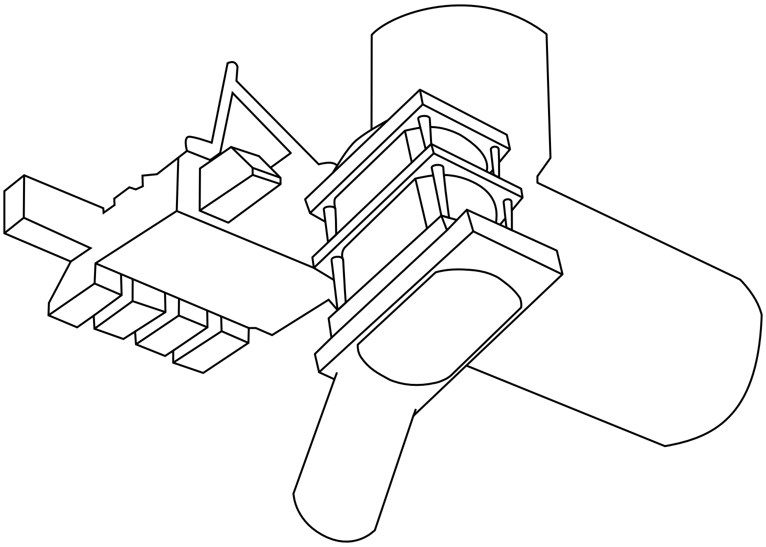
Development of a Cluster Control Board for the JEM-EUSO Mission



Diplomarbeit
eingereicht von

Jörg Bayer

Eberhard Karls Universität Tübingen
Mathematisch-Naturwissenschaftliche Fakultät
Kepler Center for Astro and Particle Physics
Institut für Astronomie und Astrophysik
Abteilung Astronomie

März 2011

Zusammenfassung

Das geplante ‘Extrem Universe Space Observatory’ (EUSO) ist ein Teleskop, welches an das japanische Experimentiermodul ‘Kibo’ (JEM) der Internationalen Raumstation (ISS) angebracht werden soll. Nach dem Start, der für den Anfang des Jahres 2017 geplant ist, wird JEM-EUSO zur Erforschung der energiereichsten Teilchen im Universum beitragen. Wenn solche Teilchen (vorwiegend Protonen, aber auch schwerere Kerne) auf die Erdatmosphäre treffen, interagieren sie mit den Luftmolekülen, wodurch neue Teilchen entstehen und sich eine Teilchen-Lawine mit nahezu Lichtgeschwindigkeit in der Atmosphäre ausbreitet. Das dabei entstehende Fluoreszenzlicht mit Wellenlängen im Bereich von 330 nm bis 400 nm - also im Ultravioletten - wird vom JEM-EUSO Detektor registriert und aufgezeichnet. Die JEM-EUSO Mission und ihre wissenschaftlichen Ziele werden in Kapitel 1 beschrieben.

Um die entstehende Datenmenge zu reduzieren und die gültigen Ereignisse von den ungültigen, beziehungsweise dem Hintergrund, zu trennen, wurde ein mehrstufiges, hierarchisches Filterschema entwickelt. Ein Überblick der technischen Details ist in Kapitel 2 gegeben.

Das Hauptziel dieser Arbeit war es, den Filteralgorithmus der letzten Stufe in Hardware zu implementieren und zu testen. Für diesen Zweck wurde zuerst der Algorithmus an die begrenzten Möglichkeiten von weltraumtauglicher Hardware angepasst und dann mit Hilfe der Hardwarebeschreibungssprache VHDL in einem FPGA[1] implementiert.

Aufgrund der Tatsache, dass die Übertragung der Daten noch ungeklärt ist, was aber eine entscheidende Rolle für die Implementierung des Algorithmus spielt, wurde außerdem ein schnelles Dateninterface entwickelt.

Kapitel 5 beschreibt schließlich die Testumgebungen, welche entwickelt wurden, um die Leistung und Stabilität des Interfaces zu qualifizieren. Dies wurde hauptsächlich durch das Bestimmen der Bitfehlerhäufigkeit in einem Langzeittest erreicht. Zusätzlich wurde in einem weiteren Test die korrekte Implementierung des Algorithmus verifiziert.

[1] Engl. Field Programmable Gate Array: ein konfigurierbarer Schaltkreis für digitale und logische Schaltungen.

Abstract

JEM-EUSO is a large UV telescope that will be deployed in space to investigate the nature and origin of the ultra high energy cosmic rays by observing the fluorescence light produced in extensive air showers. The mission itself and its scientific objectives will be described in Chapter 1.

To reduce the large amount of data produced by the detector and to discriminate the valid events from the invalid ones, a hierarchical trigger scheme over several levels was developed. An overview of the technical aspects of the telescope and its electronics is given in Chapter 2.

The main purpose of this work was the hardware implementation of the trigger algorithm which is foreseen to be used as the last stage in the trigger scheme. As a first step, the algorithm was simplified and adapted to the possibilities of the electronics and then implemented in an FPGA[2] with the help of the hardware description language VHDL.

Additionally, due to the fact that the transmission of the data between the detector electronics and the last trigger stage is not yet defined - which is a crucial part for the implementation of the algorithm - a fast data interface was developed.

Finally in Chapter 5, the test environments are introduced to qualify the performance and the stability of the interface and to verify the correct implementation of the trigger algorithm.

[2] Field Programmable Gate Array: an integrated digital circuit which can be customized to perform logic functions.

Contents

1 The JEM-EUSO Mission

1.1 Overview

The planned Extreme Universe Space Observatory (EUSO) - attached to the Japanese Experiment Module (JEM) ‘Kibo’ of the International Space Station (ISS) - will transform (after its launch early in 2017) the Earth’s atmosphere into a huge detector for the most energetic particles coming from the universe. When such an Ultra High Energy Cosmic Ray (UHECR) interacts with a nucleus in the atmosphere, it produces a cascade of ionized particles and electromagnetic radiation called Extensive Air Shower (EAS). The fluorescence light generated by these EASs during their propagation through the atmosphere will be detected by JEM-EUSO.

The main instrument of JEM-EUSO is a super-wide ±30° Field-of-View (FoV) Ultra-Violet (UV) telescope, which will be able to trace the fluorescence tracks generated by the primary particles with a timing resolution of 2.5 µs and a spatial resolution of 0.1° (corresponding to about 750 m on ground), allowing to reconstruct the incoming direction of the UHECR with an accuracy better than a few degrees (see Figure 1.1).

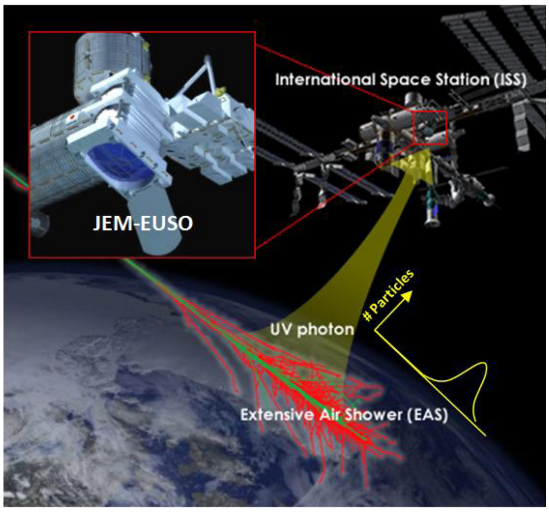


Figure 1.1: The observation principle of JEM-EUSO: Mounted on the ISS, the main instrument will monitor the UV-light produced by extensive air showers which are induced by Cosmic Rays (CRs) (Image: JEM-EUSO Collaboration).

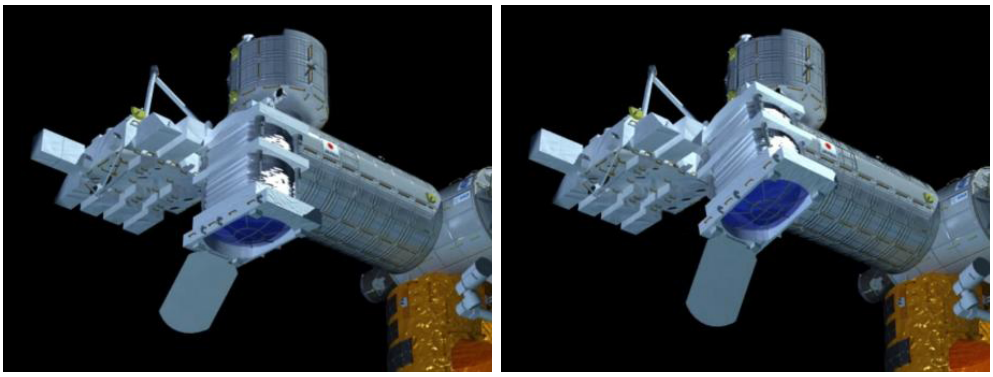


Figure 1.2: The JEM-EUSO telescope in nadir mode (left) can be tilted (right) to increase the effective area (Image: JEM-EUSO Collaboration).

JEM-EUSO is the successor of EUSO, which was originally selected as a mission mounted onto the European Columbus module of the ISS by the European Space Agency (ESA). In July 2004 the EUSO Phase-A study was successfully completed but due to financial problems within ESA and the European countries, the start of the Phase-B study was postponed. EUSO was then redefined by the Japanese and U.S. teams as a mission attached to the Japanese Experiment Module and renamed JEM-EUSO. Based on improvements of the detector (higher quantum efficiency) and of the triggering algorithm, the energy threshold for the primary particles has been reduced down to 5×10^{19} eV. Additionally, by introducing the possibility of tilting the instrument to about 30° (the so called 'tilted mode' in contrast to the 'nadir mode', see also Figure 1.2) it is possible to increase the effective area of the telescope by a factor of approximately 2.4 (from around $1.2 \times 10^5\,\mathrm{km}^2$ in nadir mode to roughly $2.9 \times 10^5\,\mathrm{km}^2$ in tilted mode) [1].

1.2 Science Objectives

JEM-EUSO is devoted to problems in fundamental physics and high energy astrophysics and primarily will explore the most energetic component of the Cosmic Ray (CR) spectrum for energies $>5 \times 10^{19}$ eV, where the flux is around a few particles per km^2 and century (see Figure 1.3).

As the EASs will be observed from space, the target volume used by the detector exceeds the one available to ground based experiments which results in higher statistics for those rare events. Besides this main objective JEM-EUSO will also contribute to the

[1] These values are depending on the height of the ISS and were calculated for a height of 350 km (Shinozaki, 2010).

investigation of other phenomena intrinsic to the Earth's atmosphere such as Transient Luminous Events (TLEs) and meteoroids (see Subsection 1.2.2).

1.2.1 Astronomy and Astrophysics through the Particle Channel

Cosmic Rays with an energy greater than 10^{20} eV were observed first in 1962 by John Linsley with the 'Volcano Ranch'-experiment[2] in New Mexico (Linsley, 1963). Since then, scientists discovered lots of such events - including the 'Oh-My-God'- particle, recorded in 1991 by the 'Fly's Eye'-Detector[3] in Utah with an energy of 3.2×10^{20} eV (Bird et al., 1995) which was identified as a proton.

The existence of particles with energies higher than 4×10^{19} eV (commonly called UHECRs) is therefore today beyond controversy, but raises questions such as:

- what are the sources?
- what is the mass composition (e.g. protons, iron nuclei)?
- what are the acceleration mechanisms?
- how do they propagate through the interstellar medium?

Due to the very low flux of the UHECRs - a few particles per km^2 and century (see also Figure 1.3) - the experiments have to monitor an area as large as possible to attain some significant statistics. The largest currently operating experiment is the Pierre Auger-Observatory in Mendoza, Argentina which covers an area of around $3000\,\mathrm{km}^2$ (see Figure 1.4 for a comparison of the exposures[4] of different experiments). As it is difficult to achieve areas larger than one order of magnitude with ground based experiments (besides the disadvantage of only being able to monitor a part of the sky), JEM-EUSO will be deployed in space. Due to accurate measurements of the primary energy, of the arrival direction, of the composition of the UHECR and with its high statistics (it is expected to record hundreds of events per year with an energy $>5 \times 10^{19}$ eV) JEM-EUSO will therefore contribute to the answers of the questions above.

1.2.1.1 Propagation of Cosmic Rays on their way to Earth

To understand how CRs propagate through the galactic magnetic field, we introduce the Larmor-radius, which describes the radius of the circular motion of a charged particle

[2] The MIT Volcano Ranch Air Shower array was composed of 19 scintillation detectors arranged on a surface of $2\,\mathrm{km}^2$ to detect the secondary particles coming from the EAS (Linsley et al., 1961).

[3] Fly's Eye is a detector composed of 67 fluorescence telescopes (therefore using the same observation principle as JEM-EUSO), pointing at different locations in the sky to provide a full sky coverage.

[4] The unit for the exposure is named in honor of John Linsley and is given by the product of measuring surface, solid angle and observation time: $1\,\mathrm{Linsley} = 1\,\mathrm{L} = 1\,\mathrm{km}^2 \cdot \mathrm{sr} \cdot \mathrm{yr}$.

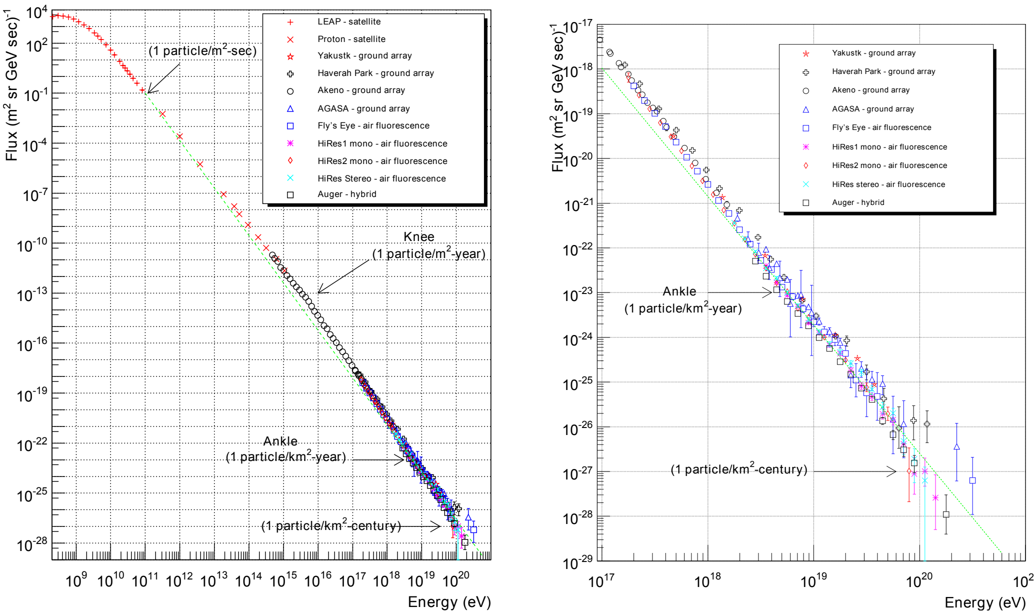


Figure 1.3: The all particle CR-spectra from the main experiments with the main features ('Knee' and 'Ankle'), where the slope of the spectra changes. As can be seen from the high-energy part (right) the statistic is very low (Hanlon, 2008).

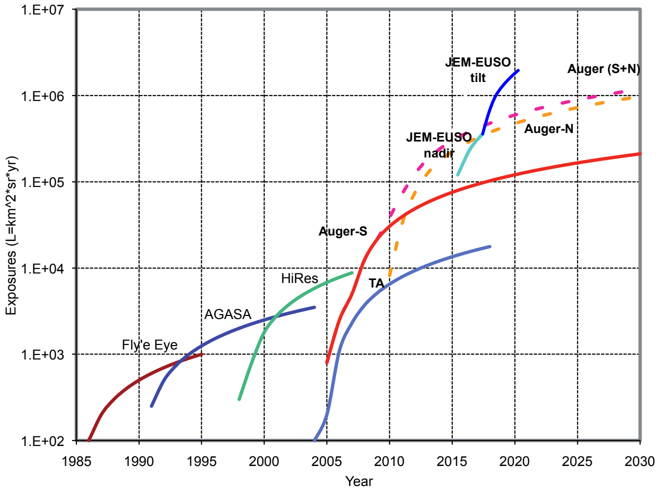


Figure 1.4: The exposure of various experiments: JEM-EUSO in nadir and later in tilted mode will have a much higher exposure than the currently operating largest experiment (Auger South) - Auger North is now postponed (Image based on: JEM-EUSO Collaboration, 2010).

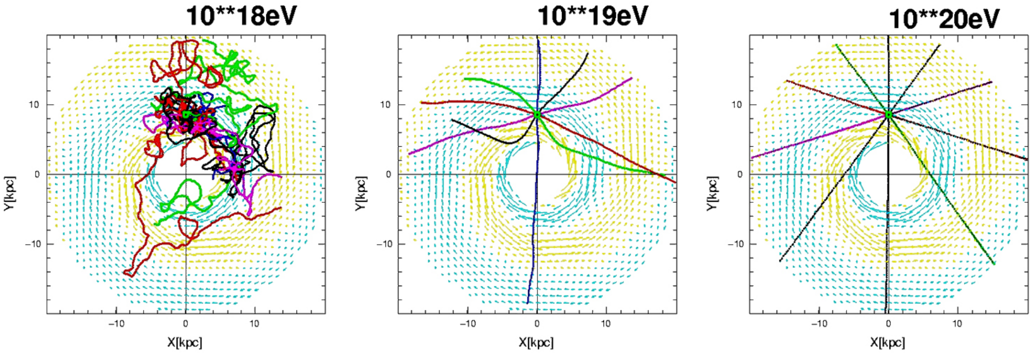


Figure 1.5: Simulated propagation of a proton, injected from Earth into the galactic magnetic field. At low energies, the particle will be trapped inside the galaxy - but with increasing energy, the deflection caused by the magnetic field gets weaker (Image: courtesy of M. Teshima).

passing through a (uniform) magnetic field

$$r_{\text{Larmor}} = \frac{E}{ZeB} \simeq \frac{1}{Z} \cdot \left(\frac{E}{10^{18}\,\text{eV}}\right) \cdot \left(\frac{\mu\text{G}}{B}\right) \text{kpc} \tag{1.1}$$

with E being the energy of the particle, Z the atomic number, e the elementary charge and B the magnetic field. The approximation was made for typical astronomic values and is used to estimate the magnitudes (see Table 1.1). If r_{Larmor} (respectively the energy) is small compared to the dimensions of the galaxy[5] the particle will be trapped within the magnetic field of the galaxy. On the other hand, if r_{Larmor} is comparable to the dimensions of the galaxy, the particle could escape and propagate through the intergalactic medium (see Figure 1.5), which happens at energies

$$E \gtrsim ZeB \cdot r_{\text{Galaxy}} \simeq 2.7 \times 10^{19}\,\text{eV} \cdot Z \left(\frac{B}{3\,\mu\text{G}}\right) \cdot \left(\frac{r_{\text{Galaxy}}}{10\,\text{kpc}}\right). \tag{1.2}$$

This means that, at energies where JEM-EUSO will observe, the particles will not be strongly deflected and we could have the possibility to identify some sources by looking

Table 1.1: Larmor-radius for a proton at different energies, propagating in the galaxy.

Energy	10^{18} eV	10^{19} eV	10^{20} eV
r_{Larmor}	0.36 kpc	3.6 kpc	36 kpc

[5] The radius of the Milky Way is around 20 kpc and the magnetic field a few µG (Tanco et al., 1998).

at the arrival direction of the UHECR.

However, if their energy is high enough, UHECRs could interact with the photons of the Cosmic Microwave Background Radiation (CMBR). This energy limit was independently calculated by Kenneth Greisen (Greisen, 1966), Vadim Kuz'min and Georgiy Zatsepin (Zatsepin and Kuz'min, 1966) in 1966 (also known as the 'Greisen-Zatsepin-Kuzmin (GZK) cut-off') and gives a theoretical upper limit on the energy of CRs from distant sources. The reason for this is, that a high energy proton could be raised to its first excited state by a photon from the CMBR and then will decay to a pion and a nucleon

$$\begin{aligned} p + \gamma_{2.7\,\mathrm{K}} \rightarrow \Delta^{+} &\rightarrow p + \pi^{0} \\ &\rightarrow n + \pi^{+}. \end{aligned}$$

Due to this effect, the primary particle will lose energy until its energy is below the threshold for pion production which lies approximately at

$$E_{\mathrm{th}}^{\pi} \simeq 5 \times 10^{19}\,\mathrm{eV}.$$

Another loss of energy (but not that effective at the highest energies) could happen due to pair production

$$p + \gamma_{2.7\,\mathrm{K}} \rightarrow p + e^{+}e^{-}$$

for which the energy threshold is around

$$E_{th}^{e^{+}e^{-}} \simeq 7 \times 10^{17}\,\mathrm{eV}.$$

Taking the density of the CMBR photons[6] and the cross section[7] of the proton-photon interaction into account, it can be calculated that a proton will drop down to an energy of 5×10^{19} eV after around 100 Mpc nearly independent of its energy before the interaction (see Figure 1.6, left). Therefore, a steepening in the extra-galactic proton spectra is expected - a weak suppression of the flux at energies around 7×10^{17} eV and a strong one around 10^{20} eV (see Figure 1.6, right).

1.2.1.2 Origin of UHECR

Another reason for the steepening of the UHECR spectrum could occur due to the limit of the mechanism which accelerates the particles (also referred as 'bottom-up' model). For this mechanism, the maximum achievable energy of the particles can be estimated

[6] $\rho_{\gamma_{2.7\,\mathrm{K}}} \approx 410\,\mathrm{cm}^{-3}$ (Bennett et al., 2003)

[7] The mean cross section is about 200 µb (Greisen, 1966).

