Università degli Studi di Napoli Federico II

DIPARTIMENTO DI FISICA "ETTORE PANCINI"

Dottorato di Ricerca in Fisica Fondamentale ed Applicata

XXVIII ciclo

2013-2016

PhD Thesis

Development of FPGA-based High-Speed serial links for High Energy Physics Experiments

Supervisors

Prof. Mariagrazia Alviggi

Dr. Raffaele Giordano

Dr. Vincenzo Izzo

Candidate

Sabrina Perrella

CONTENTS

1 INTRODUCTION

During this PhD work, I focused on the study of the state-of-art High-Speed serial links in reprogrammable electronics devices, specifically in Field Programmable Gate Arrays (FPGA). During this work, I describe and discuss the main properties of a serial link and the most important features of its architecture, when implemented in FPGA.
Data transmission technology over serial links, on copper cables or optical fibre, makes possible to transfer the data acquired by an experimental apparatus of large dimensions over a relatively small number of acquisition channels. In this way it is possible to achieve a high density in connections between the experimental apparatus and the acquisition environment, along with an almost total absence of noise, when using fibres.

In this PhD work, I developed three different high-speed serial links oriented to be used in High Energy Physics (HEP) experiments. High Energy Physics experiments generate high volumes of data which need to be transferred over long distance.
Due to their extreme high bandwidth, serial links have been extensively used in Trigger and Data Acquisition (TDAQ) systems of HEP experiments. For example, many experiments at Large Hadron Collider (LHC) at CERN, intensively employed serial links, both for transmission of Timing, Trigger and Control signals (TTC) [1] and for trigger and data readout [2, 3]. Their use in the framework of trigger systems is particularly interesting from the research point of view since, in these architectures, serial links have to guarantee a fixed latency. Originally, they were designed and deployed by using custom Application Specific Integrated Circuits (ASICs). However, recent studies [4, 5] show that last generation Field Programmable Gate Array devices (FPGAs) allow to implement the same links in cost-effective designs. In addition, being built around an FPGA device they represent a powerful solution in the replacement of out-of-the-art (but still used) links, whose commercial components has become obsolescent [**aloisio2010emulating:proc**, 6].

In this thesis will be presented the serial links I developed:

- a high-speed self-adapting serial link, which can be easily used in different application fields;
- the serial in/output interface of two electronic boards in the read out system of the muon spectrometer of ATLAS Experiment.

The frequency agile, auto-adaptive serial link, deployed in this work, is capable to analyse the incoming data stream, by scanning the Unit Interval, and to find the highest transmission line rate, according to a given failure rate. It uses a new feature (*RX eye margin analysis*) of the Receiver (RX) side of the Xilinx 7

series FPGAs high-speed transceivers (GTX), in order to measure and graphically display the quality of the link. The original contribution of this work consists in the build-up, design and optimization of a full architecture, which drives the GTX in order to modify the line rate of the link, runs consecutive eye scans for various line rate and analyses the results of the different scans, in order to find the maximum line rate sustainable by the link. The application can be deployed as a monitoring tool in HEP experiments, in order to remotely monitor a transmission system or detect issues in the serial link physical layer. Besides, it can be easily adapted in different frameworks, as it can be used on top of any user's existing link, as it has no specific requirement about link specification or protocol.

The other two serial interface developed in this PhD project are in the framework of the so called Phase-I upgrades of ATLAS experiment.
The main focus of Phase-I upgrade is on the Level-1 muon trigger, where upgrades are planned to be completed by 2019. In particular, for the end-cap region of the muon spectrometer, the installation of a new set of precision tracking and trigger detectors was approved (New Small Wheels). It will be instrumented with micro-mesh gaseous structure detectors (MM) and small-strip Thin Gap Chambers (sTGC). With their introduction, new electronics need to be developed, in particular new trigger electronics for both the MM and sTGC. I was involved in the development of serial interface of the Xilinx Kintex-7 FPGA-based sTGC trigger board that uses information from the coarse sTGC readout pads. Hence, in this thesis will be outlined my contribution to the design of the first prototype of the board and to the development of the serial interfaces with either the on-detector electronics board (for which fixed-latency constraints are required) and the monitoring and controlling chips.
Another Phase-I Level-1 trigger upgrade consists of a new Muon to Central Trigger Processor Interface (MUCTPI). The MUCTPI receives muon candidate information from each of the muon detectors, selects muon candidates and sends them to the Central Trigger Processor (CTP). In the first runs of ATLAS, the L1 Barrel trigger candidate data were transferred to the MuCTPI via copper cables. In order to cope with the trigger upgrade, serial optical links are necessary. The optical links will provide a much higher bandwidth (up to 6.4 Gbps) which will be used to transfer additional information from the off-detector trigger modules. I developed the new Interface board between the new MUCTPI and the Resistive Plate Chambers (RPC) muon trigger, using the Xilinx Artix-7 FPGA GTP transceivers. Then it will be illustrated the work done in designing the PCB board, for which particular attention was paid to the FPGA power supply and the clocking tree distribution (in order to allow the GTP FPGA internal transceiver to work correctly). Beside the fixed-latency link, developed for studying the feasibility of the new serial link will be described.

The fist chapter of this work provides an introduction to serial and parallel link architectures and to the working principles of the serializer-deserializer devices. Various examples of serial links used in HEP Experiments are presented.

In the second chapter, the FPGAs internal architecture is described as a background for the description of the implementation of the high-speed auto-adaptive serial link. Its internal architecture and the tests carried out are provided.

In the third chapter, after a brief introduction on the SM, the LHC complex and the ATLAS detector are described.

The four chapter covers the work done in the ATLAS Upgrades framework. In order to contextualize the project, an overview on the improvements foreseen for the Muon spectrometer is given. Hence, after an introduction to the Level-1 Muon Barrel Trigger system, the MUCTPI Interface board is introduced, and internal architecture of the new Interface board is illustrated. In the second part of the chapter, the purposes of NSW project are outlined. Finally, the pad trigger board, the interface logic implemented for it on FPGAs and the tests done are presented.

2

SERIAL LINK

High-speed serial links are nowadays widely used in many applications. For example, computer networks for office communications, serial buses, building and manufacturing automation, Internet and also ISDN. However, historically, for high-speed communication, the most used system was the parallel one.

Starting from the 90's, the decreasing cost of integrated circuits, combined with greater consumer demand for higher speed and longer distance, has led to the replacement of parallel links by serial ones: in hard disks, the Integrated Drive Electronics (IDE) [7] commonly known as AT Attachment with Packet Interface (ATA/ATAPI) [8] has been completely overcome with Serial AT Attachment (SATA) [9]; analogously IEEE-1284 [10] bi-directional parallel communications have been replaced by Universal Serial Bus (USB) [11] standard.
The driven factors which led to this change are strictly related to the features of this two kind of data transmission.

This chapter provides an introduction to serial and parallel link architecture and the State-of-the-Art of serial links of fundamental importance in order to have an outline of the thesis, which is mostly devoted to the development of a frequency agile, auto-adaptive serial link and high-speed, fixed-latency serial links for High Energy Physics (HEP) applications.

2.1 SERIAL VERSUS PARALLEL TRANSMISSION

Digital data transmission can occur in two basic modes: serial or parallel. The type of transmission mode used generally depends upon distance and required data rate.

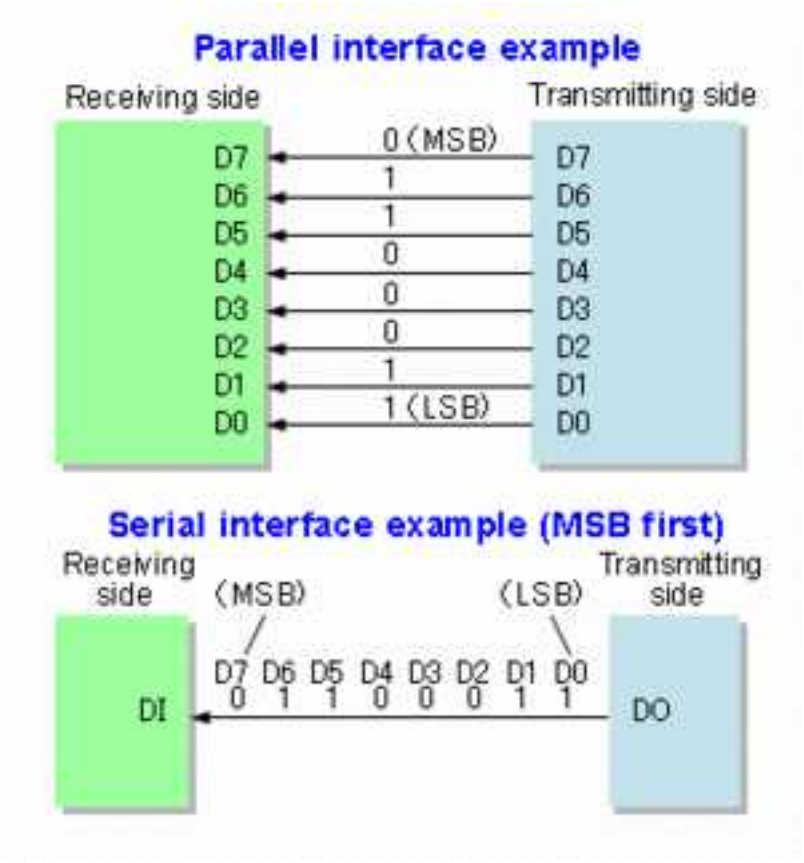


Figure 2.1: Data Transmission Methode.

In a serial transmission, data is sequentially transferred, one bit at a time, over a single transmission line. This means a simplified transmissive channel in spite of a more complex transmitter ad receiver devices required for synchronization and the flow control.

In parallel transmission, multiple bits (usually a byte or an its multiple) are simultaneously sent on different channels and synchronized to a clock.
In a parallel transmission, data can therefore be transferred in words of one or more bytes at a time. As a result, at a first glance, parallel bit rate results higher than the serial bit rate. However, this is not always true and, a further and deeper look into the features of serial transmission easily shows why. Indeed, in spite of this misconception, in many application, especially when data needs to be transferred over long distances, serial communication is preferred to parallel one. The reasons can be found in the following advantages:

1. In parallel links, since all bits of a data are transferred at the same instant, separate wires are used to carry each bit. Hence, the physical layer in the parallel links use many wires. On the other hand, serial links use a limited amount of wires. In the worse case, in order to guarantee a better transmission, a differential protocol is preferred, and only in this case a serial transmission uses two wires; otherwise a single wire is enough.

2. In parallel links the length of the connection can be an issue. When signal are carried over long distance, using cables, electromagnetic interference creates problems at higher data rates. Each wire acts as an antenna and capture a lot of noise from the environment: it corrupts data to be transferred. Since serial communication only use a limited number of wires, it can use long wires with less interferences. Since the space between two line can be greater than in a parallel link, the signal can be protected from electromagnetic interference by shielding the wires.

3. Besides, other factors limited the transmission rate in a parallel communications, that causes it to be sometimes slower than the serial one. As anticipated, a superficial comparison between a parallel and a serial link shows that, if both them operate at the same frequency, data transfer of a n-bit parallel link is n times higher than the serial link. The higher value of *throughput* [1] seems clearly in favour of a parallel transmission. However, in practical designs, the *clock skew* [2] among the parallel wires reduces the speed of data transmission to the slowest of all of the parallel transmission wires. Finally, the clock skew limits the speed of data transmission in a parallel link in respect to a single serial link.

Then, although a serial link may seem inferior to a parallel one, since it can transmit less data per clock cycle, it is often the case that serial links can be clocked considerably faster than parallel links so they can achieve a higher data rate.
Summarizing, the following factors allow serial link to be clocked at a higher rate:

1. clock skew between different channels is not an issue;

2. crosstalk is not a much significant issue, because there are fewer conductors in proximity and a better isolation of the single channels is allowed;

1 The *throughput* is one of the characteristic features of a transmission system. It is a measure of the effective used transmission capacity. The throughput must not be confused with channel capacity of the link. The latter is the maximum rate at which information can be reliably transmitted over a communications channel for any given degree of noise contamination (as defined by Shannon's theorem) while the throughput is the useful rate of the transferred data. Both the quantities are measured in bits per second (bits or bps)

2 *Clock skew* is defined as the variation in the clock commutation edges in function of different position of the components in the same integrate circuit (IC). An IC displays skew when clock length paths are not perfectly balanced. By definition, the skew value is the same for every clock cycle, for this reason it doesn't affect the value of the clock period, but it only determines a constant shift. In a synchronous digital system clock skew between two point i and j is defined as $t_{skew}(i;j) = t_i - t_j$; where t_i and t_j are the time distances between the real clock front end edge on the i and j component, respect to the ideal reference clock edge

3. in many cases, serial is a better option because it is cheaper to implement;
4. many ICs can have fewer pins and become less expensive.

The outcome of this conditions has made serial communication more and more often preferred over a parallel one and consequently more and more serial I/O methods and standards have been developed.

One of the fundamental characteristic of a transmission system are:

1. the data flow direction: if data flow in one direction, from the transmitter to the receiver (simplex) or in both simultaneously (full-duplex);
2. maximum data rate: it determines the capacity of a link, measuring the speed at which the data can be transmitted. The data rate is characterized by the number of bits transmitted per time unit and is measured in bits per second (bps).
3. the latency i.e. the amount of tame it take a bit to be transmitted from source to destination;
4. the number of extra data transmitted, in addition to the real data information, such as flow control signals, error signals and so on.

2.2 BINARY CODING OF DATA

Binary coding of data defines how the two binary values ('0' and '1') are distinguished. Indeed, in the transmission medium these value are assigned, according to the data format, to an *High* signal or to a *Low* signal. If the zero is represented by the *Low* signal while the one by the *High* signal, the logic is called positive on the contrary the logic is negative. Besides, more than one option exists on how the *High* and *Low* signals are define, according to different values and definition of the following variables:

1. amplitude values;
2. edges (level changes);
3. phase relationship;
4. frequencies

Among the most common data format there are: Non-Retun-to-Zero format (NRZ) and Return-to-Zero format (RZ), Manchester coding, 8B/10B encoding and scrambling/descrambling.

Depending on the particular application, a data format is preferred over the others. For example, in certain kind of transmission (synchronous data transmission) there is the necessity of transmitting clock from the sender to the receiver. This information can be transferred using an extra line or can be included in the data. When the latter is the case, only a coding which permit the receiver to recover the

clock rate from data flow, by ensuring a minimum number of transitions, can be used.

Another characteristic to take into account is the value of the DC component of the signal. In some application, indeed, DC component is forced to be zero. In order to reach a DC balancing, different types of coding can be used:

- 8B/10B encoding;
- scrambling/descrambling.

It could be more appropriate to refer to them as conversions, since they transform a data sequence of bits (also NRZ and RZ coding) into another one, according to fixed rules. The new sequence guarantees a DC-free serial signal and a minimum number of transitions which helps the clock recovery from data.

2.2.1 *8B/10B Encoding*

The 8B/10B Encoding [12] was developed by IBM and it converts each 8 bits of a sequence into 10 bits.
In order to achieve a DC-free signal, every 10-bit sequence should have the same number of ones and zeros, i.e. zero (or neutral) disparity. However, according to the 8B/10B Encoding process, some 8 it words can be converted in two possible sequence of 10 bits: one with two more 1 bits (disparity $= +2$) , and one with two more zeros (disparity $= -2$). In order to identify them the concept of positive and negative Running Disparity (RD) has been introduced. Then, depending on the running disparity of the current word, since the total number of ones and zeros is monitored, the next 10-bit character is chosen based on what is needed to bring the DC balance back in line.
In the conversion 8B/10B, 12 special characters are allowed. They are called control word or K-characters and are used for control function or for synchronization). For example, the 7-bit pattern '1100000' provides such a sequence that can only be found in three of the control characters. One of these three control characters can be chosen as a boundary for data, or comma as it is commonly called. Commas preceding data enable the receiver to correctly synchronize to the sequence of data words.
The inconvenience of this kind of coding is the introduction of an overhead of 25% not always compatible with the application since it reduces data rate .

Just for example, Fig. 2.2 shows a simplified version the 8B10B Encoding, 3B4B Encoding, which convert a 3-bit word in a 4-bit word.

2.2.2 *Conditional Inversion Master Transition protocol: CIMT*

The Conditional Inversion Master Transition (CIMT) protocol was introduced together with the G-Link chipset (see 2.8) in the Hewlett-Packard Journal in 1992 [13]. The coding scheme is designed to transmit a stream which is made of either 20 or 24 bit-wide data words. Each word contains 16 or 20 data bits

3B Input Data	4B Output Data
000	0011
001	0101
010	0110
011	1001
100	1010
101	1100

3B Input Data	4B Output Data	
	Even Words	Odd Words
110	0100	1011
111	0010	1101
SyncA	0111	1110
SyncB	1000	0001

Figure 2.2: Simple 3B/4B code example.

(D-Field) and 4 control bits (C-Field). The efficiency can be as high as 21/24 or 87.5% [14].
The DC balance of the frame is guaranteed by the transmitter which conditionally inverts words in such a way to minimize the bit disparity. On the receiver side, in order to restore the original information, an inversion is performed again, if necessary: the decision being made by reading the C-Field content.
The protocol guarantees a transition in a fixed position of the C-Field and the receiver checks for this transition in received data in order to to restore frequency and phase of the low frequency clock. Bit clock is reconstructed by dividing the frame clock by 20 or 24. This allows synchronization and word alignment avoiding the need for periodical interruption of the service in order to send synchronization characters. The four control bits have the additional function of flagging each word as a data word, a control word or an idle word. Idle words are used, at link start-up, in order to synchronize the link and to perform an initial handshake mechanism (which is able to notify the transmitter the receiver is locked on the stream). During the normal operation, instead, they help to keep the link phase-locked.

2.2.3 *Scrambling*

Another, coding technique is the bit scrambling which uses Linear Feedback Shift Register, LFSR[3] [15], to produce pseudo-random bit sequences. The benefit in using a scrambler is the possibility of rearrange the bits (the original sequence) in order to improve the quality of data communication. The LFSR is opportunely chosen sash as:

3 A Linear Feedback Shift Register is a sequential shift register whose input bit is a linear combinational function of its previous state. Commonly, the function is the exclusive-or (XOR) of single bits. The bit positions that affect the next state are called the taps. The choice of taps determines how many values there are in a given sequence before it repeats. Besides, once the initial value of the LFSR is chosen the stream is uniquely determined. However, an LFSR with a well-chosen feedback function can produce a sequence of bits which appears random and which has a very long cycle. The behaviour of the LFSR is well described by the finite fields theory also called Galois algebra.

1. a too long sequence of the same characters (zeros or ones), which complicates clock recovery on the receiver side, is overcome;
2. DC-balanced of the signal is improved.

Then, since transmitted data are converted at the transmitter side, in order to recover the information, at the receiver side, a descrambler has to be used. In Fig. 2.3 show the working scheme of a scrambler followed by a descrambler.

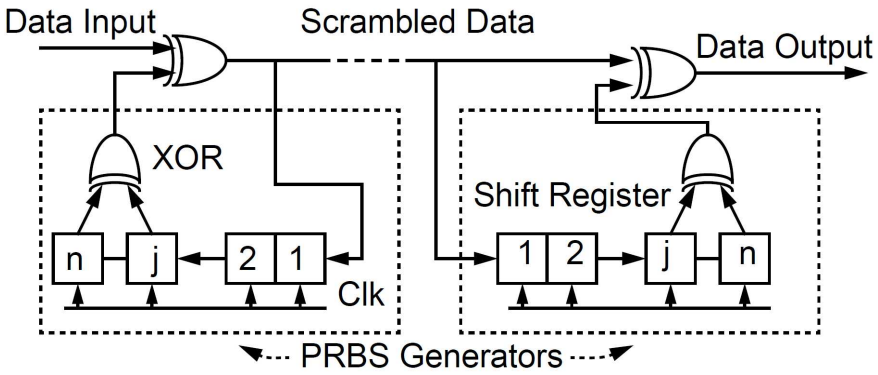


Figure 2.3: Scrambling. A feedback shift register is used to randomize data and in order to restore original data another register, which reverses the process, is used at receiver.

2.3 TRANSMISSION MEDIA

For serial data transmission, different transmission supports can be used, depending on the application. The signals can be transmitted either electrically, as light pulses or via radio waves. The choice of the medium is influenced by several factors:

1. costs and installation effort;
2. transmission safety: interference susceptibility, error probability, etc.;
3. maximum data rate;
4. distances among the devices to be connected.

2.3.1 *Electric lines*

Electric lines are simple and cost-effective, however, attenuation of signals and interference susceptibility (dependent on which kind of cable is used and on the interface specification) can be great disadvantages. The action of an electrical line on the transmitted signal is generally described by the equivalent circuit in Fig. 2.4. In this model an infinite series of sub-networks (made of resistors, capacitors, and inductors all expressed in their unit per unit length) come one after the other. The

resistors take into account the distributed resistance which cause the drops of the static signal level. The capacitors and inductors represent the distributed inductance and the capacitance C between the two conductors, respectively. Together with the resistor they create the filters which take into account the effects on the signal edge transitions.

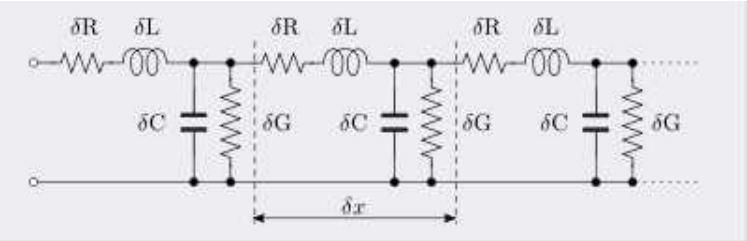


Figure 2.4: Equivalent circuit diagram of a transmission cable.

Examples of electrical cables are:

1. Conductor in a printed circuit board (PCB), Microstrip: a transmission line on a PCB is a thin flat conductor which is parallel to a ground plane, obtaining with a strip of copper on one side of a printed circuit board (PCB) or ceramic substrate while the other side is a continuous ground plane. The width of the strip, the thickness of the insulating layer (PCB or ceramic) and the dielectric constant of the insulating layer determine the characteristic impedance;

2. Coaxial cable: Coaxial cable is an unbalanced line where an inner conductor is surrounded by an insulating layer plus a braided conducting shield. However, on short distances, line rate can reach values of many Gbps.

3. Twisted pair: twisted pairs, like Ethernet LAN cable, are commonly used for telephonic communications. In such cables, many pairs are grouped together. The format is also used for data network distribution inside buildings, but the cable is more expensive because the transmission line parameters are tightly controlled. Their purposes is that of cancelling out electromagnetic interference (EMI) from external sources; for instance, electromagnetic radiation from unshielded twisted pair (UTP) cables, and crosstalk between close pairs. It was invented by Alexander Graham Bell;

2.3.2 *Fiber optics*

An optical fiber consists of a transparent core (usually made by drawing glass, silica) surrounded by another glass cladding with a lower index of refraction and an external plastic cladding (Fig. 2.5). The different value of refraction indexes is such that the light is confined to the core by total reflection. The reflections are almost free of any loss and thereby the light is conveyed within the core fiber only. The total diameter of a typical optical fiber is $0.1\,\mathrm{mm}$, however the internal

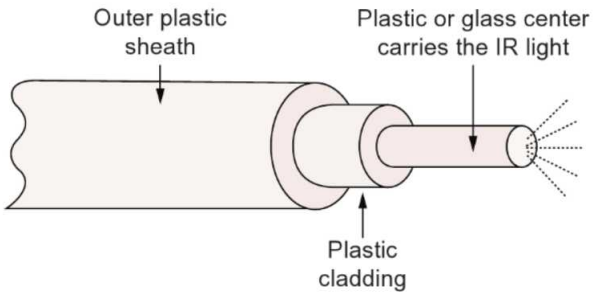


Figure 2.5: Structure of a an optical fiber cable.

core can vary between 5 µm and 60 µm. Since transmitted signal consists of on-off light pulses, on the optical transmission link ends (boundaries), devices converting electrical signals (serial binary data) are foreseen. Usually, optical carrier are generated by a LED on the transmitter side and converted back to electrical signal by photosensitive semiconductor on the receiver side. Due to optical loss dependency on the wavelength value, typical value of light wavelength are in the near infrared (IR), where the attenuation is at its minimum.
Fiber optic cables are widely used for transmission at extremely high data rates and over very long distances, since signals are resistant to electromagnetic interferences and only slightly attenuated.

2.4 CLOCK AND DATA RECOVERY

Independently from the used technique both transmitter and the receiver requires a synchronization clock signal.
In an asynchronous transmission, two separate clock source (one on the transmitter side and the other receiver side) running at the same frequencies are usually allowed. Indeed, when data flow is synchronized by start/stop bit flags, small frequency differences are not a big issue.
For a synchronous transmission, for which this situation has to be avoided, two methods exists in order to compensate a little clock frequency difference. If the use of a separate line, in addition to data line, is feasible, the transmitter can provide clock signal to the receiver. Otherwise, as anticipated, there is the possibility for the receiver of extracting the clock from data: the recovered clock. In this case, the technique used by receiver in order to generate the clock from data stream is called clock and data recovery (CDR). CDR algorithm needs a reference clock, whose frequency is approximately the same frequency of the transmitter clock. Hence, for clock recovery, the receiver phase aligns the reference clock to the transitions on the incoming data stream. Then, for data recovery, data on serial lines is sampled with the recovered clock.

The basic working scheme of a CDR is shown in Fig. 2.6: it includes a clock recovery and data retiming blocks.

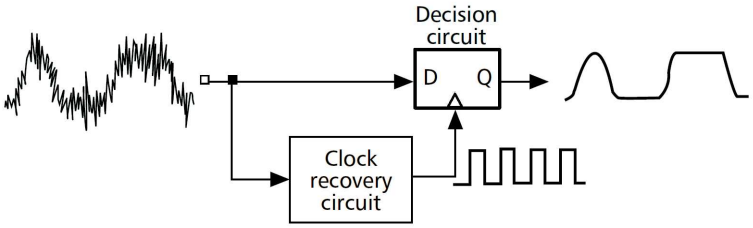


Figure 2.6: The role of a CDR circuit in retiming data.

The function of the clock recovery circuit is to detect the transitions in the received data and generate a periodic recovered clock. This recovered clock must satisfy the following conditions:

- its frequency must be equal to the input data rate;
- it should have reasonable timing with respect to the input data: the rising edge of the recovered clock should sample at the centre of the data bit, to provide maximum margin for jitter[4] and other time uncertainties;
- it should exhibit a minimum jitter because the jitter of the clock contributes to the retimed data jitter.

The data retiming circuit uses a Delay Flip Flop (DFF), which is triggered by the recovered clock to retime the received data. A Phase-Locked Loop (PLL) is

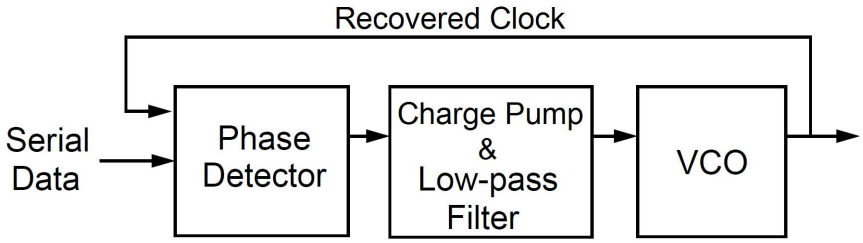


Figure 2.7: An example of CDR implementation.

usually used at the receiver to recover the clock from the data. A PLL is a negative feedback system where a clock generated by the Voltage Controlled Oscillator (VCO) is phase and frequency locked to an input data. The basic building blocks of PLL is shown in Fig. 2.7.

The function of the phase detector is to measure the phase difference between two incoming signals. One DFF with an XOR gate is a simple example of linear phase detector (PD), for which the output is linearly proportional to the phase

4 Jitter is defined as the amount of variation in the waveform from their ideal position at zero crossing on the time axis. It will be better analyze in the section 2.7.3

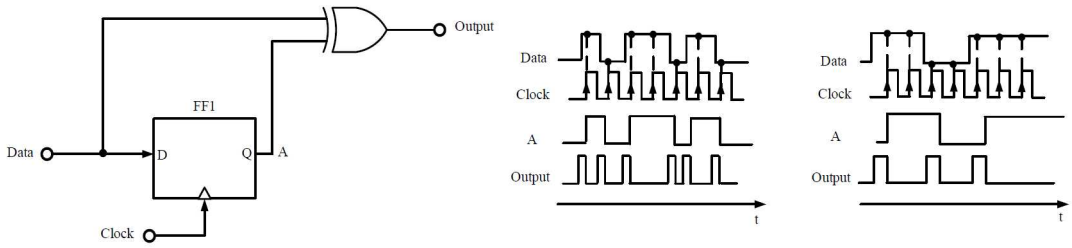


Figure 2.8: Basic linear phase detector (left) and output waveforms (rigth).

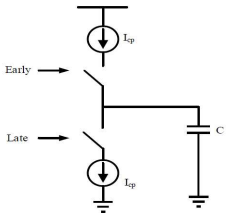


Figure 2.9: Charge pump in PLL.

difference, Fig. 2.8. A combination of two FF is able to determine whether the clock is earlier or later than the input data.

The function of the Charge Pump (CP) is to convert the output voltage of the phase detector to current. This current is then fed to a Low Pass Filter (LPF), where a capacitor is either charged or discharged depending on the phase detector output. The circuit diagram of the charge pump with a capacitor is shown in Fig. 2.9.

When the early node is high, closing the early switch, the capacitor starts charging and continues to charge until the early node goes low, opening the early switch. Similarly, when the late node goes high, the capacitor starts to discharge and will continue to discharge until the late node goes low. Designing a charge pump is not an easy task, because to achieve zero net voltage on the capacitor, the charging current should be equal to the discharging current. Even if the charging and discharging currents are designed to be close to equal, there will still be leakage current through the charge pump circuit, resulting in an offset voltage on the capacitor. One way to minimize this offset voltage is to calibrate the charge pump circuit by using a feedback loop circuitry. The function of the low pass filter (LPF) is to convert the charge pump current into control voltage. The function of the VCO is to generate the clock signal at its output, the frequency of which can be changed by varying the input control voltage. A signal, produced by the Phase detector when no frequency or phase difference occurs, force the LPF to fix its output DC value and consequently the frequency of the VCO signal. In order for this circuit to work well, transitions on the serial data must be enough frequent to guarantee the PLL to remain locked. For this purpose, often, a suitable cod-

ing method, which guarantees a minimum number of transitions of the signal, is chosen.

2.5 SERIAL TO PARALLEL AND PARALLEL TO SERIAL CONVERSION: SERDES

The increasing requests for reliable high speed transmission over long distances has often made serial link preferred over parallel ones. However, since data elaboration is often parallel, data needs to be serialized on the transmitter side and deserialized on the receiver side. Circuit specialized for doing this conversion has been designed and are constantly developing; they are called SerDes. The

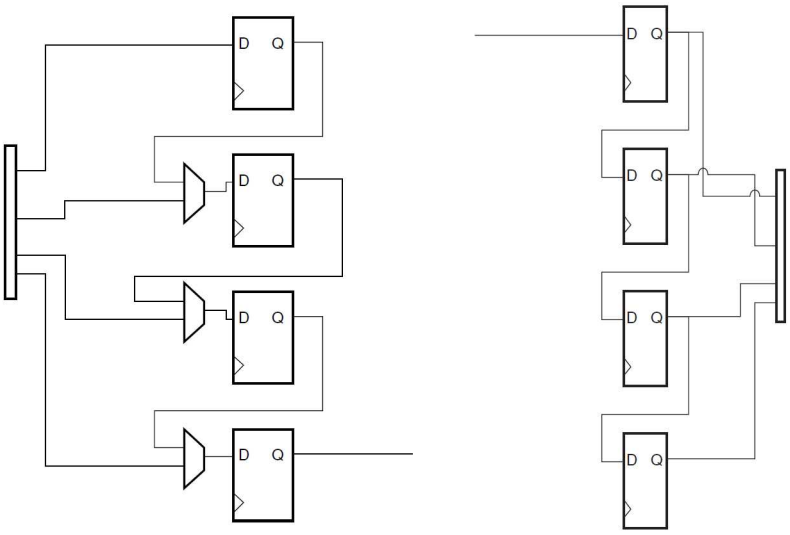


Figure 2.10: Working principles of a PLL in a CDR circuit.

basic working principle of a SerDes is relatively simple (Fig. 2.10). A parallel data is clocked into the parallel register synchronously to a slow clock and shift out, one bit at a time, at a higher serial clock rate. This block is called PISO (Parallel Input to Serial Output). The reverse operation is carried out by the SIPO (Serial Input to Parallel Output) block. It is simply composed by two registers. A shift register which receives data one bit after the other and transmits its contents to a storage register used to hold the data for the slower, parallel clock. The shift register needs to be clocked on the high serial clock (in such application the recovered clock form CDR circuit). Clearly, this scheme can work well only if the transmitter and receiver clocks are perfectly synchronized.

2.6 EQUALIZATION

Since the transmission medium generally acts like a low pass filter, the signal at the receiver side of a link can be significantly distorted. Equalization is the process used to compensate signal degradations at the transmitter and/or at the

receiver. It works in such a way as to distort the signal with the only aim of ensuring a proper decoding.

2.7 LINK PERFORMANCE: BIT ERROR RATIO TEST AND JITTER TOLERANCE

In order to evaluate the data integrity at the end of the link, the most important measure is called the Bit Error Ratio, or BER.

2.7.1 *BER*

The BER can be defined as the estimate of error probability, e.g. the probability that any bit transmitted through the system will be erroneously received. In practical tests, the BER is measured by transmitting a finite number of bits through the system and counting the number of errors received. The ratio between the number of bits received in error and the total number of bits transmitted is the BER.

$$\mathrm{BER} = \frac{\text{Errors}}{\text{Total Number of Bits}}. \tag{2.1}$$

The quality of the BER estimation increases as the total number of transmitted bits increases untill, in the limit of infinity transmitted bits, the BER becomes a perfect estimate of the *true* error probability $P(E)$. However, in practical tests, finite testing times are required and then a less than perfect BER estimation must be accepted. Since the BER is a statistical average, it is possible to introduce the BER confidence level (CL) in order to quantify the quality of the estimate. CL is defined as the probability, based on E detected errors out of N transmitted bits, that the *true* BER (BER) would be less than a specified ratio, $\mathrm{BER}_{\mathrm{limit}}$:

$$\mathrm{CL} = P(\mathrm{BER} < \mathrm{BER}_{\mathrm{limit}}), \quad \text{given E and N.} \tag{2.2}$$

In this discussion, the *true* BER is the BER that would be measured if the number of transmitted bits was infinite. The formula tells, once the $\mathrm{BER}_{\mathrm{limit}}$ and CL are fixed, if after N transmitted bits, E errors are found, there is a CL percent confidence that the true BER is less than $\mathrm{BER}_{\mathrm{limit}}$. Statistical methods, involving the binomial distribution function and Poisson theorem, help to calculate how many bits need to be transmitted in order to have the desired CL:

$$N = \frac{1}{\mathrm{BER}_{\mathrm{limit}}}\left(-\ln(1-\mathrm{CL}) + \ln\left(\sum_{k=1}^{E}\frac{(N * \mathrm{BER}_{\mathrm{limit}})^k}{k!}\right)\right), \tag{2.3}$$

In the equation $\ln()$ is the natural logarithm, while the other variables represent the quantities previously introduced. The details of the calculations and derivations are known in literature [16] and are beyond the scope of this work.
To better understand how the equation (2.3) [23] can be used, an example can be explicative in which it must be determined how many bits must be transmitted in order to have a 95% CL that the *true* BER is less than 10^{-10}, if no error has been

N * BER			
Errors	CL = 90%	CL = 95%	CL = 99%
0	2.30	3.00	4.61
1	3.89	4.74	6.64
2	5.32	6.30	8.40

Table 2.1: N * BER for confidence levels of 90%, 95%, and 99%.

detected. In this simple example, $BER_{limit} = 10^{-10}$, CL = 0.95 and E = 0, so the second term of the equation is null. The result is

$$N = \frac{1}{10^{-10}} \left(-\ln(1-0.95)\right) \approx 3/10^{-10} = 3 * 10^{10} \tag{2.4}$$

or defining the normalized number of bits as $NxBER_{limit}$

$$N * BER_{limit} = \left(-\ln(1-0.95)\right) \approx 3 \tag{2.5}$$

Table 2.1 shows the normalized number of bits that must be transmitted, N x BER, in order to get a BER estimate with different values of CL (90%, 95% and 99%) and for 0, 1 or 2 transmission errors. By dividing the values in the table by the desired BER, it is possible to obtain the number of bits to be transmitted for a specific confidence level and bit error.

