Department of Physics and Astronomy
University of Heidelberg

Bachelor Thesis in Physics
submitted by

Carsten Grzesik

born in Görlitz (Germany)

2014

Fast optical readout for the Mu3e experiment

This Bachelor Thesis has been carried out by
Carsten Grzesik

at the
Physikalisches Institut Heidelberg

under the supervision of
Dr. Niklaus Berger

Zusammenfassung

Für das Mu3e Experiment, das den Lepton-Flavour verletzenden Zerfall $\mu^+ \rightarrow e^+e^+e^-$ mit einer Sensitivität von 10^{-16} nachweisen soll, ist ein triggerloses Auslesesystem vorgesehen. Die Spuren von bis zu $2 \cdot 10^9$ Zerfallselektronen/s werden von einem Siliziumdetektor mit 280 Millionen Pixeln vermessen. Jede Elektron-Trajektorie wird in der Filter-Farm rekonstruiert und ausgewählte Ereignisse gespeichert. Dadurch muss das Auslesesystem eine hohe Bandbreite von ungefähr 1 Tbit/s bereitstellen. Durch die Detektorgeometrie und -größe steht für das Auslesesystem (Kabel, Elektronik) nur sehr wenig Raum zur Verfügung. Da optische Fasern bei kleinem Platzbedarf eine hohe Bandbreite ermöglichen und zusätzlich eine galvanische Trennung des Detektors von der Filter Farm bieten, wird eine optische Auslese für Mu3e verwendet.

Im Zuge dieser Arbeit wurde eine optische Signalübertragung auf ihre Eignung für Mu3e untersucht. Hauptsächlich wurde dabei die Bandbreite untersucht, die verschiedene Aufbauten mit möglichst kleiner Bitfehlerrate (BER, engl. bit error rate) ermöglichen. Dabei wurden für eine bidirektionale 8 Kanal Verbindung mit SFP (engl. small form-factor pluggable) Sendeempfängern BER von $< 10^{-16}$ (95 % C.L.) bei 6.4 Gbit/s, sowie $(1.041 \pm 0.008) \cdot 10^{-12}$ bei 8.0 Gbit/s erreicht. Für einen bidirektionalen 4 Kanal Aufbau mit QSFP (engl. quad small form-factor pluggable) Sendeempfängern ergaben sich BER von $(3.29 \pm 1.04) \cdot 10^{-16}$ bei 11.3 Gbit/s und $< 10^{-16}$ (95 % C.L.) bei 9.3 Gbit/s.

Abstract

A trigger less readout system is proposed for the Mu3e experiment, that is designed to search for the lepton flavour violating decay $\mu^+ \rightarrow e^+e^+e^-$ with a sensitivity of 10^{-16}. The tracks of up to $2 \cdot 10^9$ decay electrons per second are detected in a silicon pixel detector with 280 million pixels. Each electron track will be reconstructed in filter farm PCs and selected tracks are stored. Thus the readout system has to provide high data rates of about 1 Tbit/s. There is little space available for all readout components (cables, electronics) due to the detector geometry and size. There are several reasons for using an optical readout for the Mu3e detector, namely the high data bandwidth combined with low space requirements and a galvanic separation of the detector from the filter farm.

In the course of this thesis, the suitability of an optical data transmission for Mu3e has been tested. Mainly the bit error rates (BER) of different setups have been observed at highest possible data bandwidths. For a 8 channel full-duplex connection with small form-factor pluggable (SFP) transceivers, a BER $< 10^{-16}$ (95 % C.L.) at 6.4 Gbit/s, and $(1.041 \pm 0.008) \cdot 10^{-12}$ at 8 Gbit/s has been measured. For a 4 channel full-duplex connection with quad small form-factor pluggable (QSFP) transceivers, a BER $(3.29 \pm 1.04) \cdot 10^{-16}$ has been reached at 11.3 Gbit/s, and $< 10^{-16}$ (95 % C.L.) at 9.3 Gbit/s.

Contents

Part I

Introduction, Background and Theory

1 Introduction

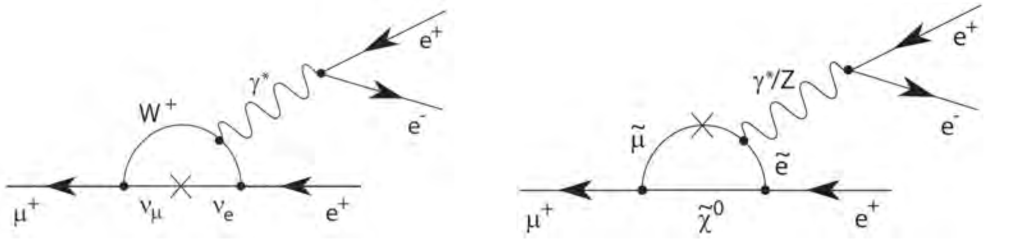
The standard model (SM) of particle physics summarizes our knowledge of elementary particles and their interactions. The SM has passed all experimental tests with flying colors and has again proven its predictive power by the recent discovery of a particle that is very likely the SM Higgs boson [1]. On the other hand, several observations, such as dark matter or neutrino oscillations are not explained by the SM. Furthermore, the gravitational force is not integrated. In summary, these are good reasons to search for physics beyond the standard model (BSM).

The proposed Mu3e experiment [2] searches for the lepton flavour violating decay $\mu^+ \rightarrow e^+e^+e^-$. By measuring the branching ratio of this decay, BSM theories can be tested. The SINDRUM experiment already showed that the branching ratio of the $\mu \rightarrow eee$ decay has to be lower then 10^{-12} [3]. In order to improve this measurement, the Mu3e experiment is designed to reach a sensitivity of 10^{-16}. In order to reach this sensitivity in a reasonable measurement time, a muon beam rate above 10^9s^{-1} is required. Such a rate will be provided by the proposed *High Intensity Muon Beam* (HiMB) at the Paul Scherrer Institut (PSI).

The un-triggered readout of the Mu3e detector combined with the high muon decay rate requires a fast data acquisition system (DAQ) able to handle data rates in the order of Tbit/s. In this thesis, parts of the data readout system of the Mu3e pixel detectors are tested. An optical data transmission is selected because it can handle the required data rate, and provides an electrical separation of the detector from the PC farm. The data ordering, formatting, buffering and switching is realized by field programmable gate arrays (FPGAs). In the filter farm Graphics Processing Units (GPUs) are reconstructing the electron tracks. The main concept of the DAQ is that a single GPU can reconstruct the whole detector for a small timeslice. So the data of all detector components have to reach every GPU in the filter farm which is realized with the help of a switching optical network. The thesis shows the results of testing optical links for the DAQ system.

2 Particle Physics Motivation

Mu3e is an experiment to search for the decay $\mu^+ \rightarrow e^+e^+e^-$. It is designed to reach a sensitivity of 10^{-16}. The decay is forbidden in the standard model (SM) of particle physics. In the SM the lepton flavour is a conserved quantity. However, there have been observations of lepton flavour violating reactions based on neutrino mixing. Figure 2.1a shows a possible Feynman diagram for the $\mu^+ \rightarrow e^+e^+e^-$ reaction through neutrino mixing. The branching ratio is $\ll 10^{-50}$ in the SM with neutrino mixing and thus unobservable. Models beyond the SM are predicting much higher branching ratios based on new particles involved, e.g. additional Higgs bosons or super-symmetric (SUSY) particles [4]. Mu3e is searching for such decays.



(a) $\mu \rightarrow eee$ by neutrino mixing [2]

(b) $\mu \rightarrow eee$ involving possible super-symmetric particle ($\tilde{\chi}^0$) [2]

Figure 2.1: Feynman diagrams for possible $\mu \rightarrow eee$ decays.

The SINDRUM experiment has been searching for the $\mu \rightarrow eee$ decay already. From 1983 to 1986 it was operating at the PSI, did not detect the decay, and pushed the branching ratio limit to $1.0 \cdot 10^{-12}$ at 90% C.L. [3].

3 Mu3e

3.1 Mu3e Setup

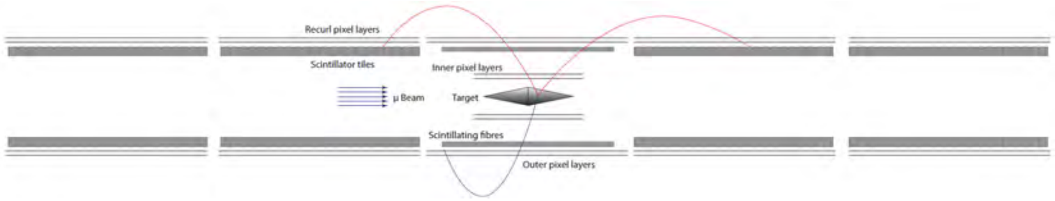


Figure 3.1: Scheme of full detector cut along beam axis including possible tracks [2]

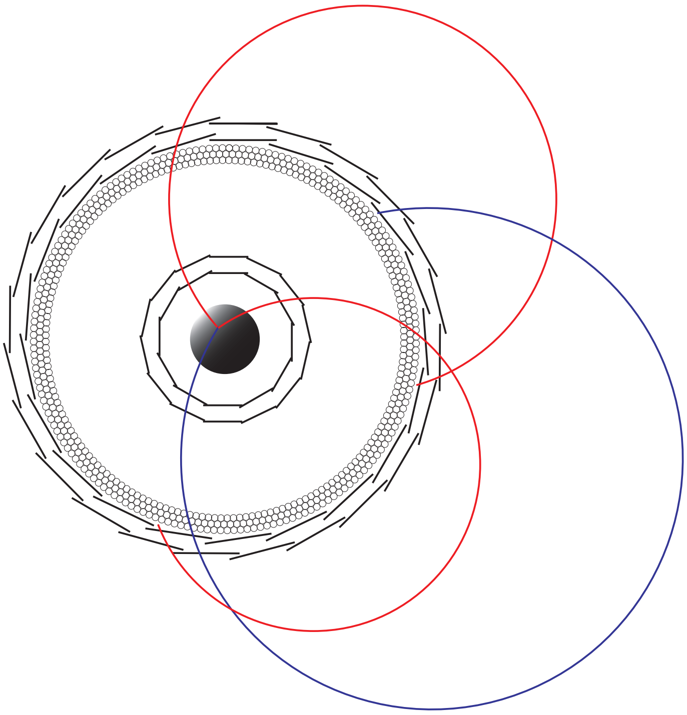


Figure 3.2: Schematic view of middle cylinder cut transverse to the beam axis with possible positron (red) and electron (blue) tracks. Fibres are not drawn to scale.

The Mu3e experiment is designed to measure or exclude the decay $\mu \rightarrow eee$

with a sensitivity of about 10^{-16}. To reach this goal, it is necessary to run at high muon decay rates up to 2 GHz. The muons will be stopped in a hollow double cone shaped aluminum target. The whole setup will be integrated in a solenoidal magnetic field produced by a superconductive magnet that bends the electron tracks to measure their momentum. The decay electrons and positrons are detected by silicon pixel sensors and scintillating fibers and tiles. To suppress the irreducible background decay $\mu \rightarrow eee\nu\nu$, a very high momentum resolution is required in order to detect the missing momentum carried away by the neutrinos. To get a high geometrical acceptance and measure recurling tracks, which greatly improves the momentum resolution, the detector is designed as a five-cylinder setup shown in figure 3.1. Combinatorial background will be suppressed by good time and vertex resolution. These aims will be reached by using the scintillating tiles and fibers for precise time measurement and as few material in the active part of the detector as possible to reduce scattering of decay electrons and positrons for high vertex and momentum resolution. Therefore the pixel sensors will be thinned to 50 μm. The length of the whole detector will be about 2 m. Around the target there are two inner pixel layers of about 12 cm length and 1.9 cm and 2.9 cm radius. Scintillating fibres are placed in three layers at the inner side of the two outer pixel layers, which have a length of about 36 cm. The outer layer has a mean radius of about 9 cm. The design of the recurl stations up- and downstream is based on the outer layers of pixel sensors with scintillator tiles inside of the pixel layers. The space for all service devices (e.g. cables, amplifiers, and FPGAs) is limited to the insides of the recurl stations. Besides the galvanic separation and the data rates, less required space is another reason for using an optical instead of an electrical readout.

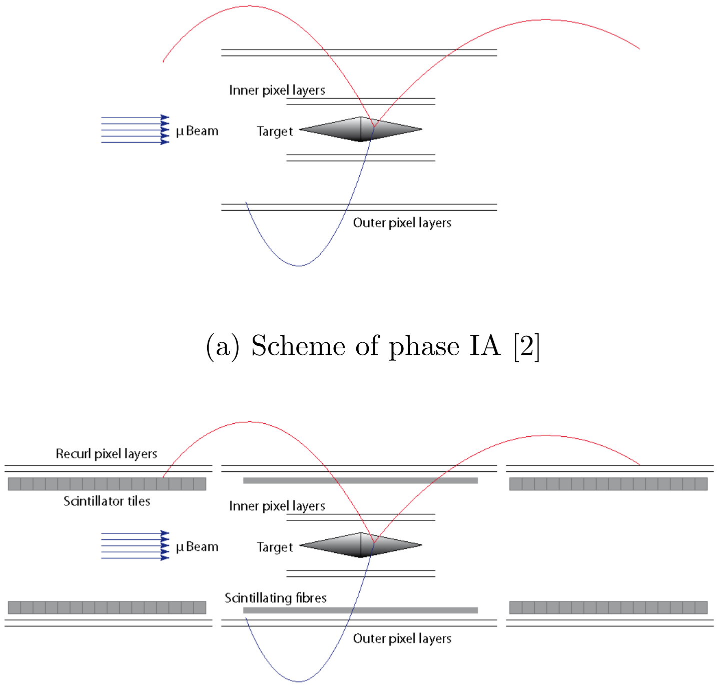
The experiment will be situated at muon beam lines provided by the Paul Scherrer Institut (PSI) in Villigen/Switzerland.

3.2 Detector phases

Due to detector modularity, it will be possible to start using the detector before completing the whole setup. By using just the central cylinder equipped with the four pixel layers and the target, one can run measurements with the setup at stopping rates below those that are already possible with existing beamlines at PSI. This setup is called phase IA, shown in figure 3.3a.

In phase IB , shown in figure 3.3b, the scintillating fibres and a complete recurl station on each side of the central cylinder will be added. This setup will be running at the full stopping rate of up to $10^8\mu$/s available at the πE5 beamline at PSI.

The final setup, described in section 3.1 is called phase II and will be operating at a planned High-Intensity Muon Beam (HiMB) at PSI providing in excess of $10^9\mu$/s.



(a) Scheme of phase IA [2]

(b) Scheme of phase IB [2]

Figure 3.3: Phase I detector setup cuts along the beam axis including possible tracks.

3.3 Mu3e Readout Overview

The Mu3e detector will be read out trigger-less. This means there will be no hardware trigger in the experiment that filters unnecessary data before the data is sent out of the detector. Thus, the data acquisition (DAQ) system needs to handle the full bandwidth of data produced by the detector components. While measuring, the detector elements will continuously send data to the DAQ. This architecture, that sends readout data continuously, is called push architecture and defines bandwidth requirements for the DAQ.

At the end of the DAQ system, there are the filter farm PCs which perform the online track reconstruction. As tracking in Mu3e is very non-local due to the re-curling tracks, each filter farm PC needs to receive the data from the whole detector for a short time period [2, p.63]. Figure 3.4 shows how the DAQ is proposed (in phase II) to deal with this requirement. Two levels of FPGAs serve as a switching network, producing time sorted data out of location sorted data.

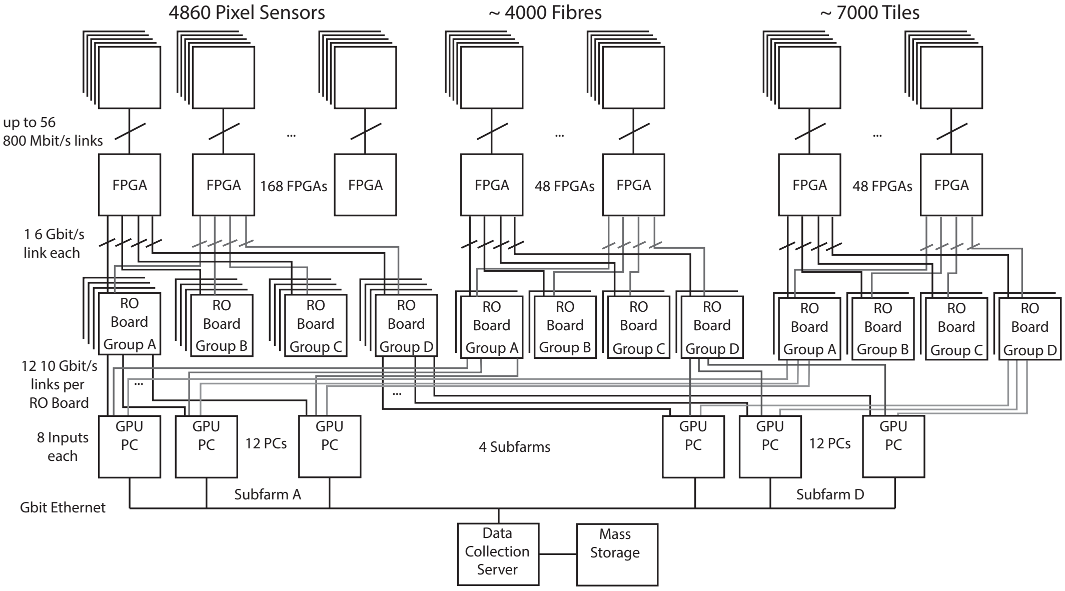


Figure 3.4: Shematic overview of the whole detector readout [5].