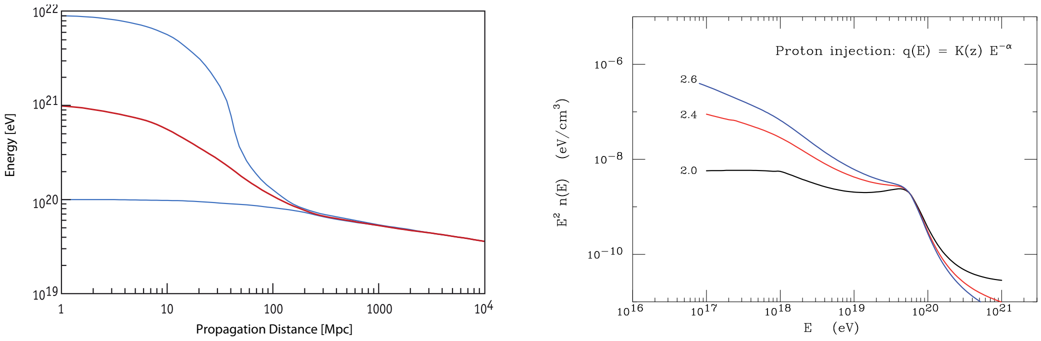


Figure 1.6: Due to the interaction with the photons from the CMBR a proton will drop to an energy around 10^{20} eV within 100 Mpc (left image based on: Aharonian and Cronin, 1994) - therefore, a steepening in the spectrum is expected at 7×10^{17} eV caused by the pair-production and at 10^{20} eV due to the pion-production (right image based on: Lipari, 2008).

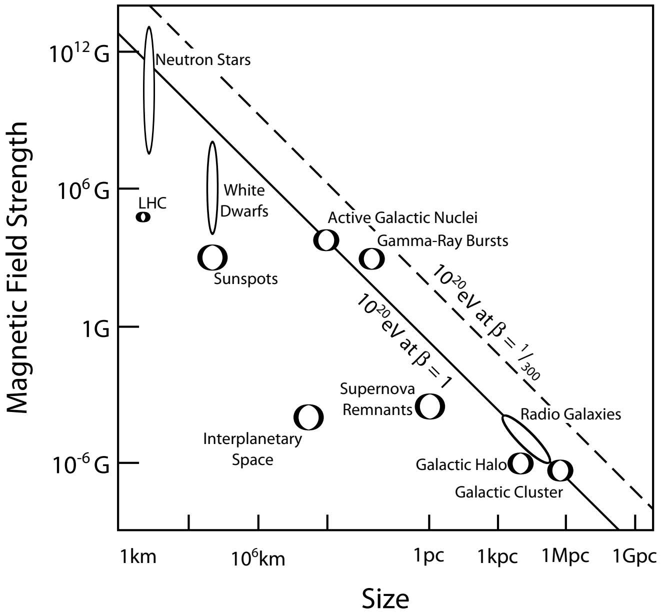


Figure 1.7: Possible sources for particle acceleration, plotted over size and their magnetic field. Sources below the diagonal line are not able to accelerate a proton over an energy of 10^{20} eV - the dashed line accounts for β, which can be seen as the effectiveness of the acceleration mechanism (Image based on: Hillas, 1984 and JEM-EUSO Collaboration, 2010).

by their Larmor-radius within the accelerator region - as they have to stay inside - and is therefore proportional to the charge of the particle and the strength and size of the magnetic field of the accelerator (see Figure 1.7). The known sources which could be able to accelerate a proton up to an energy of 10^{20} eV are

- neutron stars,
- jets from active galactic nuclei,
- gamma-ray bursts,
- radio galaxies and
- galactic clusters.

As all these sources are very close to the theoretical limit for an acceleration up to 10^{20} eV, again a steepening in the spectrum is expected (in addition to the GZK cut-off). On the other hand, if this limit will not be seen, the existence of unknown sources in the top-right area of Figure 1.7 is also possible as there is no theoretical reason, that such sources do not exist.

In the second commonly proposed model on the origin of UHECR (top-down model), the UHECR are generated by the decay or annihilation of Super-Heavy Particles (SHPs) with masses of around 10^{22}-10^{25} eV/c^2.

To distinguish between different sources and models, an analysis of the arrival direction and an accurate flux measurement is necessary, which will be provided by JEM-EUSO with high statistics and full sky coverage.

1.2.2 Exploratory Objectives

Besides the main objectives, JEM-EUSO will also be able to account for the exploration of other phenomena occurring in the Earth's atmosphere. A small summary will be given in the following

- Detection of extreme energy gamma rays:
  The shower profile of an EAS which is induced by an UHECR is different from the one induced by an extreme energy gamma ray and therefore can be distinguished. As the cross section for bremsstrahlung and pair-production decreases for gamma rays due to the Landau-Pomeranchuk-Migdal effect, the shower profile gets elongated. Additionally, the extreme energy gamma rays can interact with the magnetic field of the Earth and produce positron-electron pairs. As this happens long before reaching the upper atmosphere (around 1000 km above the ground), the maximum of the shower will occur much earlier. JEM-EUSO will be able to detect more than 3000 events above 5×10^{19} eV in its lifetime. For more details on the proton-gamma discrimination from space observations see (Supanitsky and Medina-Tanco, 2011).

- Detection of extreme energy neutrinos:
  Just like for gamma ray events, the EASs from neutrinos can be clearly identified by their shower profile. Due to the low neutrino-nucleus interaction cross section, the EAS will develop deep in the atmosphere or be recognized as an upwards-going shower for an interaction with the Earth's crust. For more details on the detection of extreme energy neutrinos with JEM-EUSO see Bittermann (2010).

- Study of the galactic magnetic field:
  If point sources for events above 10^{20} eV are seen, it will be possible to identify other events from these sources with lower energies. Based on their arrival direction, and from their deflection, the Galactic Magnetic Field (GMF) could be determined (see also Section 1.2.1.1).

- Verification of the relativity and quantum gravity effects:
  With the possibility to investigate energies, which are more than three orders of magnitude higher than the ones reached in the Large Hadron Collider (LHC), the Lorentz invariance can be probed at Lorentz factors up to $\gamma \approx 10^{11}$ and quantum gravity theories can be tested.

- Observation of nightglows, Transient Luminous Events and meteors:
  Nightglow is a phenomenon at an altitude of around 95 km and is observed as light emitted by OH-molecules which is formed as stripes with a width of around 40 km (see Figure 1.8).

  Transient Luminous Event is a general term for various discharge phenomena associated with lightning, which are developing from the top of thunderclouds up to an altitude of around 100 km (see Figure 1.9).

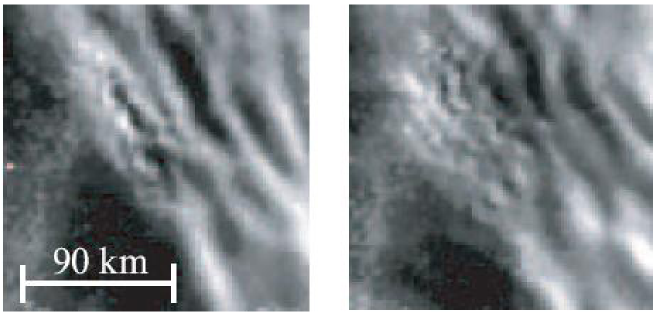


Figure 1.8: Picture of the nightglow from OH-molecules with 40 km wide stripes (Image: JEM-EUSO Collaboration).

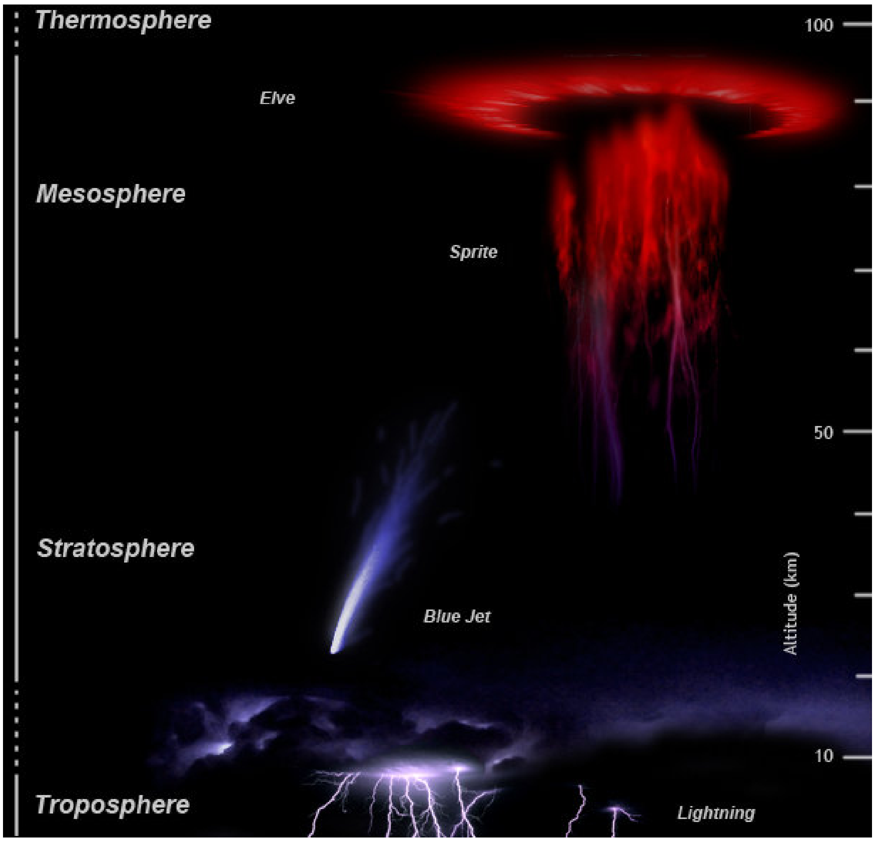


Figure 1.9: Artistic view of various Transient Luminous Events (Image: Abestrobi).

1.3 Observation Principle and Requirements

1.3.1 Observation Principle

As already mentioned, an UHECR which reaches the Earth will interact with nuclei from the atmosphere. This first interaction will produce secondary particles, which again interact with the atmosphere to produce more particles. Therefore, an UHECR will initiate a chain of interactions where the amount of secondary particles (which is proportional to the energy of the primary particle) rapidly increases up to a maximum and will then decrease when no more secondary particles can be produced. The particles of such a shower are forming a thin disk, which travels at nearly the speed of light through the atmosphere.

Most of the produced particles are electrons, which are able to excite the molecules (mainly nitrogen) of the atmosphere and therefore produce fluorescence light - for this reason, the amount of light is also proportional to the energy of the UHECR. The dominant wavelength of the emitted light is located in the range from 330 nm to 400 nm and therefore in the UV band.

Besides the fluorescence light, an EAS will also produce Cherenkov light due to the fact, that a large quantity of secondary particles travel faster than the speed of light within the atmosphere.

By monitoring the atmosphere from space, JEM-EUSO will detect the fluorescence light as a small spot moving at nearly the speed of light on a straight path. While the shower reaches its maximum, the luminosity of the spot will also increase and then fade out (see also Chapter 3). Based on the recorded light profile and the movement of the spot over time, the event can be reconstructed in terms of arrival direction, type and energy of the primary particle. For more details on the event reconstruction with JEM-EUSO see Mernik (2009).

1.3.2 Requirements

The following list will give an overview of the mission requirements (JEM-EUSO Collaboration, 2010). The requirements of the subsystems are given in the accordant subsections of Chapter 2.

Science requirements:

SR1: Specification of UHECR origin by arrival direction analysis with determination accuracy better than a few degrees

SR2: Determination of trans-GZK structure in cosmic ray energy spectrum

SR3: UHECR primary identification capability: discriminating among nucleus, gamma ray and neutrino

SR4: Observation capability of TLEs

Observation requirements:

OR1: Observation area: $\geq 1.3 \times 10^5 \left(\frac{H_{orbit}}{400\,\mathrm{km}^2}\right)\,\mathrm{km}^2$

OR2: Arrival direction determination accuracy: $\leq 2.5°$ for $\mathrm{E} = 10^{20}$ eV and 60° zenith angle

OR3: Energy determination accuracy: $\leq$30% for $\mathrm{E} = 10^{20}$ eV and 60° zenith angle

OR4: X_{max} determination accuracy: $120\,\mathrm{g\,cm^{-2}}$ for $\mathrm{E} = 10^{20}$ eV and 60° zenith angle

OR5: Energy threshold: 5.5×10^{19} eV

OR6: Monitoring the average signal rate for all pixel every 3.5 s

OR7: Capability of observing TLE with time scales shorter than 1 s

Instrument requirements:

IR1: Observation duty cycle: $\geq$ 17%

IR2: Instrumental duty cycle: $\geq$ 30%

IR3: Lifetime: longer than 3 years. Two-year storage is also required before the launch

IR4: Attitude determination accuracy: $\leq$ 0.05° (this may be achieved by offline analysis)

IR5: Precision of absolute time: $\leq$ 1 µs

IR6: Wavelength band: covering major fluorescence lines in 330 nm to 400 nm

IR7: Synchronization precision among subsystems: $\leq$ 200 ns

IR8: Light shielding against stray light: less than GHz level

2 Technical Aspects of JEM-EUSO

2.1 Overview of the JEM-EUSO System

As already mentioned in Section 1.1, JEM-EUSO is the successor of EUSO and is now led by the Japan Aerospace Exploration Agency (JAXA). Based on technological advances, the JEM-EUSO design is an improved version of the EUSO design with the aim to reduce the threshold energy and to increase the effective area.

The telescope is mainly composed of the optics for collecting and focusing the UV photons, the focal surface detector to detect the light, the electronics for read-out, controlling and monitoring - and the mechanical structure (see Figure 2.1). Besides these main components there also will be an Atmospheric Monitoring System (AMS) - basically an Infra Red (IR) camera and a LIght Detection And Ranging (LIDAR)-unit - to gather information about the composition of the atmosphere which will be used for the reconstruction of the EAS.

In the following chapters, some of the sub-systems will be introduced in more detail, depending on the relevance for this work.

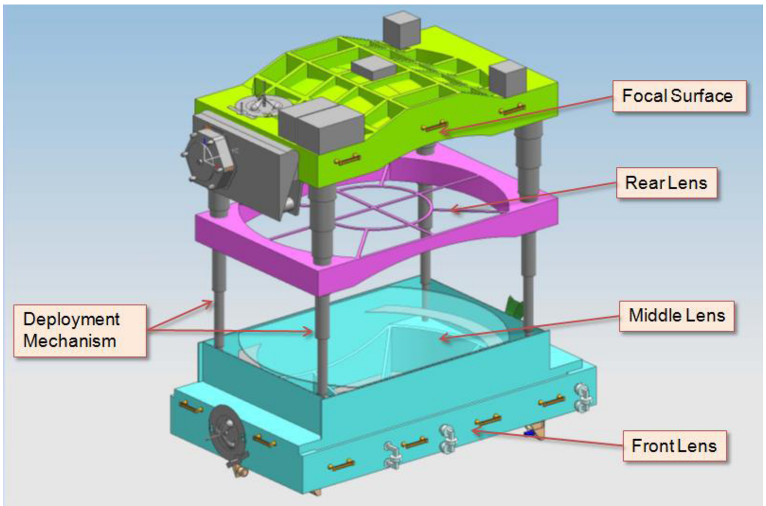


Figure 2.1: The main instrument consists of the optics (front-, middle- and rear-lens), the detector with its electronics (focal surface) and the mechanical structure including the deployment mechanism (Image: JEM-EUSO Collaboration).

2.2 Optics Subsystem

2.2.1 Introduction

In comparison to the old two lens design of EUSO, the new Optics Module (OM) contains now three lenses - two curved double-sided Fresnel lenses and one intermediate curved precision Fresnel lens to correct the chromatic aberration.

In the current 'Baseline' design all three lenses are made out of PolyMethyl-MethAcrylate (PMMA) which is widely used in space environments. In parallel, an 'Advanced' design is being studied, where the front lens is made out of Amorphous Peffluoro Alkenylvinylether (CYTOP) and the middle and rear lens remains PMMA (also referred to as 'CPP'-design[1]). CYTOP was never used in space but has better performance than PMMA (e.g. better transparency and lower chromatic aberration).

2.2.2 Design and Performance

Both designs for the Optics Module are using a 'side-cut' configuration where the lenses are no longer round (compared to the EUSO design with 2.5 m diameter). The lenses have now a diameter of 2.65 m with two parallel sides cut to 1.9 m in order to utilize the complete available room within the H-II Transfer Vehicle (HTV)[2].

The cross sections of the two designs are given in Figure 2.2 where one can see the arrangement of the three lenses and the focal surface (right most curve). Due to the fact, that the iris in the Baseline design is directly in front of the first lens (the two small horizontal lines), the most outer part of this lens will not be used which reduces the mass by 9 kg. In addition, as the density of CYTOP is nearly twice the one from PMMA, the overall mass of the Baseline design is lower compared to the Advanced design. Besides this disadvantage, the CPP design has better optical characteristics as can be seen from Figure 2.2 and 2.3. In Table 2.1, an overview of the requirements and the current design parameters is given.

As the available storage room within the HTV is limited and especially the maximum height (around 2 m) is too small to accommodate the telescope (nearly 4 m), it is foreseen to use extendable masts which will be retracted during transportation to the ISS (the so called 'stowing configuration') and then extended to full size after the telescope is mounted on the Exposed Facility (EF). As this could introduce some uncertainties for the final lens positions (which in turn will reduce the optical performance), it is planed to have some mechanics to correct for displacement.

[1] CPP stands for the materials the lenses are made of: CYTOP-PMMA-PMMA for the Advanced design and PPP respectively for the Baseline design.

[2] The HTV is an unmanned spacecraft used to resupply the JEM and will transport JEM-EUSO to the ISS.

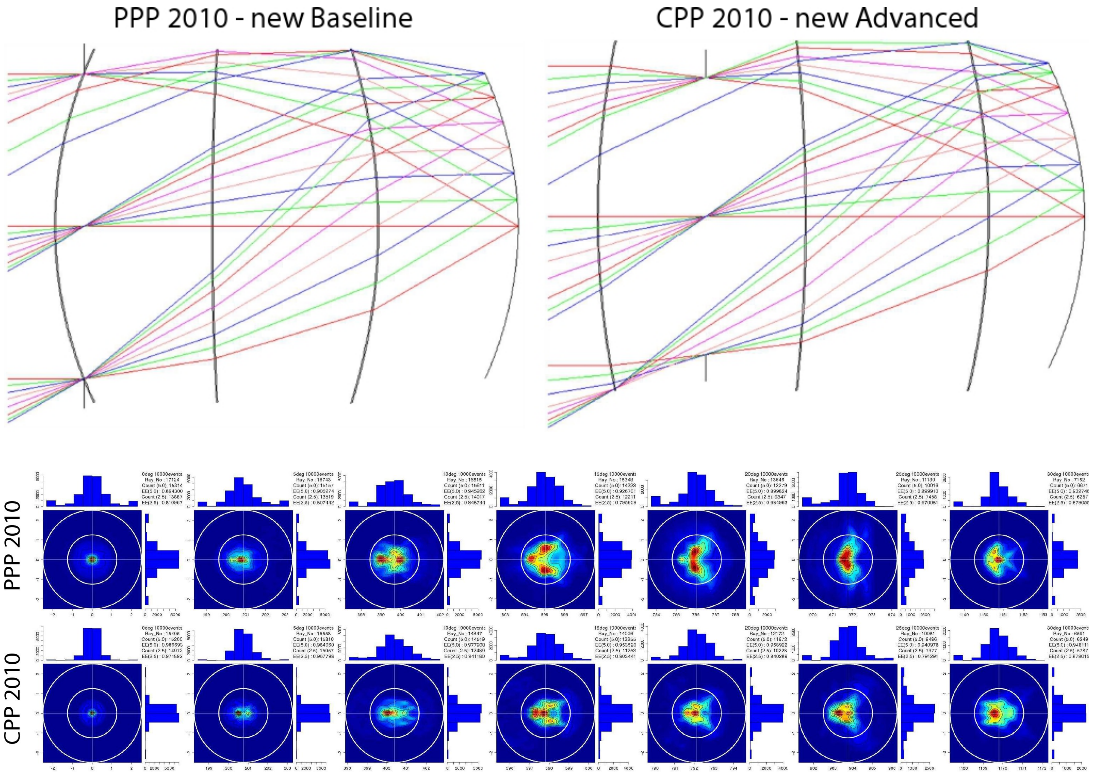


Figure 2.2: Cross sections of the current Baseline and Advanced optics design (including ray tracing simulation) and spot the sizes for 0° to 30° in 5° steps. The diameters of the circles in the spot-size graphics are 2.5 mm for the inner and 5 mm for the outer circle (Image: courtesy of Zuccaro Marchi and Takizawa).

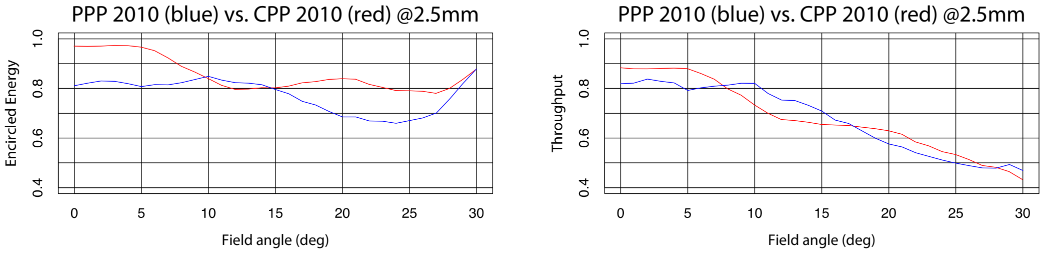


Figure 2.3: Comparison of the encircled energy (left image: courtesy of Zuccaro Marchi and Takizawa) and the throughput (right image based on: JEM-EUSO Collaboration) between the Baseline (blue) and the Advanced (red) design (both for a spot size of 2.5 mm).

