



FACULTY OF MATHEMATICS AND NATURAL SCIENCES
DEPARTMENT OF PHYSICS
UNIVERSITY OF WUPPERTAL

Design and development of the IBL-BOC firmware for the ATLAS Pixel IBL optical datalink system

Diploma thesis
of
TIMON HEIM

CERN-THESIS-2011-383
28/09/2011



September 28, 2011
WU D 11-08

Abstract

The Insertable b -Layer (IBL) is the first upgrade of the ATLAS Pixel detector at the LHC. It will be installed in the Pixel detector in 2013. The IBL will use a new sensor and readout technology, therefore the readout components of the current Pixel detector are redesigned for the readout of the IBL. In this diploma thesis the design and development of the firmware for the new IBL Back-of-Crate card (IBL-BOC) are described. The IBL-BOC is located on the off-detector side of the readout and performs the optical-electrical conversion and vice versa for the optical connection to and from the detector.

To process the data transmitted to and received from the detector, the IBL-BOC uses multiple Field Programmable Gate Arrays (FPGA). The transmitted signal is a 40 Mb/s BiPhase Mark (BPM) encoded data stream, providing the timing, trigger and control to the detector. The received signal is a 160 Mb/s 8b10b encoded data stream, containing data from the detector. The IBL-BOC encodes and decodes these data streams.

The developed firmware includes the logical blocks required for the transmission of a BPM encoded data stream to the detector, reception and decoding of the 8b10b data stream from the detector, as well as communication to and control of these logical blocks. The firmware is tested on a commercial FPGA evaluation platform and is able to operate in a standalone manner controlled by a PC. The basic functionalities of the IBL-BOC firmware are demonstrated and tested with this setup. An exemplary readout of the new IBL detector hardware further proves the versatile design of the firmware.

Contents

1	Introduction	1
2	Particle Physics	3
2.1	The Standard Model	3
2.1.1	Fermions	3
2.1.2	Bosons	6
2.2	Higgs Mechanism	8
2.3	High Energy Particle Physics	10
3	The Large Hadron Collider	13
3.1	The Accelerator Complex	13
3.2	LHC Design Parameters	15
3.3	Experiments at the LHC	16
3.3.1	CMS	16
3.3.2	ALICE	17
3.3.3	LHCb	17
4	The ATLAS Experiment	19
4.1	Purpose of the ATLAS Experiment	20
4.2	Structure of the ATLAS Detector	20
4.2.1	The Inner Detector	20
4.2.2	The Calorimeter Systems	22
4.2.3	The Muon Spectrometer	23
4.2.4	The Magnet System	24
4.2.5	Particle Identification in the ATLAS Detector	24
4.3	The ATLAS Trigger System	25
4.3.1	Operation of the ATLAS Detector	26
5	The Pixel Detector	29
5.1	The Pixel Module	30
5.1.1	The Pixel Sensor	30
5.1.2	The Pixel Front-End Electronics	32
5.1.3	Module Control Chip	33
5.2	The Pixel Detector Data Acquisition	33
5.2.1	Optoboard	34

5.2.2	Back-of-Crate Card	35
5.2.3	Read-out-Driver	38
5.3	The Pixel Detector Services	38
6	The Insertable b-Layer	41
6.1	The IBL Design and Requirements	42
6.2	The IBL Module	45
6.3	The IBL Data Acquisition	49
7	Design of the IBL Back-of-Crate Card	51
7.1	Overview	51
7.1.1	Requirements	51
7.1.2	Design Goals	52
7.1.3	Hardware Prototype	53
7.2	Firmware Evaluation Platform	56
7.2.1	Spartan 6 Evaluation Board	56
7.2.2	Mezzanine Daughter Boards	57
7.3	Logical Blocks Overview	57
7.4	Communication Bus	59
7.4.1	Simulation of the Internal System Bus	62
7.4.2	Simulation of the Setup Bus	63
7.4.3	Communicating via the Setup Bus	64
7.4.4	Communication to a PC	64
7.5	TX path	66
7.5.1	BPM Encoding	67
7.5.2	Variable Delay	69
7.5.3	Alternative Concept	72
7.6	RX Path	73
7.6.1	ISERDES and Phase Alignment	74
7.6.2	Word Alignment and 8b10b Decoder	75
7.6.3	BOC to ROD Mux	76
7.6.4	Measurement of the Data Reception	77
7.6.5	ROD FIFO	78
7.6.6	Monitoring	78
7.7	Test System	80
7.7.1	Software	80
7.7.2	Register Scan	81
7.7.3	Loopback Scan	82
7.8	Combined Test with FE-I4	86
7.8.1	Proof of Principle	88
7.8.2	Digital scan	88
7.8.3	Outlook	90

8	Conclusion	93
8.1	Firmware Development and Outlook	93
8.2	First FE-I4 Tests	94
8.3	Future DAQ Concepts	94
A	Appendix	95
A.1	MEBOC Address Map	95
A.2	FE-I4 Standard Configuration	99
A.3	FE-I4 USBpix Digital Scan	102

Chapter 1

Introduction

Each piece, or part, of the whole nature is always an approximation to the complete truth, or the complete truth so far as we know it. In fact, everything we know is only some kind of approximation, because we know that we do not know all the laws as yet. Therefore, things must be learned only to be unlearned again or, more likely, to be corrected ... The test of all knowledge is experiment. Experiment is the sole judge of scientific “truth”. - Richard P. Feynman [Fey05]

Gaining knowledge about the innermost structure of the matter all around us, describing them in mathematical formulas and prove this knowledge and formalism in experimental setups, is the goal of high energy particle physics. Currently the best approach to do so, is the Standard Model of particle physics, combining the known elementary matter and interaction particles. The Standard Model is not a complete theory, there are many open questions and parameters. To answer some of these questions and to prove or disprove theoretical models, the biggest particle accelerator, the Large Hadron Collider (LHC), was built. In this machine protons collide with energies of $\sqrt{s} = 14 \text{ TeV}$ ¹, which were never achieved till now. The experiments at the LHC seek to find the particles predicted by theoretical models or even unpredicted “new” particles. The ATLAS detector is one of the four large experiments at the LHC and its innermost part the Pixel detector is one of six sub-detectors of the ATLAS detector. Due to higher luminosities which will be achieved by the LHC in later operation phases, the Pixel detector will be upgraded in the year 2013. The first step in this upgrade is the Insertable *b*-Layer (IBL), which will be the fourth innermost layer in the Pixel barrel detector. For this part of the detector the requirements on some parts of the readout alter, so it is necessary to change the current readout design to fit the new requirements. One part of the Pixel detector optical datalink is the optical-electrical converter, the Back-of-Crate card (BOC), located on the off-detector side of the readout chain.

In this diploma thesis the firmware of the new BOC, the IBL-BOC, is designed, developed and tested, with the purpose of being compatible to the current Pixel detector readout system, as well as meeting the requirements for the IBL.

¹ $\sqrt{s} = 7 \text{ TeV}$ in the first phase of the LHC operation.

Therefore the IBL-BOC heavily takes advantage of the newest FPGA technology. The logical blocks placed in these FPGAs, used for the transmission and reception of the data streams, are based on a preliminary study [Neu11]. To control and test these blocks, a communication path is developed and tested. Based on this design, all logical blocks are combined in a standalone test setup, which is used to investigate and advance the behavior of the logical blocks of the firmware for the IBL-BOC.

Chapter 2

Particle Physics

The present conception of the world of elementary particles is combined in a theory called the Standard Model of particle physics. This theory is not yet a complete theory, it still leads to open questions and parameters, which are currently supplemented by experimental measurements. To answer these questions many different theories exist, either expanding the Standard Model or even taking a different approach. The not yet discovered Higgs boson is e.g. giving the answer to the question, why some particles have mass.

2.1 The Standard Model

In the Standard Model (SM) [Pov09] all known elementary particles and their interactions, except for gravity, are described and parameterized. The SM is a combination of quantum chromodynamics and the unified theory of electroweak interaction. Particles with half-integral spin are known as fermions, which are all the matter particles and particles with integral spin are known as bosons, which are the interaction particles.

2.1.1 Fermions

The group of fermions is split into two families, quarks and leptons (see Table 2.1 and 2.2), each consisting of a total of 12 particles. For every fermion exists an anti-fermion, carrying the opposite quantum numbers, but possessing the same mass. A common characteristic of fermions is the decay or transformation into fermions of lower mass, while obeying certain rules. Quarks interact through all SM forces, strong, weak and electromagnetic, leptons on the other hand do not interact through the strong force.

Quarks and leptons are grouped in three generations, sorted by their rising mass and a particle doublet per generation, which share physical behavior. Only the particles of the first generation build up the stable matter, the second and third generation particles have a short lifetime and decay in their final state to particles of the first generation. Also a particle of the second and third generation is only

Table 2.1: The three quark generations of the SM [eaPDG10].

Charge	1. Generation	2. Generation	3. Generation
$+\frac{2}{3}e$	up (u) 2.4 MeV/ c^2	charm (c) 1.27 GeV/ c^2	top (t) 173.1 GeV/ c^2
$-\frac{1}{3}e$	down (d) 4.8 MeV/ c^2	strange (s) 104 MeV/ c^2	bottom (b) 4.2 GeV/ c^2

Table 2.2: The three lepton generations of the SM [eaPDG10].

Charge	1. Generation	2. Generation	3. Generation
$-1e$	e 0.511 MeV/ c^2	μ 105.7 MeV/ c^2	τ 1.77 GeV/ c^2
0	ν_e < 2.2 eV/ c^2	ν_μ < 0.17 MeV/ c^2	ν_τ < 15.5 MeV/ c^2

produced in high energy collisions, e.g. experimental setups like the LHC or the cosmic radiation interacting with the atmosphere of the earth.

Quarks

The group of quarks are elementary particles, which build composite particles e.g. the nuclei of atoms. The six types of quarks, up, down, charm, strange, top and bottom (u , d , c , s , t and b) are also called flavors and have an electric charge of $-\frac{1}{3}e$ and $\frac{2}{3}e$ respectively. Quarks possess the property of color charge (red, blue, green), which allows them to interact with each other through the strong force. Due to a phenomenon called confinement, which only allows neutral (white) color states, quarks can not be observed isolated, however they group up in composite particles called hadrons. A hadron made of a quark anti-quark pair is called meson and one made of three quarks is called baryon, these are the only observed white particles¹. These hadrons contain valence and sea quarks. Valence quarks contribute to the quantum numbers and charge of the hadron, while sea quarks are virtual quark anti-quark pairs only visible through scattering and the fraction of momentum they carry.

If e.g. a collider experiment produces an isolated quark, due to confinement this quark will produce quark anti-quark pair from vacuum and build a hadron, this process is called hadronization. If the quark has enough energy, it will produce many hadrons forming a particle jet. These jets form a cone-like shape in a particle detector and need special attention in the analysis of the data, because a jet can originate from one or more quarks. For this reason, properties of quarks can only be investigated by observation of hadrons.

Quarks can transform into other quarks of lower mass (flavor change) through a charged current, where a transformation from one generation to another is highly suppressed compared to one inside a generation. A particle which can transform via

¹In theory a composite particle build of four quarks is possible, called tetra-quark.

a charged current into an u -quark, is called a down-type quark and can be seen as the eigenstate $|d'\rangle$ of the superposition of the flavor eigenstates:

$$|d'\rangle = V_{ud}|d\rangle + V_{us}|s\rangle + V_{ub}|b\rangle \quad (2.1.1)$$

In analogous way the strange-type and bottom-type quarks are described by a superposition of flavor eigenstates.

The result of this behavior is described by a mathematical model called the Cabbibo-Kobayashi-Maskawa (CKM) quark mixing matrix V_{CKM} , which gives rise to the tendency for every quark transforming into another quark through a charged current interaction [eaPDG10]:

$$|V_{ij}| = \begin{pmatrix} |V_{ud}| & |V_{us}| & |V_{ub}| \\ |V_{cd}| & |V_{cs}| & |V_{cb}| \\ |V_{td}| & |V_{ts}| & |V_{tb}| \end{pmatrix} = \begin{pmatrix} 0.974 & 0.225 & 0.003 \\ 0.225 & 0.973 & 0.041 \\ 0.009 & 0.040 & 0.999 \end{pmatrix} \quad (2.1.2)$$

Showing the magnitude $|V_{ij}|$ of the CKM elements and hence the tendency for a quark i to transform into a quark j . A flavor changing neutral current, e.g. $c \rightarrow u$, has not been observed yet and is also suppressed in the SM.

Leptons

The group of leptons are elementary particles, which can be found, in contrast to quarks, as isolated particles, because they do not possess a color charge and are not subject of confinement. Hence they do not interact via the strong force. The three generations consist of one charged lepton: electron, muon and tau (e , μ and τ), and their neutral counterpart lepton, each called neutrino: electron-neutrino, muon-neutrino and tau-neutrino (ν_e , ν_μ and ν_τ). The lifetime of a τ and μ lepton is in the order of 10^{-12} s and 10^{-6} s, respectively. Both decay leptonic into a lepton of lower mass:

$$\begin{aligned} \tau^- &\rightarrow \mu^- + \bar{\nu}_\mu + \nu_\tau \\ \tau^- &\rightarrow e^- + \bar{\nu}_e + \nu_\tau \\ \mu^- &\rightarrow e^- + \bar{\nu}_e + \nu_\mu \end{aligned} \quad (2.1.3)$$

While the τ can also decay hadronic into a π meson:

$$\tau^- \rightarrow \pi^- + \bar{\nu}_\tau \quad (2.1.4)$$

These decays follow a certain law of conservation, called conservation of lepton number, the number of leptons must be conserved for every lepton generation in article interactions or decays. Electrons on the other hand are stable particles and because they are encountered in every day life, were the first observed elementary particles.

Neutrinos show a different behavior, alongside interacting with virtually nothing, because they can only interact via weak force, they do not decay into other neutrinos,

but show characteristics in some extent similar to quark mixing called neutrino oscillation. In neutrino oscillation the flavor eigenstates $|\nu_\alpha\rangle$ ($\alpha = e, \mu, \tau$) are not the mass eigenstates $|\nu_i\rangle$ ($i = 1, 2, 3$), a flavor eigenstate can be expressed as a superposition of all mass eigenstates. In an analogous way to the CKM matrix, the Pontecorvo–Maki–Nakagawa–Sakata (PMNS) matrix U_{PMNS} describes this as follows:

$$\begin{pmatrix} |\nu_e\rangle \\ |\nu_\mu\rangle \\ |\nu_\tau\rangle \end{pmatrix} = \begin{pmatrix} U_{e1} & U_{e2} & U_{e3} \\ U_{\mu1} & U_{\mu2} & U_{\mu3} \\ U_{\tau1} & U_{\tau2} & U_{\tau3} \end{pmatrix} \begin{pmatrix} |\nu_1\rangle \\ |\nu_2\rangle \\ |\nu_3\rangle \end{pmatrix} \quad (2.1.5)$$

Neutrino oscillation solved the solar neutrino problem. The sun should produce an enormous amount of ν_e flux in its fusion process, as described by the Standard Solar Model (SSM), but the predicted ν_e flux was higher than the one measured on earth. If neutrino oscillation is taken into account, a certain amount of ν_e oscillate to ν_μ or ν_τ . Measurements show that this overall $\nu_{e,\mu,\tau}$ flux fits with the SSM. Neutrino oscillation requires the neutrinos to have a non-zero mass, which was not part of the former SM and needed to be corrected, yet experiments could only give an upper limit on the neutrino masses (see table 2.2).

2.1.2 Bosons

The SM describes three² fundamental forces: the electromagnetic, weak and strong force. The unification of electromagnetic and weak force into the electroweak force was a big success, as the SM is aimed to be a “unified theory of particle physics”, but yet all attempts to unify the electroweak and strong force in the Grand Unified Theory (GUT), were not successful.

The forces of the SM are mediated by five gauge bosons listed in Table 2.3. In every

Table 2.3: The five gauge bosons, mediating the three fundamental forces [eaPDG10].

Force	Gauge boson	Mass	Charge
Electromagnetic	Photon (γ)	0	0
Weak interaction	Z-Boson (Z)	91.2 GeV/ c^2	0
	W-Boson ($W^\pm$)	80.4 GeV/ c^2	$\pm 1e$
Strong interaction	Gluon (g)	0	0

interaction certain quantities are conserved, listed in Table 2.4. Energy, momentum and angular momentum are known to be conserved from classical mechanics. The CP symmetry is the composition of charge conjugation C and parity P , combined they describe the symmetry between matter and anti-matter. While being conserved in an electromagnetic and strong interaction, the weak interaction violates the CP symmetry, showing a discrepancy between matter and anti-matter in certain weak

²Gravitation is not included in the SM, due to its very small effect on the scale of elementary particles.

Table 2.4: Laws of conservation for the specific interactions.

Quantity	EM	Weak	Strong
Energy E	✓	✓	✓
Momentum $\vec{p}$	✓	✓	✓
Angular momentum $\vec{l}$	✓	✓	✓
CP	✓	×	✓
Baryon number B	✓	✓	✓
Lepton number L	✓	✓	✓

processes. The conservation of number of baryons and leptons is an empirical quantity, as no violation of this is observed yet, e.g. the proton lifetime is measured to be $\geq 10^{31}$ years.

Electromagnetic Force

The range of the electromagnetic force is unlimited, because its gauge boson the photon is massless. It is well described in the quantum electro dynamics (QED), which is the unification of quantum field theory and electro dynamics, e.g. QED describes the energy levels of an electron inside a hydrogen atom very precisely. An example for an electromagnetic interaction is given in Figure 2.1, a photon producing a e^+e^- pair. The interaction is visualized in Feynman Graph [Fey49].

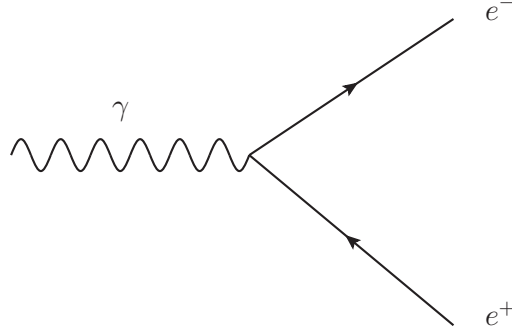


Figure 2.1: Example for an electromagnetic interaction, the e^+e^- pair production.

This is possible when the photon exceeds the energy of two electrons masses (at rest).

Weak Force

In nature the weak force is primarily observed in β -decays of unstable atoms (shown in Figure 2.2) In this process one of the d -quarks of the neutron transforms into an u -quark via a virtual W^- boson, producing an e^- and $\bar{\nu}_e$. After the observation of this process, the existence of the neutrino was postulated. The observed momentum of the electron was continuous, thus another particle has to participate in the

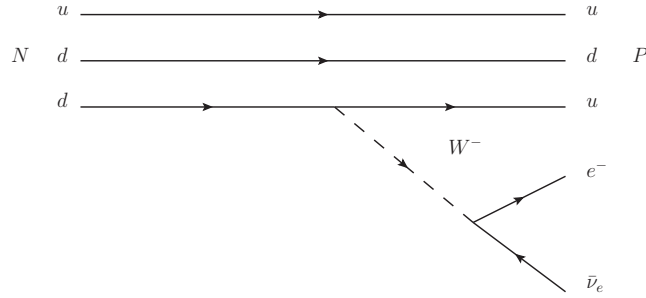


Figure 2.2: Transformation of a neutron into a proton via a charged current flavor change, the process responsible for the β -decay of atoms.

decay, which shares momentum with the electron, being very light and neutral. The three gauge bosons of the weak force are very heavy, hence the range of the weak interaction is limited.

The weak force is the only force known to slightly violate the CP -symmetry, favoring matter over antimatter. This has been observed in Kaon and certain B -hadron decays, processes mediated by the weak force.

Strong Force

The strong force is responsible for holding the quarks inside the hadrons together. It is described in quantum chromo dynamics (QCD) and mediated through gluons. Gluons couple to the color charge of quarks, but as they also carry a color charge, they can actively participate in strong force interactions. Gluons always carry two color charges, one color and one anti-color, these colors superimpose and build a total of 8 different gluons.

The strong force has a different behavior than other forces, it gets stronger if two quarks or gluons veer away from each other (confinement). This is the reason for the hadronization process every time an isolated quark is “produced”. Tearing it apart from its hadron, the energy increases with distance and at some point it is energetically beneficial that the isolated quark produces a quark-anti-quark pair lowering the distance between each quark.

2.2 Higgs Mechanism

The SM lacks consistency at a crucial point. The behavior of particles and interactions is in good accordance with the experimentalists observations, but the attribute of particle mass is not explained. This is solved by the Higgs mechanism [Hig64], giving the particles mass. Yet the Higgs boson was not experimentally observed.

The Higgs mechanism introduces a Higgs field which is distorted by particles passing through it, hence giving the particle a mass. The crucial point of the Higgs mechanism, is a spontaneous symmetry breaking, allowing gauge bosons to acquire mass.

The Higgs boson is the quantum of the Higgs field, the strength of the coupling of a particle to the Higgs field is proportional to the mass of the particle. A Higgs boson is a neutral particle with spin 1. Since the SM does not predict the mass for the Higgs boson, a mass below $m_H = 114.4$ GeV is excluded by LEP [fHBS03] and a mass between 148 GeV to 180 GeV by Tevatron [CDF11]. The search for the Higgs boson is one of the tasks of current high energy particle experiments.

In collider experiments the Higgs boson is primarily produced through gluon fusion, boson fusion, associative production with W/Z boson or $t\bar{t}$ pair, while gluon fusion should dominate the process. The detector will identify the Higgs boson by one of its decay channels, most significant are the following:

$$\begin{aligned}
 H &\rightarrow \gamma\gamma \\
 H &\rightarrow WW \rightarrow l\nu l\nu \\
 H &\rightarrow WW \rightarrow l\nu qq \\
 H &\rightarrow ZZ \rightarrow lll \\
 H &\rightarrow ZZ \rightarrow ll\nu\nu \\
 H &\rightarrow ZZ \rightarrow llqq
 \end{aligned} \tag{2.2.1}$$

The latest results regarding the SM Higgs search from the ATLAS experiment are shown in Figure 2.3, showing the confidential limits on the exclusion of a Higgs boson in dependence its mass m_H . The searches for the Higgs boson from the different

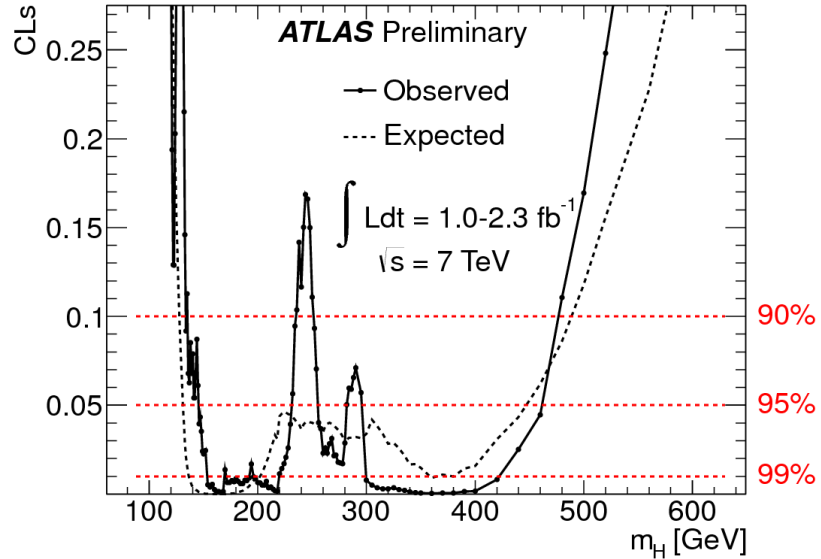


Figure 2.3: Expected and observed confidence levels on exclusion limits of the Higgs boson mass m_H [ATL11a].

decay channels are combined, allowing the exclusion over a broad mass range. The prior exclusion from LEP and Tevatron are expanded to nearly 500 GeV, although gaps in the range of 220 to 300 GeV leave the possibility for a SM Higgs boson.

2.3 High Energy Particle Physics

High energy particles produced and collided in particle accelerators are used to do in depth research of subatomic structures or the production of non stable particles. Collisions can either be produced by using a fixed target or colliding two beams of high energy particles with contrary velocity, the latter gives the advantage of adding up the energy of both participating particles. The results of these collisions are measured in cross sections σ . The cross sections σ_x of different interesting processes for $p\bar{p}$ and pp collisions is shown in Figure 2.4. The effectiveness of a collider is

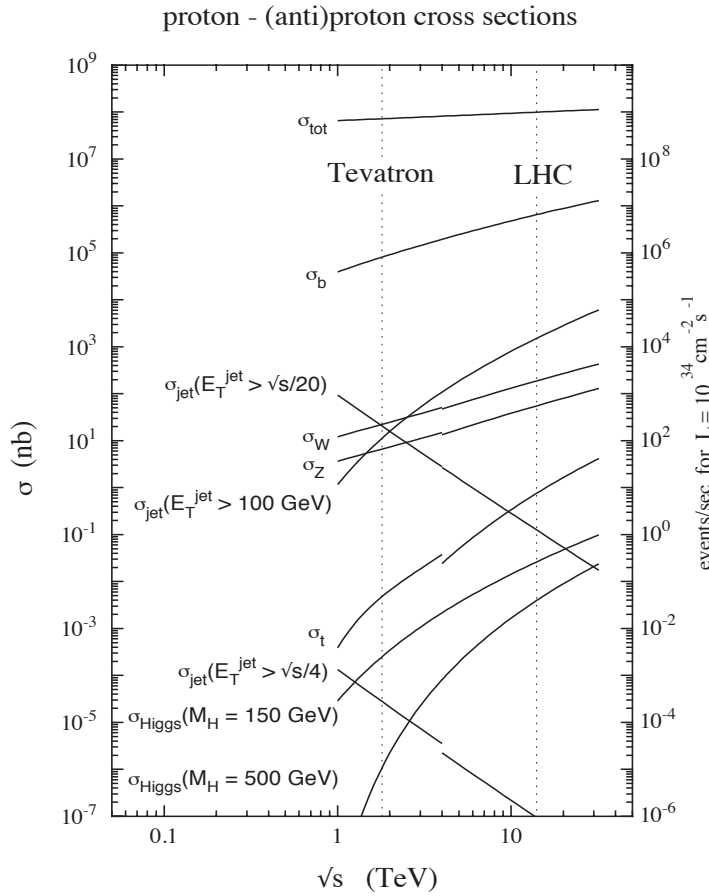


Figure 2.4: Different process cross sections in $p\bar{p}$ and pp collisions. Highlighted are the design energies of the Tevatron and LHC [ATL98].

measured by a quantity called luminosity $\mathcal{L}$, given by:

$$\mathcal{L} = \frac{N_b^2 n_b f_{\text{rev}} \gamma_r}{4\pi \epsilon_n \beta^*} F \quad (2.3.1)$$

The luminosity is determined by N_b the number of particles per bunch, n_b the number of bunches per beam, f_{rev} the revolution frequency, γ_r the relativistic gamma factor, ϵ_n the normalized transverse beam emittance and β^* the beta function of the collision point [Dop11].

By multiplying the luminosity $\mathcal{L}$ and the cross section of a process σ_x , the expected number of events per second N_x (as seen in Figure 2.4 for $\mathcal{L} = 10^{34} \text{ cm}^{-2}\text{s}^{-1}$) for this process can be calculated:

$$N_x = \mathcal{L}\sigma_x \quad (2.3.2)$$

Hence for recording a statistical significant number of events, for e.g. the Higgs boson, higher luminosities are needed.

Chapter 3

The Large Hadron Collider

The Large Hadron Collider (LHC) [EB08] is a 27 km circumference circular hadron collider, located at CERN¹ near Geneva (Switzerland) in the former LEP² tunnel 100 m under the ground. The LHC is designed for proton-proton and lead-lead collisions, at a center of mass energy of $\sqrt{s} = 14$ TeV (7 TeV per particle for proton-proton collisions).

3.1 The Accelerator Complex

The protons, which are accelerated and collided in the LHC, are supplied by a chain of accelerators (also shown in Figure 3.1):

1. LINAC 2: Linear particle accelerator, 50 MeV
2. BOOSTER: Proton synchrotron booster, 1.4 GeV
3. PS: Proton synchrotron, 25 GeV
4. SPS: Super proton synchrotron, 450 GeV

The 450 GeV protons from the SPS are then injected into the LHC and ramped up to the desired energy or a maximum of 7 TeV.

The particle acceleration is done via a Superconducting Radio-Frequency (SRF) cavity system, one for each beam. The SRF cavity systems are concentrated between the ALICE and CMS experiments. In these systems the particles gain 485 keV per turn, resulting in a ramp duration of 20 minutes for a 7 TeV beam.

The most challenging part for the LHC was to bend the protons onto a quasi circular trajectory. This is done by 1232 dipole elements shown in Figure 3.2. One element is about 16.5 m long, has a diameter of 570 mm, a mass of 27.5 t and is curved by an angle of 5.1 mrad in the horizontal plane. The nominal magnetic field inside the dipole for a 7 TeV beam is 8.33 T, to achieve the necessary bending radius of 2803.98 m. This enormous field strength is producible by using superconducting

¹European Organization for Nuclear Research (Conseil Européen pour la Recherche Nucléaire)

²Large Electron Positron collider

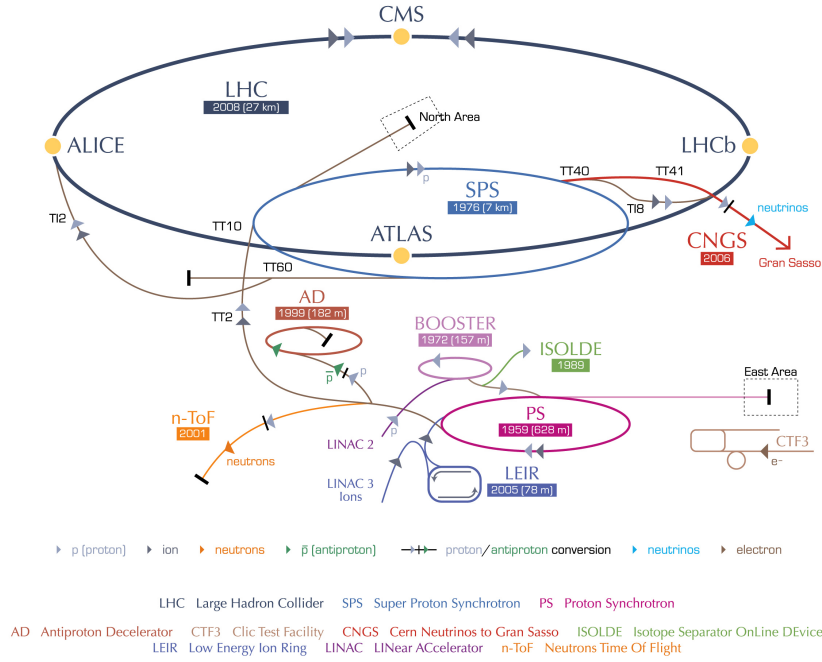
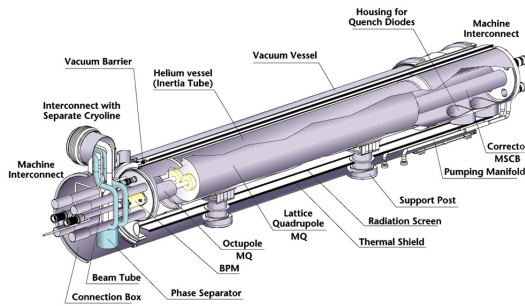


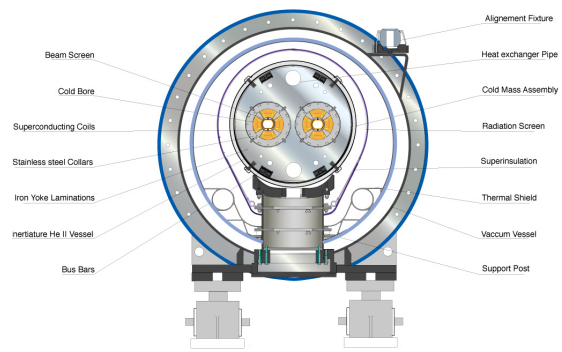
Figure 3.1: A schematic of CERN's accelerator complex, showing the LHC and its experiments, but also the injection chain needed for preparation of the particles [Lef08].

LHC Short Straight Section



3.2.1: Straight section [Car98].

LHC quadrupole cross section



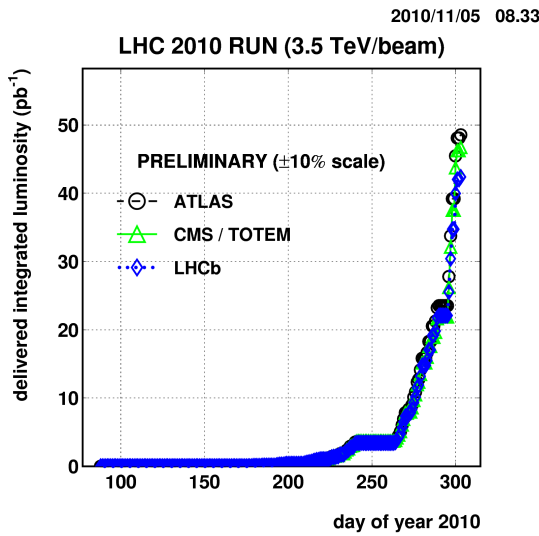
3.2.2: Cross section [Car98].

Figure 3.2: Schematic drawing of a LHC dipole element, showing the two beam pipes, the different magnet coils and yoke housed in the helium vessels, machine interconnect and service connections.