3.4 Pixel Detector Readout

In this thesis I will focus on the Mu3e pixel detector readout because the pixel detector development is also performed at PI Heidelberg [6]. The other subdetector readout systems will be implemented in a similar way.

3.4.1 Readout Chain

In figure 3.5, the readout chain of a single MuPix sensor is shown. As shown in figure 3.4, the readout chain mainly consists of three layers. Namely front-end, readout FPGAs, and the filter farm PCs. The MuPix sensor, like all Mu3e sub-detectors, produces zero-suppressed data and sends it off-chip via Low Voltage Differential Signaling (LVDS) link on Kapton-flex prints to the front-end FPGA. The front-end FPGA collects data from multiple pixel sensors and routs it via optical links to the readout FPGAs, which basically act as a switch between the front-end stage and the filter farm PCs.

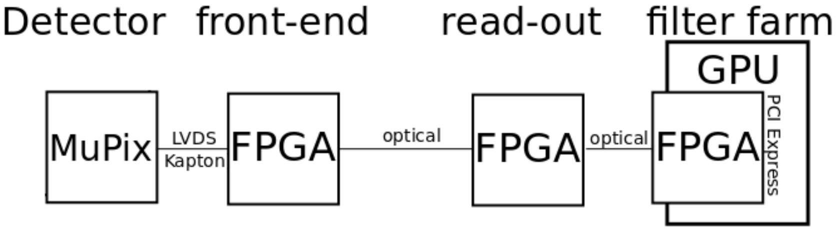


Figure 3.5: Shematic readout chain for a pixel sensor.

3.4.2 MuPix Sensor

The tracking detector of the Mu3e is built from silicon pixel sensors in the novel High Voltage Monolithic Active Pixel Sensor (HV-MAPS) technology [7]. As the MAPS technology combines the sensor and readout functionality in one device, which can additionally be thinned to 50 μm the amount of material in the tracking region is less then with other sensor concepts (e.g. hybrid sensors). This improves the track reconstruction performance, especially at low track momentum.

The readout part of the chip provides zero-suppressed hit data containing the hit address and a time stamp. The data from the chip is serialized, 8B/10B encoded (see section 4.3.1) and sent to the front-end FPGA located outside of the sensitive area of the detector.

3.4.3 LVDS Flex-print Link

To connect the pixel sensors to the front-end FPGA, a low material budget and low power consuming link is needed. As Kapton[1] is used for the mechanical structure of the pixel modules, it can also be used as a carrier for thin aluminum traces due to its electrical insulation. These flex-print links with a maximal length of 30 cm use a low voltage differential signaling (LVDS) standard with a bandwidth of 800 Mbit/s.

LVDS is a differential data transmission method, which has the advantage of low common mode noise dependence. The low voltage refers to the lower voltage compared to other signal transmission techniques. The differential signal of LVDS is produced by using a 100 Ω resistor applied between the two signal lines. The driver of an LVDS is implemented as current mode driver with a limited current value of 4.5 mA. The impedance of the transmission medium is 100-120 Ω to prevent reflections from the termination resistor.

So the combination of LVDS links implemented on Kapton flex-prints fulfills the requirements for low power consumption, low noise and low material budget for links in the sensitive area of the detector.

3.4.4 Front End FPGA

The front-end FPGA collects the data from 15 pixel sensors for the inner layers and 36 for the outer layers. Due to a higher hit rate for each inner layer pixel sensor they require a higher bandwidth. So the inner layer sensors are connected to the front-end FPGA via 3 LVDS links each, whereas the outer layer sensors have one link to the front-end FPGA. The pixel sensor readout is not strictly time ordered. Hits with later time stamps can reach the FPGA earlier then others. The FPGA buffers the data after ordering it by time

[1] Kapton® is a polyimide film developed by DuPont. It offers a combination of mechanical, and electrical properties that suits for the mechanical structure of the pixel detector and flex print wires.

stamps and routs it to optical transceivers which drive optical links to the readout FPGAs.

3.4.5 Readout Boards

The readout boards will be implemented based on FPGAs as well. The readout boards receive the data of half the central pixel detector or one recurl station within a small time slice. Basically they are working as a switch to rout the data to an idle filter farm GPU via optical links. As shown in figure 3.4, the readout boards are divided into four groups. Each group of readout boards belongs to a filter sub-farm consisting of 12 PCs each.

3.4.6 Filter Farm PCs

The data arrives at the filter farm PCs via optical links coming from the readout FPGAs. Therefor, FPGA cards with Peripheral Component Interconnect Express Generation 2 (PCIe Gen 2) or even Gen 3 ports are installed to transfer the data to the memory of graphics processing units (GPU) using direct memory access (DMA) via the PCIe bus and performing a coordinate transformation from the sensor to a global (laboratory) coordinate system. The GPU performs data selection per track reconstruction and sends the selected data to the PC's memory. The CPU ships the data via Ethernet to the central data storage computer [2, p.71].

3.4.7 Optical Links

There are two forms of optical links in the Mu3e readout system. The first connects the front-end FPGAs and the readout boards, and the second one links the readout boards and the filter farm PCs. Both are using 850 nm wavelength diode emitters in the transceivers combined with 50/125 nm multimode optical fibers. These are industrial standards, and commonly used in particle physics detector systems as well [8].

As described in [5], the links between front end and readout boards will be running at 5 to 6.25 Gbit/s. They are proposed to have a length of about 25 m. The characterization of these links is the core topic of this thesis.

The links from the readout boards to the filter farm are supposed to have a bandwidth of 8.5 to 10 Gbit/s and a length of 10 m.

4 Signal Transmission

4.1 Signal Theory

Signal or data transmission is very important in both everyday life and in particle physics applications. For both cases it has to handle ever higher data bandwidths. In this thesis, the effect of the transmission system on the transmitted signal is studied for the optical readout system used for the Mu3e experiment.

Signal A signal is the time dependent amplitude of a physical observable. In this case it is usually the voltage on both sides of the transmission system, while it can be transferred to other physical quantities in the transmission system, e.g. the intensity of a light signal in an optical transmission scheme. It can be continuous or discrete in time. The continuous signal is also called analog signal, while the discrete one is called digital. For digital data transmission the signal is ideally described by a square wave signal that represents the discrete binary values. For each binary value, there is a defined level of the physical quantity (e.g. voltage, intensity). In reality this square wave signal is slightly deformed due to the fact that every electric circuitry has a capacity and inductance. So it can be seen as a band pass for electrical signals. As the square wave signal can be described as superposition of an infinite amount of sine functions up to infinite frequencies, the signal form depends of the system's frequency bandwidth. The interesting quantity for practical applications is how the signal is received and how good it is converted to a square signal, again. In this thesis this is checked by bit error rate testing (BERT) as described in section 4.4.

4.1.1 Electrical signal transmission

As already mentioned, the electrical signal transmission is based on measuring the voltage amplitude of an electrical circuit. This circuit commonly consists of a signal/current source, wires and a resistor on the receiving side to measure a time-dependent voltage. There are two methods to connect the transmitter and receiver side. Single-ended signaling uses one wire which has an electric potential to a common ground. Differential signaling uses two wires for every signal channel. One channel has the inverted polarity compared to the other. This signaling method is less affected by parasitic signals from outside because both wires are affected nearly the same, as the wires are located nearby to each other. By subtracting both values at the receiver, the parasitic signal is eliminated.

For a digital data transmission, the period of one bit, which is the shortest discrete time step, defines the data transmission rate. For higher data rates the signal transmission usually is more difficult due to higher impact of properties of the transmission system.

4.1.2 Optical signal transmission

Intensity of light is another physical quantity that is used to transmit signals. Usually a laser is converting an electrical signal to an optical one. For a low-loss transmission of light an optical fiber is used. Based on total reflection at the boundary surface between the fiber core and the sheathing of the fiber, the light propagates nearly lossless through the fiber. The receiver consists of a detector, e.g a photo diode, that converts the light amplitude back to an electrical signal.

4.2 Tools

In the following section, I will introduce some of the tools employed in this thesis to study the quality of signal transmission.

4.2.1 Eye Diagram

Besides BERT, the transmission quality can be estimated by so called eye diagrams. This is a sample of graphs for the data signal pattern triggered on the same point, e.g a clock transition. A typical eye diagram is shown in figure 4.1, where the rising edge is less steep than the falling one. The vertical eye opening (height) shows how good the separation between logic values 0 and 1 is and if the threshold of the receiver is exceeded. The horizontal opening (width) indicates how long a level detection is possible compared to the ideal case where the horizontal opening is $\mathrm{T} = \frac{1}{\mathrm{f}}$ with T the period and f the frequency of the square wave. Jitter is a misalignment in the time axis of different data patterns to each other. It can indicate a phase shift between the (recovered) clock and the data pattern. The offset of the crossing level indicates a shift between falling and rising edges of the signal.

4.2.2 Phase Locked Loop (PLL)

A phase-locked loop is a non-linear circuit that correlates the phase of an output signal to a reference signal. The phase difference between these two signals is held constant. Figure 4.2 shows the basic scheme for all variants of PLLs. The phase detector (PD) compares the phase difference between the incoming reference signal ($\mathrm{S_{in}}$) and the feedback signal ($\mathrm{S_{fb}}$) from the output. The PD outputs a signal ($\mathrm{S_p}$) that indicates the phase difference between both incoming signals. This signal passes a filter. In most cases it is a low pass

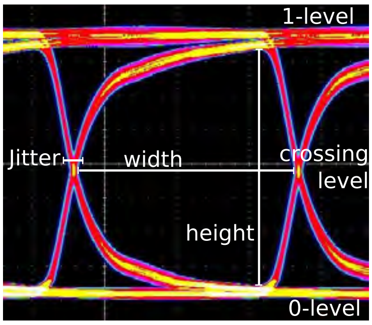


Figure 4.1: Eye diagram indicating eye width, height, levels and jitter.

filter (LPF). The filtered signal ($S_{\text{p-low}}$) is used to control a variable frequency oscillator (VFO). This provides the phase locked output signal (S_{out}), which is also the feedback signal to be fed into the PD again. By tuning all components, the PLL reaches a stable state where it keeps the phase difference between S_{in} and S_{fb} constant. One of the application of PLLs is the synthesis of frequencies.

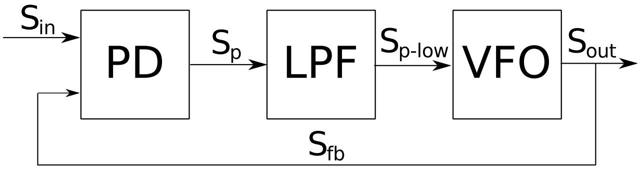


Figure 4.2: Block diagram of the basic PLL concept.

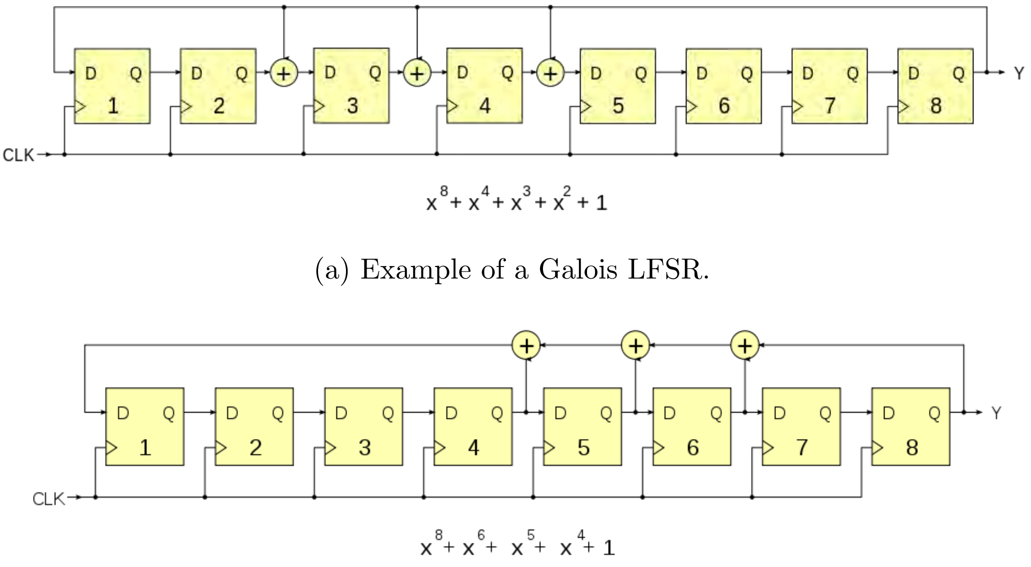
This can be done by adding a frequency divider in the feedback part. The output signal then has the frequency of the reference source multiplied by the factor the frequency divider provides and its phase is constant to the source. Another application is the clock recovery from serial data streams by using the data stream as feedback signal and a reference signal with approximately the frequency of the data stream. In FPGAs both types are used to generate different frequencies out of references from oscillators and to recover the clock from the data on the receiving side.

4.2.3 Linear Feedback Shift Register (LFSR)

A linear feedback shift register is a register that can be used to produce deterministic pseudo random numbers. These are no real random numbers because they depend on each other, while true random numbers are fully indeterministic. Starting with the same seed, a LFSR will always produce the same sequence. Despite this pseudo-randomness, these sequences have basically the same statistic properties as real random numbers and are more easily checked and implemented. Thus they are used here to simulate an arbitrary data stream.

In an electronic realization a LFSR consists of flip-flops that can store one bit and XOR gates to generate the sequence. There are two kinds of implementations for LFSRs, namely Galois and Fibonacci configurations, shown in figure 4.3.

In this thesis mainly the Fibonacci configuration was used with the data pattern being the register. For each clock cycle the register is shifted by one bit and the lowest position bit is generated out of the generator polynomial. Choosing the polynomial fixes the properties of a LFSR. To get the maximal period length of $2^n - 1$ of the pseudo random number pattern, primitive polynomials are used, where n is the polynomial degree. The pattern consisting exclusively of zeros is excluded because the LFSR is trapped in a steady state there.



(a) Example of a Galois LFSR.

(b) Example of a Fibonacci LFSR.

Figure 4.3: General setup of two different implementations of LFSRs. CLK indicates the clock input and Y indicates the output. Underneath the wiring diagram there is the polynomial for each LFSR. Both examples have equal properties. (Public domain picture taken from Wikimedia).

4.3 Coding

Coding is an injective mapping operation between a set of symbols (set of inputs) into a set of other symbols (set of outputs). For digital signal transmission, the coding of transmitted data can be used for many tasks, e.g. cryptography, error detection and correction, or increase in transmission robustness. These are reasons for the variety of different coding methods. In this thesis several coding methods for increasing the transmission quality were evaluated. In the following section these methods will be presented.

4.3.1 8B/10B Encoding

8B/10B is a coding method that transforms an 8 bit data pattern into a 10 bit pattern. There are many ways to implement it. One of the ways was published in 1983 by Widmer and Franaszek [10]. It was developed to ensure a DC balanced data transmission, which means that the long-term ratio of "1" and "0" in the data stream is 1. Each eight bit pattern to encode is divided into a 5 and a 3 bit pattern, those are transformed separately into 6 respectively 4 bit patterns but depend on each other. So 8B/10B coding includes the coding parts 5B/6B and 3B/4B. Some of these 6 and 4 bit patterns have an equal number of "1" and "0". Other patterns have a surplus of two for one digit. For the unequal case, there are always two possibilities to encode an element of the set of inputs. One pattern of the set of outputs consists of two more "0", while the other one is inverted and thus consists of two more "1". The final coded 10 bit pattern is formed by a sequence of coded 6 bit and 4 bit patterns. To ensure DC balancing the running disparity (RD) parameter is used. It can take two values, namely -1 and +1. It starts at -1 and the encoder chooses the pattern for the unequal distribution case with respect to this value. With the disparity of the coded word as the difference of the number of ones and zeros in the pattern, table 4.1 shows the rule for selecting the pattern. In addition,

current RD	possible disparity of coded word	chosen disparity	new RD
-1	±2	+ 2	+1
±1	0	0	±1
+1	±2	- 2	-1

Table 4.1: Running disparity in the 8B/10B encoder scheme for different possible coded words disparities

8B/10B encoded words have a limit of sequential identical characters of 5 in the data stream and additional patterns are provided due to the fact that these are 4 times the number of possible patterns compared to 8 bit patterns. Especially 12 so-called K words, that are not a possible code for an input pattern are available for control functions.