Table 2.1: Overview of the requirements and the current design parameters of the OM. The lens diameter of the first lens in the Baseline design can be reduced down to 2.3 m due to the iris.

	Requirement	Baseline	Advanced
f/# (F number)	< 1.25	1.0	1.0
Lens diameter	≥ 2.5 m	(2.3 m) 2.65 m	2.65 m
Spot size (RMS)	≤ 5 mm	≤ 2.2 mm	≤ 1.6 mm
Throughput	50%@0°-10° 40%@10°-20° 30%@20°-30°	59%@0°-10° 52%@10°-20° 39%@20°-30°	62%@0°-10° 58%@10°-20° 42%@20°-30°
Mass	≤264 kg	154 kg	202 kg

2.3 Focal Surface Subsystem

2.3.1 General

The Photo-Detectors (PDs) together with the read-out electronics and the mechanical structure form the Focal Surface-Subsystem which is located in the focal surface of the optics. Depending on the design of the Optics Module the focal surface has a diameter of 2.4 m and a spherical curvature of 2.785 m for the Baseline design and 2.44 m, 2.625 m respectively for the Advanced design.

Over 5000 MAPMTs with 64 channels each, were arranged on the focal surface and clustered in a hierarchical way (see also Figure 2.6). Four MAPMTs form the so called Elementary Cell (EC) and nine ECs constitute the Photo-Detector Module (PDM) as can be seen in Figure 2.4.

The main requirements for the detector are (JEM-EUSO Collaboration, 2010):

- The Focal Surface (FS) shall be capable to detect the EAS by observing the fluorescence light produced during the EAS development and the Cerenkov light.
- The FS shall be capable to determine the position of the arriving photons as a function of time and to follow the space-time development of the EAS.
- The FS shall have single-photon sensitivity in the 330 nm to 400 nm wavelength range to detect EAS as faint as possible.
- The FS shall have fast response (well below 0.1 µs) to follow the space-time development of the EAS.
- The FS pixel FoV shall be not greater than $\Delta\alpha = 0.1°$.
- Cross-talk between neighboring pixels shall be less than 10 % (TBC).

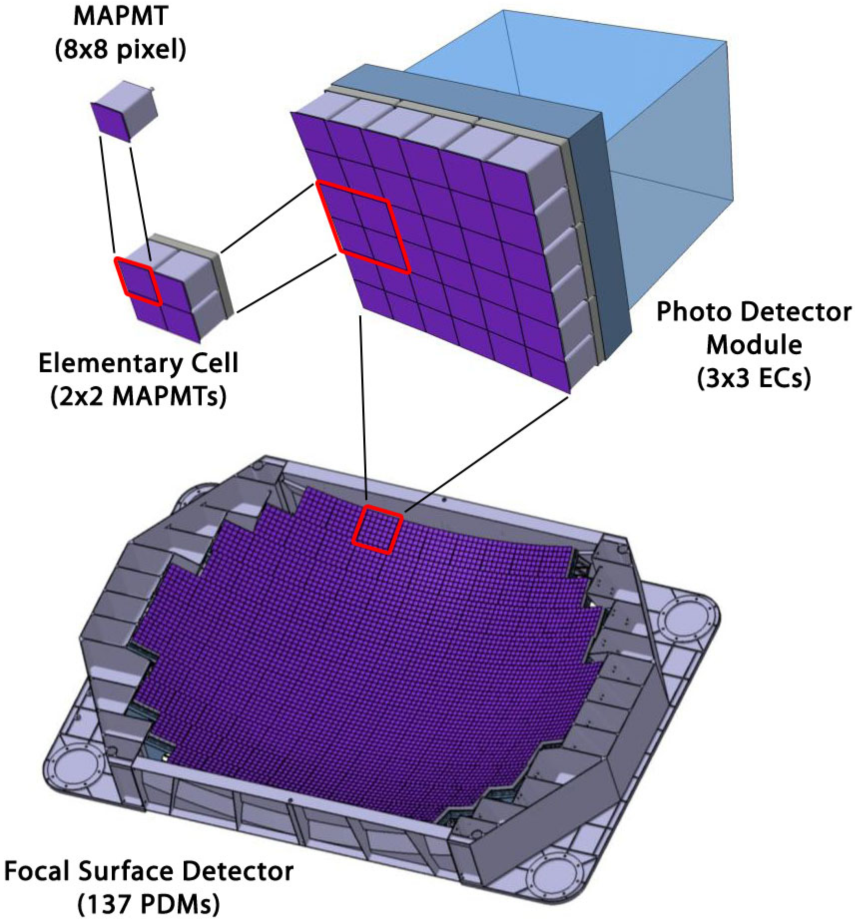


Figure 2.4: The modular concept of the Focal Surface: Four MAPMTs were grouped in the Elementary Cell and nine ECs constitute the Photo-Detector Module which are arranged on the Focal Surface (Image based on: JEM-EUSO Collaboration).

- The noise rate of the electronics shall be two orders smaller than the rate of the night air glow background.
- The overall detection efficiency shall be good and uniform.
- The overall detection efficiency, averaged over all the FS, shall be $\epsilon_{PD} \geq 0.12$.
- The FS shall cover the optical focal surface with a sensitive area as large as possible by reducing dead or inefficient areas.
- The FS shall have low sensitivity to magnetic fields of the order of magnitude of 1 G.

- The FS shall be highly reliable and stable over at least 5 years.
- The FS shall be compatible with the requirements imposed by a Space Mission.

2.3.2 Focal Surface Detector

The Multi Anode PhotoMultiplier Tube (MAPMT) used as basic detector element was developed by RIKEN[3] in collaboration with Hamamatsu Photonics K.K. and has 64 channels in an 8x8 array. Compared to the one selected for EUSO, it was possible to improve the performance of the new MAPMT (see Figure 2.5) - the quantum efficiency is 1.75 times higher, the sensitive area by a factor of 1.9 larger, the total detection efficiency by a factor of 3 higher and the cross-talk between the channels has been reduced. The main specifications of the R11265-M64 MAPMT are given in Table 2.2.

Table 2.2: Main specifications of the R11265-M64 MAPMT (JEM-EUSO Collaboration, 2010)

Specification	
pixel size	2.88 mm
pixel pitch	2.88 mm
sensitive area	23.04 mm x 23.04 mm
physical dimensions	26.2 mm x 26.2 mm x 20.25 mm
mass	27.3 g
wavelength	330 nm to 400 nm
quantum efficiency	$> 35\%$ (maximum 40%)
gain	order of 10^6 at 0.9 kV
anode pulse rise-time	about 1.5 ns
cross-talk between pixels	about 1%
operating temperature	−10 °C to 30 °C
magnetic field effects	0.1 relative gain variation at 2 G
gain degradation after 3(+2) years operation	81%(78%)
reliability for 3(+2) years operation	99.1%(98.5%)

[3] RIKEN is a large natural sciences research institute with approximately 3000 scientists on seven campuses across Japan, founded in 1917.

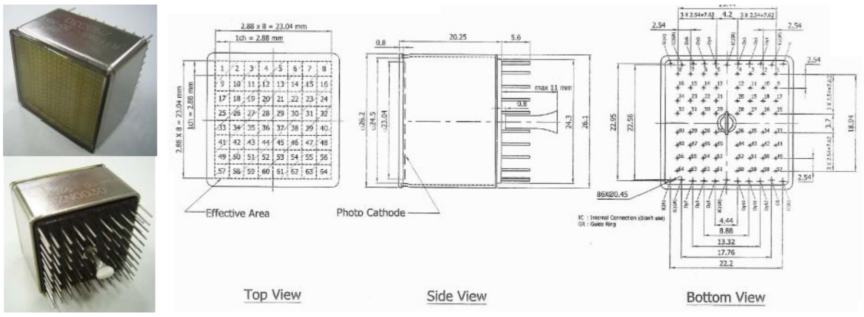


Figure 2.5: The MAPMT for the detector was developed by RIKEN in cooperation with Hamamatsu Photonics K.K. (Image: JEM-EUSO Collaboration).

2.3.3 Focal Surface Electronics

The main purposes of the FS electronics are (1) to read out the MAPMTs, (2) provide the analog to digital conversion, and (3) to reduce/store the scientific data. Besides these components additional electronics is needed to control and monitor the status of the whole system.

The electronics for data generation is divided into the Front End Electronics (FEE), which performs the analog to digital conversion, the PDM electronics (first stage of data reduction) and the cluster control electronics (second stage of data reduction).

The control/monitor electronics include (JEM-EUSO Collaboration, 2010):

- ISS (JEM) communication interface:
  handling the JEM-EUSO communication for commands and data collection with the external world

- power supply and ISS (JEM) power interface:
  generates the necessary high and low voltages

- FS assembly communication interface:
  controls the link between FS assembly and control electronics for command and data

- Scientific functions management system:
  includes the science relevant hard- and software (e.g. data acquisition system and the on board storage capability)

- Housekeeping (HK) collection management system:
  collects regularly the engineering HK data from various locations within the instrument

- LIDAR and IR camera management system:
  manages the whole atmospheric sounding system (including power supply, configuration commands issuing, data retrieval and data storage)
- Thermal control management system:
  providing the temperature sensor read out and heaters activation
- Calibration, monitoring and alignment devices management system (TBD)
- Lid mechanism management system:
  responsible for the proper motor drive control and the position sensors to operate the lid
- Management of the system for the orientation of the JEM-EUSO Instrument (TBC):
  determines the instantaneous absolute orientation of JEM-EUSO in the case that the standard resources (e.g. from ISS) are not able to provide this information with the necessary precision

2.3.3.1 Trigger and Read-Out

As the electronics has to handle over 315 000 pixels, the FS has been partitioned into subsections - the PDMs - and a multi-level trigger scheme has been developed (see Figure 2.6 for an overview). Basically, there are two trigger levels: the first level trigger (L1) which is implemented within the PDM electronics and the second level trigger (L2) which is implemented in the Cluster Control Board (CCB). The L1 trigger has three sub-levels.

Table 2.3 gives an estimate of the expected trigger rate for each level, and the expected rate of Cosmic Ray events (depending on the effective threshold of the detector, this value could fluctuate by around one order of magnitude).

The 1$^{\text{st}}$ sub-level is implemented within the Application Specific Integrated Circuit (ASIC) - which will be described in Section 2.3.3.2 - as a fast discriminator for each channel (anode) of the MAPMT. As the anodic pulses from photoelectrons can be

Table 2.3: Estimation on the trigger rates at each trigger level (JEM-EUSO Collaboration, 2010)

Level		trigger rate at PDM level	trigger rate at FS level
L1	Photon trigger	9.2×10^8 Hz	1.4×10^{11} Hz
	Counting trigger	7.1×10^5 Hz	1.1×10^8 Hz
	Persistency trigger	7 Hz	10^3 Hz
L2	Linear Track Trigger	6.7×10^{-4} Hz	0.1 Hz
Expected rate of CR events		6.7×10^{-6} Hz	10^{-3} Hz

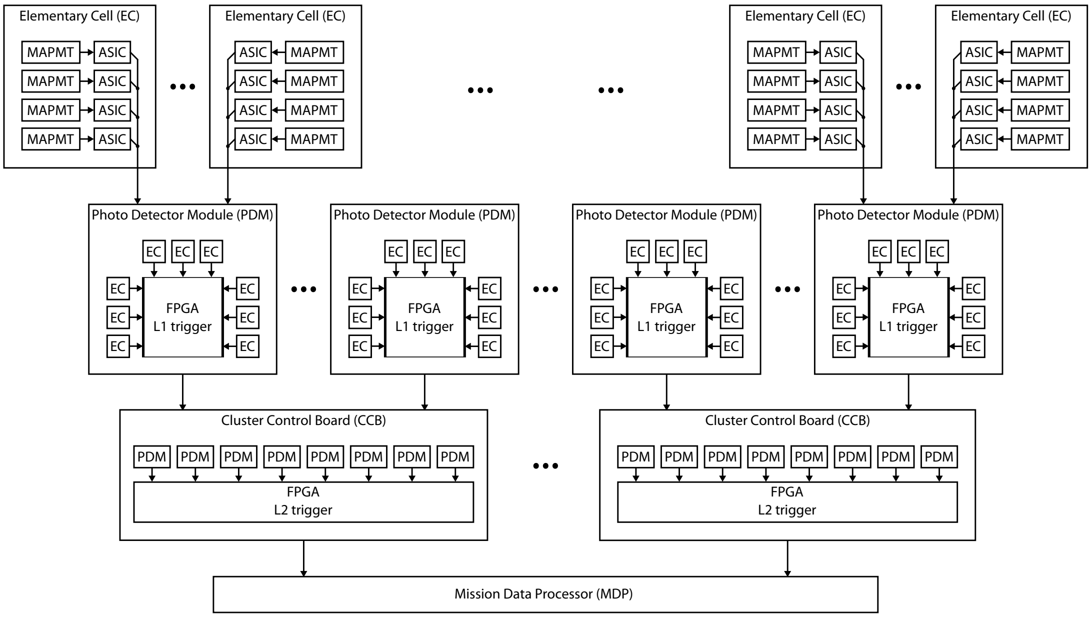

Elementary Cell (EC)
MAPMT
ASIC
Photo Detector Module (PDM)
EC
FPGA
L1 trigger
Cluster Control Board (CCB)
PDM
FPGA
L2 trigger
Mission Data Processor (MDP)

Figure 2.6: Overview of the trigger and read-out scheme: The L1 trigger is implemented at PDM level (at least the 2nd and 3rd sub-level) and the L2 trigger is implemented at PDM cluster level. For more details see Section 2.3.3.1.

discriminated above the electronic noise from the preamplifier, this noise can be easily reduced at this stage. The fast discriminator allows to distinguish pulses with a time separation of 10 ns to 15 ns. If there is a signal over the discriminator threshold, a new fast digital pulse is generated which will be used by the digital part of the ASIC to increment the photon counters.

The 2nd sub-level trigger is implemented inside a Field Programmable Gate Array (FPGA) at PDM level. This FPGA reads out the counters from a cluster of 36 ASICs (or 9 ECs) and constantly compares them with a predefined (digital) threshold. A ‘pixel trigger’ is issued for the 3rd sub-level in case this threshold is reached. The counters will be reset after a predefined time period, the Gate Time Unit (GTU), which is currently defined as 2.5 µs.

The 3rd sub-level trigger is implemented inside the same FPGA as the 2nd sub-level trigger and ‘looks’ for continuous (over some GTUs) activity within one PDM. This is referred to as Progressive Track Trigger (PTT). If a 2nd sub-level trigger occurs, this PDM is ‘marked’ as active and in the case that this activity continues over several successive GTUs a trigger for the L2 trigger is generated.

The 2nd level trigger is the final stage for the decision. Then read-out procedure starts and it will be described in detail in Chapter 3.

2.3.3.2 Front-End Electronics

Readout ASIC: The pulses generated by the MAPMT will be read out and converted to digital numbers within the front-end ASIC, which then can be processed by the following trigger stages in a digital way - the purpose of the ASIC is therefore the analog-to-digital conversion of the incoming photons (together with the MAPMT).

As the ASIC has to handle the pulses from a wide range of light sources with large differences in their amplitude and time structure (e.g. from neutrino induced showers which are faint and last a few nanoseconds up to TLEs which are very bright and last for several hundred milliseconds) it must be sensitive to single photons as well as to bunches of indistinguishable photons. Therefore, the ASIC has a 'photon counting' mode (for low photon rates) and a 'charge integration' mode (for overlapping photons respectively bright sources).

The basic scheme of the ASIC, whose name is Spatial Photomultiplier Array Counting and Integrating Read Out Chip (SPACIROC), is given in Figure 2.7. The signal from the 64 anodes of the MAPMT is pre-amplified and fed into three different paths (no shaper at all, a fast shaper and a very fast shaper) and then discriminated (1$^{\text{st}}$ sub-level trigger) to generate a fast pulse for the counters (the three possibilities for the shaper are currently under test to optimize the power consumption). Furthermore, the summed signal from a group of 8 channels is connected to the charge integration module (KI), which performs a charge-to-time conversion. In addition to these main modules, the last dynode of the MAPMT is also fed into another KI module whose information is used to protect the MAPMT in case of too much light (switch off or reduce the gain).

EC board: The purpose of the EC board is to group 4 MAPMTs, connect the MAPMTs to the ASICs and provide connectors for the power supply and the ASIC output signals. As there is not much space allocated for this board, it is not possible to have multiple boards for these purposes.

The current solution is to have one thick board (5 mm), to be able to plug the four MAPMTs directly into the board and to mount the four ASICs and the connectors on the other side of this board (see Figure 2.8). As the pins of the MAPMTs are not allowed to go through the whole board, it requires a new technology (called 'Holtite') but it is not clarified yet if this can be used in space and if a manufacturer will be found.

PDM Electronics: The PDM electronics will collect the digital data from 36 ASICs (9 ECs respectively), perform the L1 trigger and send the data to the CCB in case of a trigger. In addition, the PDM electronics is also responsible for setting the configuration parameters of the ASICs which will be received from the Mission Data Processor (MDP).

Due to the large number of input/output pins needed for the communication with the ASICs and the CCB it is foreseen to separate the PDM electronics on three boards - two providing the connectors to the ASICs and one containing an FPGA for the processing

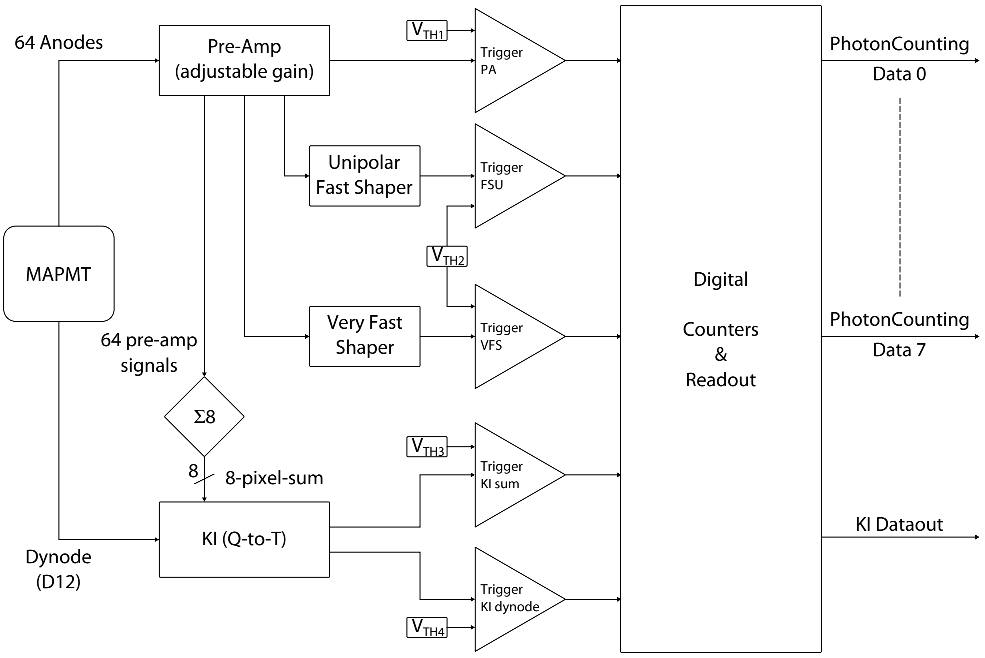


Figure 2.7: Block diagram of the SPACIROC: The 64 channels coming from the MAPMT are pre-amplified and a fast digital signal is generated which will increment the counters for each channel ('photon counting' part). Additionally the charge coming from 8 channels is integrated ('KI') in the case of a bright source (e.g. lightning) when single photon counting is not possible (Image based on: Ahmad et al., 2010).

and its associated electronics. These boards, together with the high and low voltage power supplies will be mounted within the PDM mechanical structure.

The main building blocks of the PDM electronics are (JEM-EUSO Collaboration, 2010):

- System Control (SC):
  Control of PDM system for operation
- ASIC Bus Interface Unit (ABIU):
  Interface between ASIC and PDM by customized communication
- Cluster control board Bus Interface Unit (CBIU):
  Interface between PDM and CCB by customized or commercial communication
- Housekeeping Interface Unit (HIU):
  Interface between PDM and HK
- CPU Interface Unit (CIU):
  Interface between PDM and MDP

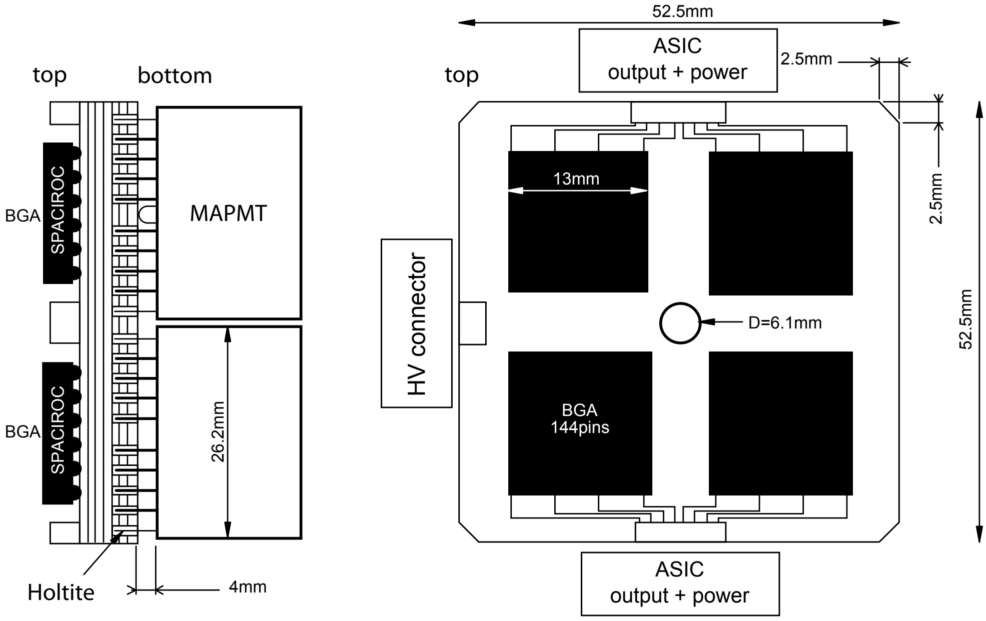


Figure 2.8: Scheme of the current EC board: The MAPMTs are directly plugged into the board but the pins do not go through the whole board. With this technology (‘Holtite’) it is possible to mount the ASICs and the connectors on the other side of the same board (Image based on: Barrillon et al., 2010).