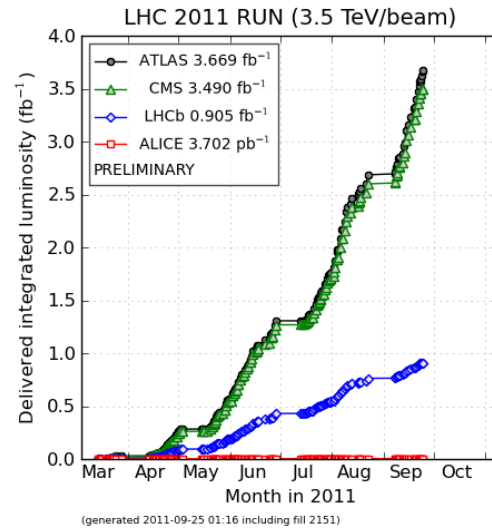
electronic cables and magnets operated at 1.9 K, cooled by superfluid helium, which allows the high currents needed for this field strength. In case of the nominal 8.33 T a current of 11850 A flows through the magnet and an energy of 6.93 MJ is stored in the magnet. The magnetic field is simultaneously ramped up with the acceleration of the particles, but cannot be simply switched off after e.g. a beam dump, but rather needs to be ramped down slowly over about 20 min. The LHC features further systems not described here to improve the machine performance in terms of e.g. beam cross section and bunching.

At the 20.11.2009 the first proton-proton beams circulated successfully in the LHC, without further acceleration at 450 GeV. The acceleration of the protons was done slowly in several steps, in order to ensure the flawless operation of the dipole magnets, as a malfunction could lead to a quench of the magnets seen on 19.09.08, which is why the start of the LHC was delayed by more than one year.

In 2010 the machine was first studied with 450 GeV per beam, then slowly ramped to the energy of $\sqrt{s} = 7$ TeV or 3.5 TeV per beam for the first phase of operation. In 2010 the LHC successfully delivered the first 50 pb^{-1} integrated luminosity as shown in Figure 3.3.1. In 2011 the LHC already delivered 3.6 fb^{-1} integrated lumi-



3.3.1: 2010 [CER10].



3.3.2: 2011 [CER11].

Figure 3.3: By the LHC delivered, integrated luminosity per day.

nosity (as seen in Figure 3.3.2) and holds the record for the highest beam energy and luminosity.

3.2 LHC Design Parameters

On the physics side, the purpose of the LHC is to allow the experiments searches for the Higgs boson and physics beyond the SM. On the machine side, this requires

the LHC to fulfill certain parameters or optimize them as far as achievable:

- A maximum magnetic field of 8.33 T, allowing a maximum energy per beam of 7 TeV.
- 10^{11} protons per bunch
- 2808 proton bunches with a separation of 25 ns
- 40 MHz collision rate

All these parameters result in a design luminosity of $\mathcal{L} = 10^{34} \text{ cm}^{-2}\text{s}^{-1}$. These are the parameters used for the design of LHC and are a nominal goal for the operation.

3.3 Experiments at the LHC

Several experiments reside at the LHC, two of them are general purpose detectors designed for high luminosities, one specialized on ion collisions and another one specialized on b -quark physics.

The ATLAS detector is discussed in Chapter 4, the other three large experiments are briefly outlined in the following.

3.3.1 CMS

The Compact Muon Solenoid (CMS) [CMS06] detector shown in Figure 3.4, is one of the high luminosity general purpose detectors. The CMS detector is 21 m long, 15 m

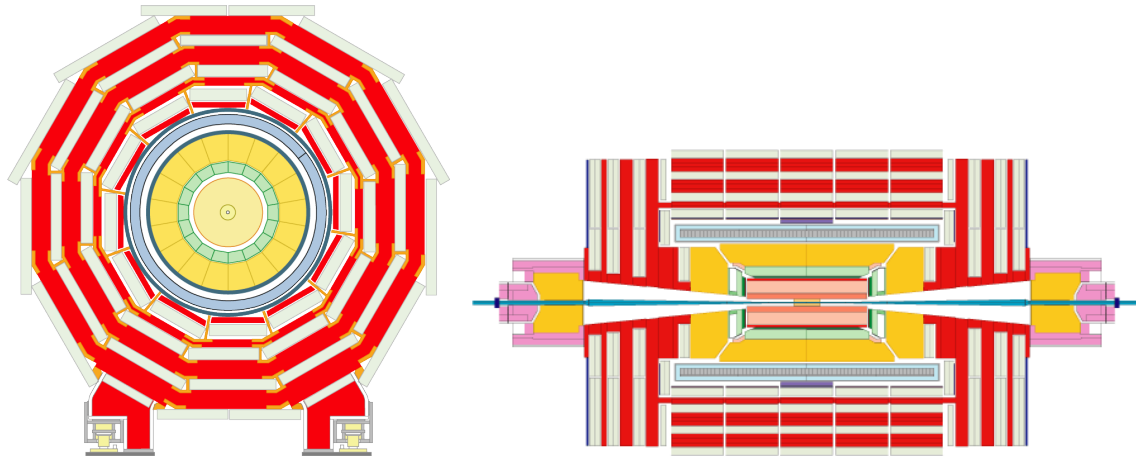


Figure 3.4: Cross and long section of the CMS detector. The different colors represent the different parts of the detector. Red: muon chambers, grey: iron yoke, blue: solenoid, yellow: hadronic calorimeter, green: electromagnetic calorimeter, light yellow: tracker [CMS11].

wide, 15 m high and has a total weight of 14000 t. Except the muon chambers the

rest of the detector is surrounded by a superconducting solenoid magnet producing a magnetic field of 4 T inside and 2 T outside of it. Being a general purpose detector, capable of detecting all known particles, CMS will search for the Higgs boson and physics beyond the SM.

3.3.2 ALICE

A Large Ion Collider Experiment (ALICE) [PKL⁺98] is shown in in Figure 3.5. The

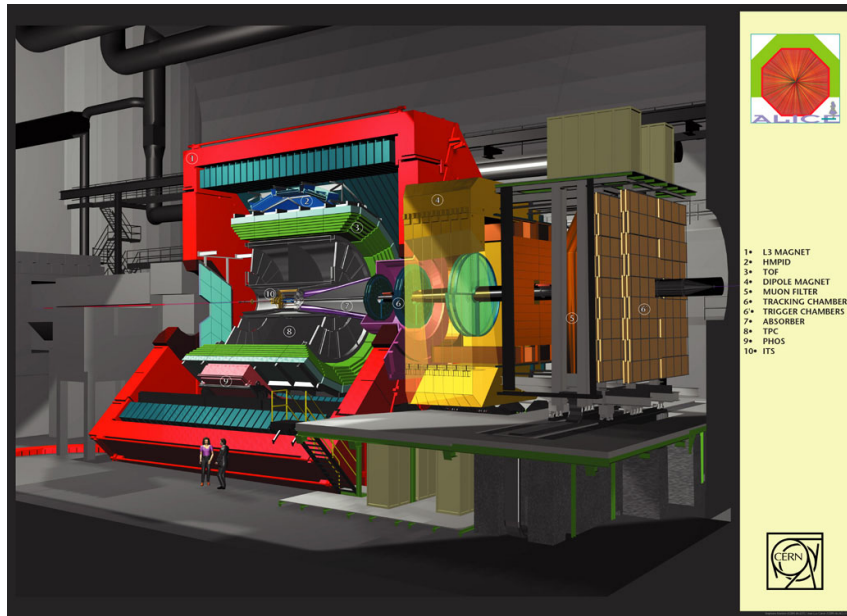


Figure 3.5: Layout of the ALICE detector, showing the cylindrical detector part and the forward muon spectrometer [Mar06].

ALICE detector is 25 m long, 16 m wide, 16 m high and has a total weight of 10000 t. It is specialized in quark-gluon plasma produced in heavy ion collisions, in this plasma quarks and gluons are not constrained by confinement, i.e. exist as free particles and are not bound in hadronic states. In theory this was the state of the universe up to 10^{-6} s after the big bang.

3.3.3 LHCb

The Large Hadron Collider beauty (LHCb) [LHC98] experiment is shown in Figure 3.6. LHCb is specialized in b -quark physics and its properties, featuring a sole forward detection and very precise vertex reconstruction. It is 21 m long, 13 m, 10 m high and has a total weight of 5600 t. The LHCb experiment aims to answer the question why our universe is dominated by matter and where all the anti-matter is gone by investigating the decay of $b\bar{b}$ pairs.

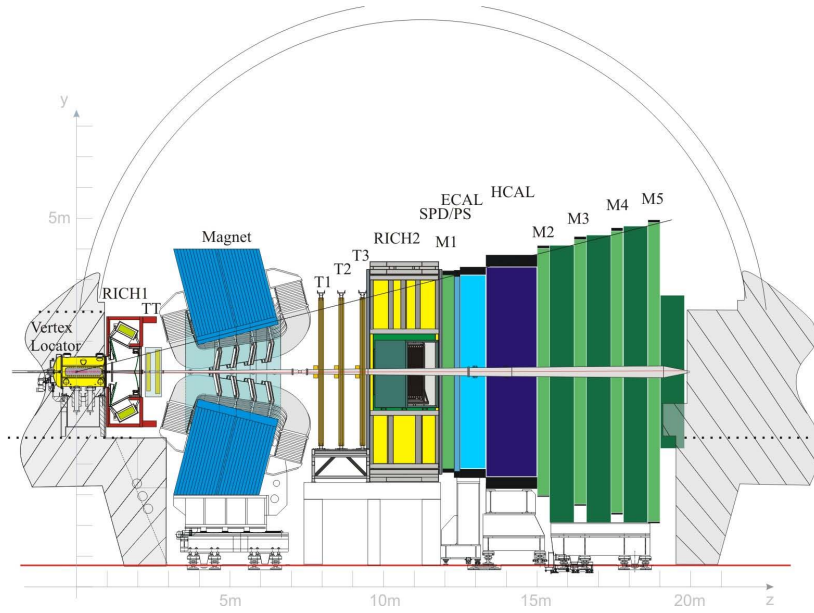


Figure 3.6: Schematic of the LHCb detector, showing its forward geometry with the interaction region at $z = 0$ [Lin08].

Chapter 4

The ATLAS Experiment

The ATLAS¹ [ATL99a] detector is shown in Figure 4.1. It is the largest of the LHC

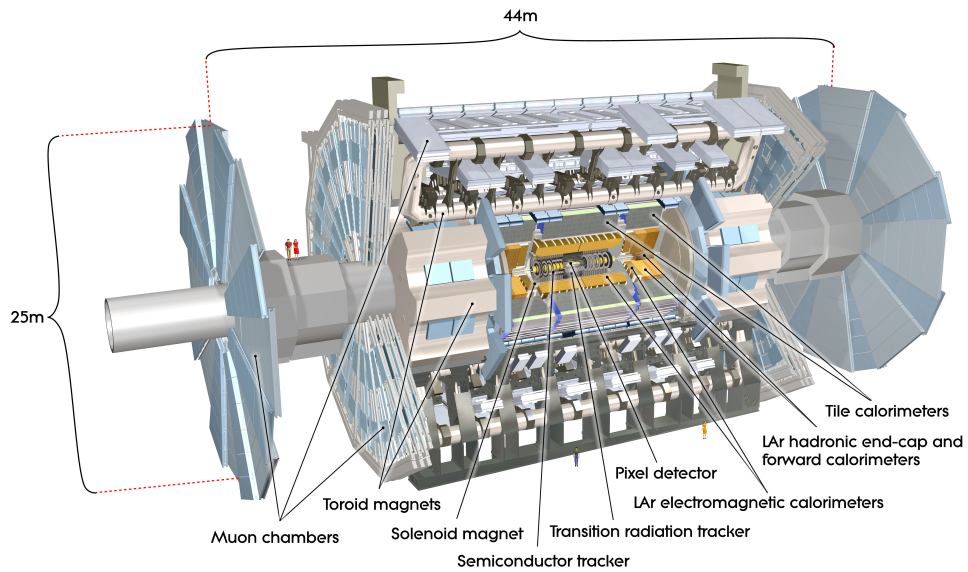


Figure 4.1: Layout of the ATLAS detector, showing the subdetectors and magnets [Peq08c].

experiments, being 45 m long, 22 m wide, 22 m high and has a total weight of 7000 t. Alongside the CMS detector, ATLAS is a general purpose detector, capable of detecting every known particle.

This thesis is focused on the development of an upgrade of the ATLAS Pixel detector, hence it will concentrate on the ATLAS detector. The following part will discuss the various subdetectors, the magnets and the trigger system of the ATLAS detector.

¹Abbreviation for “A Toroidal LHC Apparatus”.

4.1 Purpose of the ATLAS Experiment

The ATLAS experiment is capable of doing researches of many different types [ATL99b]. One of the most important goals of ATLAS is, searching for evidence of the SM Higgs boson or if no evidence is found, extending the SM Higgs boson mass exclusion. Also focused by the ATLAS experiment is the search for physics beyond the SM, e.g. SUSY. Furthermore the ATLAS experiment will participate in the investigation of the SM and many of the open questions regarding it, e.g. do quarks and leptons have a substructure, CP violation, t -quark physics or the existence of a fourth quark generation.

To cover all the fields named above, the ATLAS detector was optimized in high resolution measurements of photons, leptons and muons, secondary vertex detection and high resolution calorimetry.

4.2 Structure of the ATLAS Detector

The ATLAS detector primarily consists of four parts, the inner detector, the calorimeter, muon spectrometer and magnet system. The massive amounts of data gathered by the detector systems, is reduced to a manageable amount by the ATLAS trigger system.

The coordinate system the detector is described in, is chosen to be R , ϕ and η . The z -axis, used in mechanical descriptions, is along the beam pipe, with its origin in the middle of the detector. R is the radius and ϕ is the azimuthal angle around the z -axis. The pseudorapidity η is expressed by the polar angle θ :

$$\eta = -\ln \left(\tan \left(\frac{\theta}{2} \right) \right) \quad (4.2.1)$$

Hence $\eta = 0$ describes the plane orthogonal to the z -axis.

4.2.1 The Inner Detector

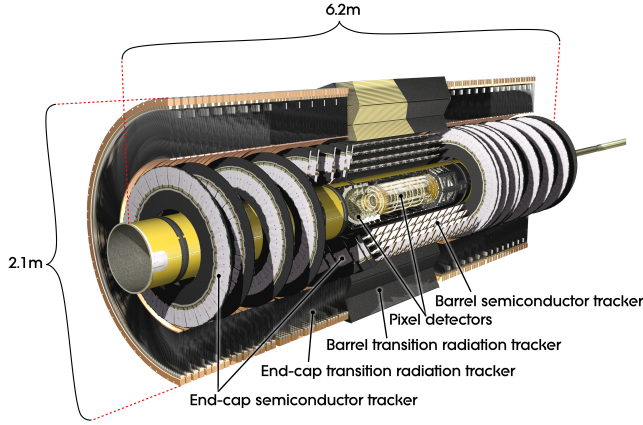
The inner part [ATL97a] is built of three subdetectors, all responsible for particle tracking. But as the surface of each subdetector increases proportional to the distance from the interaction point, three different tracking concepts are used.

Its layout is shown in Figure 4.2, in total it is 6.2 m long and has a diameter of 2.1 m. Every subdetector consists of a barrel and disks part, fully covering the volume up to $|\eta| = 2.5$, a detailed list of parameters is given in Table 4.1. The whole inner detector is built as lightweight as possible, not to obstruct the particles energy and trajectory.

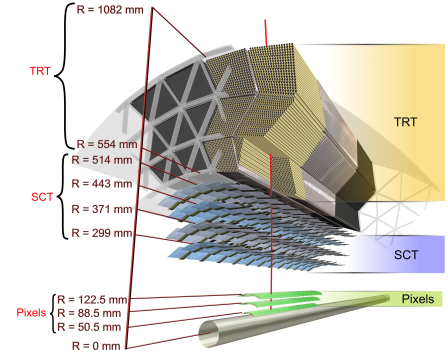
The Pixel Subdetector

The Pixel detector [WH98] is the innermost subdetector of ATLAS and nearest to the interaction point. It will be discussed in Chapter 5.

4.2. Structure of the ATLAS Detector



4.2.1: Cross section of the inner detector.



4.2.2: Schematic of the different barrel radii.

Figure 4.2: Layout of the ATLAS inner detector, composed of the Pixel, SCT and TRT subdetectors [Peq08b].

Table 4.1: Parameters of the inner detector [ATL97a].

System	Position	Area [m ²]	Resolution [μm]	Channels [10 ⁶]	η coverage
Pixel	1 <i>b</i> layer	0.28	$R\phi = 15, z = 115$	13	± 2.5
	2 barrel layers	1.16	$R\phi = 15, z = 115$	54	± 1.7
	3 end cap disks on each side	0.14	$\phi = 15, R = 115$	13	1.7 - 2.5
SCT	4 barrel layer	34.4	$R\phi = 16, z = 580$	3.2	± 1.4
	9 end cap wheels on each side	26.7	$\phi = 16, R = 580$	3.0	1.4 - 2.5
TRT	Axial barrel straws		170 (per straw)	0.1	± 0.7
	Radial end-cap straws		170 (per straw)	0.32	0.7 - 2.5
	36 straws per track				

The SCT Subdetector

The Semi Conductor Tracker (SCT) [Jac05] uses silicon strips for particle detection. The strips are arranged in sandwiched double modules, one having the strips parallel to the beam pipe and one rotated by 40 mrad, to achieve a higher resolution of $580\ \mu\text{m}$ in z direction, while having a resolution of $16\ \mu\text{m}$ in $R\phi$ direction.

The TRT Subdetector

The Transition Radiation Tracker (TRT) subdetector [ea04] is built of 4 mm straws forming 73 barrel layers and 14 end-cap wheels. These straws are aluminum coated polyamide tubes with a length of 1.44 m, filled with a Xe/CO₂/CF₂ gas mixture. The $30\ \mu\text{m}$ gold coated tungsten anodes inside the straws are operated at 1.5 kV. Located between the straws are two different types of foils, a polypropylene and polyethylene, respectively. By this the dielectric constant alternates between the foil materials and transversing relativistic particles emit transition radiation. The signature of the transition radiation allows the identification of electrons in the TRT subdetector.

4.2.2 The Calorimeter Systems

The calorimeters of the ATLAS detector are divided into two subdetectors (shown in Figure 4.3), the electromagnetic and hadronic calorimeter. The calorimeter sys-

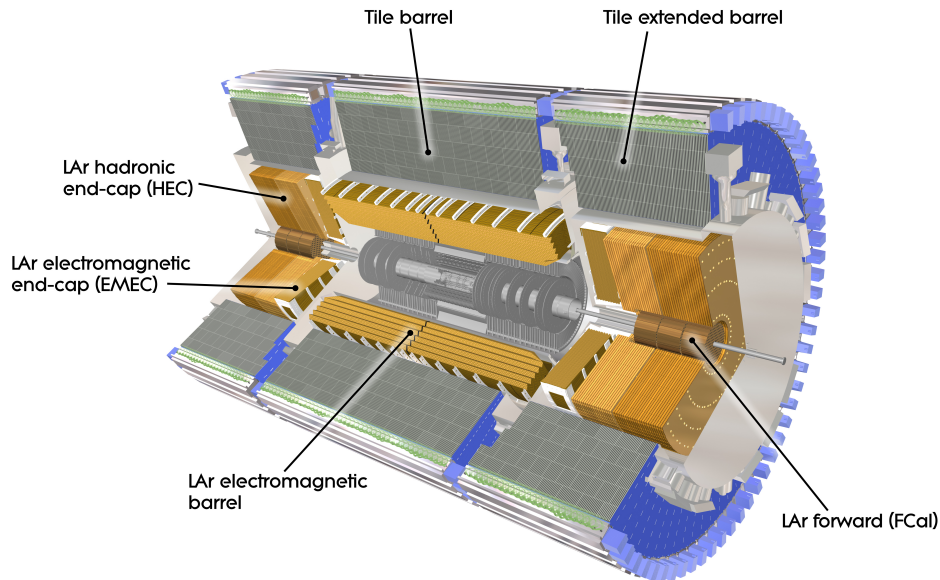


Figure 4.3: Calorimeter system of the ATLAS detector, the electromagnetic LAr calorimeter is surrounded by the hadronic Tile calorimeter [Peq08a].

tem measures the energy of particles by letting them interact with matter. In the

electromagnetic calorimeter the incoming particle energy is measured by the interaction via electromagnetic force, releasing a particle shower of low energy electrons and photons. Thus the electromagnetic calorimeter can measure the energy of incoming electrons² and photons, by measuring the energy of the particle shower they introduce. Also all incoming electrons and photons will be absorbed in the electromagnetic calorimeter.

Hadrons will pass through the electromagnetic calorimeter without much interaction, because much more absorber material is needed before they interact via strong force. This absorber material is contained in the hadronic calorimeter and it will detect and absorb any particle (except muons and neutrinos), which can pass through the electromagnetic calorimeter. The energy of the particles is also measured through the released particle shower.

The Electromagnetic Calorimeter

The Liquid Argon electromagnetic calorimeter [lar96] is divided into a barrel part and two end-caps, where the outer part of the end-caps is also responsible for hadronic calorimetry. The calorimeter consists of accordion shaped electrodes, lead absorber and liquid argon as active material. Incoming particles will ionize the liquid argon and the free charges produced are collected by the electrodes, the energy of the particle is then proportional to the amount of produced charge in the liquid argon.

The Hadronic Calorimeter

The hadronic Tile Calorimeter [til96] is built of iron and scintillating tiles, as the absorber and active material, respectively. The signal produced in the scintillator is measured in photo multiplier tubes and proportional to the deposited energy.

4.2.3 The Muon Spectrometer

The muon spectrometer [muo97] is the largest subdetector of the ATLAS detector (as seen in Figure 4.1), as it encloses all the other detector systems. It is partly built in the barrel toroid, as the air-core coils accommodate space for the muon chambers. The whole muon spectrometer is built of four detector concepts, optimized for their position or the purpose of triggering or precision measurement. The precision chambers are cathode strip chambers (CSC) and monitor drift tubes (MDT), the trigger chambers are the resistive plate chambers (RPC) and thin gap chambers (TGC). The instrumentation of the different muon systems are listed in Table 4.2.

²Muons will pass through the calorimeter without much interaction and taus will decay into muons or electrons mostly before they arrive at the calorimeter.

Table 4.2: Muon spectrometer instrumentation [ATL97b]

	Precision chambers		Trigger chambers	
	CSC	MDT	RPC	TGC
Number of chambers	32	1194	596	192
Number of readout channels [10^3]	67	370	355	440
Area covered [m^2]	27	5500	3650	2900

4.2.4 The Magnet System

The ATLAS experiment has two superconducting magnet systems operated at 4.5 K, which bend the trajectory of moving charged particles and allow measurement of their momenta. The inner detector is enclosed in the central solenoid magnet, which produces a mean field of 2 T. The central solenoid has an inner diameter of 2.44 m, outer diameter of 2.63 m, a length of 5.3 m and a total weight of 5.7 t. Placing the central solenoid in front of the calorimeter, demanded the central solenoid design to be minimized in terms of material.

The outer air-core toroid magnet is divided into two parts, the barrel toroid and end-cap toroid. They produce a field of 0.5 T to 2 T throughout the muon chambers. The end-cap toroids have an outer diameter of 10.7 m and are inserted into the barrel toroid, a length of 5 m and each one has a total weight of 239 t. The eight coils of the barrel toroid enclose the calorimeter and have an outer diameter of 20.1 m, a length of 25.3 m and a total weight of 830 t.

The field orientations of the barrel/end-cap toroid and the central solenoid are different. The central solenoid field is parallel to the z -axis, while the barrel/end-cap toroid field is orthogonal the z -axis.

4.2.5 Particle Identification in the ATLAS Detector

How different types of particles are detected by the ATLAS detector is shown in Figure 4.4. The tracker (inner detector) is able to detect every charged particle. Hence neutral hadrons e.g. neutrons, or photons are invisible to the tracking systems. Besides tracking the TRT subdetector allows early electron identification, due its detection concept based on transition radiation. Followed by the tracker, the electromagnetic calorimeter detects particles via electromagnetic interaction, i.e. mostly electrons and photons. Photons are then identified by leaving no tracks, but their energy is measured in the electromagnetic calorimeter. The hadronic calorimeter can then detect and measure the energy of the remaining particles, which should primarily be hadrons. Like the photon, a neutral hadron is identified by leaving a signal in the hadronic calorimeter, but no tracks in the inner detector. The only remaining particles behind the hadronic calorimeter are muons and neutrinos. Muons will be detected by the outermost detector system, the muon spectrometer. Neutrinos on the other hand will, due to their minimal interaction with matter, not be detected at all. By conservation of momenta the missing transverse energy E_t^{miss} in an event is attributed to neutrinos.

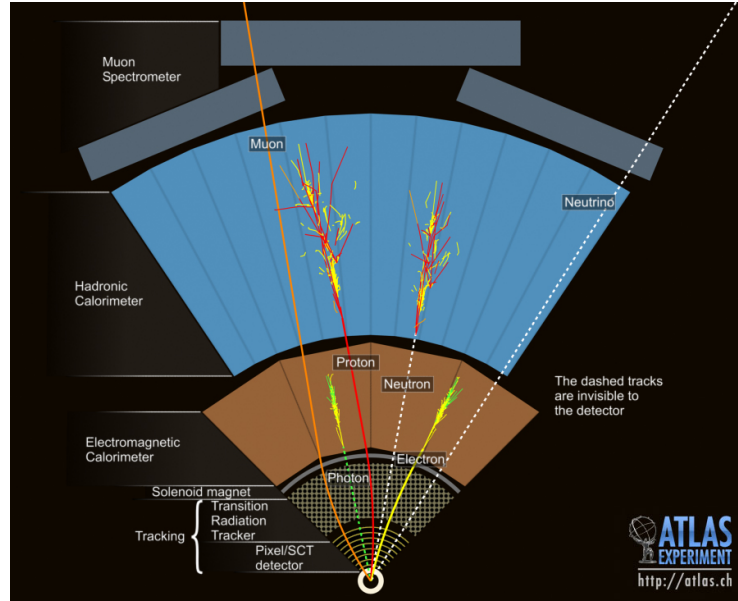


Figure 4.4: The detection of the various particles in the ATLAS detector [Peq08e].

4.3 The ATLAS Trigger System

The ATLAS experiment faces the problem that due to the interaction rate of 10^9 interactions per second, not all the data can be stored. Only interactions with interesting events should be stored, while background being rejected. This is done by the ATLAS trigger system in three steps (see Figure 4.5). At the LHC bunch crossing rate of 40 MHz, a decision for a trigger must be made every 25 ns. This is done by the Level 1 (LVL1) trigger, a hardware trigger reacting on simple signatures of interesting events, based on the data of the calorimeter and muon system. These signatures could be: Detected muons, a high energetic cluster in the calorimeter or other predefined signatures. The LVL1 trigger decides in about $2 \mu\text{s}$ to record or reject an event, in this time the specific detector readout parts need to buffer the event. The event rate is slowed down after the LVL1 trigger to the order of 100 kHz, which is still too much data to be recorded, hence the LVL1 trigger is followed by the LVL2 trigger. The LVL2 trigger decides on regions of interest and also takes the tracker data into account. The processing time per event is about 10 ms, in the meanwhile the data is stored in the ReadOut-Buffers. Having more decision time, the regions of interest are chosen by more detailed, e.g. a b -tag or isolated particles. The last step before the event is stored, is the event filter. In the event filter, the full event is collected and analyzed. The final recorded event rate is then in the order of 100 Hz, which is still a massive amount of data, but manageable by the CERN computing farms (Tier 0).

The trigger rates described above are the design trigger rates. Currently the trigger rates in the operation at a luminosity of $\mathcal{L} = 3 \cdot 10^{33} \text{ cm}^{-2}\text{s}^{-1}$ for the LVL1, LVL2 trigger and event filter are around 40 kHz, 3 kHz and 500 Hz, respectively. Therefore

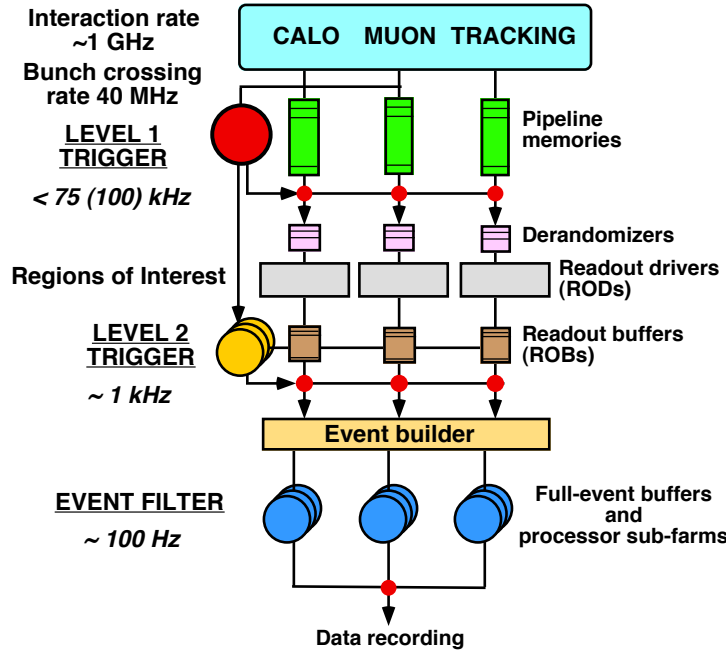


Figure 4.5: Layout of the ATLAS trigger system [ATL98].

the rates for the LVL2 and event filter already exceed the design rates.

4.3.1 Operation of the ATLAS Detector

Since march 2010 the LHC is operating with $\sqrt{s} = 7$ TeV and the ATLAS experiment has recorded a notable amount of data. Figure 4.6 shows the integrated luminosity delivered by the LHC and the recorded amount by the ATLAS detector. The LHC delivered a total of 3.62 fb^{-1} and the ATLAS detector recorded 3.40 fb^{-1} , hence ATLAS was operated with a data taking efficiency of 94%.

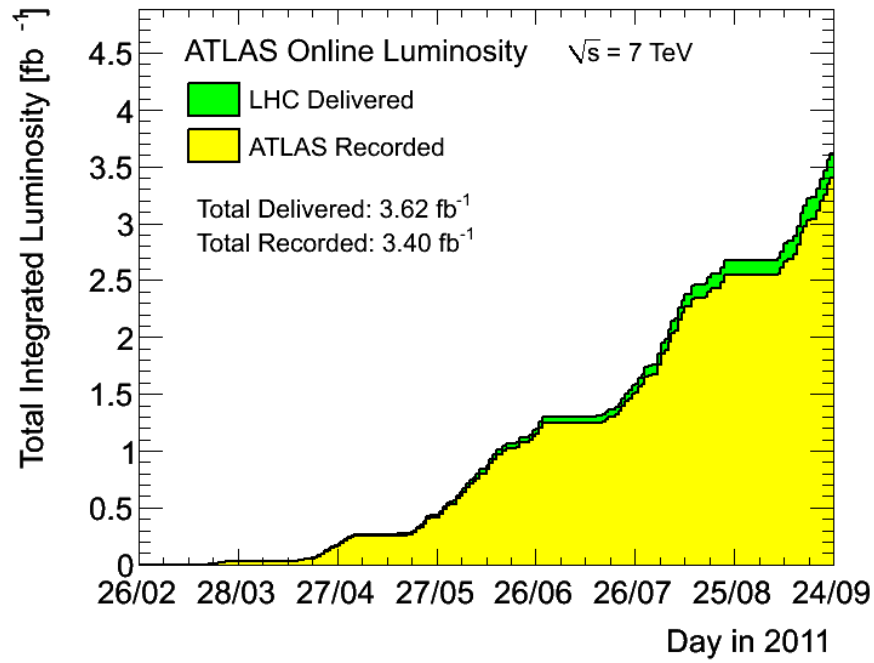


Figure 4.6: Integrated luminosity delivered by the LHC and recorded by the ATLAS detector at $\sqrt{s} = 7$ TeV [ATL11b].

Chapter 5

The Pixel Detector

The Pixel detector [WH98] is the innermost detector of the ATLAS experiment, responsible for detecting tracks of particles nearest to the interaction point. Figure 5.1 shows the opened Pixel detector in its support structure without services. The Pixel

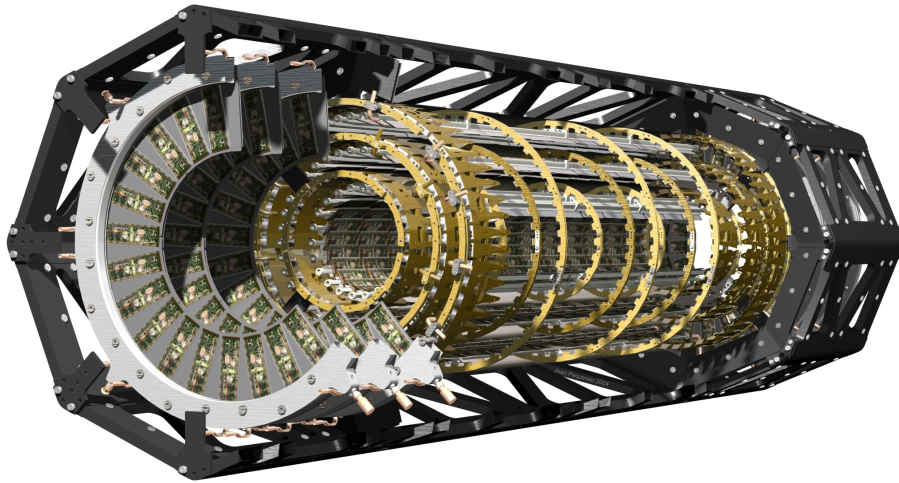


Figure 5.1: Computer generated image of the Pixel detector, opened for a better view [Peq08d].

detector is optimized for high resolution particle tracking and vertexing in a high multiplicity environment. The Pixel detector including its support structure has a length of 1.4 m, diameter of a 430 mm and a total weight of around 4.4 kg. It is divided into a barrel and disk part, consisting of a total of 3 barrel layers and 3 disk layers on each side. It provides particle tracking in a pseudo rapidity range up to $|\eta| = 2.5$. Being the innermost subdetector, the Pixel detector has to fulfill special design requirements concerning radiation hardness and light weight construction.