4.3.2 64B/66B Encoding

64B/66B is a scheme to encode a 64 bit pattern in a 66 bit one. The first two bits are used to indicate the type of the following bits. A "01" indicates a following 64 bit data pattern and "10" and a following 8 bit type word followed by a 56 data or control pattern. Both other combinations are not used. This ensures a bit transition at least every 65 bits. Most of the 64B/66B implementations include a scrambler for the 64 bit part to achieve a statistically given DC balancing. In contrast to 8B/10B coding, the running disparity can become much bigger. Thus the DC balance is less bound, which means it is not assured that it will stay near zero. Simulations have shown that it can reach values $> 10^3$ for an 8 bit random number in an 80 bit word. The advantage of 64B/66B over 8B/10B is obviously the lower overhead of 3 % compared to 20 %.

4.3.3 64B/67B Encoding

The Interlaken protocol [12] uses a modified 64B/66B encoding scheme by adding a bit for disparity control. Therefore, the first bit before the 64B/66B preamble is used to indicate whether or not the 64 bit data pattern after the preamble is inverted. This disparity control bounds the running disparity to a range of ± 66 and gains a better DC balancing than 64B/66B encoding by a total overhead of 4.5 %.

4.3.4 Scrambler

A scrambler is a coding scheme that converts a pattern from an input set into a pattern in the same set. So the scrambler does not create any overhead. The function can be generated out of a linear feedback shift register (LFSR) or fixed tables. An example of a scrambler is shown in section 7.1.3.

4.4 Bit Error Rate testing (BERT)

The bit error rate is a measure for the quality of a digital data transmission system. It represents the ratio of wrongly transmitted bits and the total amount of transmitted bits. Different test patterns can be applied to determine the BER. In this thesis, mainly a pseudo random pattern generated by a linear feedback shift register (LFSR) is used. The data can be checked by the receiver through using the same LFSR as the generator. Therefore, the previously received data pattern is used as a seed for the LFSR. By counting the number of bits not matching between the expected pattern and the received pattern, and summing it over all data patterns, the number of error bits is determined. Receiving the wrong pattern[1] will produce a different pattern in the transmitter for the next cycle. This wrongly generated pattern in the receiver is taken into account by halving the number of bit errors there. The bit error rate is the ratio of wrongly received bits and the number of totally transmitted bits:

$$\mathrm{BER} = \frac{\text{number of error bits}}{\text{number of transmitted bits}} = \frac{\mathrm{n_{err}}}{\mathrm{n_{tot}}} \tag{4.1}$$

4.4.1 Determining measurement accuracy

To determine the accuracy of a bit error rate test, it is described as a Bernoulli process. For every received bit there are two possibilities. The received bit is

[1] This means received pattern and generated pattern do not match.

the same as the transmitted one, or not.
The probability distribution is binomial with n being the total number of transmitted bits, p the bit error rate and k the number of error bits. For large values of n, $\mathrm{p} \ll 1$ and finite $\mu = \mathrm{n} \cdot \mathrm{p}$, the binomial distribution converges to the Poisson distribution. For BERT at high data rates, usually n becomes high within fractions of seconds of measurement time and $\mathrm{p} < 10^{-4}$. So the Poisson distribution will be suitable for BERT.
The Poisson distribution is given by:

$$\mathrm{P}(\mathrm{k}) = \frac{\mu^{\mathrm{k}}}{\mathrm{k}!} \mathrm{e}^{-\mu} \tag{4.2}$$

The variance of the Poisson distribution is given by:

$$\mathrm{Var}(\mathrm{k}) = \sigma_{\mathrm{k}}^2 = \mu \tag{4.3}$$

For determining the measurement accuracy, two cases have to be distinguished:

4.4.2 Measurements with bit errors occurring

For BERT where bit errors occur, $\mathrm{k} > 0$ is measured. The measured k value gives the BER:

$$\mathrm{BER} = \mathrm{p} = \frac{\mathrm{k}}{\mathrm{n}} \tag{4.4}$$

The standard deviation of the BER is calculated from the standard deviation of k.

$$\sigma_{\mathrm{p}} = \frac{\sigma_{\mathrm{k}}}{\mathrm{n}} = \frac{\sqrt{\mathrm{k}}}{\mathrm{n}} \tag{4.5}$$

4.4.3 Measurements without bit errors occurring

The approach used in the first case can not be used for measurements where all bits are correctly received for n transmitted bits. A Poisson distribution with an expected value of 0 is ill defined. But an upper limit of the real value $\mathrm{k}_0 = \mu$ can be given with a certain confidence level. It is claimed that the measured value $\hat{\mathrm{k}}$ or lower values are measured with a probability of α. Requiring a confidence level C.L. $= \beta = 1 - \alpha$ and estimating the Poisson distribution for BERT with discrete values for k, one gets:

$$\alpha = 1 - \beta = \sum_{\mathrm{k}=0}^{\mathrm{k}=\hat{\mathrm{k}}=0} \frac{\mu^{\mathrm{k}}}{\mathrm{k}!} \mathrm{e}^{-\mu} = \mathrm{e}^{-\mu} \tag{4.6}$$

For a C.L. of 95% the best estimation for the upper limit of the BER is shown in equation 4.7

$$p = \frac{\mu}{\mathrm{n}} = \frac{-\ln \alpha}{\mathrm{n}} = 2.996 \cdot \frac{1}{\mathrm{n}} \tag{4.7}$$

To summarize, the BER limit for measurements without wrongly transmitted bits is about three times the inverted number of total transmitted bits.
All results shown in chapter 7 follow the described methods. For the measurements without bit errors the stated values are upper limits at 95 % C.L.

Part II

Experimental Setup

5 Hardware

In the following part, the hardware used for the thesis is explained. Figure 5.1 shows a setup with a field programmable gate array development board, the adapter board for small form-factor plugs and an optical fiber. Figure 5.2 shows different setups used to test optical links with small form-factor pluggable (SFP) optical transceiver.

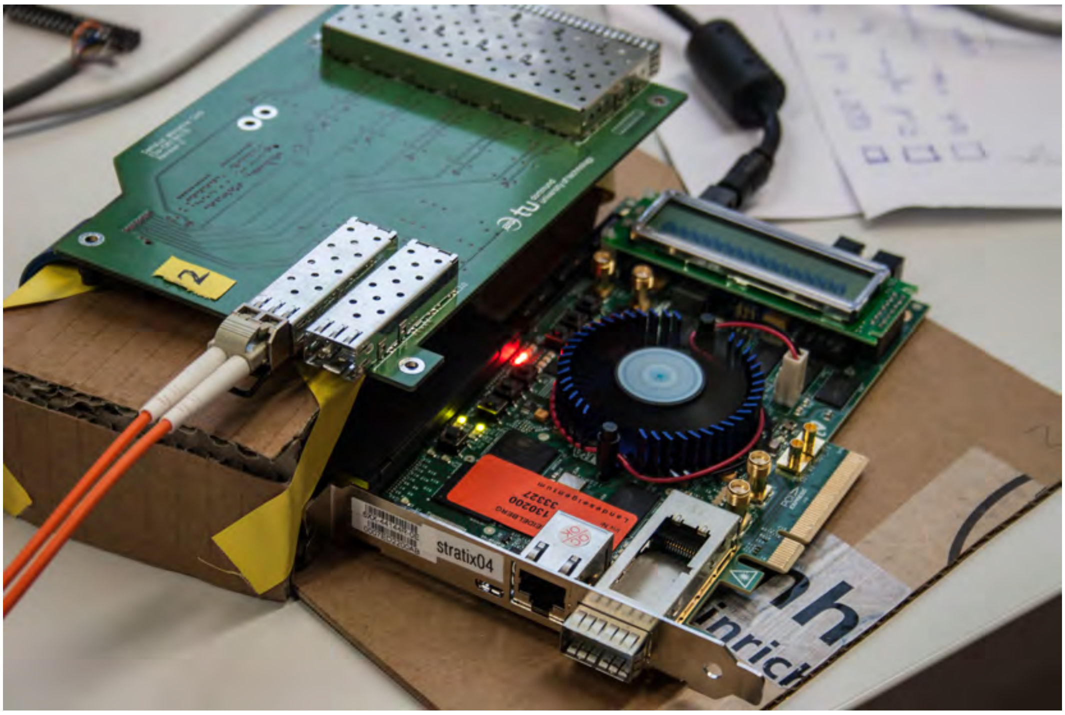


Figure 5.1: Picture showing the setup containing FPGA board, SantaLuz board, and SFP transceiver with optical fiber.

5.1 FPGA

FPGA stands for field programmable gate array, an integrated circuit that can be reprogrammed by the user after manufacturing. It combines advantages of application-specific integrated circuits (ASICs) and software programs on processor-based systems, namely the flexibility of software with the massively parallel processing possible of ASICs. Like hardware implemented ASICs, the FPGA processes the different operations in parallel with separate resources. ASICs are also integrated circuits that are customized to the users specifications but can not be changed after manufacturing. In general they are more

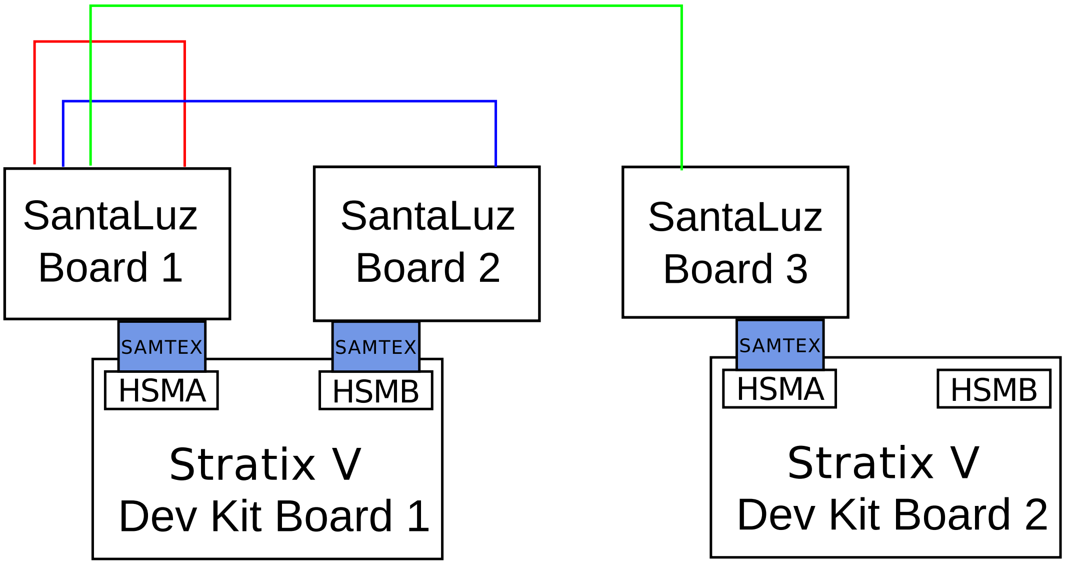


Figure 5.2: Scheme of basic setups used to test optical links. The SantaLuz boards are connected to the FPGA Development boards via Samtec HSMC cables. Different optical fibers connect the same SantaLuz board (red), two SantaLuz boards on different HSMC ports of one FPGA (blue) and two different FPGAs (green).

efficient then FPGAs in terms of logic density, speed and energy consumption. Beside the flexibility through reprogramming, FPGA on the other hand avoid the high initial cost for ASICs, and are thus especially suited for prototyping and small series production.

5.1.1 Architechture

FPGAs mainly consist of three essential parts:

- logic elements (LE)
- interconnects
- Input/Output (I/O) ports

A LE is usually consisting of a lookup table and a register. The lookup table contains a truth table for the logic function of this LE. It is programmed by the user and mostly implemented by the use of static random access memory (SRAM) cells. The output of the lookup table can be additionally registered in a flip-flop and is then wired to other LEs using the wire array of the FPGA. These wires are oriented horizontally and vertically over the entire FPGA and every LE and I/O port can be connected to them. The wires can be linked via programmable switches at each intersection. As a result the wiring can provide connections between the LEs.

5.1.2 Configuration

To configure the FPGA, a synthesis and a fitter program are needed. The synthesis program can create a so-called netlist that specifies all truth table contents and the interconnect paths. Mostly the user implements his desired circuit in an hardware description language (e.g. VHDL or Verilog) and the program synthesizes the netlist. For special applications intellectual property (IP) cores can be used. These are reusable parts of chip designs made by a developer fulfilling special tasks on the FPGA. Two kinds of IP cores exist. Soft IP cores are preconfigured programs or netlists and are used like user's own programs. Hard IP cores are directly implemented hardware circuits on the FPGA and can not be changed anymore, much like ASICs. The fitter program then distributes the elements of the netlist onto the resources available in the FPGA whilst trying to fulfill timing constraints for the propagation of signals between registers.

Because the configuration of SRAM-based FPGAs is lost after a power off, it has to be loaded again at a restart. This can be done from a PC via a JTAG[1] interface or from an on board flash memory.

5.1.3 Implementation

Usually FPGAs are mounted on multi-layer printed circuit boards (PCB) and are combined with other components (e.g. interfaces, oscillators, or memories).

5.2 Stratix V

Altera distributes a Development Kit based on the Stratix V FPGA. The Stratix V is a SRAM-based FPGA produced in a 28nm process designed for high bandwidth applications. We use the "DSP Development Kit, Stratix V Edition" because it combines Alteras Stratix V FPGA with a large amount of useful hardware and software components like [17]:

FPGA features

- 457,000 logic elements, 864 user I/Os
- 36 transceivers
- 174 full duplex low voltage differential signaling (LVDS) links
- 24 phase-locked loops (PLLs)

[1] JTAG is a test access port and boundary-scan architecture for digital integrated circuits defined in IEEE 1149.1 [16]. It is named after the Joint Test Action Group which proposed it.

Development Board features

- FPGA configuration via USB Blaster II or loadable files in 2x512 MB flash storage via Ethernet
- 2 High Speed Mezzanine Card (HSMC) ports
- Peripheral Component Interconnect Express (PCIe) x8 edge connector
- a Quad Small Form-factor Pluggable (QSFP) adapter
- freely configurable 8 dual in-line package (DIP) switches, 8 LEDs, 3 push buttons and a LCD header
- clock circuitry (50MHz, 100MHz, 125MHz and programmable oscillators)
- Quartus II design software for synthesis and ModelSim for simulation

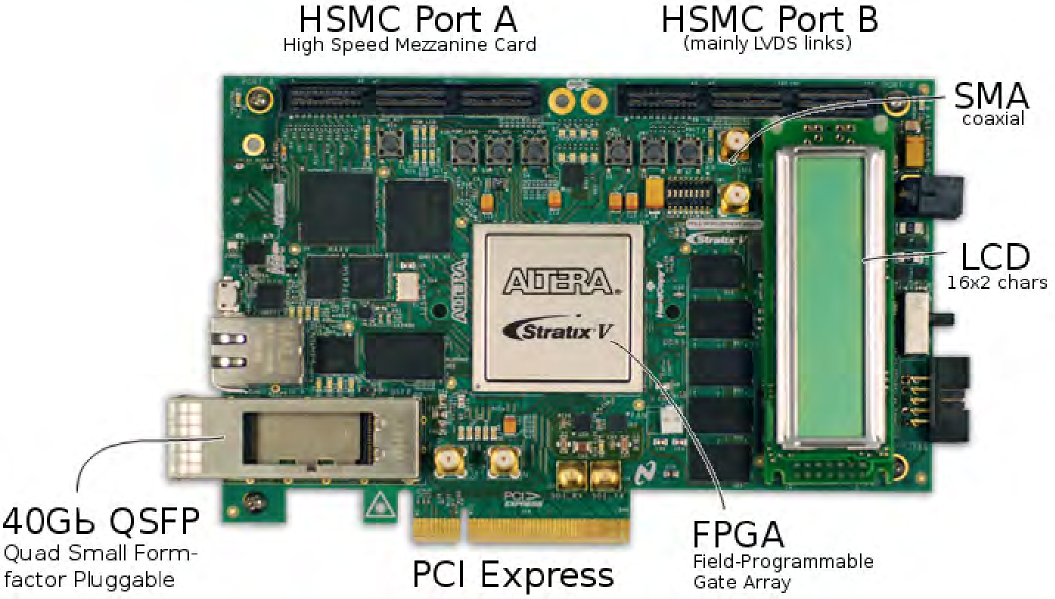


Figure 5.3: Stratix V Development Board [17].