2.7.2 *Equipment and Procedures*

The conventional method for BER testing utilizes a pattern generator and an error detector (2.11). The pattern generator transmits the test pattern into the system, while the error detector counts the number of bit received with errors. For checking the correctness of the data it usually replicates the test pattern of the generator and performs a bitwise comparison between this data and the data received from the link. In order to accurately compare the bits received from the pattern generator with the bits received from the system under test, the error detector must be synchronized to both bit streams and it must compensate for the time delay through the system under test. Test results depend on the particular

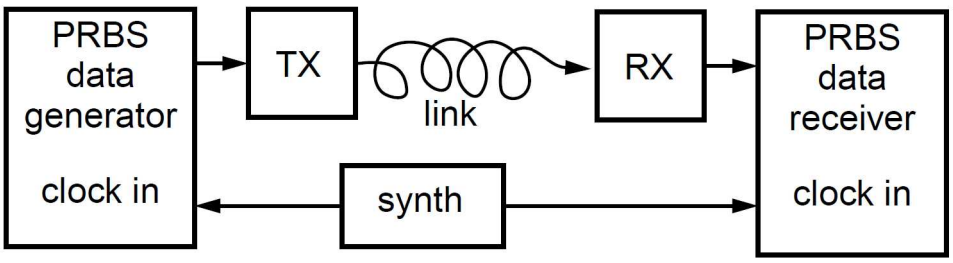


Figure 2.11: Typical BER Test set-up.

test pattern (bit sequences) sent into a system, different rate of bit errors may

occur. Usually, a particular stressful test pattern is chosen, which is intended to emulate random data like Pseudo-Random Bit Sequences (PRBSs), which are based on Galois' Algebra. In some cases, however, a replica of the kind of data that is expected to occur during normal operation is more useful.

2.7.3 *Jitter*

Together with BER, jitter is an increasingly important quantity in the development and specification of serial data links. Indeed, as clock speeds and communication channels run at ever higher frequencies, the analog nature of digital signals (binary data is no more then a digital waveform) reveals itself and jitter becomes critical in designing digital synchronous systems.
Jitter can be defined in different ways [17]: from an analog point of view, jitter can be seen as phase noise [18] while in a digital communications channel, it is the time deviation from ideal timing of a signal transition through a decision threshold or more simply, how early or late a signal transitions with reference to when it should transition. As a consequence, there are different domains of analysis: the time domain (jitter amplitude versus time), the modulation domain (phase versus time), and the frequency domain (amplitude of jitter components versus their frequency, or analysis of phase modulation sidebands). Jitter measurements can be quantified in the same fashion as all time varying signals using either RMS or peak-to-peak displacement in the time domain. Alternatively it can be also quantify, in the frequency domain, in terms of power spectral density over a given bandwidth.
Jitter impacts the maximum frequency of a synchronous system (reducing the timing budget for basic operations), as well as the quality of a transmission (data errors can result from sufficiently large time deviations from ideal). Any part of a system which generates, transmits or receives signals, can be source of jitter [19, 20]. As a result, in order to determine the total performances of a system it is of primary importance to evaluate how much jitter every elements of the system introduces. In this scheme it has become fundamental to characterize, model, simulate and estimate all the jitter sources.
Digital definition of jitter allows many ways of representing measuring it [21–23] because of the ambiguity in the choice of the reference, ideal transition: it can be the transition through a decision threshold of a reference or reconstructed clock, but arithmetic quantity can be used too, like the nominal bit period ad a given frequency. The following are the major types of jitter:

1. Long Term Jitter: it measures the change in a clock's output from the ideal position, over several consecutive cycles. It represents the cumulative effect of jitter on continuous clock cycles over a long time interval.

2. Absolute jitter: it is Root Means Squared (RMS) evaluation of many edge measurements from an expected location or absolute time reference;

3. Period Jitter: it is the deviation of a clock signal with respect to the ideal period over a number of randomly selected cycles;

4. Cycle to Cycle Period Jitter: it measure the variation of a signal between consecutive cycles, over a random sample of adjacent cycle pairs. Its peak value defines the maximum difference between two rising edge of any two consecutive clocks and it can be expressed in ps as well as in RMS;

5. Time Interval Error (TIE): it is the time deviation of an edge of the signal from a reference point position which is obtained using the actual transmitter clock or a reconstruction of it from the data sampled. In effect, TIE is the discrete time domain representation of phase noise expressed in seconds;

6. Phase Noise Jitter: it is described as either a set of noise values at different frequency ranges, or as a continuous noise plot over a unique range of frequencies. Usually, in a square wave, most of the energies are located at the frequency carrier (f_c). However, some signal energies are significant also over a range of frequencies on both sides of the carrier (since the signal is assumed to be symmetric). Phase jitter is the integration of phase noises over a certain spectrum and expressed in seconds or RMS.

Types of Jitter

Conventionally two general categories of jitter are defined: deterministic jitter (DJ) and random jitter (RJ).
DJ is often bounded, periodic, non-Gaussian, and has a specific cause. It can be further classified into periodic jitter and data-dependent jitter, as Fig. 2.12 shows. Examples of deterministic jitter are:

1. Duty-Cycle Distortion (DCD): from asymmetric risefall times;

2. Intersymbol interference (ISI): from channel dispersion or filtering;

3. Sinusoidal: from power-supply switching;

4. Uncorrelated: from crosstalk by other signals.

RJ is Gaussian and it is often referred to as background or thermal noise, although it can easily be the most significant contribution to total jitter. Because random jitter is modelled as a Gaussian distribution, its unbounded: the probability of some values is very small, but not zero. In the light of the foregoing, the total jitter is the result of both random component and deterministic one, but it is not the simple sum of them. Indeed, while the RJ has a Gaussian distribution and its peak-to-peak value has no sense, for the DJ the opposite is true. Then, if in the phase domain, the total phase error function can be written as:

$$\phi_j(t) = \phi_j(t)^R * \phi_j(t)^D, \tag{2.6}$$

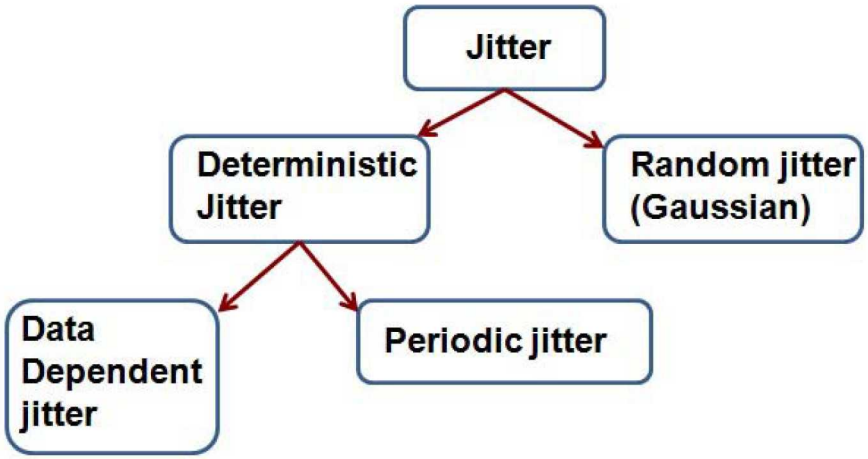


Figure 2.12: Types of Jitter.

where $\phi_j(t)^R$ is the random component and the $\phi_j(t)^D$, the deterministic one, for the evaluation of the performances of a system it is more useful to speak of jitter in terms of Probability Density Function, PDF (the function which gives the probability of a logic transition occurring at a given time, x). Then a measurement of the jitter PDF can be performed measuring, separately, each component of the jitter distribution, including RJ and all types of DJ. Indeed, assuming RJ e DJ are independent, the PDF of the total jitter, TJ(x) is the convolution of the single terms

$$TJ(x) = RJ(x) * DJ(x). \tag{2.7}$$

However, in this approach it is necessary to know the PDF of each components. For the random jitter it is relatively simple since the PDF is a Gaussian function defined by the formula:

$$PDF_{RJ}(x) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{x^2}{2\sigma^2}\right) \tag{2.8}$$

Then, from a histogram with the appropriate value of samples, it is possible to estimate the values of μ and σ and from this, the expected RJ value. It is worth to remember that in a Gaussian distribution the peak-to-peak value is infinite, since the distribution is no limited. However, once the number of samples or the BER is fixed, the jitter of a signal whose only aggression is of random kind, can be defined as [24]:

$$J^R_{RMS} = N_{BER}\ \sigma, \tag{2.9}$$

with N_{BER} depending on the agreed upon BER [25].
For the deterministic component, J^D_{pp}, the situation is more complicate: a separation and identification of the single causes of jitter it is necessary, but not always possible. The study of how the single component influence the digital system is important, however it is not generally possible to separate a single attack from another one. On the other hand, it is usually possible and simpler to measure the total jitter in which all the contributions are superimposed. This is the reason

why, together with the decomposition model, which pretend to obtain an estimate of the total jitter starting from its single component, other models have been developed [26]. Among these, the Dual-Dirac Model [27] is the most universal accepted.

Dual-Dirac model

The dual-Dirac model is based on the following hypothesis:

1. The total jitter can be decomposed in RJ and DJ;
2. RJ is completely described in terms of the width of the Gaussian function 2.8, σ;
3. DJ is limited;
4. DJ is the simple sum of two Dirac delta functions;
5. jitter is a stationary phenomenon.

According to this model, the dual-Dirac PDF for DJ is the simple sum of two Dirac delta functions, one centred at μ_L and one at μ_R,

$$\mathrm{PDF}_{dual-Direc}\mathrm{DJ}(x) = \delta(x-\mu_L) + \delta(x-\mu_R). \tag{2.10}$$

If, like the model suppose, the distributions of different components of jitter that are independent, like RJ and DJ, it is possible to combine their PDF through convolution. Since, the PDF for RJ is well described by 2.8, for the total distribution is:

$$\begin{aligned}
\mathrm{PDF}(x) &= \mathrm{PDF}_{RJ}(x) * \mathrm{PDF}_{dual-Direc}\mathrm{DJ}(x) \\
&= \int [\mathrm{DF}_{dual-Direc}\mathrm{DJ}(u) * \mathrm{PDF}_{RJ}(x-u)]\, du \\
&= \frac{1}{\sqrt{2\pi\sigma}} \int [\delta(u-\mu_L) + \delta(u-\mu_R)] \exp -\frac{(x-u)^2}{2\sigma^2}\, du \\
&= \frac{1}{\sqrt{2\pi\sigma}} \left[\exp\left(-\frac{(x-\mu_L)^2}{2\sigma^2}\right) + \exp\left(-\frac{(x-\mu_L)^2}{2\sigma^2}\right)\right].
\end{aligned} \tag{2.11}$$

Once we have the jitter PDF, BER(x) can be calculated as the probability that an error will occur if the sampling point is positioned at the time-delay position, x. Then, BER(x) is given by

$$\mathrm{BER}(x) = \rho_T \int_x^{+\infty} \mathrm{PDF}(u)\, du + \rho_T \int_{-\infty}^{x} \mathrm{PDF}(u-T)\, du \tag{2.12}$$

where ρ_T is the ratio of the number of transitions to the number of bits (logic transition density). Plugging the dual-Dirac model, Eq. 2.11 into Eq. 2.12, get:

$$\mathrm{TJ}(\mathrm{BER}) = 2\, Q_{BER}\, J^R_{\delta-\delta} + J^D_{\delta-\delta}, \tag{2.13}$$

BER	Q_{BER}
10^{-10}	6.35
10^{-11}	6.70
10^{-12}	7.05
10^{-13}	7.35
10^{-14}	7.65

Table 2.2: Minimum values of Q_{BER} for different BERs

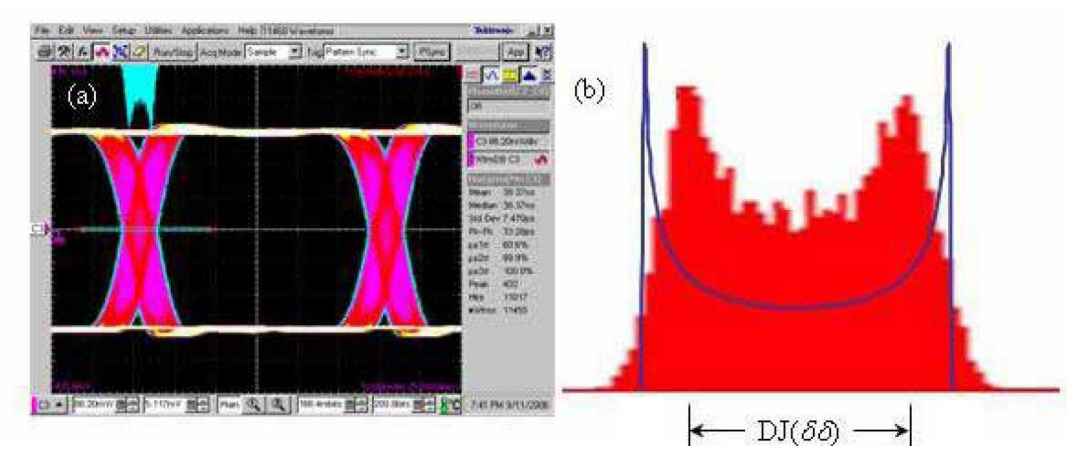


Figure 2.13: An eye diagram (left) and the corresponding TJ histogram (right).

where, Q_{BER} is a factor that relates the BER to the distance from the centre of the Gaussian, and $J^{D}_{\delta\delta}$ is a model dependent parameter that is easy to define and can be measured in many different ways. Q_{BER} is listed in Table 2.2 for different BER values. In conclusion the dual-Dirac model provides an easy way to estimate $TJ(BER)$ by measuring the RJ, which is given by σ, and DJ, J^{D}_{pp}, which is given by the separation of the two Dirac-delta functions, $\mu_R - \mu_L$:

$$TJ(BER) = 2\,Q_{BER}\,\sigma + (\mu_R - \mu_L). \tag{2.14}$$

Measuring jitter

Measurements of jitter can be obtained using an oscilloscope capable of providing an Eye Diagram of the signal, which is a composite view of all the bit periods of a captured waveform superimposed upon each other and of making a histogram of the crossing point (Fig. 2.13). A bathtub plot (bottom of Fig. 2.14) can be used too, since it represents a measurement of the BER as a function of the time-delay position of the sampling point, BER(x). In this case, $TJ(BER)$ is the amount of eye closure at a given BER, that is the nominal bit period, T_B, less the eye opening at that BER (the distance between the two sloping curves in Fig. 2.14) at the desired BER:

$$TJ(BER) = T_B - \text{eye opening}(BER). \tag{2.15}$$

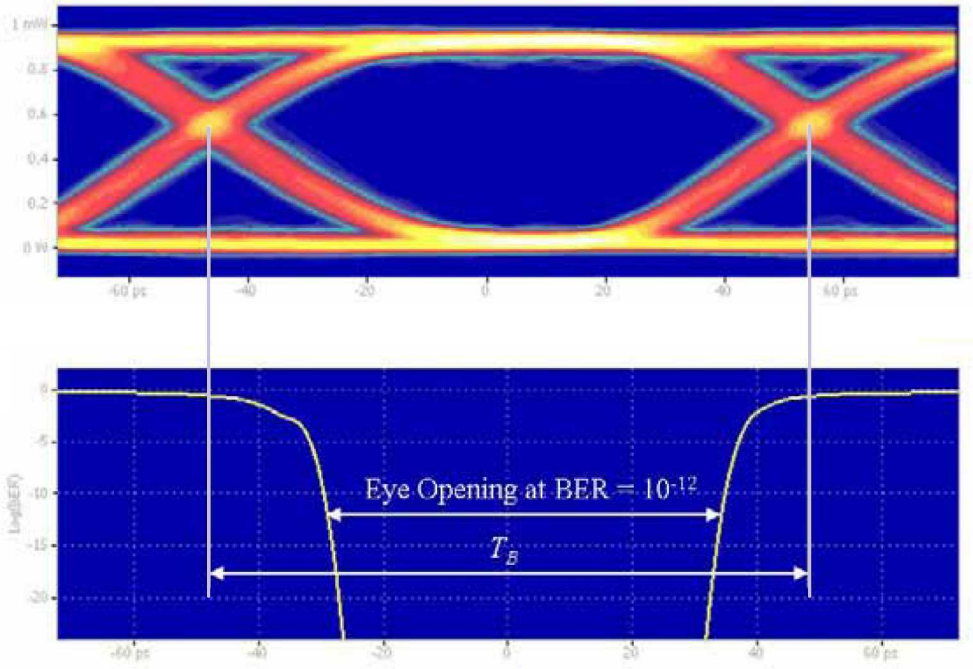


Figure 2.14: An eye diagram (top) and the corresponding bathtub-plot: The bit error ratio as a function of the time-position of the sampling point with respect to the ideal logic transition time, BER(x).

From this discussion, it could be seem very simple to measure TJ(BER): it is only necessary to do a measure of BER(x), as in Fig. 2.13 and 2.14. Another technique will be describe in the chapter 3.2.3. However, in all the cases, measuring BER(x), with a fixed CL, which gets sufficient statistic to reference eyes closure to BER, can take a long time, as previously discussed.

2.8 LEGACY HIGH-SPEED SERIAL LINKS IN HEP: G-LINK AND S-LINK

Current Field Programmable Gate Arrays devices, FPGAs[5],offer integrated digital transceivers with very high data rates which has lead to the implementation of many high speed serial link, based on the FPGA technology. The application field are wide: portable devices, medical devices, wireless, wire line and military markets [28, 29] as well in scientific research (Applications of Field-Programmable Gate Arrays in Scientific Research [30], Compressed Baryonic Matter experiment [31], the Positron Emission Tomography system [32], and the future KM3NeT undersea network [33]). Data acquisition systems of HEP experiments have been often based on serial link. In this section, two serial link protocols will be described: G-Link and S-Link. Both of them show how an FPGA-embedded MGTs based solution can be even more suitable than an implemen-

5 A field-programmable gate array is an integrated circuit designed to be configured by a customer or a designer after manufacturing, hence *field-programmable*. It will be better described in the section 3.2

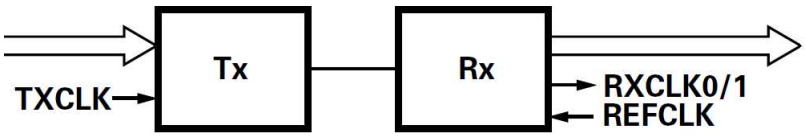


Figure 2.15: G-Link chipset.

tation based on commercial MGTs. Such an implementation takes advantage of the re-programmability of the FPGAs as well as of their embedded MGT devices, giving the user the possibility of increasing the performances of the link in terms of bandwidth, BER, or adding features like the fixed latency.

The G-Link chip-set

The G-Link chip-set, for instance, has been deployed in the Alice [34], ATLAS [35], Babar [36], CDF [37], CMS [38] and Nemo [39] experiments and due to its popularity CERN produced a radiation hard serializer compatible with it [40]. The original Agilent GLink chip-set (Fig. 2.15) consists of a transmitter (HDMP-1032A) and a receiver (HDMP-1034A). They are used together to implement a high-speed data link for point-to-point communication with line rates up to 1 Gbps. The chip uses the CIMT scheme (2.2.2) to encode data and to ensure the DC balance of the link. In order to read serial data, the receiver extracts a clock from the CIMT stream and locks its phase to the master transition. The recovered clock synchronizes all the internal operations of the receiver and it is available as an output named RXCLK0/1 (Fig. 2.16). Received data, RX(0:15), is transmitted synchronously with the recovered clock. One of the strong points of the chip-set architecture relies in the deterministic latency of the entire link. Moreover, the receiver structure is such that, thanks to a dedicated Parallel Automatic Synchronization System (PASS), it is also capable of synchronizing the output data with a local receiver clock, while guaranteeing a constant phase relationship with the transmission clock.

A Simple Link Interface: the S-LINK

S-LINK [41] is a custom and high-performance data transmission protocol developed at CERN, which has been used in some of the data acquisition systems of the LHC experiments (Alice [34], ATLAS [35], CMS [38]).
S-LINK is based on a simple FIFO-like user interface, which remains independent of the technology used to implement the physical layer. Indeed, several versions of S-LINK have been built that using optical fibers and electrical cables and that use different components driving the links. For examples the S-LINK transmitter implementation, High-speed Optical Link for ATLAS or HOLA, and the S-Link64 (a 64-bits version of the protocol) have been extensively used in ATLAS and CMS, respectively. In particular, HOLA, used the general purpose SerDes TLK2501 by

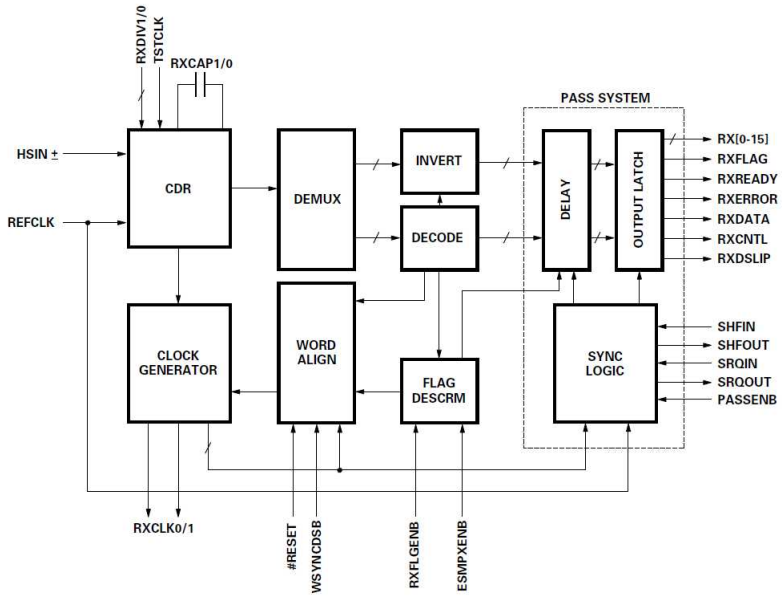


Figure 2.16: Block Scheme of the Receiver chip in G-Link.

Texas Instruments.
The specification [42] describes the interface between the Front-end Motherboard (FEMB) and the Link Source Card (LSC) and the interface between the Link Destination Card (LDC) and the Read-out Motherboard (ROMB). The S-LINK data format is different for the transmitting lines on the LSC and on the LCD (the return channel). However, for both of them, a 32-bits bus of data is serially sent from the FEMB, UD(31..0), and from ROMB, LD(31..0). In parallel to data extra lines, UCTRL and LCTRL(1:0), are used for the handshaking mechanism and for synchronization.
In addition to simple data movement, S-LINK includes: error detection, a self-test function and a return channel for transmitting flow control commands.
The S-LINK concept is shown in the Fig. 2.17.

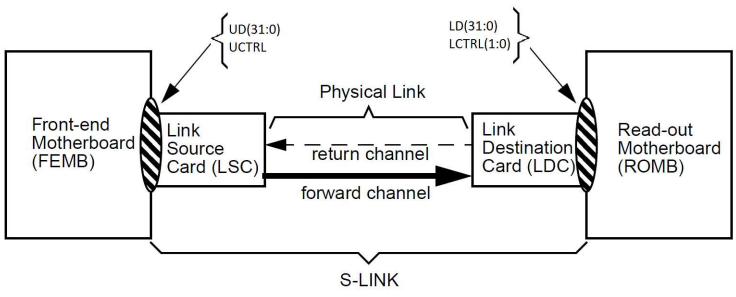


Figure 2.17: S-LINK Concept and data format.

Unfortunately, these serial links suffered the fact that a few years ago the production of the G-LINK chip-set and the TLK2501 Serdes were discontinued. In order to find replacements for them, alternative solutions were investigated: in

both the cases, FPGA embedded solutions were explored [6, 40]. The new architectures were based on last-generation Xilinx FPGAs, with high-speed serial transceivers in the multi Gbps domain. Their success relies in the characteristics inherited from FPGAs, in terms of re-programmability, portability and versatility. Indeed, specific implementation flavors target different requirements: increasing the bandwidth performances of the link, keeping it backward compatible with the previous ASIC version, or adding fixed-latency for real-rime critical operations.

2.9 STATE-OF-ART OF HIGH SPEED DATA LINKS IN HEP

As the demand for bandwidth and speed increases, not only in the telecommunication but in every field of electronics industry (military, medical, networking, video, etc.), a constant improvements in I/O design has demanded. Multi-gigabit serial I/O are the dominant implementation of I/O interfaces at speeds of 1 Gbps and higher. Multi-Gigabit Transceivers (MGTs) are an improved version of the simple SerDes described in the previous sections. Indeed, beyond the serialization and deserialization functions, Multi-Gigabit Transceivers incorporate additional technologies, such as:

- Differential signalling,
- Phase-locked loops,
- Clock and Data Recovery,
- Emphasis,
- Encoding/Decoding,
- Channel bonding.

Most of this functions has been described in the previous section, and it is clear the fundamental (critical) role they play in data link designs, when operations at high line rate has to be guarantee. Due to their intrinsic capacities, MGTs are increasingly being used for communications because they provide extremely fast, efficient and reliable communication data links. These are the reason why many industrial standards such as:

- FiberChannel (FC);
- PCI Express;
- RapidIO Serial;
- Serial ATA;
- 1-Gb Ethernet and 10-Gb Ethernet (XAUI);
- Infiniband,

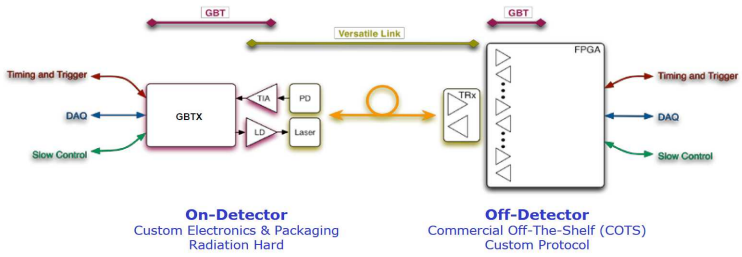


Figure 2.18: Radiation-hard optical link architecture.

use multi-gigabit SerDes. Regarding the implementation of MGTs for a particular application, a number of possibilities are available, using:

1. an ASIC;
2. an off-the-shelf IC (Application Specific Standard Product or ASSP), eventually with an external FPGA;
3. an FPGA with its embedded MGT.

Usually, the key factor to take into account (is tempered by several factors) for the decision are: cost, volume, form-factor and flexibility.
ASICs offer the most flexibility for power optimization, packaging and volume cost reduction, but come with prohibitive up-front, non-recurring engineering expenses.
ASSPs have been a popular choice for designs, especially when there is either no additional logic is needed or it is small enough that it can be implemented into a small external FPGA. This solution is not suitable and unattractive for the increase in the device's physical size and absorbed power, as well as for the cost of a two-chip solution. Besides, for most of the actual high-speed serial link, it has become rather common the request of adding a complex extra logic. Then FPGA vendors presented FPGA equipped with re-programmable MGTs. Serial links completely FPGA-based offer a single-chip solution, similar to ASSPs, with all the advantages related to the re-programmabilty of an FPGA devices: flexibility, timing and testability.
Finally, depending on the case one among the previous solutions prevailed over the other. Examples of different implementations of serial links will be described which are of special importance for High Energy Physic Experiments (HEP).

2.9.1 *An ASIC SerDes: the GBT*

The GigaBit Transceiver (GBT) architecture and transmission protocol has been proposed for data transmission in the physics experiments of the upgrade of the LHC accelerator, SLHC. Due to the high beam luminosity planned for the SLHC, the experiments will require high data rate links and electronic components capable of sustaining high radiation doses. The GBT ASICs is a radiation-hard bi-directional 4.8 Gbps optical fibre link for the transmission between the counting

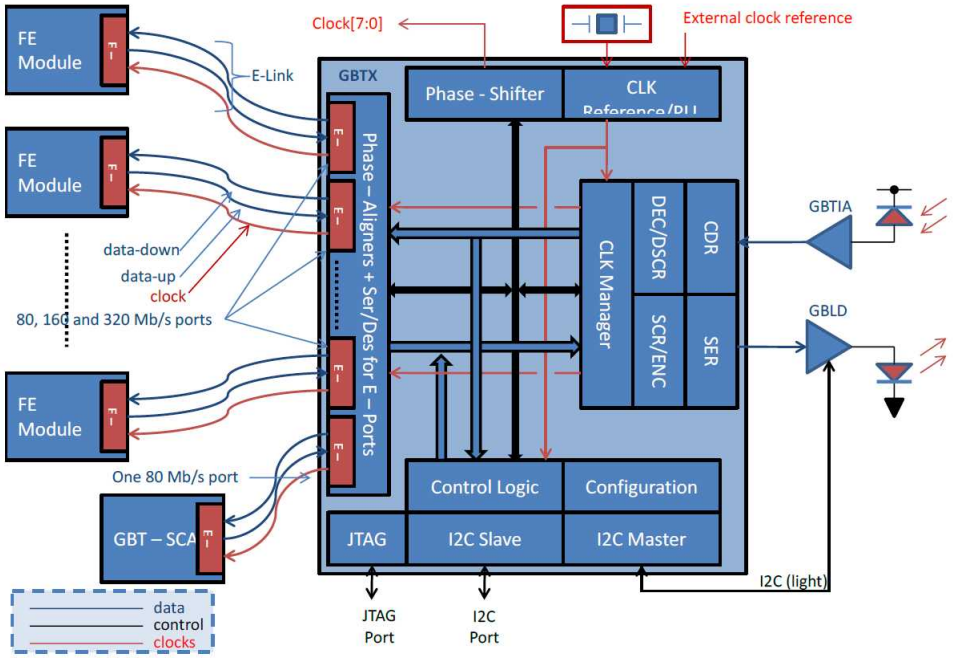


Figure 2.19: A simplified block diagram of the GBTx.

room and the experiment. The basic architecture of the GBT protocol is shown in Fig. 2.18.
One half of the system resides on the detector and hence in a radiation environment, therefore requiring custom electronics. It consists of the following components: GBTx, a serializer-de-serializer chip for data; GBTIA: a trans-impedance amplifier receiving the 4.8 Gbps serial input data from a photodiode; GBLD: a laser-driver ASIC to modulate 4.8 Gbps serial data on a laser; GBT-SCA: a chip to provide the slow-controls interface to the front-end electronics.
The other half of GBT is off-detector then, it is free from radiation and can use commercially-available components: consists of a Field-Programmable-Gate-Array (FPGA), programmed to be compatible with the GBT protocol and to provide the interface to off-detector systems. A parallel project, the Versatile Link project, has been started in order to produce a system of opto-electronics components for the optical data transmission.

In Fig. 2.19 a simplified block diagram of the GBT is reported. Some of the blocks are typical components of transceivers (CDR and PISO/SIPO converters), others are specific to the trigger functions (trigger logic and bunch emulator), perform control of error conditions that are common in HEP environments only (for example, Single Event Upset (SEU) monitor, watchdog), or help testability. Many parallel and serial ports following well established industrial standards are available: 1–Wire, I2C, JTAG. The format of the GBT data packet is shown in Fig. 2.20. A fixed header (H) is followed by 4 bits of slow control data (C), 80 bits of user data (D) and the Reed-Solomonforward error correction (FEC) code of 32 bits. The coding efficiency is therefore 88/120 = 73%, and the available

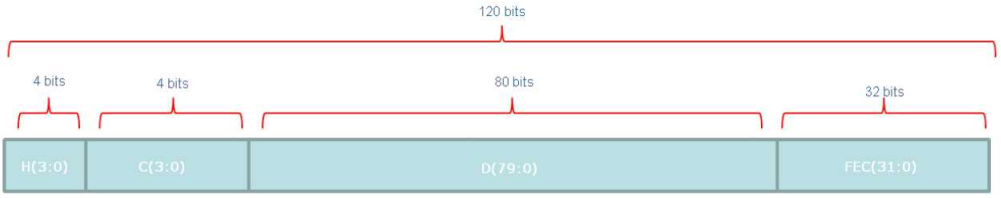


Figure 2.20: GBT frame format.

user bandwidth is 3.2 Gbps. Even though, the real the GBTx is an ASIC chip, together with the GBT project, the GBT-FPGA project started too, with the intend of emulating the GBTx serial link and test the first GBTx prototypes. Thanks to this project, now there is a full library [43], targeting FPGAs either from Altera and Xilinx, which allows the implementation of GBT links on FPGAs.

2.9.2 *Off-The shelf SerDes*

Due to their importance and extensive use many vendors offer commercial SerDes solutions, capable of supporting wide input frequency and data ranges, and thereby support wide data ranges covering most of the applications mentioned.

Texas Instruments (TI), for example, produces general purpose SerDes, such as the TLK1501, TLK2501 and TLK1501, which can be used for high-speed bidirectional transmission. They cover a range of line rate from 0.6 Gbps to 3.125 Gbps. The block diagram is almost the same for all the devices and it is shown in Fig. 2.21, where the blocks already mentioned can be easily identified. On the top there is serializer block. Data, TX(15:0), are first stored in a register and then encoded (if this functionality is enabled). Finally, data are serialized by the Parallel To Serial Block and differentially transmitted. On the bottom, the deserialized section is presented which provide to the opposite function. In the centre, the block for clock and data recovery is shown.

