

Available online at [www.sciencedirect.com](http://www.sciencedirect.com)**ScienceDirect**

Procedia Technology 17 (2014) 319 – 326

**Procedia**  
Technology

Conference on Electronics, Telecommunications and Computers – CETC 2013

## Emulation in FPGA of G-Link Chip-set of Tile Calorimeter Electronic System

José Alves<sup>a,b\*</sup>, José Silva<sup>a</sup>, Guiomar Evans<sup>b,c</sup>, José Soares Augusto<sup>b,d</sup><sup>a</sup>Laboratório de Instrumentação e Física Experimental de Partículas, Av. Elias Garcia 14-1, Lisbon 1000-149, Portugal<sup>b</sup>Faculdade de Ciências da Universidade de Lisboa, Campo Grande, Lisbon 1749-016, Portugal<sup>c</sup>Centro de Física da Matéria Condensada, Campo Grande, Lisbon 1749-016, Portugal<sup>d</sup>Instituto de Engenharia de Sistemas e Computadores, Rua Alves Redol 9, Lisbon 1000-029, Portugal

---

### Abstract

The ATLAS Tile Calorimeter front-end and the back-end electronics use the Agilent HDMP-1032/1034 Transmitter/Receiver chip-set as physical layer. In the back-end is implemented the G-link receiver, the HDMP-1034, which is used to receive and deserialize the data sent by a G-Link transmitter, the HDMP-1032 in the front-end. The emulation of their functionality, in a FPGA, was required to be used in a TileCal front-end electronics tester, called MobiDICK4, implemented at CERN, with future utilization planned for the upgrade of the TileCal electronic system. We have implemented the emulation of the functionality of this chip-set in a Xilinx ML605 board equipped with a Virtex-6 FPGA, around the high-speed serial GTX transceivers embedded in that FPGA. A description of the functionality and the CIMT coding scheme used by this chip-set will be presented. Also, the implementation of our design in the FPGA will be described. Finally, the tests performed, the results and the conclusion about this work will be presented.

© 2014 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/3.0/>).

Peer-review under responsibility of ISEL – Instituto Superior de Engenharia de Lisboa, Lisbon, PORTUGAL.

*Keywords:* TileCal, G-link, chip-set, FPGA

---

\* Corresponding author. Tel.: +351 967-348-876;  
E-mail address: [jalves@lip.pt](mailto:jalves@lip.pt)

## 1. Introduction

The Tile Calorimeter (TileCal) is the central hadronic calorimeter of the ATLAS detector installed at the Large Hadron Collider (LHC) of CERN. It provides accurate measurement of the energy of jets of particles from the proton-proton collisions that occur during the LHC experiments, the measurement of Missing Transverse Energy and the identification of low-pt muons. It has a fast and precise electronics system responsible for acquisition, processing, conditioning and digitalization of the signals produced by the calorimeter (front-end electronics), and provides data for the back-end electronics which is also fast and precise [3]. The LHC [1] is a particle collider with 27 km of perimeter; its main goal is to allow collisions of two beams of protons or heavy ions traveling in opposite directions. The collisions occur at key points, where are located the LHC detectors (ATLAS, CMS, LHC-b and ALICE) [2]. The protons travel in bunches that intersect in the key points with a frequency of 40 MHz.

A tester coined MobiDICK (*Mobile Drawer Integrity Checking System*) is used to check the integrity and the correct behavior of the TileCal front-end electronics [4]. One of the functions of MobiDICK is to test the readout from the Tile Cal front-end electronics, by checking incoming data send by the G-Link Transmitter (HDMP-1032) using optical links [3, 4]. A new version of MobiDICK was recently implemented at CERN; this new version is implemented around a ML507 board equipped with a Virtex-5 FPGA. To make possible the readout test from the front-end electronics, a module which emulates the functionality of the G-link receiver (HDMP-1034) had to be included in the MobiDICK, to receive the data sent by the front-end electronics. The Agilent Company discontinued the production of this G-Link chip-set, so the G-Link transmitter emulation was performed for future utilization planned for the upgrade of the TileCal electronic system.

This paper is structured in 6 sections. In section 2 is presented the TileCal front-end electronics. Section 3 is dedicated to the description of G-Link chip-set basic functionality. The implementation of our design in the FPGA is described in section 4. Section 5 is dedicated to the description of the tests performed and the results. Finally, in section 6, there are the final considerations and conclusions about this work.