5.3 Stratix V Transceiver

The normal I/O ports of the FPGA can not drive fast serial links because the FPGA works at frequencies < 1 GHz. For connecting the FPGA to high speed data links, dedicated transceivers are required. All transceivers implemented in Stratix V Development Boards are hard IP cores, and are full-duplex which means including both, receiving and transmitting parts. As shown in figure 5.4, Stratix V transceivers are divided into two different parts. The physical medium attachment (PMA) part connects the FPGA to the transceiver channel, serializes the data, and generates the required clocks. The physical coding sub layer (PCS) performs digital processing between the PMA and the FPGA core. In Stratix V devices there are three PCS blocks available: Standard PCS, 10G PCS, and a PCIe Gen3 PCS supporting the PCIe Gen3 Base specification. The transceivers are grouped in 6 channel transceiver blocks, sharing the same reference clock.

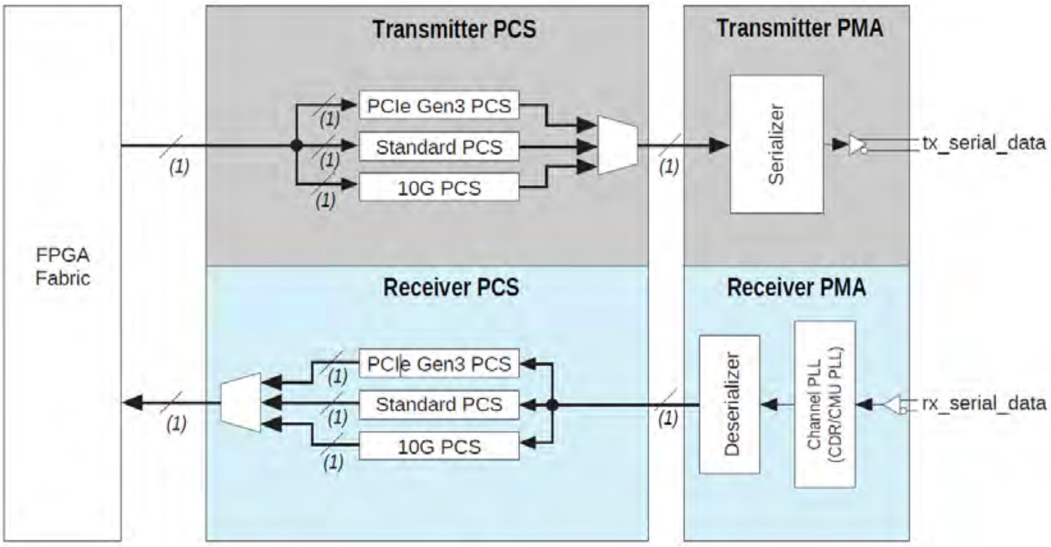


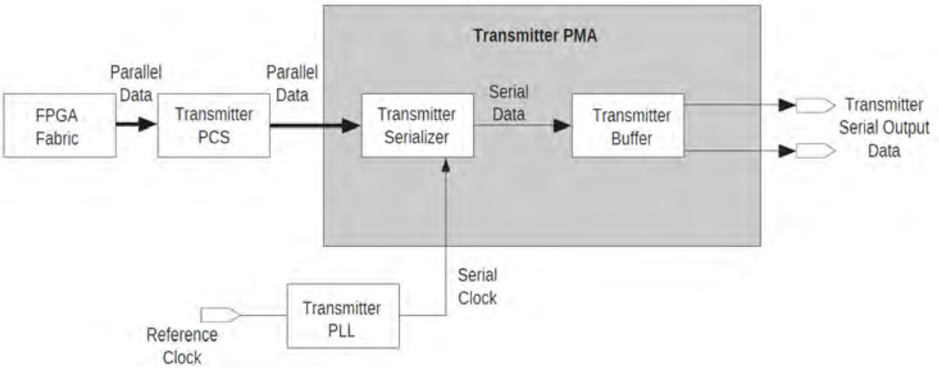
Figure 5.4: Scheme of Stratix V Transceiver [18].

Physical medium attachment (PMA)

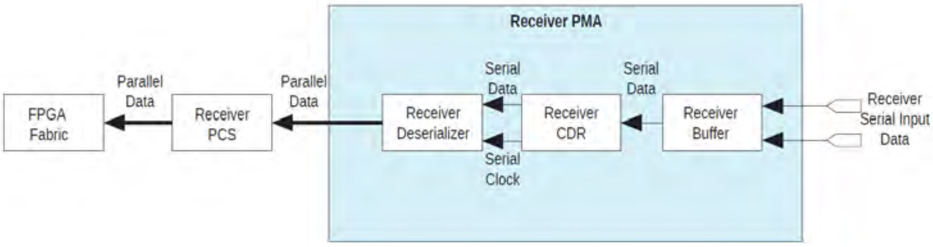
Serializer The incoming low-speed parallel data from the PCS or FPGA framework is converted to serial data with the desired frequency by the serializer. In Stratix V devices parallel data of 8 bit and 10 bit, 16 bit and 20 bit, 40 bit and 64 bit can be serialized. The transmitter serializer sends the data to the transmitter buffer [18, p. 1-20].

Transmitter Buffer The transmitter buffer includes additional circuitry to improve signal integrity and drives the data off-chip. The user can adjust transceiver analog settings, and PCIe receiver detect capability [18, p. 1-21].

Analog Settings The transmitter analog settings can improve signal integrity depending on the transmission hardware, e.g. wires and plugs. These analog settings include programmable output differential voltage, three-tap pre-emphasis, transmitter on-chip termination (OCT), and link coupling [18, p. 1-21], as described below:



(a) Transmitter PMA [18].



(b) Receiver PMA [18].

Figure 5.5: Schemes of transceiver PMAs.

Programmable Output Differential Voltage (V_{OD}) The output differential voltage defines the voltage amplitude of the signal coming out from the transmitter [18, p. 1-21].

Pre-Emphasis The pre-emphasis increases high frequency signal parts of the outgoing data signal. Thus, the rising edge of the signal is steepened, and the data eye can be more opened to compensate attenuation in the data transmission part. There are three pre-emphasis taps that can be changed: pre-tap (16 settings), first post-tap (32 settings), and second post-tap (16 settings). The pre-tap configures the pre-emphasis before the transition, whereas the first post-tap sets it during the bit transition, and the second post-tap sets the pre-emphasis at the following bit. The pre-tap and second post-tap also provide inversion control, which means a kind of deamplification instead of an amplification [18, p. 1-22].

Programmable Transmitter On-Chip Termintaion (OCT) The transmitter buffers are current mode drivers, which means they provide a fixed current value for a digital one, and the V_{OD} value at the transmitter output depends on the termination. The transmitter buffers include on-chip differential termination. The termination is adjusted during the calibration and provides the following values: 85Ω, 100Ω, 120Ω, 150Ω, or OFF. The OFF value is designed for an external termination resistance [18, p. 1-22].

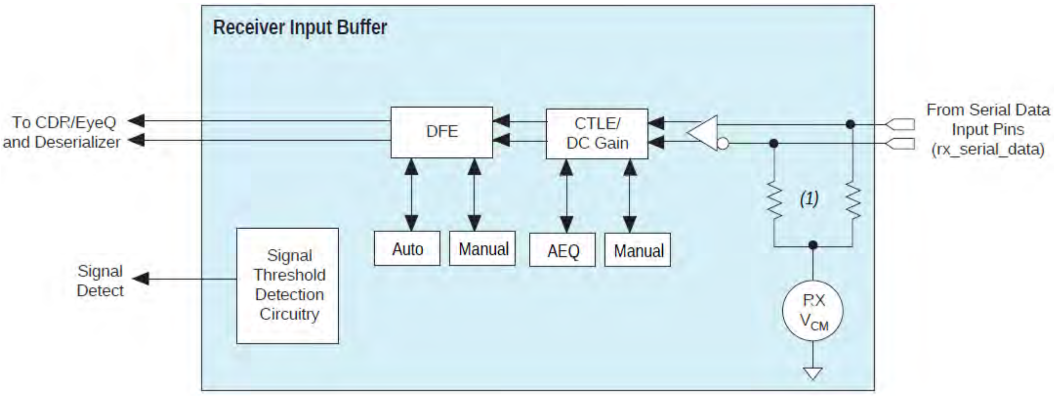

Receiver Input Buffer
To CDR/EyeQ and Deserializer
DFE
CTLE/ DC Gain
From Serial Data Input Pins (rx_serial_data)
(1)
Auto
Manual
AEQ
Manual
Signal Detect
Signal Threshold Detection Circuitry
RX V_{CM}

Figure 5.6: Scheme of receiver buffer [18].

Receiver Buffer A scheme of the receiver buffer is shown in Figure 5.6. The receiver buffer receives the data from the serial data input port and routes it to the CDR and deserializer. It supports several features, that are described in the following.

Receiver Equalizer Gain Bandwidth Depending on the data rate there are two equalizer gain bandwidth modes [18, p. 1-10].

Programmable Transmitter On-Chip Termination (OCT) The receiver buffer supports the same OCT values as the transmitter buffer [18, p. 1-11].

Programmable common-mode voltage (V_{CM}) The receiver buffer provides the required V_{CM} at the receiver input [18, p. 1-11].

DC gain and Continuous Time Linear Equalization (CTLE) The DC gain amplifies incoming signals equally over the whole frequency spectrum. The amount of amplification can be changed by the user, and the upper limit is 8 dB.
To boost the high-frequency parts of the incoming signal, five independently programmable equalization circuits are integrated in the receiver buffer. They provide up to 16 dB frequency boost, and two different modes. In manual mode, the user can tune the different parameters. In adaptive equalization(AEQ) mode this is automatically done by the device based on comparing the incoming frequency spectrum and a reference signal [18, p. 1-11].

Decision Feedback Equalization (DFE) In addition to the equalization from DC gain and CTLE, the DFE boosts the high-frequency parts of the signal by compensating for inter-symbol interference (ISI). In this case the amplitude depends on the previously received bits [18, p. 1-12].

Clock Data Recovery (CDR) The CDR circuit recovers the high-frequency clock from the incoming data stream and by dividing it, the slower parallel clock is generated. The CDR is implemented as a phase-locked loop (PLL) with two different modes. First, the PLL goes in locked-to-reference mode, where the PLL is locked to the phase and frequency of the incoming reference clock. Once the CDR is in locked-to-reference mode and it detects an incoming data stream, it will switch to the locked-to-data mode. In locked-to-data mode the PLL is driven by the incoming serial data, and the reference clock is used to ensure the stability of the recovered frequency [18, p. 1-13].

Receiver Deserializer The receiver deserializer uses the incoming high-speed serial data, the fast serial recovered clock, and the slow parallel recovered clock from the CDR to deserialize the data and forwards it to the receiver PCS or FPGA fabric. As expected, the deserializer supports all the parallel word lengths the transmitter serializer supports [18, p. 1-15].

Bit Slipping As described above, the receiver deserializer uses the incoming serial data to transform them into parallel data words. Since the serial data does not contain any information about the beginning, or end of a data word in the continuous data stream, an alignment has to be done. Therefore the receiver deserializer provides a bit slip feature, that shifts the parallel word by one bit. Additionally there is also a transmitter bit slip feature, that slips a bit in the data words before they are sent to the PMA. This has to be done to eliminate offsets between different transmitter channels [18, p. 1-15].

Physical Coding Sublayer (PCS)

The three PCS types (standard, 10G, PCIe) provide optional functions all implemented in hard IP cores. Figure 5.7 shows the data path in a standard PCS. The whole PCS, as well as any component can be bypassed. Thus the user can select the required options.

Phase compensation FIFOs Each transmitter and receiver channel includes a FIFO to separate the low-speed parallel clock from the user logic and the high-speed serial clock. It can only compensate different phases between the two clocks [18, p. 1-37].

Byte serializer and deserializer The PCS frequency has an upper limit. When the frequency limit is exceeded, the byte serializer and deserializer are required. They can double the word length (e.g. 8 bit to 16 bit) by halving the PCS frequency [18, p. 1-38].

8B/10B encoder and decoder The 8B/10B encoder in the transmitter PCS generates 10 bit code words from 8 bit data using the IEEE 802.3 specification. Furthermore a 1-bit control identifier is generated. When it is asserted the 8 bit word is encoded as a 10 bit control word. The 8B/10B

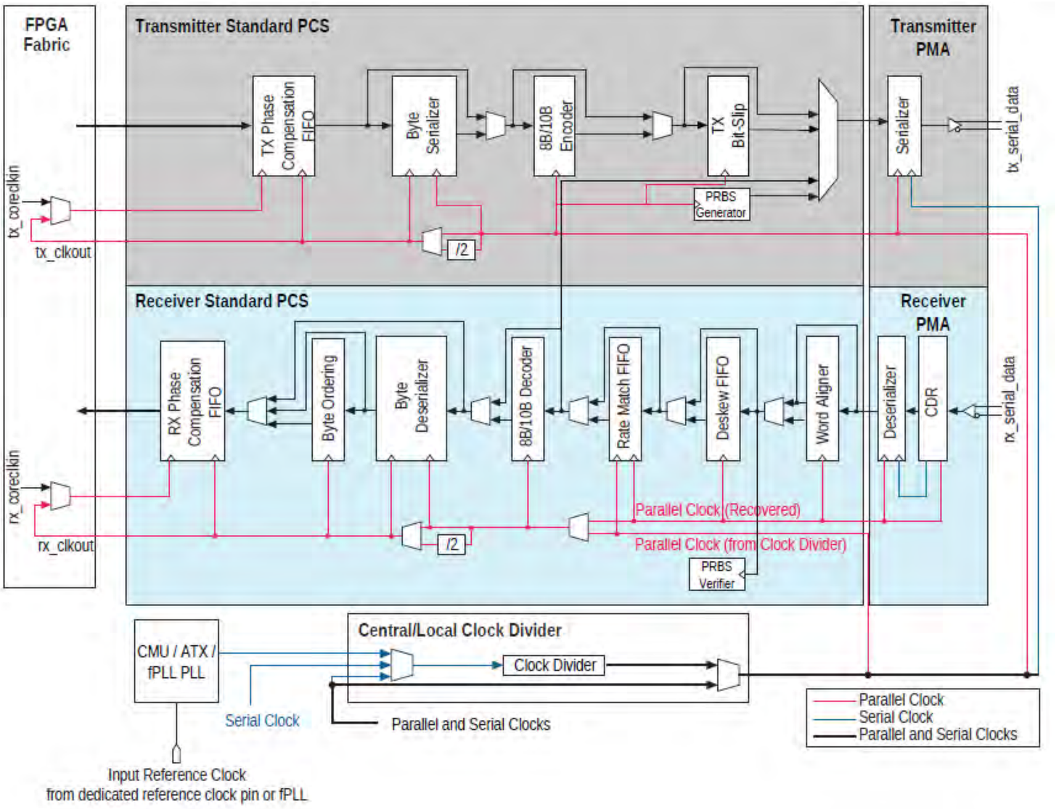


Figure 5.7: Standard PCS data path in Stratix V transceiver [18].

decoder in the receiver PCS decodes the incoming 10 bit coded data to 8 bit words [18, p. 1-39].

10G PCS The 10G PCS provides additional functionality for several data transmission protocols, mainly the 10GBASE-R protocol for 10 Gbit Ethernet transmission as described in IEEE 802.3 clause-49 [19]. Some of these features are listed below.

- cyclic redundancy check (CRC32) generator and checker
- 64B/66B encoder and decoder
- scrambler and descrambler including pseudo random permutation (PRP) generator and verifier
- disparity generator and checker
- Bit error rate (BER) monitoring

For my work this 10G PCS and the third PCS for the PCIe Gen 3 protocol do not really fit the requirements. For more details, refer to [18, p. 1-42].

5.4 HSMC SantaLuz Board

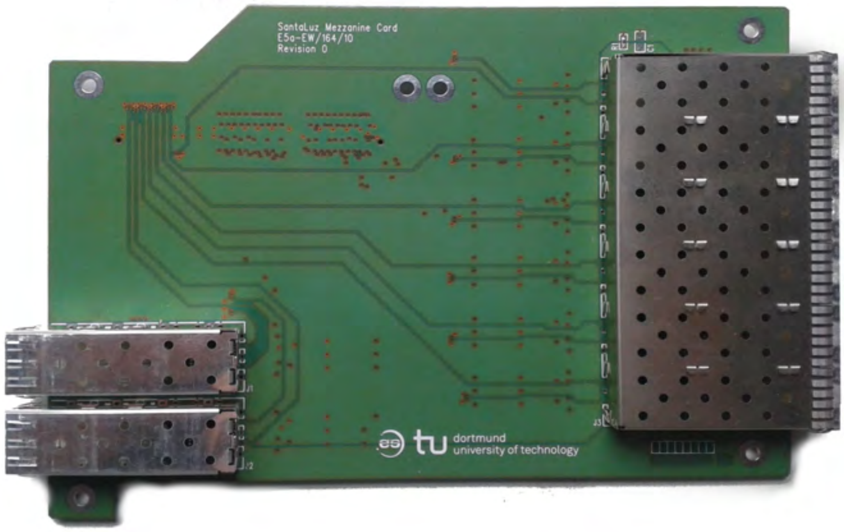


Figure 5.8: SantaLuz board [20].

