4 dB to 30 dB).

The AFE is comprised of a commercial 1 GS/s 16-bit DAC board (DAC5681ZEVM) [5] and a prototype 2 GS/s 6-bit TI-ADC [4]. The DAC is interfaced to the FPGA via 16 low-voltage-differential-signaling (LVDS) channels (1 Gb/s each). The TI-ADC is connected to the FPGA via 12 LVDS channels at 1 Gb/s each. This converter includes sampling phase calibration circuits that can be controlled from a computer (see [4] for details). The AFE boards are interconnected by a *communication channel* that is based on electrical low-pass

filters (LPF) like [6]. The channel bandwidth can be selected according to the required dispersion scenario.

The digital part of the platform is implemented on a Terasic DE4 kit which includes a Stratix IV GX EP4SGX230 FPGA [7]. The DAC and prototype TI-ADC are connected to Altera Kit by two High-Speed Mezzanine Card (HSMC) interfaces. This board is connected via USB and Ethernet to a computer to interact with the system. Additionally, a graphical user interface (GUI) software was developed to configure the TI-ADC registers (1 kb) via a USB port.

Finally, note that a fully digital loopback option (inner loopback) is included (see Fig. 2). It is used to test the DSP performance without considering the analog domain effects from AFE. As we show in Section III, the inner loopback is used for comparison purposes. In next sub-sections, the digital implementation in the FPGA is detailed.

A. Transmitter

The transmitter sub-system generates a pseudo-random binary sequence (PRBS) that can be merged with an additive white Gaussian noise (AWGN) signal to define the required signal-to-noise ratio (SNR) (Fig. 2). The PRBS sequence ($2^9 - 1$ symbols length) is implemented in a parallel architecture like [8]. The PRBS binary symbols are then encoded in BPSK modulation to obtain the ± 1 signal in 16-bit *fixed-point* resolution.

On the lower branch of the transmitter, the noise signal is generated by implementing eight parallel instances of an AWGN IP block with different seeds [9].

The transmitter also includes independent 16-bit gain coefficients for both, signal and noise paths. This allows the user to set any required SNR value (i.e. from pure noise signal to high SNR signal), with very high resolution and precision. The resulting output signal is then serialized, transmitted, and synthesized directly to the physical channel by the DAC board.

B. Receiver

The core of the receiver is an adaptive fractional spaced equalizer (FSE) filter (Fig. 2). The FSE is implemented with a parallel architecture based on the new generation of coherent optical systems architecture [1]. The receiver input registers operate at a synchronous $T/2$ sampling rate (i.e. 2 GS/s) but the FSE clock is the symbol rate T (1 GHz). Previous to the FSE inputs, a DC offset cancellation can be set so that the DC offset mismatch between the parallel ADCs can be compensated¹.

At the output of the FSE and slicer blocks, the signal is decoded and sent to the bit-error counter. This last block is used for measurement purposes and it allows for the verification of system performance, comparing theoretical BER values with the real platform results. The block includes a correlation algorithm for detection and synchronization of the known PRBS sequence.

¹Note that synchronization and timing recovery blocks are not required because $T/2$ sampling rate is used and an external clock signal is shared between the FPGA and converter boards.

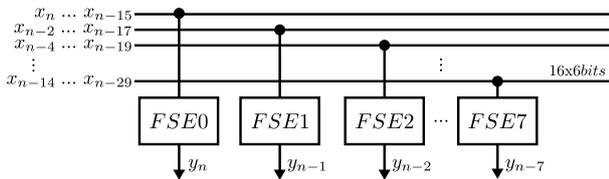


Figure 3. Parallel implementation of the FSE.

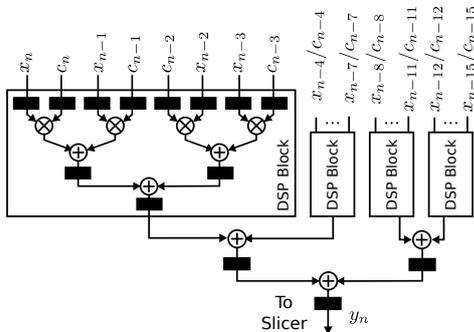


Figure 4. Fractional spaced equalizer

The other relevant block for this platform is the MSE estimator. It provides the feedback signal required for the TI-ADC calibration algorithm. The MSE estimation is obtained using a recursive filter defined by

$$\text{MSE}_n = (1 - \alpha)\text{MSE}_{n-1} + \alpha e_n^2, \quad (1)$$

where MSE_n and MSE_{n-1} are the current and previous values of the MSE estimation respectively, α is the updating factor and e_n is the current error between the FSE and slicer output. The parallel architecture used for this recursive filter is based on [10]. The operation of this block is controlled externally, so that the α factor, reset signals and outputs can be controlled and monitored by the user via the control unit.

The FPGA implementation of the receiver was previously designed and verified in a system level simulator which was coded using the SystemC library to validate the architecture. The first version of the simulator was written using *floating-point* variables and then a *fixed-point* version was developed to define the quantization resolution required in the signal paths.