- Trigger Unit (TU):
  Decide best trigger for observation simultaneous trigger signals
- Data Processing Unit (DPU):
  Process the data of the PDM
- Calibration Run Unit (CRU):
  Control the calibration run
- Run Summary (RS):
  Save all useful information for data

CCB: The CCB is the last stage of the FS read-out structure and mainly performs further management and reduction of the data. The CCB receives data from a cluster of PDMs, performs the necessary analysis to generate a trigger in case of a potentially good event and sends the data to the MDP. For a detailed description of the CCB see Chapter 4.

MDP: The MDP (see Figure 2.9) controls the whole instrument including the communication with the ISS and the temporary storage of the scientific data. The system consists of two identical Storage and Control Units (SCUs) to have a back-up unit in case of a failure. Each SCU includes several boards connected together through a main board (JEM-EUSO Collaboration, 2010):

- Power Supply Module:
  generates the necessary supply voltages

- Supervisor Module:
  contains the Central Processing Unit (CPU) including Electrically Erasable Programmable Read-Only Memory (EEPROM) for software storage and Static Random Access Memory (SRAM) for software execution

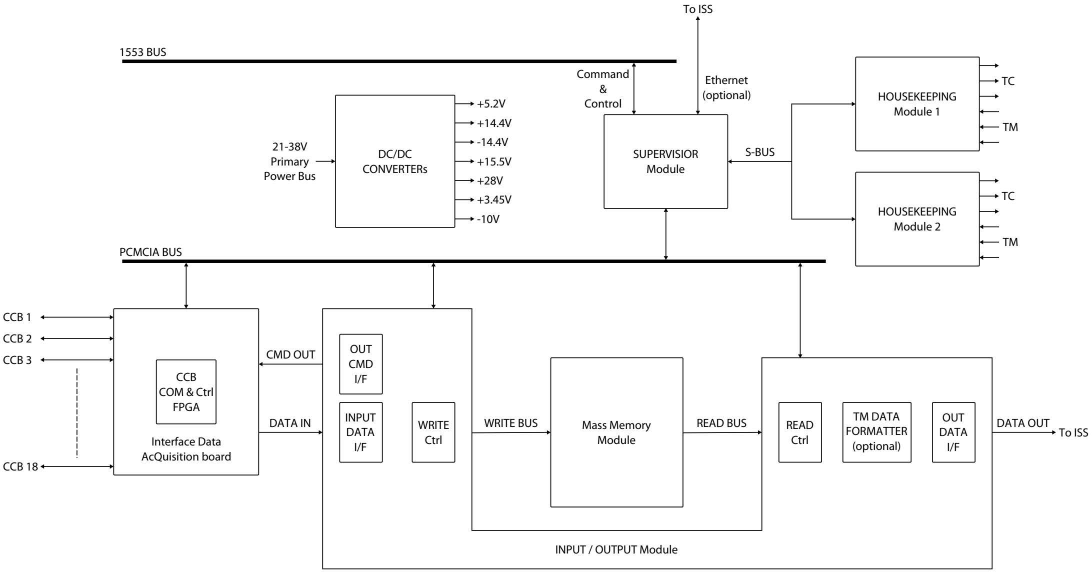
- Housekeeping Module:
  monitors and gathers engineering data of the system

- Memory Module:
  a mass memory device dedicated to the temporary storage of data before they are send to the ISS (to ground respectively) which allows to buffer data in the case that the down link is not available

- Input/Output Module:
  basically manages the input and output to/from the Memory module

- Interface Data AcQuisition (IDAQ) Module:
  handles the communication and data transfer between the detector (e.g. the CCBs, IR camera, LIDAR...) and the Memory/Supervisor module

Clock and Time Synchronization Board This board is basically responsible for generating and distributing the system clocks, to provide a time synchronization of the events and to measure the live-/dead-time of the detector.



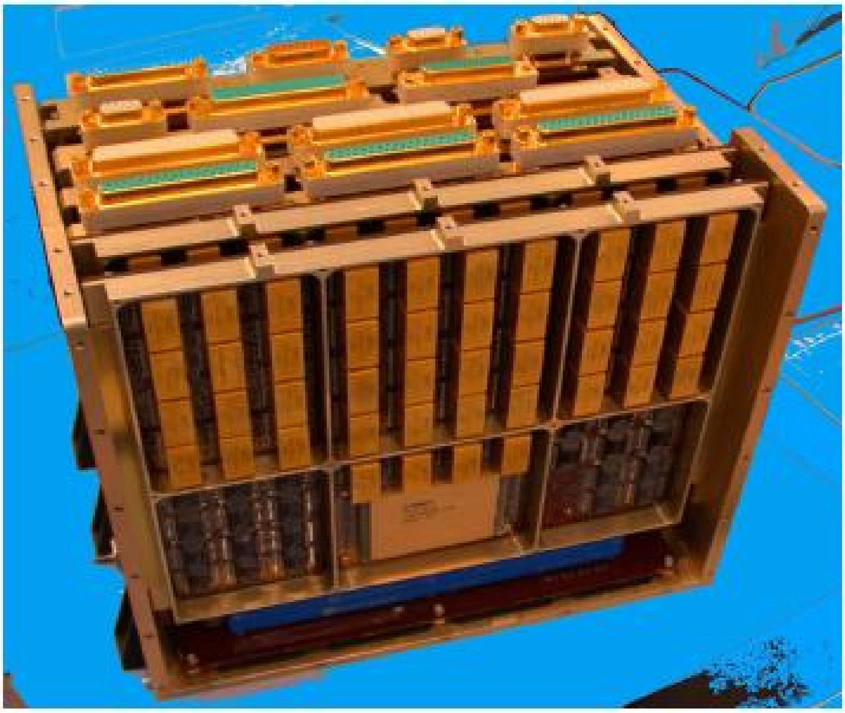


Figure 2.9: Block diagram (top) of the SCU with the main modules (see also Paragraph MDP in 2.3.3.2) and a picture (bottom) with removed panels, showing the Memory module in front (Image: courtesy of Thales Alenia Space).

3 L2 Trigger Algorithm

3.1 Overview

The fluorescence light from EASs (induced by UHECRs) looks like a thin luminous disk, which travels approximately with the speed of light on a straight path through the atmosphere. As the EAS produces more particles, the luminosity of the disc will also increase until the shower reaches its maximum and then fades out. A typical proton induced shower with an energy of 10^{20} eV, will be detected as several photons per pixel and µs during a typical duration of tens to hundreds µs. As the detector has some exposure time for each image, the fluorescence light will appear as a small spot, which moves on a straight line from image to image. The speed and the direction of this moving spot is obviously depending on the incoming direction of the UHECR that is from the shower axis.

The principle of the L2 trigger algorithm is therefore trying to follow the movement of this spot over some predefined time, to distinguish this unique pattern from the background. In the case of an L1 trigger, the PDM electronics will send (together with the frame data) a starting point, which contains the pixel coordinates and the GTU which generated the trigger - also called ‘trigger seed’. The L2 algorithm will then define a small box around this trigger seed, move the box from GTU to GTU and integrate the photon counting values. This integrated value is then compared to a threshold above the background and an L2 trigger will be issued if the threshold is exceeded.

It should be stressed out, that the L2 algorithm is constrained by the limited available computing power on board, due to power-, weight- and size-requirements and the space-qualification of the hardware (see also 3.3).

3.2 Mathematical Approach

In order to follow the movement of the spot on the detector, the speed and the direction in terms of detector pixels has to be calculated.

We define in the following θ as the zenith angle ($\theta = 0°$ means nadir direction of JEM-EUSO) and ϕ as the azimuth angle of the EAS ($\phi = 0°$ means direction of the ISS movement). Since the EAS travels at the speed of light, the photons reaching JEM-EUSO

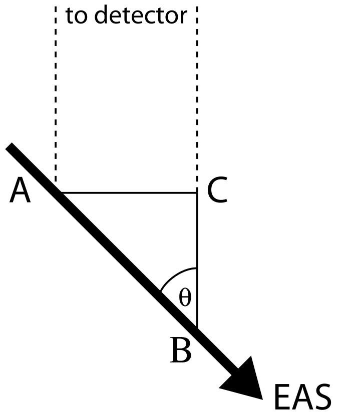


Figure 3.1: Kinematics of an EAS.

from any point on the EAS are lagging behind the passing EAS-front by the time:

$$\Delta t = \frac{\overline{AB} + \overline{BC}}{c} \tag{3.1}$$

with c being the speed of light and $\overline{AB} + \overline{BC}$ as defined in Figure 3.1. Together with

$$\sin\theta = \frac{\overline{AC}}{\overline{AB}}, \quad \tan\theta = \frac{\overline{AC}}{\overline{BC}}, \quad \tan\theta = \frac{\sin\theta}{\cos\theta} \quad \text{and} \quad \tan\frac{\theta}{2} = \frac{\sin\theta}{1+\cos\theta}$$

it follows

$$\Delta t = \frac{\overline{AC}}{c \cdot \tan\frac{\theta}{2}} \tag{3.2}$$

as can be derived from Figure 3.1. If we define now Δx and Δy as the projections on the focal surface along x and y expressed in number of pixels and ΔL as the FoV at ground of a pixel in a time Δt (chosen as the GTU) we find

$$\overline{AC} = \Delta L \cdot \sqrt{\Delta x^2 + \Delta y^2} \tag{3.3}$$

and therefore

$$\theta = 2\arctan\left(\frac{\Delta L}{c \cdot \Delta t} \cdot \sqrt{\Delta x^2 + \Delta y^2}\right) \tag{3.4}$$

on the other hand we can find

$$\phi = \arctan\left(\frac{\Delta y}{\Delta x}\right) \tag{3.5}$$

3.3 Number of Directions

As the incoming direction of the EAS is unknown, the approach of the L2 trigger algorithm is simply to try out a set of directions which covers the complete space ($\theta = 0° \dots 90°$ and $\phi = 0° \dots 360°$). It should be clear, that the integrated value will have a maximum when we 'hit' the (nearly) correct direction, because in this case the integrating box will follow the spot.

As the integration over the directions is a time consuming task, a trade-off between the number of directions and the available computing power has to be found. In addition, as the hardware only works on whole pixels, some directions will produce the same offsets Δx and Δy - for example, if $\theta = 0°$ (which means the shower axis is parallel to the optical axis of the telescope), then the spot is not moving at all and for this reason, there is no need to integrate over different ϕ angles.

So one of the first tasks was to define a minimum set of directions, which contains enough θ-ϕ combinations to recognize all events without wasting computing power.

3.3.1 Minimum Set of Directions

The procedure for generating a minimum set of directions is based on the requirements that

- it should cover the complete range of θ and ϕ,
- it should not contain directions producing the same offsets,
- and the 'overlap' of the integrating boxes (for nearby angles) should be minimized.

Therefore, a simple iterative scheme was developed, with the following steps to build up the list (see also Figure 3.2):

1. Calculate the offsets for a starting direction (e.g. $\theta = 90°$ and $\phi = 0°$) and for the number of GTUs which will be used for the integration. Currently it is foreseen to integrate over ± 7 GTUs, that is 15 GTUs in total centered around the trigger seed. Use these offsets as an imaginary track (the red boxes in Figure 3.2a) and generate the integration boxes (currently 3x3 pixels) around these offsets (the black boxes in Figure 3.2a).

2. Increase the angle ϕ of the track until some part of the track 'runs out' of the integration boxes (Figure 3.2b) - which means that we probably lose some counts of the track while integrating over that direction.

3. Now increase ϕ of the integration direction as long as the complete track is still within the new integration boxes (the green boxes in Figure 3.2c) - this θ-ϕ combination is then the next necessary direction, where we don't lose any signal of the track (at least in this idealized assumption) and where the 'overlap' of the boxes is minimized.
4. Start over again with step 2 (Figure 3.2d), until one 360° turn in ϕ is made.
5. Now it is time for the next θ angle which is nearly the same process as for ϕ. Decrease θ of the track (so the track will 'shrink') until the first (GTU# -7) or the last (GTU# +7) box no longer contains a part of the track and then decrease θ of the integration direction until the track is still within the boxes - this is the next necessary θ.
6. Start over again with step 2 until $\theta = 0°$ is reached, which means the direction list covers now the complete range ($\theta = 0° \ldots 90°$ and $\phi = 0° \ldots 360°$).

3.3.2 Evaluation

The list built up this way, contains now 67 different directions (see Table A.1) and it has to be evaluated in order to see if the number of directions is enough or rather to see if the trigger efficiency is declined. For this purpose over 4000 events - which were generated by the simulation group with the EUSO Simulation and Analysis Framework (ESAF) - were analyzed, and the results from the new generated list were compared with the one implemented in ESAF. The ESAF list contains nearly 5x more directions which is a big difference (a total of 324 directions for $\theta = 5° \ldots 85°$ and $\phi = 5° \ldots 355°$ in 5°-steps). See also Fenu (2008) for more details on ESAF and simulations.

The first test was simply a comparison of the integrated values for a few randomly chosen events. The results of two events are given in Figure 3.3 - a low energy ('Event 0') and a high energy event ('Event 3') where Figure 3.3a is generated with the long list from the simulation and Figure 3.3b with the reduced list. The 'Direction Index' on the x-axis is just the number of the direction in the list, which is arranged in the following way: $(\theta_0|\phi_0), (\theta_0|\phi_1), \cdots, (\theta_0|\phi_n), (\theta_1|\phi_0), \cdots (\theta_n|\phi_n)$ with increasing angles for θ and ϕ.

By looking on Figure 3.3a, the first thing to be mentioned are the parts where the plot is horizontal over several directions. This means that exactly the same integrated value for different directions is produced. It happens for example at the beginning of the curve (from direction 0 to approximately direction 40) which is exactly the case of a very inclined shower (θ close to 0°), so the spot is not (or slow) moving and there is no reason to integrate over different ϕ angles. As this does not happen for the short list, no time is 'wasted' on something which does not generate new information - which was one of the goals for the list.

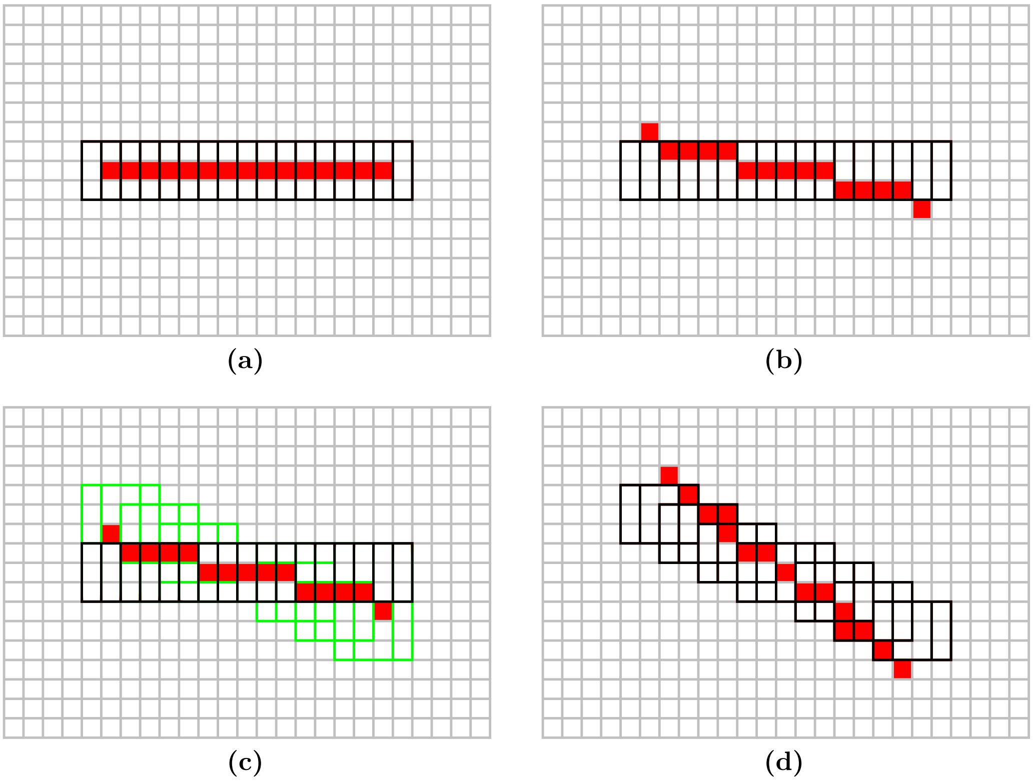
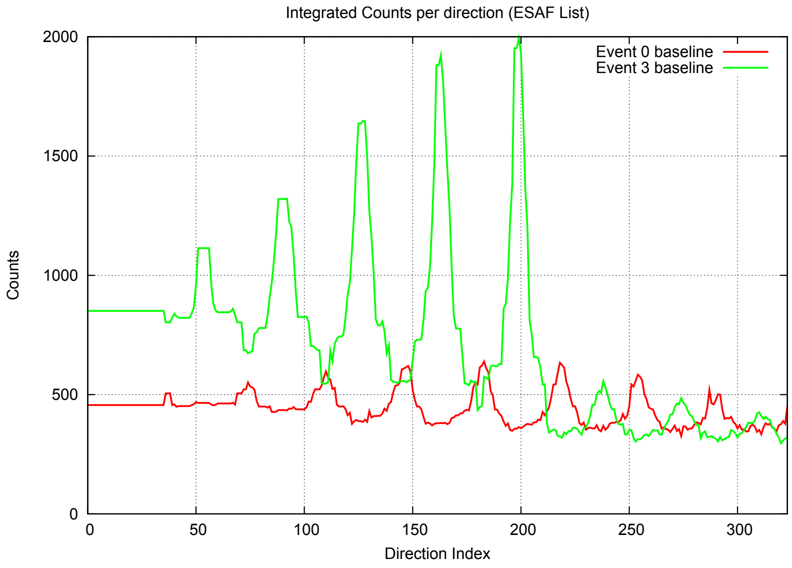


Figure 3.2: The main steps for the generation of the direction list (see also Subsection 3.3.1): (a) start with one direction; (b) increase ϕ of the track until it 'runs out' of the integration boxes; (c) increase ϕ of the integration boxes as long as the track is still within the boxes; (d) start over again for the next ϕ.

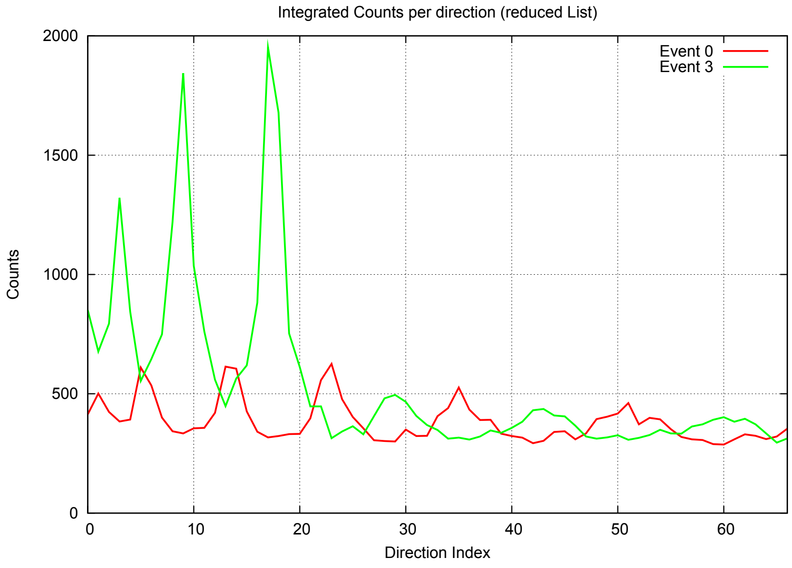
The second thing one can notice by looking at the plots are the peaks and the increasing of the peaks up to a maximum value. Whenever the current integration direction has the correct ϕ angle of the shower, the integrated value will have a (local) maximum since the integrating box follows the spot at least over some pixels. In addition, as the θ of the integration direction approaches the correct θ angle of the shower, the distance in which the box follows the spot gets larger, therefore, producing a higher integrated value until the maximum is reached when the correct θ-ϕ combination is found.

As a conclusion, it can be clearly seen, that the short list is also generating these peaks and that the maximum value is nearly the same as for the baseline list. This is a quite satisfying result, as this means that the number of directions should be enough and the trigger efficiency will not decrease.