## 2. Tile Calorimeter Front-end Electronics

The current front-end electronics for signal acquisition from the TileCal modules is located inside compact structures usually called drawers (Fig. 1(a)). Two drawers forms the so-called, *super-drawer* and each one is associated with a module of the TileCal, and is able to acquire data from 32 or 45 channels of the detector [3, 5]. The Tile Calorimeter is a sampling calorimeter made of steel as absorber medium and scintillating tiles as active material [4]. The ionizing particles crossing the TileCal produce photons in the scintillating tiles, and the intensity of these photons is proportional to the energy deposited by the particles. These photons are absorbed and guided by Wavelength Shifting Fibers (WLSFs) to a set of photomultiplier tubes (PMTs). The PMTs are responsible to convert light signals into electrical impulses, which are sent to a set of cards called *3-in-1* cards in the front-end electronics. The *3-in-1* cards perform the conditioning and processing of those analog signals, and send them to the digitizer system where they are digitized (sampling frequency of 40 MHz) and organized in data packets and sent to an Interface Board [3, 5]. The Interface Board collects the sampled data from all the digitizers, serializes and transmits them (using the G-link transmitter) to the back-end electronics (includes the G-link receiver) using optical links [3]. All this process is shown in Fig.1 (b).

The integrity of the data received by the back-end is checked with a CRC (Cyclic Redundancy Check) algorithm. The DMU (Data Management Unit) of the Digitizer system is responsible for organizing the digitized samples in packets of data. There are 8 digitizer boards in each super-drawer, and each one has 2 DMU devices, so there are 16 DMU devices per super-drawer. Each DMU builds its packet of data, computes two CRC values (one for even bits and other for odd bits) and appends them to the packet before sending it to the Interface Board. The Interface Board computes again the CRCs over each packet of data of each DMU, checking their integrity. If there is no error, the Interface Board builds a packet containing all the DMU packets of data and computes a global CRC over all the data of all DMUs before sending it to the back-end. The back-end electronics computes again the CRC of each DMU

data packet and of the global CRC, and compares them with the CRC values received. If the CRC computed by the back-end is equal to that sent by the front-end, there are no transmission errors [5].

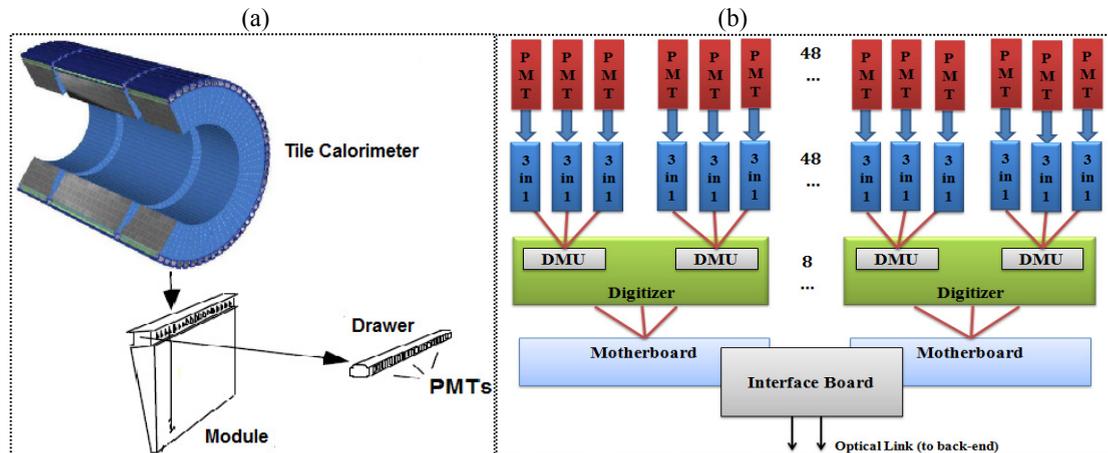


Fig.1. (a) TileCal drawer geometry; (b) TileCal front-end electronics.

The Motherboard provides low voltage power and digital control signals to four digitizer boards, one Interface Board and to the circuits needed for trigger summation and distribution [3].