A detailed listing of the components in the Pixel detector is given in Table 5.1. The smallest unit of the Pixel detector is a Pixel module, carrying the Pixel sensor and its readout electronics. One Pixel module has 16 Front-End (FE) readout chips. 6 or 7 of these Pixel modules build one readout unit, two of these readout units and

Table 5.1: Components of the Pixel detector [Fli06].

Barrel					
	Radius [mm]	Staves	Modules	Chips	Channels [10 ⁶]
<i>b</i> Layer	50.5	22	286	4576	13.1
Layer 1	88.5	38	494	7904	22.8
Layer 2	122.5	56	676	10816	31.1
Subtotal		118	1456	23296	67
Disks					
	<i>Z</i> [mm]	Modules	Chips	Channels [10 ⁶]	
Disk 1	±495	48	1536	2.3	
Disk 2	±580	48	1536	2.3	
Disk 3	±650	48	1536	2.3	
Subtotal (both sides)		288	4608	13	
Total		1744	27904	80	

accordingly 13 modules are mounted onto a stave, in case of the barrel layers. The barrel layer are also referred to as (from inner to outer): The *b*-Layer (Layer 0), Layer 1 and Layer 2. Each of the disks is built of 8 sectors with 6 modules per sector, hence a disk carries a total of 48 modules. Also one sector or 6 modules of the disks build one readout unit.

Every module has a total 46080 pixels, which results in more than 80 million channels to be read out. This is about 90% of the channels of the complete ATLAS detector.

5.1 The Pixel Module

A Pixel module (shown in Figure 5.2) [GA08] consists of several electronic parts, one silicon sensor, 16 FEs and one flexible circuit board carrying the Module Control Chip (MCC). The sensor is bump bonded to the FEs, which are connected through a flex to the circuit board and the MCC. The whole Pixel module is interfaced by an ELCO connector connected to a Type0 cable.

5.1.1 The Pixel Sensor

The Pixel sensor is the sensitive part of the Pixel detector, with an active area of $6.08 \times 1.64 \text{ cm}^2$ ($z \times R\phi$). The sensor is partitioned in pixels in the size of $400 \times 50 \mu\text{m}^2$, hence a total of 47323 pixels per sensor. The pixels are arranged in arrays of 144 columns and 328 rows, in each column 8 pairs of pixels are used for

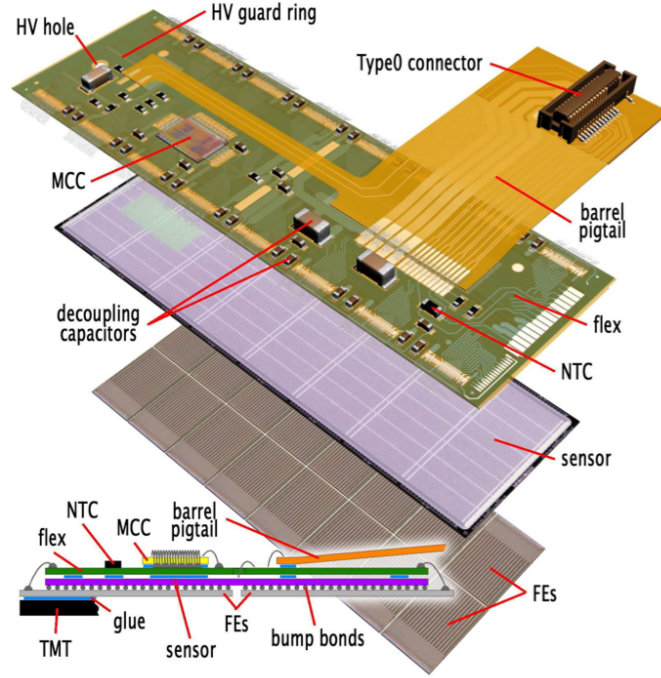
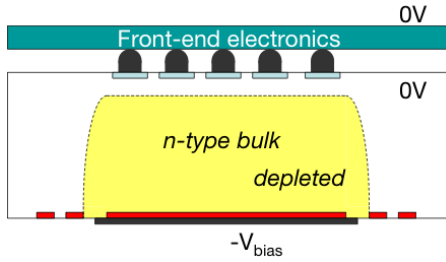


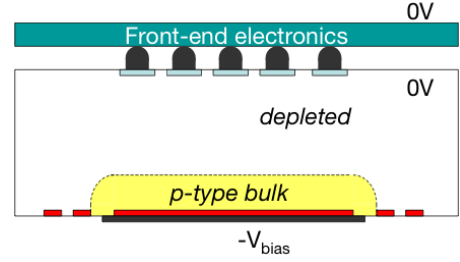
Figure 5.2: Exploded view and cross section of a Pixel module [GA08].

the readout, resulting in 46080 sensitive pixels per sensor.

A pixel is a reverse biased diode (shown in Figure 5.3), functioning as a solid state ionization chamber. A charged particle crossing the depletion zone ionizes the ma-



5.3.1: Before irradiation.



5.3.2: After irradiation and type inversion.

Figure 5.3: Detailed view of the n^+ -in-n pixel sensor and the depletion zone [GA08].

terial and the released charge is measured in the n-implantation (pixel).

Being located in a high radiation area, the n^+ -in-n bulk will undergo a type inversion. The n-type bulk will invert into a p-type bulk and the depletion zone (first growing from the bottom to the top), will grow from the top to the bottom (in reference to Figure 5.3). Therefore, the depletion voltage needs to be adjusted according to the situation of the type inversion in the sensor. The crystal structure of

the silicon sensor will suffer from irradiation in terms of cracks and degradation, to compensate this the depletion voltage V_{bias} changes with ongoing irradiation (from 150 V up to 700 V).

5.1.2 The Pixel Front-End Electronics

The readout of the Pixel sensor is done by 16 FEs, the FE-I3 chips (shown in Figure 5.4), each connected to 2880 pixels. Every pixel of the sensor is connected via a

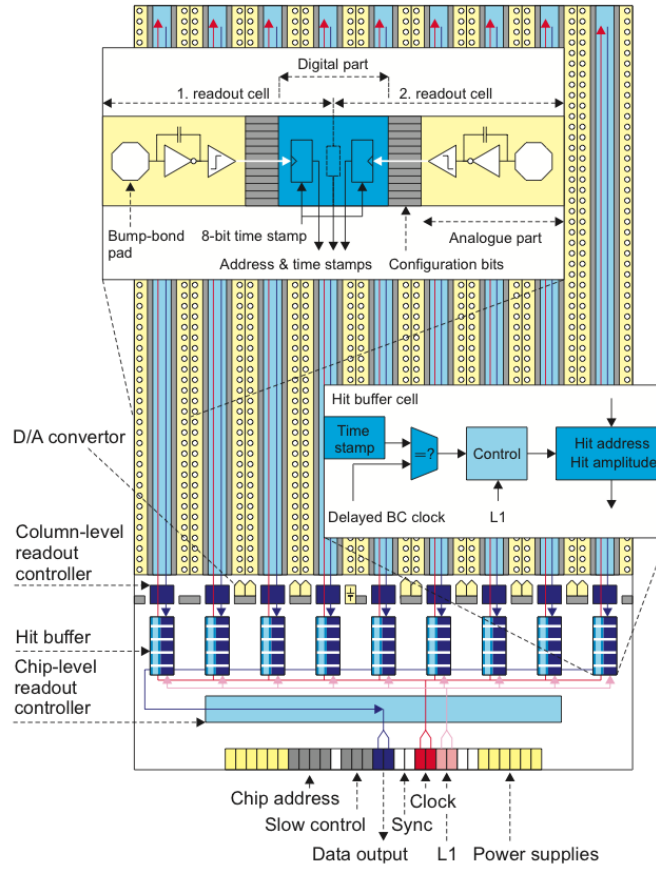


Figure 5.4: Schematics of the FE-I3 chip electronics [GA08].

bump bonds to one pixel readout cell of the FE-I3, which is an analog amplifier and discriminator. The resulting outgoing signal is a digital pulse, whose length is called the Time over Threshold (ToT). It depends on the charge deposited in the Pixel sensor. The pixel readout cell also provides testing mechanisms to calibrate each cell. Capacitors in front of the analog amplifiers can be used to inject a known charge into the readout cell and configure the various DACs¹ and thresholds. The basic functionality of the FE-I3 can be tested by a digital injection, where a digital pulse emulating the discriminator output with a known ToT value is injected.

¹Digital to Analog Converter

At the registration of a hit, the pixel readout cell saves a timestamp of the rising and falling edge of the hit in reference to a counter running with the 40 MHz reference clock. Upon the finished registration of the hit, the column based readout is initiated and controlled by the end of column logic. The end of column logic always pairs two columns, an odd and even one, and will read out the double column upon receiving the information that a pixel cell registered a hit. The hit information is then stored in a buffer and if a Level 1 trigger (LV1-trigger) with the right timestamp is received, the hits of all double column buffers matching the timestamp will be read out and send to the MCC.

5.1.3 Module Control Chip

The MCC is the interface between the FEs and the higher level readout. It receives a 40 MHz reference clock and 40 Mb/s commands and has to distribute them to the FEs. The other way around, the MCC receives the hit data from the 16 FEs and needs to send them to the higher readout. The hit data from the FEs will be buffered and processed by a event fragment builder. The event fragment builder will build a frame based serial data stream, putting the hit information from every FE marked with its respective FE number and attaching the LV1-trigger Id (LV1Id) and Bunch Crossing id (BCId). The LV1Id is incremented for every trigger received. The BCId is a internal counter incremented by the 40 MHz and therefore a tag for each event in a bunch crossing. The serial data output can have a bandwidth of either 40 Mb/s (one data output with 40 Mb/s), 80 Mb/s (two data outputs with 40 Mb/s or one data output with 80 Mb/s) or 160 Mb/s (two data outputs with 80 Mb/s).

5.2 The Pixel Detector Data Acquisition

The Pixel detector Data Acquisition (DAQ) system is connected to the detector via an optical link. The structure of the optical link and the connected off-detector electronics is shown in Figure 5.5, showing the data path for a single Pixel module. The Optoboard is the electro-optical converter at the on-detector side. It will send the data via 80 m of optical fiber to the Back-of-Crate card (BOC) [Fli06], which is paired inside a VME² crate with a Read-Out-Driver (ROD) [JJF⁺06]. Each VME crate contains one Trigger and Control Interface Module (TIM) [BLPW04]. The VME crate itself is controlled by a single-board computer.

The timing, trigger and control signals (TTC) of the TX path are BiPhase Mark (BPM) [Neu11] encoded, allowing the 40 MHz clock and data being merged into one DC-balanced signal. The data output or RX path respectively is Non-Return-to-Zero (NRZ)³ encoded.

The *b*-Layer data output has a bandwidth of 160 Mb/s and is therefore split into

²Standardized by IEC as ANSI/IEEE 1014-1987 [IEE85].

³Easiest data encoding possible. A low signal corresponds to logical “0” and a high signal to a logical “1”.

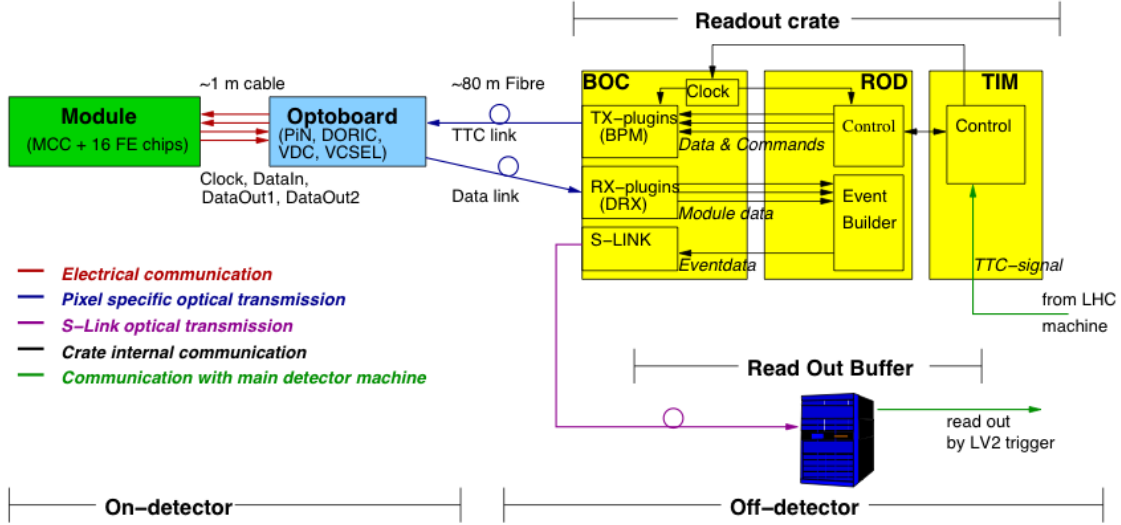


Figure 5.5: Schematic of the Pixel detector DAQ structure [Fli06].

two 80 Mb/s streams. Layer 1 and the end-cap disks each have a single 80 Mb/s stream per module, while Layer 2 is operated at a bandwidth of 40 Mb/s.

5.2.1 Optoboard

The Optoboards are located on the Patch Panel 0 (PP0) and are connected to one readout unit of 6/7 modules. The BPM encoded TX signals are optically received and converted into electrical signals by a PiN⁴ diode. The BPM stream is then decoded in a special ASIC⁵ called DORIC⁶ to the 40 MHz reference clock and commands. The data stream from the MCC is converted into an optical signal by the VDC⁷ connected to a VCSEL⁸.

There are two types of Optoboards, one for Layer 1, 2 and the end-cap disks and one for the *b*-Layer. Because the *b*-Layer has two data outputs for each module, the VDC and VCSEL components on the *b*-Layer Optoboards are doubled.

Every Optoboard has the capability of 8 channels, hence all the hardware necessary for the electro-optical conversion exists in arrays of 8. In the optical fiber this is done by connecting it to the PiN diodes or the VCSEL with an MT connector, holding a ribbon of 8 fibers.

⁴Positive intrinsic Negative

⁵Application Specific Integrated Circuit

⁶Digital Opto-Receiver Integrated Circuit

⁷VCSEL Driver Chip

⁸Vertical Cavity Surface Emitting Laser

5.2.2 Back-of-Crate Card

On the off-detector side the optical signals are received and transmitted by the BOC. The picture in Figure 5.6 shows the hardware layout of the BOC, overlaid with four colored layers, representing the main functions of the BOC: The control section, the S-Link section, the clock and command transmission (TX) section and data reception (RX) section.

Clock and Control Section

The main 40 MHz clock on the BOC is synchronized to the LHC clock, received through the VME backplane from the TIM. This clock drives all the synchronous logic on the BOC and is also the same on the ROD. The BOC is controlled from the ROD via the Setup Bus, an asynchronous parallel bus, which is in principle a simplified extension of the VME bus over the ROD. Via the Setup Bus all registers and DAC settings of the different section are controlled.

S-Link

The S-Link HOLA [OB97] is an optical transceiver daughterboard developed by the CERN electronics group. On the BOC it is used to transmit the processed data from the ROD to the higher level readout. The optical device used is a commercial module with 1 transmission and 1 reception channel.

TX Section

The TX section receives the TTC signal from the ROD and routes them to the TX plugin (shown in Figure 5.7). Located on the TX plugin is the BPM-12 ASIC, used for BPM encoding and signal adjustment and an 8 way VCSEL array (like the one on the Optoboard). The TX plugin is controlled over five parameters listed in Table 5.2. The Laser Forward Current DAC setting controls the driving current of

Table 5.2: TX section control parameters

Parameter	Range (in DAC settings)	Comment
Laser Forward Current	0 - 255	160 $\Leftrightarrow$ 10 mA forward current
BPM MarkSpace	0 - 31	$\approx \pm 5$ ns around the 50% setting
BPM CoarseDelay	0-31	step size 25 ns
BPM FineDelay	0 - 127	step size ≈ 0.3 ns
BPM Inhibit	0 or 1	1 = only clock is included into the BPM stream

the VCSEL and thus the laser output light power. The last four settings are used to adjust the signal after BPM encoding. The Mark to Space Ratio (MSR), also known as duty cycle, is adjusted with the BPM MarkSpace setting, allowing a pulse width

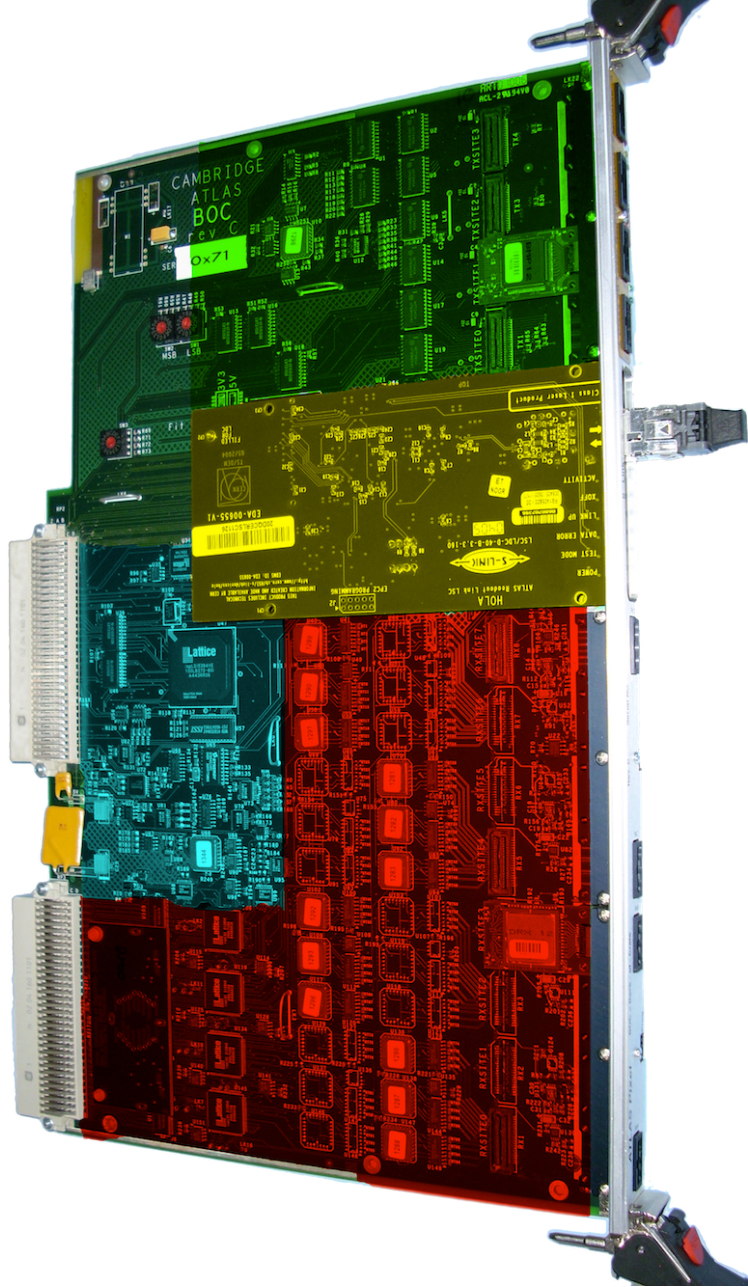


Figure 5.6: Picture of the BOC hardware layout, the overlaying colors represent the TX section (green), S-Link (yellow), clock and control section (light blue) and RX section (red).



Figure 5.7: Picture of a TX plugin used on the BOC to BPM encode the TTC signal and convert them into optical signals.

change of ± 5 ns around the 50% value. The MSR adjusted signal is then delayed with a coarse delay (1 clock cycle/25 ns per setting) and a fine delay (330 ps per setting). The BPM inhibit setting controls the BPM encoding process, by either including the data stream or encoding the 40 MHz clock with a constant “0” data value, resulting in a 20 MHz clock. The number of TX plugins on one BOC is equal to the number of readout units it is designated to read out.

RX Section

The RX section features two main functions, it converts the optical data stream into electrical signals and samples the data stream. The optical signal is received by the RX plugin (shown in Figure 5.8) and converted into an electrical signal by a PiN diode and the DRX ASIC, an amplifier chip. For the incoming optical

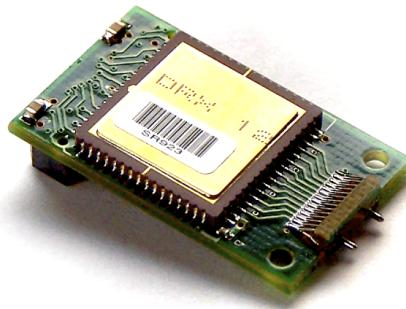


Figure 5.8: Picture of a RX plugin used on the BOC to receive the optical data stream and convert it into electrical signals.

signal a threshold controlled by 256 DAC settings can be applied, where each step corresponds to $1 \mu\text{A}$ PiN current threshold. The electrical data stream is sampled with the reference clock. To achieve the best sampling, the data stream can be

delayed by the PHOS-4 ASIC [Gro00] in the range of 0 to 24 ns⁹. The number of RX plugins on one BOC is defined by the bandwidth of the readout unit with the maximum of 32×40 Mb/s data channels acceptable for the ROD. The modules of Layer 2 send the data with 40 Mb/s, therefore a Layer 2 BOC is connected to 4 readout units (4 RX plugins). Layer 1 has a 80 Mb/s data stream that is split into two 40 Mb/s data streams on the BOC, one carrying the even bits and the other the odd bits¹⁰, hence a BOC connected to Layer 1 is connected to 2 readout units (2 RX plugins). In case of *b*-Layer module, 1 TX channel corresponds to 2 RX channels, thus a *b*-Layer BOC is equipped with 1 TX plugin and 2 RX plugins, serving 1 readout unit running at a bandwidth of 160 Mb/s (80 Mb/s per RX plugin).

5.2.3 Read-out-Driver

The ROD (shown in Figure 5.9) is the main data processing unit in the Pixel DAQ system. It is responsible for sending commands to the modules, process the data received from the modules and provide monitoring capabilities during operation. The ROD is interfaced from the single board computer via the VME bus and the LV1-trigger signals are received from the LHC machine over the TIM. The 40 MHz system clock is received from the BOC and is synchronized to the LHC clock.

The module data passed from the BOC to the ROD is processed in 8 formatters. Each formatter is able to process 12×40 Mb/s data stream, whereof only 4 channels are used in the Pixel detector readout. The formatters check the format of the data and rearrange it into one data stream, as it was split up on the BOC in 2 or for higher bandwidth 4 streams. The module data is then fed into the event fragment builder, which detects errors in the header/trailer, unphysical data and also checks the data for given MCC or FE errors. The router receives the data from the event fragment builder and formats the data into its final format. The final event is then passed to the S-Link on the BOC and transmitted to the Read-Out-Buffer of the higher level readout.

A fraction of the data is copied by the router to 4 Digital Signal Processors (DSP), where online monitoring of the data and histogramming for calibration is performed.

5.3 The Pixel Detector Services

The services of the Pixel detector are organized in 5 different patch panels (shown in Figure 5.10). Patch panels are intersections containing further hardware for the provided services. The power supplies for High Voltage (HV) and Low Voltage (LV), the Detector Control System (DCS) and DAQ system are located in the counting rooms about 80 m away from the detector. One HV and two LV power supply channels supply one readout unit. The PP4 splits up the channels into 6/7 channels one for each module. The PP3 translates analog signals e.g. temperature, humidity

⁹By inverting the reference clock, an overall sampling range of 0 to 49 ns is achieved.

¹⁰Corresponding to the sampling of one 80 Mb/s data stream with the 40 MHz clock and the inverted 40 MHz clock.

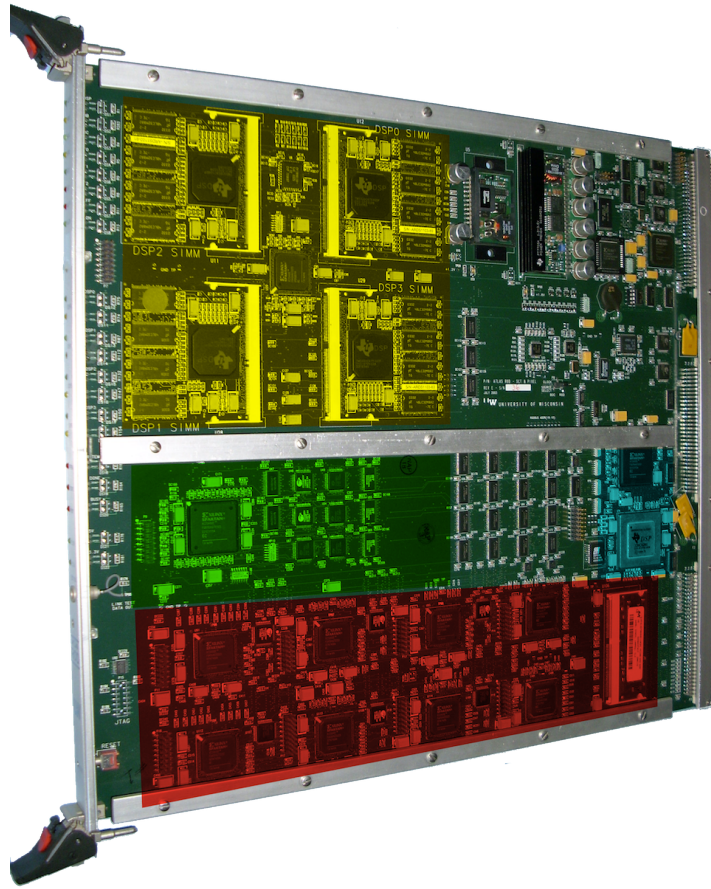


Figure 5.9: Picture of the ROD hardware layout, the overlaying colors represent the 8 formatters (red), event fragment builder (green), DSP farm and router (yellow) and control section (light blue).

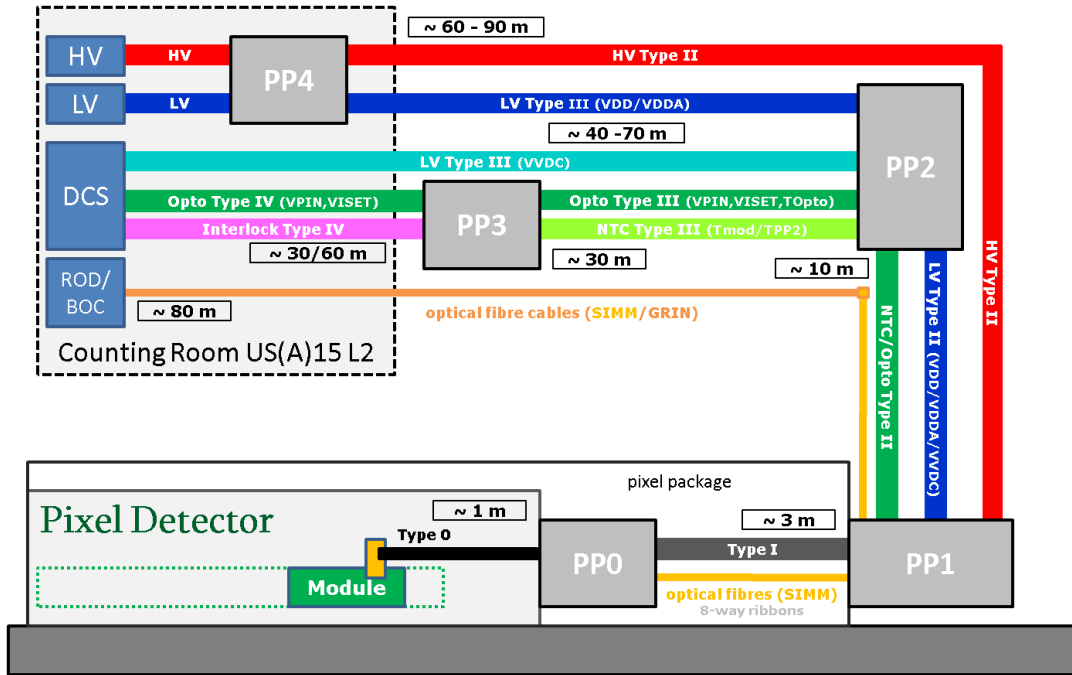


Figure 5.10: Schematic of the Pixel detector services organized in 5 patch panel intersections [Lan11].

and voltages into digital signals interpreted by the DCS. The LV supply voltage for the Pixel modules and the Optoboard needs to be regulated close to the detector, due to the voltage drop over the long distance cable. This is done in the PP2, located in the region of the muon spectrometer in the outer region of the ATLAS detector. The PP1 is located at the Inner Detector End Plate (IDEP) and marks the end of the Pixel detector service quarter panels. From the PP1 all supplies, monitoring signals and the optical link are routed over the service quarter panels to the PP0, at the end of the Pixel detector. For the readout the electro-optical conversion is done by the Optoboards on the PP0 and all the Pixel modules are connected to the corresponding supply and readout channel.

Chapter 6

The Insertable b -Layer

The Insertable b -Layer (IBL) [CDE⁺10] will upgrade the Pixel detector with a fourth innermost barrel layer. It will be deployed during the LHC shutdown in 2013. There are several motivations for the IBL:

- The b -Layer of the Pixel detector will suffer from irreparable failures before it will be replaced in 2022, due to the high radiation it is exposed to. Thereby tracking and particularly b -tagging inefficiencies will arise, which will be compensated by the IBL.
- The Pixel detector was designed for a peak luminosity of $\mathcal{L} = 10^{34} \text{ cm}^{-2}\text{s}^{-1}$. Twice this luminosity is expected before the high luminosity upgrade of the LHC, where a completely new Pixel detector will be installed. With a luminosity higher than the Pixel detector design luminosity, the event pileup will increase to a factor limiting the b -tagging and vertex reconstruction efficiency.
- The IBL will be installed even closer to the interaction point and thus allow an even more precise vertex reconstruction and b -tagging.

The vertex reconstruction efficiency in dependence of the number of pile up events for the current Pixel detector with and without the IBL is shown in Figure 6.1. At the design luminosity about 10 events happen per interaction, this number increases proportional to the luminosity and vertex reconstruction inefficiencies arise for the current Pixel detector without the IBL. At 50 pile up events per interaction, the IBL improves the vertex reconstruction by about 5% points.

As the IBL will be installed directly on the beam pipe, the ATLAS detector will receive a new beam pipe and support structures in the inner detector region. In general the technology developed for the IBL project will yield valuable information for the LHC high luminosity upgrade of the Pixel detector.

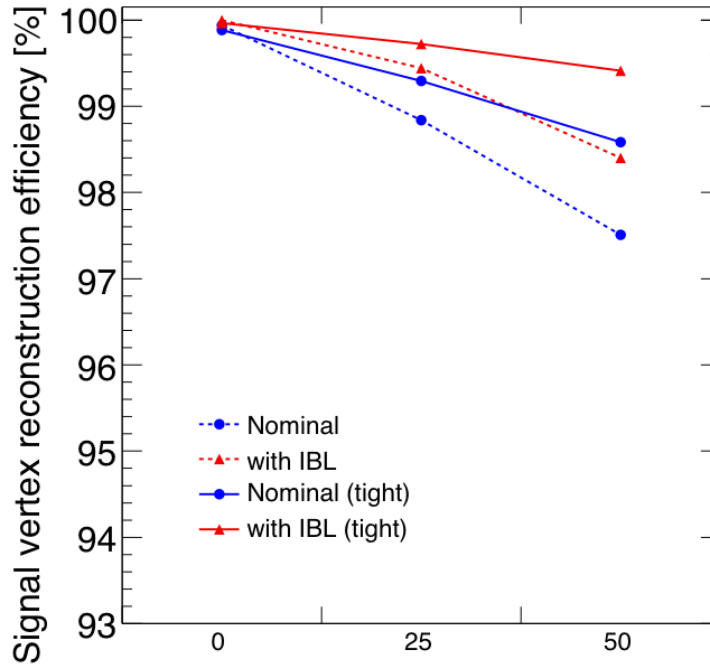


Figure 6.1: Vertex reconstruction efficiency in dependence of the number of pile up events for the current Pixel detector with and without the IBL [CDE⁺10].

6.1 The IBL Design and Requirements

In order to be installed between the current Pixel detector *b*-Layer and the new beam pipe, the IBL has to meet several requirements:

- The IBL electronics have to withstand an even higher radiation dose than the current *b*-Layer. Therefore all electronics deployed inside the detector volume, have to be developed within an even more radiation hard technology.
- The current Pixel modules have inactive edges in the longitudinal direction, thus they are shingled (along the beam pipe) and overlay each other. This allows the Pixel detector to have an active surface without gaps. Due to the small space the IBL will be built in, the IBL modules can not be shingled and must have a active fraction of above 90% with active or very slim inactive edges.
- The Pixel detector was built as lightweight as possible to minimize the radiation length. The radiation length of the IBL will be minimized even further and represent only 60% of the current *b*-Layer.
- The IBL will be inserted into the Pixel package along with the new beam pipe. The extraction of the old beam pipe and insertion of the IBL will require new tooling and redesigned services.

6.1. The IBL Design and Requirements

The IBL will have a length of 664 mm, an inner radius of 30 mm and an outer radius of 40 mm. The components of the IBL, which will add up to the Pixel barrel detector, are listed in Table 6.1. In Figure 6.2 an opened model of the IBL is shown.