To have a more detailed comparison of the two lists, the difference of the maximum



(a)



(b)

Figure 3.3: Comparison of the integrated values for two different sets of directions: the 'ESAF'-set with 324 and the optimized set with 67 directions (see Subsection 3.3.2). Shown is the number of counts over the direction, which are numbered serially (notice the difference in the x-axis), for a weak ('Event 0': $E = 1.39 \times 10^{20}$ eV) and a strong ('Event 3': $E = 3.92 \times 10^{20}$ eV) event. The most important result of this comparison is that the optimized list produces nearly the same maximum integrated value as the long 'ESAF'-list.

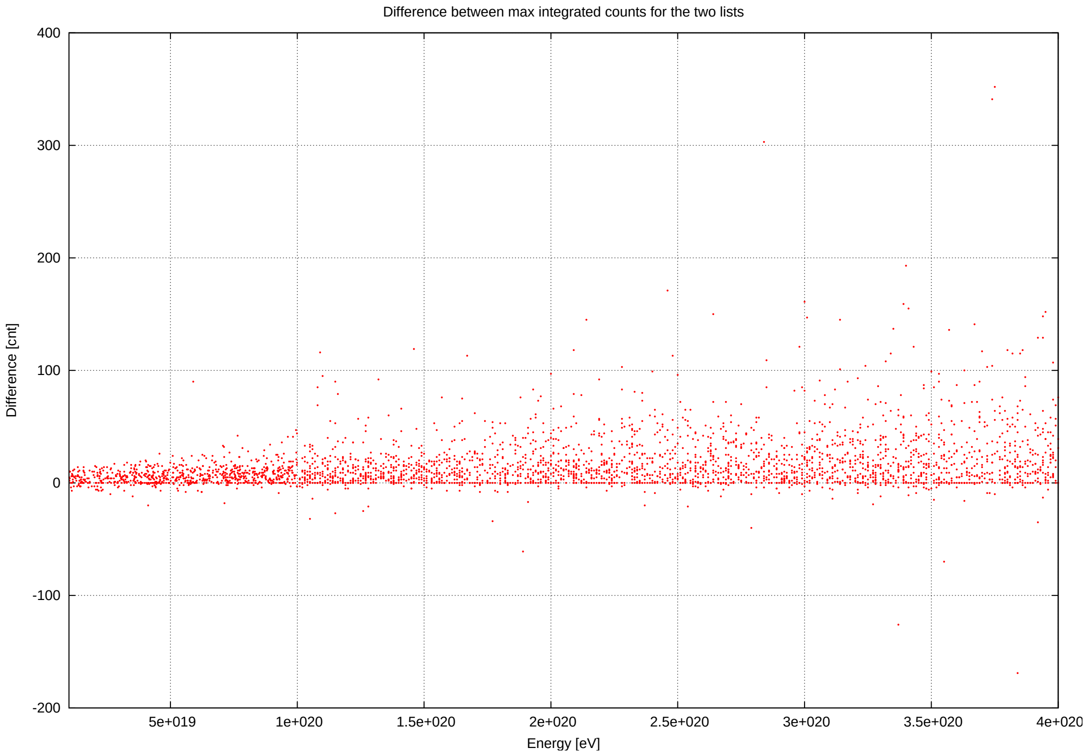


Figure 3.4: For a more detailed comparison between the 'baseline'- and the optimized-list, a set of 4000 events was analyzed. Shown is the difference between the maximum values after the integration of the two lists, over the energy of the event - a positive value means, that the baseline-list produced a higher value. As can be seen, the difference of nearly all events is within ±200 counts, which is less than the background per pixel and GTU.

integrated value for all the 4000 events was plotted: a positive value means that the maximum value of the baseline list is greater. As can be seen from Figure 3.4, the difference of nearly all events (except for three) is less than 200 counts - which is less than the background per pixel and GTU. To understand, why these three events produced such a significant difference, a deeper look was necessary. We understood, that these events were very close and rather parallel to the edge of the PDM. Due to the fact, that the directions were completely skipped, if the integration box ran out of the frame (as there is no information for the integration) and as the baseline list contains all directions which are parallel to the edge of the frame, it was clear, that in these cases the short list has a disadvantage.

Based on this knowledge, one possibility to solve the problem is to simply add all the parallel directions to the short list. Another solution could be, to change the implementation for the integration and no longer skip the direction if the box runs out of the frame but simply add zero or some value for the background - which is now the case for the hardware implementation.

3.3.3 Conclusion

Based on our study, it clearly emerged that there is no need for the large amount of directions, currently implemented in ESAF, for the L2 trigger. The baseline list can be safely reduced by around 80 directions, which produce exactly the same offsets. On the other hand a deeper study for the short list is necessary and further optimization is possible.

The study is now in progress and the reduced list will be implemented in ESAF to evaluate the trigger efficiency. Unfortunately there were no news before this work was finished.

4 Cluster Control Board

4.1 Overview

As already mentioned in Section 2.3, the detector of JEM-EUSO has a modular concept and the focal surface is partitioned over several levels with increasing area, so the electronics is able to handle the high data rate coming from over 315 000 channels.

To reduce the data rate from around 1 Tbit s^{-1} generated[1] on the whole detector to around 900 kbit s^{-1} allocated for the downlink to Earth[2], a multi-level trigger scheme was developed (see also Subsection 2.3.3.1) and the last trigger level (L2) will be implemented inside the CCB electronics. If an L1 Trigger is issued, the data of the ring buffers from 8 PDMs are transferred to the CCB and the L2 trigger algorithm is executed. The data from the PDMs are processed independently, as it is not intended to have a fixed geometrical relationship between the PDMs connected to one CCB (like the 2x2 configuration for the EC or the 3x3 configuration for the PDM). In case an L2 trigger is found from any of the 8 PDMs, the complete data are transferred via the IDAQ module to the mass memory module of the MDP (see also Subsection 2.3.3.2).

Due to the necessity for parallel processing and the need for a high number of input/output pins (for the eight fast data interfaces to the PDMs and the interface to the MDP, besides various other interfaces for control and housekeeping) it is foreseen to implement the L2 trigger algorithm inside an FPGA chip on the CCB. As a baseline, it is planed to use a radiation tolerant Xilinx Virtex-4QV FX-140 and as an advanced option the use of a radiation hard Virtex-5QV is currently under investigation.

The development of the internal FPGA hardware is done with the Very high speed integrated circuit Hardware Description Language (VHDL) together with tools from Xilinx (ISE Design Suite 12) on a prototyping board from IAF[3] with a Virtex-4 FX100 (see Figure 4.1). The FX-140 FPGA has a few more logic resources than the FX-100, which will be needed later for the space qualified implementation of the hardware.

[1] This value is only the photon counting data and does not include additional data from the charge integration module (KI) of the ASIC:
137 PDMs · 9 ECs · 4 MAPMTs · 64 Channels · 8 bit · $2.5\,\mu\mathrm{s}^{-1} \approx 1 \times 10^{12}$ bit s^{-1}.

[2] 900 kbit s^{-1} is only allowable with a temporary on board storage of 60 Gbit, so the data can be transmitted while observation is not possible (e.g. during full moon). If on board storage is not available, the allocated down link data rate is only 300 kbit s^{-1} (Tsuno, 2010).

[3] Institut für angewandte Funksystemtechnik GmbH

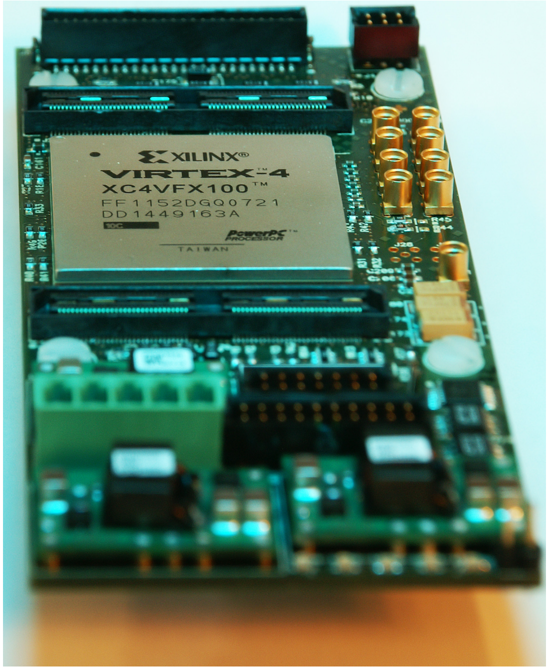


Figure 4.1: The FPGA prototyping board with a Xilinx Virtex-4 FX100 chip, which is used for the hardware development.

The current block diagram for the FPGA is given in Figure 4.2, which is basically the implementation of the L2 trigger algorithm. Besides the eight Linear Track Trigger (LTT) modules it is planned to use the microcontroller cores[4] for 'low speed' purposes, such as the configuration of the LTT modules, housekeeping and the communication with the MDP.

In order to perform the necessary calculations as fast as possible, the hardware architecture of the trigger is highly pipelined and parallelized. After receiving an L1 trigger, the data transfer from PDM to CCB starts over a fast serial data link (see Subsection 4.2). As it is not possible to store the raw data completely in the internal RAM of the FPGA, the data will be written to an external memory. Only the data necessary for the trigger calculation will be stored inside the FPGA to reduce the probably slow read/write access for the external RAM - which will be done by the 'Data Allocator'-module. This data will then be passed to the '3x3 Sum'-module which performs the summation of the 9-pixel blocks in two stages (horizontal and vertical) for the whole frame which reduces redundant 3x3 summation to a minimum. The new

[4] The Virtex-4 FX100/FX140 contains two PowerPC 405 cores, which is a 32-bit implementation of the PowerPC embedded environment architecture derived from the PowerPC architecture.

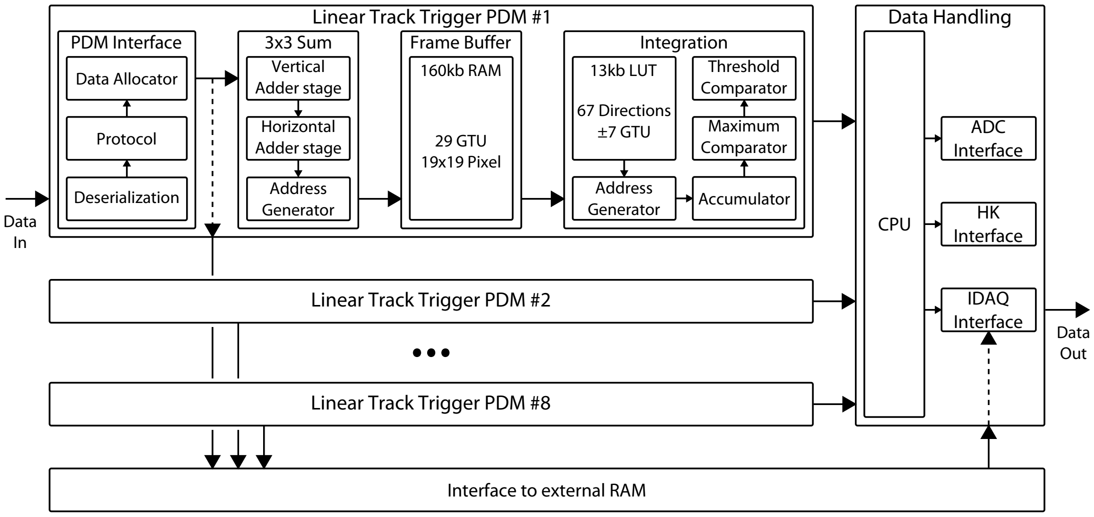


Figure 4.2: The current block diagram of the CCB FPGA. Most of the logic resources are needed for the eight 'Linear Track Trigger' modules which perform the L2 trigger algorithm and handle the fast data transfer from the PDMs. As it is not possible to hold the data from all eight PDMs inside the FPGA RAM an interface to an external memory will be needed. Additionally, it is currently foreseen to use the integrated microcontrollers of the Virtex-4 FX140 to control and monitor the trigger modules and to handle the communication with the IDAQ and the housekeeping submodule.

frame generated this way is then trimmed to a 19x19 pixel frame around the seed and stored in the 'Frame Buffer' for ±14 GTUs.

After the 'Frame Buffer' is filled completely (contains the necessary information for the integration), the 'Address Generator' allocates the different 3x3 sums - depending on the starting point[5] for the integration and the offsets for the various directions which are stored in a Look Up Table (LUT) - to the 'Accumulator'. After each direction-integration, the accumulated value is compared to the previous one (to the current maximum respectively) to select the overall maximum value and is then stored along with the information to which starting-point this value belongs ('Maximum Comparator'). When the integration process is finished for all starting points, the maximum value is compared to the trigger threshold ('Threshold Comparator') and an L2 trigger is issued if the threshold is exceeded. In this case the raw data buffer (external RAM) is sent to the IDAQ, whereas the information at which pixel the maximum value was found, could be used by the MDP to calculate the pixel coordinates on the focal surface, which then could be used as a target for the LIDAR.

[5] A total of 375 (TBC) starting points is foreseen in the current implementation. The starting points are distributed around the trigger seed - 5x5 pixels and for ±7 GTUs.

According to the current implementation, the decision of an L2 trigger could be performed within approximately 5 ms. The time needed for the data transfer between PDM, FPGA and external RAM is not included in this number as the PDM interface is still not defined. Additionally, it is difficult to find a space qualified external RAM which is fast (and large) enough to receive the data with the same speed as the data are transferred to the CCB. It should be stressed, that the actual calculation time depends on the system clock of the FPGA, therefore, on the overall implementation. However, even if the system clock has to be reduced by a factor of 5 the calculation is still a factor of 4 faster than required, as 100 ms were allocated for the L2 trigger calculation.

The following sections will describe the different modules of the internal FPGA hardware in more detail up to the current development stage.

4.2 PDM Interface

The interface between PDM and CCB for the scientific data from the detector is a critical part in the processing chain as many other parts rely on the performance of this interface (e.g. the ring buffer on the PDM, the implementation of the L2 algorithm and the dead-time of the instrument). As the interface was not defined at the beginning of this work (and still is not!), it was necessary to find a possible solution. The developed interface is primarily a feasibility study on the hardware rather than a 'ready to use' interface (e.g. no protocol is yet defined), nevertheless the overall implementation of the L2 algorithm is based on this proposal as this is currently the only one.

The following sections will describe the receiver and the transmitter, which are based on Burton (2006) and the test setup to qualify the interface, which is based on Xilinx (2005).

4.2.1 Requirements

The transfer rate is given by the amount of data to be transferred and the available time to transfer this data. As both of these values are not yet defined (e.g. the size of the ring buffer) we only can define a lower limit for the transfer rate.

The current baseline for the transfer rate is constrained by the fact, that the dead-time of the detector should be as small as possible. Therefore, if no additional dead-time should be introduced by the data transfer to the CCB, one frame has to be transferred at least in the time needed to record one frame. This will enable the possibility to keep filling the ring buffer while the 'old' data will be sent to the CCB - so one frame per GTU should be transferred, which leads to a

$$\begin{aligned}\text{data rate} &= 8\,\text{bit} \cdot 64\,\text{channels} \cdot 36\,\text{MAPMT} \cdot 2.5\,\mu\text{s}^{-1} \\ &\approx 7.4\,\text{Gbit}\,\text{s}^{-1}.\end{aligned}$$

As the PDM FPGA is running at 40 MHz (which means that around 200 lines will be needed to achieve the required data rate), a compromise between the number of data channels and higher clocking has to be found. Due to the fact that the distance between the PDM and the CCB is quite large (≥1 m) - as it is foreseen that the CCB is mounted on the back of the focal surface - the interface should not use single ended connections but differential lines which results in a more reliable interface, but doubles the number of lines needed. Furthermore, as the CCB has to handle 8 PDMs, the total number of input/output pins would be too large for the FPGA and must be reduced by serializing the data. Another constraint on the interface was to make use of the hardware primitives of the FPGA, as most of the logic resources will be needed for the trigger implementation.

4.2.2 Hardware Primitives

The input/output pins of the Virtex-4 are highly customizable and supporting a wide variety of different signaling standards[6], which Xilinx calls SelectIO™. Each SelectIO™ block contains another block of primitives (named ChipSync™, see Figure 4.3a), which provides the necessary hardware for serial interfaces and is therefore well suited for a fast data interface between PDM and CCB. A small summarization of the main primitives which are used for the developed interface is given in the following - for more details see Xilinx (2006) and Xilinx (2008).

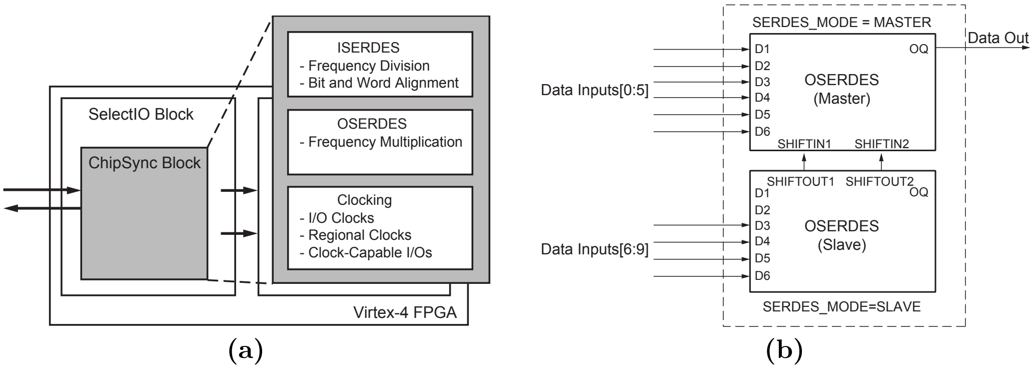


Figure 4.3: The ChipSync™ block (a) of the Virtex-4 provides all necessary hardware for a high-speed serial interface. The SERDES primitives are the main modules for the de-/serialization and two of them can be cascaded (b) to achieve a serialization of up to 10:1 (Image: Xilinx).

[6] This applies also to the Virtex-5 with some minor differences.

4.2.2.1 OBUFDS and IBUFDS

These are the input and output drivers for differential signals and configured for Low Voltage Differential Signaling (LVDS) with internal termination and 2.5 V supply voltage. Other voltages are not supported by the radiation tolerant Virtex 4 (see Xilinx, 2010). As the LVDS drivers are included within the FPGA there is no need for external drivers which reduces costs and weight.

4.2.2.2 OSERDES and ISERDES

These are dedicated parallel-to-serial/serial-to-parallel converters and configurable for a de-/serialization ratio of up to 6:1 with either Single Data Rate (SDR) or Double Data Rate (DDR)[7]. As we use a differential standard and therefore need two SelectIO blocks for each data channel, it is possible to cascade the two SERDES modules to achieve a ratio of up to 10:1 (see Figure 4.3b). In the current interface they are configured for DDR and 8:1 serialization 1:8 deserialization, respectively.

4.2.2.3 IDELAY

The IDELAY primitive is included in every ISERDES module and is a programmable delay element with a fixed tap resolution of 75 ps and a maximum of 64 taps (4.8 ns). This delay element is used to synchronize the data channels with the clock and can be configured either for a fixed or adjustable delay time. In the current design, the receiver contains a Finite State Machine (FSM) which controls the IDELAY modules in such a way that the sampling clock is placed in the middle of the data eye.

In addition, it is necessary to instantiate the IDELAYCTRL module which needs a 200 MHz reference clock and controls the fixed tap delay over temperature and power supply variations by providing an absolute reference voltage to the tap delay line.

4.2.2.4 BUFR, DCM & PMCD

The regional clock buffer BUFR, the Digital Clock Manager (DCM) and the Phase Matched Clock Divider (PMCD) are providing various possibilities for clock division and multiplication and can be used to generate the fast serial and the slow parallel clock.

The clock input requirement of the SERDES modules is a phase-matched clock relationship between the slow and the fast clock which is guaranteed by design with the following three schemes (see also Figure 4.4):

[7] Double Data Rate means, that the data bits are transferred on the rising and the falling edge of the clock - in contrast to the Single Data Rate, where the data bits are transferred only at the rising edge.

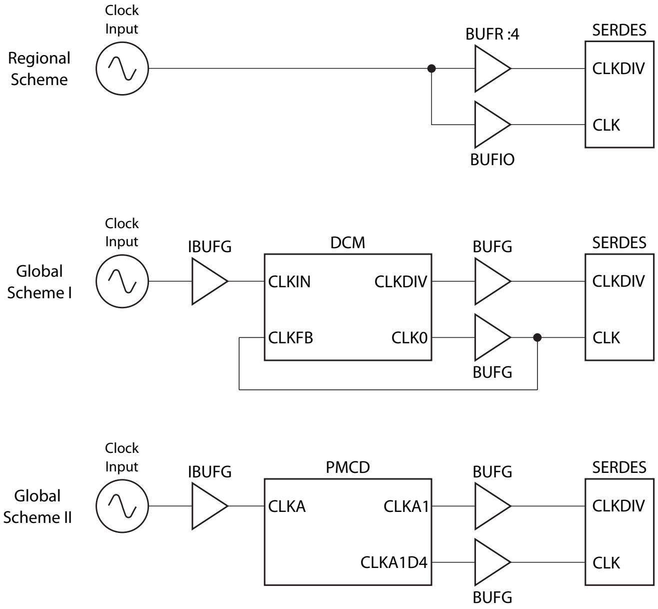


Figure 4.4: Different clocking schemes which guarantee the phase-matched clock relation by design, required from the SERDES modules. The 'Regional Scheme' uses the regional clock routing capabilities of the Virtex-4 and therefore allows higher clocking compared to the 'Global Schemes'. The clock division can be done with a Regional Clock Buffer BUFR, a Digital Clock Manager or a Phase Matched Clock Divider.