1) *Fractional Spaced Equalizer*: In Fig. 3 the basic structure of the FSE architecture is introduced. It is based on [1], [11] proposals and it is formed by 8 parallel filters with 16-tap (coefficients) each. The equalizer is adapted by a least mean square algorithm (LMS) [12] and the adapting step factor can be externally set with different values to control the convergence speed. As any other parallel architecture, the trade-off between speed and complexity has to be considered. Unlike [1], [11], where a dedicated chip was used, this implementation was conducted in an FPGA. Due to the limited resources of the latter platform, a detailed study of the dedicated DSP blocks was required in order to optimize the use of resources [13].

In Fig. 4, the detailed architecture of each parallel filter is shown. DSP blocks are used to multiply each sample by the corresponding coefficient and accumulate the partial result.

Table I
SYNTHESIS REPORT

Block	Comb. ALUTs	Registers	Memory Blocks		DSP Elements			
			M9K	M144K	18-bit	12x12	18x18	36x36
Receiver								
FSE	5684	8629	3	0	747	128	0	144
BER Counter	1386	560	0	0	0	0	0	0
MSE Estimation	612	438	1	0	52	0	8	9
Transceiver								
PRBS	568	80	0	0	0	0	0	0
GNG	3536	3622	20	0	48	0	32	0
Diagnostic and Control Unit								
NIOS System	9520	11539	815	2	4	0	0	1
Logger	379	577	3	15	0	0	0	0
Other	4343	4887	20	0	80	0	32	0
Total	26028	30332	862	17	931	128	72	154

The final result is implemented with logic elements through a parallel adder. With this topology, the amount of conventional logic resources required for multiplication/addition is drastically reduced and also the complexity of the routing is simplified. These also benefit in speed terms due to the minimization of timing problems in the signal paths.

On the other hand, it is important to emphasize that the FSE implementation design is parameterizable in many aspects such as the parallelism factor, number of coefficient taps, and signal path resolutions. However, these parameterizations are limited to the FPGA capabilities. For example, it has been determined that in the FPGA up to 20 tap filters with a parallelism of 8 and up to 28-taps with parallelism of 4 could be implemented.

C. Diagnostic and Control Unit

The diagnostic and control unit (DCU) is used to configure and register variables of the transceiver block previously described. It is based on an embedded NIOS II processor that runs a real-time operating system (RTOS). The RTOS is used to generate a socket server which enables an Ethernet connection to the emulation platform. A client application was developed to communicate with the server to perform different operations such as controlling noise injection, modifying FSE adaptation step, MSE logging, among others.

Furthermore, a massive logging system with 16,384 words of 128 bits was implemented with dedicated memory blocks RAM (dual-port) of FPGA.

III. EXPERIMENTAL RESULTS

A. Synthesis Results

The digital blocks presented in Section II were implemented using the Verilog hardware description language and properly verified through Quartus II from Altera [13] for RTL simulations. The functional verification was executed for each module individually as well as for their integration. For physical verification, the digital blocks were synthesized for the Stratix IV GX FPGA development kit from Altera [7]. The resource usage is detailed in Table I.

Table II
PROCESSING RUN-TIME

BER performance analysis			
Receiver Performance		Time Required	
SNR(dB)	BER	Simulation(min)	Emulation(min) [‡]
6	2.38×10^{-3}	1.02 [‡]	5.42×10^{-6}
10	3.87×10^{-6}	60 [‡]	1.67×10^{-3}
12	9.00×10^{-9}	1.97×10^5 [†]	1
13	1.33×10^{-10}	3.10×10^6 [†]	15

Time required to evaluate the calibration algorithm			
ADC Calibration Case		Time Required	
Symb/Iter.	N ^o Iter.	Simulation(min) [†]	Emulation(min) [‡]
10^7	100	3.24×10^3	4.33×10^{-1}
10^9	100	3.31×10^4	5.83×10^{-1}
10^{11}	100	3.26×10^7	180

[†] Estimated values

[‡] Measured values

B. Measurements

The objective of using this platform was not only verifying the functionality of TI-ADC calibration system, but also measuring its performance for AWGN channel over a SNR range ($SNR = E\{|x|^2\}/\sigma^2$). Fig. 2 depicts the typical setup used for the emulation platform, which includes the FPGA, DAC, and ADC boards. The communication channel in this setup is an electrical LPF of ~ 650 MHz [6].

Fig. 5 shows the BER versus SNR for two specific configurations. The first one uses the inner loop bypassing the DAC/ADC converters and no penalty is observed by comparing the theoretical (BPSK modulation) and simulated curves. In the second case, the results are compared operating in outer loop and show a minimal SNR penalty due to the AFE (ADC/DAC) and channel impairments. However, an extra SNR penalty (i.e. BER degradation) is observed when the TI-ADC is set in a scenario with large sampling-phase error and it demonstrates the need of calibration for this case. For the application considered in this paper (i.e., evaluation of an ADC calibration technique), values of $SNR \leq 13$ dB are sufficient to demonstrate the benefits of the proposed platform.