Table 6.1: Components of the IBL.[CDE⁺10]

	Radius [mm]	Staves	Modules	Chips	Channels [10 ⁶]
IBL	33.25	14	224	448	6.02

Visible are 8 of the 14 staves mounted on their support structure and the electrical connection made to the modules by the flex, which is installed on the outer side and wraps around the stave to the FE. The outer beam pipe radius will shrink from

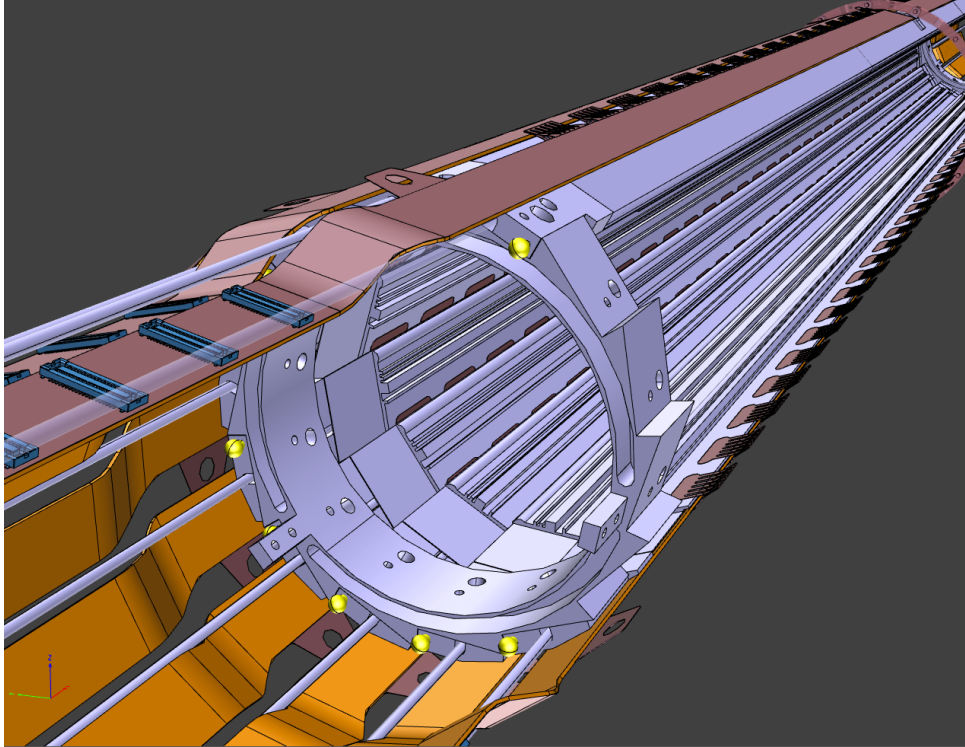


Figure 6.2: 3D model of the opened IBL, showing the staves mounted in their support structure and the electrical connection to the FEs made by the flex [Lap].

36 mm to 29 mm. The placement of the IBL with respect to the current *b*-Layer and the new beam pipe is shown in Figure 6.3. The IBL will be enveloped by the inner support tube (IST) at an outer radius of 43.5 mm. Table 6.2 lists the sensitive region parameters. The tilt angle and overlap of the staves guarantee a consistent coverage in ϕ . The z direction is not fully covered, due to dead regions of the sensor and the space in R direction does not allow a tilting and overlap in z direction. However the gap between each module is minimized by using a sensor technology with active or slim inactive edges.

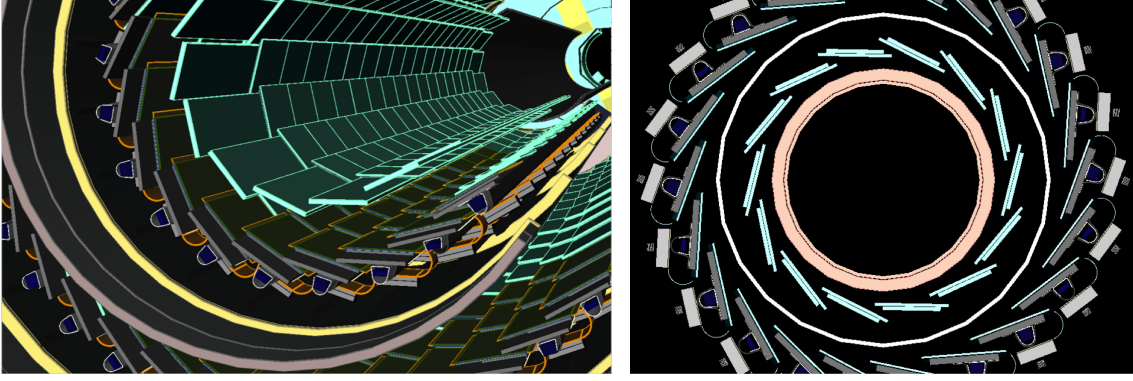


Figure 6.3: Layout of the IBL in respect to the Pixel detector b -Layer and new beam pipe [CDE⁺10].

Table 6.2: The IBL sensitive region parameters [CDE⁺10].

	Value	Unit
Number of modules per stave	16	
Module active region ($R\phi \times z$)	16.8×40.8	mm^2
Pixel size ($R\phi \times z$)	50×250	μm^2
η coverage (no vertex spread)	$ \eta < 3.0$	
η coverage (vertex spread $2\sigma = 112 \text{ mm}$)	$ \eta < 2.58$	
Stave center tilt angle in ϕ	14	$^\circ$
Stave min tilt angle in ϕ	-0.23	$^\circ$
Stave max tilt angle in ϕ	27.77	$^\circ$
Overlap in ϕ	1.82	$^\circ$

Although the IBL will benefit of new technology, it is classified as a part of the Pixel detector. Therefore design and development have to be done under the aspect to fit into the Pixel detector concept and infrastructure. This is important especially for the DAQ and services concept.

6.2 The IBL Module

An IBL module will consist of a new sensor technology and a new readout chip. Three different sensor types were developed for the IBL, a planar [Gre11], a 3D [Gre11] and a diamond [MW08] sensor. Each sensor type has specific advantages and disadvantages.

The current decision foresees a production of the planar sensor for the whole IBL and a smaller production of the 3D sensor, in case of using 3D sensors for some parts of the IBL. The new FE is a further advancement of the Pixel detector FE-I3 chip, called FE-I4. The FE-I4 is designed for the readout of a much bigger portion of the sensor than the FE-I3 and two FE-I4 will be sufficient for one IBL module.

The overall size of an IBL module is smaller than the current Pixel detector modules (about 20 mm in z), but due to the smaller pixel size of $50 \times 250 \mu\text{m}^2$ ($R\phi \times z$), it will contain 53120 pixels. These pixels are arranged in 160 columns and 336 rows, hence one FE-I4 will be connected to 80 columns and 336 rows and will perform the readout of 26560 pixels.

The basic specifications of the FE-I4 are given in Table 6.3. A block diagram of the FE-I4 in Figure 6.4 gives an overview over the layout of the logical blocks.

Table 6.3: Basic FE-I4 specifications [FE-11].

	Value	Unit
Pixel size ($R\phi \times z$)	50×250	μm^2
Maximum charge	100,000	e^-
DC leakage current tolerance	100	nA
Pixel array size	80×336	Col $\times$ Row
Hit-trigger association resolution	25	ns
Charge resolution	4	bit
Radiation tolerance	300	MRad
Operation temperature	-40 to +60	$^\circ\text{C}$
Readout initiation	Trigger	
Max. number of consecutive triggers	16	
Trigger latency (max.)	6.4	μs
Maximum sustained trigger rate	200	kHz
External clock input	40	MHz
Single serial command input	40	Mb/s
Single serial data output	160	Mb/s
Output data encoding	8b10b	
I/O signal	LVDS	

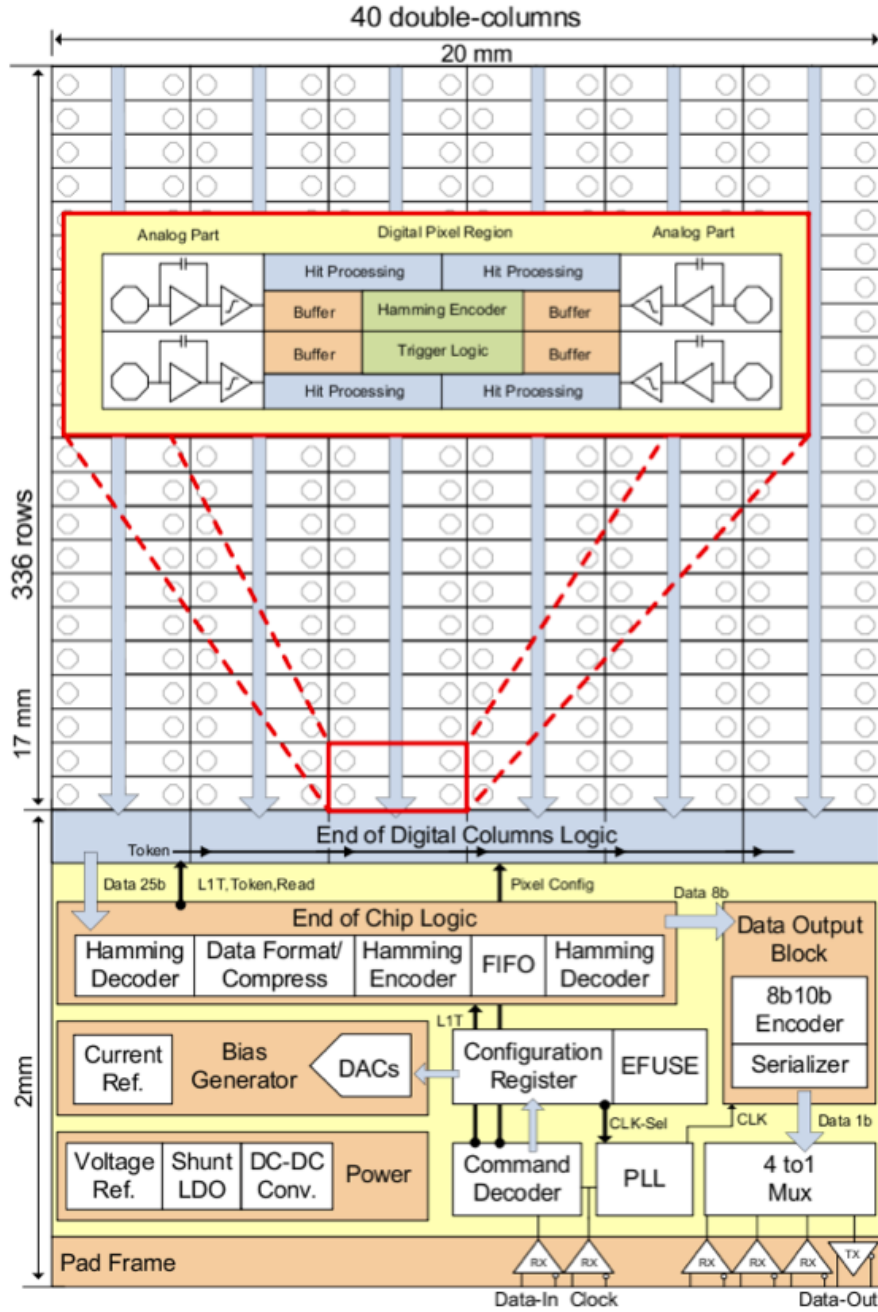


Figure 6.4: Diagram of the FE-I4 functional blocks. The detail shows two rows of a Double Column, a 2×2 pixel array, sharing the Pixel Digital Region [FE-11].

Clock and Command Input

The TTC signals are already decoded by the DORIC chip located on the Optoboard to the reference clock of 40 MHz and the command stream. A Phase Locked Loop (PLL)¹ on the FE-I4 generates 4 clocks available to the FE-I4 (40, 80, 160, and 320 MHz), synchronized to the reference clock. In normal operation the 40 MHz clock is used as the system clock and the 160 MHz clock is used in the Data Output Block (DOB) to generate the 160 Mb/s data stream.

The command stream is received by the command decoder, it is responsible for decoding the commands and control the logical blocks on the FE-I4. The commands are split into two categories, fast commands (listed in Table 6.4) and slow commands.

It has two operational modes, a run mode and configuration (conf) mode. In run

Table 6.4: FE-I4 trigger and fast commands [FE-11].

Name	Field 1 5 bits	Field 2 4 bits	Description
LV1	11101 ²	-	Level 1 trigger
BCR	10110	0001	Bunch counter reset
ECR	10110	0010	Event counter reset
CAL	10110	0100	Calibration pulse
Slow	10110	1000	Slow command header

mode only fast commands are processed and in conf mode only slow commands are processed. Slow commands are identified by the slow command header, followed by the particular slow command (listed in Table 6.5). Slow commands are used to

Table 6.5: FE-I4 slow commands [FE-11].

Name	Field 3 4 bit	Field 4 4 bit	Field 5 6 bit	Field 6	Description
RdRegister	0001	ChipId	Address	-	Read addressed global register
WrRegister	0010	ChipId	Address	16 bit Data	Write addressed global register
WrFrontEnd	0100	ChipId	xxxxxxx	672 bit Data	Write data to shift register
GlobalReset	1000	ChipId	-	-	Reset chip to its idle state
GlobalPulse	1001	ChipId	Width	-	Pulse with given width
RunMode	1010	ChipId	Mode	-	000111 = conf mode 111000 = run mode

¹A common circuit to generate different clock frequencies from one reference clock.

²Single bitflip tolerant.

configure the FE-I4 for operation or calibration scans.

Data Output

The analog part of the pixel readout cell is comparable to the one of the FE-I3. The injected charge is amplified and converted into a digital signal, whereby the measured quantity is the ToT (Time over Threshold). The Pixel Digital Region (PDR) is an array of 2×2 pixel readout cells and can store the ToTs of up to 5 “events” in this array with a resolution of 25 ns, originating in the 40 MHz system clock. When a trigger is received, the end of chip logic distributes a trigger token to all double columns and the hits matching the timestamp of the trigger are read out. The end of chip logic will then format the data and feed it into the Data Output Block (DOB), which performs the 8b10b encoding [AXW83] and serialization.

In 8b10b encoding an 8 bit word is encoded into a 10 bit word. 8b10b encoding follows a scheme known as running disparity, an encoded data word has either a positive disparity ($6 \times “1”$ and $4 \times “0”$) or negative disparity ($6 \times “0”$ and $4 \times “1”$). The disparity changes by ± 2 with every 10 bit word always trying to equal the running disparity to 0. Thus the 8b10b encoded signal is a DC-balanced waveform in the long term ratio. The 8b10b encoding has several advantages over the old NRZ encoding. First of all, an 8b10b encoded signal is DC-balanced, with at least 2 edges per 10 bit word. It allows clock reconstruction, several error identification methods and the use of control words (k-word). Control words are special 8b10b encoded words, which cannot be encoded out of normal 8 bit data words (d-word) and are used for low level control functions of the link.

The FE-I4 uses these k-words for three transmission states, Idle, Start Of Frame (SOF) and End Of Frame (EOF). This allows the readout an easy synchronization to the data stream and a clear identification of the start and end of a frame. The FE-I4 protocol defines 5 types of 24 bits record words listed in Table 6.6 and a sequence of three consecutive Idle words. These record words are used in four valid sequences (framed):

- Pixel Data:
SOF - $1 \times \text{DH}$ - 0 or $n \times \text{DR}$ - 0 or $1 \times \text{SR}$ - EOF
- Configuration (with address read back):
SOF - 1 or $n \times (\text{AR}, \text{VR})$ - EOF
- Configuration (with no address read back):
SOF - 1 or $n \times \text{VR}$ - EOF
- Service message:
SOF - 1 or $n \times \text{SR}$ - EOF

The sequences are sorted by their priority. The configuration sequence is either with or without address read back, controlled by a FE-I4 register. If no data, configuration or service message are transmitted, the DOB will output the Idle word.

Table 6.6: FE-I4 protocol record words [FE-11].

Name	Field 1	Field 2	Field 3	Field 4	Field 5
Data Header (DH)	11101	001	Flag	LV1ID [6:0]	BCID [7:0]
Data Record (DR)	Column [6:0]	Row [8:0]	ToT1 [3:0]	TOT2 [3:0]	
Address Record (AR)	11101	010	Type	Address [14:0]	
Value Record (VR)	11101	100	Value [15:0]		
Servie Record (SR)	11101	111	Code [5:0]	Number [9:0]	

6.3 The IBL Data Acquisition

In order to fit the readout to the new FE-I4 and IBL modularity, parts of the Pixel DAQ system are completely redesigned or updated. Figure 6.5 shows a schematic of the IBL readout system. The new IBL module will receive 1 TTC stream and

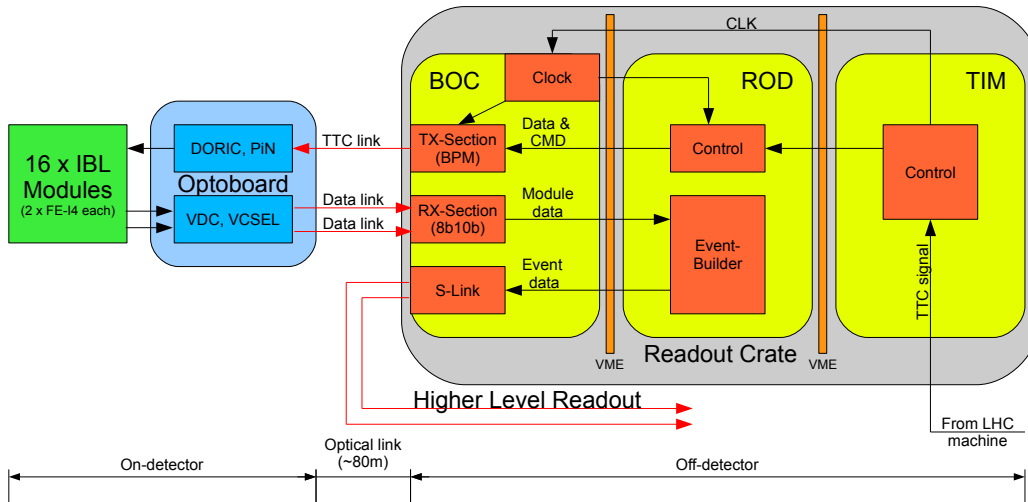


Figure 6.5: Schematic of the IBL readout system showing the connection of the detector to the off-detector components. Red connections represent optical links.

distribute it to the two FE-I4s, with every FE-I4 having its own data stream. The Optoboards used for the IBL are fulfilling the same functionality as the Optoboards used for the current Pixel detector *b*-Layer, but will be connected to a total of 8 IBL modules. Investigations on the VSCEL arrays used in the TX plugins

showed a significantly lower life time than expected. These are the same³ VCSEL arrays as used on the Optoboard and therefore the IBL Optoboards will be moved from the PP0 to a more accessible location on the inner detector endplate, referred to as Optobox.

The off-detector readout electronics, namely BOC and ROD, will receive a major redesign. They will still be placed in a VME crate to conserve compatibility to the current DAQ system. The new IBL-BOC and IBL-ROD have to meet several requirements concerning the readout of the IBL:

- Backward compatibility to the current Pixel detector readout structure
- VME environment, VME bus connected to ROD
- New FE-I4 data protocol
- Reception of a optical 160 Mb/s 8b10b encoded serial data stream per FE-I4
- Transmission of a optical 40 Mb/s BPM encoded TTC stream per IBL module (2×FE-I4)

The IBL-BOC and IBL-ROD will take advantage of the newest FPGA⁴ technology, allowing one hardware concept to be used in versatile manner.

The data flow will principally stay the same. The TIM used in the IBL readout will be the same as in the Pixel detector readout, distributing the LHC clock and trigger commands to the IBL-BOC and IBL-ROD. The IBL-BOC will convert the received optical signal into electrical signals and transmit vice versa. It is responsible for BPM encoding, signal adjustment, 8b10b decoding, data recovery and the optical link to the higher level readout. The IBL-ROD will be the main event processing unit, being fed with data from the IBL-BOC, processing the events, calibration and online histogramming. One IBL-BOC/IBL-ROD pair will be connected to one complete stave and accordingly 16 IBL modules. Resulting in a total of 16 TTC and 32 data links.

³However the VCSEL arrays on the TX plugin are contained in “normal” atmosphere (20 °C, ≈50% rel. humidity), while the VCSEL arrays on the Optoboard are contained in a very dry and very cold (-10 °C, ≈0% rel. humidity) atmosphere. No VCSEL failure of this kind has been observed on a Optoboard yet [Dop10].

⁴Field Programmable Gate Array

Chapter 7

Design of the IBL Back-of-Crate Card

The new IBL-BOC will transfer the experience obtained with the current readout electronics into a versatile readout concept. The use of FPGA technology guarantees the capability to process a massive data rate in parallel, easy prototyping and development and maintenance friendliness.

This chapter will discuss the design and development of the IBL-BOC firmware under the use of an evaluation platform. This allows the firmware and hardware prototype development at the same time.

7.1 Overview

The IBL-BOC requirements are primarily influenced by the FE-I4 and the experience from the current system. Two years of operating the current BOC showed several design flaws, either complicating the readout and calibration or parts of the more than 5 years old hardware are outdated.

7.1.1 Requirements

The IBL-BOC is designed in arrangement with the other components in the readout chain, mainly the FE-I4 and IBL-ROD, as the IBL-BOC is directly connected to these components. The FE-I4 defines the protocol and bandwidth used on the optical link:

- TTC link: BPM encoded, 40 MHz clock, 40 Mb/s commands, variable delay, MSR adjusted (50%)
- Data link: 8b10b encoded, 160 Mb/s data, FE-I4 data protocol

The IBL-BOC is connected through the VME backplane with IBL-ROD, therefore transmission of data and protocols between the two cards have to be matched properly.

Another requirement to the VME backplane mapping is made by the intention to be backwards compatible to the old ROD:

- Signal standards of the old ROD (LVTTTL 3.3¹)
- Communication via the Setup Bus
- S-Link channel
- $32 \times$ TTC signals
- 96 bits wide data bus
- LHC clock received by the TIM and distributed to the ROD

By using the VME backplane for transmission of data between the IBL-ROD and IBL-BOC, it has to be considered that the VME backplane is capable of a maximum bandwidth determined to 80 Mb/s.

7.1.2 Design Goals

Besides the given requirements, the IBL-BOC design was made with certain goals, either derived from operation of the old BOC or to develop and implement technology for future readout systems.

The current TX plugin combines the data encoding, signal adjustment and laser driver in one ASIC. The behavior of this ASIC complicated the generation of a proper BPM stream. After encoding or applying a delay, the MSR is out of tune and needs to be adjusted. This has to be done for every channel of the detector to guarantee the recovery of a good reference clock. For the IBL-BOC the laser driver and signal encoding/adjustment are disjoint. The signal encoding and adjustment are translated into a firmware block, allowing changes even after production.

A calibration of the Pixel detector optical link is a time consuming procedure, but necessary to assure the flawless data transmission. Responsible is the NRZ encoding and the used hardware. By the use of 8b10b encoding and FPGA based receivers, a self-tuning link is developed. This receiver block will adjust itself to the given data stream and should work right after power up.

The current link to the higher level readout uses the S-Link card produced by the CERN electronics group. As space on the IBL-BOC front panel could be limited due to the use of more interconnection from the IBL-BOC to components not located in the VME crate, the S-Link functionality will be implemented into a firmware block and only the optical module will allocate space on the front panel. The “soft” S-Link is not taken into account in the IBL-BOC firmware design described in this thesis, because it does not influence the remaining logical blocks of the firmware.

The current data monitoring on the ROD is solely focussed on the interpretation of the data, the 8b10b format allows the IBL-BOC to implement monitoring capabilities with focus on the quality of the data stream, error histogramming and

¹Low Voltage Transistor-Transistor-Logic 3.3 V

e.g. interpretation of the errors on the data stream. Overall the IBL-BOC will participate in much more active role in the readout operation, without the need of buffering the data.

The optical components designed for the Pixel detector are specialized for the use in the inner detector environment (radiation hard, lightweight). The optical components designed for industrial use, could be used on the off-detector as an alternative to the custom optical modules. Industrial optical modules bear the advantage of easy supply and tested functionality. It is planned to use industrial optical modules on the IBL-BOC, whereby their characteristics and usefulness in the optical detector readout will be investigated in the IBL-BOC development.

7.1.3 Hardware Prototype

By the time of writing this thesis (09/2011), the IBL-BOC is in its final production stage. The schematics and layout of the card are finished and the first PCBs (shown in Figure 7.1) have been produced. After assembly of the first card, it will be electrically tested and checked for design flaws. When no redesign is required, a total of 3 cards will be produced for the first prototype tests. After these tests the schematic and layout will be adjusted according to the results and if needed undergo a second hardware revision or the production of the first production prototypes will start. The IBL-BOC prototype (shown in Figure 7.2) will use 2 FPGA types [Xil11] for control and data processing purpose. The BOC Controller FPGA (BCF) will be a Xilinx Spartan 6 LX75T, the data processing will be done by 2 BOC Main FPGAs (north/south BMF), which are of the type Xilinx Spartan 6 LX150T. All FPGAs are connected to DDR2 RAM, additionally the BCF has three flash memories used to save the firmware for each FPGA. The BMFs are connected to FMC-LPC² connectors to attach mezzanine cards for testing purposes. Attached to the BCF is a RJ45 connector used for Gigabit Ethernet, connecting the IBL-BOC to components not located in the VME crate.

The IBL-BOC will use a wide variety of optical modules:

- $4 \times$ SFP (each 1 TX, 1 RX) [com09]: $2 \times$ S-Link connected to the higher level readout + $2 \times$ S-Link connected to the Fast Track Trigger³ [A⁺09].
- $1 \times$ QSFP (4 TX, 4 RX) [Inc09]: The same as the 4 SFP modules, but merged into one module.
- $2 \times$ TX plugin (each 12 TX) [GA08]: The TTC link does not change, therefore TX plugins could be used for transmission.
- $2 \times$ Snap12 TX (each 12 TX) [SNA02]: A possible candidate to replace the TX plugin on the off-detector side.
- $4 \times$ Snap12 RX (each 12 RX) [SNA02]: Candidate receiver module, a RX plugin from the current Pixel detector readout would not work the IBL.

²FPGA Mezzanine Connector - Low Pin Count standardized under VITA.57

³A system using tracking for Level 1 trigger decisions. Currently under development.

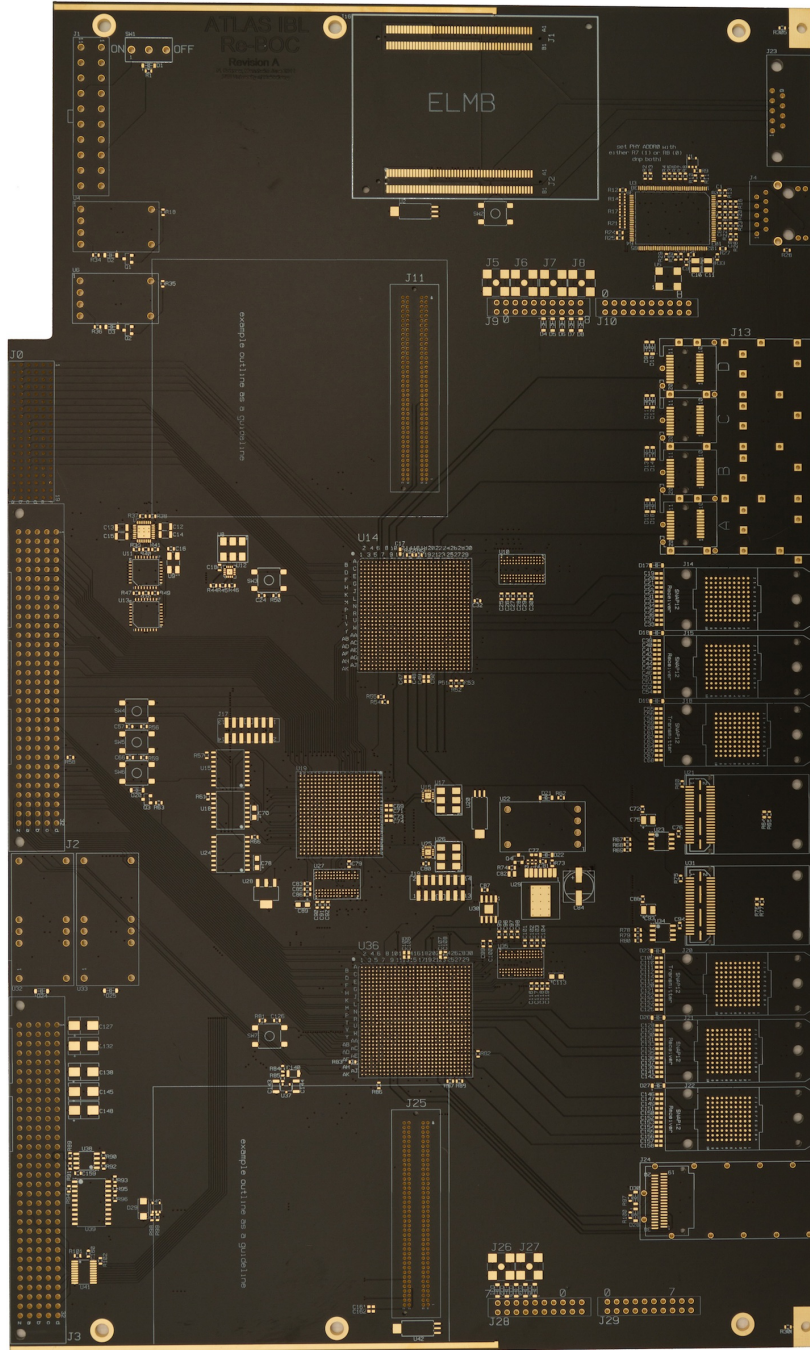


Figure 7.1: Picture of the PCB of the first IBL-BOC prototype.

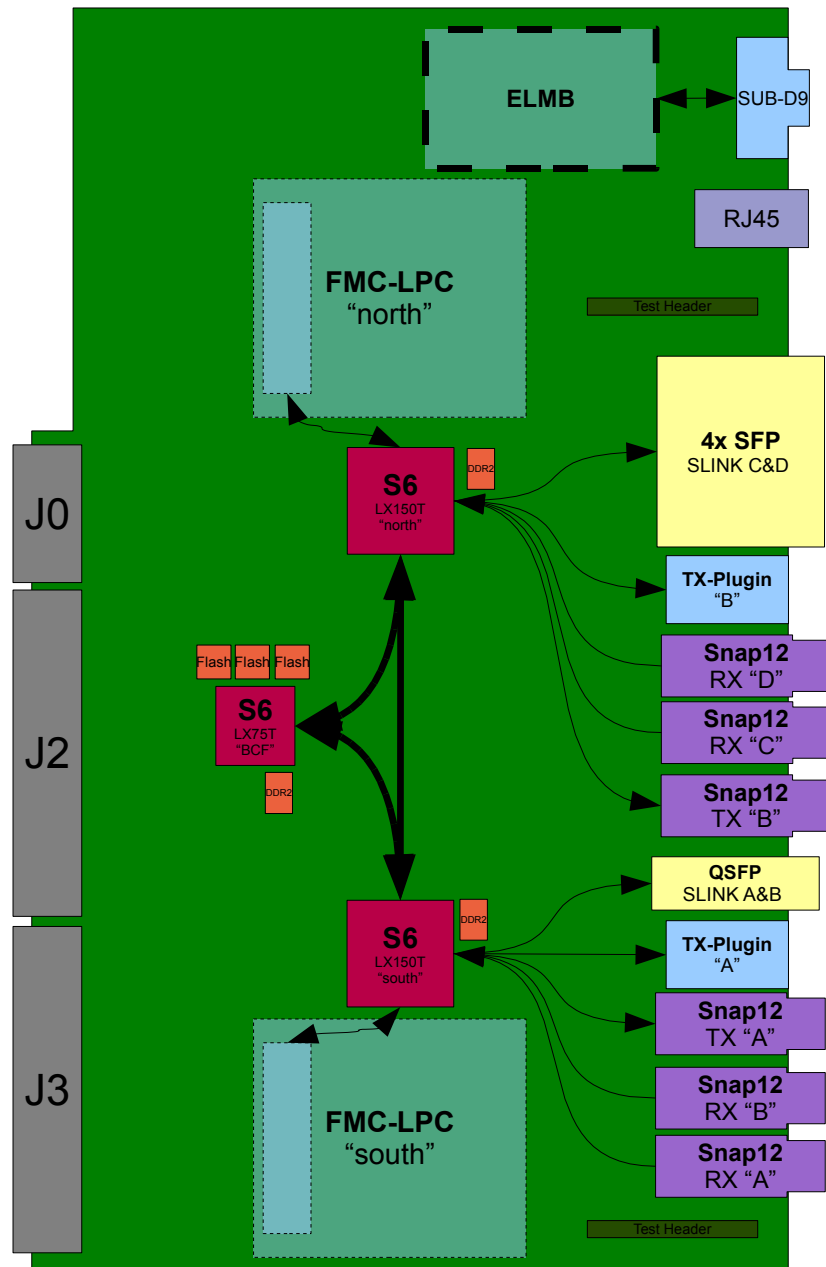


Figure 7.2: Schematic of the IBL-BOC prototype.

After evaluation of all optical modules a decision will be made, which module will be used on the final IBL-BOC. A picture of the industrial optical plugins is shown in Figure 7.3.

Furthermore each IBL-BOC will have an ELMB, responsible for monitoring the



Figure 7.3: 3 types of industrial optical modules, each shown from the electrical connector and the optical connector side. From left to right: SFP, QSFP, Snap12 TX, Snap12 RX.

temperature and power consumption. Hence the IBL-BOC will be integrated into the IBL DCS.

7.2 Firmware Evaluation Platform

The FPGA firmware for the IBL-BOC is developed on a commercial FGPA evaluation platform. This has the advantage of having the FPGA installed on an already tested environment, being able to focus on the firmware development. The IBL-BOC will use Xilinx Spartan 6 FPGAs, thus a FPGA of the same architecture is used for the firmware development.