### 3. Agilent HDMP-1032/1034 Transmitter/Receiver Chip-set

The G-Link chip-set HDMP-1032/1034 can be used to build a serial high speed data link for point-to-point communication and provides data rates that extend from 0.208 to 1.120 Gbit/s. This chip-set uses the Conditional Inversion Master Transition (CIMT) coding scheme to encode/decode the data. Some applications of this chip-set are: cellular base stations, ATM switches, video/image acquisition [6].

#### 3.1. Conditional Inversion Master Transition (CIMT)

The CIMT coding scheme ensures the DC balance of the serial line [6]. It uses three types of words: data words, control words and idle words. Each word to be transmitted consists of a Word Field (W-Field) followed by a Coding Field (C-Field). The C-Field is used to flag the type of word (data, control or idle words) to be transmitted and consists of 4 bits, which are added to the data input of 16 bits before transmission. The C-Field has a master transition in the middle (Fig.2), which allows the receiver to check for this transition in received data in order to perform word alignment and error detection. This master transition also serves as the reference for the receiver clock recovery circuit. When neither data words or control words are being sent, idle words are transmitted to maintain the DC balance in the serial link and allow the receiver to sustain frequency and phase lock [6].

The DC Balance of the serial line is ensured by monitoring the disparity (disparity is defined as the total number of high bits minus the total number of low bits) of successive encoded data words. The words are conditionally inverted in accordance with the disparity of the previously sent bits. If the sign of the current word is the same as the sign of the previously transmitted bits, then the word is inverted. If the signs are opposite, the word is not inverted. No inversion is performed if the word is an idle word [6]. If a word is inverted before its transmission, it will be inverted again at the receiver to recover the original word.

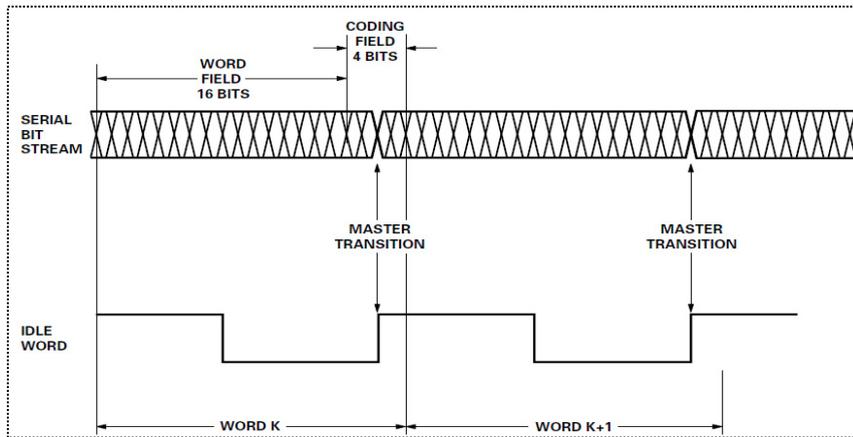


Fig. 2. CIMT coding scheme [6].

### 3.2. Basic Transmitter/Receiver Operations

The HDMP-1032 transmitter was designed to accept 16 bit wide parallel words, produces a 20 bit encoded word and transmits them over a high-speed serial line (coaxial optical cable or optical link). It performs the following functions: latching parallel word input, phase lock to the input clock signal, high speed clock multiplication, word encoding and parallel-to-serial multiplexing. A parallel word can be loaded into the transmitter chip and delivered to the receiver chip over a serial channel. The receiver chip reconstructs the serial data into its original parallel form [6].

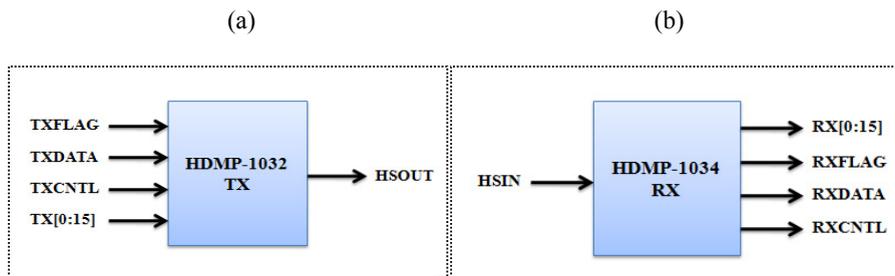


Fig.3. (a) HDMP-1032 Transmitter (left); HDMP-1034 Receiver (right).