For applications such as Gigabit Ethernet, fibre channel, 10G Ethernet, advanced SerDes needs to be development, such as the TLK3134 by TI. The TLK3134 is a flexible four-channel independently configurable serial transceiver which is ideal for bidirectional point-to-point data transmissions up to 30 Gbps data rate. It can be configured to be compliant with the 10Gbps Ethernet XAUI specification and the 1000Base-X 1Gbps Ethernet Specification. In the Fig. 2.22 the basic functions of a MGT can be easily recognized. For the XAUI core, a high level description of the TLK3134 is shown in Fig. 2.23. The structure is more complicated and four channel are contemporary used to handle the amount of data. In Fig. 2.23, two new blocks RXFIFO/CTC and TXFIFO/CTC are used. In the receive data path, a FIFO (RXFIFO/CTC) is placed on the output of the serial to parallel conversion logic. It compensates for channel skew, clock phase and frequency tolerance differences between the recovered clocks for each serial links and the re-

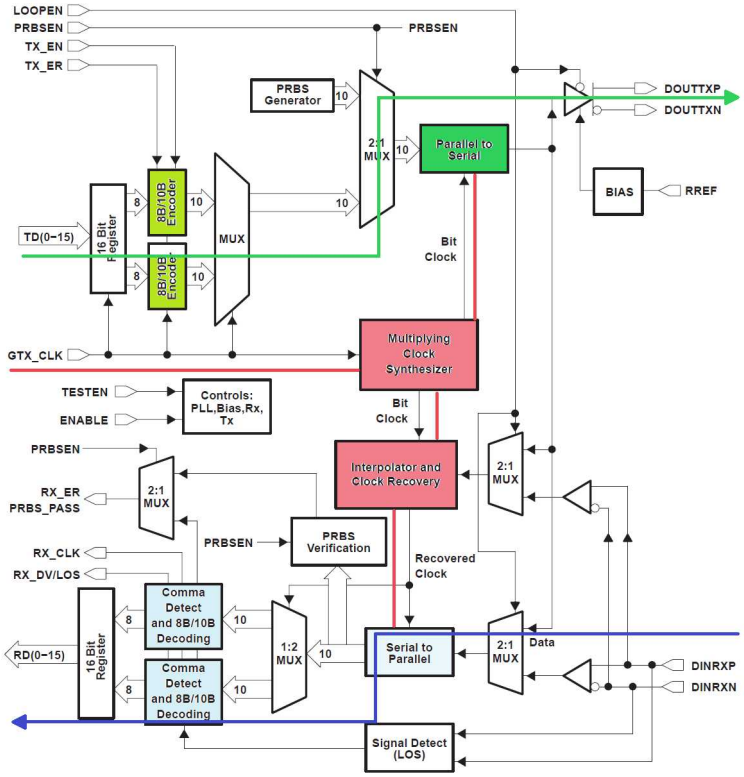


Figure 2.21: TLK1501 Block Diagram.

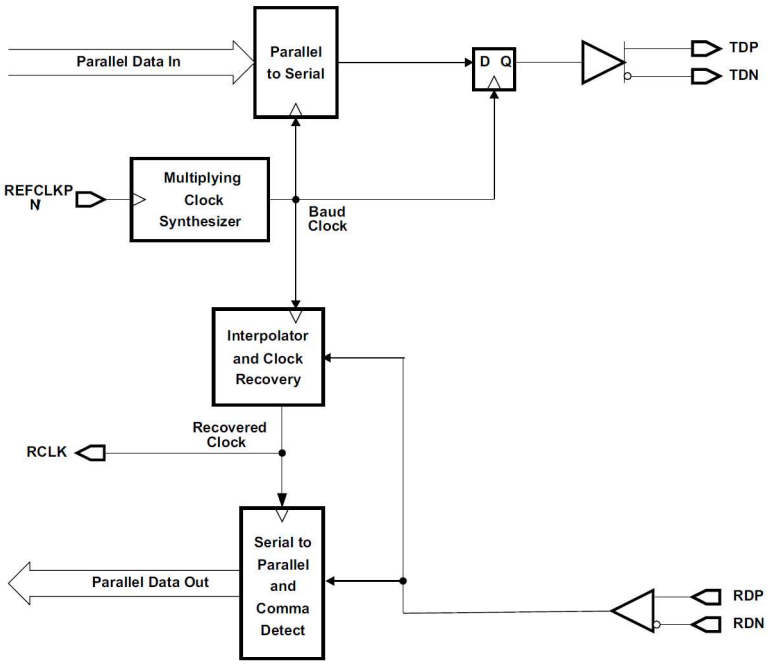


Figure 2.22: Core Block Diagram of the TLK3134 SerDes.

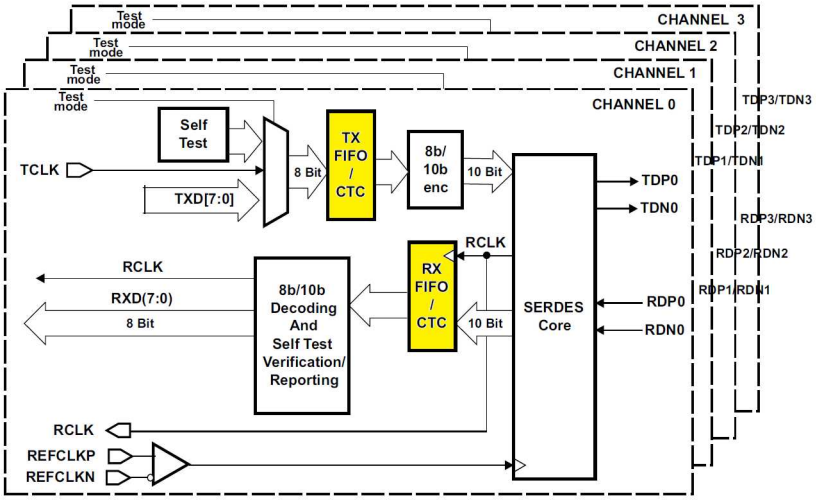


Figure 2.23: Block Diagram of LK3134 SerDes for XAUI1000BaseX applications.

ceive output clock, RCLK, synchronous with the recovered parallel data, RD(31:0). Besides, the two blocks provide for Clock Tolerance Compensation (CTC). Indeed, although the XAUI interface is defined to allow for separate clock domains on each side of the link, the reference clocks for two devices on a XAUI link cannot have the same exact frequencies. The slight differences in clock frequencies is compensated via the insertion or the removal of special characters on all channels. This technique avoid the FIFOs on the receiver/transmitter to over or under run.

3

SELF ADAPTIVE SERIAL LINK

In this chapter a frequency agile, auto-adaptive serial link is presented [44]. The system is intended as a proof-of-concept of a more general adaptive serial link, which is capable to analyze the incoming data stream and to find the highest transmission line rate, according to a given tolerated BER. In particular, it is designed around an FPGA-embedded microprocessor, which drives the programmable ports of an embedded Multi-Gigabit Transcever (MGT) in order to control the quality of the received data and to easily calculate the Bit Error Ratio (BER) in each sampling point of the eye diagram. Besides, the link is able to reconfigure the MGT parameters in order to fully benefit from the available link bandwidth, by setting the highest line rate.

This architecture can be used whenever it is crucial to have prompt information about the maximum sustainable line rate of a connection, such as on test beam sites, where the user could be forced to adapt dynamically his setup to the cabling resources available on site. The system can be also used as a monitoring tool in HEP experiments, where the connections are subject to ageing and insertion losses, and the fibres can increase their attenuation because of radiation; it can also be implemented whenever it is necessary to remotely monitor a transmission system, in order to detect issues in the serial link physical layer. As an application example, it can be proposed some of the many experiments at Large Hadron Collider (LHC) at CERN, which have been intensively using different serial links, both for transmission of Timing, Trigger and Control signals (TTC) [1] and for trigger and data readout [2, 3]. Some of the listed serial links were required also to be fixed-latency links, for transmission of trigger data, and were originally designed and deployed in the field by using custom ASICs. However, recent studies [40, 45] show that last generation FPGA features permit to implement the same links in cost-effective designs, in order to overcome the obsolescence of some component of the original link and to permit the integration in the new hardware, designed for the different upgrades of the LHC physics program [46, 47]. The solution proposed could be easily adapted in this framework, as it can be used on top of any user's existing link, as it has no specific requirement about link specification or protocol.

The first part of the chapter focuses on the state of the art in the field of adaptive serial links. Hence an introduction to FPGA devices, and the description of some architectural details of the embedded MGT, which the link is based on, are presented. Finally, the self-adaptive link project, the description of the test bench used and the experimental results are described and discussed .

3.1 STATE OF THE ART AND MOTIVATION

Different adaptive link strategies are explored in literature, mostly proposing architectures that define low power or low error solutions, as we will describe later. The main task of the link developed in this paper, on the other hand, is to provide the best performances in terms of bandwidth, compatible with the physical layer. State-of-the art research activity in electronics is presently focusing on adaptive serial links capable to perform traffic-driven rate adaptation in wireless networks or Ethernet. Different adaptive link rate (ALR) mechanisms can be found in literature [48, 49], which allow to lower the link rates when the network traffic request is reduced, thus lowering the power consumption, with little impact on network performances [50, 51]. Moreover, wireless application also have to face with the poor quality of the channel; these applications must adapt the data transmission to the link-level error rate and provide techniques to recover from transmission errors. To be more specific, as an example, it can be investigated on the Bluetooth [52] specification, which provides two techniques in order to overcome link-level transmission errors, due to the poor channel quality, which causes a relatively high error rate (typically around 10^{-2}). The two recommended techniques are:

- Forward Error Correction (FEC) coding, which adds overhead on the data payload in order to detect and correct transmission bit errors;
- Automatic Repeat reQuest (ARQ) transmission scheme, which informs the source that the last data transfer was successful or failed, thus allowing for a retransmission of data.

Unfortunately, these two schemes has the following drawback:

- FEC coding always adds overhead on the data, thus reducing the throughput of the link, even when the transmission is error free;
- ARQ scheme can typically introduce additional and variable delays due to the retransmission, which are not suitable for real-time applications, such as fixed latency trigger paths.

Another activity of research in the field of serial links concerns the power saving techniques in the serial communication between Integrated Circuits (IC) chips, mounted on supercomputer or server motherboards, with high data-processing demands, or in high-speed network routers which require a large number ($\sim$ hundreds) of links. Indeed, simulations and experiments show that most Ethernet links are underused and the energy consumption of the link could be reduced by simply operating them at a lower line rate. In order to save power, by lowering the line rate, one of the suggested techniques [53] is the use of adaptive power control. When the peak performance is not needed, adaptive power-supply regulation lowers the power consumption by reducing the supply voltage of the serial-link chips to the minimum supply value required to support the transmission

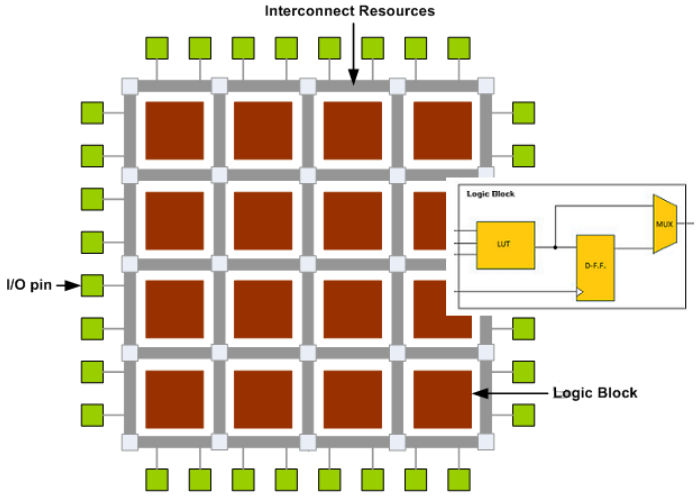


Figure 3.1: Overview of FPGA architecture.

frequency. Since the power dissipation of a CMOS is proportional to V^2f, lowering both the supply voltage V and the frequency f greatly reduces the overall power dissipation. Adaptive power supply regulation techniques usually rely on analog feedback circuits, which adjust the supply voltage of an oscillator (thus increasing/reducing its internal propagation delays) so that the modified frequency fits with the desired transmission frequency. This approach is usually adopted in ASIC chips, while it cannot be followed in an FPGA-based serial link design, where constraints on the power supply are very tight.

Different from the state-of-art research activity, this proposal aims at obtaining the maximum line rate link performances. However, since this architecture is principally designed to meet the requests of the HEP experiments, its principal goal is to obtain a safe transmission at the highest line rate for a given physical layer. In order to work at the highest line rate, without the possibility of using the FEC technique or retransmitting data, an error-free transmission has to be guaranteed. Due to its intrinsic nature and in contrast with the links previously described, this architecture can easily fit in TDAQ systems of HEP experiments with real-time requirements, as neither extra delay is added to the main functionality of the serial link nor overhead is added to the payload. It has been also intended for implementation in Xilinx SRAM FPGA, thus gaining from the flexibility and re-programmability of the device, and it can be added on top of any user's custom serial link, as it has no requirement about link specification or underlying protocol.

Before going into the detailed description of the architecture, a general description of FPGAs is given, as a background for a better understanding of the projects.

3.2 FIELD PROGRAMMABLE GATE ARRAY, FPGA

Field Programmable Gate Array devices, FPGAs, are integrated circuits containing logic blocks which can be programmed by the user *in the field* [54]. Logic blocks contain a wide set of programmable logic gates and memory elements (simple flip-flops or more complex block of memories). FPGAs contain also interconnect resources, which allow the blocks to be *wired* together and to implement almost any kind of digital circuit or system.
The basic structure of an FPGA (3.1) includes the following elements:

1. Look-up table (LUTs): blocks which performslogic operations;
2. Flip-Flop (FFs): register elements which can store the result of the LUT;
3. wires: interconnection resources;
4. IO pads: physically ports that get data in and out of the FPGA.

Although this structure allows the implementation of a wide range of digital system, in order to improve their efficiency, contemporary FPGAs incorporate additional elements, which implement specific functions such as:

1. embedded memories for distributed data storage (BlockRAMs);
2. Phase-locked loops (PLLs) for driving the FPGA fabric at different clock rates;
3. external and/or embedded microprocessors;
4. Digital Signal Processors (DSPs).

The final FPGA architecture result in a more complicate scheme shown in Fig. 3.2.
For low to medium volume productions, FPGAs provide a cheaper and faster time to market solution with respect to Application Specific Integrated Circuits (ASIC). Indeed, while obtaining the first ASIC prototype normally requires a lot of resources in terms of design time and financial resources, an FPGA is ready to use and to be configured (in less than a minute) and its cost is reasonable low. In the latest FPGAs, to improve their flexibility a further, useful feature is present. In this devices is possible, a partially reconfiguration of a portion of FPGA (through the Dynamically Reconfiguration Ports, DRP) while the rest of it is still running.

3.2.1 *FPGA's Technologies*

The reconfigurability and reprogrammability of an FPGA underlies on the technology used to implement the logic blocks and the interconnections. There are a number of technologies including static memory (SRAM), flash and anti-fuse.

1. SRAM-based FPGAs: They store logic cells configuration data in the static memory which are organized as an array of latches. SRAM-based technology is the most widely used approach, because of the re-programmable

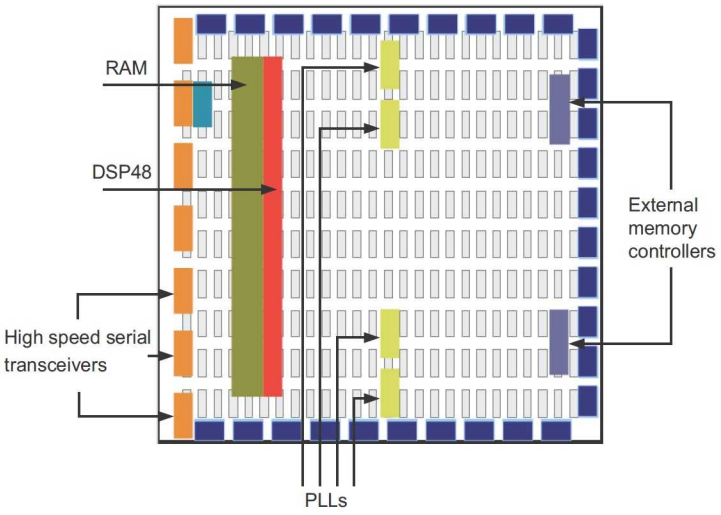


Figure 3.2: Detailed internal FPGA architecture, including macroblocks.

capability and the use of standard CMOS process technology (which means higher and higher integration and speed, and lower and lower dynamic power consumption). However, its large area, as compared to other technologies, and its volatile nature are the main drawbacks. 5

2. Flash-based FPGAs: The flash-based FPGAs uses flash or EEPROM as a primary resource for configuration storage. This technology has several advantages: it is less power and area consumptive, non volatile, and more tolerant to radiation effects. However it cannot be reconfigured an infinite number of times and its technology is based on non-standard CMS process.

3. Antifuse-based FPGAs: Antifuse-based FPGAs are different from the previous mainly because they can be programmed only once. Its primary advantage relies in its lower area occupancy as well as in the presence of low parasitic capacitance. The antifuse is a device that doesn't conduct current initially, but can be *burned* to conduct current. The antifuse-based FPGA can't be then reprogrammed since there is no way to return a burned antifuse into the initial state.

3.2.2 *Design Flow*

Design flow of FPGA starts with the behavioural description of the circuit to implement. It is usually used a hardware description language, HDL (*Very-High-Speed-Integrated-Circuit*-HDL, VHDL, or Verilog) more then a schematic design. In any case, they allow the user either to design digital systems and to simulate its operation with different level of accuracy. An automation tool (usually a company's proprietary software) is used to compile the hardware description code and convert it into a stream of bits, bitstream, which is eventually loaded into the FPGA to configure the logic blocks and the interconnections. The complete

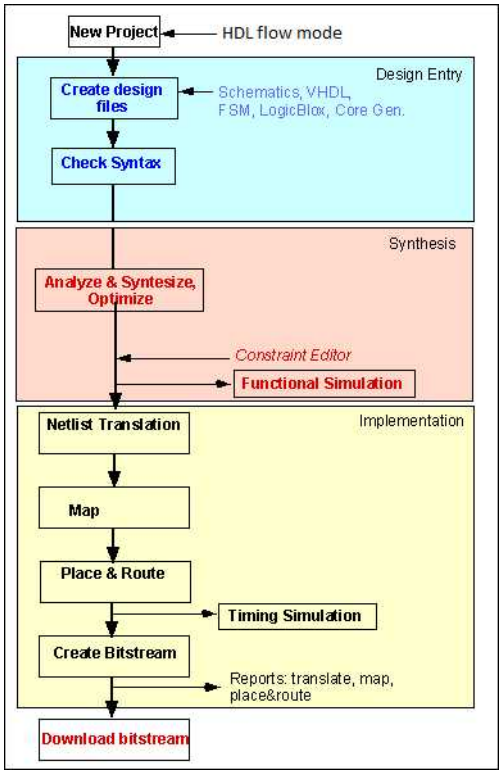


Figure 3.3: Overview of design flow of a Xilinx FPGA device.

development process is depicted in more details in 3.3 [55].

Once, the logic circuit has been developed using an HDL language, a synthesizer transforms the code into a gate-level netlist, using the components included into basic libraries. The netlist, which is a description of the various logic gates in the design with their interconnection, is device independent. The synthesis tool can optimize the network in terms of area, speed or power consumption. At this stage behavioural simulation can and must be done in order to check the proper operation of the circuit, however information about switching delays is not included, yet.

The following phase, named implementation, consists of three different steps: translation in which one or more netlists are merged together along with any design constraints; mapping: where gates in the netlist are combined into groups that will fit efficiently into the LUTs of the FPGA; the placement and routing in which, after groups are assigned to LUTs at various locations in the FPGA the tool determines how to connect them together using the routing resources in the switching matrix and and input and output pins. Any stage of the implementation must be validated via timing analysis, simulation, and other verification methodologies. It is important to highlight that going down through the design, the netlist is augmented with more and more information about how the design will be placed into the FPGA. This allows to perform a more realistic timing simulation that takes into account the effects of gate and wiring delays on the operation

of your circuit.
Finally, once the design and validation process is complete a bitstream generator takes the output of the implementation phase, combines it with a other configuration settings, and outputs a binary bitstream. This bitstream is a binary data which contains the truth tables that will be loaded into the RAM of every LUT and the connection settings for the wiring matrix that will connect them. This file is transferred into a physical FPGA chip via a serial interface (JTAG) or to an external memory device.
To simplify the design of complex systems in FPGAs, there exist libraries of predefined complex functions and circuits that have been tested and optimized to speed up the design process. These predefined circuits are commonly called IP cores, and are available from FPGA vendors and third-party IP suppliers (rarely free, and typically released under proprietary licenses). Other predefined circuits are available from developer communities such as OpenCores (typically released under free and open source licenses such as the GPL, BSD or similar license), and other sources.

3.2.3 *MGTs in Xilinx FPGAs: GTP and GTX*

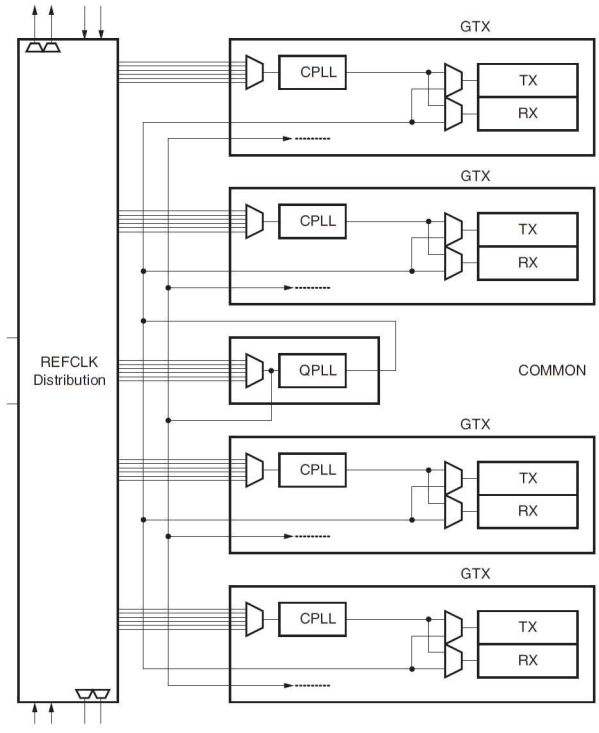


Figure 3.4: Detail of Xilinx *QUAD*QUAD. GTXs in a *QUAD*QUAD share a number of resources including a PLL, reset logic, and a DRP.

In order to better understand the way in which the link has been developed, it worth to understand some of the architectural details of embedded SerDes used. The link developed in this work is based on the Multi-Gigabit Transceiver (MGTs) of the Xilinx 7-series FPGA family [56]. Thus the following discussion focuses on the architecture of Xilinx MGTs found in these chips, though the principles are similar for MGTs in other chips and from other vendors [57]. Furthermore, the discussion primarily references Xilinx High Speed (GTX) MGTs, though the architectural details are nearly the same for the Low Power (GTP) MGTs [58] (which is used in the work presented in). Most of this information is available in more detail in [59], but in this chapter only the most relevant to this work are presented.

Inside the FPGA, GTXs are grouped into blocks called *QUAD* as shown in Fig. 3.5. Each *QUAD* contains four GTXs which share some basic components:

1. the clocking block for the distribution of the input reference clock between and inside the QUAD;

2. a Phase Locked Loop (PLL), used to generate the high speed clock used for both tiles from a reference clock via the clocking block;

3. the reset logic, which can reset all components of the *QUAD*, including the shared PLL, or it can be applied to each GTX and to some subcomponent, separately;

4. the power control logic, which allows to power down the entire *QUAD*;

5. the Dynamic Reconfiguration Ports (DRP), which allows the user logic to change certain GTX configuration settings.

GTXs are available as configurable hard-macros each of one including a pair of transceivers: a transmitter (TX) and a receiver (RX). Fig. 3.5 shows the architecture of the transmitter (TX) and the receiver (RX) included in each transceiver. Both the TX and the RX are divided into a Physical Medium Attachment (PMA) section and a Physical Coding Sublayer (PCS). The first performs the serialization (for the TX) and the de-serialization (RX) of the data, while the second one elaborates information; in the PCS section, for example, the 8B/10B circuit codes the data, if necessary. The GTX transceiver has different clock domains bridging the frequency gap from the user logic clock which presents parallel data to the high speed clock used to transmit serial data. These multiple clock regions split both the RX and the TX into a parallel and a serial section. The first one, include three clock domains, while the second one is fed by a high-speed serial clock. In the parallel section, the TX/RXUSRCLK and TX/RXUSRCLK2 are generally provided from the FPGA Fabric. Instead, a TX and a RX Phase Locked Loop (PLL) internally generate the parallel XCLK and the serial clock using two distinct reference clocks (TX/RXPLLREFCLK). However, if the transmitter and the receiver work at the same line rate, there is also the possibility of sharing the RX-PLL with a power

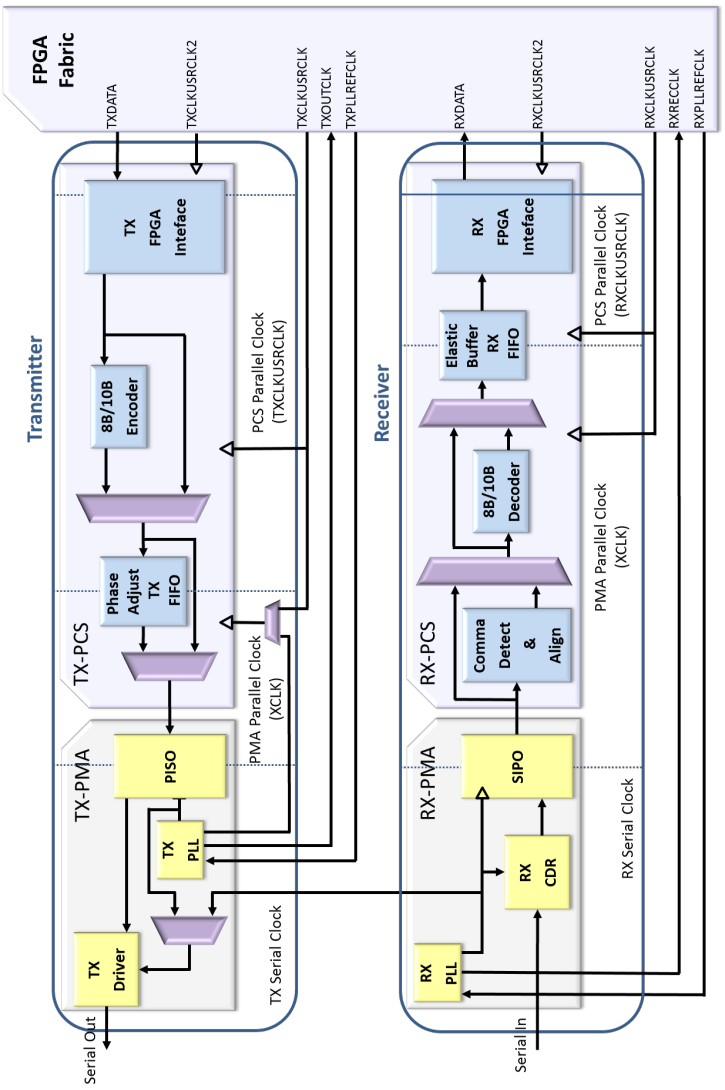


Figure 3.5: Detail of Xilinx GTX architecture.

consumption improvement. Besides, both the PLLs have dedicated outputs, TXOUTCLK and RXRECCLK, that can be potentially used to drive TXUSERCLK2 and RXUSERCLK2, respectively.

On the transmitter, the TX FPGA Interface reads the information coming from the FPGA Fabric on the positive edge of TXUSRCLK2, while on its output lines, data to be transferred to the 8B/10B encoder are written synchronously with TXUSRCLK. The TXUSRCLK rate depends on the length of data words. Indeed, the GTX allows the user to configure the width of the ports to the values 8, 10, 16, 32, 40, 64, 66-bit wide. Then, according to this value and to the TX line rate, the user or the TX-PLL feeds the FPGA Interface with the correct TXUSRCLK. In addition to the TXUSRCLK2 signal, the PCS section is characterized by another internal parallel clock: the PMA parallel clock domain, XCLK. For a safe data transmission, the XCLK rate must match the TXUSRCLK rate and no phase differences exists between the two domains. For this need, the GTX transmitter includes a TX-FIFO and a TX phase-alignment circuit which allows to minimize phase differences between the PMA clock and TXUSRCLK signal. The boundary between the PMA and PCS sections is represented by the Parallel In Serial Out (PISO) block which provides to the Parallel-to-Serial conversion. In the TX-GTX, as in the most common SerDes transmitter interface, the PISO block has two distinct input clock: the parallel clock, XCLK, and the high-speed serial clock, used to synchronize the serial output. This serial clock is internally implemented by the TX-PLL, which multiplies the reference clock (TXPLLREFCLK) up to the serial frequency or bit-rate. For this purpose the frequency of the incoming clock has to be a sub-multiple of the bit-rate.

The receiver structure is analogue to the transmitter's one described above. The serial stream is received by the Clock and Data Recovery (CDR) block. This circuit recovers the clock signal from data and uses it to drive both the serial and parallel clock inputs of the SIPO (Serial In Parallel Out). This advanced register carries out Serial-to-Parallel data conversion. The extracted clock is used to synchronize the serial data stream, but is also divided to generate a parallel recovered clock (RXRECCLK). This RXRECCLK is used to drive the SIPO parallel outputs, but it can also be brought out to the FPGA fabric logic to clock the output lines of the RX FPGA Interface. In the RX-GTX, the Comma Detect and Align block assures that parallel data coming from the SIPO are correctly aligned to word boundary (i.e. 8B/10B comma symbol). Then, if necessary, data is decoded, otherwise, it is directly transferred to the Elastic Buffer RX-FIFO which guarantees the correct transmission when data enters the RXUSRCLK domain.

RX eye margin analysis

The RX side of GTX [59] high-speed transceivers have a new feature, RX eye margin analysis, which allows to measure and display the receiver eye margin. In the standard operation, data from the serial stream is recovered by sampling a specific point (Data Sample) (Fig. 3.6). The horizontal position of the Data Sample is determined by the CDR algorithm and the vertical position is the common

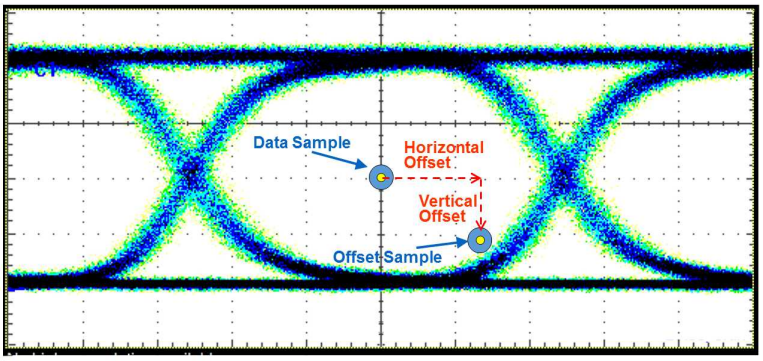


Figure 3.6: Position of Offset Sample and Data Sample.

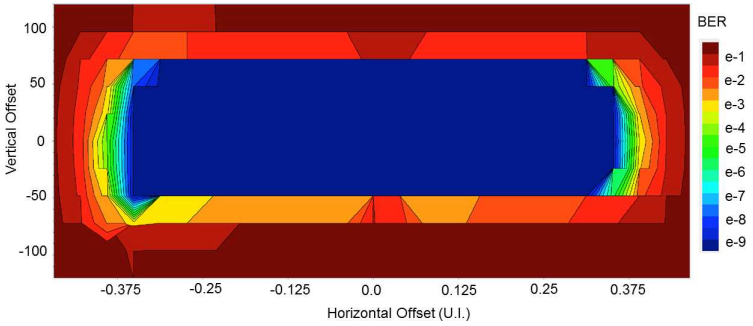


Figure 3.7: An example of Statistical Eye Diagram obtained from the RX eye margin analysis procedure.

mode voltage. When the eye scan functionality is running, an additional sampler is activated in the GTX and a new sample (Offset Sample) is acquired, with programmable (horizontal and vertical) offsets from the data sample point.
An eye scan measurement run is performed by acquiring a large number of Data Samples (which can range from tens of thousands to 10^{14} or more) and by counting the number of times the offset sample has a different value with respect to the data sample; the latter number is often called Error Count. The bit error ratio (BER) at a specific vertical and horizontal offset is given by the ratio between the Error Count and the Sample Count. By repeating the eye scan measurement for each horizontal and vertical offset in the Unit Interval (or in a part of the U.I.) a 2-D BER map can be produced which is usually called Statistical Eye (Fig. 3.7).

This functionality poses no specific requests or constraints on the data pattern to be received, and no change is required in the RX configuration. Moreover, no additional FPGA logic is required, except the possibility to read and write the GTX attributes. Thus, the Statistical Eye can be measured without compromising the link functionality, i.e. while the user's application data is being transmitted on the serial link.

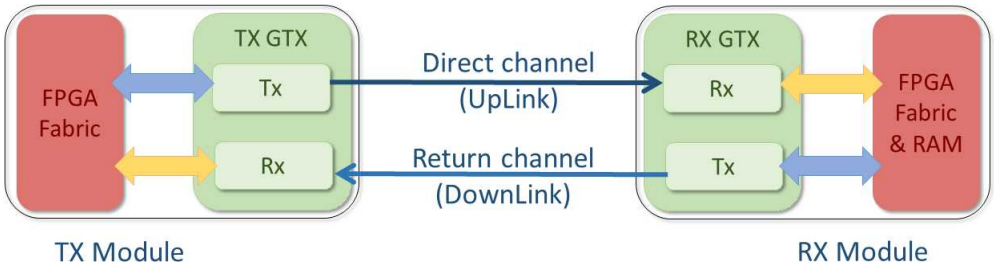


Figure 3.8: Simplified block diagram of the adaptive link.

3.3 IMPLEMENTATION

The frequency agile, auto-adaptive serial link is designed around an FPGA-embedded MGT, whose programmable ports are driven by an embedded microprocessor. Xilinx provides a standalone tool [60] that allows performing the Eye Scan Analysis on the receiver side of the GTX transceiver, using the MicroBlaze [61] Micro Controller System macro; the toolkit also includes the Eye Scan algorithm (providing the C code). Moreover, Xilinx supplies the hardware sources files for the implementation of a link based on the XAUI protocol, in which the GTXs are arranged in a loopback configuration. The original contribution of this work consists in the build-up, design and optimization of a full architecture, on top of the basic Xilinx tool, which:

1. drives the programmable ports of the GTX in order to modify the line rate of the link;
2. runs consecutive eye scans for various line rate;
3. analyzes the results of the different scans, in order to find the maximum line rate sustainable by the link;
4. manages the synchronization between the transmitter and the receiver of the link, that will be needed at each line rate change.

Before starting the description of the implementation, it is necessary to spend a few words on the definition of a handshake mechanism for the switching of the link line rates.

3.3.1 *Handshake Mechanism*

The adaptive link is made of a TX and a RX module, both implemented by a GTX module, and it needs a full-duplex channel. The serial link is presently designed to handle 16-bit words as a payload and it uses an 8B/10B coding, in order to guarantee a sufficient number of transitions on the link. A simplified block diagram of the adaptive link is shown in Fig. 3.8.