7.2.1 Spartan 6 Evaluation Board

The Xilinx Spartan 6 SP605 evaluation board [Xil10b] (shown in Figure 7.4) is equipped with a Spartan 6 LX45T FPGA. This FPGA is similar to the FPGAs used on the IBL-BOC, although it does not contain as much logic cells.

The SP605 is equipped with a wide variety of hardware and interfaces to external devices. The firmware described in this thesis uses the following hardware of the SP605:

- USB JTAG: Connected to a PC to program the FPGA.
- USB UART: Connected to a PC for communication.
- FMC-LPC connector: Connected to custom circuit boards equipped with optical modules or breakout connectors.

- 200 MHz Oscillator: Used to generate the 40 MHz system clock.

A firmware designed and tested on the SP605, is expected to work in the same manner on the IBL-BOC prototype. This is first and foremost valid for the logic deployed inside of the FPGA.

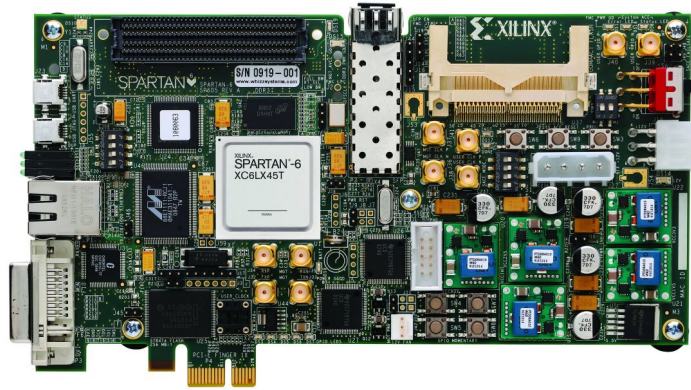


Figure 7.4: Picture of the Xilinx SP605 evaluation board [Xil].

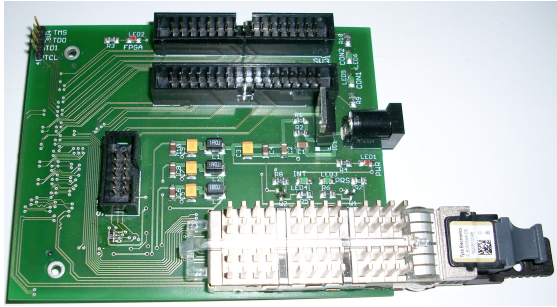
7.2.2 Mezzanine Daughter Boards

Two different mezzanine daughter boards have been designed and produced for the firmware development. One (shown in Figure 7.5.1) is equipped with two easily accessible 34-pin Harting connectors (connected to a total of 30 differential pins), a QSFP module connector and a 3.3 V power regulator for the QSFP module. Allowing measurements with an oscilloscope or logic analyzer with easy access to the signals. A 4 channel optical transceiver module (QSFP) is used for evaluation or loopback measurement with transmission over an optical link.

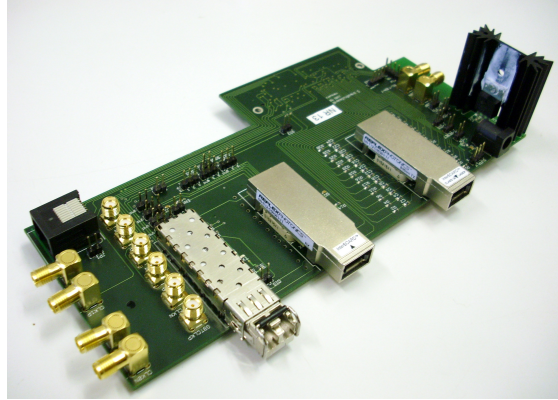
Another mezzanine daughter board (shown in Figure 7.5.2) is equipped with one Snap 12 TX socket and one Snap 12 RX socket, an RJ-45 connector with custom pin assignment and multiple SMA connectors. This board is used to mimic the optical transmission and reception of the real IBL-BOC, although with a smaller number of channels. Furthermore the RJ-45 connector is used to communicate to a real FE-I4 installed on a support card.

7.3 Logical Blocks Overview

The IBL-BOC firmware is divided into three main functions (shown in Figure 7.6): Control, TX path and RX path. The arrangement of these functions is basically the



7.5.1: Board carrying the QSFP connector and power supply, as well as two 34 pin breakout connectors.



7.5.2: Board carrying two Snap12 and one SFP connector and their power supply. Various pins are also routed to SMA connectors.

Figure 7.5: Pictures of the two mezzanine daughterboards designed for the SP605.

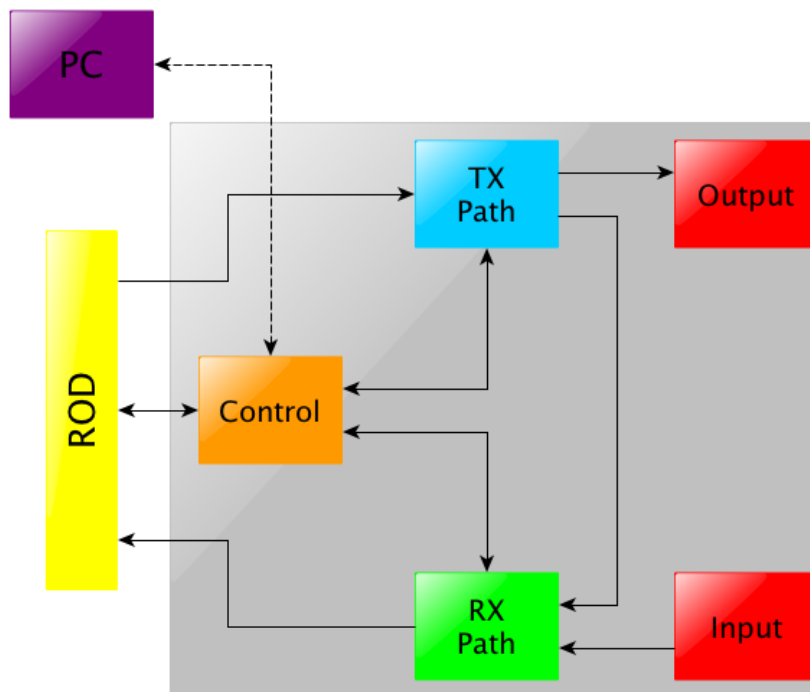


Figure 7.6: Coarse block diagram of the firmware logic blocks. Indicating the connection between the three main sections and external devices.

same as on the current BOC. The TX path is in charge of the BPM encoding and signal adjustment of the TTC stream received from the ROD. The data from the IBL modules is received by the RX path, the internal logic of this block will strongly differ from the one on the current BOC, due to the higher data rate and 8b10b encoding. However in similarity to the current system, it will act solely passive on the data stream and prepare it for the processing on the ROD. The IBL-BOC is controlled from the ROD via the Setup Bus. To distribute the controlling functions onto multiple FPGAs the IBL-BOC contains a internal system bus. For the standalone tests on the SP605 the firmware can also be controlled by a PC interfacing the inter system bus.

The logical blocks of the firmware are written in a programming language called VHDL⁴. VHDL is a hardware description language and does not work like normal computer programming language. In contrast to programs written on the PC, VHDL code describes the connection and behavior of a digital electronic circuit. A program (Xilinx ISE) translates the VHDL code into the electronic circuit and takes care of optimizing the routing and placement in the FPGA. The result of this “synthetization” is a bitstream that programs the firmware and its logic into the FPGA.

7.4 Communication Bus

The main communication path to the IBL-BOC is via the Setup Bus from the ROD. The Setup Bus is an asynchronous parallel bus, with the ROD acting as the initiator and the BOC being the target. It is capable of the basic single read and write cycles. The Setup Bus protocol is defined in the Finite State Machine (FSM) shown in Figure 7.7⁵. The Setup Bus has a 10 bit wide address and a bidirectional 8 bit wide data bus. It is controlled by 3 signals, the Strobe and Write signal from the ROD and the Busy signal from the BOC. In the Idle state of the ROD and BOC, all three signals are negated. A write cycle will be done in 3 phases and a read cycle in 4 phases. These phases and their signal transitions are shown in Figure 7.8. The Setup Bus is not optimized for high bandwidth, which is also not required on the communication path. In this bus concept the memory on the BOC is easily accessible through the VME bus and can be considered as a virtual memory of the ROD. An access to the BOC over the VME bus is simply utilized by the ROD, because the Setup Bus is a reduced form of the VME bus.

In the IBL-BOC firmware the internal communication path is realized as shown in Figure 7.9. The Setup Bus is bridged to the internal Wishbone Bus [Ope07]. Splitting the external interface from the internal bus allows communication without control of the ROD. The Wishbone Bus specification does not define a data and address bus width, endianness or bandwidth.

⁴Very High Speed Integrated Circuit Hardware Description Language.

⁵A signal name having a “!” in front of it, refers to a logical low state of the signal. Hence a signal without it, refers to logical high state.

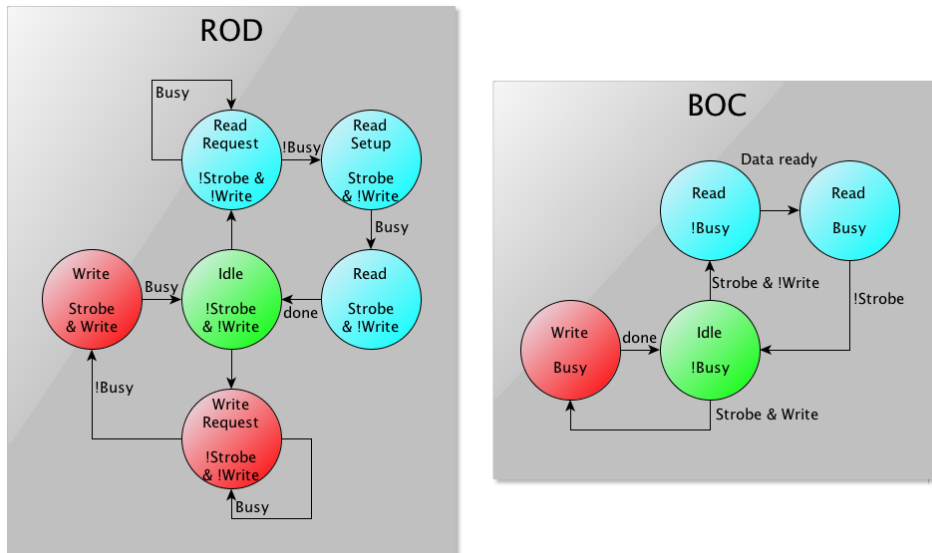
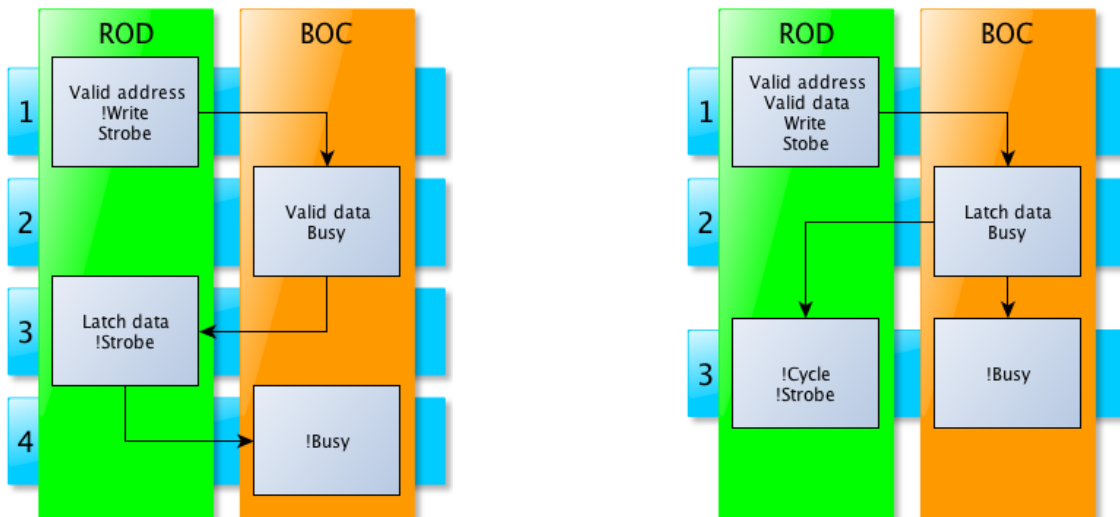


Figure 7.7: Diagram of the Setup Bus FSM in ROD and BOC perspective. The Idle state is shown in green, a write cycle in red and a read cycle in light blue.



7.8.1: Single read cycle.

7.8.2: Single write cycle.

Figure 7.8: Flowchart of the handshake between the ROD and the BOC for a single read and single write cycle.

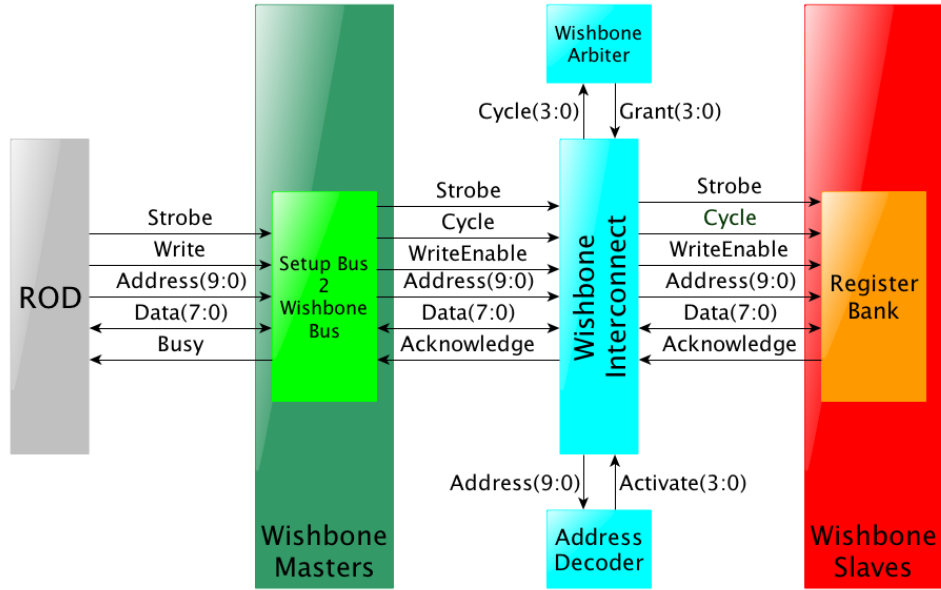


Figure 7.9: Schematic of the logical block providing communication via the internal Wishbone Bus. The shown master of the bus is a Setup Bus to Wishbone Bus bridge. As an exemplary slave a register bank is connected to the bus.

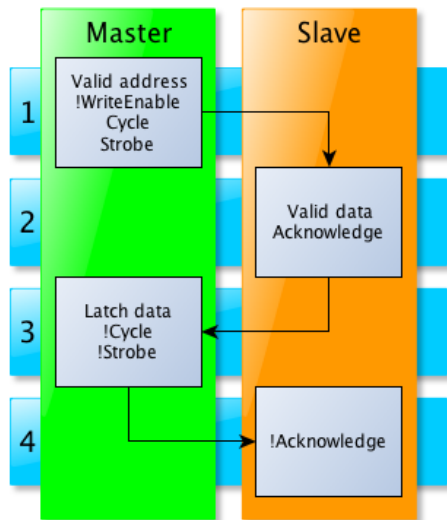
Therefore, the Setup Bus specifications are adopted by the Wishbone Bus on the IBL-BOC:

- 40 MHz clock
- 10 bit address bus
- 8 bit data bus
- Single 8 bit read and write cycles

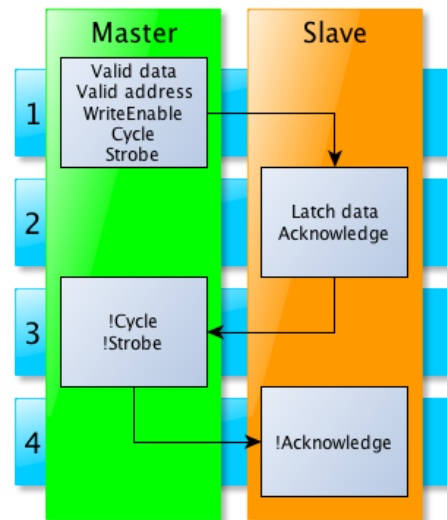
However the Wishbone Bus specification does define a handshake protocol between the master (initiator) and slave (target). The flowchart in Figure 7.10 shows the four phases of a single read and single write cycle.

A point to point connection or one master connected to one slave, respectively, is done by simply connecting the master directly to the slave. For the IBL-BOC the Wishbone Bus should be shared by multiple masters and slaves, for this purpose a logical block interconnecting the different masters and slaves is needed. If a master requests the bus by asserting the Cycle signal, the Wishbone Arbiter decides to which master the bus is granted. If no priority is associated with the different masters, the easiest way to share the bus is a round-robin arbiter. It grants the bus to the masters requesting it in a round-robin method. The Address Decoder knows which slave is associated with the valid address presented by the master. With the information from the Wishbone Arbiter and Address Decoder the Wishbone Interconnect will establish a point to point connection between the master owning the bus and the requested slave.

The Wishbone Interconnect offers the possibility to communicate with the IBL-BOC not only via the Setup Bus or the VME environment, but rather by attaching another communication master to the internal communication bus. In the firmware test system this is used for a communication via a UART⁶ interface to a PC. The Wishbone Interconnect and its peripherals will be located in the BCF. The two BMFs will be connected to the Wishbone Bus as slaves.



7.10.1: Single read cycle.



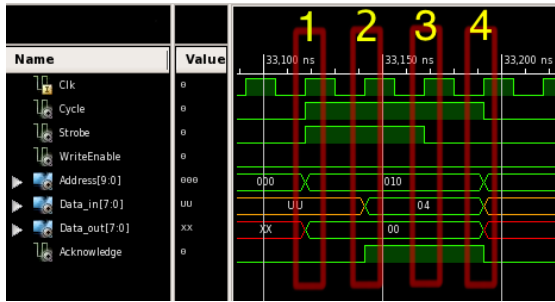
7.10.2: Single write cycle.

Figure 7.10: Flowchart of the handshake between master and slave for a single read and single write cycle.

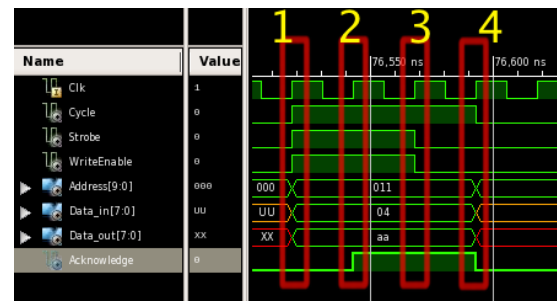
7.4.1 Simulation of the Internal System Bus

A simulation of the complete internal system bus is shown in Figure 7.11. Shown are a read and write cycle. Highlighted are the transitions for each phase of the cycle (in reference to Figure 7.10). The first phase marks the begin of the cycle initiated from the master. The second phase is the response from the slave, either the data given from the master was latched (write cycle) or valid data is located on the bus (read cycle). In the third phase the master completes the cycle and on the fourth phase the slave responds to the completion. The transition of the Cycle and Strobe signal are delayed by one clock cycle, because the Wishbone Arbiter needs one clock cycle to react on a request from the master. After the Wishbone Arbiter grants the bus to a master, a point to point connection is established to the slave, this occurs in the first highlighted field. The Strobe signal of the point to point connection is directly connected to the Strobe signal of the master and therefore negated in the third field. The Cycle signal is handled by the Wishbone Arbiter, thus the Cycle signal of the point to point connection is delayed by one clock cycle and negated in

⁶Universal Asynchronous Receiver Transmitter. A serial interface formerly known as RS-232.



7.11.1: Single read cycle.



7.11.2: Single write cycle.

Figure 7.11: Simulation of a read and write request handled by the Wishbone Bus.

the fourth field.

The Cycle signal is therefore associated to a master owning the bus, while the Strobe signal is used for the handshake with the slave.

7.4.2 Simulation of the Setup Bus

The simulation in Figure 7.12 shows a read and write cycle via the Setup Bus. Also

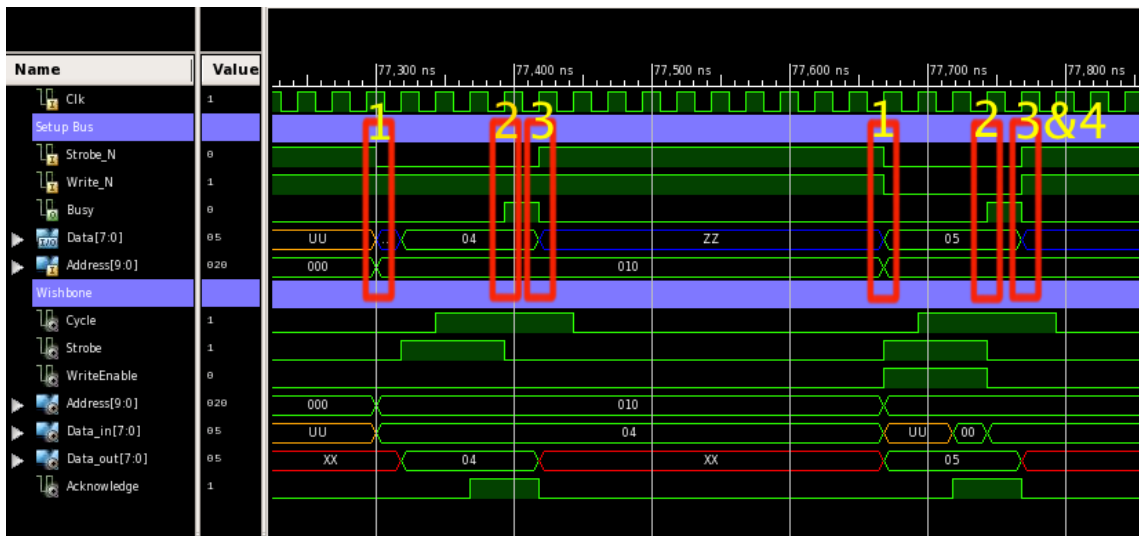


Figure 7.12: Simulation of a read and write cycle from the Setup Bus.

shown are the signals of the Wishbone Bus, because a read/write request on the Setup Bus is bridged to the internal Wishbone Bus. Highlighted are the transitions for each phase of the cycle (in reference to Figure 7.8), the first cycle is a read request and the second cycle a write request⁷. It should be noted that the physical Strobe and Write signals of the Setup Bus are active low (Strobe_N and Write_N). A

⁷Phase 3 and 4 of the write cycle occur at the same time. This is because the simulation mimicking the ROD is not 100% realistic for the ROD part. However this does not affect the behavior of the logical block on the IBL-BOC.

adjustment to the Wishbone Bus behavior was made in the Setup Bus to Wishbone Bus bridge, because normally the Cycle signal of the Wishbone Bus would be negated after the slave responds with the Acknowledge signal. The Wishbone Arbiter would now free the bus for access by other masters. But a complete Setup Bus is not completed until the Setup Bus Strobe signal is negated. Therefore, the Wishbone Bus Cycle signal is asserted until the ROD marks the end of the cycle. This is why the Cycle Signal in the simulation is kept high for one more clock cycle, compared to the sole Wishbone Bus simulation in Figure 7.11.

The response of the BOC to a request from the ROD is delayed by two clock cycles (50 ns) by bridging the Setup Bus to the Wishbone Bus. This delay does not affect the communication, because a high bandwidth is not needed.

7.4.3 Communicating via the Setup Bus

To do initial tests on the firmware implementation of the Setup Bus, the VHDL code of the corresponding counterpart on the ROD was used. It was implemented on another evaluation board referred to as Fake-ROD and was framed in a minimal design allowing the connection to the IBL-BOC. The Fake-ROD was solely capable of performing read and write requests via the Setup Bus. A measurement with a logic analyzer of a read and write cycle is shown in Figure 7.13. This proves the basic functionality of the firmware responsible for the communication via the Setup Bus and the internal system bus.

7.4.4 Communication to a PC

To control the firmware on the SP605 in a standalone manner (without the ROD), besides the “Setup Bus to Wishbone Bus” master block another master block is attached to the Wishbone Bus, the “UART to Wishbone Bus” block. It contains a minimalistic logic using the UART interface to perform single read and write cycles on the Wishbone Bus from a PC.

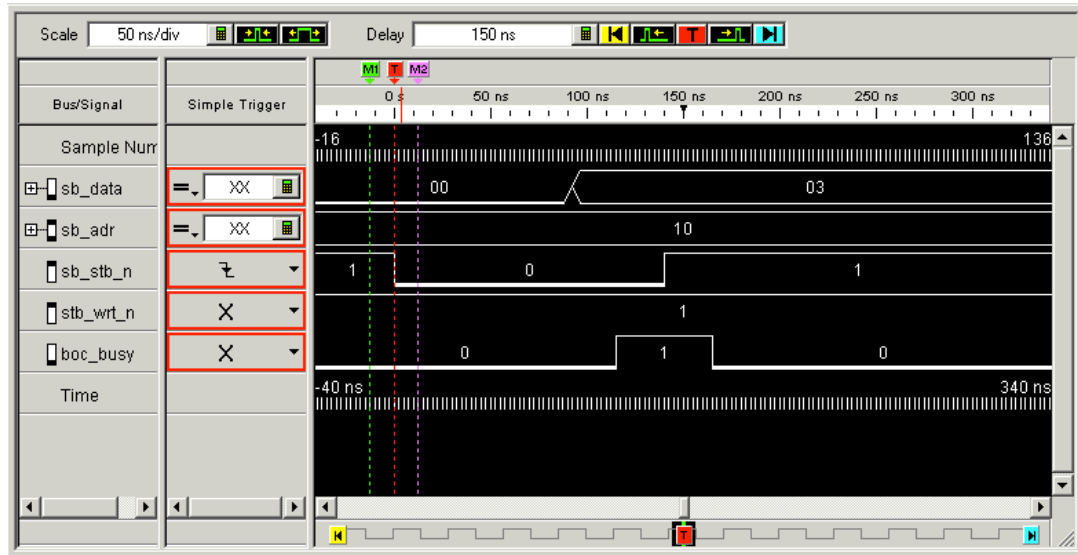
UART Interface

The UART Interface uses one serial transmission and reception stream. On the SP605 these two signals are fed into an IC⁸ translating them to a USB⁹ interface. Therefore, the hardware connection to the PC is done via USB, however the USB interface is hidden to the firmware and software on the PC. In the UART protocol the data transmitted and received is formatted in packets of 8 bits framed with one start bit and one stop bit. The bandwidth was chosen to be 921.6 kb/s¹⁰. The data received from the PC is formatted in a custom protocol listed in Table 7.1.

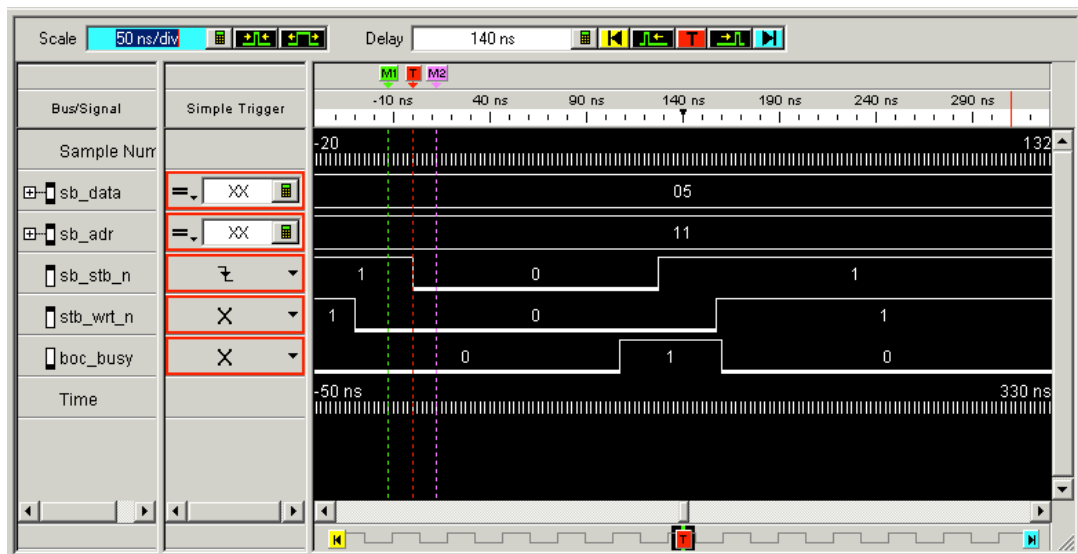
⁸Integrated circuit

⁹Universal Serial Bus

¹⁰The highest possible bandwidth for the IC.



7.13.1: Single read cycle



7.13.2: Single write cycle.

Figure 7.13: logic analyzer measurement of the communication via the Setup Bus with the Fake-ROD.

Table 7.1: Format of the data send by the PC via the UART interface.

Byte Nr.	Format	Description
1	RW(0:1) [0:1]	RW: Read or write cycle
1	Payload(0:3) [2:5]	Payload: Number of data frames
1	Address(9:8) [6:7]	Address(9:8): First part of the address
2	Address(7:0) [7:0]	Address(7:0): Second part of the address
3	Data(7:0) [7:0]	Data(7:0): Missing if read cycle
3	00111100 [7:0]	End of Frame

The data is decoded and on reception of a full frame, a FSM drives the translation to the Wishbone Bus. In case of a read cycle, the data is send back to the PC in a raw format.

Proof of Principle

To prove the functionality of the communication via the UART interface, a simple program was designed. It opens the virtual port on the PC for the hardware USB interface and performs the data transmission outlined above to read the current firmware version:

```
Opening port to BOC... done!
Writing to port: 0x01
Writing to port: 0x10
Writing to port: 0x3c
Read from port: 0x04
BOC Firmware: 0x4
```

This is just a proof of principle, in Section 7.7 the whole software concept is described in more detail.

7.5 TX path

The TX path is responsible for the transmitted serial stream to the detector. While operation of the detector, the TTC signals are received from the ROD and the IBL-BOC acts solely passive. A multiplexer was implemented at the beginning of the TX path to enable the IBL-BOC to actively use the TX path. Figure 7.14 shows a block diagram of the TX path. The BPM encoder, MSR and FineDelay adjustment blocks used are based on a preliminary study [Neu11]. To actively inject data onto

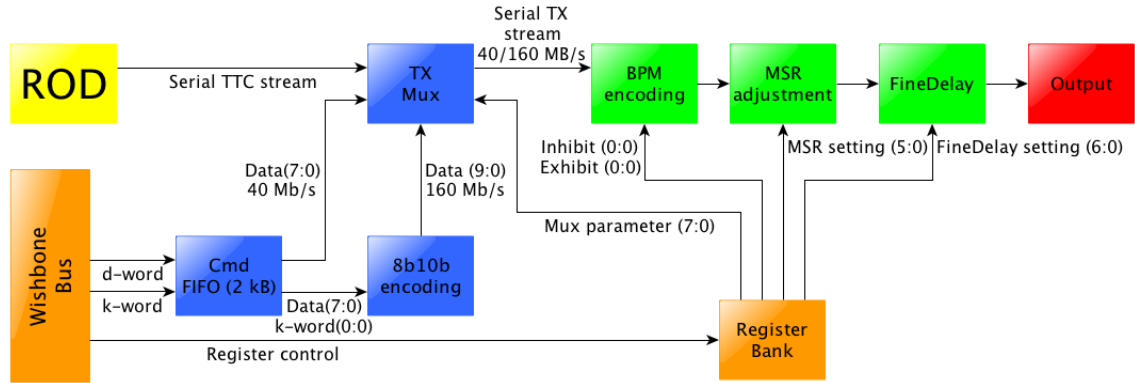


Figure 7.14: Schematic of the logical blocks used for the TX path, consisting of the main encoding and signal adjustment block (green), a testing infrastructure (blue) connected to the Wishbone Bus (orange) and the output buffer of the FPGA (red).

the TX path, a multiplexer, the TX Mux, is placed in front of the chain. The list of control parameters for the TX path are given in Table 7.2. The TX path can be configured to operate in three modes:

- Operation: The 40 Mb/s TTC stream from the BOC is transmitted.
- Standalone TTC: A 40 Mb/s TTC stream generated on the IBL-BOC is transmitted.
- Loopback: A 160 Mb/s 8b10b encoded stream generated on the IBL-BOC is transmitted.

Obviously the Standalone TTC and Loopback mode are only needed for testing purposes. The data to be transmitted is written into the CMD FIFO¹¹ via the Wishbone Bus. The CMD FIFO is addressed in two ways, one being used for raw 8 bit data and one is used for the generation of k-words in the 8b10b encoding, but the order of how the data is written into the CMD FIFO is contained.

In the Standalone TTC mode the 8 bit data is read out from the CMD FIFO, serialized into a 40 Mb/s TTC stream and injected into the TX path. In the Loopback mode the d- and k-words are read out from the CMD FIFO, 8b10b encoded, serialized into a 160 Mb/s data stream and injected into the TX path. Of course the BPM encoding should be bypassed in the Loopback Mode, but the use of the MSR and Delay adjustment is used for testing.