The Altera Stratix V Development Board does not provide optical output ports. So one has to use adapters for the given ports. Among others we use a SantaLuz Mezzanine Card, shown in figure 5.8, made by TU Dortmund that can be linked to the Development Board via an HSMC cable and has eight small form-factor pluggabble (SFP) ports. Previous measurements have obtained a bit error rate below $3 \cdot 10^{-15}$ at 6.25 Gbit/s [21].

5.5 SFP Plugs

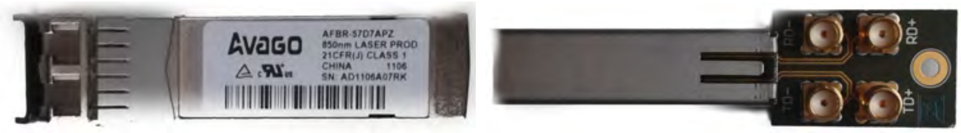
Small form-factor pluggable (SFP) are modules for fast network connections. They provide optical or electrical transceivers. Both versions have been used, and are described later. The SFP transceivers can be plugged to SFP slots.

5.5.1 Optical SFP transceiver

For transmitting and receiving optical signals from/to the electrical SFP ports on the SantaLuz card optical transceivers have to be used. These transceivers, shown in figure 5.9a, provide up to 8.5 Gbit/s signaling rates for multimode optical fibers. The optical signal is produced in a vertical-cavity surface-emitting laser (VCSEL) at $\lambda = 850$ nm [22].

5.5.2 Electrical SFP transceiver

Figure 5.9b shows the TrioFlex SFP2SMA adapter, that was basically used to observe fast serial data on the oscilloscope.



(a) SFP optical transceiver. (b) SFP electrical transceiver.

Figure 5.9: Pictures of SFP plugs.

5.6 QSFP Plugs

The quad small form-factor pluggable (QSFP) cages on the Stratix V Development board can use four transceiver channels and were used with a QSFP transceiver cable assembly made by Molex. The QSFP transceivers which are designed to provide BER of 10^{-18} [23] per link are directly connected to single-mode optical fibers.

6 Software

Using Altera FPGAs for all implementations, Altera software was used, too. The Quartus II software is included in the FPGA Development Kits and has been used to program the FPGAs.

6.1 Altera Quartus II

The Quartus II Software Development Kit is a software for programming FPGAs. The core element is a Hardware Description language (HDL) environment for VHDL and Verilog, or visual programming language. The user can assign the in- and outputs to the I/O pins of the FPGA and the synthesis tool creates netlists from the user defined logic. The Quartus software includes different software tools for analyzing and optimizing the current logic with respect to different parameters. Some of them are described in the following:

MegaWizard The MegaWizard tool provides an access to Altera's predefined Mega Functions that are implemented in Hard IP cores. A GUI is used to configure these functions and the MegaWizard tool creates all required files to implement it into the user's logic.

QSys Like the MegaWizard tool, the QSys system integration tool provides access to predefined functions, mostly hard IP cores. Different from the MegaWizard tool, it adds the possibility to connect these functions by a visual programming language environment, and in the end synthesizes the whole program in one function.

Device Programmer Using this program the compiled netlist (.sof file) can be directly written to the FPGA via an USB connection.

TimeQuest Timing Analysis Based on given clock frequencies, this tool analyzes the timing of the user defined logic. It can report the slack of the failing paths and gives an estimation of how fast the actual realization of the logic can be driven.

PowerPlay Power Analyser Calculates the power consumption of the FPGA based on the user logic. Additionally temperature values and cooling parameters can be estimated.

6.2 Transceiver Toolkit

The Transceiver Toolkit included in the Quartus II SDK is a software component that provides dynamic configuration of FPGA transceivers. It provides

real-time configuration and control tools, for pseudo random bit generator and checker, EyeQ eye diagram tool for receiving signals, and analog tuning of transceiver PMAs, described in section 5.3, with auto tune option. As the required components are implemented in the logic, the toolkit can be connected to the FPGA and detects all transmitters and receivers automatically. Transmitters and receivers in the FPGA can be linked in the toolkit. These links can be tuned manually or with an automatic tuning provided by the toolkit based on BER, or EyeQ. Additionally, the EyeQ feature of the toolkit can be used for manual tuning to estimate a good setting.
The EyeQ tool uses a sampling mode to display an eye diagram on the receiver side of a transceiver channel. So the eye diagrams shown in this toolkit feature are not real eye diagrams in the signal theory sense, but rather are sampled out of multiple runs by shifting the recovered clock by an offset for every run. The BER is measured by specific threshold values in the receiver PMA for a zero and one. The BER for every offset is drawn in the EyeQ diagram and isolines for same BER are drawn for a better overview. It has to be mentioned, that this EyeQ diagram only provides BER down to 10^{-12}. For manual tuning one can estimate the signal integrity by observing the eye opening in amplitude and time.

6.3 ModelSim

ModelSim is a simulation software developed by Mentor Graphics for ASIC and FPGA designs. Even though ModelSim was developed to simulate designs that are written in Verilog and VHDL, the special Altera version that was used does not support such mixed designs. It simulates the behavior of the user design from given HDL files and netlists. The behavior can be simulated by programmable in- and outputs of the design and all signals, including internal ones, can be observed. This is really helpful to test the design before using it on the FPGA or for debug purposes.

Part III

Measurements

7 Results

This chapter describes the results of the measurements that have been done in context of this thesis. For all measurements, the Stratix V Development Kit has been used for data generation, signal processing, transmitting and receiving the signal, and comparing received data with the transmitted one (e.g. Bit Error Rate Tests).

7.1 Measurements with optical SFP transceiver

Unless otherwise mentioned, the SantaLuz Mezzanine Card is linked to the FPGA board's HSMC port A via Samtec HSMC adapter cable, and the SFP optical transceivers are used to drive optical cable for the following measurements, as shown in figure 5.2.

7.1.1 Single channel mode with Transceiver standard configuration

The first measurements have been done without using the transceivers own analog tuning described in section 5.3. So the default settings have been used. As the SantaLuz board provides eight SFP cages, and the whole transmission chain is full-duplex capable, it is possible to run on eight data transmission channels simultaneously.
The following measurements have been done using only one of the eight channels. So one transceiver channel sends the data to a SFP optical transceiver. Another SFP optical transceiver receives the data via the optical fiber and routs the electrical signal to another transceiver channel in the FPGA. Figure 7.1 shows the described setup.

Bit Error Rate Test (BERT) for different data rates

In order to obtain an overview of hardware possibilities an optical transmission setup was chosen which is nearly "out of the box". For bit error rate testing a pseudo-random number generator is used to generate a 64 bit parallel data word by a linear feedback shift register (LFSR), explained in 4.2.3. The LFSR polynomial has a period of 255, which corresponds to an 8 bit random number (rn). So the 64 bit pattern consists of $8 \cdot 8$ bit depending random number words. Discrete data rate values are given by the multiplier generating the fast serial clock by multiplying the reference clock in the transmitter PLL.
As shown in figure 7.3, the BER stays below $3 \cdot 10^{-12}$ (C.L. 95%, as described

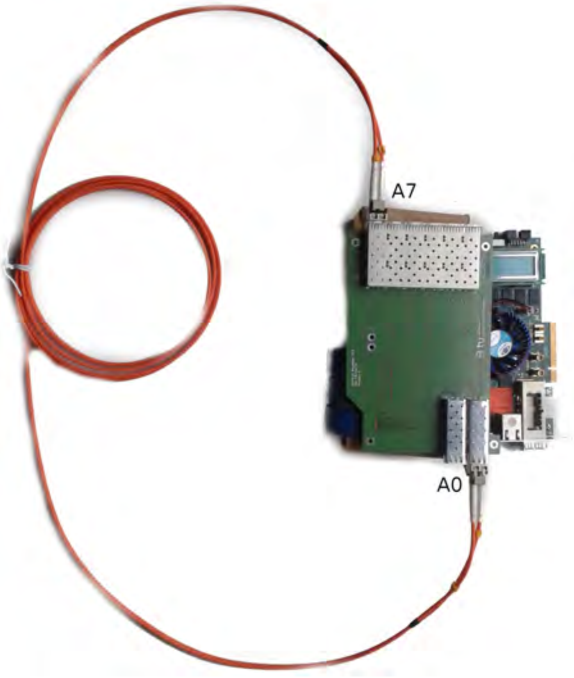


Figure 7.1: BERT Setup in single channel mode using one FPGA

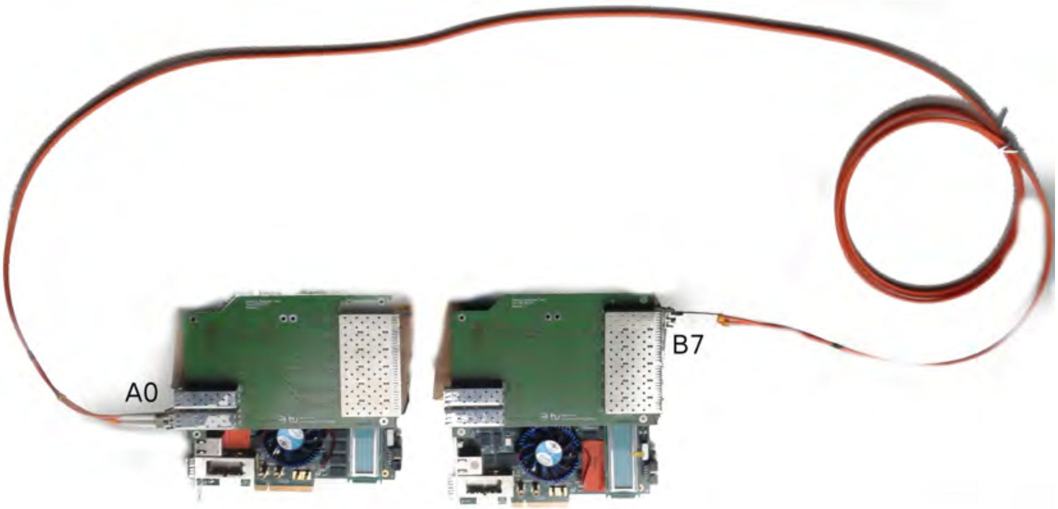


Figure 7.2: BERT Setup single channel mode using two FPGAs

in 4.4.1) for all optical transmission lengths up to a data rate of 6.4 Gbit/s. For 8 Gbit/s the BER is too high for useful data transmissions.
Long term measurements with optical cables of l=3 m and l=50 m have been performed to push the BER limit to lower values. This has been done by using three FPGAs connected via HSMC port A to a SantaLuz board each. The pattern and firmware remains the same as above. The setup is shown in figure 7.4. For l=3 m, $1.31 \cdot 10^{15}$ bits have been transmitted without an error, which gives a BER$< 2.3 \cdot 10^{-15}$ (C.L. 95%).

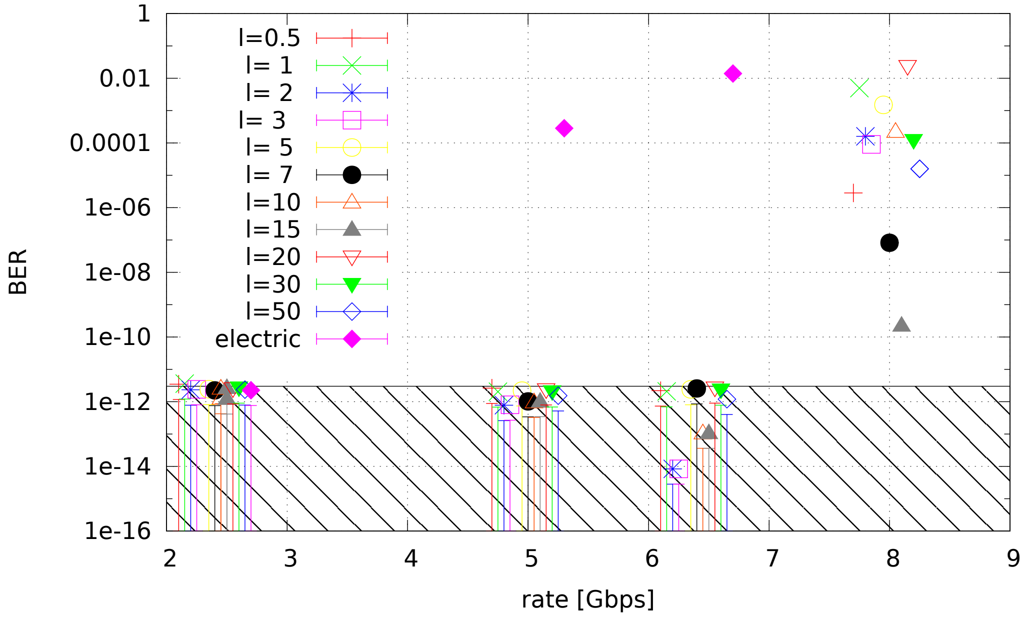


Figure 7.3: Bit error rate measured at different data rates using one channel without analog transceiver tuning. For transmissions without observing a bit error, the measurement was stopped when the number of transmitted bits $> 10^{12}$. For optical transmission several cable lengths l have been tested at data rates of 2.4, 5, 6.4, 8 Gbit/s (shifted for better visibility). "electric" stands for electric transmission using electrical SFP plugs and a SMA coaxial cable with l=0.5 m.

$1.01{\cdot}10^{16}$ bits have been transmitted for l=50 m. In this case the BER$< 3{\cdot}10^{-16}$ (C.L. 95%). In figure 7.2 basically the same setup is shown except that the receiving part is placed on another FPGA. This setup was chosen to check the more realistic situation, where the transmitting and receiving FPGA are not the same to ensure the reference clocks are uncorrelated.
Using this setup, the following results have been reached. For the optical fiber with l=3 m, $1.23 \cdot 10^{14}$ bits have been sent without the occurrence of an error. Thus, the BER$< 2.4 \cdot 10^{-14}$ (C.L. 95%) was observed.
Using the l=50 m fiber the BER$< 2.9 \cdot 10^{-16}$ (C.L. 95%) was observed by transmitting $1.03 \cdot 10^{16}$ bits without an error bit.

Bit Error Rate Test (BERT) for different SFP cages

The SantaLuz Mezzanine board provides eight SFP cages. In the following paragraph results are shown for testing all these cages with the optical transceiver with the above described setup. Therefore each cage was connected to the zeroth one and both directions of transmission were tested. Besides using the HSMC port A of the Stratix V board, the HSMC port B was tested, too. In the HSMC port B only 4 channels are connected to the FPGA fast

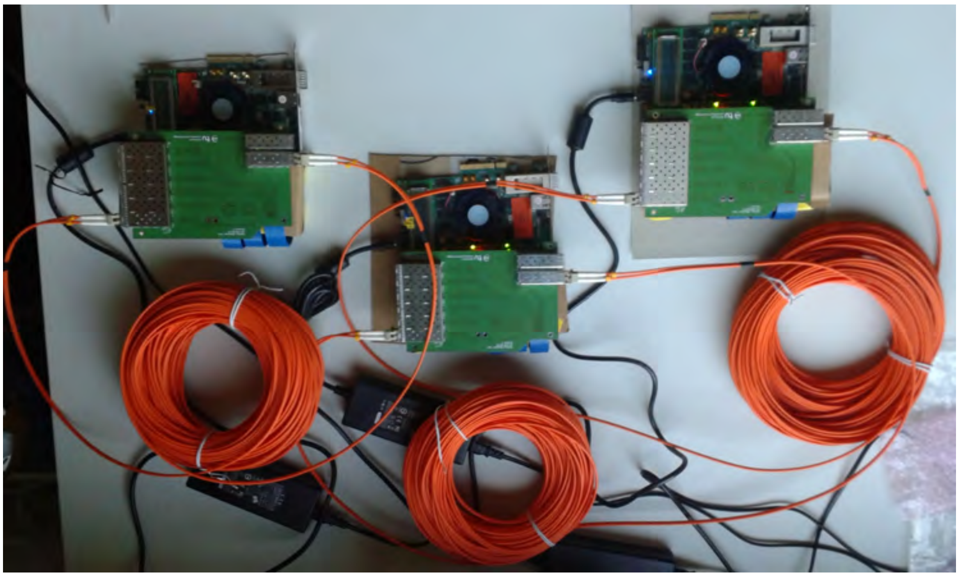


Figure 7.4: Setup for long term BERT at 6.4 Gbit/s using three FPGAs. The picture shows the setup for l=50 m.

transceivers. Thus, only 4 cages of the SantaLuz board can be used there. The measurements are made at 8 Gbit/s transmission rate to obtain possible differences between the channels. For lower rates (up to 6.4 Gbit/s), where no error occurs in a reasonable time (minutes) the differences would not be shown.