The direct channel is the physical layer under test, for which the Statistical Eye is measured, and it goes from the transmitter module to the receiver module. The

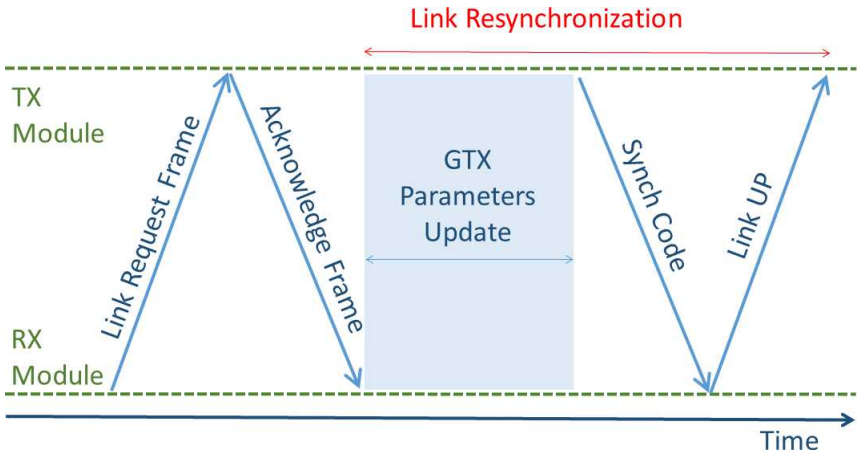


Figure 3.9: Lattice diagram of the handshake mechanism.

return channel goes from the receiver module back to the transmitter module and it is used to make the two modules communicate at a low rate. The line rate on the return channel is kept as low as possible (500 $Mbps$ for this version of GTX), assuming that such line rate is safely sustainable by the channel. A discrete set of different line rates $f_0, f_1, ..., f_{k-1}, fk, f_{k+1}, \ddot{;} f_n$, with $f_0 < f_1 < f_{k-1} < fk < f_{k+1} < f_n$, can be chosen in the definition of the link properties, both at the TX module and at the RX module. Obviously, the line rate settings must be the same for the TX and the RX side, in order to allow the link establishment.
When the system is operating in the adaptation mode and the link is established at a certain line rate f_k, a pseudorandom sequence is transmitted in order to emulate random traffic on the link, allowing to perform the 2-D BER map; the data of the pseudorandom sequence flow from the TX module to the RX one, and the Statistical Eye can be acquired for that particular line rate. When the Statistical Eye acquisition is completed, a validation of the line rate f_k, according to a specific BER, is performed, following the algorithm that will be explained later.

After this stage, the system changes the line rate f_k, both at the TX module and at the RX module, according to the following rules: if the validation of f_k was successful, the rate is increased to $f_{k+1} > f_k$, in order to perform the same analysis for a higher line rate, if a line rate higher than f_k is available; if the validation of f_k failed, the rate is decreased to fk-1 < f_k, in order to reach the previously validated and safe line rate. Thus, a change in the line rate is necessary at least once (unless the lowest line rate is not validated), and a handshake mechanism between TX module and RX module, for the negotiation of the change in line rate, must be defined. A lattice diagram of the handshake mechanism is shown in Fig. 3.9.

In our architecture, as the Statistical Eye and the rate validation are performed at the receiving end of the link, the RX module is the only able to initiate a request to change the rate, as the master of the handshake mechanism. Obviously, an established link from the RX to the TX module is necessary, in order to perform the line rate change negotiation process.

The RX module determines the need for increasing or decreasing the line rate and sends a request using a specific frame, labelled Link Request Frame (LRF), which contains the *increase rate* or *decrease rate* instruction. The TX module, which acts as slave in the handshake mechanism and contains the receiver end of the return channel, acknowledges the line rate change request, by sending the Acknowledge Frame (ACKF) on the direct channel. The ACK Frame triggers a line rate modification on both TX and RX module of the link and, in order to modify the line rate, an update of the parameters of the GTX is performed, as described later. A link resynchronization procedure will be needed at the new line rate.

3.3.2 *The Resynchronization procedure*

The resynchronization procedure is initiated by the TX module by sending a specific sequence of words (a 8B/10B comma plus user's synch code), after the generation of the ACK Frame, according to the following rules:

1. wait for an additional dead time of 10 µs, accounted for the ACK Frame propagation on the direct channel;

2. wait for the typical time needed to update the GTX programmable parameters on both modules (δ 5 µs);

3. wait for an additional dead time of 5 ms, accounted for the configuration of both the ends of the link with the new parameters.

After the reconfiguration, the logic on the RX module compares the received sequence with the expected one and, in case of successful transmission, it sends a *Link UP* code on the return channel, in order to confirm to the TX module that the link is established. If the incoming sequence on the RX module is not correct, or if no data arrives to the RX during a timeout window (default timeout is 1 s) from the arrival of the ACK Frame, the transmitter on the RX module sends a *Retry* command to the TX on the safe return channel and keeps waiting for another synch code from the TX module on the direct channel. When the TX module receives the *Retry* command, it sends again the synch code on the direct channel and waits for the *Link UP* confirmation on the return channel.

The *Retry* procedure is repeated 5 times, unless a correct data transmission is successful; after 5 *Retry* procedures, the RX module sends a LRF with the *Decrease Rate* command, in order to bring back the line rate at the previously working one.

The total time required for the handshake and the resynchronization, T_{switch}, depends on the different procedures involved in the process; anyway, the lower limit of T_{switch} is reached when no *Retry* procedure is activated and, in this case, it expected to be slightly higher than 100 ms and it slightly depends on the line rate.

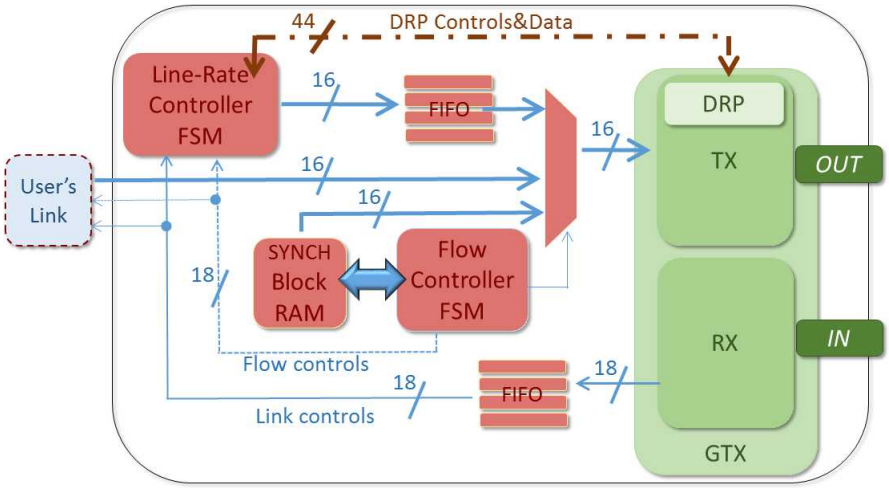


Figure 3.10: Simplified block diagram of the TX module of the adaptive link.

3.3.3 *TX and RX Module Architecture*

The simplified block diagram of the TX module, which is designed around a GTX macro, is shown in Fig. 3.10.
The main tasks of the TX module are:

1. to send data for the synchronization of the link; such data are made of synchronization specific words (commas) and of a user code stored in the SYNCH RAM;

2. to send the appropriate codes for the test of the direct channel of the link; this phase is managed by the Line Rate Controller FSM, which also reads back the information from the RX module, manages the Handshake mechanism and writes the appropriate data on the Dynamic Reconfiguration Port (DRP [62]) of the GTX in order to modify the line rate;

3. after the adaptive phase of the link, to become transparent and transmit the user's data on the link.

The dataflow of the three different phases of operation is handled by the Data-Mux multiplexer, managed by the Flow Controller FSM: such state machine is the master of the TX module, as it also controls the activity of the different parts of the architecture. A crucial role is played by the system clock, different from the clock signals used by the GTX (one clock for the TX part and one clock for the RX part, not shown in Fig. 3.10 for sake of clarity); indeed, the GTX clocks may not be available during some specific procedures (i.e. the reconfiguration of the GTX or the resynchronization procedure, when the link goes down) and the use of the system clock is fundamental for keeping the rest of the logic alive. In order to decouple the clock domain of the GTX and the system clock, two First-In-First-Out (FIFO) memories are implemented in the design. The most complex part of the system is the RX module, built around a GTX macro, the Xilinx Micro Controller

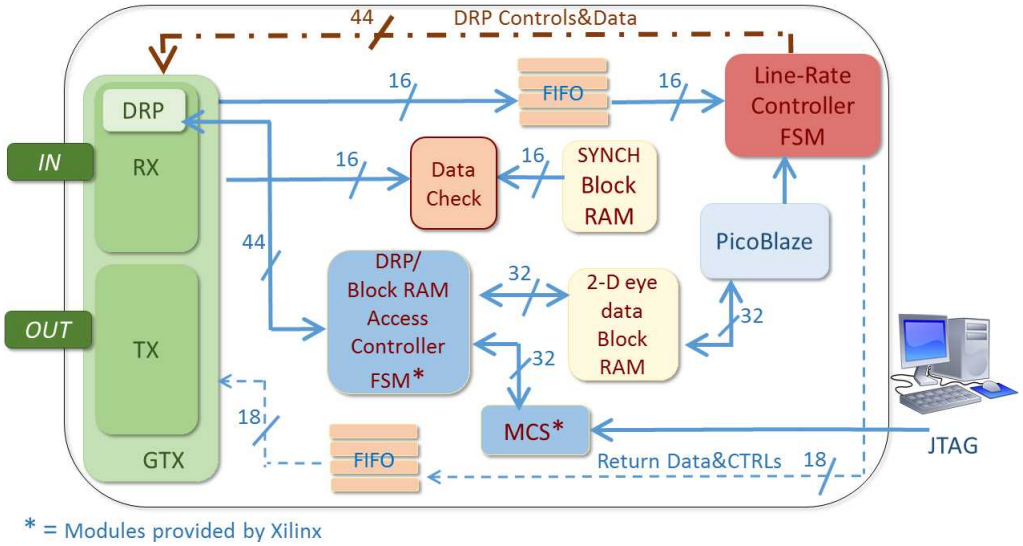


Figure 3.11: Simplified block diagram of the RX module of the adaptive link.

System (MCS) block, an embedded microprocessor (PicoBlaze [63]) and a cluster of cooperative Finite State Machines, which build up most of the fabric logic. The simplified block diagram of the receiver module of the adaptive link is shown in Fig. 3.11.
The MCS block modifies the Offset Sample position in the receiver part of the GTX, by accessing its DRP port; moreover, it acquires the Statistical Eye data from the GTX and saves the data in four 18-kbit block RAMs (2-D eye data RAM).
Because the MCS uses the same address bus both for DRP and 2-D eye data RAM, the DRPBlock RAM Access Controller Finite State Machine (FSM) controls the access to the DRP, in order to change the Offset Sample position, or the 2-D eye data memory, either for reading or for writing.

Finally, based on the results of the eye scan run, stored in the 2-D eye data RAM, the PicoBlaze processor chooses the best frequency and, through the Line-Rate Controller FSM, it configures the DRP ports of the RX part of the GTX, in order to properly change the line rate with the new specifications. In the block diagram, a communication backbone is used by the Line-Rate Controller FSM to setup the transmission frequency of the link, by setting DRP controls and data signals; the backbone is represented by the dot-and-line connection.
A Personal Computer, transferring data to Picoblaze according to the RS-232 standard, is used in order to start an eye-margin analysis run; the PC is also used in order to perform the readout of the eye-scan data from the 2-D eye data RAM, that are saved onto the PC in an ASCII file. Besides, in order to automatically elaborate and easily analyse the data collected during the scan, a Mathematica [64] code has been designed which produces various plots and maps for the characterization of the link.

3.3.4 *Line rate validation procedure*

As described in the introduction section, data from the serial stream is recovered by sampling a specific point (Data Sample) inside the Unit Interval, determined by the CDR. However, another sample can be acquired (Offset Sample) by changing the programmable (horizontal and vertical) offsets from the data sample point.
An eye scan measurement run is performed by acquiring a large number of Data Samples and by counting the number of times in which the Offset sample has a different value with respect to the Data sample. The BER at a specific vertical and horizontal offset is given by the ratio between the Error Count and the Sample Count.
By repeating the eye scan measurement for different horizontal and vertical offset in the Unit Interval, a 2-D BER map can be produced which is usually called Statistical Eye. The higher number of horizontal and vertical offset is used, the higher resolution map can be obtained, even if this can have a severe impact on the time needed for a full U.I. scan.
The statistical eye is composed of coloured BER contours, where each curve (also called *ISOBER* curve) represents the eye diagram for a specific BER performance. Familiar quality graphs of a serial link, like the Bathtub plots, showing the eye width or eye height at different BER levels, can be drawn by intercepting the statistical eye along the timing or amplitude axis. It's worth remembering that the BER value has to be estimated with a given Confidence Levels. Once the BER and its Confidence Level are set, the number of bits to be transmitted through the system is fixed. The relation between these quantities is well-known in literature and already explained in the chapter 2.1. This also brings the designer to a trade-off of test time versus BER Confidence Level.

For this reason, it has been chosen to transmit always the same normalized number of bits, equal to 6, thus sufficient to reach the desired BER with a 90% Confidence Level even in presence of 2 errors. Given the 90% Confidence Level, the test time for this approach depends on the desired BER, on the line rate and on the number of the Offset points, which influences the resolution of the map. In order to keep the test time reasonable, i.e. few minutes for the full scan at different rates, a value for BER of 10^{-8}, with a number of offset points on the horizontal and vertical axis respectively equal to 25 and 11, was chosen.
In this design, which should be intended as a proof-of-concept of the system, data from the 2-D BER map and the monotonic distribution of the *ISOBER* curves are used in order to build a *keep-out* region in the U.I..
The *keep-out* zone identifies a rectangular region which extends for 50% U.I. of the horizontal axis and for 50% of the vertical axis. The logic uses the *keep-out* zone in order to check the link under test: for a given line rate, the link is validate only if all the Offset points inside the *keep-out* region exhibit a BER less than a given value 10^{-8} with a given Confidence Level (90%).
The *keep-out* zone definition is very simple, as it is intended only to demonstrate the feasibility of the full architecture; when implemented in a more complex ver-

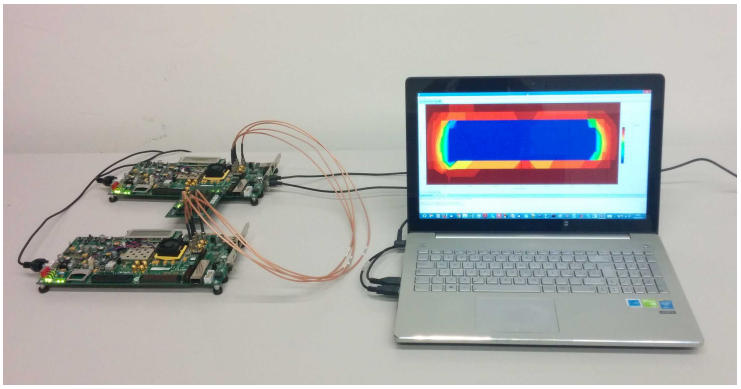


Figure 3.12: Picture of the setup for developing and testing the project.

sion of the adaptive link, to be deployed in a specific experiment, the shape of the *keep-out* zone or even the thresholds for the validation can be modified by the designer according to the specific requirements of the link. In order to easily read the test results, it was defined a Pass/Fail map, starting from the Eye map data. If the system finds 2 errors or less for a given Offset point, that point passes the test. If the system finds 3 errors or more for any Offset point, the test is not validated for that point, which therefore doesn't pass the test. In the next section we show two Pass/Fail maps, for different physical layers.
In this implementation, the line rate validation algorithm is carried out in the fabric by using custom ad-hoc logic; specifically, the full algorithm is run by the Line Rate Controller FSM and by the PicoBlaze embedded microprocessor, which is used for the calculation of the *keep-out* zone boundaries; a 18-kbit RAM is also used for data storage.

3.3.5 *Test Setup and result*

The development platform used in this project is the Xilinx evaluation board KC705. It is equipped with an FPGA XC7K325T-2FFG900C device, provided with GTX transceivers supporting a line rate up to 10.3125 Gbps. The chosen hardware setup allows using the on-board Sub Miniature version A connectors (SMA), in order to use coaxial SMA cables as link transmission medium. The evaluation board can be connected also to custom mezzanine boards (provided they have SMA connectors), that could be specifically designed in order to host different types of cables to be tested.
In principle, tests can be performed on the optical layer too. Indeed, by plugging a FM-S14 SFP+ FPGA Mezzanine Card (FMC) [65] on the KC705 board, the GTX transmitter and receiver parts can be connected by a duplex-fibre.
Besides, test set-up also includes a host computer; data transfer between the computer and the PicoBlaze microprocessor is carried out according to the RS-232 standard. Fig. 3.12 shows the experimental setup, made of the PC and two KC705 boards.

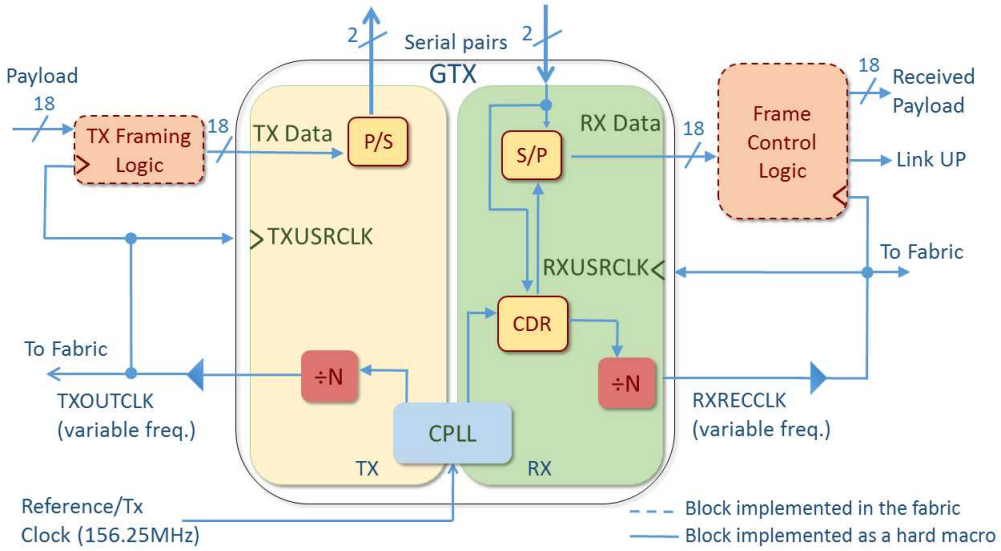


Figure 3.13: Simplified block diagram of the serial link used in the tests.

As previously discussed, the duplex channel is always necessary because of the need of a return channel between RX and TX, which is used for communication of the variations of the line rate, after the choice of the best frequency made by the Line-Rate Controller FSM.

In order to test the adaptive system, two serial links were designed, one for the direct channel and one for the return one. The two serial links have the same simple architecture and the simplified block diagram of a single GTX is shown in Fig. 3.13, where both the TX part and the RX part of a GTX macro are shown.

As previously described, reproduce the internal architecture of transmitter and receiver of a MGT. Specifically, the architecture of the link node consists of the parallel-to-serial (P/S, in the transmitter part) and serial-to-parallel (S/P, in the receiver part) data-paths toward a far-end node.
Synchronization clock signals for the serial and parallel domains are generated by a PLL (CPLL, shared between TX and RX) and the clock divider control blocks (one for the TX and another one for the RX), which divide the clock by a N value, depending on the line rate, compatible with the physical layer. The PLL uses a 156.25 MHz reference clock, provided by the low-jitter, differential oscillator of the KC705 board.
On the transmitter side, the PLL generates the high-speed serial clock and a parallel clock, TXUSRCLK. The clock signal TXUSRCLK is also routed to the FPGA fabric, in order to feed the transmission logic.
On the receiver side, the PLL provides the base clock for the Clock and Data Recovery block (CDR), while the clock divider circuit opportunely produces the parallel recovered clock, RXRECCLK, from the CDR extracted clock. A RXRECCLK replica (RXUSRCLK) is also used to clock the downstream logic on FPGA fabric.

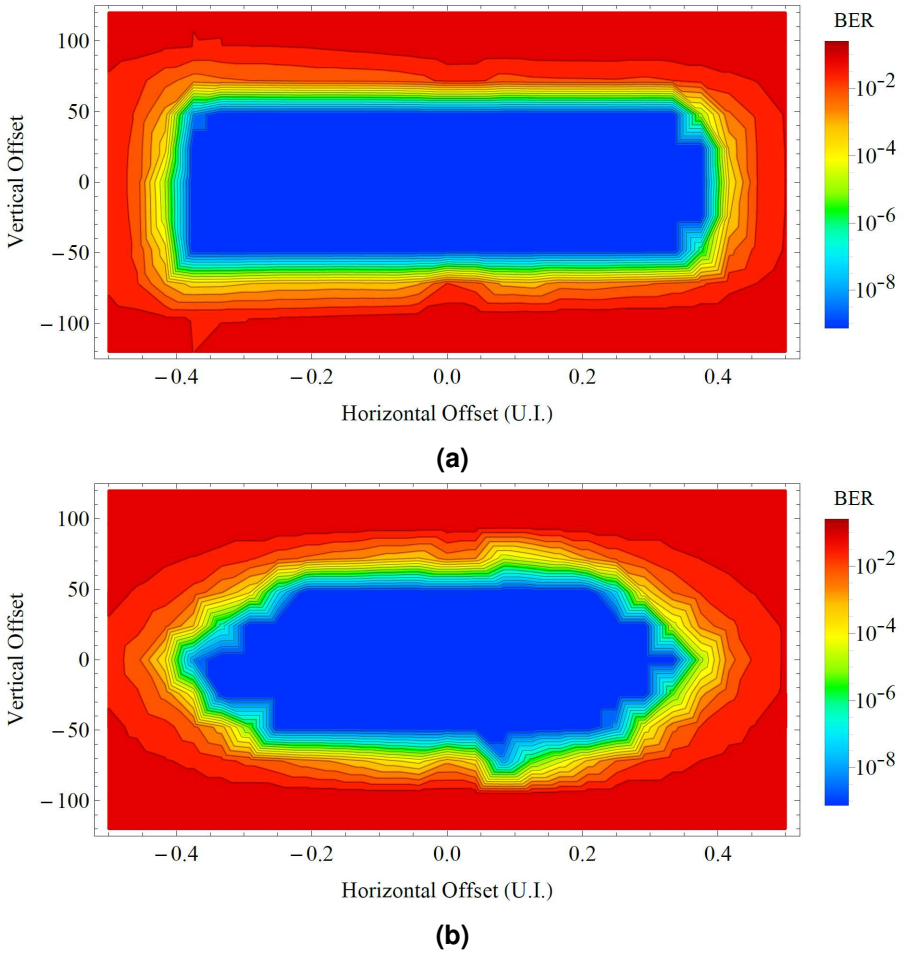


Figure 3.14: Fig. a shows the Statistical Eye produced by using a pair of one meter long SMA coaxial cables while Fig. b shows the results of the same test by using a miniSAS differential cable.

Concerning the TX-RX set on the direct channel, the values of the TXUSRCLK and RXUSRCLK clock signals are not fixed, as the line rate is variable on that channel. Indeed, the line rate is given by the product of the parallel clock (TXUSRCLK or RXUSRCLK) and the parallel data width. Instead, on the return channel, the constant line rate at 625 Mbps and the data frame (consisting in 16-bit words 8B/10B internally encoded) set the values of the TXUSRCLK and RXUSRCLK clock signals to 31.25 MHz.

Fig. 3.14 shows the Statistical Eye produced by the tool for two different tested physical layers. The test was executed by using a pair of one meter long SMA coaxial cables, resulting in the wide open eye diagram shown in Fig.a; the same test was performed by using a miniSAS [66] differential cable, largely used for high line rate transmission for storage systems. Test on a four-meter miniSAS cable exhibits a an eye diagram more closed (b). The X-axis scale shows the percentage of U.I., while the Y-axis shows the vertical offset coded with a 8-bit word, over the range spanning from −127 to 127, allowing the comparison between dif-

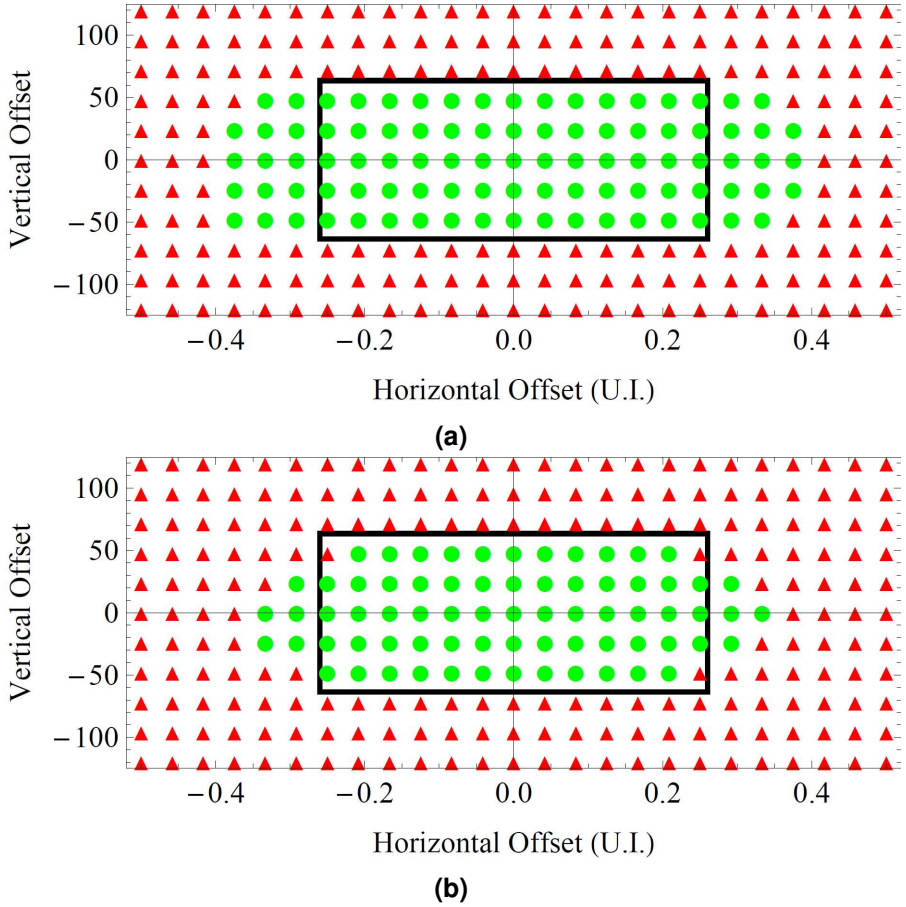


Figure 3.15: Pass/Fail map produced by the line rate validation logic, for an SMA cable (a) and for a miniSAS cable (b).

ferent transmission standards; the Data Sample point is identified by the (0,0) coordinate. Both plots were acquired with a 1.25 Gbps line rate and the different quality of the physical layers is clearly visible by comparing the two plots.
Fig. 3.15 shows the Pass/Fail map obtained from the elaboration of the Statistical Eye data (acquired for 1.25 Gbps line rate) for the SMA (a) and for the miniSAS (b) cables. In both the figures, the black rectangle identifies the *keep-out* region, the triangles represent the Offset points for which the BER is greater than 10^{-8}, while the circles indicate where the BER is less than 10^{-8}. One or more triangles inside the *keep-out* region indicate the link doesn't pass the test for that particular line rate and physical layer. A simple analysis of the two plots reveals how the link under test passed the qualification test, for a 1.25 Gbps line rate, using the SMA cables but not the miniSAS ones.

A further analysis can be conducted on the link, thanks to the ability of performing a Bathtub plot with the tool developed.
Indeed, using the Mathematica software, it is possible to conduct a statistical analysis of the data saved on the PC. As example, the Bathtub plot produced by this

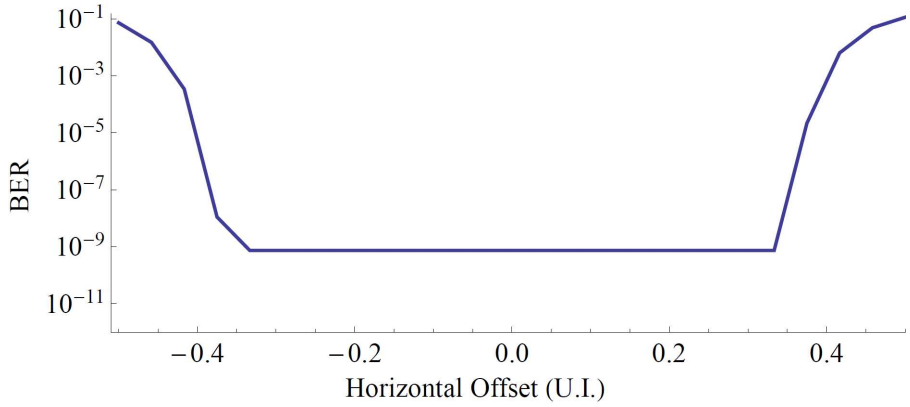


Figure 3.16: Bathtub plot at the zero vertical offset for SMA cables.

software is shown in Fig. 3.16, where SMA cables are used. The plot shows the BER values, over the U.I., at the zero value of the vertical offset.

Fig. 3.17 shows the output window of the adaptive link user console, which informs the user about the different frequency scan performed and about the maximum sustainable line rate of the serial link under test.

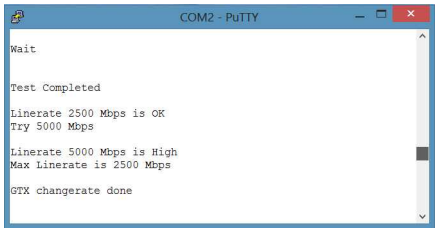


Figure 3.17: Output window of the adaptive link user console.

4

ATLAS

Founded in 1954, the CERN laboratory is based on the Franco-Swiss border near Geneva. It was one of Europe's first joint ventures and now has 21 member states. The name CERN is derived from the French *Conseil Européen pour la Recherche Nucléaire*, or European Council for Nuclear Research, a provisional council founded in 1952 by 12 European government with the mandate of establishing a world-class fundamental physics research organization in Europe. At that time, pure physics research concentrated on understanding the inside of the atom, hence the word "nuclear". At this moment, the understanding of matter goes much deeper than the nucleus, and the main area of research at CERN is particle physics, the study of the fundamental constituents of matter and the forces acting between them.

At CERN, the European Organization for Nuclear Research, physicists and engineers are probing the fundamental structure of the universe. They use the world's largest and most complex particle accelerator to study the basic constituents of matter or the fundamental particles. The particles are made to collide together at close to the speed of light. The process gives the physicists clues about how the particles interact, and provides insights into the fundamental laws of nature.

The instruments used at CERN are purpose-built particle accelerators and detectors. Accelerators boost beams of particles to high energies before the beams are made to collide with each other. Detectors, like ATLAS, observe and record the results of these collisions.

4.1 THE LARGE HADRON COLLIDER

The Large Hadron Collider (LHC) [67], built between 1998 and 2008, lies in the circular tunnel 27 km in circumference, previously occupied by the Large Electron-Positron Collider (LEP) which was shut down in November 2000. The tunnel is buried around 50 m to 180 m underground, between the Swiss and French borders on the suburb of Geneva.

LHC is a proton-proton (pp) collider which delivered collisions at $\sqrt{s} = 7\,TeV$ in 2010 and 2011, and at $\sqrt{s} = 8\,TeV$ during 2012. During the test collisions in preparation of the second operational run, it reached a total energy of $\sqrt{s} = 13\,TeV$.
One of the crucial parameters for the discovery power of a particle collider is the *instantaneous luminosity* which is proportional to the event rate $\frac{dN}{dt}$:

$$\frac{dN}{dt} = \mathcal{L} \times \sigma, \tag{4.1}$$

where σ is the cross section of the considered process.
The instantaneous luminosity of a collider depends on its intrinsic features, according to the formula:

$$\mathcal{L} = \frac{N_p^2 f k}{4\pi R^2}, \tag{4.2}$$

where N_p is the number of protons in each bunch, f is the revolution frequency of the protons in the accelerating ring, k is the number of bunches circulating in the beam and R is the mean radius of the proton distribution on the plane orthogonal to the beam direction. The instantaneous luminosity delivered by the LHC in 2015

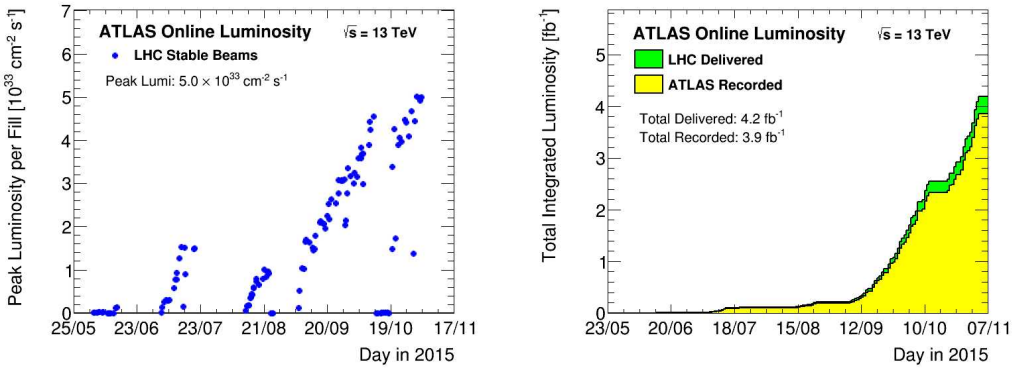


Figure 4.1: The integrated luminosity as a function of time delivered by LHC (a) and recorded by ATLAS (b) in 2012 and 2015.

reached the value of $5 \times 10^{33}\ cm^{-2}s^{-1}$ as Fig. 4.1 (a) shows, while the design peak luminosity is $1 \times 10^{34}\ cm^{-2}s^{-1}$.

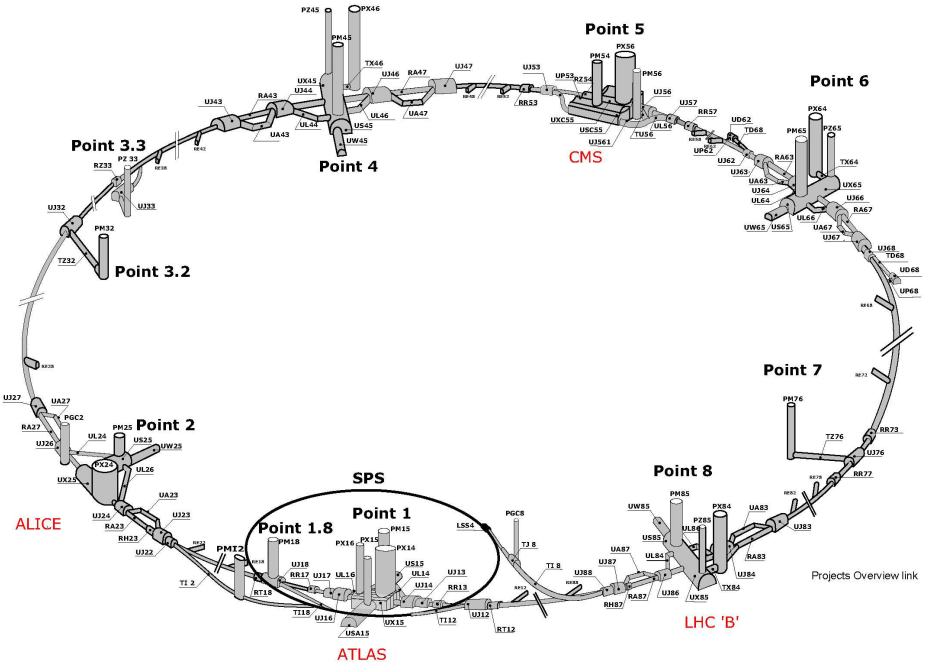


Figure 4.2: The chain of LHC collider and the different beam's collision points with their corresponding experiment.