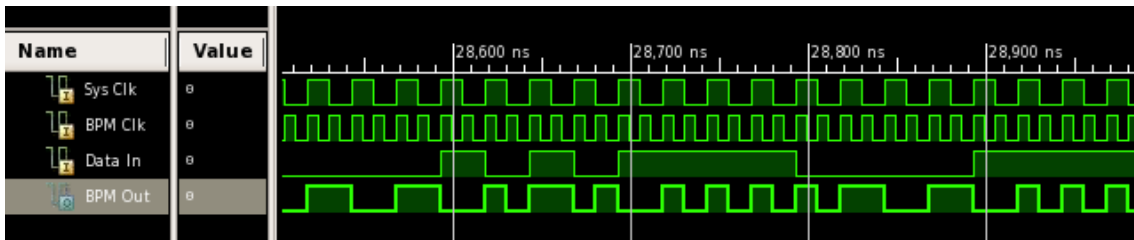
7.5.1 BPM Encoding

The BPM encoding is heavily simplified by the use of an 80 MHz clock. A simulation of the BPM encoder block is shown in Figure 7.15. The data input is sampled on the rising of the 80 MHz. When the data input is high, the output transitions on

¹¹First In First Out memory.

Table 7.2: Control parameters of the TX path.

Parameter	Range (in DAC settings)	Comment
MSR	0 - 31	$\approx \pm 5$ ns around the 50% setting
FineDelay	0 - 127	step size ≈ 0.3 ns
Inhibit	0 or 1	1 = only clock is included into the BPM stream
Exhibit	0 or 1	1 = BPM encoding is bypassed
TTC mux	0 or 1	1 = 40 Mb/s TTC stream from the BOC
RX mux	0 or 1	1 = 160 Mb/s 8b10b stream from the BOC
Drain	0 or 1	1 = CMD FIFO is drained
Repeat	0 or 1	1 = Content of CMD FIFO is repeated
Mux Enable	0 or 1	1 = CMD FIFO content is injected onto the TX Path
Loop Enable	0 or 1	1 = The TX path output is looped back internally onto the RX path

**Figure 7.15:** Simulation of the BPM encoding by the use of a 80 MHz clock. Shown are the 40 MHz system clock, 80 MHz BPM clock, the data input and the BPM encoded output (from top to bottom).

every rising edge of the 80 MHz clock, in contrast to the data input being low, the output transitions with every second rising edge of the 80 MHz clock and accordingly every rising edge of the 40 MHz clock. An oscilloscope measurement is shown in Figure 7.16. The BPM encoded output is delayed by two 80 MHz clock cycles, one

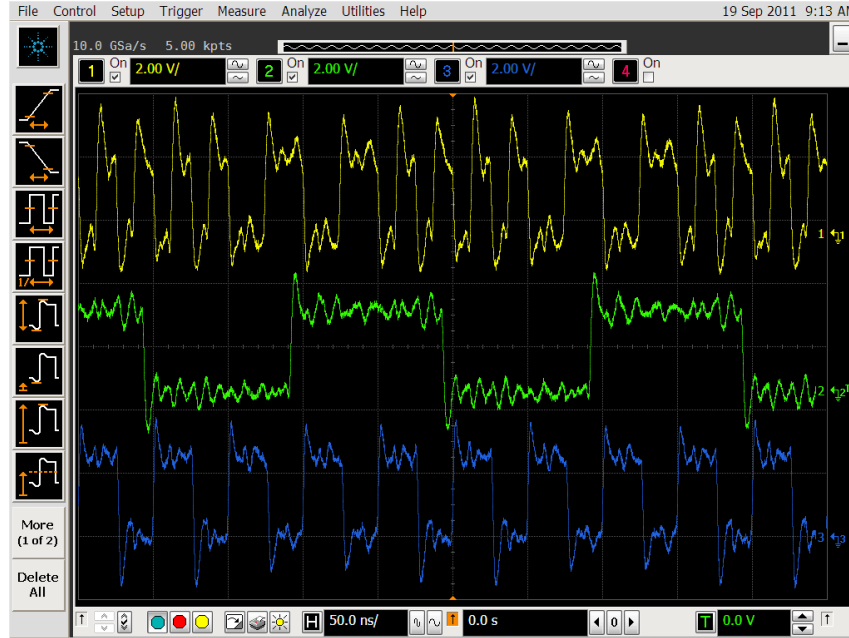


Figure 7.16: Oscilloscope measurement of the BPM encoding, the BPM encoded signal is shown in yellow, the data input in green and the BPM encoded without data input in blue (20 MHz clock).

clock cycle delay is needed to detect a transition of the data input and another clock cycle is used for resynchronization through a Flip-Flop. The raw unadjusted BPM encoded output has a duty cycle of 50.6% ($\pm 1\%$), thus a MSR adjustment is principally not needed.

7.5.2 Variable Delay

On the one hand a variable delay of the TTC stream is needed to adjust for the different cable lengths to the IBL modules, on the other hand this variable delay is used to adjust the MSR. A delay is applied by exploiting the run time of the signal through a multiplexer element. By chaining multiple multiplexers one after another a variable delay can be applied to the signal, this is often referred to as a carry chain. E.g. for the FineDelay setting 128 multiplexers are used in a carry chain. The data flow of the MSR adjustment and FineDelay is shown in Figure 7.17. In the first step the MSR is adjusted by either shortening the pulse or broadening it by two logical operation onto two differently delayed signals. The variable delay of the pulse shortening corresponds to the MSR setting range of 0 to 15, hence the variable delay of the pulse broadening corresponds to the range of 17 to 31. In principal the setting 16 should correspond to no change, because the two delays in the pulse

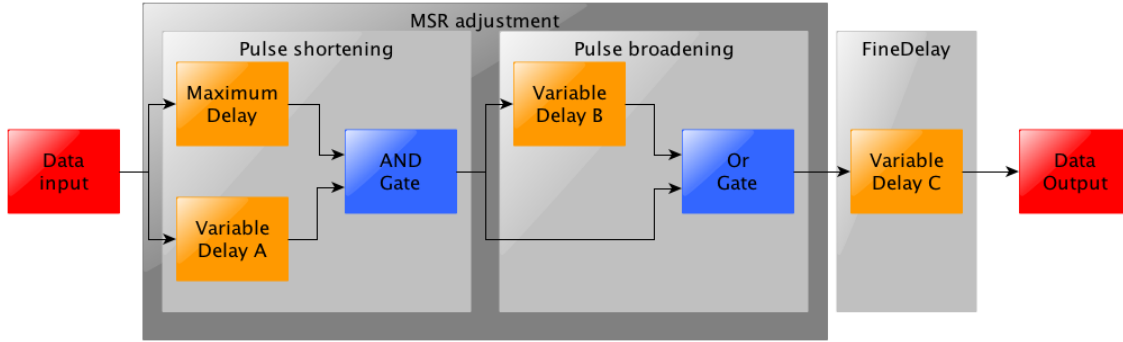


Figure 7.17: Block diagram of the MSR adjustment and FineDelay block. The Input and Output are shown in red, the carry chain delay blocks in orange and logical gates in blue.

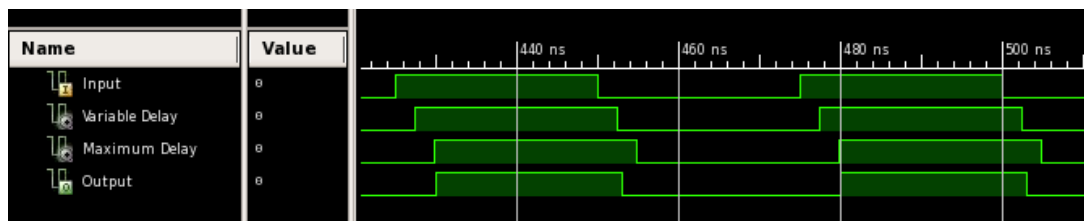
shortening block are the same and the delay of the pulse broadening block is 0. The utilization of the FineDelay is a straight forward variable delay of the signal. A simulation of this process is shown in Figure 7.18, although the simulation works with a predefined arbitrary delay per multiplexer element. The shortened pulse has by design a fixed delay and therefore the rising edge is phase stable. This is the base line delay introduced by the MSR adjustment. The input of the FineDelay block in Figure 7.18.3 is the output of the MSR adjustment block, hence the base line delay must be added to the output of the FineDelay block to determine the overall delay of the signal.

As pointed out in the preliminary study the delay steps per setting are sensitive to the placement in the FPGA. Since a linear behavior is desired for the applicable delay, the placement of each multiplexer element has to be optimized. In an first attempt this was done by manually placing each element, but this is only possible for small number of elements. Therefore, the placement was automatized using a relative location constraint (RLOC), defining the placement of each element in the carry chain, but not the absolute position of the carry chain itself. The simplest and best placement would be a long chain, but these very long chains would get to big to be implemented in the FPGA. The most homogeneous but still possible design, was achieved for rows of 3 elements, derived from the following formula giving the relative X and Y coordinate for each element i :

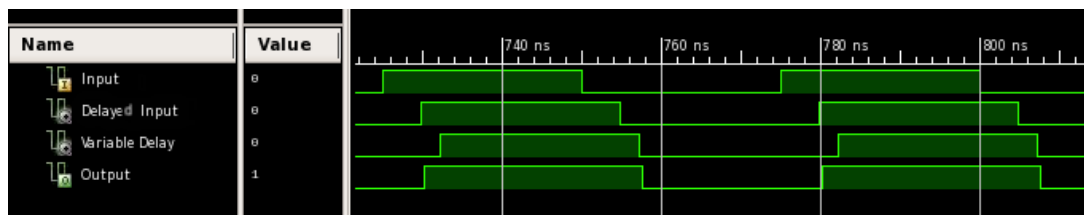
$$\begin{aligned} X &= i/3 \\ Y &= (i/2) \bmod 3 \end{aligned}$$

Only integer values of X and Y are allowed. In each coordinate a total of 2 elements can be placed.

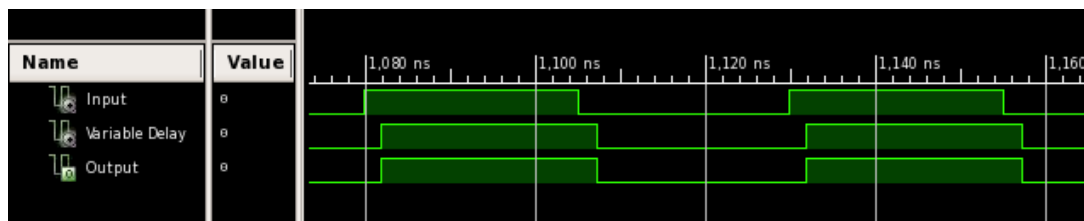
The impact of the placement optimization on the delay is shown in Figure 7.19. Shown are two measurements of the positive pulsewidth in dependence of the MSR setting with and without optimization. The placement optimization cleans the distribution from plateaus and or very big steps. The steps size per MSR setting or the slope respectively is determined to be 339 ps (± 4 ps), resulting in an overall range



7.18.1: MSR adjustment: Pulse shortening.



7.18.2: MSR adjustment: Pulse broadening.



7.18.3: FineDelay block.

Figure 7.18: Simulation of the MSR and FineDelay adjustment.

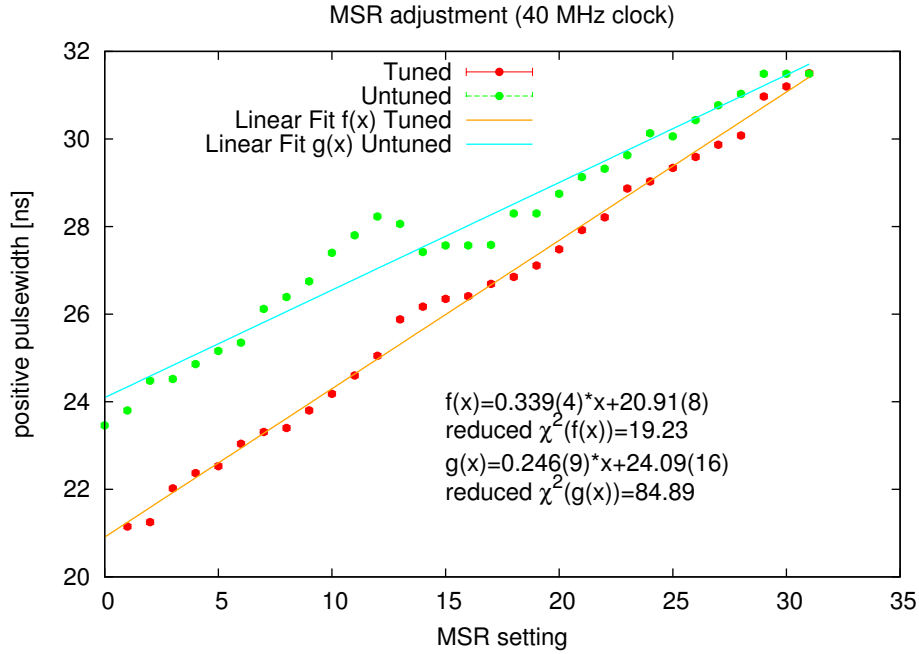


Figure 7.19: Measurement of the positive pulsewidth in dependence of the MSR setting before (green) and after (red) placement optimization.

of about ± 5 ns around the 50% setting.

Figure 7.20 shows the measurement of the delay and duty cycle in dependence of the FineDelay setting. The step size or rather the slope is determined to 275 ps (± 3 ps), thus an overall range of around 34 ns delay can be achieved. This covers the required 25 ns with enough overhead. Also shown is a rising duty cycle with higher delay (≈ 50 ps per FineDelay setting), which is a not desired side effect. This complicates the tuning of the TTC link, because every change in the delay has to be adjusted for the MSR.

The CoarseDelay is simply realized by shifting the signal through a chain of Flip-Flops, in this case the Flip-Flops are toggled with the 40 MHz clock. Thereby a coarse delay of 25 ns per setting is achieved.

7.5.3 Alternative Concept

As stated above, the output of the BPM encoder has a nearly perfect duty cycle, because it is synchronized by the internal 80 MHz clock. Thus a MSR adjustment is not needed. But the realization of a variable delay with carry chains introduces a change in the duty cycle. As an alternative concept the already built-in variable delays of the Spartan 6 could be used.

Each input/output buffer of the Spartan 6 has an IODELAY2 [Xil10a] block built-in. These blocks use ring oscillators to apply a delay in the region of 50 ps¹² per step. The realization of these IODELAY2 block was done in a preliminary study

¹²Depending on the speed grade of the FPGA.

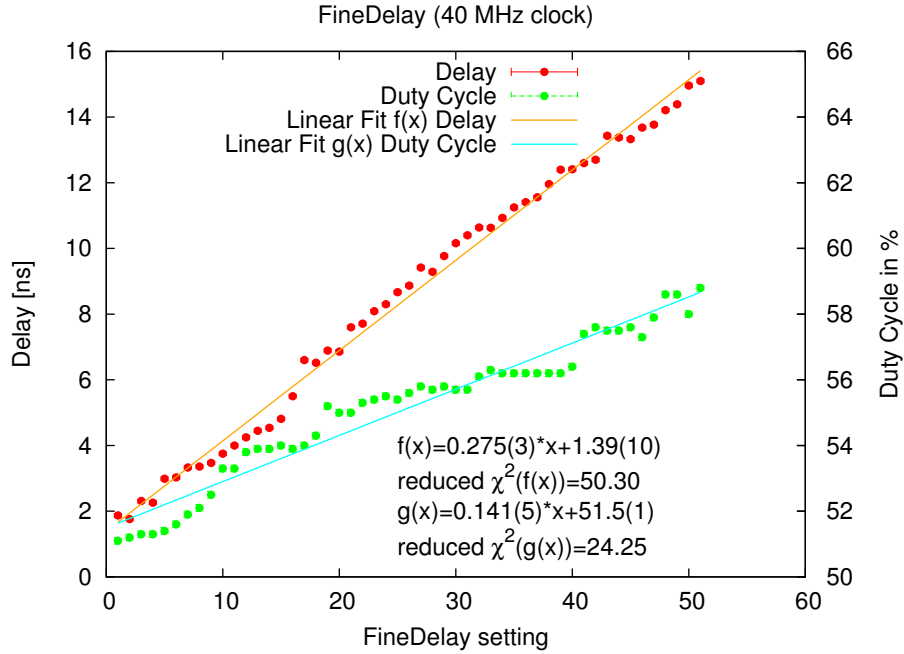


Figure 7.20: Measurement of the delay and duty cycle in dependence of the FineDelay setting.

[Neu11], but it was not investigated if this realization has an effect on the pulse width. However by the structure of these blocks it is not expected, that they effect the pulse width of a signal. Detailed measurements a planned to clarify this point. A IODELAY2 block has a total of 255 available steps, resulting in delay coverage of ≈ 12.75 ns, thus a coarse delay of 25 ns would not be sufficient to use these delay blocks. This could be compensated by using the 80/160 MHz clock available in the FPGA for the CoarseDelay, allowing a step size of 12.5/6.25 ns. This concept should be favored over the realization with carry chains, because it would be easier to tune.

7.6 RX Path

The RX path contains the logical block needed to receive the module data on the IBL-BOC, decode it and hand it over to the ROD for further processing. The data flow through the different blocks is shown in Figure 7.21. First of all the data is oversampled in an ISERDES block [Xil10a] contained in the input/output buffer of the FPGA. After a phase and word alignment the 8b10b encoded data can be decoded. The decoded data is then stored shortly in the RX FIFO, which is read out by the BOC to ROD Mux.

The synchronization and deserialization are also based on a preliminary study [Neu11], but received significant changes in the evaluation phase.

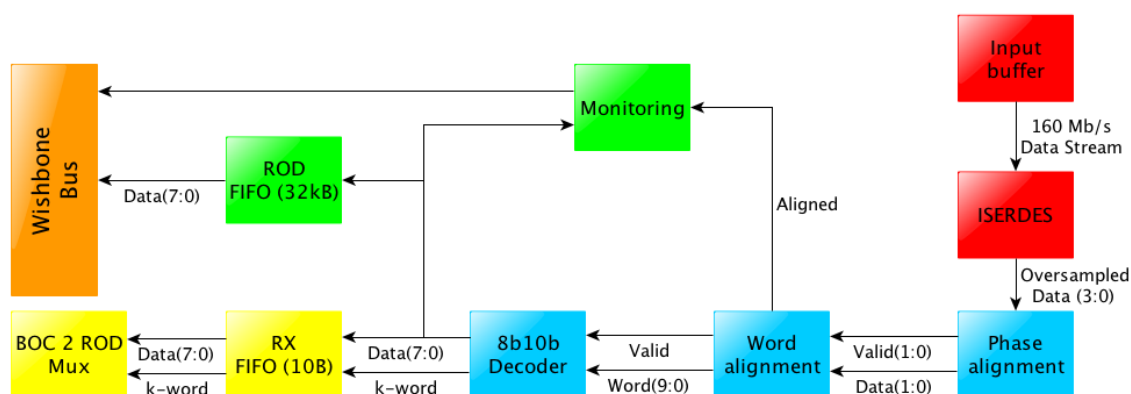


Figure 7.21: Schematic of the logical blocks used for the RX path, consisting of the input buffer with ISERDES (red), the data reconstruction and decoding blocks (blue), the BOC to ROD Mux (yellow) and monitoring blocks (green).

7.6.1 ISERDES and Phase Alignment

The ISERDES in the input buffer of the FPGA is supplied with an 640 MHz clock. The logic inside of the ISERDES oversamples the received data stream by a factor of 4 to 1. The 4 bit output is synchronous to the internal 160 MHz. The sampling point is now chosen by searching for either a rising or falling edge in this oversampled data. If a transition was found, the sampling point is set to be 2 bits behind the transition. This sampling point should not be affected by jitter and - in case of a synchronous input - not change. However if the input phase changes, the Phase Alignment block will detect this and correct the sampling accordingly. In this case it can happen, that in one 160 MHz clock cycle 2 bits are sampled, because else 1 bit would be missed in the phase correction.

A simulation of the oversampling and phase alignment process is shown in Figure 7.21. The oversampling process is labeled with the numbers 1 to 4, 4 points of

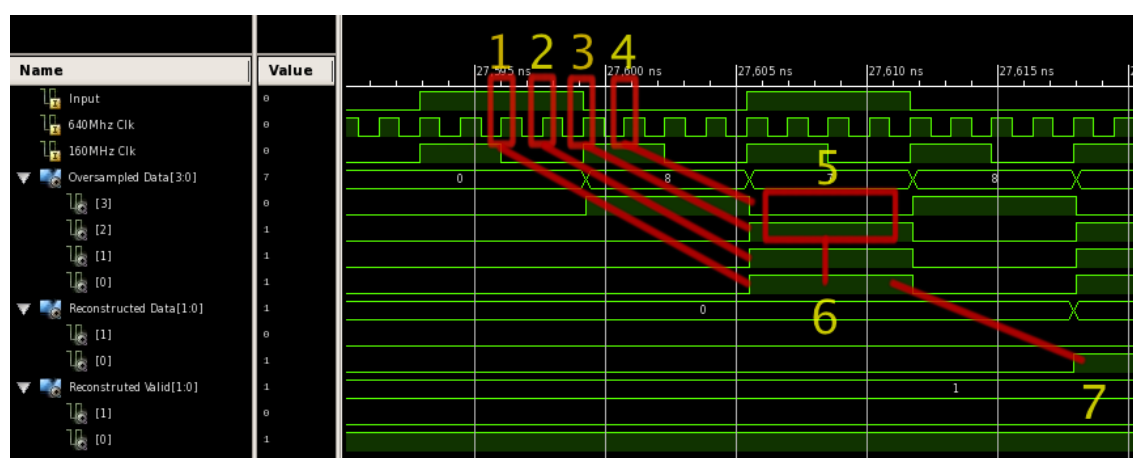


Figure 7.22: Simulation of the ISERDES output and phase alignment.

the input are sampled with the 640 MHz clock. The output of the oversampling

is shown in the 160 MHz clock cycle labeled with number 5. The falling edge of the input is detected between bit 3 and 4 (red box below the number 5), and the sampling point is chosen two bits further. The reconstructed data point is shown at number 7.

7.6.2 Word Alignment and 8b10b Decoder

The bits flagged valid after the phase alignment are stored in a shift register. To reconstruct the 10 bit words out of these single bits, the Word Alignment block needs to align itself onto a known pattern. The k-words (Idle, SOF, EOF) should be received frequently and offer a clear signature for alignment. When a k-word is detected in the shift register the Word Alignment block synchronizes to the data stream, allowing proper deserialization of the 10 bit words. A resynchronization occurs every time a k-word is received, hence a resynchronization is done at the beginning and end of every data frame via the SOF and EOF k-word. Every time a 10 bit word is collected, the 8b10b decoding is issued for this word. The decoded data is flagged valid, when Word Alignment block has been synchronized previously and the decoder accepts the words as a valid input. A simulation of this is shown in Figure 7.23. The highlighted sections labeled with the number 1, exemplary show



Figure 7.23: Simulation of the Word Alignment and 8b10b decoder block.

the process of filling the shift register with data from the Phase Alignment block.

The content of the shift register is checked for encoded k-words and in case one is found, a resynchronization is done. When the Word Alignment block is synchronized a bit counter is used to identify a fully collected 10 bit word. This is shown in the box labeled with number 2, the 10 bit word is latched and the Decoder block is enabled. The decoding process takes 1 clock cycle. The decoded word is put on the output of the Decoder and flagged as valid, labeled with number 3.

7.6.3 BOC to ROD Mux

To be pin compatible to the current ROD, the data is fed into the ROD through a 96 bit wide bus. However the IBL-BOC is designed for the reception of 32×8 bit data, which would require a not available 256 bits wide bus to the ROD. Therefore, the data from 4 RX channels are multiplexed onto a 12 bit wide bus, shrinking the bus width to the available 96 bits. The multiplexing is done by the BOC to ROD Mux shown in the block diagram in Figure 7.24. The decoded data

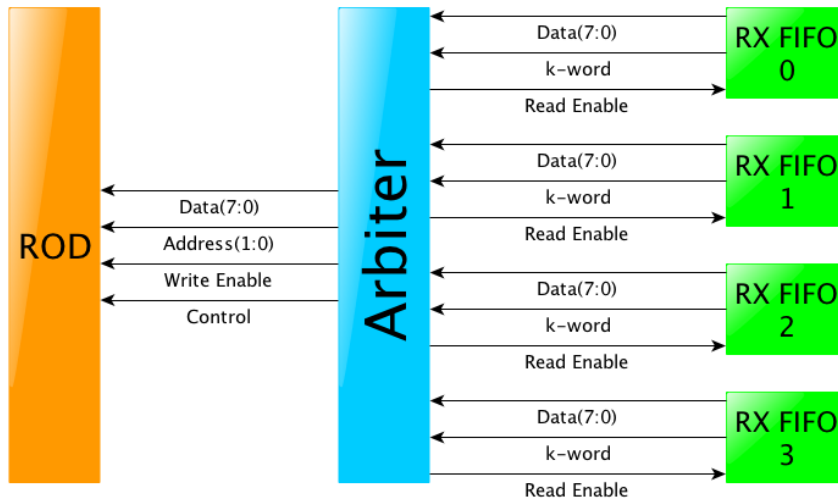


Figure 7.24: Block diagram of the BOC to ROD Mux.

is shortly stored in the RX FIFO. Four of these RX FIFOs are connected to one arbiter, which reads the data from them in a round-robin manner and feeds their data to the ROD. Since the data from multiple channels is given to the ROD, it has to be addressed for its respective channel. Four channels need an address range of 2 bit, leaving 1 bit to flag valid data on the bus (WriteEnable) and 1 bit to flag k-words (Control). The decoded data is written into the FIFOs with a frequency of 16 MHz, the arbiter reads the data from them with 80 MHz, hence every is read out at a minimum frequency of 20 MHz. Therefore, the storage of the data does not account as buffering. This is a crucial point, because otherwise the IBL-BOC has to share its busy logic with the ROD¹³. The simulation of the BOC to ROD Mux (Figure 7.25) shows that even under the highest possible data rate the data

¹³The busy logic tells the Higher Level Readout that the incoming data overextends the readout capability.

bus is not fully occupied. Also the ROD data bus protocol works as expected. By

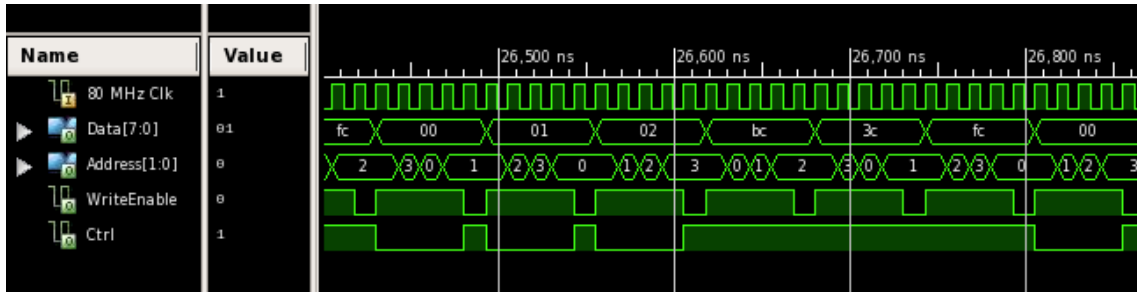


Figure 7.25: Simulation of the BOC to ROD Mux with the highest possible input data rate.

multiplexing the data of four channels on one bus, the ROD is required to know which address of the bus corresponds to which RX channel on the ROD. On the other this allows a easy parallel transmission of the data to the ROD.

This is also the first visualization of a multi channel simulation. The firmware deployed on the SP605 evaluation board is designed to implement 4 RX channels, to at least use one complete BOC to ROD Mux.

7.6.4 Measurement of the Data Reception

The measurement of the data reception utilizes the complete loopback chain in the firmware. A data pattern is written into the CMD FIFO, 8b10b encoded, serialized and then looped back external or internally into the RX path. The data reception was measured by analyzing the output of the data bus to the ROD (BOC to ROD Mux output). A picture of this setup is shown in Figure 7.26. The external loopback is done optically via the QSFP module, the ends of the attached pig-tail fiber are connected in a way, that each TX channel is connected to one RX channel. The ROD data bus is monitored with a logic analyzer connected to one of the break-out connectors on the mezzanine daughterboard. The looped back data stream was initially started with a decent amount of Idle words, to allow the receiver a proper alignment to signal. The logic analyzer measurement is shown in Figure 7.27. The transmitted data stream was the same as in Figure 7.25. Besides artifacts which can arise due to the asynchronous sampling of the logic analyzer, the data was received and decoded without errors.

It should be noted that this is a measurement performed on a multi channel implementation of the firmware. All four channels were decoded without errors and also in the same amount of time. However for a channel of higher number the time for proper alignment increases for an unknown reason. A possible cause for this behavior could be the placement of RX path logical blocks for each channel. This should be investigated in detail, when the firmware evaluation is finished.

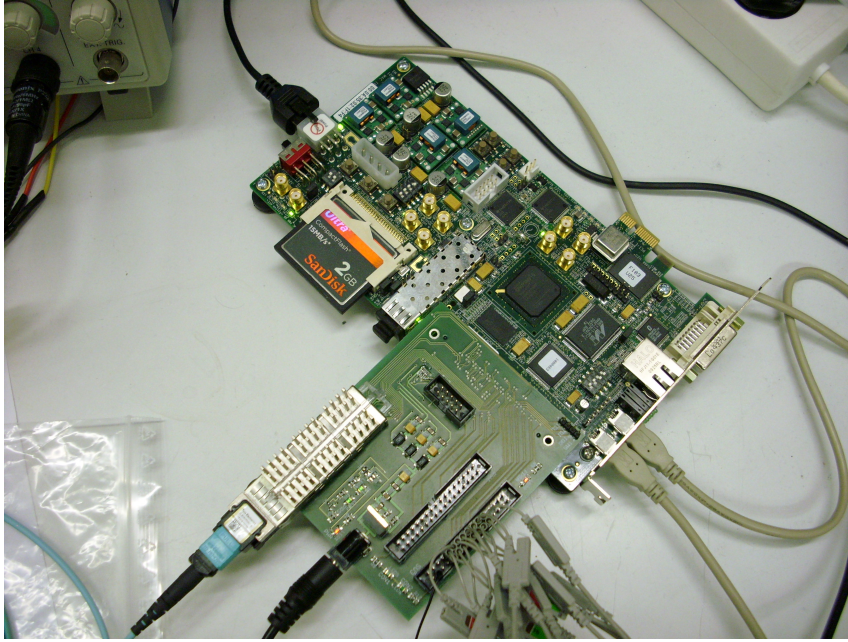


Figure 7.26: Experimental setup of the data reception measurement. A external loop-back is given by connecting the ends of a pig-tail fiber to each other (TX to RX channels). The data bus to the ROD is analyzed by a logic analyzer attached to one of the breakout connectors.

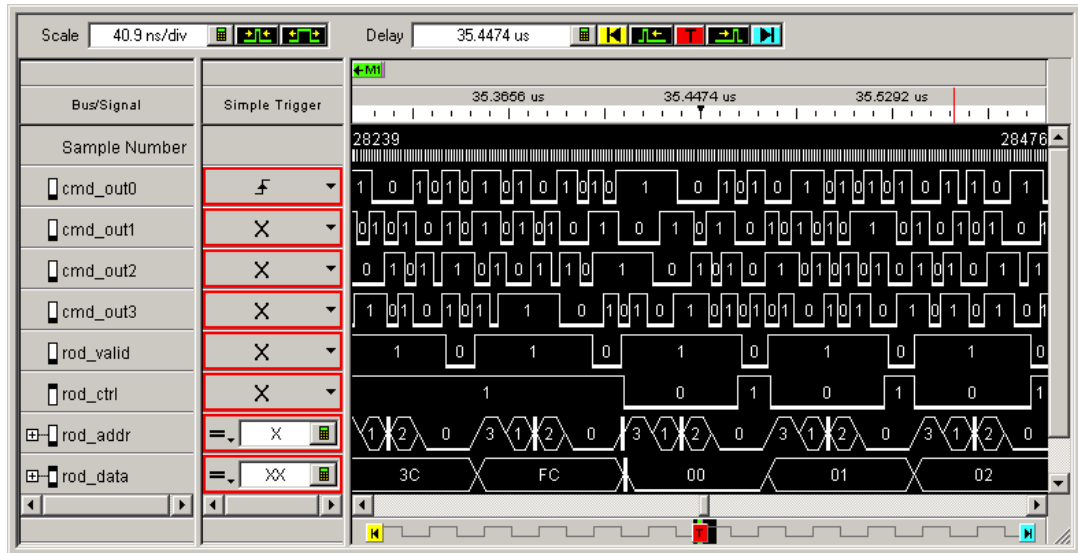
7.6.5 ROD FIFO

The ROD FIFO enables the IBL-BOC to actively record the received data. Every channel has its own ROD FIFO which is connected to Wishbone Bus. Currently the 8 bit width of the Wishbone data bus does only allow the readout of data content, but not the k-word flag. However due to the data format received by the FE-I4, this does allow a clear identification of the sequences. The depths of the ROD FIFO is chosen as big a possible, in the current firmware this is 32 kB, to record enough data to work with.