- Regional Clock Buffer:
  This scheme uses a BUFIO for the clock input and a BUFR which can be configured for a clock division between 1 and 8 in whole numbers to generate the slow parallel side clock.

- Digital Clock Manager:
  Another valid scheme is to use a global clock input buffer IBUFG and generate the two interface clocks with a DCM. Together with global clock buffers (BUFG), the CLKDIV output will be used for the slow and the CLK0 output for the fast clock.

- Phase Matched Clock Divider:
  The third scheme which guarantees the phase relation is to utilize a PMCD, using adequate outputs (e.g. CLKA1 and CLKA1D4) and route them again to BUFGs.

4.2.3 Transmitter

The transmitter module (see Figure 4.5) is uncomplicated and uses nearly no additional logic as all functionality is included in the primitives. As a starting point, a configuration

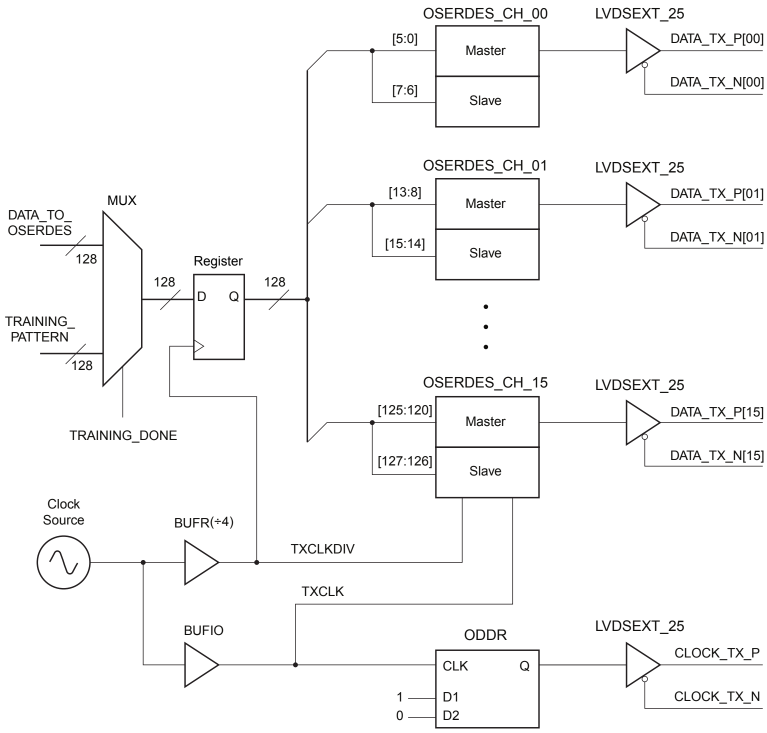


Figure 4.5: Block diagram of the transmitter for the fast data link between PDM and CCB. A configuration with 16 data channels and a serialization of 8:1 was chosen (Image based on: Burton, 2006).

of 16 LVDS channels with a serialization of 8:1 and DDR was chosen. This results in an interface where the parallel side is 128 bit wide, which means, that 16 pixels can be transferred per data frame and a total of 144 data frames are necessary for one PDM frame. If this configuration is driven with 40 MHz on the parallel side, a data rate of 320 Mbit s^{-1} per channel or 5.12 Gbit s^{-1} in total will be achieved, which is still not enough (around 60 MHz would be necessary) but a good approach to start with.

The parallel data frames are divided in 8 bit words and routed to the OSERDES modules (instantiated as ‘MASTER’ and ‘SLAVE’ to make use of the port expansion for higher serialization ratios). The serial output of the OSERDES pair is then directly connected to the LVDS output driver OBUFDS. There are two clocks needed by the OSERDES modules - the fast serial side clock (TXCLK) and the slow parallel side clock (TXCLKDIV). The clocks have to be phase-aligned with a relationship of 4:1 (as DDR is used) which can be achieved with different setups (e.g. BUFR, DCM & PMCD). In this case a regional clock buffer configured for a clock division by 4 was chosen, as a better performance was expected.

As this is a source synchronous interface the fast serial side transmission clock to the receiver must be provided by the transmitter. Clock forwarding is done (as recommended by Xilinx) by an Output Double Data rate Register (ODDR) where the two inputs reside on fixed values and the output again is routed to an LVDS output driver.

The only additional logic needed by the transmitter is a 128 bit input register and a 128 bit 2:1 multiplexer used to switch between data and a training pattern which is used by the receiver to perform the data-to-clock alignment for all the data channels with the IDELAY elements and make the word alignment by invoking the BITSLIP module which is also contained in the ISERDES modules (see Section 4.2.4).

4.2.4 Receiver

The input stage of the receiver follows the same principle as the output stage of the transmitter as can be seen in the block diagram 4.6. The data channels are routed into the FPGA fabric via the LVDS receivers IBUFDS and from there through the IDELAY primitive to the input of the ISERDES pair again in 'MASTER'/'SLAVE' configuration as in the transmitter. The slow parallel side clock is also generated by a regional clock buffer (BUFR), which is configured to divide the fast serial side clock by 4.

To account for different propagation delays between the single data channels and the transmission clock (e.g. due to cabling, connectors, routing on the Printed Circuit Board (PCB) or inside the FPGA), some additional logic was necessary to adjust these delays by introducing a tunable delay for the data channels with the IDELAY elements. During this alignment process, the transmitter continuously sends a predefined training pattern, which is used by the 'Bit Align Machine' to perform the following two steps:

1. Bit Alignment:
   The bit align procedure basically measures the data eye width by incrementing the delay of the channel and searching for the transition region of the data. After the first transition is found, a counter is incremented every time the delay is increased until the second transition is found. This counter contains now the data eye width with a resolution of 75 ps. In the last step, the final delay is set to half of this counter value, therefore, the sampling clock is placed in the middle of the data eye (see Figure 4.7).

2. Word Alignment:
   After the bit alignment is finished, it could happen that the total delay between data and clock is larger than one bit-period. In this case the 'first' clock transition will not sample the first bit of the data but the second. For this case each ISERDES module contains a BITSLIP sub-module which can reorder the sampled bits. For this purpose, the 'Bit Align Machine' will compare the data from the ISERDES module with the training pattern and invokes the bitslip operation until the correct pattern is found.

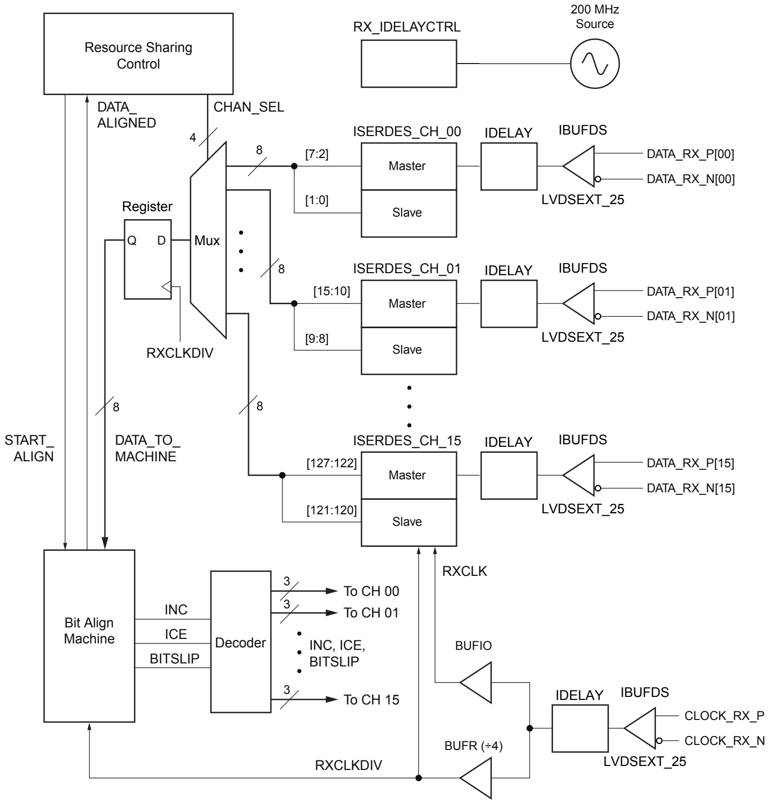


Figure 4.6: Block diagram of the 16 channels LVDS receiver. The input stage consists of the LVDS receivers, the delay elements to align the data channels with the clock and the ISERDES modules for the deserialization. The 'Bit Align Machine' controls the delay elements and performs the synchronization (Image based on: Burton, 2006).

To save logic resources, there is only one 'Bit Align Machine' for all channels, which is supported by the 'Resource Sharing Control' and the 'Decoder' module. The 'Resource Sharing Control' simply supplies the 'Bit Align Machine' with the correct data from the different channels in a round robin style. On the other hand, the 'Decoder' allocates the control signals from the 'Bit Align Machine' to the ISERDES and IDELAY primitives.

It is important to point out, that this alignment process is only needed, if the delays between the data channels and the clock are unknown. In the final design, the cable and trace lengths on the PCB are fixed (and matched), and the IDELAY elements can be set to a fixed value or omitted at all. On the other hand, the alignment procedure could also be used in the final design to correct variations in the timing due to changes over supply voltage or temperature.

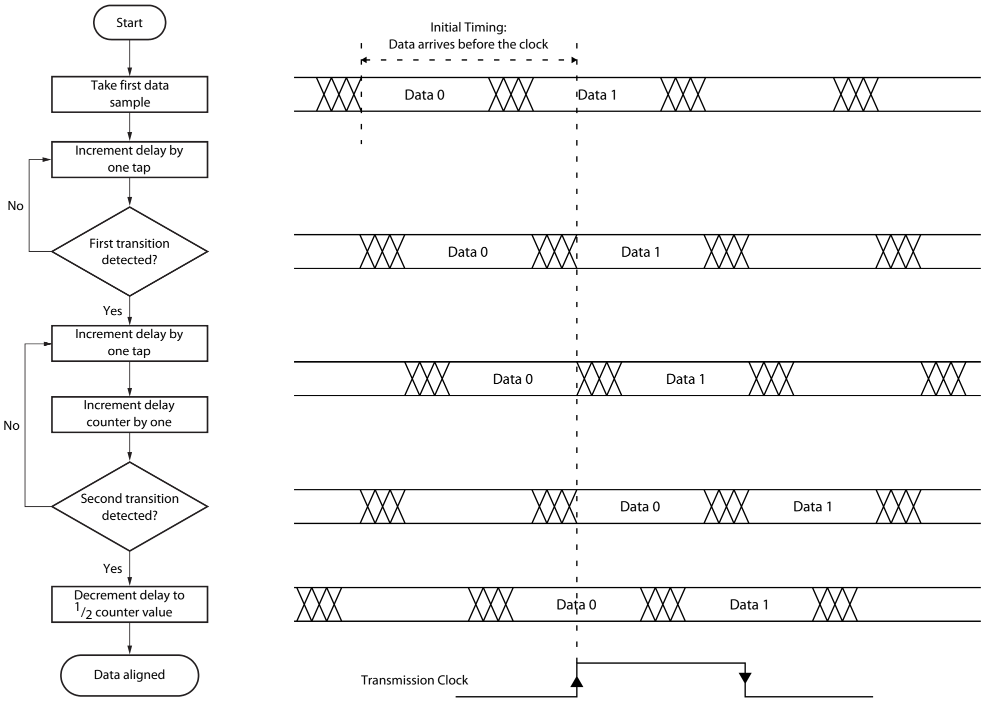

Start
Take first data sample
Increment delay by one tap
No
First transition detected?
Yes
Increment delay by one tap
Increment delay counter by one
No
Second transition detected?
Yes
Decrement delay to $^1/_2$ counter value
Data aligned
Initial Timing:
Data arrives before the clock
Data 0
Data 1
Transmission Clock

Figure 4.7: The principle and the basic flow chart of the 'Bit Align Machine', which controls the IDELAY elements of the receiver in such a way that the data eye of all channels is centered around the edges of the transmission clock.

4.3 3x3 Sum

Due to the fact, that it was neither feasible to perform the complete integration in 'real time' (completely parallel) because of the limited resources of the FPGA, nor to perform it completely serially due to the limited time, it was decided to split the integration process in two parts (a parallel one for the 3x3 summation and a serial one for the integration over time). Additionally, this decision is also based on the fact, that if the integration will be made completely serially, a large amount of 3x3 summations will be redundant. For example, as the integration over time is done from GTU-7 to GTU+7, the 3x3 box at GTU-0 will be the same for all directions (at least for one start point). By splitting the integration into these two parts, a factor of 9 in the processing time is gained (compared to the completely serial version), as only one 'pixel' has to be read from the 'Frame Buffer' instead of nine (one after the other) for the 3x3 box integration.

This module receives the data stream from the PDM interface in 128 bit-blocks (16 pixels) per clock cycle, performs the first part of the integration and stores the data in the 'Frame Buffer', which will be used for the second part of the integration.

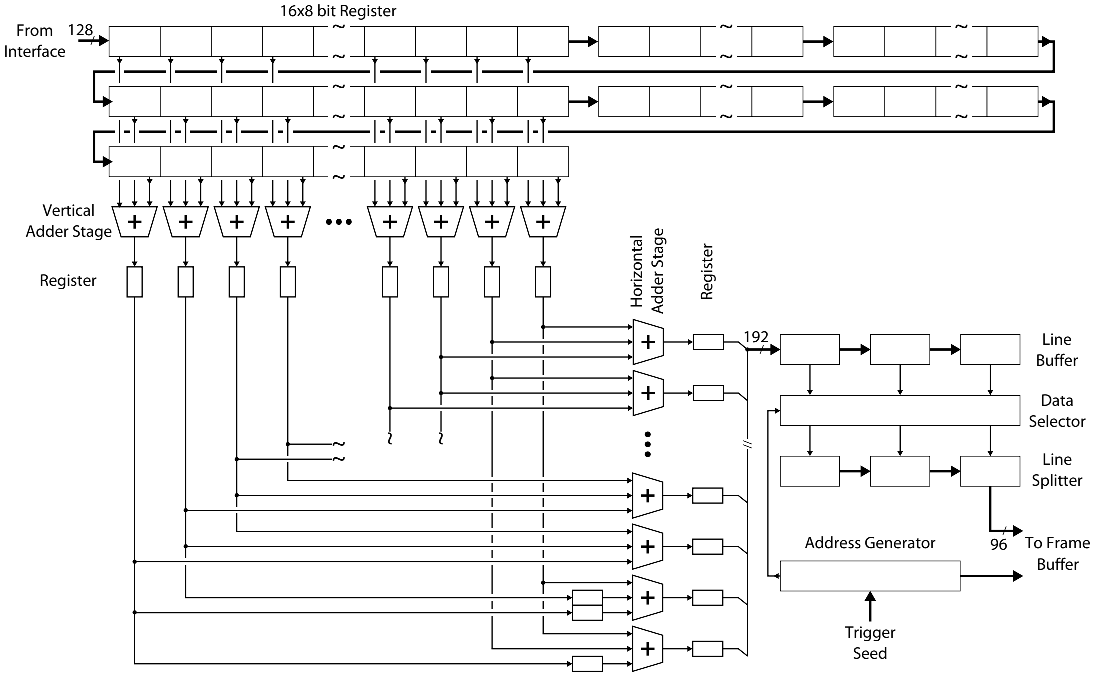


Figure 4.8: The block diagram of the '3x3 Sum' module, which basically performs the summation of the pixels inside the 3x3 integration box. This is done for the whole PDM frame. Therefore, a 'new' frame is generated, where each pixel contains the summed up value of its surrounding pixels. The second part of this module will then select the necessary data for the direction integration (depending on the trigger seed) and writes it to the 'Frame Buffer'. Imagine a small window, which has the size of the necessary data and is centered around the trigger seed coordinates. The part of the frame which is visible through this window, will be stored inside the 'Frame Buffer'.

4.3.1 Vertical Adder Stage

Based on performance issues, the 3x3 summation is done completely pipelined in two stages. The first stage performs the summation of three pixels one below the other and for 16 columns in parallel (as 16 pixels are received per clock cycle). This is achieved, by feeding sixteen 3-input-8 bit full adders with the output of a large shift-register. Each stage of this shift-register is composed out of sixteen 8 bit registers, to hold the 16 pixels from one data block and a total of 7 stages is needed to hold two complete lines and the beginning of the third (one PDM-frame is 48 pixels wide, therefore, three data blocks are needed for one line) - so the output from stage 1, 4 and 7 is the input for the adders (see Figure 4.8). The output from the adders is then sent to latches, to account for their propagation delay, therefore, to gain performance.

4.3.2 Horizontal Adder Stage

The 'horizontal' adder stage, finally sums up the output from three adjacent 'vertical' adders and therefore generates the final 3x3 sum. The problem at this stage, is that there is some overlap between the single 16 pixel blocks. For example, to calculate the sum of pixel 16, 17 and 18, the information from the next data block will be needed, which arrives one clock cycle later. This is solved by adding an additional register stage in front of some inputs of the affected adders (see the last two adders at the bottom of see Figure 4.8). The output from the adders is again connected to latches due to the performance.

4.3.3 Data Selection & Address Generator

The last part of the '3x3 Sum' module is the most complex one, even though its purpose is quite simple. The goal was, to have a 'Frame Buffer', which contains only the necessary data for the final direction integration to minimize the amount of required internal RAM of the FPGA.

The data needed for the direction integration depends on the maximum offset from GTU to GTU of the box, the time over which it will be integrated and on the amount of starting points (including their distribution). For the current baseline, a frame with 19x19 pixels for 29 GTUs around the trigger seed is needed. Therefore, these data have to be selected out of the stream coming from the last adder stage, which is done in several steps.

The data of one complete line are collected inside a shift-register 'Line Buffer' and transferred to the 'Data Selector'. This module will then trim the line, or extend it (with zeros/some predefined value) if the trigger seed is at the edge and load it again into a shift-register, from where it will be transferred to the 'Frame Buffer'. Finally, the 'Address Generator' produces the necessary control signals for the 'Frame Buffer'.

As a remark, the implementation of the data selection is more complicated than this principle - for example it also has to be decided which sums are valid or the top/bottom part of the 'Frame Buffer' has to be filled with zeros/some predefined value if the trigger seed is at the top/bottom edge of the frame and the time for this filling process has to be found. But altogether it is worth the effort, as this drastically simplifies the direction integration process.

4.4 Frame Buffer & Integration

The last stage of the L2 trigger algorithm is the 'Integrator', which finally performs the direction integration over time. Due to the effort on the 'Data Selection' and 'Address Generator' of the '3x3 Sum' module, the 'Frame Buffer' contains now some kind of 'new' frames, which are centered around the trigger seed. This makes the 'Integrator' module

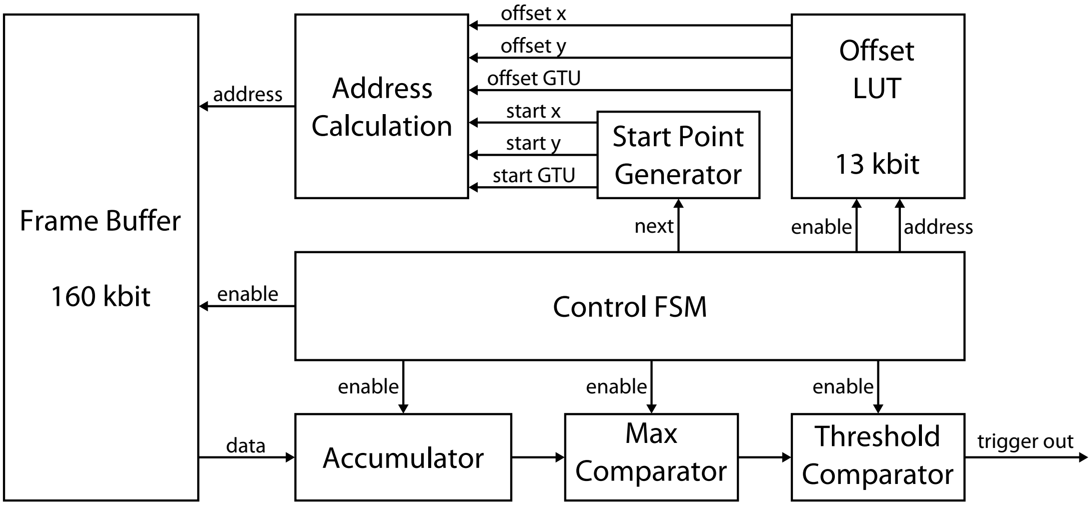


Figure 4.9: The ‘Integrator’ stage of the L2 implementation performs the direction integration. The address inside the ‘Frame Buffer’ of the correct pixel for the accumulation is based on the direction offsets, which are stored inside a Look Up Table. At the end of the integration process, the maximum value found is compared to the trigger threshold and a trigger is generated in the case that this threshold is exceeded.

completely independent from the trigger seed and therefore simplifies the integration process.

The basic principle of this module (see also Figure 4.9) is to read out the correct ‘pixels’ one after the other - which contains the summed up value of the surrounding ones - accumulate them for each direction, select the maximum value, compare this maximum value at the end of the integration process with the trigger threshold and issue a trigger if the threshold is exceeded.

4.4.1 Frame Buffer & Direction LUT

The ‘Frame Buffer’ is a simple dual port RAM inside the FPGA, with a 96 bit wide write port and a 12 bit wide read port. The large write port is necessary to store the data coming from the ‘3x3 Sum’ module in ‘real time’, while the smaller read port allows to read out exactly one pixel after the other which will be used for the direction integration.

The offsets for the integration box are precalculated for each direction and GTU and stored inside the FPGA as a LUT. In the current implementation this is a simple Read Only Memory (ROM) but for the final implementation the LUT will be made configurable (e.g. by the CPU) to enable the possibility of tuning these offsets - for example to adapt them to the height of the ISS orbit.