During the 2015 runs, the integrated luminosity (the cumulative luminosity versus time) is showed in Fig. 4.1 (b).

In order to achieve the final energy, the beam goes through an acceleration chain. The different steps in the acceleration chain and the different positions of the LHC's experiments are showed in Fig. 4.2: after their production, the protons start the acceleration process in the linear accelerator, Linac2, where they reach an energy of 50 MeV; following in the chain, there is the Proton Synchrotron Booster which accelerate the protons to an energy of 1.4 GeV and the Proton Synchrotron (PS) where they reach 26 GeV; the Super Proton Synchrotron (SPS), which is the last step, before the injection into the LHC, raises their energy up to 450 GeV. Once there, the protons are accelerated in the two opposite directions. During the first years of operation, the LHC increased the proton energies from 3.5 TeV (2011) to 4 TeV (2012) and 6.5 TeV (2015).

Since LHC accelerates two beams of same sign particles, two separate accelerating cavities and two different magnetic fields are needed: LHC is equipped with 1232 superconducting magnets and 16 radiofrequency cavities which bend and accelerate the proton beams in the two parallel beam lines in the machine. The magnetic field used to bend such energetic proton beams is of 8 T and to reach such a magnetic fields the superconducting magnets are cooled down to 1.9 K and a current of 13 kA circulates inside them.

The LHC provides collisions in four collision points along its circumference where detector experiments are located (Fig. 4.2): ALICE (*A Large Ion Collider-*

Experiment, [34]), ATLAS (*A Toroidal Lhc ApparatuS*, [35]), CMS (*Compact Muon Solenoid*, [38]) and LHCb (*Large Hadron Collider beauty*, [68]).
ATLAS and CMS are multi-purpose detectors: Precise SM measurements, the discovery of the Higgs boson and searches for new phenomena beyond the SM are the main tasks.
Instead ALICE and LHCb are focused on more specific studies: ALICE is designed to study the quark-gluon plasma produced in heavy-ions collisions[1], while LHCb focuses on the study of B-physics and CP-violation processes occurring in b and c hadron decays.
Further experiments are LHCf, designed to study the particles generated in the forward region of collisions which are relevant for the understanding of cosmic rays physics, TOTEM, which measures the total elastic and diffractive proton-proton cross section and MoEDAL which is searching for magnetic monopoles.

4.2 PHYSICS AT LHC

The principal purpose of LHC, and its experiments, is to find the missing pieces in the current description of the fundamental structure of matter, called the Standard Model. It describes the Universe as being made from twelve fundamental particles and four fundamental forces: gravity, electromagnetism, the weak and the strong nuclear force. The Standard Model was developed in the 70's, but it is a non complete theory: the most obvious of its open issues is gravity for which a widely accepted quantized theory does not even exist. However, other theories Beyond the Standard Model (BSM) are able to elegantly explain one or more of the shortcomings of the Standard Model. These theories often predict the existence of other particles which LHC was built to look for, too.

4.2.1 *The Standard Model*

All the currently knowledge on elementary particles physics is very successfully described by a theory called the Standard Model (SM). The SM is a renormalisable Quantum Field Theory (QFT) with an internal gauge symmetry group of $SU(3)_{col} \otimes SU(2)_L \otimes U(1)_Y$, where $SU(3)_{col}$ is the non Abelian symmetric group which describes the strong interaction, $SU(2)_L \otimes U(1)_Y$ is the weak isospin group which describes the unified electroweak theory, also called Glasgow-Weinberg-Salam Theory (GWS) [69–71]. In QFT, both matter and forces are particles described by the excitation of the respective quantum fields they represent.

In the SM, the particles are divided into fermions and bosons. Fermions are particles that make up all the known matter. Their spin (intrinsic angular momentum) is 1/2 and they are divided in two classes: leptons and quarks (tab. 4.1). There exist six flavours of quarks (up, down, strange, charm, bottom and top) and of leptons (electron, muon, tau and the neutral neutrinos: electron, muon and tau

1 The LHC is able to accelerate and collide lead ions at $\sqrt{s} = 2.76\,TeV$ per nucleon, and ions collisions are foreseen each year in the LHC program.

	Family			
	1	2	3	Charge
Leptons	$\begin{pmatrix} \nu_e \\ e^- \end{pmatrix}$	$\begin{pmatrix} \nu_\mu \\ \mu^- \end{pmatrix}$	$\begin{pmatrix} \nu_\tau \\ \tau^- \end{pmatrix}$	0 -1
Quarks	$\begin{pmatrix} u \\ d \end{pmatrix}$	$\begin{pmatrix} c \\ s \end{pmatrix}$	$\begin{pmatrix} t \\ b \end{pmatrix}$	$2/3$ $-1/3$

Table 4.1: Standard Model Fermions with their charge.

neutrino). Due to their symmetries, they are divided into three generations of weak isospin[2] doublets. For the quark, each doublets is composed by an up-type quark, with $I_z = 1/2$ and a down-type quark whose isospin is $I_z = -1/2$. In a parallel manner, also the leptons can be divided in families each with a charged lepton (e, τ, μ with $I_z = 1/2$) and a neutrino (ν_e, ν_τ,ν_μ) for which $I_z = -1/2$). Particles in different generations differ in mass and flavour number but they have the same properties.

No mixing between the three generations of charged leptons has been observed so far, while for the quarks the Cabibbo-Kobayashi-Maskawa (CKM) matrix [72, 73] describes their mixing. Also for neutrinos, the Pontecorvo-Maki-Nakagawa-Sakata (PMNS) matrix [74, 75] describes the mixing between the three generations and implies massive neutrinos.

Quarks are observed in nature only in bound states (hadrons), but not like free quarks (*confinement*). Possible states consist of three quarks (baryons), a quark and an anti-quark (mesons) or probably combinations of these two options like the recently discovered pentaquark states [76]. The strong charge is called colour charge, it can take on the values red, green or blue or one of its anticolour. Due to confinement, only colourless singlet states can freely propagate. Thus, a meson is made up by a quark whose colour charge is the anti colour of the anti-quark. In a similar manner the three quarks in a baryon are combined in such a way that their different (anti) colours charges are colour neutral.

The particles responsible for any interaction between fermions, are called force carriers or gauge bosons with spin 1. Depending which boson is exchanged, a different type of interaction occurs. There are four such force carriers of the SM:

- the photon (γ) which is responsible for electromagnetic interactions and couple to the electric charge. It doesn't carry colour charge nor electric charge, and they do not couple to each other. Since the photon mass is zero, the interaction is long-ranged;

- the gluon (g) is responsible for the strong interaction. Gluon carries no electric charge, but always a colour and an anticolour charge. It couples to the

2 Weak isospin, I, is a quantum number relating to the weak interaction and, like normal spin, is associated with an $SU(2)$ symmetry. Since only one of the generators I_j (corresponding to the Pauli matrices) of $SU(2)$ can be simultaneously diagonalised with weak isospin I, it is customary to only denote the third component, I_z.

colour charges, that's why leptons do not feel the strong interaction. Since only colourless singlet state particles can freely propagate, the force can not be carried over long distances and therefore is only very short-ranged;

- the charged $W^{\pm}$ and neutral Z are responsible for the weak interaction. Since they are heavy bosons, the weak interaction is limited to a very short range.

- the graviton[3] should be responsible for the gravitational force. It should have spin 2, and because of its infinite range, it must be massless.

The force carriers of the SM and their most important properties can be seen in Tab. 4.2. All the fermions are able to interact via the weak force with each other, while for their electronic neutrality, neutrinos are not allowed to interact through electromagnetic force. Since only quarks and gluons carry a colour-charge, only quarks are able to interact via the strong force.

Interaction	Simbol	Coupling Strength	Couples to
Electromagnetic	γ	$1/137$	Electric Charge
Weak	W^+, W^-, Z	10^6	Flavour
Strong	g	1	Colour
Gravitational	Graviton	10^{-39}	Mass

Table 4.2: The fundamental forces and their carriers.

The latest confirmed particle of the SM as we know it today is the Higgs boson. It is responsible for giving mass to the otherwise massless weak gauge bosons through the Higgs mechanism [77, 78], an electroweak symmetry breaking. For each particle in the SM, there is a corresponding anti-particle with same mass but opposite quantum charges. Some particles, such as the photon and the Z boson, are invariant under charge conjugation which implies that they are their own anti-particle.

Although SM is one of the best experimentally verified theories, it lacks several aspects which cannot be explained in its framework [79]. The first great shortcoming is its inability to describe gravitation as the fourth fundamental force. Another limit, relies in its free parameters and, moreover, the rather complicated unitary product group representation of $SU(3)_{col} \otimes SU(2)_L \otimes U(1)_Y$ including the peculiar way of the electroweak mixing. The Higgs mechanism does well give a mathematical description of mass within the SM, but at the price of another accompanying free parameter for each fermion. Hence, the Standard Model can be considered an effective field theory. The combined effort of theoretical and experimental particle physicists seeks for more general theories (Grand Unified Theory, GUT [80, 81]) and their verification in an experimental setup. The SM may, however, be valid up to the GUT energy scale (10^{16} GeV).

3 The graviton was hypothesised but it is not observed, yet.

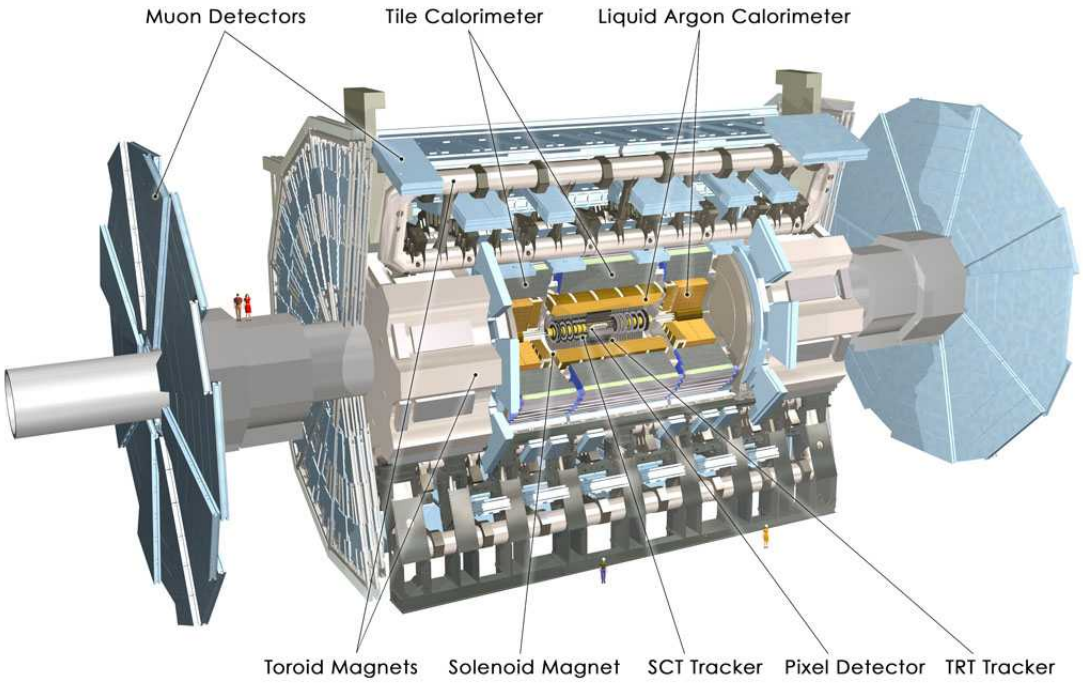


Figure 4.3: The ATLAS Detector: all the subdetectors it is composed of are shown.

4.3 THE ATLAS DETECTOR

ATLAS [35], short for A Toroidal LHC ApparatuS, is one of the four main experiments recording the collisions provided by the LHC. It is 20 m tall and 45 m long and weights more than 7000 tons.
ATLAS consist of two magnet systems and a number of subdetectors (Fig. 4.3) whose partial information is later combined to form a full picture of what happened during a collision. In the following sections details about the structure of the subdetectors will be given.

The structure has a cylindrical shape centred at the interaction point with its axis along the beam line. It is composed of several concentric subdetectors, each of them extracting various features from the particles generated in the pp collision as they fly from the centre of the detector to the outer part, as shown in Fig. 4.4. From the innermost to the outermost layer, the ATLAS experiment is composed of (Fig. 4.3):

- An inner detector: a tracking system to detect charged particles and measure their momentum and direction.
- A solenoidal superconducting magnet: providing a uniform magnetic field along the beam axis in which the inner detector is immersed.
- An electromagnetic calorimeter to measure the energy deposited by electrons and photons.

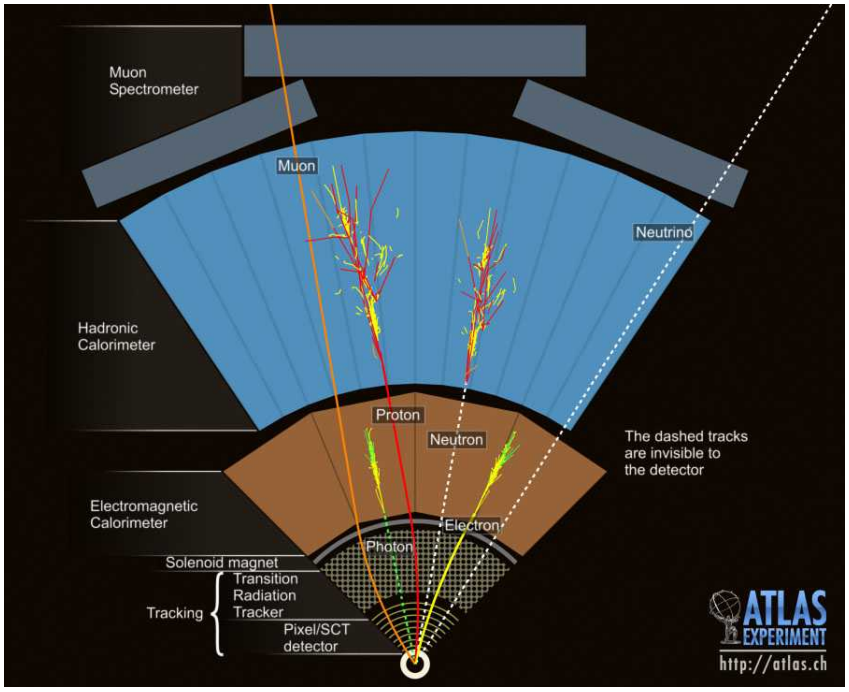


Figure 4.4: Schema of the detection of the particles produced in a proton collision while they travel through the several layers of the ATLAS detector.

- An hadronic calorimeter to measure the energy deposited by hadrons.
- An air-cored superconducting toroidal magnet system which provide the magnetic field to the muon spectrometer.
- A muon spectrometer, that is a tracking system for the measurement of muons as they travel throughout all the detector and are the only particles reaching the outer part.

The ATLAS coordinate system is a cartesian right-handed coordinate system (Fig. 4.5), with the nominal collision point as the origin, the z axis in the same direction of the beam pipe, the x axis pointing the centre of LHC and the y axis going upwards.

Due to its cylindrical symmetry, polar coordinates are usually used: the azimuthal angle ϕ is defined around the beam axis, in the $x-y$ plane, while the polar angle θ is the angle from the z axis in the $y-z$ plane. The θ variable is not invariant under Lorentz transformation, and so instead of the θ angle the *pseudorapidity*[4] η is used:

$$\eta = -\ln\left[\tan\frac{\theta}{2}\right] \tag{4.3}$$

4 Actually the real invariant variable is the *rapidity* y: $y = \frac{1}{2}\ln\frac{E+p\cos\theta}{E-p\cos\theta}$. In the ultra-relativistic limit the rapidity y can be substituted with the pseudorapidity η

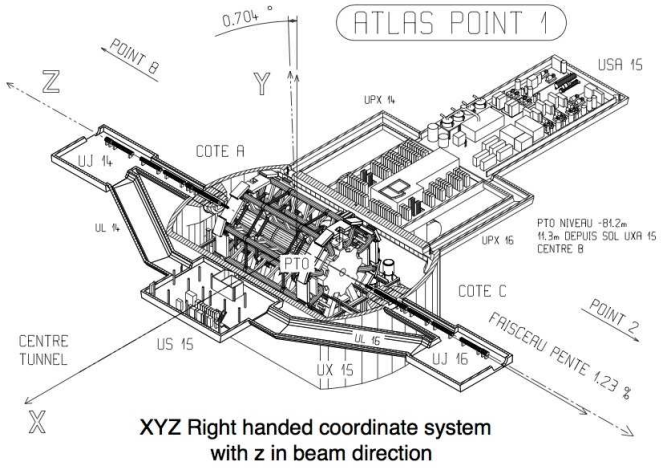


Figure 4.5: Reference system used in ATLAS.

4.3.1 *ATLAS Magnets*

The ATLAS detector is equipped with two different kind of magnets: a superconducting solenoid [82], providing a magnetic field to the inner tracking system, and a system of air-core superconducting toroidal magnets in the barrel [83] and the end-cap region [84] which are located in the outer part of the detector as shown in Fig. 4.6.

The solenoid covers the central region of the detector, provides an uniform magnetic field of approximately 2 T for the inner detector. The direction of the magnetic field, which goes parallel to the beam pipe, along the z axis, is able to bend particles in the transverse plane and in such a way for allowing the inner tracking system to measure their transverse momentum.

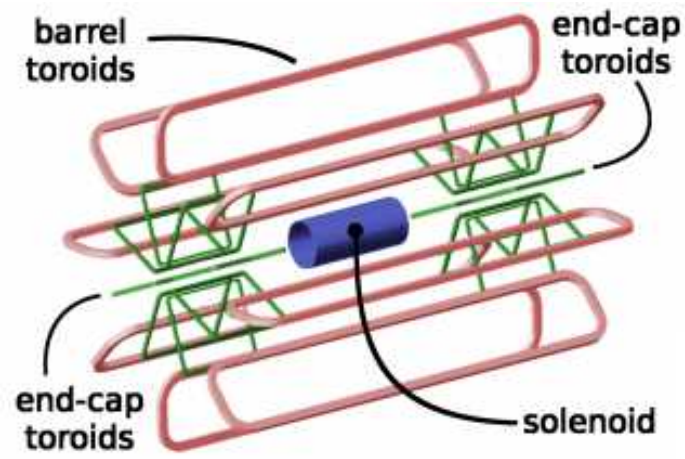


Figure 4.6: The magnetic system of the ATLAS detector: it consists of a central solenoid magnet and of toroid magnets in the external parts.

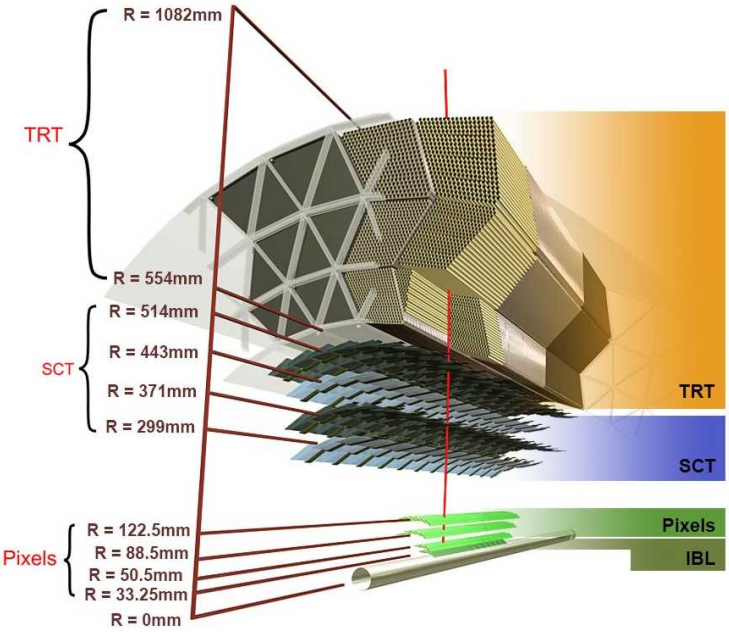


Figure 4.7: The ATLAS Inner Detector tracker: the three subdetectors (the Pixel Detector, with the fourth Insertable B-Layer (IBL), the Semiconductor Tracker and the Transition Radiation Tracker) are shown as well as their radial dimensions.

The toroid is one of the peculiarities of the ATLAS detector: it is located outside of the calorimetric system and covers the region $|\eta| < 3$, considering all its subparts. Indeed, it consists of eight coils in the barrel region (which are 25.3 m long and have a diameter of 20.1 m) and eight smaller coils on each end cap side (5 m long and 10.7 min diameter). The magnets provide a magnetic field whose field lines form an approximate cylinder around the beam pipe and whose peak intensities are 3.9 T in the central region of the detector and 4.1 T in the forward region.

The aim of such a toroid is to have a large lever arm to improve the measurement of the muon transverse momentum, and it is built "in air" in order to minimize the muon multiple scattering within the detector.

The ATLAS double magnetic system has been designed to provide two independent measurements of the muon transverse momentum in the inner detector and in the muon spectrometer, thus ensuring good muon momentum resolution from few GeV up to the TeV scale.

4.3.2 *The Inner Detector*

The ATLAS Inner Detector tracker (ID), shown in Fig. 4.7, is composed by three concentric cylindrical subdetectors [85, 86]. Its axis is centred on the z axis and it is approximately 6 m long and 2 m wide, covering the region ($|\eta| < 2.5$). A charged particle passing through these detectors will leave a series of localised energy deposits, *hits*, from which the trajectory of the charged particles, produced

in the pp collision, can later be deduced. From the curvature of the tracks, other features can be extracted: the transverse momentum (thanks to the magnetic field provided by the superconducting solenoid described in the previous section), the vertex position (the eventual secondary vertexes due to long-lived particles) and the impact parameter.

The three sub-detectors into the ID (Fig. 4.7), are:

- **Pixel Detector:** it is composed of three layers of silicon pixels, which provide high-precision track measurement since the spatial resolution on the single hit is ~10 μm in the ϕ coordinate and ~ 115 μm along the z coordinate. During the shutdown between Run I and Run II, a new subdetector, the Insertable B-Layer (IBL) [87], was installed, adding a fourth pixel layer closer to the interaction point and thus improving the resolution of both tracking and impact parameter estimation. The IBL is 64 cm long and extends from $R = 31$ mm to $R = 40$ mm. The primary reason for the IBL is the harsher conditions of Run II. The better resolution on tracking and vertexing provided by the IBL will result in a better momentum resolution which is important for triggering and for the background reduction.

- **Semiconductor Tracker (SCT):** it is the second high-precision detector of the ATLAS inner tracker. It is composed of four barrel with silicon strips on both sides (for a total of eight layers) with a spatial resolution on the single hit of 17 μm in ϕ and 580 μm along z. Both the two SCT end-cap detectors consist of nine discs with silicon detectors on either side. The Pixel Detector and the Semiconductor Tracker together provide on average eight high-precision hits per track.

- **Trasition Radiation Tracker (TRT):** it is the outermost part of the inner detector and it is composed of straw tubes chambers. The resolution of such a detector is lower than the previous one (~130 μm per straw), but it is compensated by the high number of points per track (36 on average) that it can provide. Further, the TRT provides discrimination between electrons and other charged hadrons by means of the transition radiation produced among the straws. Electrons typically have a Lorentz gamma factor much higher than any other particles, easily exceeding the threshold for this TR emission. Being radiated in a forward cone, the transition radiation photons (soft x-rays) enter with the particle and being absorbed in the gas, causing additional ionisation.

4.3.3 *The Calorimetric System*

The ATLAS calorimeters are represented in Fig. 4.8. It is composed by two detectors: an electromagnetic calorimeter used to measure the energy of photons and electrons lost by electromagnetic interactions, and a hadronic calorimeter which measure the energy deposited by hadrons after their interactions (strong) with the material.

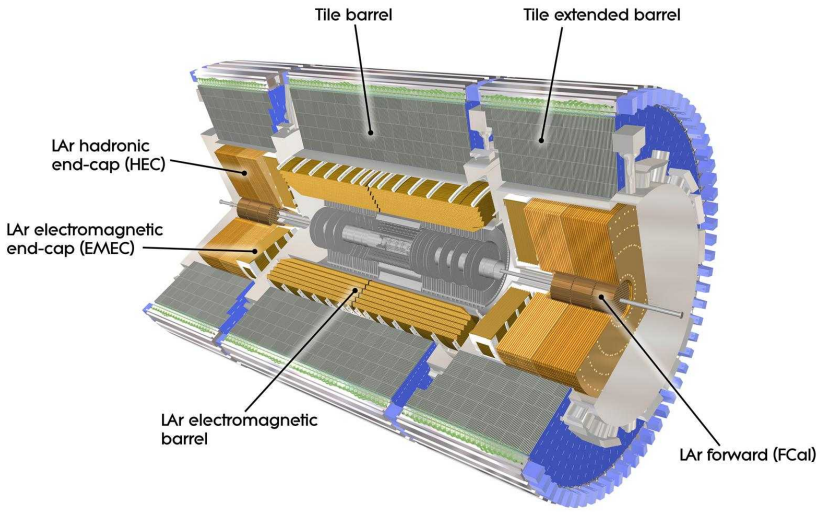


Figure 4.8: The ATLAS calorimetric system: the electromagnetic calorimeter made of liquid Argon and Lead and the hadronic calorimeter, whose composition varies as a function of η.

ATLAS uses sampling calorimeters which are done by alternating layers of absorber and active material. In the absorber medium, characterised by a short radiation length, X_0, and similarly short nuclear interaction length, λ_0[5], particles showers are produced. The active material is optimized for absorbing the produced particles and radiation.

The ATLAS calorimeter has a cylindrical shape centred around the interaction point with its axis lying on the ATLAS z axis. It is long about 13 m with an external radius for the electromagnetic calorimeter of 2.25 m and 4.25 m for the hadronic one.
The entire system ensure that high energy particles completely deposit their energy in the calorimeters.

The Electromagnetic Calorimeter

The Electromagnetic Calorimeter [88] of the ATLAS experiment covers the region up to $|\eta| < 3.2$. It has an accordion structure (Fig. 4.9) made of lead (whose thickness varies as a function of η in order to maximise the energy resolution) which is immersed in Liquid Argon, LAr, used as active material. This structure confers to the calorimeter very high acceptance and symmetry in the ϕ coordinate and is segmented longitudinally into three layers, called strips, middle and back.

5 Small values for the two parameters X_0 and λ_0 are preferred in order to minimise the number of energetic particles which are able to penetrate the calorimeter and enter the muon detectors: the called *punch through particles* effect.

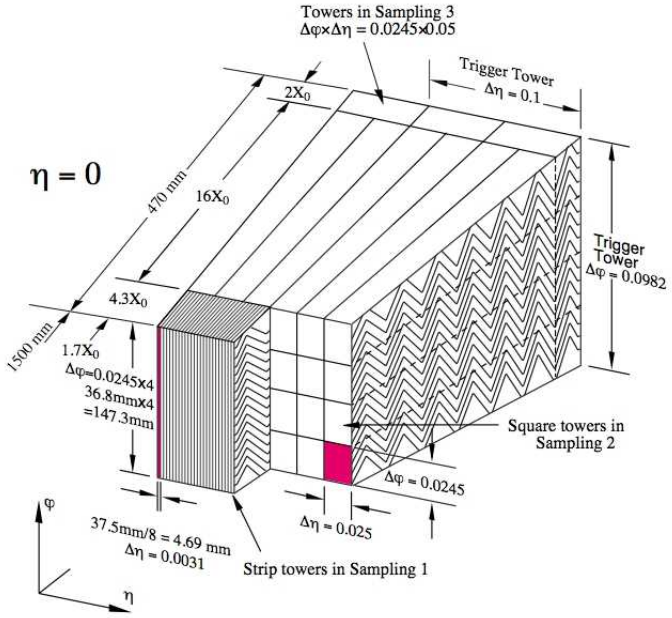


Figure 4.9: The accordion structure of the electromagnetic calorimeter and its radial segmentation.

The Hadronic Calorimeter

The Hadronic Calorimeter covers the region $|\eta| < 4.5$, and it is realized with a variety of techniques as a function of η like it is possible to check in Fig. 4.8.

The central region [89] is called *Tile Barrel* ($|\eta| < 1.7$), and it is made of alternating layers of iron, used as absorber, and scintillating tiles as active material.

The *endcap* region ($1.7 < |\eta| < 3.1$) has an accordion structure equipped with LAr and lead, as the Electromagnetic Calorimeter, while the *forward* region ($3.1 < |\eta| < 4.5$) uses a simpler geometry and copper disks as absorber medium [90]. This variety of materials and structures is due to the different radiation hardness required in the different parts of the detector.

4.3.4 *The Muon Spectrometer*

The layout of the ATLAS Muon Spectrometer [91], MS, is shown in Fig. 4.10. It is instrumented with both trigger and high-precision chambers in order to trigger and track the charged particles (especially muons) exiting the calorimeter. It is immersed in the magnetic field provided by the toroidal magnets which bends the particles along the η coordinate, and it allows to measure the muons p_T in the region $|\eta| < 2.7$.

The chambers used to reconstruct the muon track and measure their momentum are of several types depending on the η region, in order to face the different rate conditions present in the different parts of the detector.

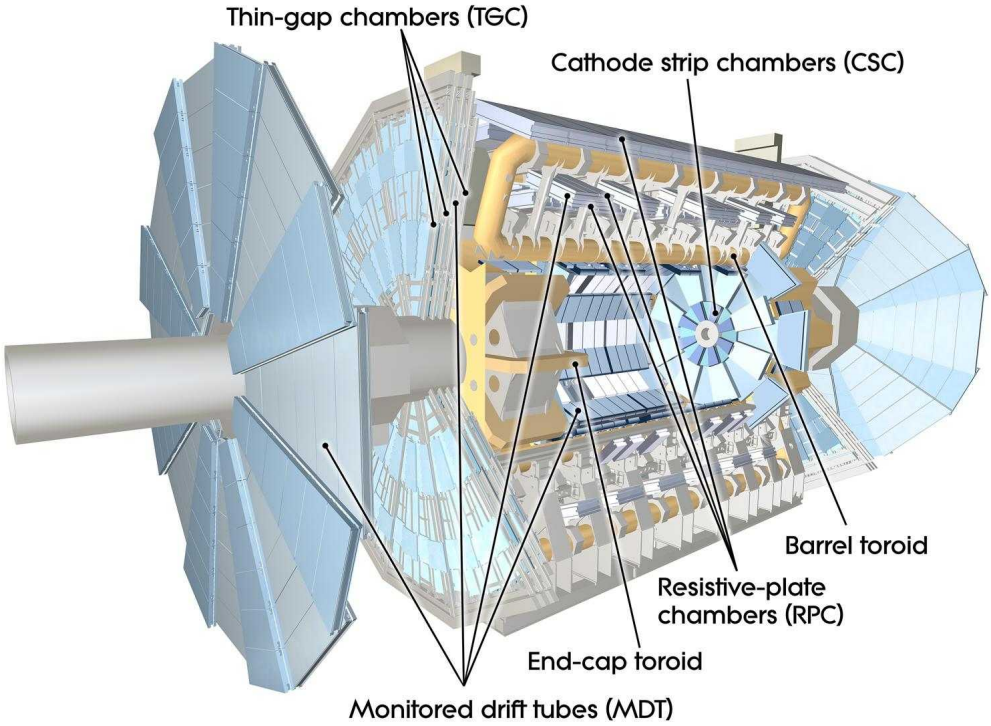


Figure 4.10: The ATLAS Muon Spectrometer.

In the barrel region ($|\eta| < 2.7$) and in the innermost end-cap layer ($|\eta| < 2$) Monitored Drift Tubes (MDTs) are used. The MDT chambers are arrays of aluminium drift tube detectors of 30 mm diameter and 400 µm thickness, with a 50 µm diameter central wire. The tubes are filled with a mixture of Argon and CO_2 at high pressure (3 bars), and each tube has a spatial resolution of 80 µm.
At higher pseudo-rapidity ($2 < |\eta| < 2.7$), where the background is much larger, the higher granularity of the Cathode Strip Chambers (CSC) is used. CSC chambers are multiwire proportional chambers in which the readout is performed using strips forming a grid on the cathode plane in both orthogonal and parallel direction with respect to the wire. The spatial resolution of the CSC is about 60 µm.

The trigger system uses the Resistive Plate Chambers (RPCs) in the barrel region ($|\eta| < 1.05$) and the Thin-Gap Chambers (TGCs) in the end-cap region ($1.05 < |\eta| < 2.4$).
The RPC consists of two high resistivity plates separated by a thin layer of ionizable gas. High voltage is applied and the signal, induced by an ionising muon passing through the gas, is read out using strips mounted on the outsides of the resistive plates. The spatial resolution provided is not so high ($\mathcal{O}(\text{cm})$), but timing is fast,($\mathcal{O}(\text{ns})$).
The TGCs are multiwire proportional chambers, but designed for high timing resolution (4 ns) and they also have a good spatial resolution 2-7 mm. This is the reason why they are used to provide muon triggers but also to complement the

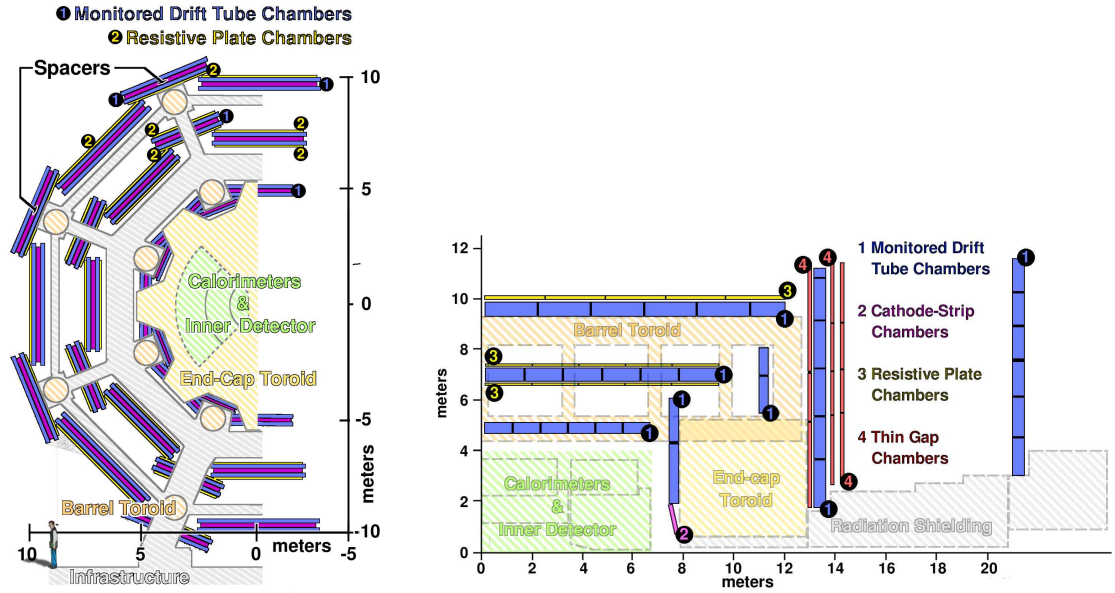


Figure 4.11: The cross-section of the barrel muon system perpendicular to the beam axis showing three concentric cylindrical layers (a). The cross-section of the muon system in a plane containing the beam axis (b).

end-cap MDTs by providing a measurement of the azimuthal component of the track coordinate.