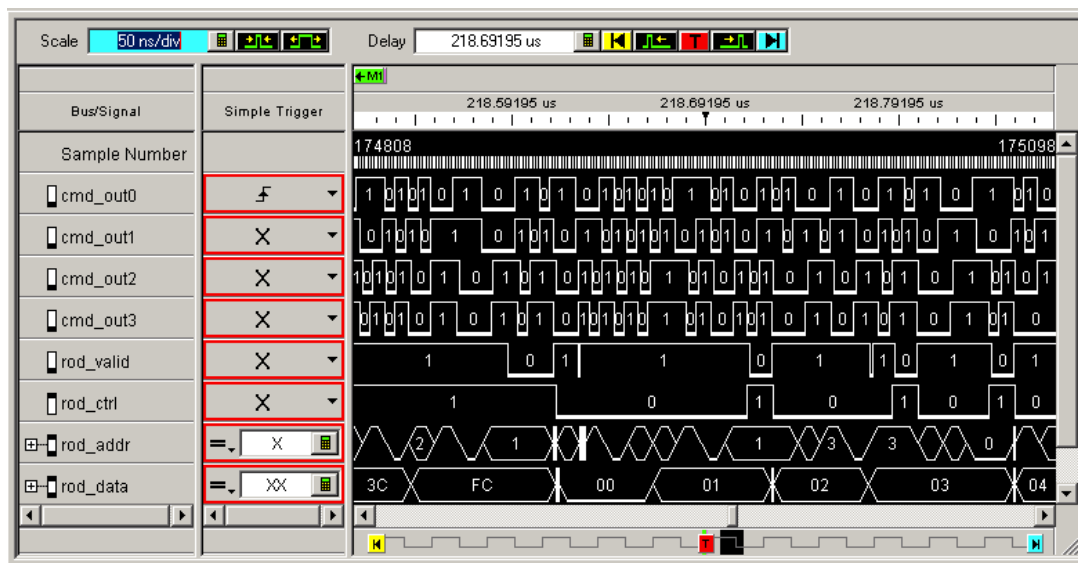
7.6.6 Monitoring

A Monitoring block is under development, monitoring the decoded data on the RX path. The FE-I4 protocol defines only four different sequences of data, a FSM (shown in Figure 7.28) in the Monitoring block follows this sequences and a error can be identified upon an anomaly. Different errors can be issued for each incorrect transition from one state to the next. Besides transition errors, specific data could be extracted or checked. Planned is a synchronization with the LV1 trigger counter, a trigger could be detected and counted on the TX path and the Monitoring block would expect to receive a specific LV1Id from the FE-I4. All detected errors could be histogrammed on the IBL-BOC or the IBL-BOC could inform the ROD upon detection of an error.

The FSM showed above is implemented in a VHDL module, being the basic log-



7.27.1: Internal loopback.



7.27.2: External optical loopback.

Figure 7.27: Logic analyzer measurement of the firmware data reception in a loopback setup. Shown are the transmitted data for each channel and the four signals the ROD data bus.

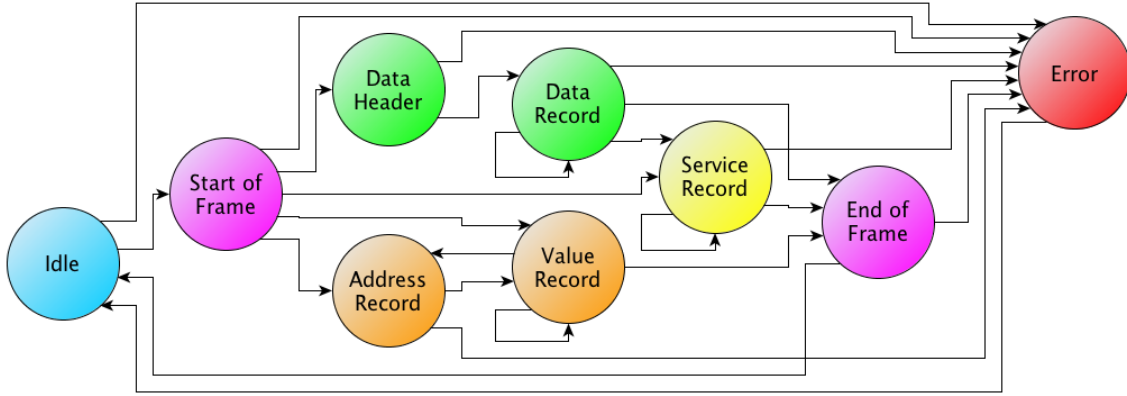


Figure 7.28: Finite State Machine of the Monitoring block following the FE-I4 data sequences.

ical block to perform monitoring on the data stream. However the logical blocks analyzing the data from the FSM are not yet completed.

7.7 Test System

The full test system consists of three parts: The SP605 programmed with the IBL-BOC firmware, one of the two mezzanine daughter boards for signal measurement and external connections and a PC to control and readout the firmware. The test system and its firmware is also referred to as the MEBOC¹⁴. The full address map of the MEBOC is listed in Table A.1 et seq.

7.7.1 Software

To control and readout the MEBOC a software Application Programming Interface (API) was designed. A overview of the software functional blocks is given in Figure 7.29. Also shown is a FE-I4 mounted on a Single Chip Card (SCC) connected to the MEBOC via LVDS and the specific block in the software API, they will be described in Section 7.8. The PrimBOC class contains the functions to utilize the UART interface to the MEBOC and use the protocol to perform simple single 8 bit read and write requests. It can be understood as a sole hardware interface and has no knowledge of the MEBOC itself. The BOC class instead contains all the controlling functions for the MEBOC, e.g. setting and getting register values. The BOC class does this, by using the PrimBOC class to perform read and write requests. In case the interfacing hardware is changed, only the PrimBOC class needs to be changed to fit to the new hardware, but the rest of the API could stay the same. The BOCtools class combines multiple functions of the BOC class to e.g. perform register or functional scans. A faked data stream from the FE-I4 chips can be generated in the fakeFE-I4 class, using the loopback feature of the MEBOC, the reception

¹⁴Mighty Evaluation BOC.

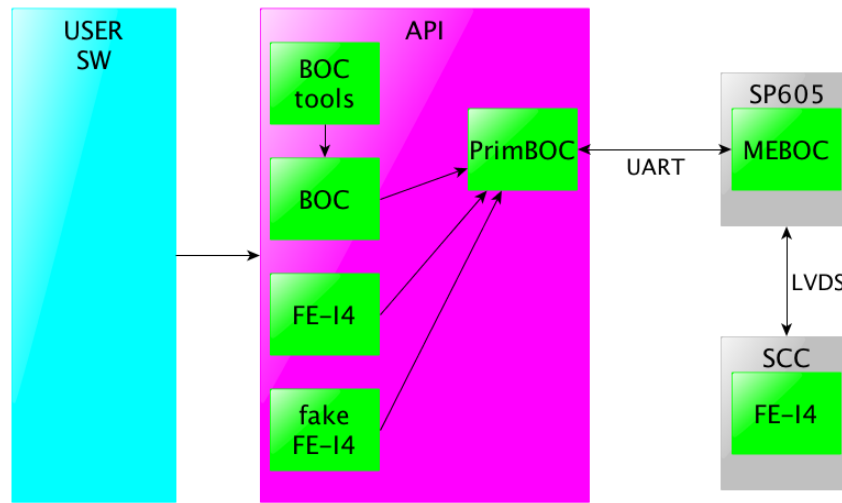


Figure 7.29: Functional blocks of the software API.

of this data can be monitored and checked for errors.

7.7.2 Register Scan

A basic routine, to test the functionality of the communication and the MEBOC, is a register scan. In this scan registers of the MEBOC are written and read back with every possible value. Currently implemented are the BPM encoder and adjustment registers:

```
Opening port to BOC... done!
BOC Firmware: 0x4
Checking MarkSpace on channel 0 from 0 to 31... ok!
Checking FineDelay on channel 0 from 0 to 127... ok!
Checking StreamInhibit on channel 0 from 0 to 1... ok!
Checking StreamExhibit on channel 0 from 0 to 1... ok!
Checking MarkSpace on channel 1 from 0 to 31... ok!
Checking FineDelay on channel 1 from 0 to 127... ok!
Checking StreamInhibit on channel 1 from 0 to 1... ok!
Checking StreamExhibit on channel 1 from 0 to 1... ok!
Checking MarkSpace on channel 2 from 0 to 31... ok!
Checking FineDelay on channel 2 from 0 to 127... ok!
Checking StreamInhibit on channel 2 from 0 to 1... ok!
Checking StreamExhibit on channel 2 from 0 to 1... ok!
Checking MarkSpace on channel 3 from 0 to 31... ok!
Checking FineDelay on channel 3 from 0 to 127... ok!
Checking StreamInhibit on channel 3 from 0 to 1... ok!
Checking StreamExhibit on channel 3 from 0 to 1... ok!
```

First of all a successful connection to the MEBOC can be tested with this scan, because every tested value of each register corresponds to one read and write cycle

via the UART interface. On the other hand, the firmware in the FPGA can be checked for being programmed successful and without errors.

In case of a working UART interface, a failure has never been observed in this scan.

7.7.3 Loopback Scan

A more advanced test routine involves the loopback capability of the MEBOC. A known pattern is 8b10b encoded and transmitted by the TX path, this data stream is internally looped back into the RX path. There it is deserialized, decoded and written into the ROD FIFO. By comparing the received data with the transmitted pattern, errors can be detected. The chronological sequence of the routine is as follows:

1. Reset all register to standard configuration and drain the CMD and ROD FIFO.
2. Write a pattern into the CMD FIFO.
3. Setup the TX path controlling parameters to the Loopback mode configuration. (160 Mb/s, 8b10b encoding, bypass BPM encoding)
4. Select internal or external loopback.
5. Select to repeat the pattern or transmit it only once.
6. Setup the MSR and FineDelay setting (if desired).
7. Start the transmission of the data pattern.
8. Wait some time.
9. Stop the transmission and reset the TX path control parameters.
10. Readout the desired amount of data from the ROD FIFO.
11. Drain the CMD and ROD FIFO.

This routine was used to debug the receiving part of the MEBOC firmware and led to several improvements.

The transmitted data stream should be optimal for the reception, because it is not distorted in any form. A distortion can be enforced by adjusting the MSR or applying a delay.

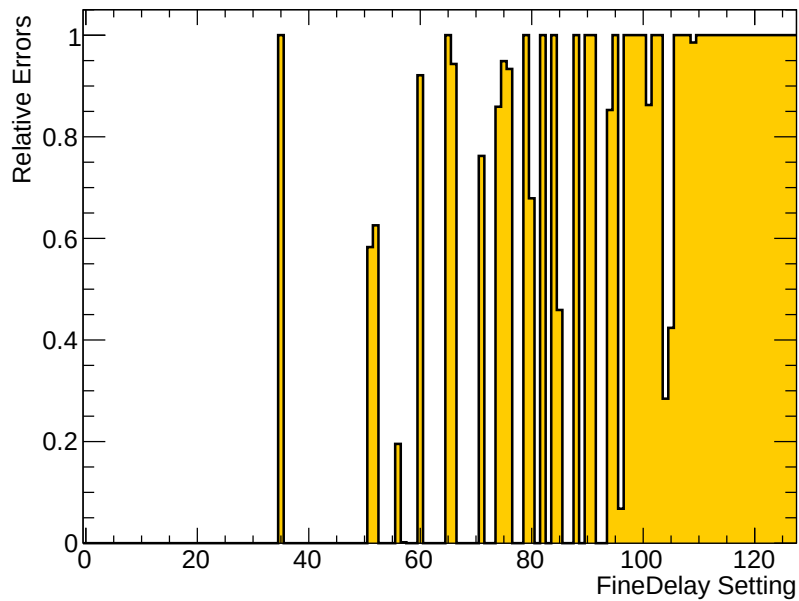
Phase Scan (samples = 5000, MSR Setting = 13)

Figure 7.30: Measured relative error rate of the receiver in dependence of the FineDelay setting or ≈ 300 ps phase shift per setting respectively.

Phase Scan

In a loopback scan the skew between transmission and reception is minimal due to the short internal signal routing, hence the received signal has nearly the same phase as the sampling clocks. However the receiver should work for every received phase. This can be tested by delaying the transmitted signal with the FineDelay. A Phase Scan is shown in Figure 7.30. Most important in this scan is the region from FineDelay setting 0 to 25, this equates to a delay from 0 to around 7.5 ns and therefore a complete phase. Over the complete phase the receiver deserialized and decoded the data without any errors. This proves the phase independence of the above described receiver concept.

First errors occur at setting 35 and with increasing FineDelay setting errors occur more frequently. After setting 100 nearly no data could be decoded right. However this is a effect caused by pulse width broadening introduced by the FineDelay as shown in Figure 7.20. It can be observed in more detail in the following Phase-MSR Scan.

MSR Scan

By varying the MSR of the transmitted data the positive width of a high bit is changed. This can be used to determine the minimal and maximal bit width (around the 100% width) the receiver can deserialize and decoded. A MSR scan is shown

in Figure 7.31¹⁵. Visible is an error free region between setting 11 and 26. The

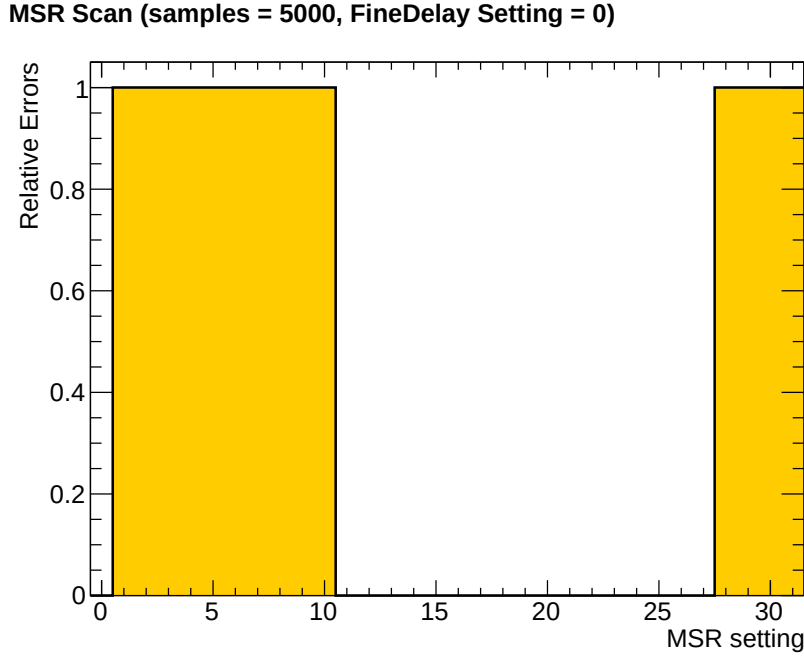


Figure 7.31: Measured relative error rate of the receiver in dependence of the MSR setting.

absolute values of the lower and upper bound of the error free region vary for every new implementation¹⁶ of the firmware, however the widths of the error free region is always about 15 MSR settings.

This behavior is not expected in a real data stream, but on the one hand it is a already installed logic to distort the signal and on the other hand it helps to understand the observation made in the Phase Scan.

Phase-MSR Scan

Combining the two scans from above and measuring the error rate of every MSR setting in dependence of every FineDelay setting, a Phase-MSR scan is achieved (shown in Figure 7.32). The Phase-MSR Scan reveals a error free tunnel, with the width of about 15 MSR settings. This width refers to the width of the error free region in the MSR Scan. The lower and upper bound of the error free region decrease with increasing FineDelay setting. The upper bound decreases by about 20 MSR settings over the full width of FineDelay settings, this corresponds to a negative pulse width change of about 6 ns. This is the compensation of the positive pulse width change introduced by the FineDelay (50 ps per setting), which is also about

¹⁵At the MSR setting of 0 the MSR and FineDelay block is bypassed.

¹⁶Translation of the VHDL code into electrical components and optimization of the placement of each logical block.

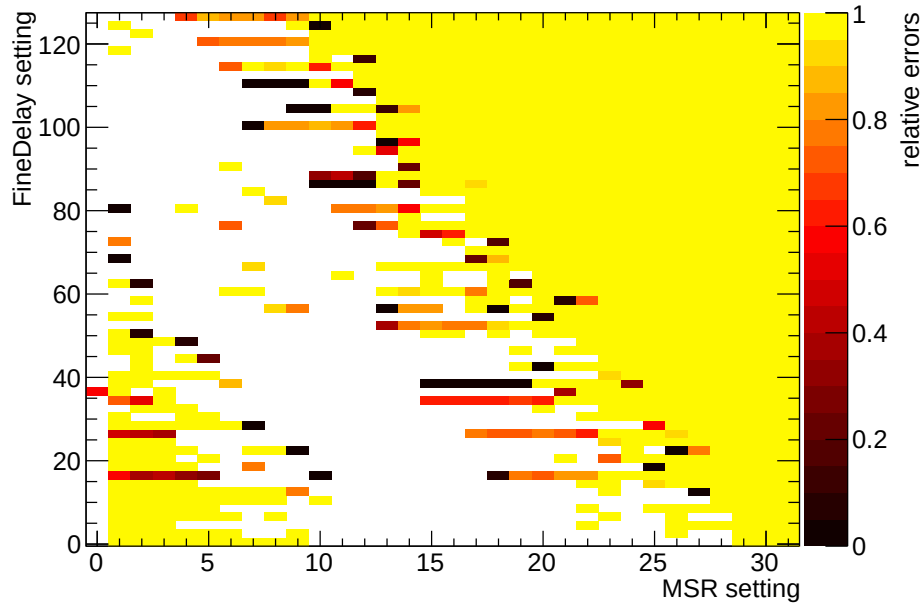
MSR Phase Scan (samples = 5000)

Figure 7.32: Measured relative error rate of the receiver in dependence of MSR and FineDelay setting.

6 ns.

This behavior explains the upper bound in the Phase Scan. If the same Phase Scan is performed with a lower MSR setting, the upper bound will move to higher FineDelay settings.

Outlook

The Phase Scan is a realistic parameter to simulate different cable/fiber lengths, but should not distort the signal. To further evaluate the RX path, parameters are planned to be implemented which distort the signal in a way a “bad” data stream would be like. The data stream from the FE-I4 will be influenced mainly by three parameters: The output power of the VCSEL on the Optoboard, jitter of the signal and the reference clock of the FE-I4 reconstructed out of the BPM encoded TTC stream by the DORIC.

The latter directly influences the 160 MHz clock the FE-I4 generates to send the 160 Mb/s data stream. A deviation in the reference clock could result in a higher/lower bandwidth of the data link. This could be simulated in the MEBOC by implementing three different clocks to drive the serialization of the data in Loopback mode, one optimal 160 MHz clock, one little bit slower and one little bit faster clock.

A logical block applying a jitter to the transmitted data stream in the Loopback mode exists, but this block uses the variable delay block also contained in the FineDelay block. Therefore, it introduces the same issues the FineDelay and MSR adjustment blocks do.

To investigate the impact of low light power on the data stream quality, a internal loopback is not sufficient. The Loopback mode of the TX path must be operated via an external external loopback with connected optical modules. The light power could then be lowered by using a attenuator and the reception in dependence of light power could be measured. However the receiving optical module is a major factor in this setup and must be considered in the interpretation of the functionality of the RX path.

7.8 Combined Test with FE-I4

A bare¹⁷ FE-I4a¹⁸ wire bonded to a Single Chip Card (SCC)¹⁹ is used to perform realistic tests with the MEBOC. Normally the SCC is used in combination with USBPix²⁰ readout system. The FE-I4 was powered by an external supply, the different voltages are listed in Table 7.3. The MEBOC is connected to the FE-

Table 7.3: Supply voltages of the FE-I4

	Voltage [V]	comment
VDDD	1.25	Digital bias
VDDA	1.5	Analog bias
VDDPLL	1.25	PLL bias

I4 via three LVDS²¹ signals. Normally the BPM stream sent to the detector, is received by the DORIC. It reconstructs the clock and data and sends them to the FE-I4. A DORIC is not located on the SCC, therefore the FE-I4 needs to receive the clock and data divided upon two LVDS signals. The third LVDS signal is used for the data output of the FE-I4. The hardware connection is made by a standard ethernet patch cable, plugged into the RJ-45 connector on the SCC and another one on a mezzanine daughter board of the SP605. A Picture of the SCC is shown in Figure 7.33. After power up the FE-I4 is in a minimal configuration (e.g. the output is turned off). Therefore, the FE-I4 has to be configured with a basic configuration (listed in Table A.6 et seq.), enabling the data output, choosing the system clocks and setting various biases and currents to a working level. The configuration was written into the FE-I4 by using the WRegister command from Table 6.5 and the standalone TTC mode of the TX path. A successful configuration is recognized by the enabled output and a higher power consumption (see Table 7.4). The software API of the MEBOC was extended by a FE-I4 class. This class was responsible for communication to the FE-I4 and has a virtual copy of all global registers. The data readout of the ROD FIFO on the MEBOC is initiated by this class, if data is expected to have arrived e.g. after a trigger or configuration read back was sent.

¹⁷No sensor.

¹⁸Preliminary version of the FE-I4 chip used for development and radiation hardness tests.

¹⁹Assembled at the University of Bonn.

²⁰Developed at the University of Bonn.

²¹Low Voltage Differential Signal

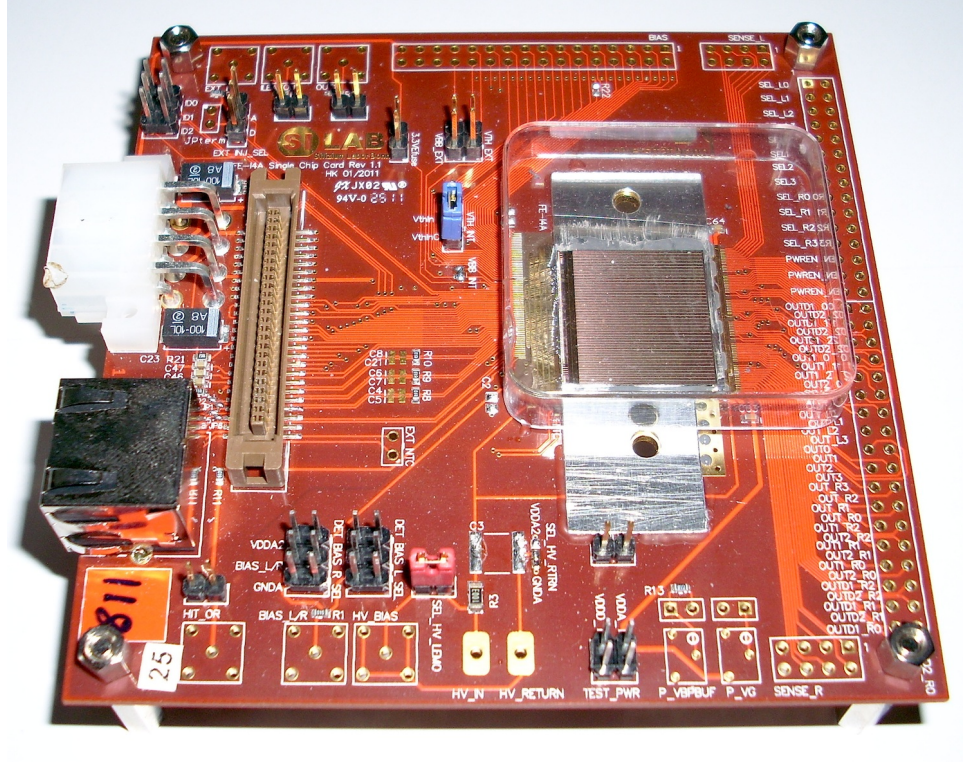


Figure 7.33: Picture of the Single Chip Card connected to a bare FE-I4 via wire bonds.

Table 7.4: Power consumption for the configured and unconfigured state of the FE-I4.

	Channel	Current [mA]
Unconfigured	Digital	≈ 90
	Analog	≈ 60
Configured	Digital	≈ 90
	Analog	≈ 300

7.8.1 Proof of Principle

The first and easiest attempt to communicate with the FE-I4, is a configuration read back. The software send a RdRegister command to the FE-I4 and then reads on frame (recognized by the SOF and EOF) from the ROD FIFO. A example of a read back could look like the following:

```
Word[0] = 0xfc
Word[1] = 0xea
Word[2] = 0x00
Word[3] = 0x02
Word[4] = 0xec
Word[5] = 0x18
Word[6] = 0x00
Word[7] = 0xbc
```

Explanation:

- 0xfc : Start of frame
- 0xea : Address record header
- 0x0002 : Register address
- 0xec : Value record header
- 0x1800 : Register value
- 0xbc : End of frame

Hence the register at address 0x02 has the value 0x1800. This simple test proves the functionality of the communication to the MEBOC, as well as the successful configuration of the FE-I4.

7.8.2 Digital scan

By performing a Digital Scan the functionality of the FE-I4 is checked and a massive amount of data is received by the MEBOC. The delivered amount of data is predictable to a certain level and allows further evaluation of the MEBOC receiver logic.

Pixel Readout Cell Configuration

A Digital Scan requires the pixel readout cells of the FE-I4 to be configured in a certain way, this is done by a sequence of commands. First of all the desired double column to be configured is chosen. Each pixel readout cell in this double column can now be masked, by setting its bit in the pixel shift register to “1”. This is done by either shifting a 672 bits long bitstream into the shift register or if the configuration of the complete double column is desired, by setting the whole shift

register to “1”. The configuration to be written into the pixel readout cells is chosen by setting the pixel strobes in global register, these are principally a mirrored version of the registers located in the pixel readout cell. The configuration stored in these pixel strobes is latched into each masked pixel readout cell, by setting the “Latch Enable” signal to “1” and sending a GlobalPulse command. Thereby a pulse is generated which is put into a AND gate together with the “Latch Enable” setting and the result toggles the pixel readout cells to latch the configuration from the Pixel strobes.

Digital Injection

After configuring the pixel readout cells for a digital scan, a digital pulse must be injected. Due to the configuration, this pulse will be treated as the output of the discriminator and the ToT value will be stored in the digital region of the pixel readout cell. The ToT value is not a straight forward conversion of the pulse length, but rather determined by a “Discriminator Config” setting located in the global register.

The digital pulse is generated with the CAL command, its length and delay is defined by a global register setting. To readout the stored ToT values a trigger must be sent to the FE-I4 with exact timing in reference to the CAL command. By first storing the commands in the CMD FIFO, the time between the CAL command and the trigger can be defined very precisely, in the resolution of 1 bit. This is necessary, because the received Trigger must match the timestamp of the registered hit.

After successfully pulsing and triggering the FE-I4, it will send out the decoded ToT values for every Hit.

Readout

A digital pulse was injected into each double column and read out consecutively. After successfully injecting and triggering, the FE-I4 sends out one Pixel Data sequence (see Section 6.2), containing every registered hit from one double column. In case of a digital injection this should be the ToT value of every pixel in the whole double column or 672 ToT values in total. Framed in the FE-I4 data protocol this is about 1 kB of data in one 8b10b frame. The data is received by the MEBOC and read out from the ROD FIFO through the software. After reading out the data, the software will decode the FE-I4 data protocol and the raw hit data is available to the user.

Result

The result of a Digital Scan is shown in Figure 7.34. The histogram shows the

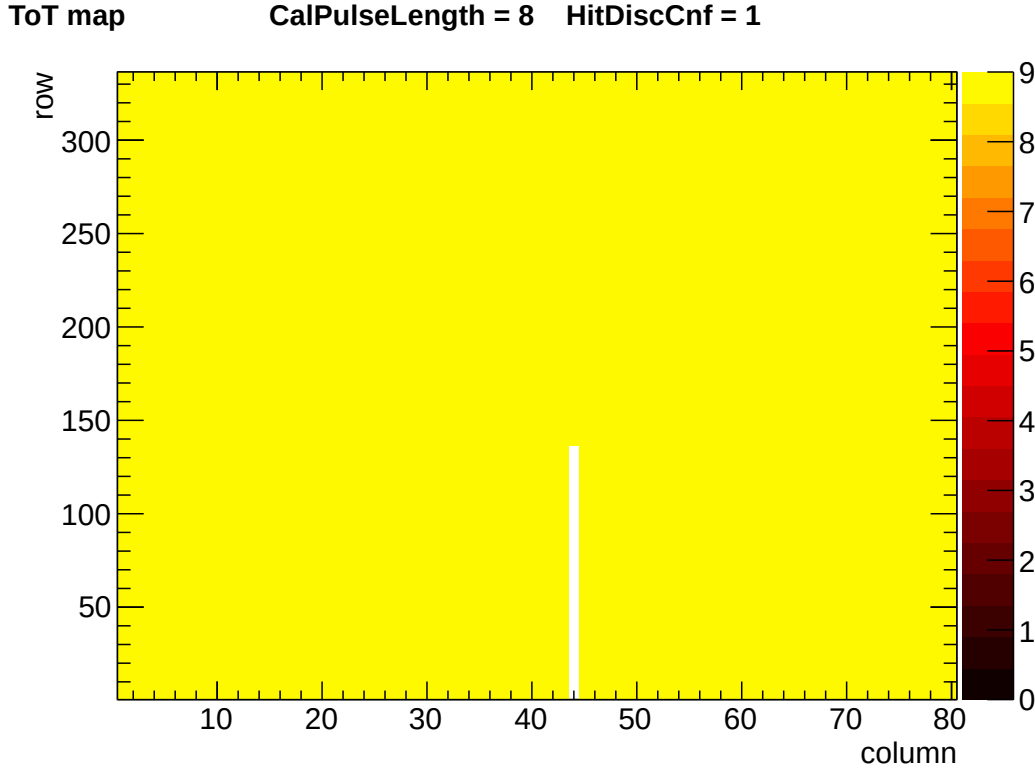


Figure 7.34: Digital scan of the FE-I4 chip performed with the MEBOC as the readout system. Shown are the received ToT values after sending the Trigger.

ToT value received from the FE-I4 after consecutively injecting a digital pulse into all double columns. The digital part of column 44 is partly broken²², which is in accordance to a reference scan with the USBpix system (see Figure A.1).

7.8.3 Outlook

Testing the MEBOC with a real FE-I4 divides the development into two fields. On the one hand the control and scan routines for the FE-I4 must be realized in the software, which is not directly correlated to the evaluation of the firmware. The next step in this development will be the utilization of an Analog Scan, allowing the first steps towards calibrating a FE-I4 for operation and the reception of more realistic data.

On the other hand the operation of the FE-I4 with the MEBOC offers the optimal environment to investigate the behavior of the firmware, especially the reception

²²The tested FE-I4 was referred to as a “yellow chip”, because the electronics were not 100% functional (referred to as “green chip”). However this does not influence the measurement used for testing the MEBOC.

of the data stream. Hence the connection of the MEBOC to the FE-I4 should be extended to a more realistic level. This could be done by using an Optoboard for an optical link, which would also have the advantage of using one BPM encoded TTC stream and not two links for the reference clock and data.

Chapter 8

Conclusion

The versatility of FPGAs enables them to be used in many different fields of the detector readout of high energy particle experiments. They perfectly fit into the custom demands of these readout concepts, which are often not satisfied by industrial supplies. In case of the IBL-BOC primarily everything, from control to data processing, is performed in the FPGAs. The firmware and its test setup developed for this project, is described in this thesis and evolved to a standalone demonstrator for the functionality of the IBL-BOC.

8.1 Firmware Development and Outlook

The current stage of the IBL-BOC firmware is not final. However all the functionality needed for the data transmission and reception, as well communication have been combined and tested. The used test setup offers a stand alone platform for further development of the IBL-BOC firmware. To advance the functionality compared to the current BOC, a detailed data stream monitoring is planned.

The structure of the firmware was designed while having the structure of the IBL-BOC prototype in mind. The logical blocks are divided into two parts, one planned for the deployment in the BCF and one for the BMFs. This should ease the act of porting the firmware from the MEBOC onto the IBL-BOC prototype. By the end of 2011 a combined test with the IBL-ROD is planned to test the firmware and hardware on both components and prepare them for the production of the IBL. Even after installation of all hardware components for the IBL, the firmware could still be corrected for bugs or upgraded with further functionality.

First tests with the optical modules, considered to be used on the IBL-BOC, just scratched the surface and have to be done in more detail. The characteristics of the optical devices have to be determined and compared to the optical devices used on the Optoboard.

8.2 First FE-I4 Tests

The first tests with a real FE-I4 were very convincing. Beside some hardware issues corresponding to the connection of the SP605 and the FE-I4, the readout worked nearly flawless and without further tuning. To perform scans on the FE-I4 the software API was upgraded and showed the possibility to be used as a standalone readout system for IBL module testing. However to readout a FE-I4 at nearly full bandwidth a alternative concept of communication to external devices has to be used.

8.3 Future DAQ Concepts

The UART interface of the MEBOC is just a emergency solution to have a connection to a PC. A promising alternative would be the connection via PCI Express (PCIe)¹. While developing the firmware for the IBL-BOC the implementation of a PCIe interface were not successful and shifted to a later point in time. However the principal concept of tying a PCIe end block to the internal Wishbone Bus should be possible, but the realization in firmware and software is more complicated.

A PCIe interface would offer the bandwidth needed for a standalone readout. It was already discussed for the current IBL-BOC to deploy the event builder from the ROD on the IBL-BOC to bypass the bottleneck of the VME backplane. In this case the received data would be processed on the IBL-BOC and then send via the S-Link to the higher level readout. The calibration would still be done by the ROD, because they need extended computing power.

In a future DAQ concept the optical detector readout could be done by a PCIe card housed in a PC. The front plate of a standard PCIe card provides enough space for three Snap12 and one QSFP module, this matches the capability of half an IBL-BOC. In operation the received data would be processed by a FPGA, the processing power needed for calibration could be utilized by a Graphic Processing Unit (GPU) [Sch10]. These are processing units specialized for numerical operations, first studies showed a significant speed up of the processing time for calibration. The GPU could be located on either the same PCIe card used for the readout or another industrial card. The system would be controlled by the PC they are housed in.

Equipping the MEBOC test system with a PCIe interface would be a significant step towards the above described concept. Studies on this system could be used for the development of future readout systems.

¹Standardized under the PCI express base specification 3.0 by PCI-SIG

Appendix A

Appendix

A.1 MEBOC Address Map

Table A.1: Preliminary address map of the MEBOC version 4. BCF Part 1.