Fig. 3 (a) represents the transmitter chip, showing the main input signals and the serial output signal (HSOUT). The data is encoded according with the state of the input control signals: TXFLAG, TXDATA and TXCNTL. If TXCNTL is high (means that the loaded TX [0:15] is to be sent as control word), then it sends the bits TX[0-13] and a C-Field (coding field) encoded as a control word no matter the state of TXDATA. If TXCNTL is low and TXDATA is high, it sends TX[0-15] and a C-Field encoded as a data word. If neither TXCNTL nor TXDATA are high, the transmitter chip assumes that the link is not being used. In this case, it submits an idle word to maintain the DC balance on the serial link and allow the receiver to maintain frequency and phase lock [6]. The entire word (Word-Field + C-Field) is constructed in parallel form and its sign is determined. The sign of the current word is compared with the accumulated sign of previously transmitted bits to decide whether to invert the word or not, in order to ensure the DC balance [6].

Fig. 3 (b) represents the receiver chip, showing the main output signals and the serial input signal (HSIN). The HDMP-1034 receiver converts the serial data signal sent from the HDMP-1032 into either 16 or 17 bit wide parallel data. It performs the following functions: frequency lock, phase lock, encoded word synchronization, de-

multiplexing, word decoding and encoding error detection [6]. The receiver performs the clock recovery from incoming data stream and all internal operations are synchronized with the recovered clock [6]. It decodes the 4 bits C-Field and determines whether the 16-bit Word-field consists in: normal or inverted; data, control, or idle words; or errors. The flag bit is also decoded from the data word [6]. If RXDATA is high it indicates that a data word is detected by the receiver. If RXCNTL is high it indicates that a control word is detected by the receiver. An idle word is detected by the receiver if RXDATA, RXCNTL and RXERROR are low [6].

#### 4. Implementation

We implement our design around the GTX transceivers embedded in the Virtex-6 FPGA. The GTX transceiver provides many features which allow physical layer support for many protocols. It provides line rates from 480 Mb/s to 6.6 Gb/s and can be configured to serialize/de-serialize data words of 8, 10, 16, 20 and 40 bits wide [7]. We configure the GTX transceiver to work with 20 bits data word, as required for the CIMT coding scheme and a parallel clock of 40 MHz was used as in the Tile calorimeter electronic system. A CIMT encoder/decoder module (written in VHDL) was developed to perform the required encoding/decoding tasks as the original chip-set. We use one GTX Transceiver which includes a transmitter and a receiver (on the same block) connected to the developed CIMT encoder/decoder.

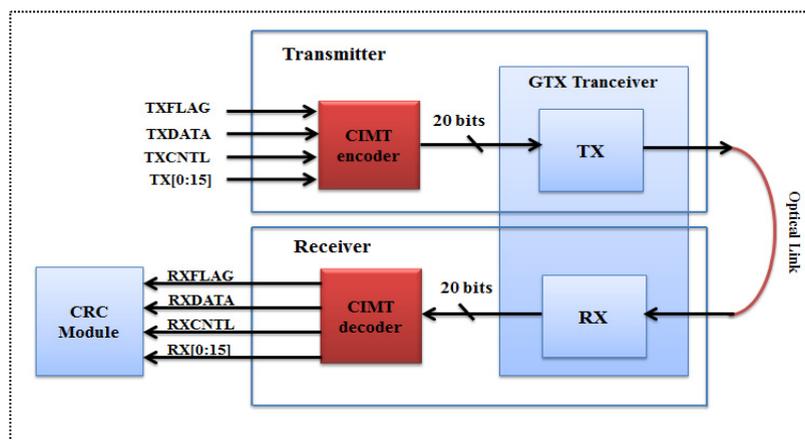


Fig.4. G-link chip set emulation.

Fig. 4 shows the implemented and tested design in a Xilinx ML607 board equipped with a Virtex-6 FPGA. This board includes a SFP (Small Form-factor Pluggable) cage connector which allows connecting the transmitter and the receiver through an optical fiber and testing the design in loopback mode.