4.4.2 Address Calculation

To read out the correct pixel from the 'Frame Buffer', which depends on the current direction, start point and GTU, a transformation between the coordinates (including the GTU) and the address inside the 'Frame Buffer' has to be made. This calculation is performed inside the 'Address Calculation' module, which receives the current offsets - offset x, offset y and offset GTU - from the 'Offset LUT' and the current start point - start x, start y and start GTU - from the 'Start Point Generator' (which is basically a counter). The calculation itself is done in four pipelined steps:

1. The current coordinates relative to the center (trigger seed) of 'Frame Buffer' (x, y and GTU) were calculated by simply summing the offsets and the start coordinates (separately for each coordinate).

2. These coordinates were 'shifted' relative to the top left corner of the first 'Frame Buffer' frame as address zero corresponds to the offset coordinates x = -9, y = -9 and GTU = -14 - which is done by subtracting these values from the coordinates.

3. In the third step, the single components of the address (pixel-address, line-address and frame-address) are calculated by multiplying the y-coordinate with the width of the frame and GTU-coordinate with the width and height. The x-coordinate corresponds already to the pixel-address and is therefore only registered.

4. In the final step, the complete address is calculated by simply summing up the single components.

4.4.3 Accumulator, Comparators & Control FSM

The actual integration is done with a simple accumulator, which receives the pixel data from the 'Frame Buffer'. After each direction is completed, the accumulated value is send to the 'Max Comparator' which simply selects the maximum integrated value by comparing it to the previous (current maximum) one. Finally, when the integration is done for all directions and starting points, the maximum value is sent to the 'Threshold Comparator', which compares it to the trigger threshold and issues a trigger if the threshold is exceeded.

The 'Control FSM' generates the address for the 'Offset LUT' and controls all the submodules by enabling them at the right clock cycle, as there are various latencies from the 'Offset LUT', 'Address Calculation' and 'Frame Buffer'.

5 Test Setup and Results

During the hardware development all submodules were continuously tested by performing mainly behavioral simulations of the VHDL code, to verify the correct operation. The present chapter will describe two major hardware tests and the software which was developed to verify the functionality and to simplify the hardware development.

5.1 Frame Analyzer Software

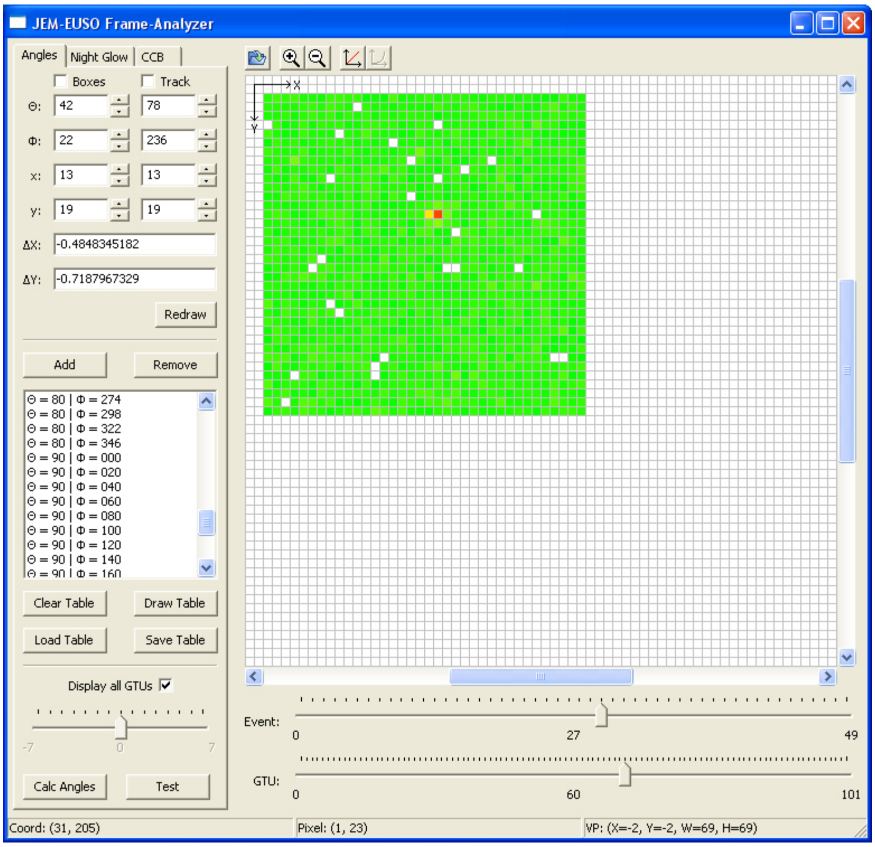
This software (see Figure 5.1) was written to simplify various tasks during this work. Its main purpose was the offset calculation for the L2 trigger algorithm (see Section 3.2) and the generation of the direction set with the algorithm described in Section 3.3. It is also capable to read files with information on events which are generated by the ESAF and to display them. Together with the possibility to load different direction sets this software was used for analysis and comparison of the generated direction set with the 'baseline' set (see Subsection 3.3.2).

During the hardware development the functionality of the software was extended, so it was possible to generate various files containing data to verify the simulation of the VHDL code (e.g. the two address generators for the 'Frame Buffer' described in Subsection 4.3.3 and 4.4.2). Besides the data generation for the simulation, the software was also used to generate the memory files for the 'Direction LUT' (see Subsection 4.4.1) and for the 'Event Buffer' (see Subsection 5.3).

5.2 Bit Error Ratio Test

To qualify the interface, a test setup was developed which is able to send a large amount of data over the interface and to detect transmission errors. It was also important to include the cable into the tests as it could (together with the connectors) reduce the maximum transfer rate dramatically (e.g. due to impedance mismatch). To account for the requirement of the distance from the interface, a 1 m high speed data cable from SAMTEC was used to transfer the data to the receiver.

The block diagram of the Bit Error Ratio Test (BERT) is shown in Figure 5.2. A 'Pattern Generator' generates the data for the transmitter and an 'Error Detector' collects the data from the receiver to detect and count incorrect bits. Both modules are controlled and monitored with the 'Chip Scope' module. This is basically an Integrated



□ Boxes □ Track

Θ: 42 78

Φ: 22 236

x: 13 13

y: 19 19

ΔX: -0.4848345182

ΔY: -0.7187967329

Θ = 80 | Φ = 274
Θ = 80 | Φ = 298
Θ = 80 | Φ = 322
Θ = 80 | Φ = 346
Θ = 90 | Φ = 000
Θ = 90 | Φ = 020
Θ = 90 | Φ = 040
Θ = 90 | Φ = 060
Θ = 90 | Φ = 080
Θ = 90 | Φ = 100
Θ = 90 | Φ = 120
Θ = 90 | Φ = 140
Θ = 90 | Φ = 160

Display all GTUs ☑

-7 0 7

Event: 0 27 49

GTU: 0 60 101

Coord: (31, 205) Pixel: (1, 23) VP: (X=-2, Y=-2, W=69, H=69)

Figure 5.1: The purpose of this software was mainly to simplify the calculation and analyses of the direction set for the L2 trigger algorithm. During the hardware development it was then extended in functionality allowing to generate various files (e.g. for the verification of the simulation results or the memory files for the LUT).

Logic Analyzer (ILA) which can be implemented within the FPGA and allows to trace internal signals and to send the captured data to a computer. Additionally, a Virtual Input/Output (VIO) module (which allows sending and reading data to/from the FPGA with a computer) is used to set up the parameters for the 'Pattern Generator' (e.g. which pattern will be generated) and to read out the number of errors from the 'Error Detector'. Besides these main tasks, this module is also used to control and monitor other implemented features such as injecting errors and manually de-/incrementing the IDELAY primitives and reading back their actual settings.

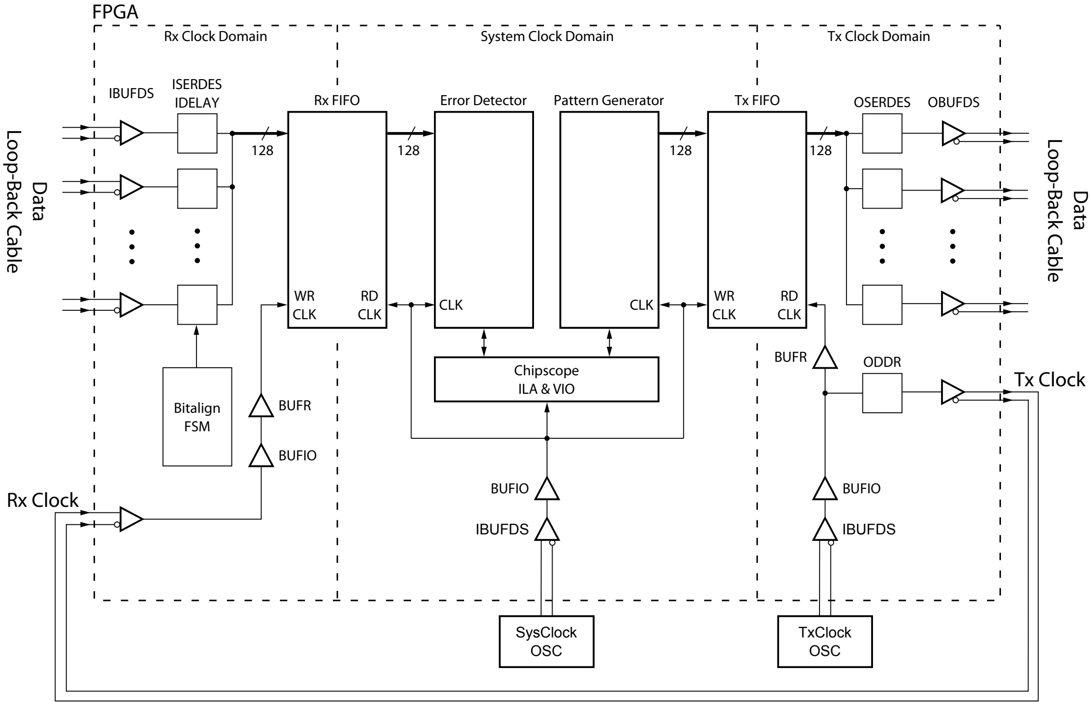


Figure 5.2: Block diagram of the BERT with the transmitter on the right and the receiver on the left side. A 'Pattern Generator' produces the data to be sent over the interface (including the cable) and an 'Error Detector' compares the received data with the sent ones (after generator and detector are synchronized) and counts the number of received bits together with the transmission errors. The Chipscope module provides an interface to the computer so generator and detector can be controlled and monitored.

5.2.1 Pattern Generator

This module produces Pseudo Random Bit Sequences (PRBSs) and other predefined patterns (e.g. different clock patterns), that can be sent through the interface. These PRBSs are useful to simulate real data as they contain a wide range of frequency components and are widely used for testing communication channels due to the fact that they are easy to implement. Moreover they are deterministic which is necessary to detect errors (see also Telecommunication Standardization Sector, 1996).

To generate a PRBS, a shift register is fed with the exclusive-OR (XOR) result from predefined register positions within the shift register (also called taps), which forms a feedback network and therefore is named Linear Feedback Shift Register (LFSR). Depending on the number and position of the taps, a maximum length LFSR can be constructed, which cycles through all possible $2^n - 1$ states of the shift register (with n being the number of registers), except the state where all bits are zero. This state is illegal because the LFSR would remain locked-up in this state as 0 XOR $0 = 0$.[1] More details on LFSRs and their implementation can be found in George and Alfke (2007) and Alfke (1996).

The 'Pattern Generator' implemented in this design is able to generate PRBSs with LFSRs of different sizes: PRBS7, PRBS15, PRBS23 and PRBS29. In addition it is possible to manually inject errors to be able to test the 'Error Detector'.

5.2.2 Error Detector

This component inherits the same LFSRs as the 'Pattern Generator' - so it is capable to generate the same PRBS after the two LFSRs are synchronized. This synchronization is handled by a FSM which stops (or slows down) the detector-LFSR until the received PRBS (produced by the generator-LFSR) matches the one of the detector. This process can last several seconds (depending on the size of the LFSR), as in the worst case the complete cycle of $2^n - 1$ states has to be traversed. When both LFSR are in phase the received data are continuously compared (XORed) to the one generated by the detector LFSR to detect and count the bit errors.

In addition to the error counter, the detector also counts the received frames so the amount of transmitted data is known. With these two values the Bit Error Ratio (BER) can be calculated and is given by

$$\text{BER} = \frac{\text{number of bit errors}}{\text{number of received bits}}$$

5.2.3 Results

The BERT was implemented in the Virtex4-FX100 and the cable under test was a 1 m 'High Speed Micro Coax Cable' from SAMTEC connected with high speed connectors (QSH-060-LDAK) on the board. A big challenge of this setup was that the signals are neither routed as differential pairs nor are they length matched on this board. Due to this, differential pairs with a differential mismatch as small as possible (the biggest mismatch in the current design is about 7.6 mm) were chosen, which led to bad routing within the FPGA. On the other hand these problems could be optimized in the final hardware which should lead to much better results. Moreover, the length mismatch between the channels of the bus is compensated by the 'Bit Align Machine'.

[1] The feedback can also be done with exclusive-NOR (XNOR), in this case, the illegal state is where all bits are one.

5.2.3.1 Transfer Rate

The first test was dedicated to measure the data eye width for different transfer rates. This was done by manually de-/incrementing the tap setting of the IDELAY module for each channel until the received data were no longer stable indicating the edge of the data eye. The width of the data eye is then given by

$$\text{data eye width} = (\text{max. tap} - \text{min. tap}) \cdot 75\,\text{ps}.$$

These values (measured with PRBS29) represent only a rough estimation to get an idea on the stability of the link and are summarized in Table 5.1.

As can be seen, even at 800 Mbit s^{-1} the data eye is still big enough (±3 tap) to allow some changes over temperature and supply voltage (see also Tables 12 ff. in Burton, 2006).

Another interesting fact is that the mean lower tab is 10 ± 1 indicating that the data is arriving around 750 ps before the clock, which is mainly caused by different propagation delays from the different primitives in the clock and data paths (and also due to routing within the FPGA). For more details please look at Tables A.2-A.6.

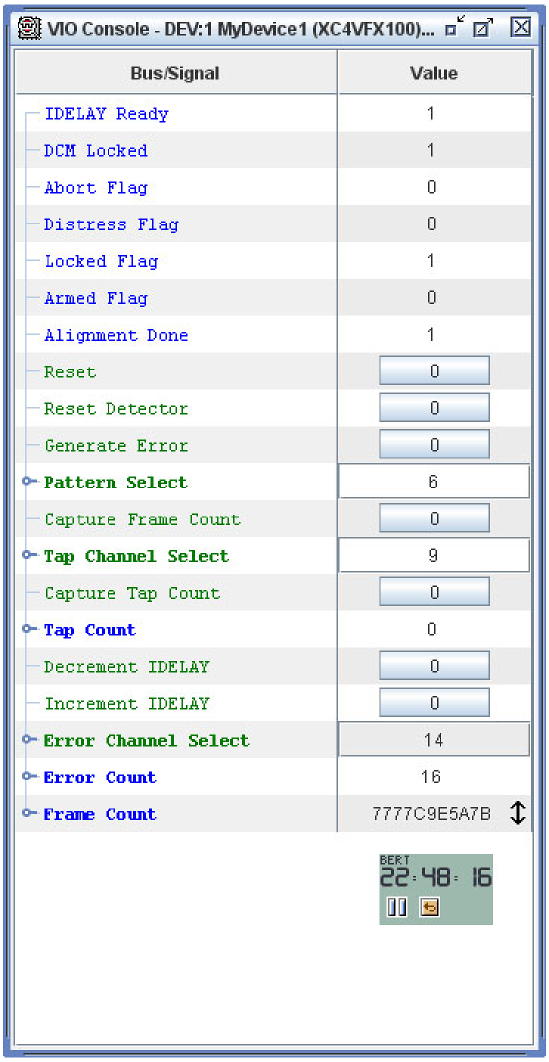
5.2.3.2 Bit Error Ratio

The second test was a long term run of the interface to calculate the BER and the confidence level. As this test is quite time consuming, it was only done for 800 Mbit s^{-1} channel speed and PRBS29. A screenshot from ChipScope Pro with a little stop watch (bottom right corner) at the end of the test is given in Figure 5.3. As can be seen[2], the interface transmitted over 131 TB in nearly 23 hours without any single bit error (the 16 erroneous bits in the 'Error Count' row were generated manually after approximately

Table 5.1: Data Eye width for different transfer rates measured with PRBS29.
*The data eye width for 200 Mbit s^{-1} is probably slightly higher because the maximum tap setting of the IDELAY module was reached as can be seen from Table A.2.

Interface Speed Gbit s^{-1}	Channel Speed Mbit s^{-1}	Theoretical Width ps	Mean Width ps
3.2	200	5000	3994*
4.8	300	3333	2606
6.4	400	2500	1819
9.6	600	1667	872
12.8	800	1250	469

[2] One frame is 128 bit and therefore: 0x7777C9E5A7B · 128 bit ≈ 1 Pbit ≈ 131 TB.



VIO Console - DEV:1 MyDevice1 (XC4VFX100)...

Bus/Signal	Value
IDELAY Ready	1
DCM Locked	1
Abort Flag	0
Distress Flag	0
Locked Flag	1
Armed Flag	0
Alignment Done	1
Reset	0
Reset Detector	0
Generate Error	0
Pattern Select	6
Capture Frame Count	0
Tap Channel Select	9
Capture Tap Count	0
Tap Count	0
Decrement IDELAY	0
Increment IDELAY	0
Error Channel Select	14
Error Count	16
Frame Count	7777C9E5A7B

Figure 5.3: Screenshot from the ChipScope Analyzer software during the long term run of the interface. This allows controlling and monitoring of the ‘Pattern Generator’, ‘Error Detector’ and the interface. Blue entries are outputs from the BERT and green entries inputs to the BERT. For example, the ‘Abort’-, ‘Distress’-, ‘Locked’-, and ‘Armed’-Flag are status signals of the ‘Error Detector’. The ‘Alignment Done’ signal is indicating that the ‘Bit Align Machine’ of the receiver has correctly finished and the ‘Error Count’ together with the ‘Frame Count’ are coming from the ‘Error Detector’.

20 hours to verify that everything is still working). This leads to a BER better than 10^{-14} with a confidence level of 99.997%.

5.3 Linear Track Trigger Test

To verify the correct operation of the 'Linear Track Trigger' module, a first simple test was developed where the 'PDM Interface' was replaced by a 'PDM Simulator' module (see Figure 5.4). This module consists of a small FSM and the 'Event Buffer' which is basically a large RAM, filled with the simulated data of an event generated by the ESAF.

The data from the 'Event Buffer' are sent to the input of the 'Linear Track Trigger' module where it will be processed as described in Section 4.3 and the following. To have a more detailed look into the processing, the 'Accumulator', 'Maximum Comparator' and 'Threshold Comparator' are connected either to a ChipScope VIO or an ILA module.

A screenshot from the VIO module of the ChipScope Analyzer software is shown in Figure 5.5. The trigger seed ('Seed GTU0', 'Seed x' and 'Seed y') is also stored inside the 'Event Buffer' as a first data frame, and the 'Integration Maximum' is the overall maximum value after the integration is done for all directions and start points (output of the 'Maximum Comparator'). As the trigger output lasts only one clock cycle, its transition is indicated by the up-down-arrow (same applies to the signals 'Integration End' and 'Event Send'). Besides the data output, the VIO module also enables the

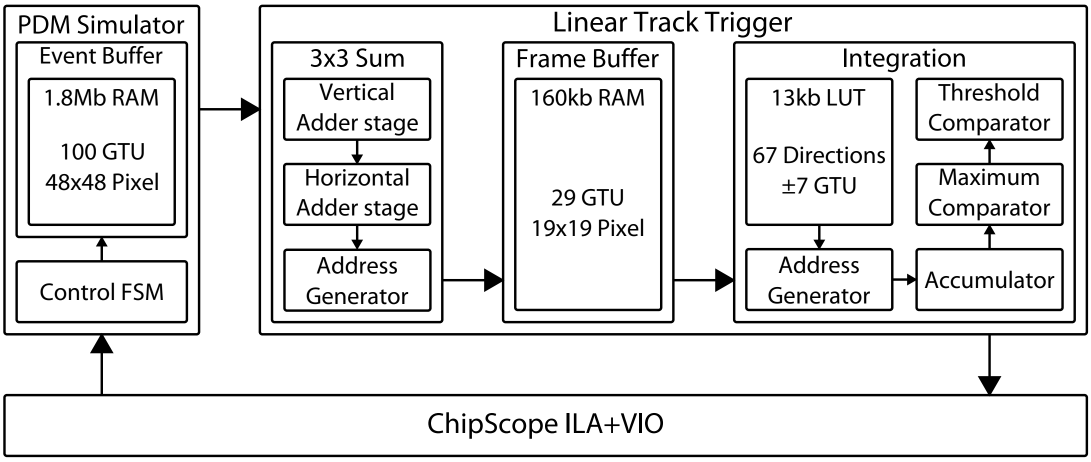
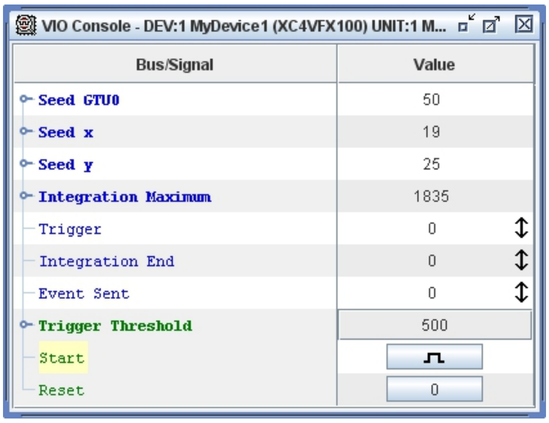


Figure 5.4: The block diagram for the test of the 'Linear Track Trigger'-module. The 'PDM Interface' was replaced by the 'PDM Simulator' module, which is basically a large RAM containing the data of an event generated by the ESAF. A ChipScope ILA and VIO module is used to read out the data from the 'Accumulator' and to verify the functionality of the 'Linear Track Trigger' module.

possibility to set the 'Trigger Threshold' and to start the transmission of the 'Event Buffer'.