Figure 7.5 shows that there are no big differences among the channels 2 to 5 and all channels connected to the B port. The 6th and especially the 7th channel seem to work better connected to the zeroth than the others. This behavior has also been observed qualitatively during other measurements.

Bit Error Rate Test (BERT) for patterns with multiple zeros in a row

As in many electrical circuits, the transmission path includes capacitors, e.g. for frequency filtering or parasitic capacities. When the signal contains a longer direct current (DC) part (a lot of identical bits following each other) these capacitors can be charged or discharged[1]. The following bit transition can be delayed by the discharge or charge process of the capacitors.

To estimate a limit for the number of same following bits, a pattern of 64 bits with a certain number of zeros in a row, followed by the random number pattern described in 7.1.1 was sent via the setup shown in figure 7.1.

Figure 7.6a shows the dependence of the BER from the number of zeros in succession in the pattern. One can see, that after a certain number the BER increases quite fast. Figure 7.6b shows, that the maximum the number of zeros

[1] a stream of logic ones corresponds to a direct current and charges the capacitors

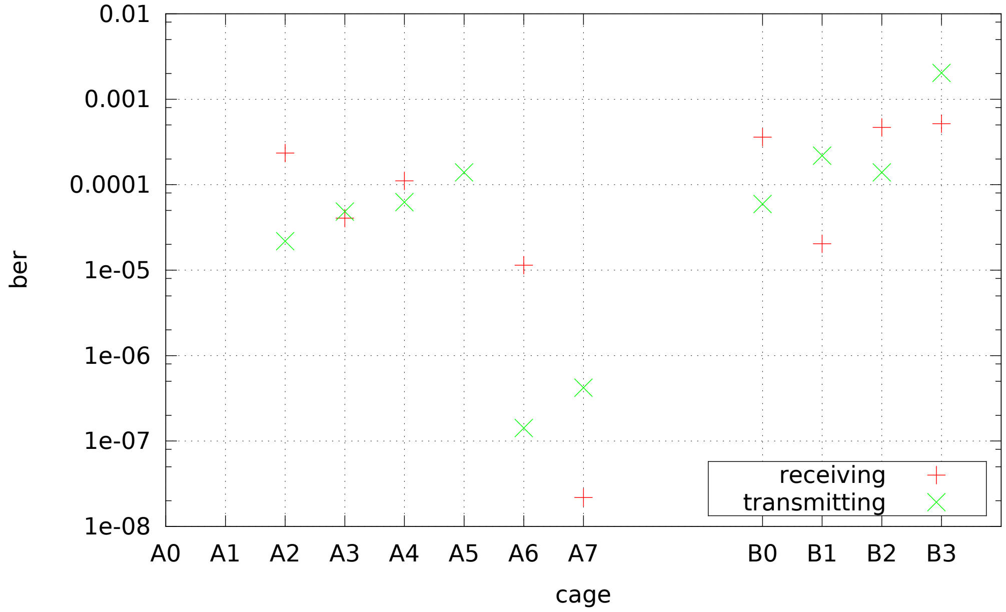
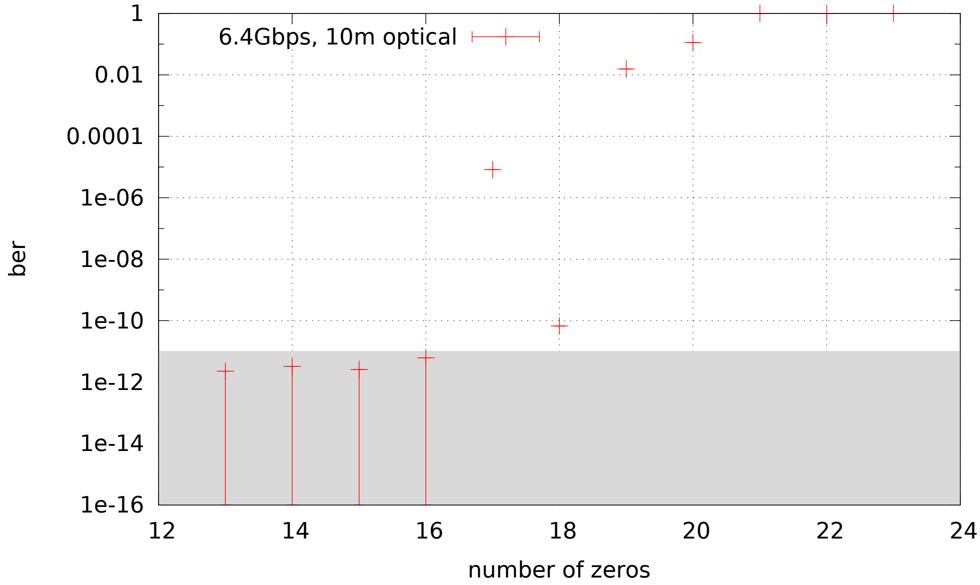
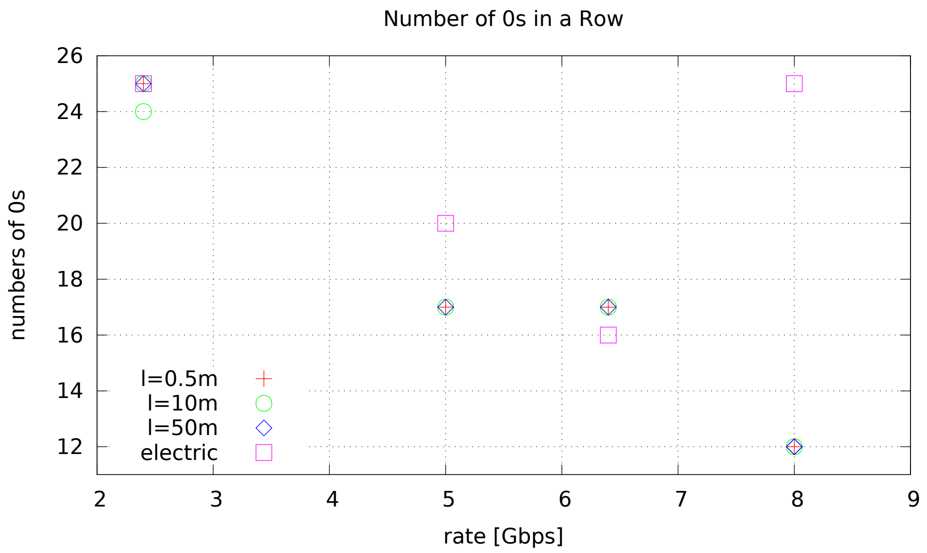


Figure 7.5: Bit error rate measured for different cages of the SantaLuz board.“Receiving” means that the indicated channel is receiving the data from the zeroth. “Transmitting” is the other way around. The ”A” indicates a channel connected via HSMC port A to the FPGA, and ”B” channels are connected via HSMC port B.



(a) Bit error rate for increasing number of zeros in the described pattern at 6.4 Gbit/s and using a 10 m optical fiber



(b) maximum number of zeros in the described pattern for which the BER $< 10^{-11}$ at different rates

Figure 7.6: Results of the measurement using data pattern with a certain number of logic zeros in a row.

in succession in a data pattern is limited. The limit depends on the data rate for a BER that should not be exceeded.

Latency measurement

In addition to bit error rate tests, a latency measurement was performed. The latency is described by the time the data signal needs to travel through the system. In this case it stands for the time gap between data generation and receiving/checking. A counter has been used to measure the time. In the transmitting part of the FPGA, this counter is running and sent via the optical transmission path to the receiving part of the same FPGA. At this point the FPGA takes the difference between the incoming signal and the generated pattern. The difference is the number of clock cycles of the parallel data part, i.e. the clock cycle which the counter uses. Knowing this clock frequency one can get the time the signal used to travel from generation to the verification in the FPGA again.

$$\mathrm{t_{lat}} = \mathrm{n_{cyc}} \cdot \mathrm{T} = \mathrm{n_{cyc}} \cdot \mathrm{f}^{-1} \tag{7.1}$$

Where $\mathrm{t_{lat}}$ is the latency, $\mathrm{n_{cyc}}$ is the discrete number of cycles, i.e. the difference between generated and received pattern, T is the period, and f is the frequency of the parallel clock. This shows that the measured latency is always a discrete value.

By measuring the latency for all available cable lengths from 0.5 m to 50 m, the signal propagation velocity in the optical fibers can be measured by fitting a linear function to the values:

$$\mathrm{t_{lat} = t_{lat_{opt}} + t_{lat_{el}} + t_{lat_{logic}} = \frac{l}{c_n} + t_{lat_{el}} + n_{logic} \cdot T} \tag{7.2}$$

With $\mathrm{t_{lat_{el}}}$ being the latency of the electrical path from the FPGA to the SFP optical transceiver. $\mathrm{t_{lat_{opt}}}$ is the latency of the optical fiber, defined by the length l and the speed of light in the fiber material [2] $\mathrm{c_n}$ and $\mathrm{n_{logic}}$ is the number of cycles the FPGA uses for the implemented logic.
The results are: $\mathrm{c_n = (2.02 \pm 0.02) {\cdot} 10^8 \frac{m}{s} = (0.67 \pm 0.01) {\cdot} c}$, $\mathrm{t_{lat_{el}} = (5.2 \pm 1.6)\,ns}$, and $\mathrm{n_{logic} = 3.8 \pm 0.1}$, $\mathrm{c = 3.0 \cdot 10^8 \frac{m}{s}}$ is the speed of light in vacuum.

8Bit/10Bit Encoding in PCS IP cores

As described in 5.3, the Standard PCS provides 8B/10B encoding and decoding, as described in 4.3.1. As the interface width between PMA and PCS is limited to 40 bits and the highest parallel frequency, shown by the TimeQuest Timing Analyzer in Quartus II software, that is possible for the used logic (Pattern generator, state machine, BERT) is limited to ≈ 150MHz, the serial data rate is limited to ≈ 6GHz using PCS features. A possible solution could be to do the 8B/10B encoding in parallel before routing it to the FPGA transceiver.

7.1.2 Multiple channel mode with Transceiver standard configuration

After elaborating the figures for single channel use, the SantaLuz card was fully used, which means all eight SFP cages are loaded with optical transceivers using them in parallel on one FPGA. The SantaLuz was connected to the HSMC port A and the 64 pit pattern containing an 8 bit random number was transmitted. All channels use the same pseudo random pattern with the same seed. Figure 7.7 shows the BER for cable lengths of 1 m and 50 m.
The measurements show that some channels perform better than others. The bit error rate increases up to about 0.01 for some channels. Several combinations are working quite well in this setup with BERs less then 10^{-12}. The two cable lengths behave quite similar. Additionally, a setup with two FPGA boards was used. A maximum of 10 SFP transceivers was available, and between 2 and 5 channels on both boards were used. The transmitting SFP cages stay the same and the receiving ones have been varied during the measurement. All cables have a length of 50 m. Figure 7.8 shows the results for 2 channels in parallel. The results using 3 to 5 cables in parallel between two FPGA boards are shown in table A.1 in the appendix. Again, it can be seen that some combinations work better than others. A dependence of BER for several channels or even combinations of channels can not be seen. But there seems

[2] neglecting the slightly longer path in the fiber

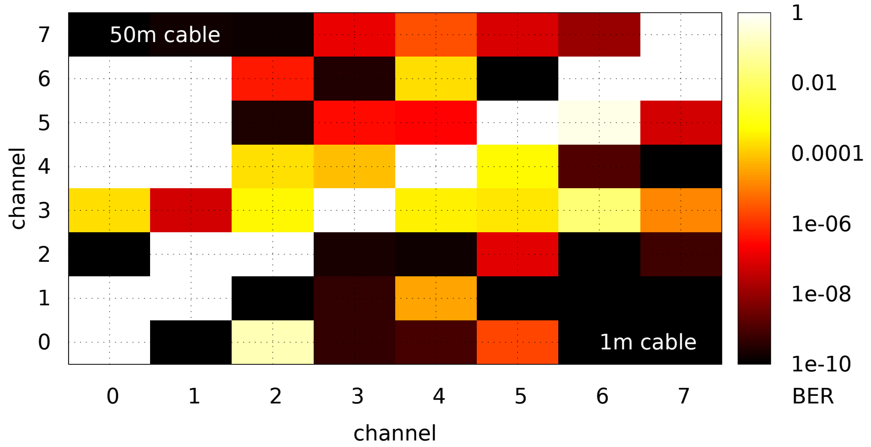


Figure 7.7: BERT results using all SFP cages in parallel. The diagram shows the BER for the transmission from the channel assigned on the x-axis to the channel on the y-axis. Combined plot for l=1 m l=50 m. In the upper left corner there are some values missing. The number of transmitted bits was about $3 \cdot 10^{11}$ for each measurement.

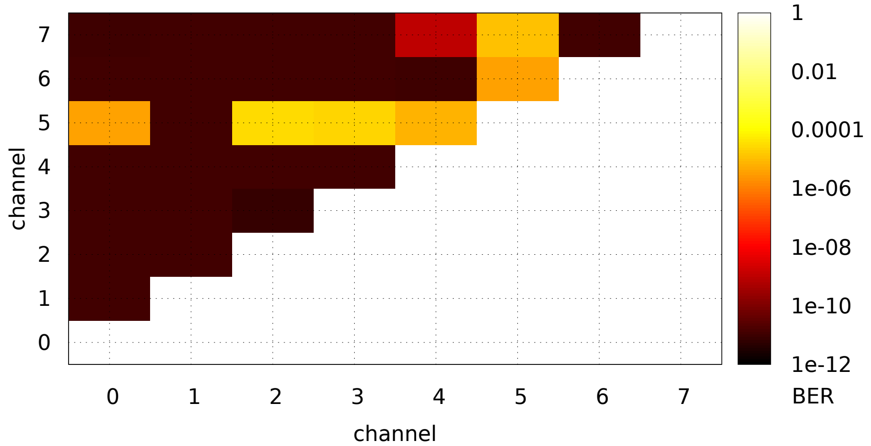
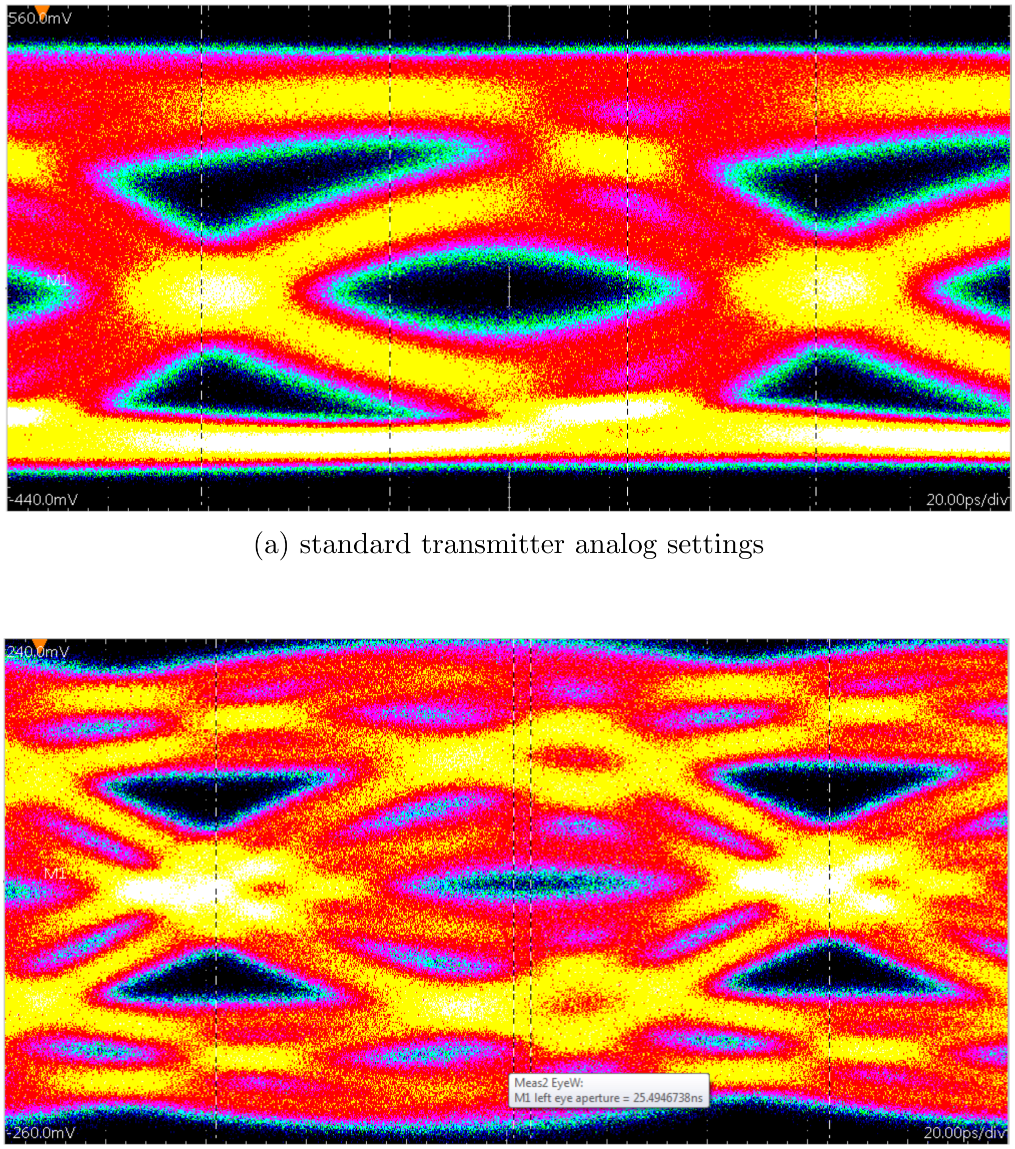


Figure 7.8: BERT plot for 2 cables used to connect two FPGAs. Transmitting channels are 0 and 7 on the first FPGA. Axis assign both receiving channels on the second FPGA. Number of transmitted bits $\approx 10^{11}$.

to be a correlation between the spacial distance of the cages on the SantaLuz board that are used and the occurrence of bit errors in the transmission.