The transmitter emulator includes a custom developed CIMT encoder and the transmitter block (TX) of one GTX transceiver, properly configured to meet our specifications. The CIMT encoder receives the 16 bit words loaded by the user, performs the encoding accordingly with the state of input control signals (TXFLAG, TXDATA and TXCNTL), adding the respective C-Field and producing a 20 bit encoded parallel word before sending it to the GTX TX. Also, the CIMT encoder module is responsible for monitoring the disparity of the transmitted bits in order to ensure the DC balance of the serial line, and it is able to transmit data, control and idle words. The GTX TX receives the 20 bit parallel words, serializes and transmits them over a serial line (in this case an optical channel).

As for the transmitter emulator, the receiver emulator includes a custom developed CIMT decoder and the receiver block (RX) of one GTX transceiver, also properly configured to meet our specifications. The GTX RX receives the serial bits and de-serializes them, producing a 20 bit encoded parallel word, before sending it to the CIMT decoder. The decoder performs the decoding process in accordance with the C-Field, and outputs the original words and three control signals (RXFLAG, RXDATA, RXCNTL and RXERROR) to the user. After the CIMT decoder performs the

data decoding, the reconstructed parallel word is sent to a CRC module. As we test the design with a well known data packet acquired from the TileCal front-end electronics, we use the same the CRC module (Fig.5) used in the actual TileCal electronic system (to check the data integrity sent by the front-end to the back-end electronics of TileCal, also described using VHDL), to check the integrity of the data sent by the transmitter and to count the errors. This CRC module is also implemented in the MobiDICK4 tester to check the data integrity in the read-out test from TileCal front-end electronics.

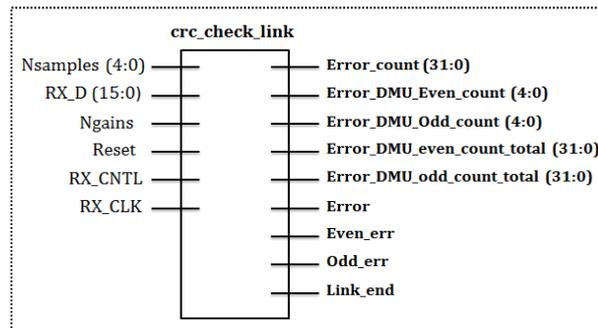


Fig.5. CRC Module used in the MobiDICK4 tester [3].

Fig.5 has the CRC module used in this test, showing the inputs and the outputs. The input signal *Nsamples* conveys the number of digital samples sent per DMU device. These samples are 32-bit words sent by each DMU to the Interface Board of the front-end. The signal *Ngains* is a signal related with *Nsamples*; if *Ngains* = '0', the number of samples is 7, and if *Ngains* = '1' the number of samples per DMU can take values up to 16. The 16-bit *RX\_D* terminal is the input of the words sent by the front-end and the CRC computation is performed on them. The signal *Reset* resets this module. *RX\_CNTL* is a control signal provided by the receiver emulator to start the CRC computations [6]. The front-end electronics performs two CRC computations for each DMU device. It performs separately one CRC computation over the even bits of the words, and another CRC computation over the odd bits. So we have two 31 bit counters, one for the errors in the even bits and another for the odd bits: *Error\_DMU\_even\_count\_total* and *Error\_DMU\_odd\_count\_total*. These two counters provide the number of errors for all packets of data received. There are another two 4-bit counters: *Error\_DMU\_Even\_count*, which provides the number of errors in each packet of data associated with even bits and *Error\_DMU\_Odd\_count*, which gives the number of errors in each packet of data associated with odd bits [3]. Also, the front-end computes a global CRC over the data of all 16 DMUs. So, the output signal *Error\_count* is a 31-bit counter providing the number of global CRC errors found for all packets of data sent by the front-end [3].

The GTX transceiver (TX and RX) has two parallel clock domains: an internal parallel clock domain (PMA parallel clock) and an external clock domain (user clock). To work and to be able to transmit/receive data properly, the rate of the two clocks must match and all phase differences between them must be resolved. For this purpose, the GTX TX includes an internal buffer and a phase-alignment circuit to resolve all the phase differences between the two clock domains [7]. There is a choice to be made for GTX transceivers applications: using an internal buffer or bypassing it and use a phase-alignment circuit. The internal buffer is robust, easy to use and the phase alignment process requires additional logic and additional constraints on clock sources, while using the phase-alignment circuit allows improving the link latency and reducing the number of registers [7]. We test both configurations, first using the internal buffer and the phase-alignment circuit, and both configurations work properly, as expected.