As shown in Fig. 4.11, in the central region the MS is arranged on a three layer or stations, with a cylindrical structure, whose radii are 5, 7.5 and 10 m; while in the forward region the detectors are arranged vertically, forming four disks at 7, 10, 14 and 21-23 m from the interaction point.

4.3.5 *Trigger and Data Acquisition System*

The primary role of ATLAS trigger and data acquisition system [92], TDAQ, is providing a fast, efficient and physics motivated selection of the collision events to be recorded. Indeed, since the rate of events produced by the LHC is far too high to be recorded, trigger is in charge to inspect among all the events and decide what is *interesting*.

The ATLAS trigger system (4.12) has a two level structure:

- the Level-1 (LVL1) trigger: a number of hardware custom processors close to the detector hardware which makes a first rough selection;
- the High Level Trigger (HLT): a number of software triggers running on farms which takes the final decision based on the full granularity information coming from all detector data.

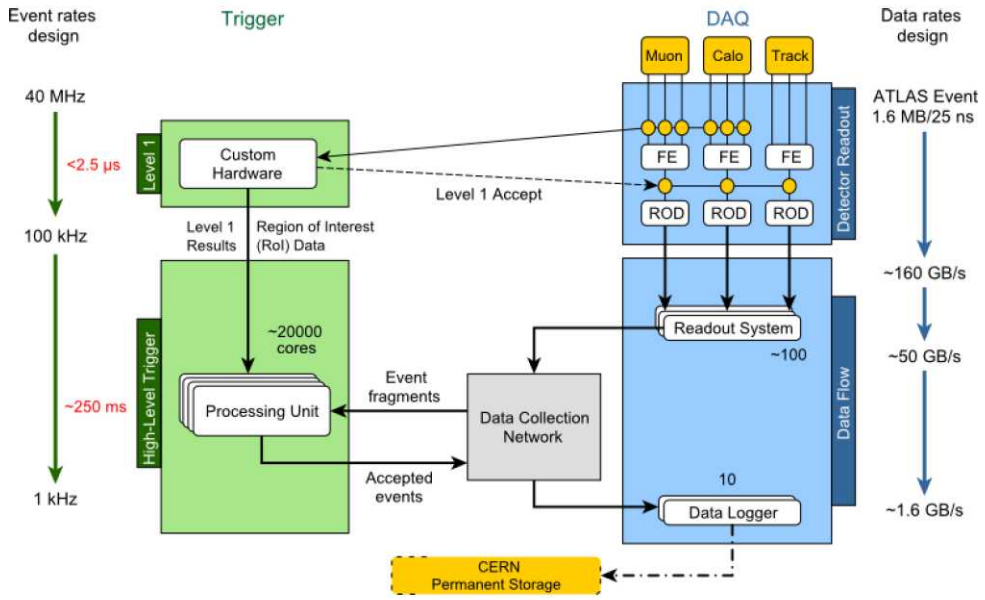


Figure 4.12: Main structure of the ATLAS trigger system: it is made of two levels, each improving the measurement of the previous levels also combining informations from different subdetectors.

The LVL1 works synchronously with the LHC bunch crossing rate (40 MHz): every 25 ns it receives data collected by the calorimetric system and the muon spectrometer and processes them with a maximum latency of 2.5 µs(Level-1 latency). For each accepted event, a list of so-called *Regions of Interest* (RoIs) are sent to the HLT in the form of η and ϕ values; which identify the regions of the detector in which interesting activity has been detected.
The L1 is designed to take a decision on the event in 2.5 µs and its output is, and The event rate is reduced to ~100 kHz from a starting value of 40 MHz.

The HTL is completely software-based. Having access to data coming from all the different subdetectors, it can perform a full reconstruction of the event and take the final choice of permanently recording the full event data into the CERN permanent storage facility.
The operational parameters of the HLT is an accept rate of 1 kHz at an average latency of O(200 ms).

4.3.6 *The Detector Upgrades*

Although with current data the Higgs-Boson was discovered, more data is required to perform precision measurements of its properties. Besides, Higgs-physics is not the only reason for collecting further data. Many other models, especially those of Super-Symmetry, predict very rare processes and thus seek for very high luminosity for their experimental investigation. Therefore, a series of shutdowns with dedicated upgrades were scheduled for LHC. A temporal

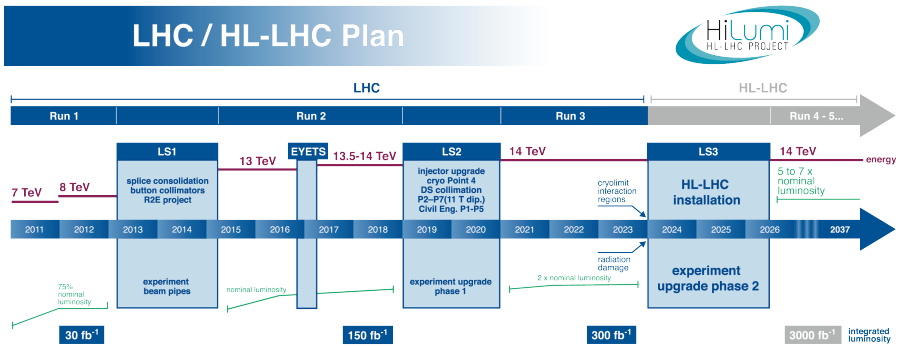


Figure 4.13: Updated status of the LHC baseline programme.

overview of the expected program is given in Fig. 4.13. This overview includes the three detector upgrades and the periods of data taking.

The first Upgrade (Phase-0 Upgrade) is already done: a new pixel layer (IBL) was installed in the pixel detector and a topological trigger processor (L1Topo) was added in the L1 Trigger system. Both these changes impacted the structure of the first level of the TDAQ system.

In the second step of the ATLAS Upgrades (Phase-I Upgrade) the most remarkable improvements of the detector will involve the introduction of new detectors in the End-Cap region (the New Small Wheel, NSW project) and the completion of the Level 1 trigger upgrade which took place between Run I and Run II. Since, the second part of this PhD work is to be intended as a part of these upgrades a deeper description of the L1 Trigger system and of the NSW project will be given in the next section.

The last Upgrade foreseen for LHC, and for the experiments installed on it, is called High-Luminosity Large Hadron Collider (HL-LHC). The project aims at increasing the luminosity of the machine by a factor of 10, beyond the LHC's design value, up to 10^{35} $cm^{-2}s^{-1}$. The upgrade will provide better chances to see rare processes and improve statistically marginal measurements. However, with the increased luminosity and pile-up, the upgrade of the experiments is mandatory. In particular, for ATLAS, a (partial) replacement of the inner tracking systems (the most radiation damaged detectors) is planned such as the electronics in various parts of the calorimeters and muon spectrometers. Besides, also the TDAQ systems will need an upgrade for coping with the new rates aspected in the HL-LHC.

5

SERIAL LINKS FOR ATLAS TDAQ UPGRADES

The Phase-I upgrade of the ATLAS Trigger and Data Acquisition (TDAQ) system will allow the ATLAS experiment to efficiently work at a higher instantaneous luminosity (up to three times the original LHC luminosity). Different improvements will concern the Level-1 Trigger System. In the end-cap region, the inner wheel will be completely replaced by the New Small Wheel (NSW). The new signals from the NSW and from the outer layer of the extended barrel of the Tile Calorimeter will be included in the Level-1 muon end-cap trigger, aiming to reduce the overall rate by rejecting a large fraction of fake triggers. For the same purpose of reducing fake trigger rate in the overlap region between barrel and end- cap, also signals from the outer layer of the extended barrel of the Tile Calorimeter will be available for the Level-1 Trigger System. The Level-1 muon trigger will also benefit form the introduction of the new topological processor (L1Topo), which will combine the information of Calorimeter processors and the new Muon to Central Trigger Processor Interface, MUCTPI, adding geometrical details for a better trigger selection. Also the Data Acquisition System and the High-Level Trigger (HLT) processing farm will be upgraded to cope with higher amount of information to be read out and processed.

Since, as a part of this PhD work, two serial links were developed, for the Phase-I upgrade of the Level-1 muon trigger electronics, a description of the changes foreseen for the ATLAS detector will be provided. In the first section, the new board designed, for the Barrel Level-1 muon trigger system, and the high-speed serial link implemented will be described. In the second section, a description of the new Pad Trigger Board designed for the new off-detector electronics of the New Small Wheel will be provided and its serial interface will be outlined.

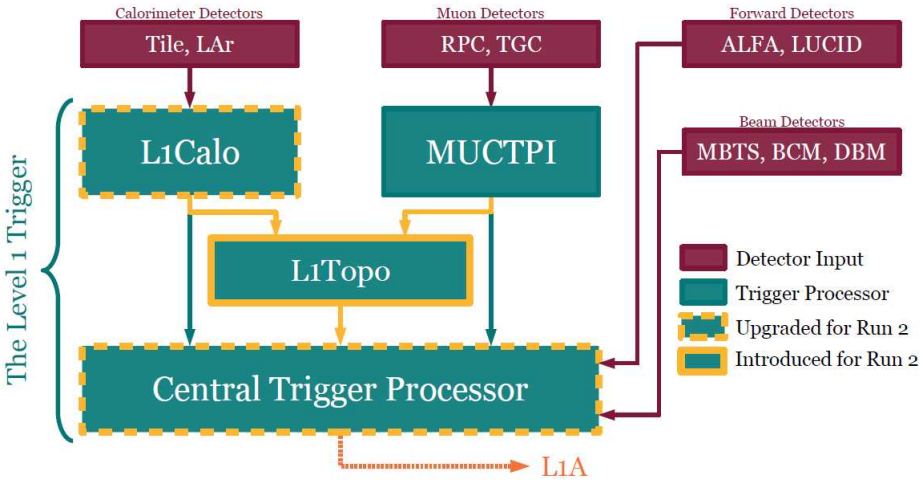


Figure 5.1: Schematic view of the Level-1 Trigger in the ATLAS detector.

5.1 A HIGH-SPEED SERIAL LINK FOR THE BARREL LEVEL-1 MUON TRIGGER

In the previous chapter, the ATLAS trigger and data acquisition system, TDAQ, was briefly introduced. A more detailed description of the Level 1 trigger will be given in this section, since part of this PhD project was focused on the design of a new interface board for the upgrade in the Barrel Level-1 muon trigger system and on the development of a high-speed serial link as a part of it.

5.1.1 *Level-1 Trigger*

The Level-1 trigger (Fig. 5.1) is composed of a number of dedicated trigger processors, the calorimeter trigger processor (L1Calo) and the Muon to CTP Interface (MUCTPI), each receiving data from the respective trigger detectors, calorimeter and muon detectors, at a constant rate of 40 MHz. As a part of ATLAS Phase-0 upgrade, an additional topological trigger processor (L1Topo) was installed. All these three processors provide inputs to the Central Trigger Processor, CTP, which is in charge of forming the Level-1 Accept signal, L1A. This signal, which is sent to all subdetectors as an electrical pulse, allows the readout data to be transferred from the pipeline of the detectors to the next stage into DAQ system. CTP decision is taken according to a trigger menu: a set of logical conditions on the input, which corresponds to signals produced by interesting physics processes.
Based on the calorimeter information, L1Calo identifies various trigger objects: electron, photon and tau/hadron candidates with transverse energy, E_T, above programmable thresholds. These quantities (type, thresholds, and multiplicity) are sent to the CTP every bunch crossing.
The MUCTPI, operating with six programmable p_T thresholds, counts the multiplicity of muon candidates above threshold. Despite of the L1Calo, the MUCTPI doesn't construct the trigger objects, but gets them directly from the RPCs and

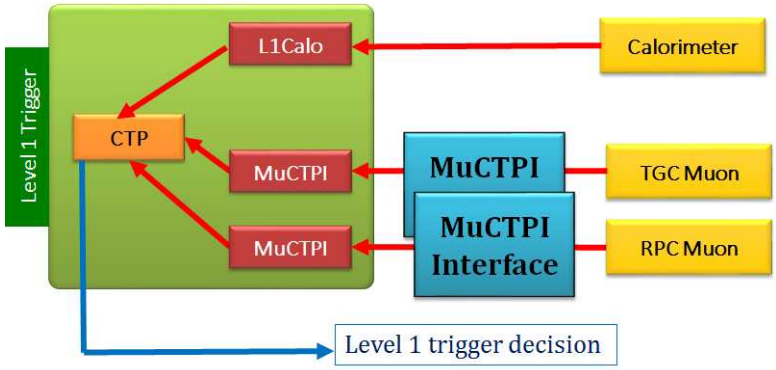


Figure 5.2: Arrangement of the MUCTPI Interface Board inside the L1 Trigger system.

the TGCs off-detector electronics via detector specific Interface Board, MUCTPI Interface (Fig. 5.2).

L1Topo uses information from both L1Calo and the MUCTPI to form trigger inputs to the CTP, which are based on geometric properties of the event rather than on sole multiplicities. For this purpose, full-granularity muon information has to be available to L1Topo. However, the present MUCTPI is not able to provide it, before the generation of the L1 Accept. To overcome this limitation, the design of a new MUCTPI trigger processor is foreseen as part of the Phase-I Level-1 trigger upgrade. As a consequence the MUCTPI Interface for the muon barrel trigger detectors will be upgraded. The new trigger board will be able to optically send trigger data from RPCs to the new MUCTPI board. The optical links will provide a higher bandwidth which will be used to transfer additional information from the Sector Logic modules, for example data for more than two muon candidates. This would allow the study of the events with more than two muon candidates produced in the same sector [93]

5.1.2 *The Barrel Trigger System*

The Barrel muon spectrometer is segmented, in the ϕ direction, in 16 different sectors. Fig. 5.3 shows such segmentation. Two kinds of sectors can be identified: the sectors occupied by the magnet coils, called "Small Sector" and the sectors between two adjacent coils, "Large Sector". Starting from this segmentation, the trigger system is segmented into 64 logic sectors, which are obtained dividing each of the 16 sectors of the spectrometer by four.

The barrel Level-1 muon trigger system uses fast geometric coincidence between the various planes of the Barrel Muon Spectrometer to detect the muons produced at the interaction point. The trigger algorithm uses the information coming from RPC detectors which have a spatial resolution of 1 cm and 1 ns of temporal resolution. In order to reduce the fake trigger rate due to the background, separate algorithms were performed in η and ϕ. As previously described RPC trigger chambers are arranged in three different concentric layers around the

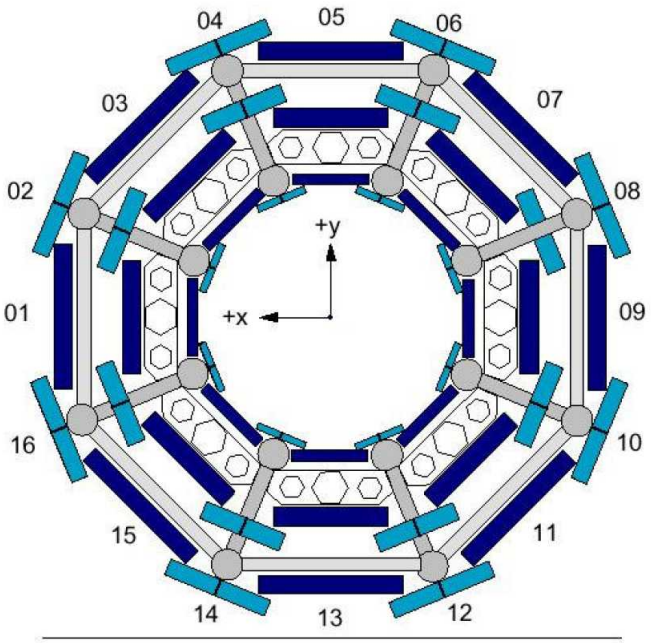


Figure 5.3: The segmentation in sectors of the spectrometer in the barrel region.

interaction point. Each detector chamber is read out by two series of strips along η and two along ϕ, for a total of six planes of RPC strips per direction.

The trigger algorithm is performed by dedicated hardware processors mounted on the detector (the Coincidence Matrix, CM, and the Pad Logic boards) and from the Sector Logic boards located in USA15 (an underground counting room far 80 meters from the detector).

Since the trigger algorithm is executed for low and high p_T, the trigger system is divided in low and high p_T Coincidence Matrix boards and low and high p_T Pad Logic boards. The schematic of the trigger principle is shown in Fig. 5.4.

For the low p_T algorithm, low p_T coincidence matrix boards perform a coincidence between the signals coming from the first trigger chambers plane (RPC1) and the pivot plane (RPC2). A valid trigger is generated if, after an hit in the RPC2 plane, the CMs find an hit also in the RPC1 looking inside a cone whose axis is on the line that conjuncts the hitted point in RPC2 with the nominal interaction point and whose vertex is on the RPC2 plane. The opening of the cone is determined by the selected p_T threshold. The low p_T coincidence matrix chips also store the data produced by the front-end electronics during the Level-1 latency period (2.5 μs).

For the high p_T algorithm, the high p_T coincidence matrix boards search for an hit in the third plane, RPC3, if a valid trigger was generated by the low p_T algorithm. Also the CM boards are responsible for storing the data from the front-end during the Level-1 latency period.
The low p_T Pad Logic boards collect the data from four coincidence matrix, generate the low PT trigger information and send it to the high p_T CM boards. Instead,

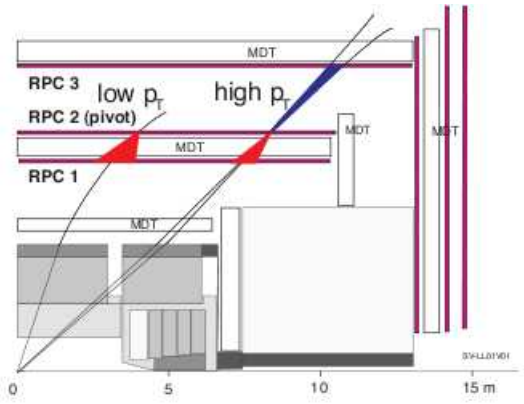


Figure 5.4: Schematic view of the first level muon trigger for the barrel in ATLAS.

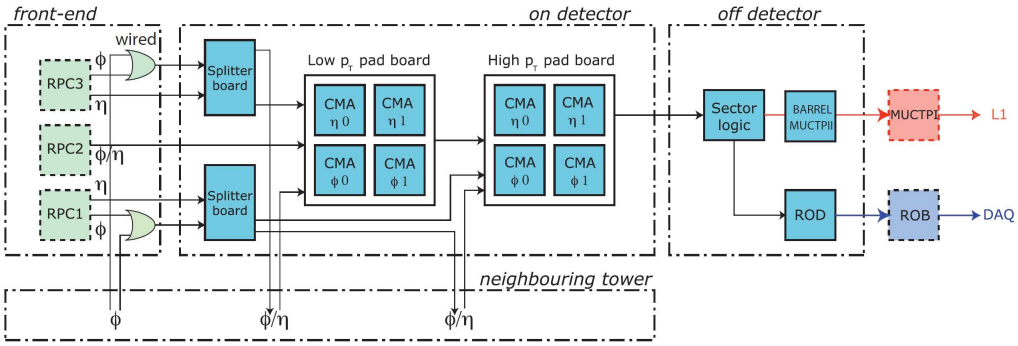


Figure 5.5: The scheme of the RPC Level-1 electronics.

the high PT Pad Logic Boards collect the data from the low p_T board and from four high p_T coincidence matrix and send these trigger informations to the Sector Logic boards.
The function of the Sector Logic board is to solve eta overlap between Pads and to choose the two muon candidates with the higher p_T to be sent to the Muon Central Trigger Processor Interface.
A simplified path of the readout data and of trigger information, for one trigger sector, is depicted in Fig. 5.5.

5.1.3 *The MUCTPI Interface Board*

Trigger data coming from the on-detector electronics related to the same trigger sector is transmitted by the Sector Logic to the MUCTPI Level-1 Trigger processor via a detector specific interface board, the MUCTPI Interface (Fig. 5.6). This Interface board is a 6U VMEx board installed in the ROD crates in the USA15 room, according to scheme depicted in Fig. 5.7.

Due to the large amount of data to be transferred, including high-speed serial links and timing signals sensitive to skew and distortion, the VMEx back-

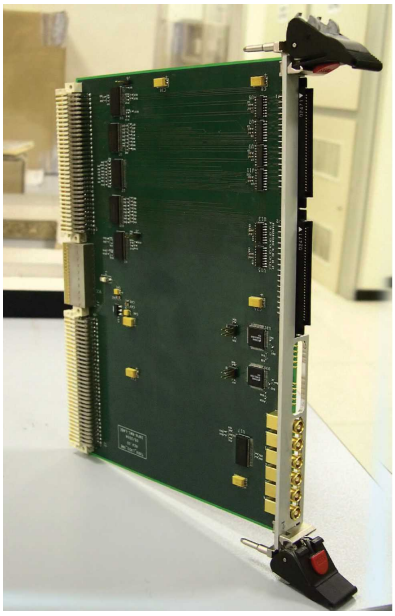


Figure 5.6: Current MUCTPI Interface board.

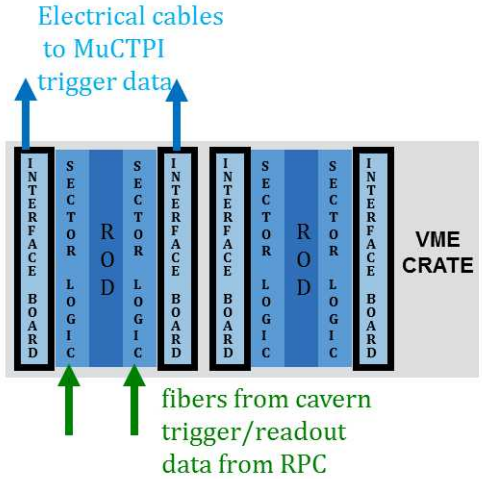


Figure 5.7: Arrangement of the different boards in a ROD crate in the USA15 room.

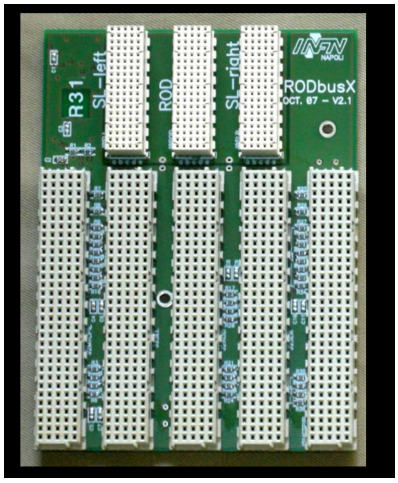


Figure 5.8: Photo of the RODbus, with the slots for ROD, Sector Logic and MuCTPI Interface.

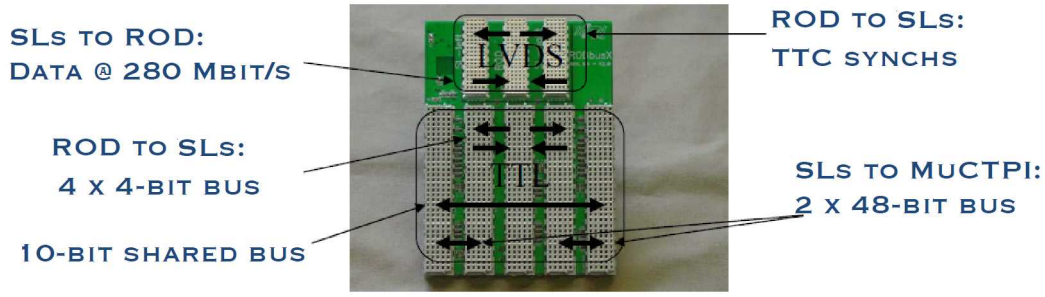


Figure 5.9: Different connections on the RODbus.

plane of the ROD crates is equipped also with the RODbus custom backplane. It connects one ROD, two Sector Logic boards and two MuCTPI Interface boards (Fig. 5.8), making possible the communication between the different boards. The ROD board [94] is the data processor of a full ATLAS sector, which takes care of the event building of data coming from the Sector Logic. As shown in Fig. 5.7, each ROD crate hosts two RODbus, thus being able to manage the data of a full ATLAS sector.
The RODbus module plugs on the rear-side on the VME64x backplane, and is equipped with a central (Jo) and a bottom (J2) connectors. The VME64x standard leaves the Jo pin-out fully definable by the user. On J2, TTL lines specified by the protocol coexist with user defined features. Fig. 5.9 shows the different connections on the RODbus. All the LVDS differential pairs, connecting the ROD with the adjacent Sector Logic boards, are routed as edge coupled microstrips across the three Jo connectors of the slots occupied by the ROD and the two Sector Logic boards. They have controlled impedance and matched lengths to deliver timing signals with the same propagation delay.
The TTL parallel buses run as single-ended lines on the J2 connectors and are terminated on board with a Thevenin scheme, according to the VME protocol. The

data word 32-bit @ 40 MHz	bit usage
3-bit	1st candidate threshold
5-bit	1st candidate RoI
2-bit	1st candidate overlap
3-bit	2nd candidate threshold
5-bit	2nd candidate RoI
2-bit	2nd candidate overlap
3-bit	BCID
1-bit	>2 candidates per sector
8-bit	not used

Figure 5.10: The current Barrel Interface Board data format.

TTL and the LVDS domains are fully decoupled in order to avoid interference between them. In the two regions, ground planes have been split to provide a separate and clean return current path to the sensitive LVDS signals. For a given full RODbus system (i.e. the five boards listed before) all the relevant LHC signals (clock, ECR, BCR, L1A) are received on the ROD board and then forwarded, on dedicated LVDS connections on Jo, to the adjacent Sector Logic boards. In Fig. 5.10 the I/O signal data format of the current MUCTPI Interface board is summarized. As previously seen, on the RODBus backplane 48-bit data is available, however only 32 bits are used for transmitting trigger data; the remaining were used for testing purposes. The 32-bit trigger data, output by the current MUCTPI Interface boards, is synchronously sent to the MUCTPI at 40 MHz on electrical cables, for an aggregate bandwidth of 1.28 Gbps. A maximum of two trigger candidates per trigger sector are sent from the SL. Each candidate is identified by 5 bits for the ROI, 2 bits for the ϕ overlap information (asserted in case the candidate originated from a region overlapping in ϕ with the adjacent sector) and 3 bits for the p_T threshold. Three additional bits of the BCID trigger word are sent for synchronisation purposes, and one additional bit is used to indicate more than two trigger candidates found in a sector.

For Phase-I Muon Barrel Upgrade, the current MUCTPI Interface will be replaced. The new Interface board designed in this PhD project has been built around an FPGA device, equipped with high-speed embedded transceivers, which optically send data to the MUCTPI. The first prototype uses a Xilinx Artix-7 XC7A200T-2FBG676C FPGA, with GTP transceivers [58] able to sustain a 6.6 Gbps data rate. The outputs of the GTX drivs a SFP+ optical transceiver (Avago AFBR-709SMZ) [95] with a LC connector for the optical fibre.
As the previous version, the new MuCTPI Interface is a 6U VMEx board, which will be installed in the ROD crates in the USA15 room in the same slots used for the current Interface board (Fig. 5.7).

A schematic block diagram of the new MUCTPI Interface board is illustrated in Fig. 5.11 (a). As a comparison also the block diagram of the present Interface board is shown in Fig. 5.11 (b). The board will be accessed by the VME64x protocol, mastered by the VME CPU already present in the ROD crates. The chosen

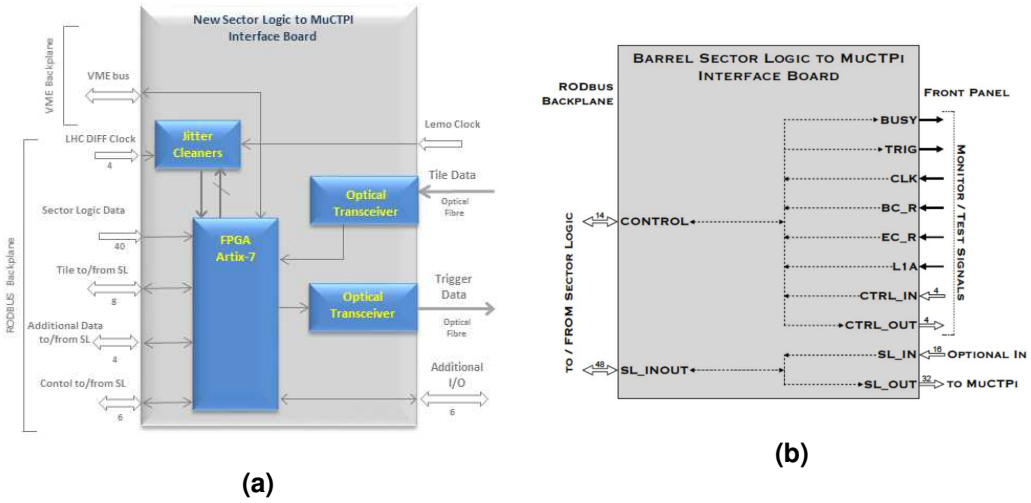


Figure 5.11: Block diagrams of the new MUCTPI Interface a and the present board b. The internal structure and the input and output signals of the two boards are depicted. (a).

transceiver for interfacing the board with the VME backplane is the Texas Instruments SN74VMEH22501 8-bit Universal Bus Transceivers [96], already deployed on the pre-existing ROD crate boards. The VME data will be received on the J1 and J2 connectors (as Jo is not used by the RODbus in the MuCTPI Interface board slot). The FPGA on the MuCTPI Interface board will take care of the hardware implementation of the VME interface on the board. The 40-bit trigger data at 40 MHz sent from the Sector Logic will be received by the FPGA, via the connections of the backplane RODbus. The FPGA will serialize at 6.4 Gbps and it will send them to the optical transceiver. Since the FPGA interface logic and the GTP will have to work synchronously with 40 MHz LHC clock, the Sector Logic board will differentially forward the clock, to the MuCTPI Interface board, using four dedicated lines on the RODBus. Two line will be used for transmitting the clock in LVDS standard and two extra adjacent lines will be grounded. The rest of the connections on the backplane has been foreseen for additional control (6) and data (4). 8 lines has been reserved for communication between the Tile and the Sector Logic. Additional I/O signals was also foreseen on the front panel: an extra optical transceiver to be used for Tile/Cal input data, 6 controls signals to be transmitted/received on a connector and a Lemo input for a backup LHC Clock. Taking advantage of the improvements related to the use of an FPGA device and to the allowed higher bandwidth (6.4 Gbps), a new, extended data format has been defined which will be able to handle up to four trigger candidates. The information which will be sent to the MUCTPI are summarized in Fig. 5.12. In order to guarantee a DC-Balance, the high-speed data will be encoded 8B/10B, since there will be enough bandwidth and the encoding overhead of 25% will not be an issue.
The new MuCTPI Interface will use the 40 MHz LHC clock, as reference clock

data word 40-bit @ 40 MHz	bit usage
3-bit	1st candidate threshold
5-bit	1st candidate RoI
2-bit	1st candidate overlap
10-bit	2nd candidate word
10-bit	3rd candidate word
10-bit	4th candidate word

Figure 5.12: The new Barrel Interface Board data format .

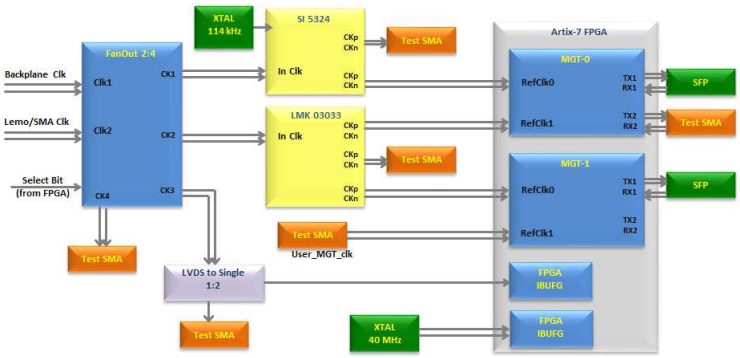


Figure 5.13: Board Clock tree scheme .

for the high-speed serial transceiver, however, it should have a maximum jitter of 68 ps, in order to correctly drive the high-speed transceiver with a 10^{-12} Bit Error Ratio. In order to respect this strict constraints (RODbus wasn't designed to transmit also a clock signal, hence no dedicated lines were foreseen to this purpose), before entering the FPGA, the LHC clock will be filtered by a jitter cleaner. Since the board is designed for test, two jitter cleaners will be mounted on the board: a Texas Instruments LMK03033 device [97], which needs a 3.3 V power supply and it is configurable by using uWire protocol via FPGA, and a Silicon Labs SI5324 [98] which could be also used as an extra board clock generator. Both the devices are jitter-attenuating, precision clock multiplier with ultra-low jitter clock outputs (~300 fs rms). The board clocks routing scheme is shown in Fig. 5.13. Both the clock coming from the Sector Logic board (Backplane Clk) and from the backup LHC Clock signal coming from the front panel of the MUCTPI Interface board (Lemo/SMA Clk) enter a fan-out chip, in order to be able to feed the two jitter cleaners and the FPGA logic. Different test points are inserted in the clock path for evaluating jitter cleaner performances and checking the impact of the single device on the clock signals (in terms of jitter and skew). The board will also have a 40 MHz local clock, used for the configuration of the board at the power-up, for standalone tests and useful for the control of the board when no external clock is available (for example if the jitter cleaner is not yet ready or programmed). An extra clock input (User_MGT_clk) is provided for skipping the fan out and the jitter cleaner and easily test the FPGA functionalities. A 3-D image

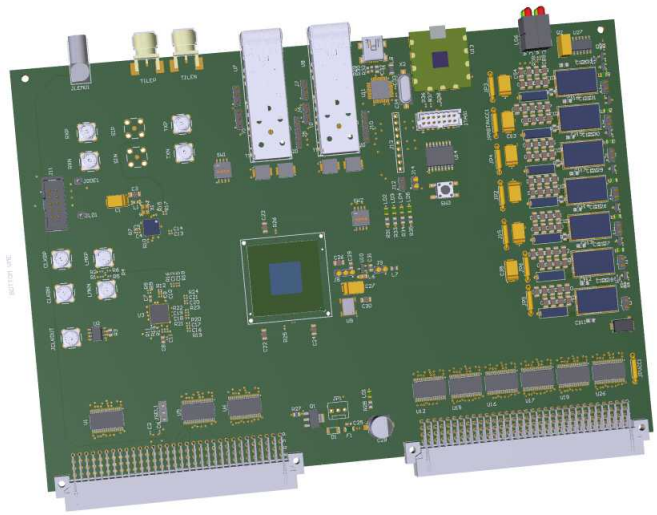


Figure 5.14: The 3-D image of the new MUCTPI Interface board prototype preliminary layout .

of the prototype preliminary layout of the new MUCTPI Interface board for the muon barrel detector can be seen Fig. 5.14.