Finally, time comparison between simulation/emulation are summarized. The simulator was based on the architecture presented in Section II running on a computer with an Intel i5-3570 processor and 16GB of RAM. Approximately, 5416 samples per second are processed by the simulator. In Table II, a comparison between the time spent obtaining the BER curves of Fig. 5 is presented. For example, the computer simulation processing time for the SNR range between 6 dB and 11 dB was ~ 180 minutes, compared with the FPGA emulation that only took $\sim 2.5 \times 10^{-2}$ minutes. Similar analysis is performed for the ADC calibration algorithm case. There, the number of symbols per each sampling-phase correction step is considered. Table II summarizes the execution time required for three cases of calibration. For example, the time of full implementation for the first case was 4.33×10^{-1} minutes compared to 3.24×10^7 minutes in the simulator. These results demonstrate the advantage of using the mentioned platforms for verification of mixed-signal calibration algorithms.

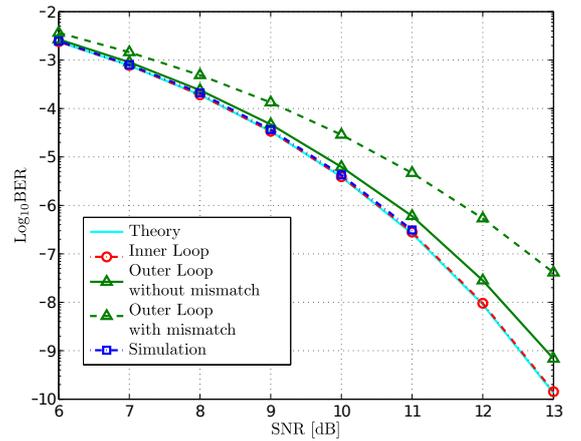


Figure 5. Receiver BER curve performance

IV. CONCLUSION

We have described a digital communication system implemented on an FPGA-based platform to investigate mixed-signal calibration techniques of TI-ADCs. We showed that the FPGA-based system described in this paper reduces drastically the emulation time by more than hundreds of magnitude orders. Experimental results with a 2 GS/s 6-bit CMOS TI-ADC have demonstrated the excellent capability of the implemented FPGA-based emulator to evaluate the performance of a mixed-signal calibration algorithm.

REFERENCES

- [1] D. Crivelli *et al.*, "Architecture of a Single-Chip 50 Gb/s DP-QPSK/BPSK Transceiver With Electronic Dispersion Compensation for Coherent Optical Channels," *IEEE Trans. Circuits Syst. I*, vol. 61, no. 4, pp. 1012–1025, Apr. 2014.
- [2] B. T. Reyes *et al.*, "Joint sampling-time error and channel skew calibration of time-interleaved ADC in multichannel fiber optic receivers," in *2012 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2012, pp. 2981–2984.
- [3] D. Morero *et al.*, "Non-Concatenated FEC Codes for Ultra-High Speed Optical Transport Networks," in *2011 IEEE Global Telecommunications Conference (GLOBECOM 2011)*, Dec. 2011, pp. 1–5.
- [4] B. T. Reyes *et al.*, "A 2GS/s 6-bit CMOS time-interleaved ADC for analysis of mixed-signal calibration techniques," *Analog Integr Circ Sig Process*, vol. 85, no. 1, pp. 3–16, Jul. 2015.
- [5] Texas Instruments, "DAC5681Z Evaluation Module - User Guide," 2008. [Online]. Available: <http://www.ti.com/lit/ug/slau236a/slau236a.pdf>
- [6] Mini-Circuits, "Low Pass Filter VLF-490+ - Datasheet," 2008. [Online]. Available: <http://www.minicircuits.com/pdfs/VLF-490+.pdf>
- [7] Altera, "Stratix IV GX FPGA Development Kit User Guide," Mar. 2014. [Online]. Available: https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/ug/ug_sivgx_fpga_dev_kit.pdf
- [8] J. J. O'Reilly, "Series-parallel generation of m-sequences," *Radio and Electronic Engineer*, vol. 45, no. 4, pp. 171–176, Apr. 1975.
- [9] G. Liu, "Gaussian Noise Generator," Jul. 2014. [Online]. Available: http://opencores.org/project_gng
- [10] K. K. Parhi, *VLSI Digital Signal Processing Systems: Design and Implementation*, 1st ed. Wiley-Interscience, May 2008.
- [11] O. Agazzi *et al.*, "A 90 nm CMOS DSP MLSD Transceiver With Integrated AFE for Electronic Dispersion Compensation of Multimode Optical Fibers at 10 Gb/s," *IEEE J. Solid-State Circuits*, vol. 43, no. 12, pp. 2939–2957, 2008.
- [12] J. R. Barry, E. A. Lee, and D. G. Messerschmitt, *Digital Communication*, 3rd ed. Springer, Sep. 2003.
- [13] Altera, "Stratix IV Device Handbook," Jun. 2015. [Online]. Available: https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/hb/stratix-iv/stratix4_handbook.pdf