7.1.3 Analog Tuning for Transceivers

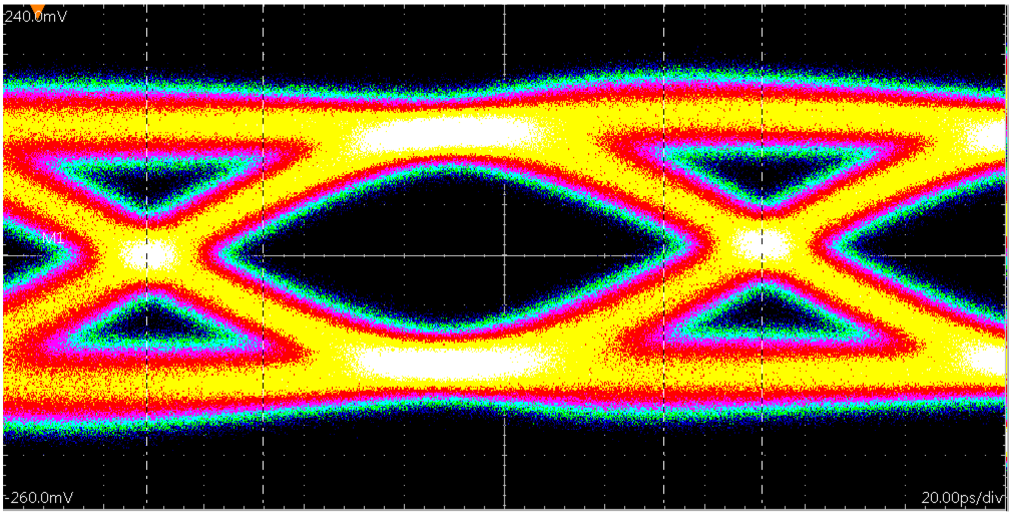
For using the whole capability of the system and to decrease error rates one can use analog tuning for the FPGA transceiver PMA described in section 5.3. The Quartus II Software provides a GUI for dynamic reconfiguration of analog settings via USB JTAG interface, the transceiver toolkit, described in 6.2. This toolkit has been used to obtain the range for optimal analog settings of the transceiver PMA for all channels. The influence of analog tuning settings can be seen in figure 7.9.



Meas2 EyeW:
M1 left eye aperture = 25.4946738ns

(a) standard transmitter analog settings

(b) $V_{od} = 25,\ 2^{nd}post = -10$



(c) $V_{od} = 26,\ 1^{st}post = 11,\ 2^{nd}post = 1,\ pre = -2$

Figure 7.9: Eye diagrams for different transmitter analog settings routed over the SantaLuz mezzanine card and an electrical SFP plug to a digital serial analyzer. Values below the picture show the changed values compared to the default settings: $V_{od} = 50,\ 1^{st}post = 0,\ 2^{nd}post = 0,\ pre = 0$

Tuning Procedure

First, the automatic analog tuning of the transceiver toolkit was used to get some start values for the analog tuning. Therefore, each parameter gets a start and end value and the toolkit goes through all combinations of these values and measures the BER or EyeQ values during a given time period. Mainly a time period of 1 second was used to check lots of parameter combinations in an acceptable total time. Using the BERT first, this measurement gave a rough estimate for the optimal settings. This can be tested again with longer measuring periods to get lower BER limits, or one can start doing a manual tuning.

For some settings, the BERT will not find an error in the relatively short time range for automatic tuning. So one can not estimate which setting fits best to the setup. For this reason, a manual tuning by observing the eye diagram given by the EyeQ tool has been done. After tuning all channels in parallel, channels that still produce bit errors in a longer time period were tuned individually to find even better settings for these channels. The resulting analog settings are shown in table 7.1.

Setup

For analog tuning measurements, a setup as described in section 7.1.2 was used. All eight cages of the SantaLuz board were loaded with SFP optical transceivers. Table 7.2 shows the connections between transmitter and receiver

found by / channel	V_{od}	1^{st} pre	2^{nd} pre	post	DC Gain	Lin. Eq.
toolkit BER						
0	18	4	0	-3	4	15
1	18	4	0	-3	0	1
2	18	4	0	-3	0	12
3	18	4	-2	0	0	0
4	18	4	0	-2	0	15
5	18	4	0	-2	4	15
6	18	4	0	-3	0	15
7	18	4	0	-3	4	15
toolkit EyeQ						
all	18	6	0	-2	0	5
manual tuning						
0	18	6	0	-3	-	5
1	18	6	0	-3	-	5
2	18	6	0	-3	-	5
3	18	6	0	-1	-	5
4	18	6	0	-2	-	5
5	18	6	0	-2	-	5
6	18	6	0	-3	-	5
7	18	6	0	-3	-	5

Table 7.1: Analog tuning values for 8 Gbit/s. The EyeQ values provide an eye opening of 8/38.

channels on the SantaLuz board. Based on the results from measuring without analog tuning, data rates of 6.4 Gbit/s and 8 Gbit/s have been chosen. For all following measurements, a parallel data pattern of 80 bit was used. The pseudo random number was generated out of 32 bit, which leads to a maximum of 31 zeros in a row.
Two coding schemes have been used for the BER measurements to ensure stable transmission between the two FPGAs or to figure out which advantage can be gained from it. These two, namely a running disparity controller and a scrambler, are described below.

Running disparity (RD) Following the advice of the SFP optical transceiver manual to use DC balanced data patterns, a running disparity circuit has been used for both bandwidth settings. Basically it compares the number of zeros and ones in each data pattern. It calculates the difference of these two numbers and sums up over all patterns. By inverting specific patterns it keeps the sum close to zero[3]. One bit of each pattern is used to indicate whether the pattern is inverted or not for the receiver part to decide inverting the pattern

[3] maximum distance to 0 is patternlength − 1

or not. The advantage of this option is the limitation of DC parts in the data transmission at the disadvantage of loosing one bit per pattern for pure data. Different lengths of the patterns that are monitored by that feature have been tested, namely 40 (RD(40)) which needs two coding bits and the complete 80 bit pattern (RD(80)) with one coding bit.

Scrambler A scrambler is an encoding scheme for data transmission. It can randomly decrease the appearance of data patterns difficult to transmit. In this case a so-called self synchronized scrambler is used. This type of scrambler uses the incoming serial data stream for encoding and decoding the data stream. In this case, a 79 bit buffer is used to store the last incoming bits.

$$\mathrm{x_{out}} = \mathrm{x_{in}} \oplus \mathrm{x_{70}} \oplus \mathrm{x_{78}} \tag{7.3}$$

Equation 7.3 shows the function to generate the transmitted bit $\mathrm{x_{out}}$ out of the data bit $\mathrm{x_{in}}$ using the $\mathrm{i^{th}}$ bit of the buffer $\mathrm{x_i}$ and the logical xor operation $\oplus$. For the next bit the buffer is shifted by one bit and the previous $\mathrm{x_{in}}$ is set as new $\mathrm{x_0}$. Since $\mathrm{a} \oplus \mathrm{a} = 0$ and $\mathrm{a} \oplus 0 = \mathrm{a}$ for $a \in \{0, 1\}$ the descrambler on the receiver side uses the same function and buffer.

ch	transmitter port	receiver port
0	1	0
1	0	1
2	3	2
3	2	3
4	5	4
5	4	5
6	7	6
7	6	7

Table 7.2: Connected channels of the SantaLuz SFP cages used for BERT with analog tuning

Loopback measurements

To compare the performance of the optical transmission path to the transceiver performance of the FPGA, loopback measurements have been done. There are two types of loopback paths available. The first one is an internal loopback circuit in the FPGA, where transmitter outputs are wired to the receiving ports directly on the FPGA. The second loopback possibility is a HSMC Debug Header Breakout Board included in the Stratix V Development kit which connects the output ports of the HSMC plug to the input port of the same channel. Both loopback cycles have been used with a data rate of 8 Gbit/s for all eight channels using running disparity control and analog tuning for the

internal one, the results are shown in table 7.3.
With these BER for both loopback paths it can be assumed, that higher BER in following measurements are originating from the usage of the non-FPGA transmission parts.

loopback	transmitted bits	BER limit (95 % C.L.)
internal	$1.3 \cdot 10^{17}$	$< 2.3 \cdot 10^{-17}$
external	$1.1 \cdot 10^{17}$	$< 2.6 \cdot 10^{-17}$

Table 7.3: BER for both loopback setups at 8 Gbit/s

BER for analog tuned 8 Gbit/s

ch	RD(80), no SC l=50 m	RD(40), no SC l=1 m	RD(80), SC l=50 m
0	$< 1.5 \cdot 10^{-13}$	$(7.24 \pm 5.12) \cdot 10^{-16}$	$< 1.4 \cdot 10^{-15}$
1	$< 1.5 \cdot 10^{-13}$	$(4.56 \pm 0.05) \cdot 10^{-13}$	$< 1.4 \cdot 10^{-15}$
2	$(1.03 \pm 0.07) \cdot 10^{-11}$	$(4.61 \pm 0.04) \cdot 10^{-12}$	$(1.83 \pm 0.01) \cdot 10^{-12}$
3	$(8.32 \pm 2.08) \cdot 10^{-13}$	$(4.00 \pm 0.12) \cdot 10^{-13}$	$(4.20 \pm 0.14) \cdot 10^{-13}$
4	$< 1.5 \cdot 10^{-13}$	$(1.45 \pm 0.02) \cdot 10^{-12}$	$< 1.4 \cdot 10^{-15}$
5	$< 1.5 \cdot 10^{-13}$	$(6.72 \pm 0.16) \cdot 10^{-13}$	$(2.93 \pm 1.12) \cdot 10^{-15}$
6	$< 1.5 \cdot 10^{-13}$	$(2.55 \pm 1.04) \cdot 10^{-15}$	$< 1.4 \cdot 10^{-15}$
7	$(3.95 \pm 0.45) \cdot 10^{-12}$	$(1.70 \pm 0.85) \cdot 10^{-15}$	$(6.08 \pm 0.06) \cdot 10^{-12}$
total	$(1.85 \pm 0.11) \cdot 10^{-12}$	$(4.56 \pm 0.05) \cdot 10^{-13}$	$(1.041 \pm 0.008) \cdot 10^{-12}$

Table 7.4: BER test with 31bit rn pattern at 8 Gbit/s with analog tuned transceivers, with and without scrambler (SC). Measurements without disparity controller (RD) are not possible because no synchronization between transmitter and receiver can be achieved. All upper limits are 95 % C.L. values.

Table 7.4 shows the results for BERT with analog tuned transceivers for all SFP ports of the SantaLuz board using SFP optical transceivers. Different channels have quite different BER values. Significant differences can not be seen between the usage of the different RD options and the usage of the scrambler (SC).

BER for analog tuned 6.4 Gbit/s

Table 7.5 shows the results for the 6.4 Gbit/s BERT with analog transceiver settings shown in table 7.1. The BER values for not using the running disparity controller are very different between the channels and are partially in a non-usable range for data transmission. This leads to a total BER summed for

ch	no RD l=1 m	RD80 l=1 m	RD40 l=1 m
0	$(6.98 \pm 0.01) \cdot 10^{-3}$	$< 7.8 \cdot 10^{-16}$	$< 7.8 \cdot 10^{-16}$
1	$(8.81 \pm 0.01) \cdot 10^{-3}$	$< 7.8 \cdot 10^{-16}$	$< 7.8 \cdot 10^{-16}$
2	$(7.79 \pm 0.03) \cdot 10^{-11}$	$< 7.8 \cdot 10^{-16}$	$< 7.8 \cdot 10^{-16}$
3	$(2.23 \pm 0.10) \cdot 10^{-2}$	$< 7.8 \cdot 10^{-16}$	$< 7.8 \cdot 10^{-16}$
4	$(9.57 \pm 0.01) \cdot 10^{-10}$	$< 7.8 \cdot 10^{-16}$	$< 7.8 \cdot 10^{-16}$
5	$(9.72 \pm 0.10) \cdot 10^{-12}$	$< 7.8 \cdot 10^{-16}$	$< 7.8 \cdot 10^{-16}$
6	$(3.81 \pm 0.01) \cdot 10^{-10}$	$< 7.8 \cdot 10^{-16}$	$< 7.8 \cdot 10^{-16}$
7	$(1.61 \pm 0.01) \cdot 10^{-10}$	$< 7.8 \cdot 10^{-16}$	$< 7.8 \cdot 10^{-16}$
total	$(4.85 \pm 0.01) \cdot 10^{-3}$	$< 9.8 \cdot 10^{-17}$	$< 9.8 \cdot 10^{-17}$

Table 7.5: BER test with 31bit rn pattern at 6.4 Gbit/s with transceiver analog settings like in the 8 Gbit/s measurements. All upper limits are 95 % C.L. values.

all channels of $(4.85 \pm 0.01) \cdot 10^{-3}$. With the usage of the running disparity control, the BER values decrease dramatically and in this order of magnitude there is no difference between the usage of RD(40) or RD(80). Both BERs are $< 10^{-16}$ (95 % C.L.).

7.2 Measurements with optical QSFP transceivers

The Stratix V Board provides a quad small form-factor (QSFP) cage directly mounted on the PCB. Two FPGA boards have been linked via a QSFP plug and cable assembly, described in section 5.6. These assemblies provide four full duplex channels. BER measurements have been taken at different data rates using an 80 bit parallel pattern with a 32 bit random number and a running disparity control, like in the SFP measurements described in 7.1.3. In addition, analog tuning has been carried out by the auto tune function of the transceiver toolkit to reach the highest possible data rates.

rate [Gbit/s]	cable length [m]	BER
9.3	3	$< 9.29 \cdot 10^{-17}$
9.3	20	$< 9.56 \cdot 10^{-17}$
11.3	20	$(3.29 \pm 1.04) \cdot 10^{-16}$

Table 7.6: Results for QSFP BER measurements at different data rates and different cable lengths. The given value summed BER over all four channels. All upper limits are 95 % C.L. values.

Table 7.6 shows the results for the BER measurements with QSFP optical transceivers. BERs for 9.3 Gbit/s data rate and both cable lengths (3 m, 20 m) and 11.3 Gbit/s at 20 m cable length are $(3.29 \pm 1.04) \cdot 10^{-16}$. The 11.3 Gbit/s BER are given all 4 channels, but for only one channel errors occurred for reasons unknown until now. Maybe the used fiber was not all right. The 7 other channels had BER $< 1.1 \cdot 10^{-16}$ (95 % C.L.).

7.3 Kapton Flex Print samples

The connections between the pixel sensors and the front end FPGAs will be implemented by Kapton flex prints, as described in 3.4.3. In the course of this thesis, 25μm Kapton type 100HN [24] foils coated with 50 nm or 25μm aluminum have been used to evaluate the production of wires.