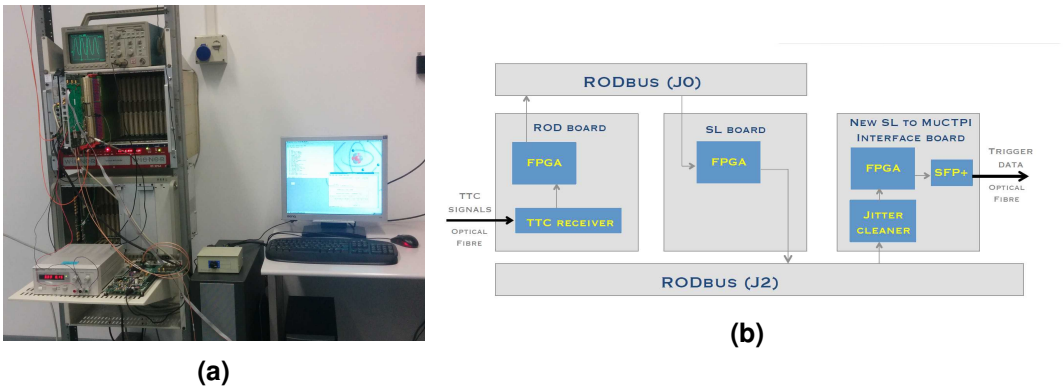


Figure 5.15: Test bench for preliminary test (a) and its block diagram (b).

A preliminary test on the feasibility of the project was made for checking the quality of the LHC signal clock forwarded by the Sector Logic board. The set-up and its block scheme are shown in Fig. 5.15a and Fig. 5.15b. The Sector Logic board in the crate sends the clock in LVDS standard to the adjacent MuCTPI Interface board slot via the J2 connector of the RODBus. On this slot, it is mounted an emulator of the MUCTPI Interface board, which is only able to receive the clock signal and which is equipped with the LMK03033C jitter cleaner. The signal is filtered and multiplied by the LMK3033 jitter cleaner in order to obtain a 160 MHz reference clock for the GTP. A multiplication of LHC clock is necessary, since according to the reference guide, the minimum value for the GTP reference clock is 60 MHz. The clock signal is correctly received by the Jitter cleaner and differentially sent to an FPGA via coaxial cables. On the FPGA, the fixed latency

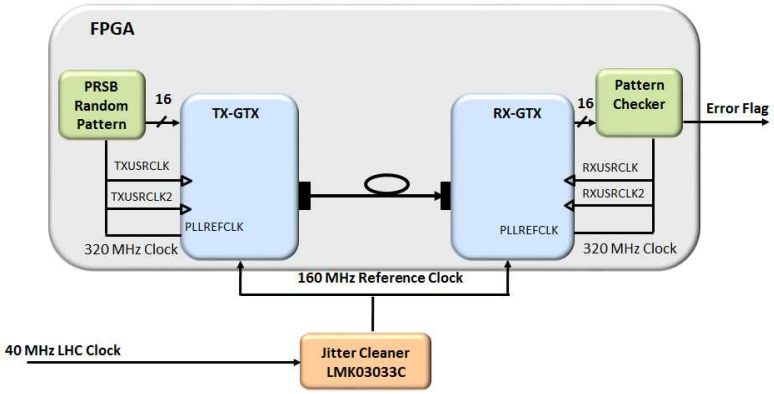


Figure 5.16: Fixed latency serial link for the MUCTPI Interface.

link in Fig.5.16 was implemented. A test using Xilinx IBERT core was performed in order to check the correctness of the communication. Other tests will be done in order to perform jitter measurements on the received signal and to evaluate the impact of the physical layer on the signal integrity of the clock itself.

5.2 SERIAL INTERFACE FOR THE PAD TRIGGER BOARD

5.2.1 *New Small Wheel Project*

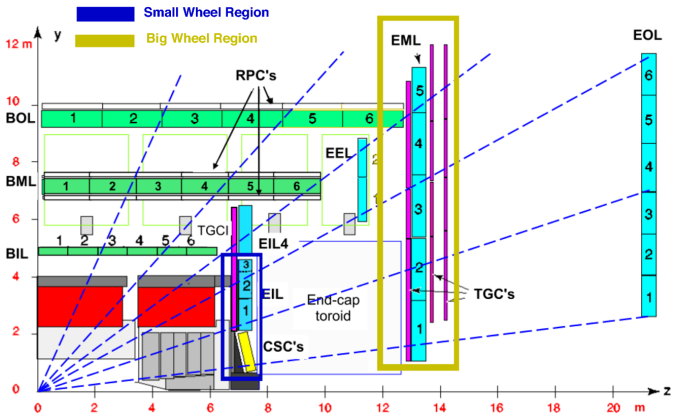


Figure 5.17: A view of the ATLAS detector in the $z-y$ plane. The MDT detectors are indicated by ciano boxes in the end-cap region (the three stations *End-cap Inner Large*, EIL, *End-cap Middle Large*, EML, and *End-cap Outer Large*, EOL), and by green boxes in the barrel. The RPC and TGC trigger detectors are indicated by the white and the magenta boxes, respectively. The yellow box in the outlined *Small Wheel* area are the CSCs. The *Big Wheel* detector regions is also highlighted.

The luminosity levels foreseen for the LHC after the Phase-I upgrade will require tighter constraints to the ATLAS first level muon trigger system, especially in the end-cap region, which provides ~ 63% of the ATLAS muon system cover-

age. Therefore high trigger performance in this region is of absolute importance in order to achieve the physics goals which LHC will be upgraded for.

The actual Muon Spectrometer, as described in the previous chapter, is not able to respect this strict requests, therefore some of its subcomponets are planned to be replaced. Indeed, with the present configuration, the trigger decision is based only on decisions of the TGC detectors of the second out of three end-cap stations (the *Big Wheel* region identified by the yellow box in the Fig. 5.17).
Since the end-cap muon L1 trigger does not use a direct bending angle determination of the muon trajectory through the end-cap toroid, it is not able to distinguish between particles originated from the interaction point, IP, and from other uncorrelated sources, as depicted in Fig. 5.18.

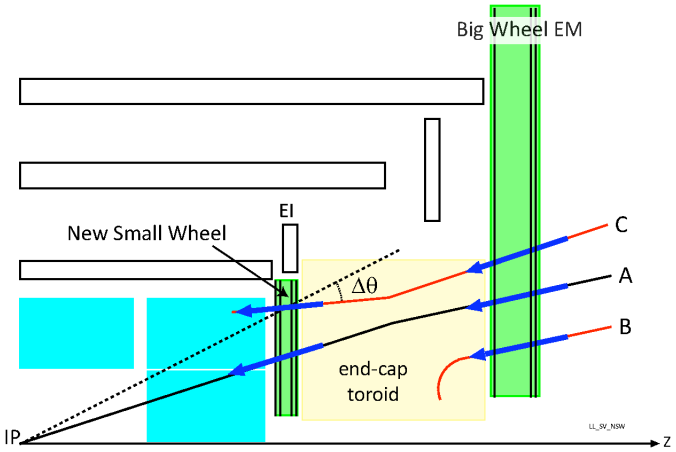


Figure 5.18: A Schematic of the Muon End-cap trigger. The existing trigger, which use only the information coming from the *Big Wheel*, accepts all three tracks: A,B, and C. However, the track A, which is pointing to the interaction point, is the only desired track.

As a consequence, most of the end-cap triggers are unrelated to muons of physics interest coming from the pp interaction point and hence, they are not reconstructible offline. Fig. 5.19 shows:

- the distribution (in η) of all trigger candidates selected as muon with at least $10\,GeVc^{-1}$ p_T (L1_MU11);
- the distribution of those candidates which an offline reconstructed muon track correspond to;
- the distribution of the muons reconstructed with $p_T > 10\,GeVc^{-1}$.

The high rate due to *fake triggers* is one of major weakness of the present muon end-cap design and it would become intolerable after Phase I upgrade. Indeed, with the higher luminosity expected ($L\sim 5*10^{34}cm{-}2s{-}1$) the present muon end-cap L1 trigger rate would exceed the allowed trigger bandwidth. Some reduction of the trigger rate may be gained by raising the p_T threshold of the trigger but with an undesired loss of physics acceptance for many physics channels. A better

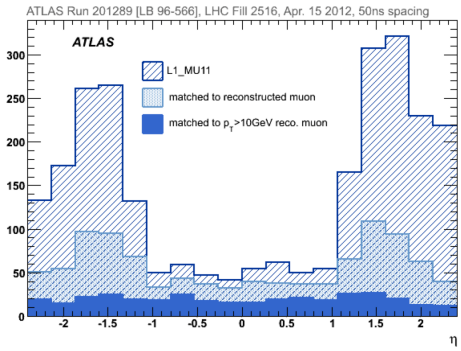


Figure 5.19: η distribution of L1 trigger signal coming from muons with $p_T > 10\,GeVc^{-1}$(L1_MU11) with the distribution of the subset signal coming from muon candidates which match (within $\Delta R < 0.2$) to offline well reconstructed muons (combined inner detector and muon spectrometer track with $p_T > 3\,GeVc^{-1}$), and offline reconstructed muons with $p_T > 10\,GeVc^{-1}$.

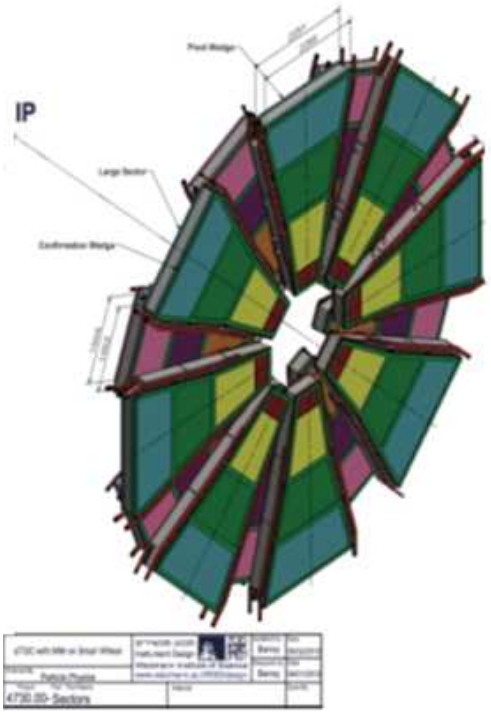


Figure 5.20: The new Small Wheel Layout.

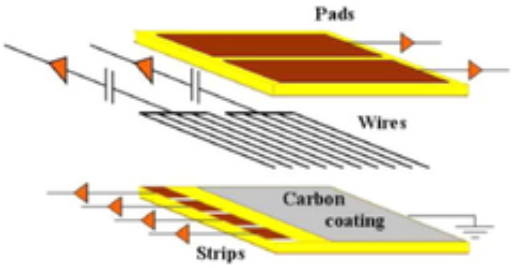


Figure 5.21: The sTGC internal structure.

control of the L1 rate can be obtained by improving the trigger p_T resolution and reducing the background. In addition to this problem, the track reconstruction efficiency of the MDT detectors will decrease to an unacceptable level. With the loss of the precise tracking information of that innermost station, the reconstruction of the muon momentum would greatly deteriorate.

The new Small Wheel (NSW) upgrade project is intended to addresses these limitations through the introduction of new detectors in to the *Small Wheel*: Micromegas (MM) for precision tracking and small Thin Gap Chambers (sTGC) for triggering. The general layout of the NSW is indicated in Fig. 5.20. The wheel is segmented in 16 ϕ-sectors, 8 large and 8 small which only differ for their dimensions. Every sector is equipped with four sTGC layers and four MM layers mounted in doublets, in order to create a fully redundant system, both for trigger and tracking. The order of the detectors, sTGC-MM-MM-sTGC, has the purpose of maximizing the distance between the sTGC layers.

5.2.2 *sTGC*

The basic Small strip Thin Gap Chamber (sTGC) structure is shown in Fig. 5.21. It is based on the technology of Multi-Wire Proportional Chambers, which was developed in 1983 [99]. The sTGCs are gaseous detectors, with operating gas 55% CO2 and 45% N-pentane, and with a grid of 50 µm gold-plated tungsten separated by 1.8 mm pitch. The distance between each of the two cathode planes and the wire plane is 1.4 mm. Around 2.9 kV of high voltage is applied to the chambers. The cathode planes are made of a graphite-epoxy mixture sprayed on a 100 µm thick G-10 plane. Behind the graphite, on one side, there are strips which run perpendicular to the wires and have a 3.2 mm pitch (much smaller than the strip pitch of the actual TGC, hence the name Small TGC for this technology). On the other side of the graphite plane there are pads which cover a large rectangular surfaces providing the list of strip bands to be read out.

5.2.3 *Micromegas*

The MicroMegas (which stand for *micro mesh gaseous structure*, MM) technology was developed in the middle of the 1990's [100]. MM detectors consist of a planar

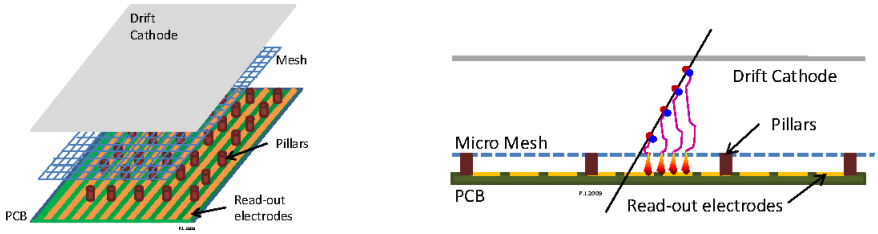


Figure 5.22: Sketch of the layout and operating principle of a MM detector.

(drift) electrode, a gas gap of a few millimetres thickness acting as conversion and drift region, and a thin metallic mesh at typically 100-150 µm distance from the readout electrode, creating the amplification region. A scheme of the MM operating principle is shown in Fig. 5.22. Charged particles traversing the drift space ionize the gas; the electrons liberated by the ionization process drift towards the mesh. With an electric field in the amplification region 50-100 times stronger than the drift field, the mesh is transparent to more than 95% of the electrons. The electron avalanche takes place in the thin amplification region, immediately above the readout electrode. The drift of the electrons in the conversion gap is a relatively slow process; depending on the drift gas, the drift distance, and the drift field it typically takes several tens of nanoseconds. On the other hand the amplification process happens in a fraction of a nanosecond, resulting in a fast pulse of electrons on the readout strip. The ions that are produced in the avalanche process move, in the opposite direction of the electrons, back to the amplification mesh. Most of the ions are produced in the last avalanche step and therefore close to the readout strip. A known problem of the MM original design was their vulnerability to sparking. If the gas contains small impurities, or the avalanche reaches the Raether limit [101] of few 10^7 electrons, sparks occur with the possibility of damaging the detector and readout electronics and/or lead to large dead times as a result of HV breakdown. To avoid this problem, for the present MM detectors, a spark protection system has been developed which consists in adding a layer of resistive strips on top of a thin insulator directly above the readout electrode. The MM become spark-insensitive since the readout electrode is no longer directly exposed to the charge created in the amplification region, however, at the same time, the signals are capacitively coupled to it. The principle of the resistive spark protection is schematically shown in Fig. 5.23.

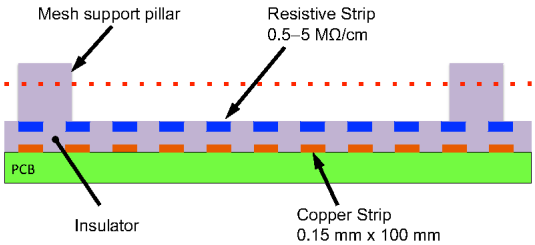


Figure 5.23: Spark protection principle.

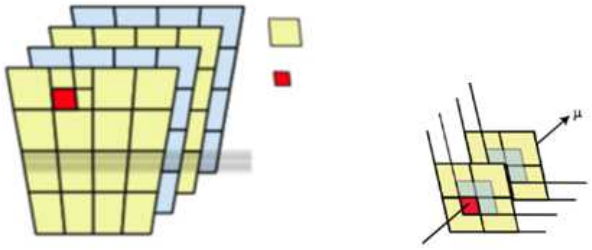
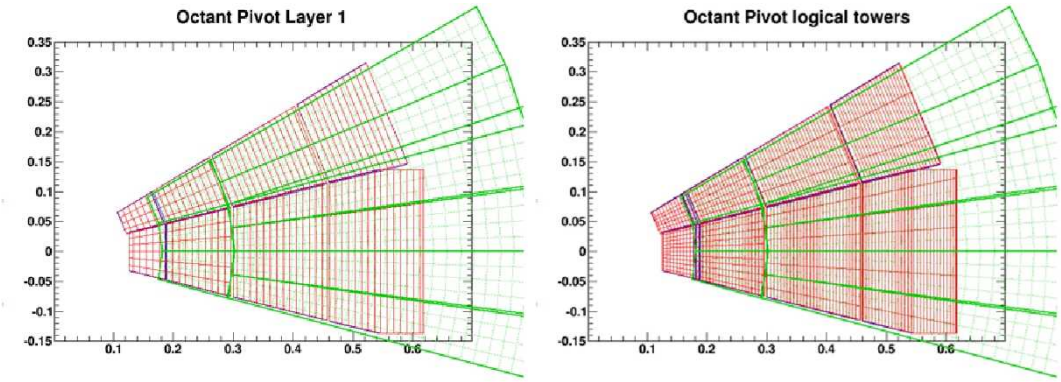


Figure 5.24: The sTGC qaudruplet and schematic view of the logical tower.

5.2.4 *sTGC trigger system*

Every 25 ns, the sTGC trigger electronics finds local tracks that point,with <1 mrad precision, to the *Big Wheel* to confirm its coincidences.
The NSW sTGC trigger system makes use of coincidences of detector pads to identify regions where a muon candidate was detected. To reduce the amount of data sent to the off-detector trigger processors, only sTGC strip information coming from the regions selected by the pad logic are transmitted off-detector. To reduce the number of pad channels, the pad layers are staggered by half a pad length in η and ϕ. This allow the trigger to make use of the logical towers, which represent one quarter of the pad area (Fig. 5.24). Fig. 5.25a and Fig. 5.25b show the physical and logical Pad distributions inside one octant (made of one small sector and one large sector). The width of the pads in ϕ decreases with lower radius to handle the increased background rate.
The Pad trigger logic looks for hit coincidences in a tower of logical Pads within one Bunch Crossing.



(a) **(b)**

Figure 5.25: The physical aand logical Pad distributions ainside one octant.

The NSW pad-trigger logic is implemented in two steps:

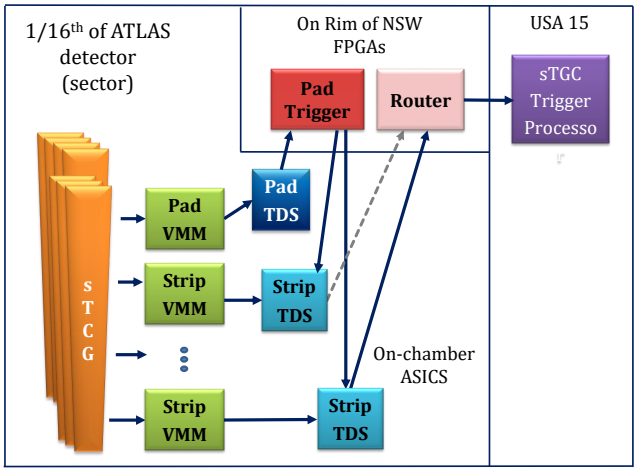


Figure 5.26: The sTGC trigger scheme.

- Single pad trigger: A three out of four majority[1] logic is required in each sTGC quadruplet, independently.
- Pad trigger: trigger decision is finally based on geometrical matching between the pivot multiplet trigger and the trigger from the confirm multiplet.

The trigger scheme for one sector of the NSW is shown in Fig. 5.26. The figure shows the signal flow from chamber to the track finding and extrapolation logic (sTGC Trigger Processor) in USA15. On-chamber, signals from strips are digitized by the strip-VMM ASIC. The 6 bits-wide word representing the peak amplitude of the signal is immediately sent serially over one line per channel, to the strip Trigger Data Serializer, strip-TDS. In this ASIC board all active strips are saved and tagged with the BCID, awaiting possible selection by the pad tower trigger for transmission off-detector. AT the same time, the Time-over-Threshold pulse of each pad is sent by the pad-VMM to the pad-TDS, where it is latched by the next BC. Up to 96 pad hits coming from all channels are serialized and sent each BC from one pad-TDS front-end chip to the pad trigger board on the rim of the NSW. The pad-TDS data are arranged into 120 bits and transmitted, every 25 ns, using a 4.8 Gbps serializer. The format is listed in Table 5.1. All data except the header bits are scrambled to keep DC-balanced. The polynomial used for the scrambler algorithm is: $1 + x^{39} + x^{58}$, which is the IEEE standard for 10 G Ethernet [102].

On the pad trigger board, the pad coincidence logic, running in parallel with the strip data collection, selects the band of strips in each layer that passes through the tower generating the pad coincidence. It is possible to send up to three trigger candidates per one sector per beam crossing, one per Inner/Middle/Outer region.

1 In the transition regions in η between the detectors, an hit in all four layers of a single quadruplet is required.

Header	Pad-TDS status	BCID	CRC
4-bits "1010"	96-bit Pad hits (1/0)	12-bits BCID	8-bit CRC

Table 5.1: The pad-TDS data format.

Geometrical coordinates		
road-ID	phi-ID	BCID
8-bit	5-bit	12-bits

Table 5.2: Data format from pad trigger board to strip-TDS ASIC.

Candidate geometrical coordinates and BCID are serialised and sent to the front-end strip-TDS chips, according to the format shown in Table 5.2: In order to keep a low latency, 2 differential lines per each output are used, one for the geometrical coordinates and one for the BCID. Hence, strip-TDS transmits its *strip charges* to the router on the periphery of the wheel. The data transmitted include the Bunch Crossing ID, band-ID,ϕ-ID and the *strip charges*. The Router then sends the data from the active strip-TDS ASICs to the centroid finding and track extrapolation logic in USA15 via optical fibre. In USA15 the active strips selected by the pad trigger are used to compute centroids and then track segments that point to the Big Wheel. These candidates are then sent via fast serial links to the Sector Logic to be combined with Big Wheel candidates. The whole process repeats every BC with a fixed latency. The total latency of the trigger system must comply with the existing Sector Logic timing which allows a total of 41 BC, 1025 ns.

5.2.5 *Pad Trigger Logic board*

As previously described, the pad trigger board implements the first sTGC trigger logic. For each sector, it looks for hit coincidences in a tower of logical pads, in order to identify only the relevant strips where a muon candidate was detected. There are 16 sectors per wheel, which means 16 pad trigger logic boards per each NSW, for a total of 32 boards. The board is designed around a Xilinx Kintex-7 FPGAs, whose programmable Multi-Gigabit Transceivers (GTXs) are used for implementing the receiver logic. The transmitter interface is implemented on standard logic in order to achieve a lower latency. The pad trigger board receives trigger data from the 24 pad-TDS ASICs on miniSAS cable [66] 4.8 Gbps. Then, data is serialized and descrambled and sent to the trigger logic. Trigger algorithm selects up to three candidates which are serialized and sent to the stirp-TDS ASIC via miniSAS cables. All the building blocks of the board, schematically shown in Fig. 5.27 will be described in this section.

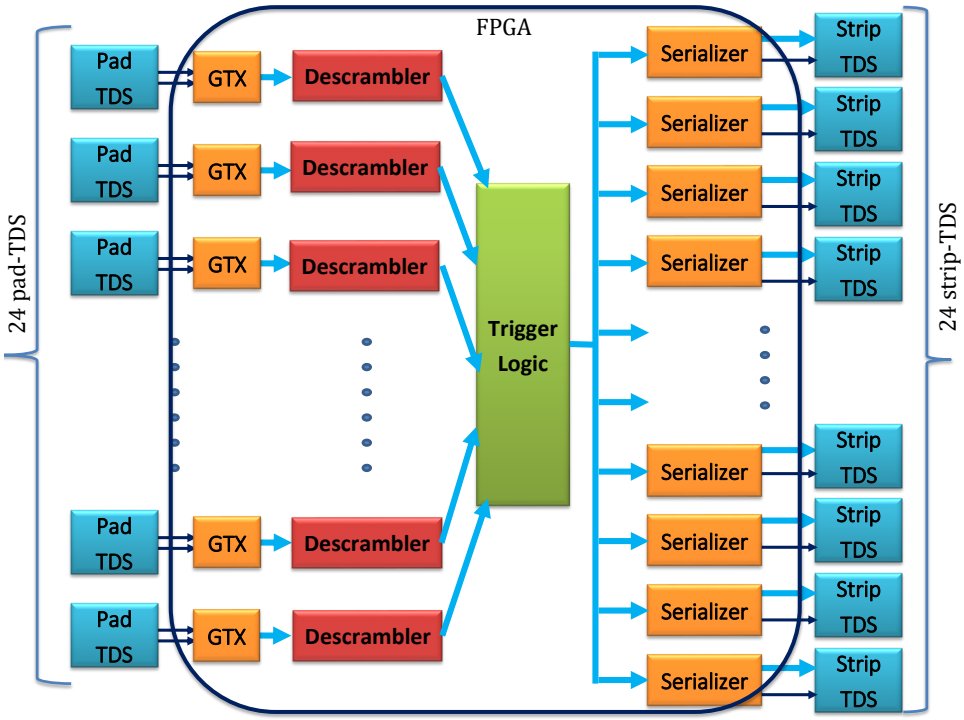


Figure 5.27: FPGA-pad trigger internal structure.

Fixed-latency Receivers

The receiver block is made by 24 embedded GTXs programmed in order to guarantee a fixed-latency[2] link which standard GTX configurations do not have. In the previous chapters, some of the essential features of the GTX were already presented. This section focuses on how latency variations can be introduced in the transmission, by the serial and parallel section of the MGT, and how the receiver was implemented for the pad board in order to overcome these variations. In the serial section, latency variations can be the result of phase variations in the recovered clock produced by CDR circuit. Indeed, starting form the serial stream, the CDR algorithm derives a high-speed serial clock, HSCLK, which feeds the serial part of the SIPO, and a slow parallel, recovered clock which is used on the parallel side of the SIPO for clocking parallel data out (5.28). In order to clarified the concept, an example will be used. If 10-bit words are serialized, each edge of HSCLK can be labelled by a bit position in a 10-bit symbol. At each power-up, the GTX can randomly produce a recovered clock aligned with one of the ten edges of HSCLK. Then, if at the transmitter side of the link, parallel data is transmitted synchronous with TXUSCLK, while at the receiver side, data is clocked by RECCLK, the phase difference between the two clock domain determines a latency variation in terms of integer numbers of UIs (T_{HSCLK} =UI):

$$\Delta L_{serial} = (n-1)UI,$$

2 In a synchronous system, the latency through a path, from a node A to a node B, is the time interval needed for data to go from A and arrive on B following that path

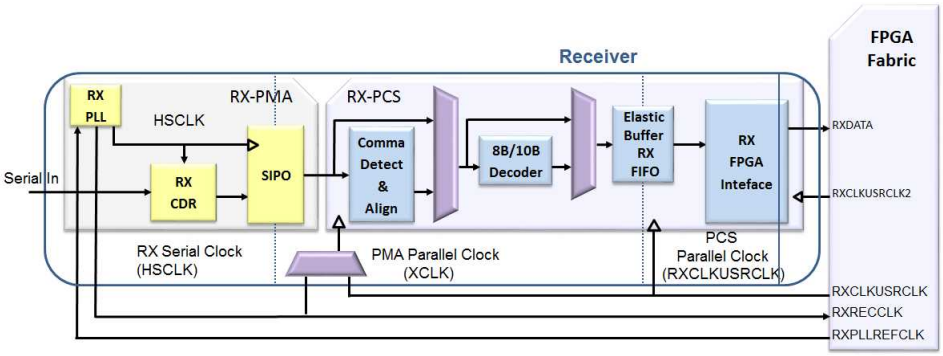


Figure 5.28: Receiver section of the GTX transceiver.

where n is an integer whose values varies form 0 to the width of the parallel word on the GTX.
In Fig. 5.29 the relation between the serial stream, synchronous with the HSCLK,

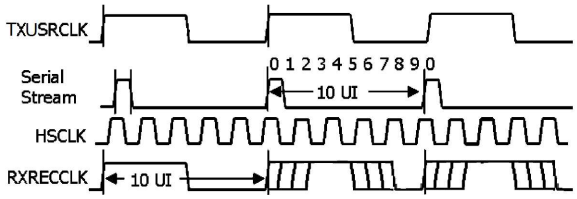


Figure 5.29: Latency variation introduced by the CDR recovered clock.

the transmitter clock, TXUSRCLK, and the recovered clock, RECCLK, is shown, for the case of a parallel data of 10 bits explained. Besides, in particular applications, the signal for clocking data at the receiver interface, RXUSRCLK2, is request to be a divided version of the recovered clock. When this is the case, even if recovered clock is generated always with the same phase offset with respect to the serial stream, its division is uncorrelated with respect to the stream. The uncertainty of the RXUSRCLK2 phase with respect to serial stream makes the latency variable of an integer number of recovered clock period. Finally, in the serial section, the both these effects produce a variation of the latency between two subsequent resets of:

$$\Delta L_{serial} = (n-1)UI + mT_{rec}\,,$$

where T_{rec} is the parallel clock period, UI is the time duration of one serial symbol, and n and m are integer numbers in whose range depends on the configuration of the GTX.

As far as it concerns the parallel section, latency uncertainties can be introduced by elastic buffers, which are usually implemented by means of FIFOs. Elastic buffer are used to null the phase difference between the recovered parallel clock (XCLK) domain and the FPGA parallel clock (RXUSRCLK) domain. However, assuming an equal write and read rate from the buffer, after each reset the latency trough the buffer is determined by the difference of the write and read pointers.

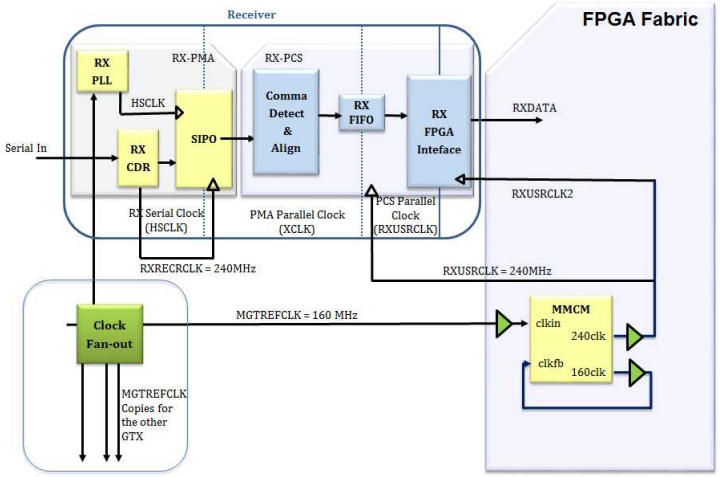


Figure 5.30: Fixed latency receiver architecture.

This difference may vary depending on the behaviour of the logic accessing the buffer and causes variations in terms of integer numbers of periods of the clocks for buffer read/write. So, a dedicated mechanism is needed to ensure that always the same number of words has been written in the buffer before they start being read (the receiver elastic buffer of the GTX implements this feature).

Keeping in mind this discussion, the architecture implemented is shown in Fig. 5.30. Since data comes from the pad-TDS ASICs at 4.8 Gbps a receiver with the same line rate has been implemented. Pad-TDS data (120 bits) are sent in four packets of 30-bits. It would be easier to deserialise data in the same format: parallel 30-bit wide word synchronous with a 160 MHz clock, however, GTX supports words with a width of 8, 10, 16, 20, 32, 40, 64 and 80-bits. Hence, in order to keep the length of the parallel recovered word a sub-multiple of the original data and to reduce the latency[3], it has been chosen to operate with 20-bit symbols. As a consequence, according to the user guide ([59]), RXUSRCLK and RXUSRCLK2 have to run at the same frequency of 240 MHz. The advanced DLL and frequency generator on the FPGA (MMCM) has been used to produce this signals from the 160 MHz reference clock. The clock buffer delay on the MMCM "x1" output is compensated since it is inserted in the feedback loop. The other clock buffer can be considered compensated as long as buffer delays can be considered identical. Under this approximation, the skew between the reference (on-board) clock and the FPGA logic clock is due to the input buffer delay on the MMCM clkin pin and the corresponding routing from the FPGA input pad ($\sim 1\,ns$). In the final prototype the situation will be slightly different. The 160 MHz reference clock will be directly provided by ATLAS. However, since for feeding 24 GTX up to 3 reference clocks will be necessary, a clock fan out will be used. The clock fan out will introduce a little additional jitter on the ATLAS clock as well as little skew. Then, the additional skew, coming from the additional path on the board, will have to be taken into account .

3 In the GTX serdes, latency depends on RX word width

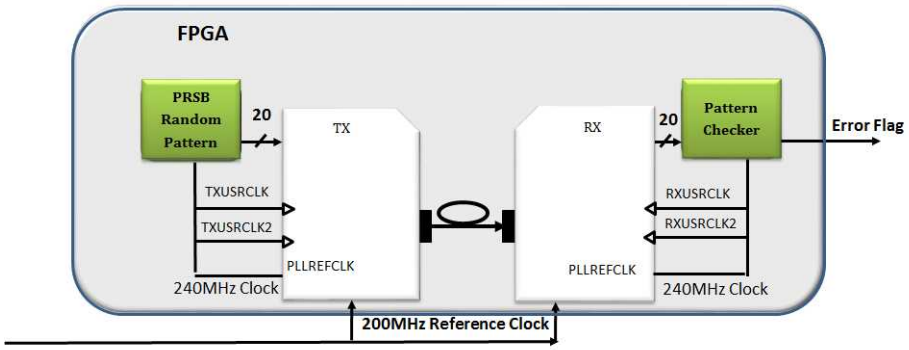


Figure 5.31: Fixed latency link.

	#of RXUSRCLK periods	Block Latency (ns)
Serial Section (PMA)	4.7	19.7
Alignment block	2	4.2
Elastic Buffer (FIFO)	3	12.6
FPGA Interface	2	8.4
Total Latency	11.7	46.1

Table 5.3: Latency contribution of the different GTX blocks.

In order to test the receiver, a transmitter with the same clock architecture was deployed, in order to obtain the fixed latency link shown in Fig. 5.31. Data is transmitted synchronously with transmitter board clock and received synchronously with the receiver board clock. Since the reference clocks of the transmitter and the receiver are generated by the same source, the recovered clock is always generated with a fixed phase relationship with respect to the serial stream ([103]). This means that the recovered clock has always the same phase difference with respect to RXUSRCLK, even after a power-cycle. In order to compensate any phase difference between the transmission and reception clocks, so to achieve fixed latency, the Elastic Buffer (FIFO) on the receiver is enabled. Indeed, in the FIFO data is written synchronously with XCLK and read with RXUSRCLK, which are phase misaligned but synchronous. The GTX guarantees ([59]) that, after a reset, the FIFO is always filled with the same number of words before starting to output them. This ensures a deterministic latency through the FIFO. In order to have also the lowest possible latency, the minimum value of used words is set to its minimum (3 words).
The latency on the receiver, estimated by means of the user guide, is given in Table 5.3. For each component, latency is reported in terms of RXUSRCLK periods or UIs and its absolute value in nanoseconds.