The output of the 'Accumulator' is sampled by the ILA module every time the integration of one direction is finished. This allows to compare every single integrated value with the ones produced by the 'Frame Analyzer' software and to verify the correct operation of the 'Linear Track Trigger'. Unfortunately, the sample buffer of the ILA module is limited (as it uses the internal RAM of the FPGA) and it was not possible (with this simple test) to store all 25125 integrator values. Therefore, only around 16000 integrator values are plotted in Figure 5.6. But as all sampled values (including the overall maximum) are equal to the ones of the 'Frame Analyzer' software, it is assumed that this test was passed successfully - also due to the fact, that the correct behavior of the the 'Linear Track Trigger' module was verified in advance with simulations.



VIO Console - DEV:1 MyDevice1 (XC4VFX100) UNIT:1 M...

Bus/Signal	Value
Seed GTU0	50
Seed x	19
Seed y	25
Integration Maximum	1835
Trigger	0 ↕
Integration End	0 ↕
Event Sent	0 ↕
Trigger Threshold	500
Start	⎍
Reset	0

Figure 5.5: Screenshot of the VIO module of the ChipScope Analyzer software. The first three rows is the trigger seed (which is also sent by the 'PDM Simulator'), the 'Integrator Maximum' is the overall maximum after the integration and the 'Trigger' is the output of the 'Threshold Comparator'. The arrows are indicating a transition of signals which are faster than the VIO update period (e.g. one clock cycle).

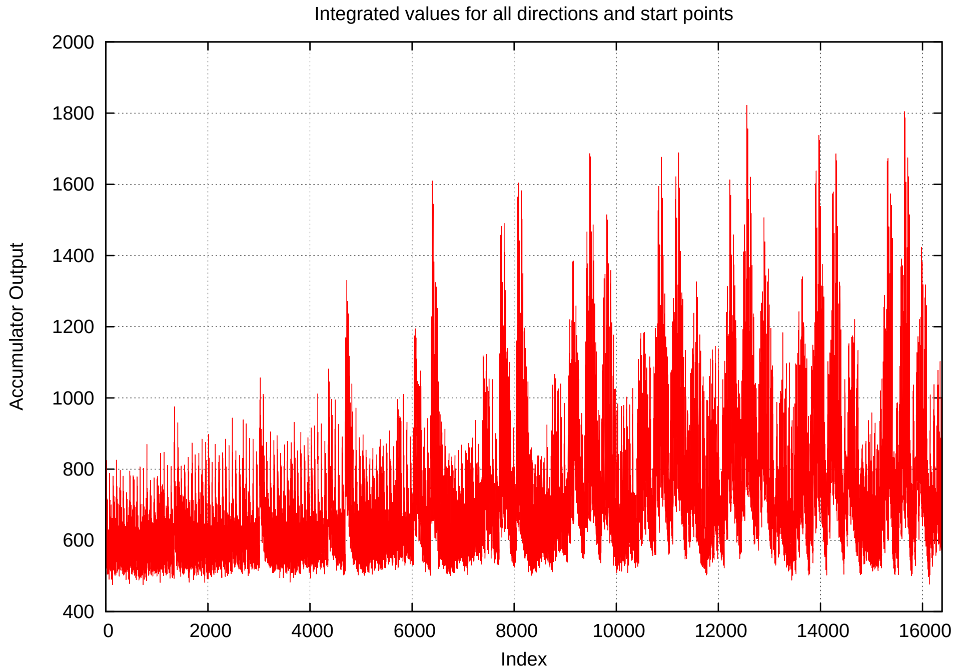


Figure 5.6: The output of the 'Accumulator' which is sampled by the ILA module. As the integration starts with the most distant start point from the trigger seed, the integrated value is quite low until around sample 4000. Beyond that point, the trigger algorithm starts to produce the peaks as described in Subsection 3.3.2.

Conclusion & Outlook

The Cluster Control Board is a key element of the JEM-EUSO electronics. Its design, implementation of the functionalities and its testing has triggered this work.

The developed hardware is in an early stage and a long way is necessary to reach the final space-qualified Cluster Control Board. However many conclusions have been reached with this work.

The hardware implementation of the 'Linear Track Trigger' algorithm as L2 trigger is performing very well. Based on the separation of the calculation process in a parallel and a serial part, the requirement on the processing time of the L2 trigger could be exceeded by a factor of 20. It should be stressed, that this value is highly dependent on the overall implementation of the CCB FPGA but at least the requirement was met.

The interface developed during this work could be a possible interface between the PDMs and the CCB on account of its simplicity and its performance. The whole serialization and deserialization process is handled by the hardware primitives of the FPGA which minimizes the use of additional logic. Actually, the logic consumed by the 'Bit Align Machine' could be either omitted as the tap settings of the IDELAY modules can be set to a fixed value after routing and cabling is done or could be used to account for changes over temperature and supply voltage.

One problem still to be discussed is the clocking of the interface. As the current PDM-FPGA runs at 40 MHz system clock, this leads (for the configuration described here) to a maximum transfer rate of $320\,\mathrm{Mbit\,s^{-1}}$ which is too slow ($5.12\,\mathrm{Gbit\,s^{-1}}$ compared to $7.4\,\mathrm{Gbit\,s^{-1}}$). On the other hand there are still some parameters which allow further customization of the interface (e.g. serialization ratio and channel count). As the interfaces with the PDMs and the MDP are crucial parts for the further development of the CCB, it is suggested to define these interfaces as soon as possible.

The next steps of the CCB development is the implementation of the 'slow event trigger' (e.g. for meteoroids and TLEs) which is also not yet clearly defined and the designing of a first laboratory model of the CCB PCB. As the funding of this board is already clarified it could be possibly done in the next several months.

This work can be considered as just the beginning of development of the JEM-EUSO CCB. It is obvious, but necessary to say that this work will hopefully end when the JEM-EUSO mission will fly!

Acknowledgements

I would like to express my gratitude to all the people who supported me during the time of the thesis and my studies. I really enjoyed the time I had at the institute and it was a great experience for me. Especially, I am pleased to thank the following people:

> Prof. Dr. Andrea Santangelo, for giving me the possibility to contribute to such an exciting project, for all the experiences I gained, especially during the meetings all around the world and for helping me with the English.
>
> Thomas Schanz & Dr. Christoph Tenzer, for introducing me in VHDL and FPGAs, helping me with the English and always having their door open for all issues - a good portion of the nice atmosphere at the institute is their contribution. And of course, for the possibility to contribute to TOBOR.
>
> Dr. Eckhard Kendziorra, for introducing me to the institute and supplying me with hardware-related jobs whereby I attained a lot of knowledge in electronics, which really helped me during the quite theoretical studies. And of course, for sharing his (not only hardware related) experiences and all the anecdotes.
>
> Jürgen Dick & Siegfried Vetter, for sharing their knowledge on electronics and PCB layout with me.
>
> Mario Bertaina & Francesco Fenu, for helping me with the trigger algorithm, especially the theoretical part.
>
> Alessandro Zuccaro Marchi, for 'taking care' of me in foreign countries - I really enjoyed the time.
>
> Fam. Wax & Fam. Zanette, for always welcoming and supplying me, especially in bad or stressful times - I really appreciate the great friendship.

And finally, I want to thank my family who made all this possible.

Thank you.

Bibliography

Abestrobi. Representation of upper-atmospheric lightning and electrical-discharge phenomena. Online, June 2008. URL http://en.wikipedia.org/wiki/Transient_luminous_event.

F. A. Aharonian and J. W. Cronin. Influence of the universal microwave background radiation on the extragalactic cosmic-ray spectrum. *Phys. Rev. D*, 50(3):1892–1900, August 1994. doi: 10.1103/PhysRevD.50.1892. URL http://link.aps.org/doi/10.1103/PhysRevD.50.1892.

S. Ahmad, P. Barrillon, S. Blin, S. Dagoret, F. Dulucq, C. de La Taille, Y. Kawasaki, and I. Hirokazu. JEM-EUSO ASIC: SPACIROC measurements results. Presentation, October 2010.

Peter Alfke. Efficient Shift Registers, LFSR Counters, and Long Pseudo-Random Sequence Generators. Application Note 052, Xilinx, July 1996.

P. Barrillon, S. Ahmad, , S. Blin, D. Cuisy, S. Dagoret, F. Dulucq, C. de La Taille, R. Sliwa, and J-L. Socha. EC board for JEM-EUSO: Design investigations. Presentation, October 2010.

C. L. Bennett, M. Halpern, G. Hinshaw, N. Jarosik, A. Kogut, M. Limon, S. S. Meyer, L. Page, D. N. Spergel, G. S. Tucker, E. Wollack, E. L. Wright, C. Barnes, M. R. Greason, R. S. Hill, E. Komatsu, M. R. Nolta, N. Odegard, H. V. Peiris, L. Verde, and J. L. Weiland. First-year wilkinson microwave anisotropy probe (wmap) observations: Preliminary maps and basic results. *The Astrophysical Journal Supplement Series*, 148(1):1, 2003. URL http://stacks.iop.org/0067-0049/148/i=1/a=1.

D. J. Bird, S. C. Corbato, H. Y. Dai, J. W. Elbert, K. D. Green, M. A. Huang, D. B. Kieda, S. Ko, C. G. Larsen, E. C. Loh, M. Z. Luo, M. H. Salamon, J. D. Smith, P. Sokolsky, P. Sommers, J. K. K. Tang, and S. B. Thomas. Detection of a Cosmic Ray with Measured Energy Well Beyond the Expected Spectral Cutoff due to Cosmic Microwave Radiation. *Astrophysical Journal*, 441:144–150, March 1995. doi: 10.1086/175344. URL http://arxiv.org/abs/astro-ph/9410067.

Klaus Bittermann. Studies on Detection of Neutrinos from Space: The Capabilities of JEM-EUSO. Diploma thesis, University of Tübingen, July 2010.

Greg Burton. 16-Channel, DDR LVDS Interface with Per-Channel Alignment. Application Note 855, Xilinx, October 2006.

Francesco Fenu. A simulation study of space based missions for Ultra High Energy Cosmic Ray search. Diploma thesis, University of Tübingen, June 2008.

Maria George and Peter Alfke. Linear Feedback Shift Registers in Virtex Devices. Application Note 210, Xilinx, April 2007.

Kenneth Greisen. End to the Cosmic-Ray Spectrum. *Phys. Rev. Lett.*, 16(17):748–750, April 1966. doi: 10.1103/PhysRevLett.16.748. URL http://link.aps.org/doi/10.1103/PhysRevLett.16.748.

William F. Hanlon. *The Energy Spectrum of Ultra High Energy Cosmic Rays measured by the High Resolution Fly's Eye Observatory in stereoscopic mode.* PhD thesis, University of Utah, December 2008. URL http://www.physics.utah.edu/~whanlon/spectrum.html.

A. M. Hillas. The Origin of Ultra-High-Energy Cosmic Rays. *Annual Review of Astronomy and Astrophysics*, 22:425–444, 1984. doi: 10.1146/annurev.aa.22.090184.002233.

JEM-EUSO Collaboration. Report on the Phase A Study. Technical report, 2010.

John Linsley. Evidence for a Primary Cosmic-Ray Particle with Energy 10^{20} ev. *Phys. Rev. Lett.*, 10(4):146–148, February 1963. doi: 10.1103/PhysRevLett.10.146. URL http://link.aps.org/doi/10.1103/PhysRevLett.10.146.

John Linsley, Livio Scarsi, and Bruno Rossi. Extremely Energetic Cosmic-Ray Event. *Phys. Rev. Lett.*, 6(9):485–487, May 1961. doi: 10.1103/PhysRevLett.6.485. URL http://link.aps.org/doi/10.1103/PhysRevLett.6.485.

Paolo Lipari. Problems in High Energy Astrophysics, August 2008. URL http://arxiv.org/abs/0808.0417.

Thomas Mernik. Reconstruction of UHECR Events for the JEM-EUSO Mission. Diploma thesis, University of Tübingen, December 2009.

Kenji Shinozaki. Impacts of natural & artificial light sources and related simulation activties. Presentation, May 2010.

A. D. Supanitsky and G. Medina-Tanco. Extreme high energy proton-gamma discrimination from space observations. *ArXiv e-prints*, February 2011.

Gustavo A. Medina Tanco, Elisabete M. de Gouveia Dal Pino, and Jorge E. Horvath. Deflection of ultra-high-energy cosmic rays by the galactic magnetic field: From the sources to the detector. *The Astrophysical Journal*, 492(1):200, 1998. URL http://stacks.iop.org/0004-637X/492/i=1/a=200.

Telecommunication Standardization Sector. Specifications of measuring equipment. Recommendation O.150, International Telecommunication Union, May 1996.

Thales Alenia Space. JEM-EUSO: FSA (Focal Surface Assembly). Presentation, October 2010.

Katsuhiko Tsuno. Data Generation Budget Proposal for JEM-EUSO. Presentation, December 2010.

Xilinx. ML450 Bit Error Rate Tester (BERT) User Guide. User Guide 089, Xilinx, December 2005.

Xilinx. Advanced ChipSync Applications. Application Note 707, Xilinx, October 2006.

Xilinx. Virtex-4 FPGA User Guide. User Guide 070, Xilinx, December 2008.

Xilinx. Space-Grade Virtex-4QV FPGAs: DC and Switching Characteristics. Product Specification 680, Xilinx, April 2010.

G. T Zatsepin and V. A Kuz'min. Upper Limit of the Spectrum of Cosmic Rays. *Journal of Experimental and Theoretical Physics Letters*, 4:78–80, August 1966. URL http://www.jetpletters.ac.ru/ps/1624/article_24846.shtml.

Alessandro Zuccaro Marchi and Yoshiyuki Takizawa. JEM-EUSO: Updates on Optics Module. Presentation, December 2010.

List of Acronyms

ABIU	ASIC Bus Interface Unit
AMS	Atmospheric Monitoring System
ASIC	Application Specific Integrated Circuit
BER	Bit Error Ratio
BERT	Bit Error Ratio Test
CBIU	Cluster control board Bus Interface Unit
CCB	Cluster Control Board
CIU	CPU Interface Unit
CMBR	Cosmic Microwave Background Radiation
CPU	Central Processing Unit
CR	Cosmic Ray
CRU	Calibration Run Unit
CYTOP	Amorphous Peffluoro Alkenylvinylether
DCM	Digital Clock Manager
DDR	Double Data Rate
DPU	Data Processing Unit
EAS	Extensive Air Shower
EC	Elementary Cell
EEPROM	Electrically Erasable Programmable Read-Only Memory
EF	Exposed Facility
ESA	European Space Agency
ESAF	EUSO Simulation and Analysis Framework
EUSO	Extreme Universe Space Observatory
FEE	Front End Electronics
FoV	Field-of-View
FPGA	Field Programmable Gate Array
FS	Focal Surface
FSM	Finite State Machine
GMF	Galactic Magnetic Field
GTU	Gate Time Unit
GZK	Greisen-Zatsepin-Kuzmin
HIU	Housekeeping Interface Unit
HK	Housekeeping
HTV	H-II Transfer Vehicle

IDAQ	Interface Data AcQuisition
ILA	Integrated Logic Analyzer
IR	Infra Red
ISS	International Space Station
JAXA	Japan Aerospace Exploration Agency
JEM-EUSO	Extreme Universe Space Observatory on board Japanese Experiment Module
JEM	Japanese Experiment Module
LFSR	Linear Feedback Shift Register
LHC	Large Hadron Collider
LIDAR	LIght Detection And Ranging
LTT	Linear Track Trigger
LUT	Look Up Table
LVDS	Low Voltage Differential Signaling
MAPMT	Multi Anode PhotoMultiplier Tube
MDP	Mission Data Processor
ODDR	Output Double Data rate Register
OM	Optics Module
PCB	Printed Circuit Board
PD	Photo-Detector
PDM	Photo-Detector Module
PMCD	Phase Matched Clock Divider
PMMA	PolyMethylMethAcrylate
PRBS	Pseudo Random Bit Sequence
PTT	Progressive Track Trigger
RAM	Random Access Memory
ROM	Read Only Memory
RS	Run Summary
SC	System Control
SCU	Storage and Control Unit
SDR	Single Data Rate
SHP	Super-Heavy Particle
SPACIROC	Spatial Photomultiplier Array Counting and Integrating Read Out Chip
SRAM	Static Random Access Memory
TBC	To Be Confirmed
TBD	To Be Defined
TLE	Transient Luminous Event
TU	Trigger Unit
UHECR	Ultra High Energy Cosmic Ray
UV	UltraViolet
VHDL	Very high speed integrated circuit Hardware Description Language

VIO	Virtual Input/Output

Appendix

A.1 Direction Set of the L2 Algorithm

Table A.1: This list contains the reduced set of direction for the L2 Algorithm, which is implemented in Hardware.

θ	ϕ
10	45
27	22
27	112
27	202
27	292
43	20
43	65
43	110
43	155
43	200
43	245
43	290
43	335
57	15
57	55
57	95
57	135
57	175
57	215
57	255
57	295
57	335
69	12
69	42
69	72
69	102
69	132
69	162
69	192
69	222
69	252
69	282
69	312
69	342
80	10
80	34
80	58
80	82
80	106
80	130
80	154
80	178
80	202
80	226
80	250
80	274
80	298
80	322
80	346
90	0
90	20
90	40
90	60
90	80
90	100
90	120
90	140
90	160
90	180
90	200
90	220
90	240
90	260
90	280
90	300
90	320
90	340

A.2 BERT Data Eye Width Measurement

The following tables contain the detailed results from the data eye width measurement for different transfer rates. The middle tap position is the one set by the 'Bit Align Machine', the lower and upper tap position are determined by manually de-/incrementing the delay until the data were no longer stable which indicates the transition region.

As already remarked in Table 5.1, the highest delay (tap position 63) was reached for 200 Mbit s^{-1} while the received data were still stable. Therefore the data eye was probably slightly wider.

Table A.2: Results from 200 Mbit s^{-1} per channel

Channel	Middle Tap	Lower Tap	Upper Tap	Data Eye width [ps]
0	36	10	63	3975
1	36	10	63	3975
2	37	10	63	3975
3	36	9	63	4050
4	36	10	63	3975
5	36	10	63	3975
6	36	10	63	3975
7	36	10	63	3975
8	36	9	63	4050
9	36	9	63	4050
10	37	10	63	3975
11	36	9	63	4050
12	36	10	63	3975
13	36	10	63	3975
14	36	10	63	3975
15	36	10	63	3975

Table A.3: Results from 300 Mbit s^{-1} per channel

Channel	Middle Tap	Lower Tap	Upper Tap	Data Eye width [ps]
0	29	10	45	2625
1	30	11	46	2625
2	30	11	46	2625
3	28	9	44	2625
4	30	11	46	2625
5	30	11	46	2625
6	30	11	46	2625
7	29	10	45	2625
8	29	10	44	2550
9	29	10	44	2550
10	30	11	45	2550
11	30	11	45	2550
12	31	11	46	2625
13	29	9	44	2625
14	30	10	45	2625
15	28	8	43	2625

Table A.4: Results from 400 Mbit s^{-1} per channel

Channel	Middle Tap	Lower Tap	Upper Tap	Data Eye width [ps]
0	23	9	34	1875
1	24	10	35	1875
2	24	10	35	1875
3	22	8	33	1875
4	23	10	34	1800
5	24	11	35	1800
6	24	10	35	1875
7	23	9	34	1875
8	23	10	33	1725
9	22	9	32	1725
10	24	11	34	1725
11	23	10	33	1725
12	24	10	35	1875
13	22	8	33	1875
14	23	10	34	1800
15	22	9	33	1800

Table A.5: Results from 600 Mbit s^{-1} per channel

Channel	Middle Tap	Lower Tap	Upper Tap	Data Eye width [ps]
0	18	10	22	900
1	18	10	22	900
2	18	11	23	900
3	17	10	22	900
4	19	12	22	750
5	18	11	21	750
6	17	10	21	825
7	17	10	21	825
8	17	9	22	975
9	17	9	22	975
10	18	10	22	900
11	18	10	22	900
12	17	10	21	825
13	17	10	21	825
14	18	11	23	900
15	16	9	21	900

Table A.6: Results from 800 Mbit s^{-1} per channel

Channel	Middle Tap	Lower Tap	Upper Tap	Data Eye width [ps]
0	14	10	17	525
1	14	10	17	525
2	15	12	18	450
3	13	10	16	450
4	14	11	17	450
5	15	12	18	450
6	15	12	18	450
7	13	10	16	450
8	14	11	17	450
9	13	10	16	450
10	15	11	18	525
11	15	11	18	525
12	14	11	18	525
13	13	10	17	525
14	16	13	18	375
15	13	10	15	375

Erklärung der Selbstständigkeit

Hiermit erkläre ich, dass ich die Diplomarbeit mit dem Titel

Development of a Cluster Control Board for the JEM-EUSO Mission

selbständig verfasst und dabei keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Ort, Datum

Jörg Bayer