In order to allow the serial data to be used as parallel data, we configure the GTX transceiver to be able to align commas to byte boundaries. The RX includes a comma alignment circuit, which can be configured for this purpose. The transmitter was configured to send a recognizable sequence (comma). The receiver searches for the comma in the incoming data. When it finds it, it moves the comma to a byte boundary so the received parallel words match the transmitted parallel words [7]. In our implementation we use an idle word as the comma, which is always transmitted before any data.

### 5. Tests and Results

We test this design by connecting the transmitter and the receiver through an optical fiber in loopback configuration. A data packet acquired from the TileCal front-end electronics was used in this test. This data packet consists of around 297 words of 16 bits, which are sent sequentially with a period of 25 ns (40 MHz), providing a data rate of 800 Mb/s. The packet is stored in a RAM memory block and each word is read every 25 ns and is applied to the transmitter emulator to be transmitted. This data packet includes CRC fields which allow us to check the integrity of the data. The CRC module (VHDL) [3] is used to check the integrity of the data sent by the transmitter emulator.

We use the *Chipscope Analyzer* from Xilinx to monitor the signals of interest inside the FPGA, and to verify if the emulator pair is working properly (performing correct transmissions of the full data packet) by looking directly to the received data signal and by checking the CRC counters. When good data (as acquired from the front-end) is stored in the RAM, we expect no errors and the counters to have values of zero (correct transmission). When we corrupted the data packet stored in the RAM block, we were expecting that the CRC module is able to detect and count the errors.

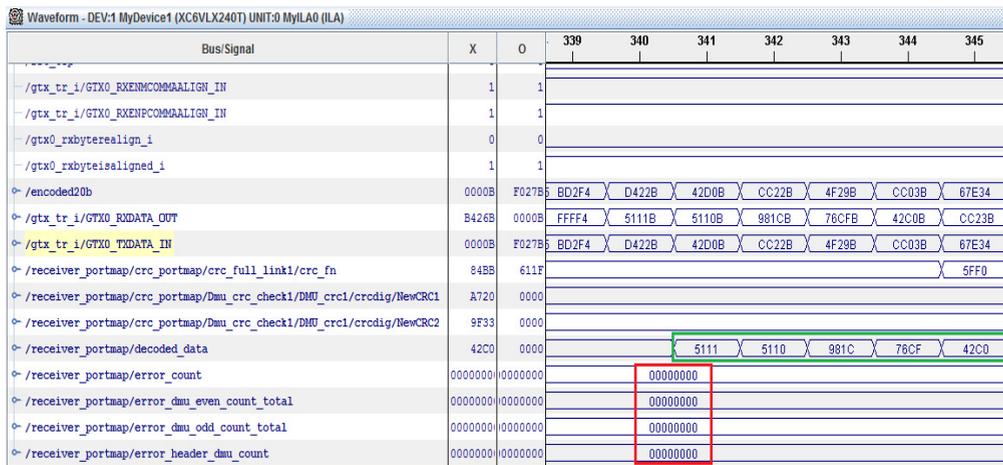


Fig. 6. Example of a correct transmission.

Fig. 6 shows an example when uncorrupted data is stored in RAM, and in this case the emulator pair performs a correct transmission and receipt of the data packet. We observe that parallel received words match the parallel transmitted words sent serially through an optical fiber. The CRC module checks and confirms this correct transmission/reception of the data packet and we can see that all counters remain at zero. We check that our design is working properly as expected. We also measure the link latency of our design, shown in table 1. From the G-link chip-set datasheet, the link latency of the original chip-set is around 4 clock cycles. Our design has a higher latency, introduced mainly by the GTX transceiver operations, but this is not critical issue for our application.

Table 1.Link latency.

GTX Configuration	Latency	
	Clock cycles	Period (ns)
Internal buffer	25	625
Phase-alignment circuit	15	375

From the data in the table, we conclude that we improve the latency using the phase-alignment circuit instead of the GTX internal buffer as described by Xilinx for GTX applications.