Address	Range	Default setting	Location	Comment
0x010	0-255	-	BCF - Reg Bank	Main Firmware revision
0x011	0-255	-	BCF - Reg Bank	Spare
0x012	0-255	-	BCF - Reg Bank	Spare
0x013	0-255	-	BCF - Reg Bank	Spare
0x014	0-255	-	BCF - Reg Bank	Spare
0x015	0-255	-	BCF - Reg Bank	Spare
0x016	0-255	-	BCF - Reg Bank	Spare
0x017	0-255	-	BCF - Reg Bank	Spare
0x018	0-255	-	BCF - Reg Bank	Spare
0x019	0-255	-	BCF - Reg Bank	Spare
0x01A	0-255	-	BCF - Reg Bank	Spare
0x01B	0-255	-	BCF - Reg Bank	Spare
0x01C	0-255	-	BCF - Reg Bank	Spare
0x01D	0-255	-	BCF - Reg Bank	Spare
0x01E	0-255	-	BCF - Reg Bank	Spare
0x01F	0-255	-	BCF - Reg Bank	Spare

Table A.2: Preliminary address map of the MEBOC version 4. BCF Part 2.

Address	Range	Default setting	Location	Comment
0x020	0-255	-	BCF - BRAM	Spare
0x021	0-255	-	BCF - BRAM	Spare
0x022	0-255	-	BCF - BRAM	Spare
0x023	0-255	-	BCF - BRAM	Spare
0x024	0-255	-	BCF - BRAM	Spare
0x025	0-255	-	BCF - BRAM	Spare
0x026	0-255	-	BCF - BRAM	Spare
0x027	0-255	-	BCF - BRAM	Spare
0x028	0-255	-	BCF - BRAM	Spare
0x029	0-255	-	BCF - BRAM	Spare
0x02A	0-255	-	BCF - BRAM	Spare
0x02B	0-255	-	BCF - BRAM	Spare
0x02C	0-255	-	BCF - BRAM	Spare
0x02D	0-255	-	BCF - BRAM	Spare
0x02E	0-255	-	BCF - BRAM	Spare
0x02F	0-255	-	BCF - BRAM	Spare

Table A.3: Preliminary address map of the MEBOC version 4. BMF Part 1.

Address	Range	Default setting	Location	Comment
0x100	0-31	0	BMF - Reg Bank	MSR setting channel 0
0x101	0-127	0	BMF - Reg Bank	FineDelay setting channel 0
0x102	0-1	0	BMF - Reg Bank	Inhibit setting channel 0
0x103	0-1	0	BMF - Reg Bank	Exhibit setting channel 0
0x104	0-31	0	BMF - Reg Bank	MSR setting channel 1
0x105	0-127	0	BMF - Reg Bank	FineDelay setting channel 1
0x106	0-1	0	BMF - Reg Bank	Inhibit setting channel 1
0x107	0-1	0	BMF - Reg Bank	Exhibit setting channel 1
0x108	0-31	0	BMF - Reg Bank	MSR setting channel 2
0x109	0-127	0	BMF - Reg Bank	FineDelay setting channel 2
0x10A	0-1	0	BMF - Reg Bank	Inhibit setting channel 2
0x10B	0-1	0	BMF - Reg Bank	Exhibit setting channel 2
0x10C	0-31	0	BMF - Reg Bank	MSR setting channel 3
0x10D	0-127	0	BMF - Reg Bank	FineDelay setting channel 3
0x10E	0-1	0	BMF - Reg Bank	Inhibit setting channel 3
0x10F	0-1	0	BMF - Reg Bank	Exhibit setting channel 3

Table A.4: Preliminary address map of the MEBOC version 4. BMF Part 2.

Address	Range	Default setting	Location	Comment
0x110	0-255	0	BMF - FIFO	Channel 0 Write: CMD-FIFO d-word Read: ROD-FIFO
0x111	0-255	0	BMF - FIFO	Channel 0 Write: CMD-FIFO k-word
0x112	0-255	0	BMF - TX Mux	Mux parameter channel 0
0x120	0-255	0	BMF - FIFO	Channel 1 Write: CMD-FIFO d-word Read: ROD-FIFO
0x121	0-255	0	BMF - FIFO	Channel 1 Write: CMD-FIFO k-word
0x122	0-255	0	BMF - TX Mux	Mux parameter channel 1
0x130	0-255	0	BMF - FIFO	Channel 2 Write: CMD-FIFO d-word Read: ROD-FIFO
0x131	0-255	0	BMF - FIFO	Channel 2 Write: CMD-FIFO k-word
0x132	0-255	0	BMF - TX Mux	Mux parameter channel 2
0x140	0-255	0	BMF - FIFO	Channel 3 Write: CMD-FIFO d-word Read: ROD-FIFO
0x141	0-255	0	BMF - FIFO	Channel 3 Write: CMD-FIFO k-word
0x142	0-255	0	BMF - TX Mux	Mux parameter channel 3
0x150	0-255	0	BMF - FIFO	All channels Write: CMD-FIFO d-word Read: ROD-FIFO
0x151	0-255	0	BMF - FIFO	All channels Write: CMD-FIFO k-word
0x152	0-255	0	BMF - TX Mux	Mux parameter all channels

Table A.5: Detail of the Mux parameter register as referred to in Table 7.2.

Bit(s)	Range	Default setting	Comment
0	0-1	0	Loop Enable
1	0-1	0	TTC Mux
2	0-1	0	RX Mux
3	0-1	0	Repeat
4	0-1	1	Drain
(6:5)	0-3	0	Spare
7	0	0	Mux Enable

A.2 FE-I4 Standard Configuration

Table A.6: Standard configuration of the FE-I4 used in the combined tests with the MEBOC. Part 1.

Global Register Address	Sub Register Number	Name	Mask	Value
2	0	TriggerCount	0xf000	0x1
2	1	ConfAddrEnable	0x800	0x1
2	2	CFGspare2	0x7ff	0x0
3	0	ErrorMask(15:0)	0xffff	0x0
4	0	ErrorMask(31:16)	0xffff	0x0
5	0	PrmpVbpR	0xff00	0x2b
5	1	Vthin	0xff	0xff
6	0	DisVbnCPP	0xff00	0x3e
6	1	PrmpVbp	0xff	0x2b
7	0	TdacVbp	0xff00	0xff
7	1	DisVbn	0xff	0x1a
8	0	Amp2Vbn	0xff00	0x4f
8	1	Amp2VbnFol	0xff	0x1a
9	0	PrmpVbpT	0xff00	0x0
9	1	Amp2Vbp	0xff	0x55
10	0	FdacVbn	0xff00	0x32
10	1	Amp2Vbpff	0xff	0xd
11	0	PrmpVbnFol	0xff00	0x6a
11	1	PrmpVbpL	0xff	0x2b
12	0	PrmpVbpf	0xff00	0x14
12	1	PrmpVbnLcc	0xff	0x0
13	0	S1	0x8000	0x0
13	1	S0	0x4000	0x0
13	2	PixelStrobes	0x3ffe	0x0
13	3	Spare1	0x1	0x0
14	0	LvdsDrvIref	0xff00	0xab
14	1	BonnDac	0xff	0xed
15	0	PlIbias	0xff00	0x58
15	1	LvdsDrvVos	0xff	0x69

Table A.7: Standard configuration of the FE-I4 used in the combined tests with the MEBOC. Part 2.

Global Register Address	Sub Register Number	Name	Mask	Value
16	0	TempSensIbias	0xff00	0x0
16	1	PlIcp	0xff	0x1c
17	0	DAC8SPARE1	0xff00	0x0
17	1	PlsrIdacRamp	0xff	0xb4
18	0	DAC8SPARE2	0xff00	0x0
18	1	PlsrVgOPamp	0xff	0xff
19	0	PlsrDacBias	0xff00	0x60
19	1	DAC8SPARE5	0xff	0x0
20	0	VthinAltCoarse	0xff00	0x80
20	1	VthinAltFine	0xff	0x80
21	0	Spare2	0xe000	0x0
21	1	HITLDIN	0x1000	0x0
21	2	DINJOVERRIDE	0x800	0x0
21	3	DIGHITINSEL	0x400	0x1
21	4	PlsrDAC	0x3ff	0x96
22	0	Spare3	0xfc00	0x0
22	1	ColprMode	0x300	0x0
22	2	ColprAddr	0xfc	0x0
22	3	Spare4	0x3	0x0
23	0	DisableColumnCnfg(15:00)	0xffff	0x0
24	0	DisableColumnCnfg(31:16)	0xffff	0x0
25	0	LatCnfg	0xff00	0xd2
25	1	DisableColumnCnfg(39:32)	0xff	0x0
26	0	CNT(12:0)	0xff8	0x0
26	1	StopModeCnfg	0x4	0x0
26	2	HitDiscCnfg	0x3	0x2
27	0	PlIEn	0x8000	0x1
27	1	EfuseSense	0x4000	0x0
27	3	StopSlkPulse	0x2000	0x0
27	4	ReadErrorReq	0x1000	0x0
27	5	ReadSkipped	0x800	0x0
27	6	Spare5	0x7c0	0x0
27	7	GateHitOr	0x20	0x0
27	8	CalEn	0x10	0x0
27	9	SrClr	0x8	0x0
27	10	LatchEn	0x4	0x0
27	11	SRCLOCK	0x2	0x0
27	12	CNT(13)	0x1	0x0

Table A.8: Standard configuration of the FE-I4 used in the combined tests with the MEBOC. Part 3.

Global Register Address	Sub Register Number	Name	Mask	Value
28	0	lvdsDrvSet06	0x8000	0x1
28	1	Spare6	0x7c00	0x0
28	2	EN40M	0x200	0x1
28	3	EN80M	0x100	0x0
28	4	CLK1S0	0x80	0x0
28	5	CLK1S1	0x40	0x0
28	6	CLK1S2	0x20	0x0
28	7	CLK0S0	0x10	0x0
28	8	CLK0S1	0x8	0x0
28	9	CLK0S2	0x4	0x1
28	10	EN160M	0x2	0x1
28	11	EN320M	0x1	0x0
29	0	Spare7	0xc000	0x0
29	1	no8b10bModeCnfg	0x2000	0x0
29	2	clkToOutCnfg	0x1000	0x0
29	3	EmptyRecordCnfg	0xff0	0x0
29	4	Spare8	0x8	0x0
29	5	lvdsDrvEn	0x4	0x1
29	6	lvdsDrvSet30	0x2	0x1
29	7	lvdsDrvSet12	0x1	0x1
31	0	PlsrRiseUpTau	0xe000	0x7
31	1	PlsrPwr	0x1000	0x1
31	2	PlsrDelay	0xfc0	0x2
31	3	ExtDigCalSW	0x20	0x0
31	4	ExtAnaCalSW	0x10	0x0
31	5	Spare9	0xf	0x0

A.3 FE-I4 USBpix Digital Scan

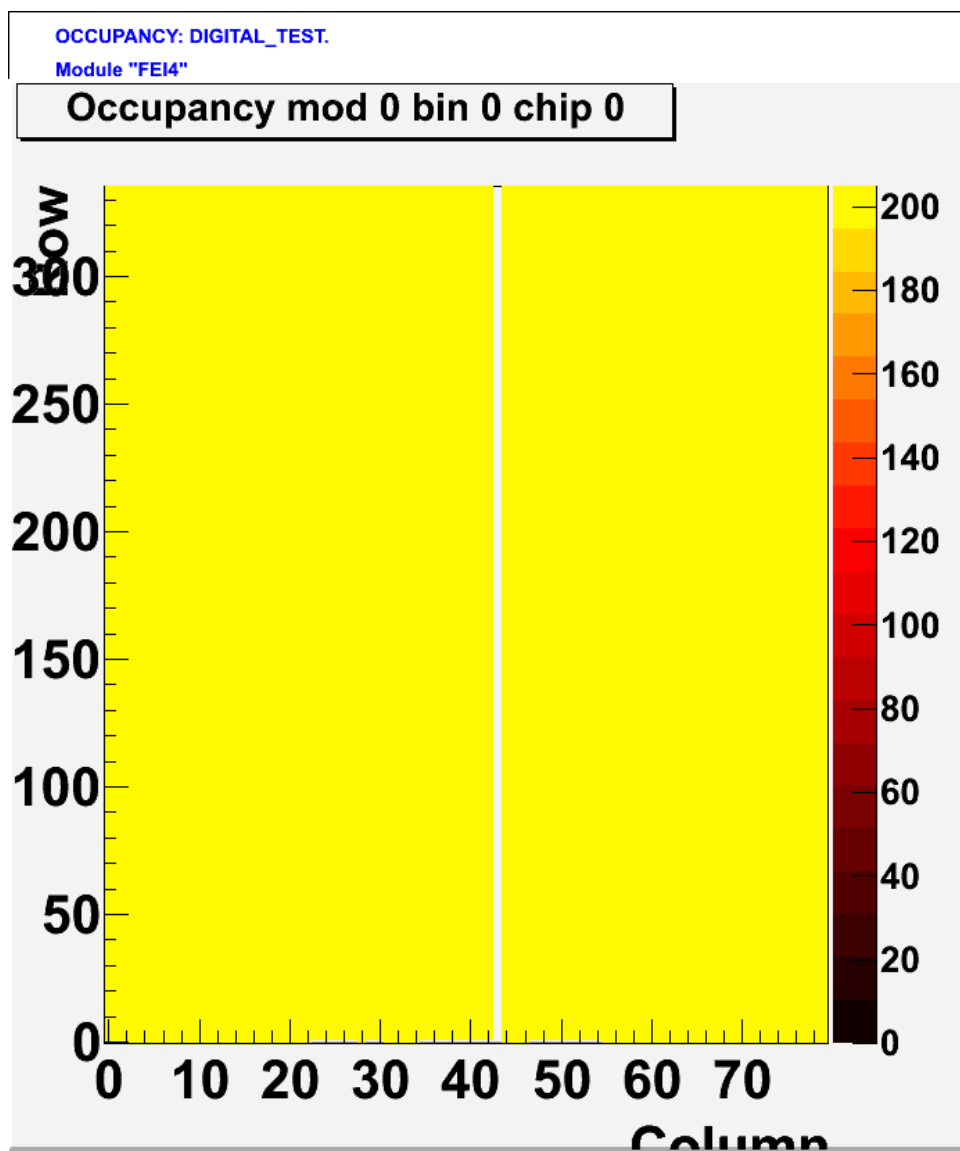


Figure A.1: Reference of a Digital Scan performed with the USBpix system. Column 44 appears completely broken, Most probably due to configuration of the FE-I4 by USBpix.

List of Tables

2.1	The three quark generations of the SM [eaPDG10].	4
2.2	The three lepton generations of the SM [eaPDG10].	4
2.3	The five gauge bosons, mediating the three fundamental forces [eaPDG10].	6
2.4	Laws of conservation for the specific interactions.	7
4.1	Parameters of the inner detector [ATL97a].	21
4.2	Muon spectrometer instrumentation [ATL97b]	24
5.1	Components of the Pixel detector [Fli06].	30
5.2	TX section control parameters	35
6.1	Components of the IBL.[CDE ⁺ 10]	43
6.2	The IBL sensitive region parameters [CDE ⁺ 10].	44
6.3	Basic FE-I4 specifications [FE-11].	45
6.4	FE-I4 trigger and fast commands [FE-11].	47
6.5	FE-I4 slow commands [FE-11].	47
6.6	FE-I4 protocol record words [FE-11].	49
7.1	Format of the data send by the PC via the UART interface.	66
7.2	Control parameters of the TX path.	68
7.3	Supply voltages of the FE-I4	86
7.4	Power consumption for the configured and unconfigured state of the FE-I4.	87
A.1	Preliminary address map of the MEBOC version 4. BCF Part 1. . . .	95
A.2	Preliminary address map of the MEBOC version 4. BCF Part 2. . . .	96
A.3	Preliminary address map of the MEBOC version 4. BMF Part 1. . . .	96
A.4	Preliminary address map of the MEBOC version 4. BMF Part 2. . . .	97
A.5	Detail of the Mux parameter register as referred to in Table 7.2. . . .	98
A.6	Standard configuration of the FE-I4 used in the combined tests with the MEBOC. Part 1.	99
A.7	Standard configuration of the FE-I4 used in the combined tests with the MEBOC. Part 2.	100
A.8	Standard configuration of the FE-I4 used in the combined tests with the MEBOC. Part 3.	101

List of Figures

2.1	Example for an electromagnetic interaction, the e^+e^- pair production.	7
2.2	Transformation of a neutron into a proton via a charged current flavor change, the process responsible for the β -decay of atoms.	8
2.3	Expected and observed confidence levels on exclusion limits of the Higgs boson mass m_H [ATL11a].	9
2.4	Different process cross sections in $p\bar{p}$ and pp collisions. Highlighted are the design energies of the Tevatron and LHC [ATL98].	10
3.1	A schematic of CERN's accelerator complex, showing the LHC and its experiments, but also the injection chain needed for preparation of the particles [Lef08].	14
3.2	Schematic drawing of a LHC dipole element, showing the two beam pipes, the different magnet coils and yoke housed in the helium vessels, machine interconnect and service connections.	14
3.3	By the LHC delivered, integrated luminosity per day.	15
3.4	Cross and long section of the CMS detector. The different colors represent the different parts of the detector. Red: muon chambers, grey: iron yoke, blue: solenoid, yellow: hadronic calorimeter, green: electromagnetic calorimeter, light yellow: tracker [CMS11].	16
3.5	Layout of the ALICE detector, showing the cylindrical detector part and the forward muon spectrometer [Mar06].	17
3.6	Schematic of the LHCb detector, showing its forward geometry with the interaction region at $z = 0$ [Lin08].	18
4.1	Layout of the ATLAS detector, showing the subdetectors and magnets [Peq08c].	19
4.2	Layout of the ATLAS inner detector, composed of the Pixel, SCT and TRT subdetectors [Peq08b].	21
4.3	Calorimeter system of the ATLAS detector, the electromagnetic LAr calorimeter is surrounded by the hadronic Tile calorimeter [Peq08a].	22
4.4	The detection of the various particles in the ATLAS detector [Peq08e].	25
4.5	Layout of the ATLAS trigger system [ATL98].	26
4.6	Integrated luminosity delivered by the LHC and recorded by the ATLAS detector at $\sqrt{s} = 7$ TeV [ATL11b].	27

5.1	Computer generated image of the Pixel detector, opened for a better view [Peq08d].	29
5.2	Exploded view and cross section of a Pixel module [GA08].	31
5.3	Detailed view of the n^+ -in-n pixel sensor and the depletion zone [GA08].	31
5.4	Schematics of the FE-I3 chip electronics [GA08].	32
5.5	Schematic of the Pixel detector DAQ structure [Fli06].	34
5.6	Picture of the BOC hardware layout, the overlaying colors represent the TX section (green), S-Link (yellow), clock and control section (light blue) and RX section (red).	36
5.7	Picture of a TX plugin used on the BOC to BPM encode the TTC signal and convert them into optical signals.	37
5.8	Picture of a RX plugin used on the BOC to receive the optical data stream and convert it into electrical signals.	37
5.9	Picture of the ROD hardware layout, the overlaying colors represent the 8 formatters (red), event fragment builder (green), DSP farm and router (yellow) and control section (light blue).	39
5.10	Schematic of the Pixel detector services organized in 5 patch panel intersections [Lan11].	40
6.1	Vertex reconstruction efficiency in dependence of the number of pile up events for the current Pixel detector with and without the IBL [CDE ⁺ 10].	42
6.2	3D model of the opened IBL, showing the staves mounted in their support structure and the electrical connection to the FEs made by the flex [Lap].	43
6.3	Layout of the IBL in respect to the Pixel detector b -Layer and new beam pipe [CDE ⁺ 10].	44
6.4	Diagram of the FE-I4 functional blocks. The detail shows two rows of a Double Column, a 2×2 pixel array, sharing the Pixel Digital Region [FE-11].	46
6.5	Schematic of the IBL readout system showing the connection of the detector to the off-detector components. Red connections represent optical links.	49
7.1	Picture of the PCB of the first IBL-BOC prototype.	54
7.2	Schematic of the IBL-BOC prototype.	55
7.3	3 types of industrial optical modules, each shown from the electrical connector and the optical connector side. From left to right: SFP, QSFP, Snap12 TX, Snap12 RX.	56
7.4	Picture of the Xilinx SP605 evaluation board [Xil].	57
7.5	Pictures of the two mezzanine daughterboards designed for the SP605.	58
7.6	Coarse block diagram of the firmware logic blocks. Indicating the connection between the three main sections and external devices. . .	58

7.7	Diagram of the Setup Bus FSM in ROD and BOC perspective. The Idle state is shown in green, a write cycle in red and a read cycle in light blue.	60
7.8	Flowchart of the handshake between the ROD and the BOC for a single read and single write cycle.	60
7.9	Schematic of the logical block providing communication via the internal Wishbone Bus. The shown master of the bus is a Setup Bus to Wishbone Bus bridge. As an exemplary slave a register bank is connected to the bus.	61
7.10	Flowchart of the handshake between master and slave for a single read and single write cycle.	62
7.11	Simulation of a read and write request handled by the Wishbone Bus.	63
7.12	Simulation of a read and write cycle from the Setup Bus.	63
7.13	logic analyzer measurement of the communication via the Setup Bus with the Fake-ROD.	65
7.14	Schematic of the logical blocks used for the TX path, consisting of the main encoding and signal adjustment block (green), a testing infrastructure (blue) connected to the Wishbone Bus (orange) and the output buffer of the FPGA(red).	67
7.15	Simulation of the BPM encoding by the use of a 80 MHz clock. Shown are the 40 MHz system clock, 80 MHz BPM clock, the data input and the BPM encoded output (from top to bottom).	68
7.16	Oscilloscope measurement of the BPM encoding, the BPM encoded signal is shown in yellow, the data input in green and the BPM encoded without data input in blue (20 MHz clock).	69
7.17	Block diagram of the MSR adjustment and FineDelay block. The Input and Output are shown in red, the carry chain delay blocks in orange and logical gates in blue.	70
7.18	Simulation of the MSR and FineDelay adjustment.	71
7.19	Measurement of the positive pulsewidth in dependence of the MSR setting before (green) and after (red) placement optimization.	72
7.20	Measurement of the delay and duty cycle in dependence of the FineDelay setting.	73
7.21	Schematic of the logical blocks used for the RX path, consisting of the input buffer with ISERDES (red), the data reconstruction and decoding blocks (blue), the BOC to ROD Mux (yellow) and monitoring blocks (green).	74
7.22	Simulation of the ISERDES output and phase alignment.	74
7.23	Simulation of the Word Alignment and 8b10b decoder block.	75
7.24	Block diagram of the BOC to ROD Mux.	76
7.25	Simulation of the BOC to ROD Mux with the highest possible input data rate.	77

7.26	Experimental setup of the data reception measurement. A external loopback is given by connecting the ends of a pig-tail fiber to each other (TX to RX channels). The data bus to the ROD is analyzed by a logic analyzer attached to one of the breakout connectors.	78
7.27	Logic analyzer measurement of the firmware data reception in a loopback setup. Shown are the transmitted data for each channel and the four signals the ROD data bus.	79
7.28	Finite State Machine of the Monitoring block following the FE-I4 data sequences.	80
7.29	Functional blocks of the software API.	81
7.30	Measured relative error rate of the receiver in dependence of the FineDelay setting or ≈ 300 ps phase shift per setting respectively. . .	83
7.31	Measured relative error rate of the receiver in dependence of the MSR setting.	84
7.32	Measured relative error rate of the receiver in dependence of MSR and FineDelay setting.	85
7.33	Picture of the Single Chip Card connected to a bare FE-I4 via wire bonds.	87
7.34	Digital scan of the FE-I4 chip performed with the MEBOC as the readout system. Shown are the received ToT values after sending the Trigger.	90
A.1	Reference of a Digital Scan performed with the USBpix system. Column 44 appears completely broken, Most probably due to configuration of the FE-I4 by USBpix.	102

Bibliography

- [A⁺09] A Annovi et al. Development of FTK architecture: a fast hardware track trigger for the ATLAS detector. Technical Report arXiv:0910.1126, Oct 2009.
- [ATL97a] ATLAS collaboration. *ATLAS inner detector: Technical Design Report, 1*. Technical Design Report ATLAS. CERN, Geneva, 1997.
- [ATL97b] ATLAS Collaboration. *ATLAS muon spectrometer technical design report*. CERN, 1997.
- [ATL98] ATLAS collaboration. ATLAS trigger performance: Status report. 1998. CERN-LHCC-98-15.
- [ATL99a] ATLAS collaboration. *ATLAS detector and physics performance: Technical Design Report, 1*. CERN, 1999.
- [ATL99b] ATLAS collaboration. *ATLAS detector and physics performance: Technical Design Report, 2*. CERN, 1999.
- [ATL11a] ATLAS collaboration. Combined Standard Model Higgs Boson Searches in pp Collisions at $\sqrt{s} = 7$ TeV with the ATLAS Experiment at the LHC. *ATLAS-CONF-2011-135*, 2011.
- [ATL11b] ATLAS Collaboration. Total Integrated Luminosity in 2011. <https://twiki.cern.ch/twiki/bin/view/AtlasPublic/LuminosityPublicResults>, 8 2011.
- [AXW83] P. A. Franaszek A. X. Widmer. A dc-balanced, partitioned-block, 8b/10b transmission code. *IBM Journal of Research and Development*, 27(5), 1983.
- [BLPW04] J. Butterworth, J.B. Lane, M. Postranecky, and M.R.M. Warren. TTC Interface Module for ATLAS Read-Out Electronics. 10th Workshop on Electronics for LHC and Future Experiments LECC, 2004.
- [Car98] Jean-Luc Caron. LHC quadrupole cross section. AC Collection. Legacy of AC. Pictures from 1992 to 2002., May 1998.

- [CDE⁺10] M Capeans, G Darbo, K Einsweiler, M Elsing, T Flick, M Garcia-Sciveres, C Gemme, H Pernegger, O Rohne, and R Vuillermet. ATLAS Insertable B-Layer Technical Design Report. Technical Report CERN-LHCC-2010-013. ATLAS-TDR-019, CERN, Geneva, Sep 2010.
- [CDF11] CDF, D0 collaboration. Combined cdf and d0 upper limits on standard model higgs boson production with up to 8.6 fb⁻¹ of data. *arXiv*, 2011. arXiv:1107.5518v2.
- [CER10] CERN. LHC luminosity plots for 2010. http://lpc.web.cern.ch/lpc/lumiplots_2010.htm, 2010.
- [CER11] CERN. Lhc luminosity plots for 2011. <http://lpc.web.cern.ch/lpc/lumiplots.htm>, 2011.
- [CMS06] CMS collaboration. *CMS Physics Technical Design Report Volume I: Detector Performance and Software*. Technical Design Report CMS. CERN, 2006.
- [CMS11] CMS collaboration. CMS detector drawings. <http://cms.web.cern.ch/cms/Media/Images/Detector/Detector%20Drawings/index.html>, 2011.
- [com09] SFF committe. Sff-8431 specifications for enhanced small form factor pluggable module sfp+. <ftp://ftp.seagate.com/sff>, July 2009.
- [Dop10] Jens Dopke. Dead Channel Estimate. Presentation during the Pixel Taskforce, August 3rd, 2010.
- [Dop11] Jens Dopke. *Commissioning of the ATLAS Pixel Detector optical data transmission and studies for readout of the ATLAS IBL and future trackers*. PhD thesis, University of Wuppertal, June 2011. To be published.
- [ea04] T. Akesson et. al. Status of design and construction of the transition radiation tracker (trt) for the atlas experiment at the lhc. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 522(1-2):131 – 145, 2004. TRDs for the Third Millenium. Proceedings of the 2nd Workshop on Advanced Transition Radiation Detectors for Accelerator and Space Applications.
- [eaPDG10] K. Nakamura et al. (Particle Data Group). Particle physics booklet. JP G 37, 2010.
- [EB08] Lyndon Evans and Philip Bryant. LHC Machine. *Journal of Instrumentation*, 3(08):S08001, 2008.
- [FE-11] FE-I4 collaboration. The FE-I4 Intgerated circuit guide v11.2. 2 2011.

- [Fey49] R. P. Feynman. Space-time approach to quantum electrodynamics. *Phys. Rev.*, 76(6):769–789, Sep 1949.
- [Fey05] Richard P. Feynman. *The Feynman Lectures of Physics*. Benjamin-Cummings, 2005.
- [fHBS03] LEP Working Group for Higgs Boson Searches. Search for the standard model higgs boson at lep. *Physical Letters B*, 565, July 2003.
- [Fli06] Tobias Flick. *Studies on the Optical Readout for the ATLAS Pixel Detector*. PhD thesis, University of Wuppertal, 2006.
- [GA08] et. al. G. Aad. ATLAS pixel detector electronics and sensors. *Journal of Instrumentation*, 3, 2008.
- [Gre11] P Grenier. Silicon sensor technologies for the atlas ibl upgrade. Technical Report ATL-INDET-PROC-2011-006, CERN, Geneva, Sep 2011.
- [Gro00] CERN Microelectronics Group. PHOS4 - 4 Channel delay generation ASIC with 1 ns resolution, 2000.
- [Hig64] Peter W. Higgs. Broken Symmetries and the Masses of Gauge Bosons. *Phys. Rev. Lett.*, 13(16):508–509, Oct 1964.
- [IEE85] IEEE. *VMEbus Specification Manual*, 1985.
- [Inc09] Zarlink Semiconductor Inc. Parallel fiber optic transceiver (4 + 4) x 3.125 gbps data sheet. ZL60505, July 2009.
- [Jac05] J N Jackson. The atlas semiconductor tracker (sct). *Nucl. Instrum. Methods Phys. Res., A*, 541:89–95, 2005.
- [JJF⁺06] John Joseph, Richard Jared, Douglas P. Ferguson, Andreas Korn, and Trevor Vickey. ATLAS Silicon ReadOut Driver (ROD) Users Manual, 2006.
- [Lan11] Kerstin Lantzsch. *The Finite State Machine for the ATLAS Pixel Detector and Beam Background Studies with the ATLAS Beam Conditions Monitor*. PhD thesis, University of Wuppertal, 2011. To be published.
- [Lap] Didier Laporte. Extraction from a 3d model of the ibl.
- [lar96] *ATLAS liquid-argon calorimeter: Technical Design Report*. Technical Design Report ATLAS. CERN, Geneva, 1996.
- [Lef08] Christiane Lefèvre. The CERN accelerator complex. <http://cdsweb.cern.ch/record/1260465>, Dec 2008.
- [LHC98] LHCb collaboration. *LHCb : Technical Proposal*. Tech. Proposal. CERN, Geneva, 1998.

- [Lin08] Rolf Lindner. LHCb layout. LHCb Collection, Feb 2008.
- [Mar06] Stephane Maridor. ALICE A Large Ion Collider Experiment at CERN LHC : global layout. ALICE Collection, Feb 2006.
- [muo97] *ATLAS muon spectrometer: Technical Design Report*. Technical Design Report ATLAS. CERN, Geneva, 1997. distribution.
- [MW08] Markus Mathes and N Wermes. *Development and Characterization of Diamond and 3D-Silicon Pixel Detectors with ATLAS-Pixel Readout Electronics*. oai:cds.cern.ch:1337735. PhD thesis, Bonn U., Bonn, 2008. Presented 18 Dec 2008.
- [Neu11] Manuel Neumann. Readout components for the insertable b-layer of the atlas pixel detector based on field programmable gate arrays. Master's thesis, University of Wuppertal, 2011.
- [OB97] E. van der Bij O. Byole, R. McLaren. the s-link interface specification. Technical report, CERN, 1997.
- [Ope07] OpenCores Organization. *WISHBONE System-on-Chip (SoC) Interconnection Architecture for Portable IP Cores*, 9 2007.
- [Peq08a] Joao Pequenaio. Computer Generated image of the ATLAS calorimeter. Mar 2008.
- [Peq08b] Joao Pequenaio. Computer generated image of the ATLAS inner detector. Mar 2008.
- [Peq08c] Joao Pequenaio. Computer generated image of the whole ATLAS detector. Mar 2008.
- [Peq08d] Joao Pequenaio. Computer generated images of the Pixel, part of the ATLAS inner detector. Mar 2008.
- [Peq08e] Joao Pequenaio. Event Cross Section in a computer generated image of the ATLAS detector. Mar 2008.
- [PKL⁺98] François Piuz, W Klempt, L Leistam, J De Groot, and Jürgen Schükraft. *ALICE high-momentum particle identification: Technical Design Report*. Technical Design Report ALICE. CERN, Geneva, 1998.
- [Pov09] B. Povh. *Teilchen und Kerne*, volume 8. Springer Verlag, 2009.
- [Sch10] Nicolai Schroer. Study of fpga and gpu based pixel calibration for atlas ibl. ATL-INDET-SLIDE-2010-080, 2010.
- [SNA02] SNAP12 Multi-Source Agreement, Specification Appendix, May 15th 2002.

- [til96] *ATLAS tile calorimeter: Technical Design Report*. Technical Design Report ATLAS. CERN, Geneva, 1996.
- [WH98] Norbert Wermes and G Hallewel. *ATLAS pixel detector: Technical Design Report*. Technical Design Report ATLAS. CERN, Geneva, 1998.
- [Xil] Xilinx. Sp605 evaluation board. <http://www.xilinx.com/products/boards-and-kits/EK-S6-SP605-G.htm>.
- [Xil10a] Xilinx. *UG381 - User Guide: Spartan-6 FPGA SelectIO Resources*, December 2010.
- [Xil10b] Xilinx. *UG526 - User Guide: SP605 Hardware User Guide*, September 2010.
- [Xil11] Xilinx. *DS126 - Spartan-6 FPGA Data Sheet: DC and Switching Characteristics*, September 2011.

Erklärung

Hiermit versichere ich, dass ich diese Arbeit selbstständig verfasst und keinen anderen als die angegebenen Quellen und Hilfsmittel benutzt, sowie Zitate kenntlich gemacht habe. Diese Diplomarbeit ist in keinem weiteren Fachbereich einer wissenschaftlichen Hochschule vorgelegt worden.

Wuppertal, der 28.09.2011

(Timon Heim)