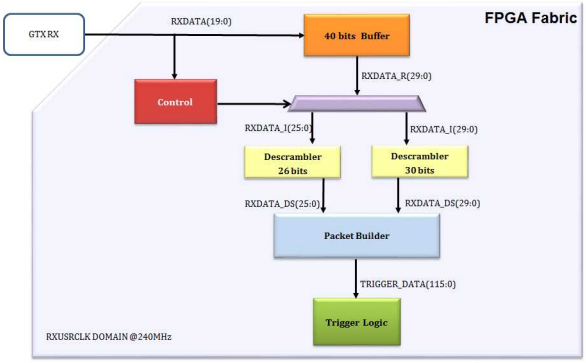


Figure 5.32: Descrambler logic block diagram.

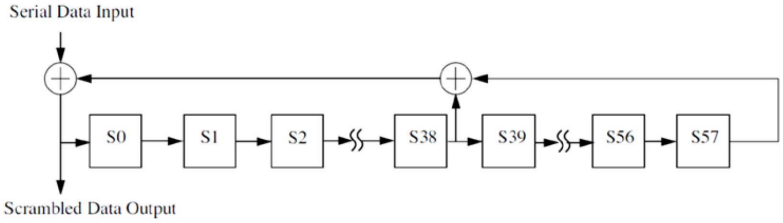


Figure 5.33: Descrambler logic block diagram.

Descrambler

Since the GTX Receiver does not support 30-bit length for the SIPO parallel output, the width is set to 20 and thus the corresponding RXUSRCLK is 240 MHz for 4.8 Gbps. However, the pad-TDS 120-bits frame is divided in 4 packets of 30-bits, coded according the same scrambler scheme used in IEEE Standard 802.3-2012 for 10 Gb/s Physical layer implementations. In order to decode data the original packets of 30 bits are rearranged, paying specific attention to the first packet, for which the four bit of header ("1010") are not scrambled.

In Fig. 5.32, the schema of this function is shown. The control block, designed around a 4-module counter, selects which part of the recovered 20-bits data has to be sent to the following block, the descrambler, which implements the polynomial function $1 + x^{39} + x^{58}$, represented in Fig. 5.33. The latency introduced by this block is estimated to be of 25.2 ns (1 RXUSRCLK for the descrambler + 5 user clock for the arrival of the entire data packet of 120 bit).

Transmitter

Every 25 ns, once the trigger logic has selected the trigger candidate in the sector region[4], the candidate geometrical coordinates together with its corresponding BCID need to be sent to the front-end strip-TDS chips of the region. In order to guarantee the synchronization between the pad trigger board and the front-end strip-TDS chips, additional information is sent. A total of 7 differential lines per

4 each layer is divided into the region: Inner Middle and Outer

Line	Data
1	synchronization signal with the frequency of 320 MHz
2	frame signal indicating the beginning of a packet
3	2-bit header + 12-bit BCID + 2-bit idle
4	TDS0: 2-bit header + 8-bit road ID + 5-bit phi ID + 1-bit idle
5	TDS1: 2-bit header + 8-bit road ID + 5-bit phi ID + 1-bit idle
6	TDS2: 2-bit header + 8-bit road ID + 5-bit phi ID + 1-bit idle
7	TDS3: 2-bit header + 8-bit road ID + 5-bit phi ID + 1-bit idle

Table 5.4: Data format from pad trigger board front-end strip-TDS chips.

front-end strip-TDS chips are used. Data format is show in Table 5.4. With the exception of Line 1 and 2, data are serially send at 640 Mbps; such a line rate is obtained by transmitting data on both the edge of a 320 MHz clock (DDR mode) In order to keep the number of differential outputs (and of the connectors) low, for Line 1, 2 and 3 a fan-out is foreseen on the front-end board, since their are in common to all the chips. The block diagram of the transmitter block is show in Fig. 5.34. When enabled, on every rising edge of the 320 MHz clock, the DDR registers serialize data present on their input. The purpose of the bank of multiplexers is presenting the right words (2-bits wide) to the input of the register, in such a way that the trigger information is properly serialized. In order to implement this function, a 7-module counter is used, which drives the selection input of the multiplexer.

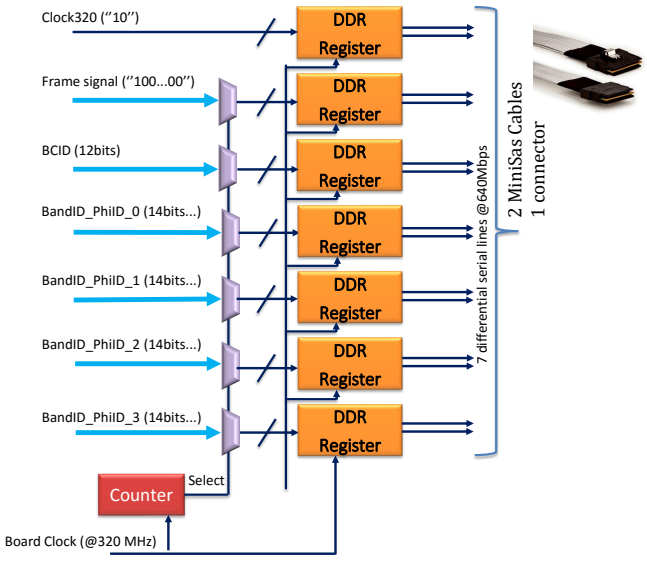


Figure 5.34: Output logic to the strip-TDS .

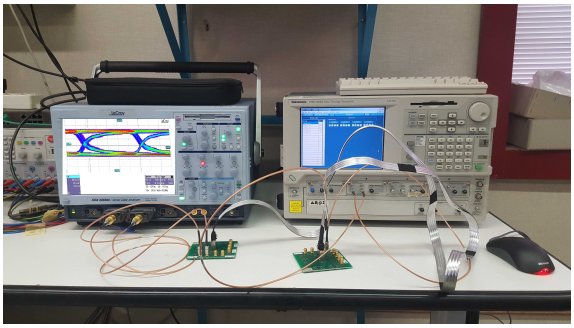


Figure 5.35: Set up for testing the transmission quality of the miniSAS cable.

Tests

The development platform used in this project is the Xilinx evaluation board KC705. It is equipped with an FPGA XC7K325T-2FFG900C device, provided with GTX transceivers supporting a line rate up to 10.3125 Gbps. In order to check the correctness of the logic and evaluate the latency value, all the logic implemented on FPGA was only simulated, since a prototype of the pad-TDS transmitter is presently under production. However, tests on the quality of the transmission using the miniSAS cables were done, in order to qualify its use in the project.
The set-up for the first test is shown in Fig.5.35. For different data rate, a PRBS pattern was generated by the Tektronix Data Timing Generator (DTG) 5334 [104] and sent on the cable. The serial stream at the end of the cable was analysed by using an oscilloscope [104], in order to build an Eye Diagram. Since either the pattern generator and the oscilloscope inputs only accept SMA coaxial cables a custom interface board, SMA-to-miniSAS/miniSAS-to-SMA, was used. Eye Diagrams were measured for two different lengths of the cable: 2 m and 4 m at 800 Mbps and 1250 Mbps.
The results of the test, obtained by transmitting ~ 100Kb at 800 Mbps on the 4 m miniSAS cable, are presented in Fig. 5.36. They show that, as soon as the length of the cable becomes highe,r the Eye Diagram becomes closer. Increasing the line rate led to the same effect. However, since the set-up didn't allow to completely reject the miniSAS solution, other measures were done. For the second set of tests, IBERT was intensively used. IBERT is a Xilinx Logic Core which can be used to evaluate and monitor the functionality of GTX. Since it can have access to the Dynamic reconfiguration port (DRP) of the GTX, IBERT allows to monitor and change any attribute in any of the GTX transceivers. Besides, IBERT can drive a pattern generator to send data out through the transmitter and a pattern checker, which can take the data coming in through the receiver and check it with an internally generated pattern. Using this Xilinx feature, the quality of the cable was tested using two KC705 evaluation boards on which a simple link at 4.8 Gbps was built . Data were sent form the Transmitter section of GTX on one board, to the GTX receiver, on the companion board. Since the only allowed pins for accessing the GTXs were routed to SMA connectors, in order to test the 2 m miniSAS cable, the same interface board used for the Eye Diagram tests was employed to connect

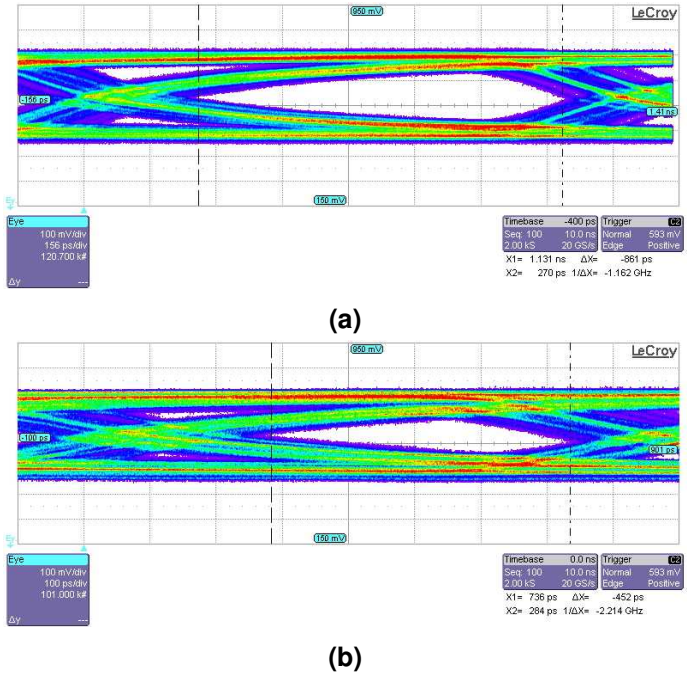


Figure 5.36: The Eye Diagram obtained sending on a 4 m miniSAS cable a pseudo random pattern at 800 Mbps(a) and 1250 Mbps (b) .

the two ends of the link. In this condition, no communication on the link was observed. Then, accessing the DRP port of the transmitter and of the receiver, the quality of communication was improved, by using the GTX pre-Emphasis feature on the transmitter side and adjusting the termination voltage on the receiver side. A wide range of settings were explored however, in the best case only a BER of $6.5 10^{-3}$ was reached.

Finally, in order to implement the link, using only the minSAS cable, a custom FMC mezzanine board designed by INFN Roma-I, was used. A picture of the mezzanine board is shown in Fig. 5.38. The board is equipped with one the

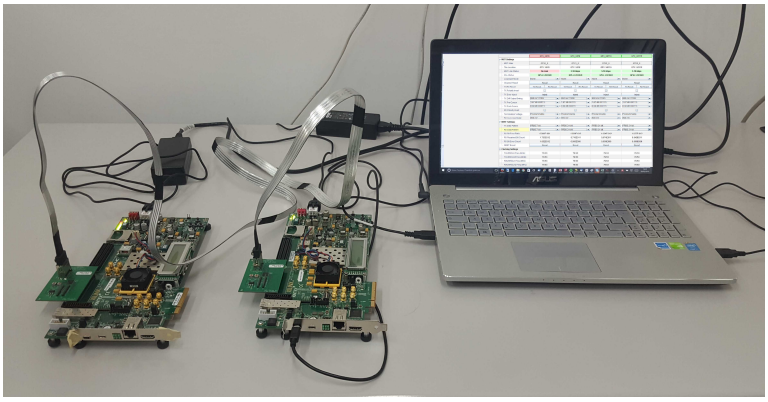


Figure 5.37: Set up for testing the transmission quality of the miniSAS cable.

Figure 5.38: The FMC mezzanine board.

connector for the miniSAS cable and one TI DS100BR410 repeater [105], a specific chip able to compensate for channel loss. This chip, whose programmable settings can be applied via pin mode, is mounted just before the inputs of the receiver. On the mezzanine board, the setting of the repeater can be driven either by switches or by the FPGA. Although, if the simpler switches were initially used, in view of the final installation on the pad trigger board, a circuit was developed around the Microblaze embedded microprocessor [61] in order to program the repeater via RS232 protocol.
The introduction of the repeater on the transmission path allowed to qualify the link with a BER of 10^{-12}, once the best configuration was chosen.

Board Prototype

A block diagram of the PAD trigger board prototype is shown in Fig. 5.39. During this PhD work I also took care of the design of the FPGA power system tree and I/O interfaces for control and monitoring of the board. Indeed, in the previous section a detailed description of the interface developed on this work has been done.
However, additional logic has been and will be developed in order to implement the prototype. Indeed, the Pad trigger logic board functions are performed inside the FPGA, however the pad board is foreseen to be equipped with one GBT-SCA chip, providing local I2C and JTAG signals. Fig. 5.40 shows how this chip is connected to two GBTx chips [106].
A control PC in USA15 communicates with the Pad board via GBT for uploading the FPGA firmware, configuring the FPGA and the other chips (DC-DC converters, I/O registers, etc.), and for monitoring the board (voltage, current, temperature). For the first prototype, however, an alternative JTAG line is foreseen in order to communicate with the board also without the GBT-SCA chip. At this scope both firmware (based on Microblaze embedded microprocessor) and software in order to access the FPGA's internal registers via RS232 protocol, were implemented.
Also GBTx E-links will be mounted on the board for providing TTC signals (CLK, L1A, BCR, ECR). Two GBTx chips per board are foreseen for redundancy, and they

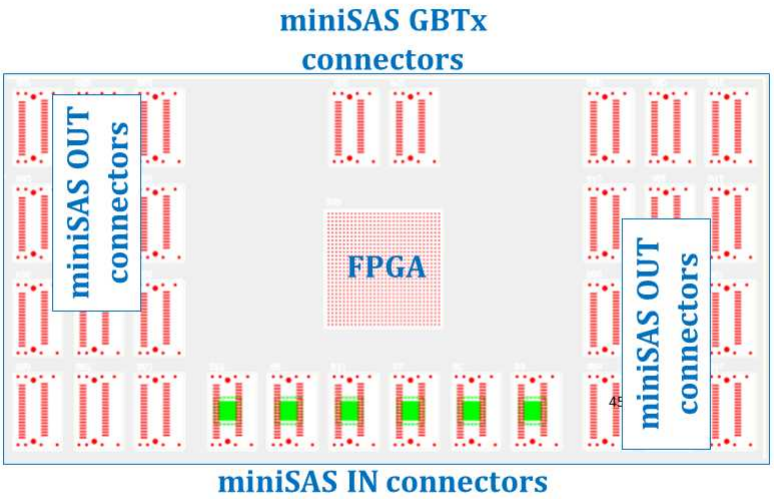


Figure 5.39: Pad trigger board preliminary Layout.

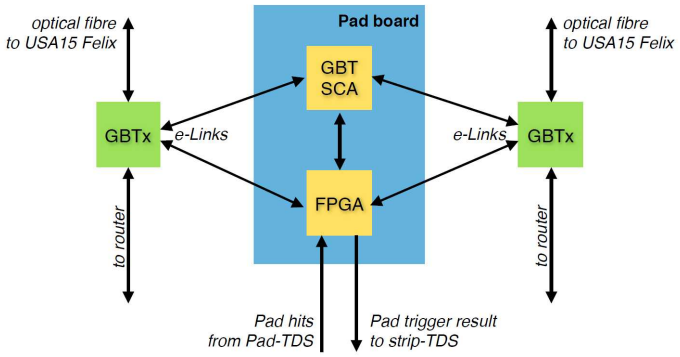


Figure 5.40: Pad trigger board to GBT connections.

will probably be shared with the adjacent Router boards. The GBTx is also used for board configuration/monitoring (in association with a local GBT-SCA chip) and for sending readout/busy/monitor data to the ATLAS TDAQ and control system.

6 CONCLUSIONS

In this PhD work I presented the three high-speed serial links which I developed during the PhD course.

The first link is an auto adaptive serial link capable to dynamically modify its internal structure in order to work at the highest transmission rate, according to a given BER. For this purpose, it was developed to analyse the incoming stream, at different line rates, checking the quality of the transmission for each f them. In order to do the test, it builds a 2-D Eye Diagram by using the *RX Margin Analysis* features of the GTX.
I developed the full internal architecture which is in charge of varying the GTX settings, in order to work at different line rate, to drive the Eye scan analysis and to evaluate the results of the test for each line rate. Since in the project, the Picoblaze microprocessor was used, I took care of writing the VHDL code for the FPGA logic and the software for the microprocessor. The architecture implemented and the tests done are illustrated. The work was also presented at the RT2016 conference (poster and mini-oral) and published in the IEEE Transactions on Nuclear Science journal [44].

The second part of my PhD was devoted to the feasibility studies of two fixed-latency serial links for the Level-1 ATLAS Trigger system. The first link will be implemented on the new MUCTPI Interface board, one of the off detector electronics foreseen in the new muon Barrel system.
I took care of the design of the first prototype PCB board, which will be equipped with an FPGA device. In this framework, I took care of studying and designing the chain for the power supply of the FPGA, as well as the distribution tree of the LHC clock inside the board. This logic is of fundamental importance for the project which has to respect the request on fixed latency of the Level1 Trigger system. I also take care of implementing a fixed latency serial link in order to test the feasibility of the project.

The third part of the work was dedicated to de development and design of the new pad trigger board. This board will perform the sTGC fast-trigger logic. It will receive a data stream at 6.4 Gbps for 24 for on-detector ASIC chip (pad-TDS) and will send back the trigger decision to the strip-TDS.
I implemented the receiver logic of the board on a GTX transceiver, ensuring a fixed latency, and a full-fixed latency link in order to test the project. I also implemented the decoding logic and the serial output interface. Besides, since the new board will be equipped with a programmable repeater, I wrote the firmware and the software for accessing the repeater configuration registers, by using the embedded MicroBlaze microprocessor. The same logic and codes could be used, in future, for accessing the configuration registers of the FPGA. The implementation and test has been illustrated. The first prototype is in production. Besides, I

collaborated to the design of the prototype, taking care of the power supply chain of the FPGA and testing the repeater and the cable which will be used in the experiments. The work done has been presented, such as the results of the tests.

At the moment, I'm continuing to work on the finalization of the two boards in order to better qualify the work already done and to complete it.

BIBLIOGRAPHY

[1] B. Taylor. «TTC distribution for LHC detectors». In: *Nuclear Science, IEEE Transactions on* 45.3 (1998), pp. 821–828.

[2] O. Boyle et al. «S-LINK, a data link interface specification for the LHC era». In: *IEEE Trans. Nucl. Sci* 45.4 pt 1 (1998), pp. 1845–1848.

[3] H. Agilent. *HDMP 1032–1034 Transmitter-Receiver Chip-Set Datasheet, Agilent, 2001*. URL: www.physics.ohiostate.edu/~cms/cfeb/datasheets/hdmp1032.pdf.

[4] R. Giordano and A. Aloisio. «Fixed-latency, multi-gigabit serial links with Xilinx FPGAs». In: *Nuclear Science, IEEE Transactions on* 58.1 (2011), pp. 194–201.

[5] R. Giordano and A. Aloisio. «Protocol-independent, fixed-latency links with FPGA-embedded SerDeses». In: *Journal of Instrumentation* 7.05 (2012), P05004.

[6] A. Aloisio et al. «S-LINK on a Chip for Embedded Applications». In: *Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), 2012 IEEE*. IEEE. 2012, pp. 1090–1091.

[7] D. A. Sanders et al. «Redundant Arrays of IDE Drives». In: *IEEE Transactions on Nuclear Science* 49.4 (2003), pp. 1834–1840.

[8] H.-K. Kim et al. «A Single-Chip Storage LSI for Home Networks». In: *IEEE Communication Magazine* 43 (5 2005), pp. 141–148.

[9] S. A. W. Group, ed. *Serial ATA*. 2007. URL: http://www.sata-io.org.

[10] IEEE. *IEEE Standard for Data Delivery and Logical Channels for IEEE 1284 Interfaces*. The Institute of Electrical and Electronics Engineers, 2000. DOI: 10.1109/IEEESTD.2000.92293.

[11] A. Depari et al. «USB Sensor Network for Industrial Applications». In: *IEEE Transaction on Instrumentation and Measurement* 57.7 (2008), pp. 1344–1349.

[12] A. Widmer and P. Franaszek. «A DC-balanced, partitioned-block, 8b/10b transmission code». In: *IBM J. RES. DEVELOP* 27.5 (1983).

[13] C.-S. Yen et al. «G-Link: A chipset for gigabit-rate data communication». In: *Hewlett Packard Journal* 43 (1992), pp. 103–103.

[14] R. C. Walker et al. «A two-chip 1.5-GBd serial link interface». In: *Solid-State Circuits, IEEE Journal of* 27.12 (1992), pp. 1805–1811.

[15] K. K. Saluja. «Linear Feedback Shift Registers Theory and Applications». In: *Department of Electrical and Computer Engineering, University of Wisconsin-Madison* (1987), pp. 4–14.

[16] D. Mitić, A. Lebl, and Ž. Markov. «Calculating the required number of bits in the function of confidence level and error probability estimation». In: *Serbian Journal of Electrical Engineering* 9.3 (2012), pp. 361–375.

[17] M. Miiller, R. Stephens, and R. McHugh. «Total Jitter Measurement at Low Probability Levels, Using the Optimized BERT Scan Method». In: *Design and Test for Multiple Gbps Communication Devices and Systems* (2005), p. 379.

[18] Agilent, ed. *Jitter Analysis Techniques for High Data Rates, Application Note 1432*. 2008. URL: http://cp.literature.agilent.com/litweb/pdf/5988-8425EN.pdf.

[19] J. Hancock et al. «Jitter–Understanding It, Measuring It, Eliminating It. Part 1: Jitter Fundamentals». In: *High Frequency Electronics* 4.4 (2004), pp. 44–50.

[20] J. Hancock and S. Draving. «Jitter—Understanding it, Measuring It, Eliminating It Part 2: Jitter Measurements». In: *High Freq. Electron* (2004), pp. 20–28.

[21] M. P. Li. *Jitter, noise, and signal integrity at high-speed*. Pearson Education, 2007.

[22] Note. *Clock Jitter and Measurement*. 2009.

[23] Note. *An introduction to jitter in communications systems*. 2003.

[24] R. Stephens. *What the dual-Dirac model is and what it is not*. 2006.

[25] N. Radhakrishnan et al. «Stressed jitter analysis for physical link characterization». In: *Electromagnetic Compatibility (EMC), 2010 IEEE International Symposium on*. IEEE. 2010, pp. 568–572.

[26] M. P. Li and J. B. Wilstrup. «On the accuracy of jitter separation from bit error rate function». In: *null*. IEEE. 2002, p. 710.

[27] R. Stephens. «Jitter analysis: The dual-Dirac model, RJ/DJ, and Q-scale». In: *Agilent Technologies Whitepaper, Literature Number* (2004).

[28] B. Iantovics and R. Kountchev. *Advances in Intelligent Analysis of Medical Data and Decision Support Systems*. Springer, 2014. ISBN: 978-3-319-00466-2. DOI: 10.1007/978-3-319-00467-9.

[29] A. Y. Zomaya and Y. C. Lee. *Energy efficient distributed computing systems*. Vol. 88. Wiley & Sons, 2012.

[30] H. F.-W. Sadrozinski and J. Wu. *Applications of field-programmable gate arrays in scientific research*. CRC Press, 2010.

[31] F. Lemke et al. «A unified DAQ interconnection network with precise time synchronization». In: *Nuclear Science, IEEE Transactions on* 57.2 (2010), pp. 412–418.

[32] Aliaga et al. «PET System Synchronization and Timing Resolution Using High-Speed Data Links». In: *IEEE Transactions on Nuclear Science* 58 (4 2011), pp. 1596–1605.

[33] H. Le Provost et al. «A readout system-on-chip for a cubic kilometer submarine neutrino telescope». In: *Journal of Instrumentation* 6.12 (2011), p. C12044.

[34] N. Ahmad et al. «ALICE Technical proposal». In: *CERN/LHCC* (1995), pp. 95–71.

[35] A. Collaboration et al. «ATLAS detector and physics performance Technical Design Report, Volume I». In: (1999).

[36] B. Collaboration et al. *Technical Design Report*. Stanford Linear Accelerator Center, 1995.

[37] C. I. Collaboration et al. *The CDF II Detector Technical Design Report*. Fermi National Accelerator Laboratory, 1996.

[38] C. Collaboration. «Technical proposal». In: *CERN/LHCC* 94 (1994), p. 38.

[39] P. Piattelli, N. collaboration, et al. «The neutrino mediterranean observatory project». In: *Nuclear Physics B-Proceedings Supplements* 143 (2005), pp. 359–362.

[40] A. Aloisio et al. «Emulating the GLink chip set with FPGA serial transceivers in the ATLAS Level-1 Muon trigger». In: *Nuclear Science, IEEE Transactions on* 57.2 (2010), pp. 467–471.

[41] H. Van der Bij et al. «S-LINK, a data link interface specification for the LHC era». In: *Nuclear Science Symposium, 1996. Conference Record., 1996 IEEE*. Vol. 1. IEEE. 1996, pp. 465–469.

[42] O. Boyle, R. McLaren, and E. van der Bij. «The S-LINK interface specification». In: *ECP division CERN* (1997).

[43] M. B. Marin et al. «The GBT-FPGA core: features and challenges». In: *Journal of Instrumentation* 10.03 (2015), p. C03021.

[44] A. Aloisio et al. «A Frequency Agile, Self-Adaptive Serial Link on Xilinx FPGAs». In: *Nuclear Science, IEEE Transactions on* 62.3 (2015), pp. 955–962.

[45] A. Aloisio et al. «Design, implementation and test of the timing trigger and control receiver for the LHC». In: *Journal of Instrumentation* 8.02 (2013), T02003.

[46] A. Zibell. «Development of a read out driver for ATLAS micromegas based on the Scalable Readout System». In: *Journal of Instrumentation* 9.01 (2014), p. C01038.

[47] S. Martoiu, H. Muller, and J. Toledo. «Front-end electronics for the Scalable Readout System of RD51». In: *Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), 2011 IEEE*. IEEE. 2011, pp. 2036–2038.

[48] B. Zhang et al. «Real-time performance analysis of adaptive link rate». In: *Local Computer Networks, 2008. LCN 2008. 33rd IEEE Conference on*. IEEE. 2008, pp. 282–288.

[49] C. Werner et al. «Modeling, simulation, and design of a multi-mode 2-10 Gb/sec fully adaptive serial link system». In: *Custom Integrated Circuits Conference, 2005. Proceedings of the IEEE 2005*. IEEE. 2005, pp. 709–716.

[50] C. Gunaratne et al. «Reducing the energy consumption of Ethernet with adaptive link rate (ALR)». In: *Computers, IEEE Transactions on* 57.4 (2008), pp. 448–461.

[51] X. Shiny and R. Jagadeeshkannan. «A vigorous distributive approach to adaptive link scheduling in Mobile Ad-Hoc Networks». In: *Current Trends in Engineering and Technology (ICCTET), 2013 International Conference on*. IEEE. 2013, pp. 161–163.

[52] S. Bluetooth. «Bluetooth core specification version 4.0». In: *Specification of the Bluetooth System* (2010).

[53] J. Kim and M. A. Horowitz. «Adaptive supply serial links with sub-1-V operation and per-pin clock recovery». In: *Solid-State Circuits, IEEE Journal of* 37.11 (2002), pp. 1403–1413.

[54] I. Kuon, R. Tessier, and J. Rose. «FPGA architecture: Survey and challenges». In: *Foundations and Trends in Electronic Design Automation* 2.2 (2008), pp. 135–253.

[55] U. Farooq, Z. Marrakchi, and H. Mehrez. «FPGA architectures: An overview». In: *Tree-based Heterogeneous FPGA Architectures*. Springer, 2012, pp. 7–48.

[56] Xilinx. *Xilinx 7 Series FPGAs Overview*. 2014. URL: www.xilinx.com/.

[57] A. Athavale and C. Christensen. «High-speed serial I/O made simple». In: *Xilinx Inc* 4 (2005).

[58] Xilinx. *7 Series FPGA UG482 GTP Transceiver User Guide*. 2014. URL: www.xilinx.com/support/.../user.../ug482_7Series_GTP_Transceivers.pdf.

[59] Xilinx. *7 Series FPGA UG476 GTX Transceiver User Guide*. 2015. URL: www.xilinx.com/support/.../user.../ug476_7Series_Transceivers.pdf.

[60] M. Jenkins and D. Mahashin. «Eye Scan with MicroBlaze Processor MCS». In: (2012).

[61] I. Xilinx. «Microblaze processor reference guide». In: *reference manual* 23 (2006).

[62] Xilinx. *7 Series FPGA UG470 Configuration User Guide*. 2015. URL: www.xilinx.com/support/documentation/user_guides/ug470_7Series_Config.pdf.

[63] Xilinx. *PicoBlaze 8-bit Embedded. Microcontroller User Guide*. 2011. URL: www.xilinx.com/support/documentation/ip_documentation/ug129.pdf.

[64] S. Wolfram. *Mathematica*. URL: http://www.wolfram.com.

[65] Faster. *FM-S14 Quad SFP/SFP+ transceiver FMC*. 2016. URL: http://www.fastertechnology.com/fileadmin/pdf-forms/S14_Product_Briefv2.pdf.

[66] 3M. *High Routability Internal miniSAS Cable Assembly, 68-position*.

[67] T. LHC Study Group et al. *The large hadron collider, conceptual design*. Tech. rep. CERN/AC/95-05 (LHC) Geneva, 1995.

[68] A. A. Alves Jr et al. «The LHCb detector at the LHC». In: *Journal of instrumentation* 3.08 (2008), S08005.

[69] S. L. Glashow and M. Gell-Mann. «Gauge theories of vector particles». In: *Annals of Physics* 15.3 (1961), pp. 437–460.

[70] S. Weinberg. «A model of leptons». In: *Physical review letters* 19.21 (1967), p. 1264.

[71] A. Salam and N. Svartholm. «Elementary particle theory». In: *Almquist and Wiksell, Stockholm* (1968), p. 367.

[72] N. Cabibbo. «Unitary symmetry and leptonic decays». In: *Physical Review Letters* 10.12 (1963), p. 531.

[73] M. Kobayashi and T. Maskawa. «CP-violation in the renormalizable theory of weak interaction». In: *Progress of Theoretical Physics* 49.2 (1973), pp. 652–657.

[74] Z. Maki, M. Nakagawa, and S. Sakata. «Remarks on the unified model of elementary particles». In: *Progress of Theoretical Physics* 28.5 (1962), pp. 870–880.

[75] B. Pontecorvo. «Neutrino Experiments and the Question of Leptonic-Charge Conservation.» In: *Old and New Problems in Elementary Particles* (2012), p. 251.

[76] T. Skwarnicki. «Observation of J/ψp Resonances Consistent With Pentaquark States». In: (2015).

[77] P. W. Higgs. «Broken symmetries, massless particles and gauge fields». In: *Physics Letters* 12.2 (1964), pp. 132–133.

[78] F. Englert and R. Brout. «Broken symmetry and the mass of gauge vector mesons». In: *Physical Review Letters* 13.9 (1964), p. 321.

[79] J. Ellis. «Limits of the standard model». In: *arXiv preprint hep-ph/0211168* (2002).

[80] H. Georgi and S. L. Glashow. «Unity of all elementary-particle forces». In: *Physical Review Letters* 32.8 (1974), p. 438.

[81] G. Ross. «Grand unified theories, Frontiers in Physics». In: (1984).

[82] A. Collaboration et al. «ATLAS Magnet System Technical Design Report». In: *Technical Design Report ATLAS. CERN, Geneva* (1997).

[83] C. ATLAS. *Barrel Toroid Technical Design Report*. Tech. rep. CERN/LHCC/97-19, 1997.

[84] C. ATLAS. *End-Cap Toroids Technical Design Report*. Tech. rep. CERN/LHCC/97-20, 1997.

[85] A. Collaboration et al. «ATLAS Inner detector technical design report». In: *Technical Design Report ATLAS. CERN, Geneva* 216 (1997).

[86] S. Haywood et al. «ATLAS inner detector: Technical Design Report, 2». In: *Technical Design Report ATLAS. CERN, Geneva* 28 (1997), pp. 29–32.

[87] M. Capeans et al. *ATLAS insertable B-layer technical design report*. Tech. rep. ATLAS-TDR-019, 2010.

[88] A. Collaboration et al. «Calorimeter performance technical design report». In: *CERN/LHCC* (1996), pp. 96–40.

[89] A. Collaboration et al. *ATLAS tile calorimeter: Technical design report*. CERN, 1996.

[90] A. Collaboration et al. *ATLAS liquid argon calorimeter technical design report*. 1996.

[91] A. Collaboration et al. «Atlas muon spectrometer technical design report». In: *CERN/LHCC* 97.22 (1997), p. 281.

[92] A. Brandt, S. Xella, and X. Wu. *Atlas trigger perforamance*. Tech. rep. Tech. Rep. ATL-COM-PHYS-2008-067, CERN, Geneva, 2008.

[93] N. Tal Hod et al. *Search for Lepton Flavor Violation in* $\tau \longrightarrow 3\mu$ *Decays*. Tech. rep. ATL-COM-PHYS-2014-1271. Geneva: CERN, Oct. 2014. URL: https://cds.cern.ch/record/1951374.

[94] A. Aloisio et al. «The Read-out Driver for the RPC of the ATLAS Muon Spectrometer». In: *Real-Time Conference, 2007 15th IEEE-NPSS*. IEEE. 2007, pp. 1–7.

[95] Avago. *AFBR-709DMZ,10Gb/1Gb Ethernet, 850nm SFP+ Transceiver*. 2013.

[96] T. Instruments. *8-BIT UNIVERSAL BUS TRANSCEIVER AND TWO 1-BIT BUS TRANSCEIVERS WITH SPLIT LVTTL PORT, FEEDBACK PATH, AND 3-STATE OUTPUTS*. 2010.

[97] T. Instruments. *LMK03000 Family Precision Clock Conditioner with Integrated VCO*. 2013.

[98] S. Labs. *Any-Frequency precision clock multiplier jitter attenuator*. 2014.

[99] S. Majewski et al. «A thin multiwire chamber operating in the high multiplication mode». In: *Nuclear Instruments and Methods in Physics Research* 217.1 (1983), pp. 265–271.

[100] Y. Giomataris et al. «MICROMEGAS: a high-granularity position-sensitive gaseous detector for high particle-flux environments». In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 376.1 (1996), pp. 29–35.

[101] H. Raether. *Electron avalanches and breakdown in gases*. Butterworths, 1964.

[102] *IEEE Standard for Ethernet, Section Four*. IEEE Standard 802.3, 2012.

[103] A. Aloisio et al. «High-speed, fixed-latency serial links with FPGAs for synchronous transfers». In: *Nuclear Science, IEEE Transactions on* 56.5 (2009), pp. 2864–2873.

[104] Tektronix. *Data Timing Generator (DTG5078-DTG5274-DTG5334) Data Sheet*. 2011. URL: http://www.tek.com/datasheet/dtg5000-pulse-generator.

[105] T. Instruments. *DS100BR410 Low Power Quad Channel Repeater with 10.3125 Gbps Equalizer and De- Emphasis Driver*. 2013.

[106] *The GBT: A proposed architecture for multi-Gb/s data transmission in high energy physics*. 2007.