An identical solution is presented in [8] targeting a different application, where the emulation was performed in a Xilinx Virtex-5 FPGA around the GTP transceivers. The GTP transceivers functionality is similar to the GTX transceivers functionality used in this work. The difference lies in the fact that the GTX transceivers provide higher data link rate in comparison with the GTP transceivers. As in the solution presented [8], our solution is capable to perform correct transmissions/reception of data with a fixed latency (as the original chip-set does), although we have a higher latency of 15 clock cycles (375 ns) against of the latency of 12 clock cycles (312.5 ns) from the solution presented in [8].

## 6. Conclusion

This paper describes the emulation of a chip-set (HDMP-1032/1034), which can serve to build a high speed serial link. This chip-set is used as the physical layer in the ATLAS Tile Calorimeter electronics system, where the HDMP-1032 transmitter is implemented in the front-end electronics to serialize and transmit data to the HDMP-1034 receiver in the back-end which is used de-serialize the received data. The emulated HDMP-1034 receiver will be included in the ATLAS Tile Calorimeter front-end electronics tester (MobiDICK) implemented at CERN, and the emulated HDMP-1034 transmitter is planned to be used in the future upgrade of the Tile Calorimeter electronic system.

Our design was implemented in a Virtex-6 FPGA mounted in a ML605 evaluation board from Xilinx and tested by connecting the transmitter and the receiver in loopback mode through an optical fiber. In the test, we transmit a reference data packet (acquired from the TileCal front-end electronics) with CRC fields, and after the reception we check data integrity using the TileCal CRC module. We verify that our emulator is able to perform correct transmissions/reception of data with a fixed latency (as the original chip-set). So, by configuring properly the embedded GTX transceiver to meet our specifications and by using some logic resources in the FPGA, we could emulate the functionality of the original chipset.

## References

- [1] The Large Hadron Collider, the LHC Study Group, CERN/AC/95-05, 1995. <https://cds.cern.ch/record/291782/files/cm-p00047618.pdf>
- [2] Tile Calorimeter Technical Design Report, CERN/LHCC 96-42, 15-12-1996. [http://atlas.web.cern.ch/Atlas/SUB\\_DETECTORS/TILE/TDR/TDR.htm](http://atlas.web.cern.ch/Atlas/SUB_DETECTORS/TILE/TDR/TDR.htm)
- [3] P. Moreno, J. Alves, D. Calvet, F. Carrió, M. Crouau, K. HeeYeun, I. Minashvili, S. Nemecek, G. Qin, V. Schettino, C. Solans, G.Usai, A. Valero, "A new portable test bench for the ATLAS Tile Calorimeter front-end electronics", TWEPP 2012 Topical Workshop on Electronics for Particle Physics (Oxford), 2012 - Journal of Instrumentation, Vol. 8, Feb.2013 (P. Moreno et al 2013 JINST 8 C02046 doi:10.1088/1748-0221/8/02/C02046, <http://iopscience.iop.org/1748-0221/8/02/C02046>).
- [4] R. Bonnefoy, D. Calvet, R. Chadelas, M. Crouau, F. Martin, "MobiDICK: a mobile test bench for the TileCal super-drawer", ATLAS-TILECAL-2004-003, March 2004. <https://cds.cern.ch/record/681350/files/tilecal-2004-003.pdf>
- [5] J. Alves, J. Silva, G. Evans, J. Soares Augusto, "Communication Interfaces for a New Tester of ATLAS Tile Calorimeter Front-end Electronics" 9<sup>th</sup> Portuguese Meeting on Reconfigurable Systems, Institute of Systems and Robotics – University of Coimbra, February 2013.
- [6] Agilent HDMP1032/1034 Transmitter/Receiver Chipset Datasheet.
- [7] Xilinx, GTX Transceiver User Guide UG366 (v2.6) July 27, 2011.
- [8] A. Aloisi, F. Cevenini, R. Giordano, V. Izzo, "An FPGA-based Emulation of the G-Link Chip-Set for the ATLAS Level-1 Barrel Muon Trigger", Topical Workshop on Electronics for Particle Physics, Paris, France, September 2009. <http://indico.cern.ch/event/49682/session/30/contribution/80/material/paper/0.pdf>
- [9] Xilinx, LogiCORE IP Virtex-6 FPGA GTX Transceiver Wizard, UG516 (v1.8) December 14, 2010.