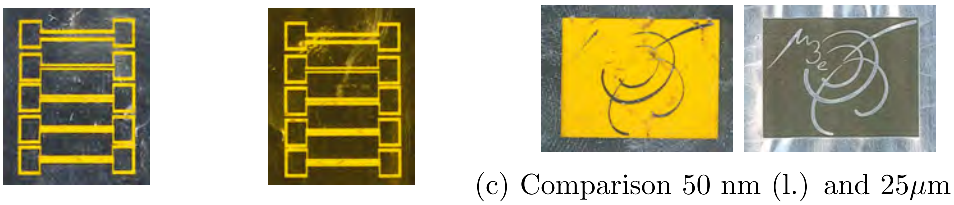
7.3.1 Laser Platform

The Kapton aluminum foil was processed by a PLS6MW laser platform from Universal Laser Systems to evaporate the aluminum or even cut the complete foil. It was equipped with a 50 W CO_2 laser at 9.3 μm wavelength and a 40 W fiber laser at 1.06 μm, both in a pulsed mode. In combination with the delivered software and drivers, the platform can be used like a printer. A given vector graphic file is executed. There are several settings that can be adjusted, e.g. laser movement speed, frequency, power, type of wave and the focus position in z-direction. These settings can be assigned to different colors of the graphic file. In addition, it can be chosen whether the Laser is doing a vector or raster movement. The material evaporated by the Laser is blown away by an integrated gas output at the laser head. In this case, air has been used.

7.3.2 Observations

First, the settings for the platform have been optimized to evaporate the aluminum without damaging the Kapton foil. Values to cut the complete foil and to evaporate the aluminum (area) for both foils were determined. These values are given in table 7.7. It has to be mentioned that the z position has changed from time to time without an apparent reason. In this case the settings did not work and the focus in z direction had to be found again. The foil with thinner aluminum layer can be processed with the Kapton side on top. The Kapton seems to be almost transparent for both of the used wavelengths. The thicker foil has to be processed backside up. Using the same orientation, the aluminum is not evaporated completely and the evaporated aluminum burns through the Kapton. In this case, the Kapton is destroyed and no electrical separation can be achieved. Figure 7.10a shows the 50 nm aluminum side and figure 7.10b the Kapton side of the same structure.
The laser platform was used to produce several flex print samples, shown in figures 7.10 and 7.11. Figure 7.10a shows the 50 nm aluminum side and fig-

ure 7.10b the Kapton side of the same structure. Figure 7.10c compares the same pattern for the thinner and thicker foil. On the right side of figure 7.10c, burned glue between the Kapton and aluminum layers can be seen.

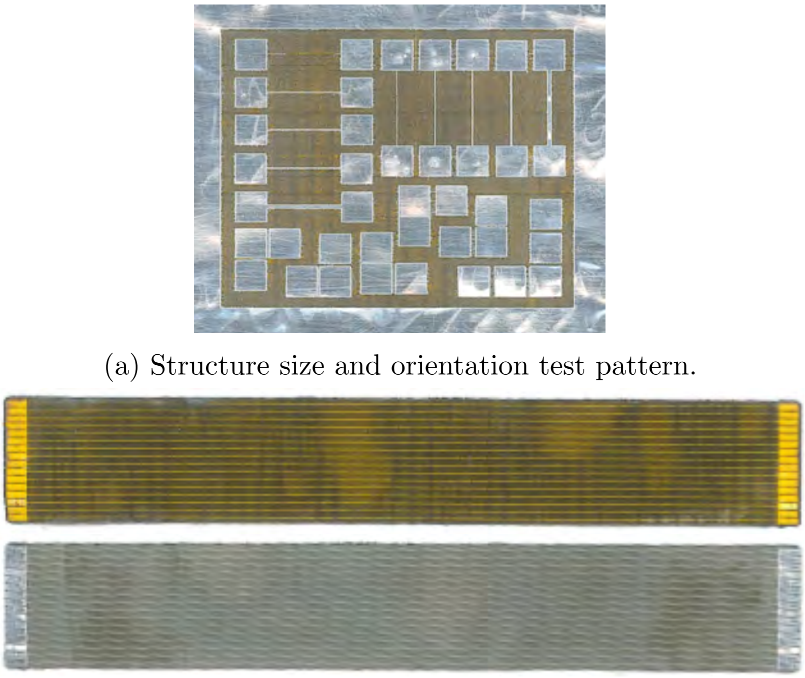


(a) Aluminum side.

(b) Kapton side.

(c) Comparison 50 nm (l.) and 25μm (r.) Al.

Figure 7.10: 25μm Kapton foil with 50nm/25μm aluminum coating. Original size.



(a) Structure size and orientation test pattern.

(b) Kapton cable structures test pattern from both sides.

Figure 7.11: 25μm Kapton foil with 25μm aluminum coating. Different test pattern. Original size.

The structure shown in figure 7.11a was produced to test the minimal distance between patterns to electrically separate them and the minimal width of a aluminum trace to be conductive. Table 7.8 shows the minimal feature sizes for the whole pattern screened horizontally. The same pattern is performed parallel and perpendicular to the "printer head" motion. The values are not the actual values on the pattern, but are the values that where given in the graphic file.

type	power [%]	speed [%]	freq. [MHz]	z [mm]	wave	contrast [%]	definition [%]	density [%]
area, $7\mu m$	100	45	76	1.5	0	20	10	80
cut, $7\mu m$	100	7	30	1.5	-	-	-	-
area, $25\mu m$	95	40	30	1.5	0	20	10	80
area*, $25\mu m$	100	30	30	1.5	0	20	10	80
cut, $25\mu m$	100	7	30	1.5	0	-	-	-

Table 7.7: Setting for laser platform PLS6MW to produce small structures in aluminum coating on Kapton. *These values are better suited for small values, e.g. between two pads in figure 7.11b.

size [μm]	separation	connection
parallel	50	100
perpendicular	400*	50

Table 7.8: Minimal sizes for separation and connection of $25\mu m$ Al components. *Settings for normal area instead of the for dedicated small ones from table 7.7 are used.

8 Conclusion and Outlook

In the following, the results of the measurements with both plug types, SFP and QSFP transceivers, and the observations for the Kapton flex-print samples are summarized.

8.1 SFP transceiver measurements

Summarizing the measurement results, one can say that the high frequency data transmission chain is always limited by the least performant part. It seems that the SantaLuz Mezzanine card is the bottleneck for the setups using its SFP cages. For using only one channel on this board the previously measured 6.25 Gbit/s bandwidth has been confirmed with a BER $< 2.9 \cdot 10^{-16}$ (CL. 95%) for 6.4 Gbit/s and 50 m optical fiber length for standard analog transceiver settings of the FPGA. It has to be mentioned that there was no obvious dependence of the BER on the length of the cable used.
By using several SFP cages on the same SantaLuz board or increasing the bandwidth to 8 Gbit/s without adapting the analog settings, the BER increases to over 10^{-12}. Besides the observation that some channels having lower BER at 8 Gbit/s than others, we suppose the SantaLuz board has cross talk between the traces on the board, which is why the BER increases for all-channel use.
In addition, other measurements besides the random number data BERT have been performed. For high data rates one also has to review the possible signal patterns. By measuring the BER as a function of the number of subsequent zeros in the data pattern it has been shown that for a fixed BER limit these number decreases with increasing data rate. In addition, the clock data recovery of the receiver seems to have problems with these type of patterns. To ensure that long lasting same digit patterns do not produce bit errors, it is possible to make sure they are not used in the data set or to use coding schemes, e.g. 8B/10B encoding.
By using analog tuning for the FPGA transceivers and a running disparity control we could greatly decrease the BER for all channel use of the SantaLuz board at 6.4 Gbit/s and 8 Gbit/s. The best value we achieved for 8 Gbit/s was $(1.041 \pm 0.008) \cdot 10^{-12}$ by using a scrambler, which has a small positive effect on the BER. This may also be true at 6.4 Gbit/s data rate, but seems to be unnecessary due to BER $< 10^{-16}$ achieved using the running disparity controller. These results show that the setup fulfills the required data rate of 5 to 6.25 Gbit/s per channel for the front-end links of the Mu3e experiment [5].

8.2 QSFP transceiver measurements

Instead of using the Mezzanine card with SFP plugs, BER measurements with QSFP optical transceivers, by using the QSFP cages directly on the FPGA board have been performed. This setup provides four full duplex channels where the data rate could be increased to 11.3 Gbit/s per channel with a BER $(3.29 \pm 1.04) \cdot 10^{-16}$ with one channel producing errors. At 9.3 Gbit/s BER decreases to $< 9.56 \cdot 10^{-17}$. So the QSFP connections fulfill the bandwidth requirements for the Mu3e readout between the readout FPGAs and the Filter Farm PCs, which are proposed to have a data rate of 8.5 to 10 Gbit/s per channel [5]. One problem is the fixed connection of all four channels to the QSFP plugs. Each Mu3e readout FPGA will be connected to 12 filter farm PCs. So it has to be figured out if there are other solutions to connect the plugs and fibers, for example MTP/MPO 12-way ribbon connectors [25].

8.3 Outlook for optical readout

Tasks for future projects could be the following:

- Find a smaller form factor for the plugs and connectors to fit into the small detector interior.
- Develop higher level data protocols for encoding data, synchronizing readout parts and controlling detector and readout parameters.
- Generate FPGA firmware to sort and switch the data from the detector to the filter farm.
- Add a monitoring system for the detector and the readout into the DAQ.

8.4 Kapton flex prints

In the course of this thesis, a first attempt for producing Kapton LVDS flex prints has been made. The aluminum coated Kapton foil seems to be processable with the available laser platform. In the future, the possible structure sizes have to be evaluated to be sufficient for usage as LVDS links.
The following tasks are still open and could be checked in a prospective project:

- For the cable structure, shown in figure 7.11b, parameters, e.g. the resistance and the size of the traces, has to be measured. For the first results refer to [26].
- Since structures suitable for LVDS cables in terms of impedance and resistance at useful lengths are producible, the assembling of the wires with plugs has to be characterized.

- Kapton films with a layer thickness of 12 μm are available. As this aluminum layer thickness lies between the thickness of the used foils, an application seems to be interesting and has to be evaluated.

- A suitable solvent to remove all residues of the 25 μm Al layers treatment has to be found.

- Furthermore, Kapton coated on both sides with an aluminum layer is available. A processing of this material can be evaluated.

Acknowledgments

First of all, I would like to thank Dr. Niklaus Berger for giving me the opportunity to work out this thesis in his Research Group and supporting me whenever it was needed.
I really like to thank Simon Corrodi for collaborating in doing the measurements, discussing a lot of issues and cross reading this thesis. I would also like to thank Dr. Dirk Wiedner for the support and proofreading during the whole process, and the whole Mu3e group for providing a pleasant working environment.
Professor Hans-Christian Schultz-Coulon kindly agreed to serve as the second examiner for this thesis, for which I would like to thank him.
A very special thanks go to my parents who supported me very well during my whole bachelor study.

A Appendix

sent from ch	0	7	5	3	2	
sent to ch	7	6	0	2	5	$(2.3 \pm 1.3)\,10^{-12}$
	7	6	0	2	4	$5.4 \cdot 10^{-13}$
	0	1	2	3	4	$(6.2 \pm 4.4)\,10^{-14}$
	0	1	2	3	4	$(4.9 \pm 1.7)\,10^{-15}$
	0	1	2	3	5	$(2.1 \pm 2.1)\,10^{-14}$
	7	6	2	0		$2.3 \cdot 10^{-12}$
	7	6	1	0		$5.3 \cdot 10^{-12}$
	7	6	3	0		$4.0 \cdot 10^{-12}$
	7	6	4	0		$2.9 \cdot 10^{-12}$
	7	6	5	0		$3.5 \cdot 10^{-12}$
	0	1	6			$2.3 \cdot 10^{-12}$
	0	1	7			$9.0 \cdot 10^{-12}$
	0	1	5			$(1.8 \pm 0.6)\,10^{-11}$
	0	1	4			$1.0 \cdot 10^{-11}$
	0	1	3			$9.8 \cdot 10^{-12}$
	0	1	2			$8.1 \cdot 10^{-012}$
	0	2	5			$(2.0 \pm 2.0)\,10^{-12}$
	0	4	5			$(1.186 \pm 0.001)\,10^{-5}$
	0	2	3			$1.0 \cdot 10^{-11}$
	0	2	4			$1.0 \cdot 10^{-11}$
	0	3	4			$9.3 \cdot 10^{-12}$
	1	3	4			$1.0 \cdot 10^{-11}$
	2	3	4			$9.4 \cdot 10^{-12}$
	1	2	3			$8.6 \cdot 10^{-12}$
	6	1	3			$9.5 \cdot 10^{-12}$
	6	1	2			$7.1 \cdot 10^{-12}$
	6	2	3			$9.3 \cdot 10^{-12}$
	6	3	4			$7.2 \cdot 10^{-12}$
	6	4	5			$2.1 \cdot 10^{-10}$

Table A.1: Results for all cages measurement for two FPGA boards, descriped in 7.1.2.

Bibliography

[1] The ATLAS Collaboration, *"Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC"*, 2012, (`arXiv:1207.7214`).

[2] A. Blondel et al., *"Research Proposal for an Experiment to Search for the Decay $\mu \to eee$"*, ArXiv e-prints, January 2013, (`arXiv:1301.6113 [physics.ins-det]`).

[3] U. Bellgardt et al., [SINDRUM Collaboration], *"Search for the Decay $\mu^+ \to e^+e^+e^-$"*, Nucl.Phys., **B299** 1, 1988.

[4] M. Hirsch, W. Porod, L. Reichert and F. Staub, *"Phenomenology of the minimal supersymmetric $U(1)_{B-L} \times U(1)_R$ extension of the standard model"*, Phys.Rev., **D86** 093018, 2012, (`arXiv:1206.3516 [hep-ph]`).

[5] S. Bachmann et al., *"The proposed trigger-less TBit/s readout for the Mu3e experiment"*, January 2014.

[6] H. Augustin, *Charakterisierung von HV-MAPS*, Bachelor thesis, Heidelberg University, 2012.

[7] I. Perić et al., *"High-voltage pixel detectors in commercial CMOS technologies for ATLAS, CLIC and Mu3e experiments"*, Nucl.Instrum.Meth., **A731** 131–136, 2013.

[8] J. Troska, Technical Specification VERSATILE TRANSCEIVER (a.k.a CERN SFP+, VTRx, VTTx), CMS Document 4802-v2, 2012.

[9] M. Goresky and A. M. Klapper, *"Fibonacci and Galois representations of feedback-with-carry shift registers"*, IEEE Transactions on Information Theory, **48**(11) 2826–2836, 2002.

[10] P. A. Franaszek A. X. Widmer, *"A DC-Balanced, Partitioned-Block, 8B/10B Transmission Code"*, IBM Journal of Research and Development, **27**(5) 440, 1983.

[11] R.C. Walker, B. Amrutur and R.W. Dugan, 64b/66b decoding, for packetized serial data, U.S. Patent EP1133123, 2001.

[12] Cortina Systems and Cisco Systems, *"Interlaken Protocol Definition"*, October 7, 2008.

[13] V. Blobel and E. Lohrmann, *Statistische und numerische Methoden der Datenanalyse.* May 2012.

[14] W. Wolf, *FPGA-Based System Design -*. Pearson Education, Amsterdam, 1st edition, 2004.

[15] National Instruments, *FPGA Fundamentals*, 2012 Mai.

[16] IEEE Computer Society, *"IEEE Standard Test Access Port and Boundary-Scan Architecture"*, June 2001.

[17] Altera Corporation, *DSP Devlopment Kit, Stratix V Edition FPGA Development Board Reference Manual*, 12.0 edition, July 2012.

[18] Altera Corporation, *Stratix V Device Handbook Volume 2: Transceivers*, 2013.05.06 edition, May 2013.

[19] IEEE Computer Society, *"IEEE Standard for Ethernet"*, December 2012.

[20] S. Swientek, *A FPGA based Fast Readout for the Off-Detector Electronics of the LHCb Outer Tracker*, PhD thesis, Technical University Dortmund, 2014, in preparation.

[21] Helmholtz Alliance Ha-101, *"Annual Report 2011"*, June 2012.

[22] Avago Tehnologies, *AFBR-57D7APZ, Digital Diagnostic SFP, 850 nm, 8.5/4.25/2.125 GBd Low Voltage (3.3 V) Fibre Channel RoHS Compliant Optical Transciever, Data Sheet*, January 2013.

[23] Molex, QSFP+ Interconnect Solution, `http://www.molex.com/molex/products/family?key=quad_small_formfactor_pluggable_plus_qsfp_interconnect_solution&channel=products&chanName=family&pageTitle=Introduction`.

[24] DuPont, DuPont^{TM} Kapton$^{®}$ Polyimide Film, Motor and Magnet Wire Industry Bulletin, `http://www2.dupont.com/Kapton/en_US/assets/downloads/pdf/H29423_3_Kapton_motor.pdf`.

[25] Cisco, Cisco 40GBASE QSFP Modules Data Sheet, `http://www.cisco.com/en/US/prod/collateral/modules/ps5455/data_sheet_c78-660083_ps11541_Products_Data_Sheet.html`.

[26] S. Corrodi, *Fast readout for the pixel detector of the Mu3e experiment*, Master's thesis, ETH Zürich, 2014, in preparation.

Erklärung

Ich versichere, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Heidelberg, 12.02.2014